DPSTM-500

J
O
S
H
U
A

THE JOSHUA SYSTEM

Vol. 5 BASIC DATA
PROCESSING

Approved by

H. C. Honeck, Research Manager
Computer Applications Division

NOTICE

~~This document contains descriptions of the JOSHUA Operating System and Application Modules. The document is prepared as a set of loose-leaf notebooks which will be periodically updated. All of the material is preliminary and should not be referenced.~~

May 1971

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Vol. 5
BASIC DATA
PROCESSING

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

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

DISCLAIMER

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

DISCLAIMER

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

JOSHUA DPSTM-500	Title Part A - Overview of Physics and Engineering Data Base and Data Processing	Vol. V	Page A.O.1
		Rev.	Date 5/71

Contents

- A.1 Overview of Neutronics Data Processing
- A.2 Overview of Photonics Data Processing (to be issued)
- A.3 Overview of Engineering Data Processing

JOSHUA DPSTM-500	Title Part A - Overview of Physics and Engineering Data Base and Data Processing	Vol. V	Page A.O.2
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title Overview of Neutronics Data Processing	Vol. V	Page A.1.1
		Rev.	Date 5/71

Report Written By

J. W. Stewart

Work Done By

J. W. Stewart

Contents

	<u>Page</u>
1. Description of the Neutronics Data Sets	A.1.3
2. Neutronics Data Processing Modules	A.1.5

JOSHUA DPSTM-500	Title Overview of Neutronics Data Processing	Vol.	Page
		V	A.1.2
		Rev.	Date
			5/71

(Blank)

JOSHUA DPSTM-500	Title Overview of Neutronics Data Processing	Vol. V	Page A.1.3
		Rev.	Date 5/71

1. Description of the Neutronics Data Sets

Neutronics data sets in the JOSHUA system are designed to provide as gross or as detailed a representation of a cross section as desired by a user. The neutronics data sets range from a pointwise representation to cell spectrum and composition-averaged data in few groups. Temperature dependences are accounted for throughout all neutronics data sets.

The neutronics data sets may loosely be thought of as falling into the following hierarchy.

- ENDFB Data Set - pointwise data in a format very similar to the standard ENDF/B format.
- FINEGRP Data Set - isotopic multigroup data in hundreds of energy groups.
- NUCPARAM Data Set - nuclear constants and parameters which are independent of energy group structure.
- MULTIGRP Data Set - isotopic multigroup data in tens of energy groups.
- FEWGRP Data Set - cell spectrum and composition-averaged multigroup data with less than ten energy groups.

Within each of these data sets several versions may be available, each corresponding to a different energy group structure and/or energy spectrum. Standard sets of data are available in each data set for general usage; special versions may be created at any time to contain special-purpose libraries.

In the above descriptions no essential distinction has been made between the FINEGRP and MULTIGRP data sets other than MULTIGRP has generally fewer energy groups. The FINEGRP data set is designed to accept as many or as few energy groups as is necessary to retain the integrity of the cross sections for a given energy spectrum. Since the need for energy spectrum detail decreases as group widths decrease (number of group increases) it is possible to get accurate cross sections in the FINEGRP data set by use of larger numbers of energy groups when no detailed energy spectrum is available. Alternatively, if a highly detailed energy spectrum is known, a fewer number of energy groups may be utilized to accurately represent the cross sections. The MULTIGRP data set assumes that a detailed energy spectrum has been used, and that the fewest possible energy groups are used that give an accurate representation of the cross sections. The data structure used in the MULTIGRP data set is also defined so as to be more easily utilized in repetitive production type calculations.

JOSHUA DPSTM-500	Title Overview of Neutronics Data Processing	Vol.	Page
		V	A.1.4
		Rev.	Date
			5/71

All neutronics data sets are capable of being expanded to include new types of data or new representations of data by including additional, uniquely-named data records.

A detailed description of the neutronics data sets is contained in Volume III of this manual.

JOSHUA DPSTM-500	Title Overview of Neutronics Data Processing	Vol. V	Page A.1.5
		Rev.	Date 5/71

2. Neutronics Data Processing Modules

The several neutronics data sets are interconnected by data processing or physics computational modules that allow energy spectrum, energy group structure, cell geometry and material composition to be properly accounted for in each data set. Figure 1 shows the relationship and flow of data between the data sets, and a set of computational modules for data processing.

Module JAMMED processes data from ENDF/B tape format (sequential files) to JOSHUA ENDFB data set format (direct access files with named data record cataloging).

Module ETØJ averages isotropic, pointwise neutronics data from the ENDFB data set into isotopic multigroup form in the FINEGRP and NUCPARAM data sets. Averaging is performed over any arbitrarily defined energy group structure and weighting spectrum, or over any energy group structure and weighting spectrum stored in the FINEGRP data set.

Module COFIN converts neutronics data from a FINEGRP representation to a MULTIGRP representation. Compression from one group structure to a second group structure (with a reduced number of energy groups) is accomplished by data weighting with an energy spectrum contained in the LATTICE data set.

Module PRESS compresses a MULTIGRP set of data from one energy structure to another via LATTICE spectrum weighting.

Module MULTIMAT produces MULTIGRP cross section sets for composite 'isotopes'.

The MULTIGRP and NUCPARAM data sets provide nuclear data input interface for a variety of lattice physics modules. The modules produce cell composition- and spectrum-averaged few group cross sections (FEWGRP), resonance integrals (RILIB), and spectra (LATTICE) data.

Module HØMØCELL homogenizes few group cell-averaged cross section sets into a supercell-averaged set in the FEWGRP data set.

In addition to the computational modules discussed above, a variety of maintenance modules are provided for data reporting and management, including the following:

JOSHUA DPSTM-500	Title Overview of Neutronics Data Processing	Vol. V	Page A.1.7
		Rev.	Date 5/71

FINED	-	FINEGRP edition.
JMPG1	-	Loads FINEGRP data from HAMMER cards.
CØMB	-	Combines FINEGRP data.
MULTED	-	MULTIGRP editor.
NUCED	-	NUCPARAM editor.
FEWED	-	FEWGRP editor.

Each of these modules is described in the following sections of this manual.

JOSHUA DPSTM-500	Title Overview of Neutronics Data Processing	Vol. V	Page A.1.8
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title Overview of Engineering Data Processing	Vol. V	Page A.3.1
		Rev.	Date 5/71

Report Written By

M. R. Buckner

Work Done By

M. M. Anderson
M. R. Buckner

Contents

	<u>Page</u>
1. Description of PROPERTY Data Set.	A.3.3
2. Description of Engineering Data Processing.	A.3.5

JOSHUA DPSTM-500	Title Overview of Engineering Data Processing	Vol. V	Page A.3.2
		Rev. .	Date 5/71

(Blank)

1. Description of PROPERTY Data Set

The PROPERTY data set contains engineering properties in the form of polynomial correlations in one or more engineering variables. Correlations depending on only one engineering variable are catalogued by property name. The correlation data is listed by material name within the single variable records. Multi-variable correlations are catalogued by both the property name and material name. The applicable range of the expansion variables over which the correlation applies and the units of both the property and expansion variables are also given in these records.

The materials included in the data set are those presently used in the reactor charge designs at the Savannah River Plant. Other materials will be added as a need is developed and additional data becomes available.

JOSHUA DPSTM-500	Title Overview of Engineering Data Processing	Vol. V	Page A.3.4
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title Overview of Engineering Data Processing	Vol. V	Page A.3.5
		Rev.	Date 5/71

2. Description of Engineering Data Processing

The property processing routine PRØPTY is designed to eliminate duplication of the programming effort required to process the correlation data in the PRØPERTY data set. This routine searches the PRØPERTY records for the proper correlation data for a material and evaluates the polynomial. In order to minimize disk accesses, the routine holds in core up to twenty single variable correlations and five multi-variable correlations. If additional correlations are required for a calculation, the correlation data is simply read in and processed. The only information required for a property evaluation is the property name, material name, PRØPERTY data set version name, and the value of the correlation expansion variables along with the units of each. The property value is returned in the units specified in Volume III of this document.

JOSHUA DPSTM-500	Title Overview of Engineering Data Processing	Vol. V	Page A.3.6
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title Part C - FINEGRP Data Processing	Vol.	Page
		V	C.O.1
		Rev.	Date
			5/71

Contents

- C.1 FINEGRP to MULTIGRP Processor - Module CØFIN.
- C.2 HAMMER to FINEGRP Processor - Module JMPG1.
- C.3 FINEGRP Editor - Module FINED
- C.4 FINEGRP Data Combinations - Module CØMB.

JOSHUA DPSTM-500	Title Part C - FINEGRP Data Processing	Vol. V	Page C.O.2
		Rev.	Date 5/71

(Blank)

Written By

W. V. Baxter
J. W. Stewart

Work Done By

W. V. Baxter
J. W. Stewart

Contents

	<u>Page</u>
1. <u>Theory</u>	C.1.3
1.1 Introduction	C.1.3
1.2 Cross Section Definitions	C.1.3
1.3 Temperature Interpolation	C.1.7
1.4 Compression	C.1.7
2. <u>Program Logic and Coding Details</u>	C.1.9
2.1 Overlay Structure and Computational Sequence.	C.1.9
2.2 Data Storage Allocation.	C.1.13
3. <u>User Information</u>	C.1.19
3.1 Input	C.1.19
3.2 Output	C.1.23
3.3 Module Execution	C.1.23
3.4 Error Stops	C.1.23

JOSHUA DPSTM-500	Title FINEGRP to MULTIGRP Processor - Module CØFIN.	Vol. V	Page C.1.2
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title FINEGRP to MULTIGRP Processor - Module CØFIN.	Vol.	Page
		V	C.1.3
		Rev.	Date
			5/71

1. Theory

1.1 Introduction

The purpose of the module CØFIN is to convert neutronics data from the FINEGRP structure to the MULTIGRP structure. This conversion may optionally include compression from one energy group structure to another reduced structure.

1.2 Cross Section Definitions

The cross sections catalogued in the FINEGRP and MULTIGRP data sets are specified in Volume 3 of this manual. The formulae used to translate data from the FINEGRP representation to the MULTIGRP representation are outlined in Table I. The notation used in the table is conventional. The symbol $\sigma(E)$ implies units of barns/eV and the symbol σ implies units of barns. Outscatter cross sections are catalogued with the scattering matrix elements.

TABLE I

Correspondence Between FINEGRP and MULTIGRP Cross Sections*

<u>MULTIGRP Data</u>		<u>FINEGRP Data</u>	
<u>Principal Cross Sections</u>			
Total	σ_t^g	$\sigma_t^g(E)$	ΔE^g
Capture	σ_c^g	$\sigma_c^g(E)$	ΔE^g
Fission	σ_f^g	$\sigma_f^g(E)$	ΔE^g
Nu	ν^g	ν^g	
Transport	σ_{tr}^g	$[\sigma_t^g(E) - \sigma_{el,l}^g(E) - \sigma_{in,l}^g(E)]$	ΔE^g
(n,2n)	σ_{n2n}^g	$\sigma_{n2n}^g(E)$	ΔE^g
Chi	χ^g	$\chi^g(E)$	ΔE^g
<u>Secondary Cross Sections</u>			
(n,v)	σ_{nv}^g	$\sigma_{nv}^g(E)$	ΔE^g
(n, α)	$\sigma_{n\alpha}^g$	$\sigma_{n\alpha}^g(E)$	ΔE^g
(n,p)	σ_{np}^g	$\sigma_{np}^g(E)$	ΔE^g
(n,d)	σ_{nd}^g	$\sigma_{nd}^g(E)$	ΔE^g
(n,t)	σ_{nt}^g	$\sigma_{nt}^g(E)$	ΔE^g
(n, ν^*)	$\sigma_{n\nu^*}^g$	$\sigma_{n\nu^*}^g(E)$	ΔE^g
<u>Delayed Neutron Data</u>			
Prompt $\nu\sigma_f$	$(\nu\sigma_f)_p^g$	$[\nu^g - \sum_j \nu_{Dj}^g] \sigma_f^g(E)$	ΔE^g
Delayed Chi	χ_{Dj}^g	$\chi_{Dj}^g(E)$	ΔE^g

TABLE I (Cont.)

<u>MULTIGRP Data</u>		<u>FINEGRP Data</u>	
<u>Scattering Matrices</u>			
Elastic	$\sigma_{el,l}^{g' \rightarrow g}$	$\sigma_{el,l}^{g' \rightarrow g} (E)$	ΔE^g
Inelastic	$\sigma_{in,l}^{g' \rightarrow g}$	$\sigma_{in,l}^{g' \rightarrow g} (E)$	ΔE^g
(n,2n)	$\sigma_{n2n,l}^{g' \rightarrow g}$	$\sigma_{n2n,l}^{g' \rightarrow g} (E)$	ΔE^g
Total	$\sigma_{tot,l}^{g' \rightarrow g}$	$\sigma_{tot,l}^{g' \rightarrow g} (E)$	ΔE^g
<u>Outscatter Cross Sections</u>			
(Elastic, Inelastic, n2n, and Total)			
$\sigma_{s,l}^g$		$\sum_{g'} \sigma_{s,l}^{g \rightarrow g'} (E)$	$\Delta E^{g'}$

* g denotes energy group index; l denotes Legendre order.

JOSHUA DPSTM-500	Title FINEGRP to MULTIGRP Processor - Module CØFIN.	Vol. V	Page C.1.6
		Rev.	Date 5/71

(Blank)

1.3 Temperature Interpolation

The temperature dependence of neutronics data is catalogued in the FINEGRP data set. Module CØFIN processes this data to obtain MULTIGRP data at a particular specified temperature. FINEGRP data may be stored for a set of temperatures. The following conventions are employed in processing this temperature dependent data:

- Data stored at only zero °K are assumed to be temperature independent.
- Interpolation between two non-zero catalogued temperatures is performed assuming

$$\ln \sigma(T) = \frac{a}{T} + b$$

- Interpolation between two temperatures, one of which is zero °K, is performed assuming

$$\sigma(T) = \frac{c}{T} + d$$

Module CØFIN will not extrapolate the catalogued temperature dependence. If the temperature at which the MULTIGRP value is desired is not between catalogued FINEGRP temperatures a warning message is written and the isotope is not processed.

1.4 Compression

A multigroup Legendre flux spectrum is used to weight neutronics data if a FINEGRP set of data with a given energy group structure is to be translated to a MULTIGRP representation with a reduced energy group structure. With three exceptions, the following formulae are used to compress cross section and scattering matrix data:

$$\sigma_l^g = \sum_{i \in g} \sigma_l^i \phi_l^i / \bar{\phi}_l^g$$

$$\sigma_{l \rightarrow g}^{g' \rightarrow g} = \sum_{i \in g} \sum_{j \in g'} \sigma_{l \rightarrow j}^{i \rightarrow j} \phi_l^i / \bar{\phi}_l^g$$

where ϕ_l^i is the flux in group i , Legendre order l , " $i \in g$ " means all groups i contained in group g , and

$$\bar{\phi}_l^g = \sum_{i \in g} \phi_l^i$$

The three exceptions to the above formulae are:

$$\chi^g = \sum_{i \in g} \chi^i$$

$$\chi_{Dj}^g = \sum_{i \in g} \chi_{Dj}^i$$

$$\nu^g = \frac{\sum_{i \in g} \nu^i \sigma_f^i \phi_0^i}{\sum_{i \in g} \sigma_f^i \phi_0^i}$$

Implicit in compression is the requirement that full scattering matrices must be catalogued for the compressed energy group structure as detailed balance is inapplicable.

If only down-scatter elements are catalogued, the up-scatter elements to thermal groups (only) are calculated from detailed balance:

$$\sigma_{(E)}^{g' \rightarrow g} M(E_{g'}) = \sigma_{(E)}^{g \rightarrow g'} M(E_g)$$

where $M(E)$ is the Maxwellian distribution; or,

$$\sigma_{(E)}^{g' \rightarrow g} = \sigma_{(E)}^{g \rightarrow g'} \frac{\bar{E}^g}{\bar{E}^{g'}} \exp \left\{ - \frac{1}{KT} [\bar{E}^g - \bar{E}^{g'}] \right\}$$

where $\bar{E}^g$ is the characteristic energy for group g , K is Boltzmann's constant (8.6166×10^{-5}), and T is absolute temperature ($^{\circ}K = ^{\circ}C + 273.18$). Note that the expression of detailed balance given above applies to scattering elements in units of barns/eV only.

2. Program Logic and Coding Details

2.1 Overlay Structure and Computational Sequence

The CØFIN module is overlaid to conserve machine core requirements. The overlay structure is shown in Figure 1. CØFIN calculations are basically one of two types, namely: 1) translation of data from a FINEGRP format to the MULTIGRP format without compression of the energy group structure, and 2) compression of data from a FINEGRP format to a reduced energy-structure MULTIGRP format. In either type calculation a printed edit is provided. The overlay structure reflects these options.

The root segment of CØFIN is composed of a MAIN subroutine, JOS utility routines, IBM utility routines, and the SRL.FØRTLIB utility routine CLEAR. The MAIN subroutine determines (from problem input data) if simple translation or compression is required. If the former is required a call is issued to subroutine SETUP; if the latter is required a call is issued to SETUPP. Either SETUP or SETUPP initiates a series of calls to routines which process the neutronics data. Control is then returned to MAIN which calls MULTED to produce a printed edit and the execution terminates.

The sequence of calculations is shown in Figure 2. The purpose of each routine is tabulated in Table II.

Figure 1. Overlay Structure in Module CØFIN

* Named CØMMØN blocks. All other references are to subroutine entry points.

Figure 2. Sequence of Calculations

JOSHUA DPSTM-500	Title FINEGRP to MULTIGRP Processor - Module CØFIN.	Vol.	Page
		V	C.1.12
		Rev.	Date
			5/71

TABLE II

Purpose of Routines in Module CØFIN

<u>Routine Name(s)</u>	<u>Purpose</u>
MAIN	Reads some input data. Calls either SETUP (for translation) or SETUPP (for compression). On return, calls MULTED to produce edit. Issues "normal termination" message and terminates.
SETUP,SETUPP	Calculates origin of variably-dimensioned arrays within CØMMØN/DATARA/ or CØMMØN/DATARP/. Determines if enough storage has been allocated. Calls CØMPUT or CØMPUP.
CØMPUT,CØMPUP	Reads input data. Processes GRØUPS and CØNTENTS records. Issues a series of calls to routines which process various types of data. Routine CØMPUP also issues call to FREAD.
FREAD	Reads weighting spectrum data and computes sums of weights over reduced group structure. Writes MULTIGRP 'WEIGHTS' record.
PRIN,PRIP	Processes principal cross section data and computes limits of nonzero data.
SECØN,SECØP	Processes secondary cross section data and computes limits of nonzero data.
DELAY,DELAP	Processes delayed neutron data and computes limits of nonzero data.
TMAT,TMAP	Processes scattering matrices for elastic, inelastic, or (n,2n) reactions and computes limits of nonzero data.
TØTMAT,TØTMAP	Processes total scattering matrix data and computes limits of nonzero data.
TINT,TINP	Performs temperature interpolation of neutronics data.
MULTED	Reads edit options and controls editing calls to routine GUTS, isotope by isotope.
GUTS	Produces printed edit for an isotope.

JOSHUA DPSTM-500	Title FINEGRP to MULTIGRP Processor - Module CØFIN.	Vol. V	Page C.1.13
		Rev.	Date 5/71

2.2 Data Storage Allocation

Eleven named CØMMØN blocks are used for data storage. The structure of each of these blocks is described in Table III.

The large working arrays required during the neutronics data processing are assigned to CØMMØN block DATARA (for translation) or DATARP (for compression). The structures of these blocks are similar and only the CØMMØN/DATARA/ will be discussed here. The block DATARA contains a single REAL*4 array DAT. In all the computational routines used in translation the array is dimensioned DAT(1), except in SUBROUTINE SETUP, where it is DAT(25000). The purpose of SETUP is to compute the origins of every working array contained in CØMMØN/DATARA/ in the subsequently called translation routines. These origins are stored as KK1, KK2,... in CØMMØN/DIMNS/. In addition, maximum dimensions are specified for such things as number of MULTIGRP versions, FINEGRP versions, energy groups, temperature, Legendre order, delayed neutron families, etc. If sufficient core space is not available SETUP branches to an appropriate error STØP. The maximum problem size treated by CØFIN can be increased by making, at most, 13 minor changes in SETUP (or SETUPP for compression problems).

Special techniques are employed in the storage of scattering matrices in the routines TMAT, TØTMAT, TMAP, and TØTMAP. In TMAT and TØTMAT, which process scattering matrix data for translation, only down-scatter matrix elements are catalogued. The down-scatter elements are stored in an array dimensioned

$$\text{REAL*4 TM}(\text{MAXDTM}, \text{MAXDL})$$

where MAXDL is the maximum dimension for the number of Legendre indices and MAXDTM is expressed in terms of MAXDG, the maximum dimensions for the number of energy groups, as

$$\text{MAXDTM} = \text{MAXDG} * (\text{MAXDG} + 1) / 2$$

The element $\sigma_L^{I \rightarrow J}$ for scattering from group I to group J for Legendre order L is stored in

$$\text{TM}(\text{N}, \text{LX})$$

$$\text{N} = \text{J} * (\text{J} - 1) / 2 + \text{I}$$

$$\text{LX} = \text{L} + 1$$

Since only down-scatter elements are stored the above holds only for $\text{J} \geq \text{I}$.

TABLE III

Contents of Named CØMMØN Blocks in Module CØFIN

<u>Variable Name and Dimension</u>	<u>Type</u>	<u>Description</u>
<u>CØMMØN/NAMP/</u>	(Used only in compression.)	
VERF	R8	FINEGRP version name.
VERM	R8	MULTIGRP version name.
KDEL	I4	Delayed neutron data processing flag. <div style="margin-left: 100px;"> { 1 = Process 0 = Don't Process </div>
KSEL	I4	Secondary neutron data processing flag.
KEL	I4	Elastic scattering data processing flag.
KIN	I4	Inelastic scattering data processing flag.
KN2N	I4	(n,2n) scattering data processing flag.
KTØT	I4	Total scattering data processing flag.
<u>CØMMØN/PARP/</u>	(Used only in compression.)	
DISØ	R8	Isotope name.
KER	I4	Processing error flag.
ITDC	I4	Temperature (°C).
IDBF	I4	Detailed balance flag.
JFTG	I4	Index of first thermal group in FINEGRP.
NEG	I4	Number of groups in FINEGRP.
JFTGN	I4	Index of first thermal group in MULTIGRP.
NEGN	I4	Number of groups in MULTIGRP.
NEGTN	I4	Number of thermal groups in MULTIGRP.
<u>CØMMØN/DATARP/</u>	(Used only in compression.)	
DAT (25000)	R4	Working array composed of a variety of variably dimensioned arrays within each subroutine.

TABLE III (Cont.)

Contents of Named CØMMØN Blocks in Module CØFIN

<u>Variable Name and Dimension</u>	<u>Type</u>	<u>Description</u>
<u>CØMMØN/NAME/</u>	(Used only in translation.)	
Same as CØMMØN/NAMP/		
<u>CØMMØN/PARM/</u>	(Used only in translation.)	
DISØ	R8	Isotope name.
KER	I4	Processing error flag.
ITDC	I4	Temperature (°C).
IDBF	I4	Detailed balance flag.
JFTG	I4	Index of first thermal group.
NEG	I4	Number of groups.
<u>CØMMØN/CØNT/</u>	(Used only in translation.)	
Same as CØMMØN/CØNP/		
<u>CØMMØN/CØNP/</u>	(Used only in compression.)	
IZAS	I4	Isotope identification number.
AWR	R4	Atomic weight ratio.
LIM(15)	I4	Working array for nonzero data group limits.
<u>CØMMØN/DIMNP/</u>	(Used only in compression.)	
KK1	I4	Origin of first array in CØMMØN/DATARP/.
...		
KK51	I4	Origin of 51st array in CØMMØN/DATARP/.
MAXDV	I4	Maximum dimension for number of MULTIGRP versions.
MAXDIF	I4	Maximum dimension for number of FINEGRP isotopes.
MAXDIM	I4	Maximum dimension for number of MULTIGRP isotopes.
MAXDG	I4	Maximum dimension for number of FINEGRP groups.

TABLE III (Cont.)

Contents of Named CØMMØN Blocks in Module CØFIN

<u>Variable Name and Dimension</u>	<u>Type</u>	<u>Description</u>
<u>CØMMØN/DIMNP/</u> (Cont.)		
MAXDGL	I4	MAXDG + 1
MAXDT	I4	Maximum dimension for number of temperatures in FINEGRP.
MAXDL	I4	Maximum dimension for number of Legendre indices.
MAXDF	I4	Maximum dimension for number of delayed families.
MAXDLT	I4	Max. (MAXDL,MAXDT).
MAXDGL	I4	Max. (MAXDL,MAXDG).
MAXDGN	I4	Max. (MAXDL,MAXDG) MULTIGRP groups.
MAXGNL	I4	MAXDGN + 1
MAXGTN	I4	MAXDGN + 1 MULTIGRP thermal groups.
MAXDTF	I4	MAXDGN + 1 fast transfer matrix.
KK130	I4	Origin of an array in CØMMØN/DATARP/.
<u>CØMMØN/DIMNS/</u> (Used only in translation.)		
KK1	I4	Origin of 1st array in CØMMØN/DATARA/.
...		
KK41	I4	Origin of 41st array in CØMMØN/DATARA/.
MAXDV	I4	Maximum dimensions for number of MULTIGRP versions.
MAXDIF	I4	Maximum dimension for number of FINEGRP isotopes.
MAXDIM	I4	Maximum dimension for number of MULTIGRP isotopes.
MAXDG	I4	Maximum dimension for number of groups.
MAXDGL	I4	Maximum dimension for number of groups Plus one.
MAXDT	I4	Maximum dimension for number of temperatures in FINEGRP.

TABLE III (Cont.)

Contents of Named CØMMØN Blocks in Module CØFIN

<u>Variable Name and Dimension</u>	<u>Type</u>	<u>Description</u>
<u>CØMMØN/DIMNS/</u>		
MAXDL	I4	Maximum dimension for number of Legendre indices.
MAXDF	I4	Maximum dimension for number of delayed families.
MAXDIM	I4	Maximum dimension for transfer matrix array.
MAXDLT	I4	Max. (MAXDL,MAXDT)
MAXDGL	I4	Max. (MAXDL,MAXDG)
KK44	I4	Origin of an array in CØMMØN/DATARA/
<u>CØMMØN/DATARA/</u> (Used only in translation.)		
DAT(25000)	R4	Working array composed of a variety of variably dimensioned arrays within each subroutine.
<u>CØMMØN/GRØUPS/</u>		
E(85)	R4	Energy group boundaries (eV). E(1) is the upper limit for group 1.
U(84)	R4	Lower lethargy boundary by group.
RV(84)	R4	Average reciprocal velocity by group.
EBAR(84)	R4	Characteristic (average) energy by group.
NEG	I4	Number of energy groups.
JFTG	I4	Index of first thermal group.
IDBF	I4	Detailed balance flag. IDBF = 1 means only down-scatter elements are stored. IDBF = 0 means full matrix is stored.

In TMAP and TØTMAP, which process scattering matrix data for compression, both down-scatter and up-scatter elements are stored for thermal final energy groups and only down-scatter elements are stored for fast final energy groups. The arrays are dimensioned

REAL*4 TMF(MAXDTF,MAXDL)

REAL*4 TMT(MAXDGN,MAXGTN,MAXDL)

for fast and thermal final energy groups, respectively. The dimensions are

MAXDTF = Maximum dimension for number of fast energy groups.

MAXDGN = Maximum dimension for number of energy groups.

MAXGTN = Maximum dimension for number of thermal energy groups.

MAXDL = Maximum dimension for number of Legendre indices.

Hence, the element $\sigma_{I \rightarrow J}^I$ for scattering into fast final group J from group I, Legendre order L is stored in

TMF(N,LX)

N = J*(J-1)/2 + I

LX = L + 1

and the element $\sigma_{I \rightarrow J}^I$ for scattering into thermal final group J from group I, Legendre order L is stored in

TMAT (IP,JP,LX)

IP = I

JP = J - JFTGN + 1

LX = L + 1

where JFTGN is the index of the first thermal group.

JOSHUA DPSTM-500	Title FINEGRP to MULTIGRP Processor - Module CØFIN.	Vol. V	Page C.1.19
		Rev.	Date 5/71

3. User Information

3.1 Input

One input data record is required when module CØFIN is executed. This record is of the type

INPUT.CØFIN.CØNTRØL.?\$JØB

where the variable qualifier \$JØB is the JOSHUA job name established at execution time. This record contains the FINEGRP and MULTIGRP version names and information specifying whether or not energy group structure compression is required. The user may select which data types are to be processed and specify up to 21 isotopes to be deleted and up to 21 isotopes to be added.

If energy group structure compression is specified in the CØNTRØL record an additional input record is required of the type

INPUT.CØFIN.CØMPRESS.?\$JØB

This record contains record qualifiers in the LATTICE record in which the Legendre flux - weighting spectrum is stored. This record is of the type

LATTICE.?DLAT.?VLAT.LFLUX.?NCASE .

In addition the reduced MULTIGRP energy structure is related to the FINEGRP energy structure by specification of

- Number of MULTIGRP energy groups.
- Index of the first thermal MULTIGRP energy group (≤ 31).
- Index of the most energetic FINEGRP group for each of the MULTIGRP groups.

A sample of these two records is given on the following pages.

3401.INPUT.CØFIN.CONTROL.SAMPLE

ISOTOPE		TEMP.	ISOTOPE		TEMP.	ISOTOPE		TEMP. (OC)
:D-D2O	:	20	:	:	0	:	:	0
:	:	0	:	:	0	:	:	0
:	:	0	:	:	0	:	:	0
:	:	0	:	:	0	:	:	0
:	:	0	:	:	0	:	:	0
:	:	0	:	:	0	:	:	0
:	:	0	:	:	0	:	:	0

3401.INPUT.CØFIN.COMPRESS.SAMPLE

LATTICE DATA SET NAME:LATTNAME

VERSION OF LATTICE DATA SET:LATTVERS

CASE NO.: 1

NO. OF GROUPS IN NEW STRUCTURE: 42

INDEX OF NEW FIRST THERMAL GROUP: 28

JOSHUA DPSTM-500	Title FINEGRP to MULTIGRP Processor - Module CØFIN.	Vol.	Page
		V	C.1.22
		Rev.	Date
			5/71

3401.INPUT.CØFIN.COMPRESS.SAMPLE

GROUP STRUCTURE INDICES ***

NEW	OLD	NEW	OLD	NEW	OLD	NEW	OLD	NEW	OLD	NEW	OLD	NEW	OLD
: 1:	1	: 2:	3	: 3:	5	: 4:	7	: 5:	9	: 6:	11	: 7:	13
: 8:	15	: 9:	17	:10:	19	:11:	21	:12:	23	:13:	25	:14:	27
:15:	29	:16:	31	:17:	33	:18:	35	:19:	37	:20:	39	:21:	41
:22:	43	:23:	45	:24:	47	:25:	49	:26:	51	:27:	53	:28:	55
:29:	57	:30:	59	:31:	61	:32:	63	:33:	65	:34:	67	:35:	69
:36:	71	:37:	73	:38:	75	:39:	77	:40:	79	:41:	81	:42:	83

3.2 Output

Appropriate self-explanatory warning messages are printed during data processing to indicate missing FINEGRP data records. Messages also indicate completion of data processing for each data type by isotope.

After completion of the data processing for all isotopes a printed edit is produced for all isotopes added.

3.3 Module Execution

Module CØFIN cannot be executed from the JOSHUA terminals since the FINEGRP data set is normally off-line. A sample JCL setup is shown below:

```
//WVBA1326 JOB (1058,L023,2,2,0,,,8690-00,L5365-01,P,01),
//          'W.V.BAXTER',MSGLEVEL=1,CLASS=M
//*          *****
/**          * USES A11133 PACK          *
/**          *****
//STEP1 EXEC JØSHUA,GOSIZE=240K,SPACE=5,GTIME=2,
//          JØB=UPDATE,BADGE=1058,MEMBER=CØFIN
//GØ.FT22FO01 DD DSNAME=JØS.JØSHUA.FINEGRP,DISP=SHR
//GØ.SYSIN DD *
DELETE=ØLDJØB

/*
```

The number of blocks (SPACE parameter) required in the job data set is strongly dependent on the number of isotopes and size of the scattering matrices processed.

3.4 Error Stops

An extensive set of error warnings and stops is programmed. The printed messages are self-explanatory.

JOSHUA

DPSTM-500

Title HAMMER to FINEGRP Processor - Module
JMPG1.

Vol.
V.

Page
C.2.1

Rev.

Date
5/71

Written By

W. V. Baxter
J. W. Stewart

Work Done By

W. V. Baxter
H. C. Honeck
R. A. Priest

Contents

	<u>Page</u>
1. <u>Theory</u>	C.2.3
1.1 Introduction	C.2.3
1.2 Neutronics Data Representation	C.2.3
2. <u>Program Logic and Coding Details</u>	C.2.4
2.1 Program Structure	C.2.4
2.2 Data Layout	C.2.9
3. <u>User Information</u>	C.2.10
3.1 Input	C.2.10
3.2 Output	C.2.20
3.3 Module Execution	C.2.20
3.4 Error Stops	C.2.20

JOSHUA DPSTM-500	Title HAMMER to FINEGRP Processor - Module JMPG1.	Vol. V	Page C.2.2
		Rev.	Date 5/71

(Blank)

1. Theory

1.1 Introduction

The purpose of JMPG1 is to process neutronics data (cross sections, scattering matrices, resonance data) from HAMMER cross section decks and to write the NUCPARAM and FINEGRP data sets. The energy group structure is restricted to the 84-group HAMMER library structure. Data for isotopes may be deleted, added, or modified using JMPG1.

1.2 Neutronics Data Representation

The FINEGRP and NUCPARAM data sets are described in Volume III of this document. The structure of HAMMER neutronics data representation is described in the reference: Suich, J. E. and H. C. Honeck, The HAMMER System, DP-1064, pages 61 and 64.

JOSHUA DPSTM-500	Title HAMMER to FINEGRP Processor - Module JMPG1.	Vol. V	Page C.2.4
		Rev.	Date 5/71

2. Program Logic and Coding Details

2.1 Program Structure

Module JMPG1 is composed of a main subroutine, ten other subroutines, and one function subprogram. These routines are described briefly below:

<u>MAIN</u>	-	This subroutine controls the sequence of calculations. MAIN calls HELP1, ELAST, TLØØP, and JSTØP.
<u>HELP1</u>	-	This subroutine is called from MAIN to process epithermal data for a single isotope.
<u>ELAST</u>	-	This subroutine is called from MAIN to compute elastic transfer matrix elements. ELAST calls ELF2.
<u>TLØØP</u>	-	This subroutine is called from MAIN to control the process of thermal data for one isotope and all temperatures. TLØØP calls LITHE1.
<u>JSTØP</u>	-	This subroutine is called from MAIN to produce printed error messages.
<u>ELF2</u>	-	ELF2 is an operational subroutine called from ELAST to generate elements of the elastic transfer matrix. ELF2 calls ELF1.
<u>ELF1</u>	-	This routine performs a numerical integration required in the generation of the elastic transfer matrix. ELF1 is called from ELF2.
<u>LITHE1</u>	-	This subroutine is called from TLØØP to read thermal data and control thermal data processing for one isotope at a given temperature. LITHE1 calls BASK and KERIN.
<u>BASK</u>	-	This subroutine is called from LITHE1 to process thermal cross section data for one isotope at one temperature. BASK uses the function ERF.

- KERIN - KERIN is called from LITHE1 to process thermal scattering matrix data for one isotope at one temperature. KERIN calls LDC.
- LDC - This routine is called from KERIN to read thermal transfer data.
- ERF - This function subprogram is called from BASK to evaluate the erf-function.

The relationship between these routines is shown in Figure 1. Figure 2 is a simplified flow diagram of subroutine MAIN illustrating the sequence of data processing. In Figure 2 the following notation is used:

'FINEGRP' → F
'NUCPARAM' → N
F.?FVERSN → FV
N.?NVERSN → NV
FV.?ISØ → FVI
NV.?ISØ → NVI

Figure 1. Interrelation of Subroutines in Module JMPG1

Figure 2. Simplified Flow Diagram of MAIN in Module JMPG1

Figure 2 (Continued)

JOSHUA DPSTM-500	Title HAMMER to FINEGRP Process - Module JMPG1.	Vol.	Page
		V	C.2.9
		Rev.	Date
			5/71

2.2 Data Storage

The bulk of the data storage used by module JMPG1 is contained in the seven named common blocks listed below:

CHI
TAPES
CØNT
HELP
LITHE
TEMPER
CØNT1

A detailed description of the data arrays is not given here due to the restricted application of JMPG1.

JOSHUA DPSTM-500	Title HAMMER to FINEGRP Process - Module JMPG1.	Vol.	Page
		V	C.2.10
		Rev.	Date
			5/71

3. User Information

3.1 Input

JMPG1 input data is read from cards. The data deck is structured as shown in Figure 1. The description of each card type is outlined in Table I.

Each data deck begins with a problem specification card. If isotopes are to be deleted the names of these isotopes appear on card type 2. If a new data set version is to be generated the energy group structure is specified on card types 3-5. Batches of neutronics data follow, isotope by isotope. Each batch of isotopic neutronics data is headed by an isotope control card (type 6), followed immediately by epithermal data on cards type 7. Thermal neutronics data is specified on cards type 8, temperature by temperature. The deck ends with a blank card.

JOSHUA DPSTM-500	Title HAMMER to FINEGRP Process - Module JMPG1.	Vol. V	Page C.2.11
		Rev.	Date 5/71

Figure 1. Structure of JMPG1 Data Deck.

Table I. JMPG1 Data Card Description

<u>Variable</u>	<u>Column</u>	<u>Description</u>
-----------------	---------------	--------------------

Card 1: Problem Specification Card (A8, 2X, 2I5)

VERSN	1-8	Eight-character FINEGRP and NUCPARAM data sets version name.
ITAG	11-15	Problem type ITAG = -1 Delete current data sets version and rebuild them. ITAG = 0 Create a new version. ITAG = 1 Update existing ver- sion.
NISD	16-20	Number of isotopes to be deleted. (NISD is ignored if ITAG $\neq$ 1.)

Card 2: Deleted Isotope Names (7 (A8, 2X))
Not required if ITAG $\neq$ 1 or NISD $\leq$ 0

DISØ(1)	1-8	Name of first deleted isotope.
DISØ(2)	11-18	Name of second deleted isotope.
etc.		

Card 3: Epithermal Lethargy Mesh (7E10.5)
Required only if ITAG = 0, Card 1.

DU(1)	1-10	Lethargy width for epithermal group 1.
DU(2)	11-20	Lethargy width for epithermal group 2.
...		
DU(54)	-	Lethargy width for epithermal group 54.

Variable

Column

Description

Card 4: Thermal Velocity Mesh (7E10.5)
Required only if ITAG = 0, Card 1.

V(1)	1-10	Thermal group 1 velocity.
V(2)	11-20	Thermal group 2 velocity.
...		
V(30)	-	Thermal group 30 velocity.

Card 5: Thermal Velocity Increments (7E10.5)
Required only if ITAG = 0, Card 1.

DV(1)	1-10	Velocity increment, thermal group 1.
DV(2)	11-20	Velocity increment, thermal group 2.
...		
DV(30)	-	Velocity increment, thermal group 30.

Card 6: Isotope Control Card (A8, 2X, 2I5)

ISØNAM	1-8	Eight-character isotope name.
LX	11-15	Elastic scattering matrix flag LX = 0 no elastic matrices LX = 2 generate P ₀ and P ₁ elastic scattering matrices.
KT	16-20	Number of temperatures for which thermal data are given.

<u>Variable</u>	<u>Column</u>	<u>Description</u>
Card 7: Epithermal Data		
Card 7.0: Header Card (F7.0, 1X, E8.5, 2X, 14A4)		
WDI	1-7	Isotope ID number.
AMAS	9-16	Atomic mass.
HØL(9)	19-72	Chemical symbol and Hollerith text.

Card 7.1: Smooth Data (I1, 2I3, 1X, 8E8.6)

One card per significant lethargy group.

I	1	Card type, always 1.
J	2-4	Group number (sequential).
K	5-7	Not used.
SMX(1,J)	9-16	P_0 component of smooth scattering.
SMX(2,J)	17-24	Smooth capture cross section.
SMX(3,J)	25-32	Inelastic cross section.
SMX(4,J)	33-40	Smooth fission cross section.
SMX(5,J)	41-48	Age number.
SMX(6,J)	49-56	P_1 component of smooth scattering.
SMX(7,J)	57-64	Neutrons per fission
SMX(8,J)	65-72	Microscopic slowing-down power.

<u>Variable</u>	<u>Column</u>	<u>Description</u>
Card 7.2 Inelastic Scattering Matrix (11, 213, 1X, 8E8.6) Use as required.		
I	1	Card type, always 2.
J	2-4	Final group index.
K	5-7	Initial group index of first entry.
(XIN(L,J), L = K,K+7)		Matrix elements from higher groups into group J, given sequentially beginning with group K. Ignore all fields after element J=L.

Card 7.3 Resonance Parameters (11, 213, 1X, 8E8.6)
Use as required.

I	1	Card type, always 3.
J	2-4	Group in which resonance occurs.
K	5-7	Sequential number of this resonance within this group.

(Data to be treated by the Nordheim method:)

ER	9-16	Resonance energy.
GMN	17-24	Neutron width.
GMGM	25-32	Radiation width.
GMF	33-40	Fission width.
SPIN	41-48	Spin factor.

(Data to be treated by the Equivalence Principle:)

R	9-16	Shielding parameter.
M	17-24	Infinite integral.
ALPHA	25-32	Fission to absorption ratio.

Variable	Column	Description
Card 7.4 Unresolved Resonance Data (11, 213, 1X, 8E8.6) Use as required.		
I	1	Card type, always 4.
J	2-4	Initial group in which resonances occur.
K	5-7	Final group in which resonances occur.
ER	9-16	Average level spacing.
GMN	17-24	Average reduced neutron width.
GMGM	25-32	Average reduced radiation width.
GMF	33-40	Average reduced fission width.
SPIN	41-48	Spin factor.

Card 7.9 End of Epithermal Data (11)
Used at end of epithermal data cards.

I	1	Card type, always 9.
---	---	----------------------

Cards 8: Thermal Data.
Use one set of cards 8 for each temperature.

Card 8.0 Header Card (3A6, 2F10.0, 5I5)

HØLB(18)	1-18	18-character description of isotope.
WA	21-30	Isotope ID number.
WB	31-40	Temperature/Model Index of the form tttm where ttt is the absolute temperature divided by 10 and rounded and m is the model number: m = 0 free gas m = 1 Nelkin, Parks m = 2 Brown - St. John m = 3 Ardente benzene

<u>Variable</u>	<u>Column</u>	<u>Description</u>
<u>Card 8.0 (Cont.)</u>		
LXA	41-45	Absorption cross section flag. LXA = 0 $\sigma_a(v) = VXA$ on card 8.1 LXA = 1 $\sigma_a(v) = VXA/v$ on card 8.1 LXA = 2 $\sigma_a(v)$ tabulated
LXS	46-50	Scattering flag. LXS = 0 $\sigma_s(v) = VXS$ on card 8.1 LXS = 1 $\sigma_s(v)$ computed via gas model LXS = 2 $\sigma_s(v)$ computed via input kernel.
LXF	51-55	Fission flag. LXF = 0 $\sigma_f(v) = VXF$ on card 8.1 LXF = 1 $\sigma_f(v) = VXF/v$ on card 8.1 LXF = 2 $\sigma_f(v)$ tabulated
NKERN	56-60	Number of kernels to be tabulated.
NTERM	61-65	Number of terms in Brown - St.John formula.

Card 8.1 Cross Section Data (7E10.5)

VXA	1-10	Value of σ_a .
VXS	11-20	Value of σ_s .
VXF	21-30	Value of σ_f .
VNU	31-40	Value of v .
AM	41-50	Mass used in source calculation.
HXS	51-60	Cross section used in source calculation.
TP	61-70	Temperature used in source calculation.

<u>Variable</u>	<u>Column</u>	<u>Description</u>
-----------------	---------------	--------------------

Card 8.2 Tabulated Absorption Cross Sections (7E10.5)
Omit if LXA $\neq$ 2, Card 8.0

XA(1)	1-10	Tabulated value of $\sigma_a(v_1)$.
XA(2)	11-20	Tabulated value of $\sigma_a(v_2)$.
...		
XA(30)	-	Tabulated value of $\sigma_a(v_{30})$.

Card 8.3 Tabulated Fission Cross Sections (7E10.5)
Omit if LXF $\neq$ 2, Card 8.0

XF(1)	1-10	Tabulated value of $\sigma_f(v_1)$.
XF(2)	11-20	Tabulated value of $\sigma_f(v_2)$.
...		
XF(30)	-	Tabulated value of $\sigma_f(v_{30})$.

Card 8.4 Brown - St. John Constants (7E10.5)
Omit if LXS $\neq$ 1, Card 8.0

T	1-10	Temperature ($^{\circ}$ C) used in kernel calculation.
(ZAMT(N),	-	Mass number, term N, in kernel calc.
ZAT(N),	-	Coefficient, term N, in kernel calc.
ZAKT(N),	-	Exponent, term N, in kernel calc.
N=1, NTERM)		

JOSHUA DPSTM-500	Title HAMMER to FINEGRP Process - Module JMPG1.	Vol.	Page
		V	C.2.19
		Rev.	Date
			5/71

<u>Variable</u>	<u>Column</u>	<u>Description</u>
Card 8.5 Kernel Data (I1, 2I3, 1X, 5E12.4, F8.0, I4) Omit if LXS $\neq$ 2, Card 8.0		
N	1	Legendre index ($N \leq 30$).
I	2-4	Initial energy group index.
J	5-7	Final energy group index for value A(1).
A(1)	9-20	Value of $P_n(I, J)$.
A(2)	21-32	Value of $P_n(I, J+1)$.
A(3)	33-44	Value of $P_n(I, J+2)$.
A(4)	45-56	Value of $P_n(I, J+3)$.
A(5)	57-68	Value of $P_n(I, J+4)$.
IA	69-76	Isotope ID number.
NSEQ	77-80	Kernel data card sequence number.

Card 9 End of Deck

Blank card.

3.2 Output

Printed output from JMPG1 includes a summary of input data and processed nuclear data. All appropriate FINEGRP and NUCPARAM records are written to the job data set.

Data Processing exceptions may result in printed warning messages which should be carefully analyzed by the user.

3.3 Module Execution

Module JMPG1 must be executed in the batch mode since the FINEGRP data set is normally kept off-line (i.e., the disk pack is mounted only when requested). Sample JCL setup is illustrated below:

```
//AAAAIIIII JØB(1234,.....,01),  
//          'your name',MSGLEVEL=1,CLASS=M  
//STEP1 EXEC JØSHUA,BADGE=1234,SPACE=3,  
// MEMBER=JMPG1,JOB=jobname,GØSIZE=240K,GTIME=1  
//GØ.FT33FOO1 DD DSN=JØS.JØSHUA.FINEGRP,DISP=SHR  
//Gd.SYSIN DD *  
JØS job delete cards followed by a blank card  
JMPG1 data deck  
/*
```

Job data set size (SPACE parameter) and program execution time (GTIME parameter) are strongly dependent on the number of isotopes, number of temperatures, and size of scattering matrices processed.

3.4 Error Stops

Data processing exceptions may result in printed warning messages which should be carefully analyzed by the user.

Six error stops are programmed as follows:

1. A new data set version name already appears in FINEGRP.CØNTENTS or NUCPARAM.CØNTENTS.
2. More than 200 isotopes are catalogued in data set(s).
3. Error in reading a data set version CØNTENTS record.
4. Error in reading FINEGRP.?FVERS.GRØUPS record.
5. Inconsistent number of groups catalogued in FINEGRP.?FVERSN.CØNTENTS versus NUCPARAM.?NVERSN.SIGMAP.
6. Error in reading NUCPARAM.?NVERSN.SIGMAP

JOSHUA DPSTM-500	Title FINEGRP Editor - Module FINED	Vol. V	Page C.3.1
		Rev.	Date 5/71

Written By

W. V. Baxter

Work Done By

W. V. Baxter
J. W. Stewart

Contents

	<u>Page</u>
1. <u>Theory</u>	C.3.3
1.1 Introduction	C.3.3
1.2 Representation of Data	C.3.3
2. <u>Program Logic and Coding Details</u>	C.3.5
2.1 Program Structure and Computational Sequence	C.3.5
2.2 Data Storage Allocation	C.3.5
3. <u>User Information</u>	C.3.7
3.1 Input	C.3.7
3.2 Output	C.3.7
3.3 Module Execution	C.3.7
3.4 Error Stops and Messages	C.3.7

JOSHUA DPSTM-500	Title FINEGRP Editor - Module FINED	Vol. V	Page C.3.2
		Rev.	Date 5/71

(Blank)

1. Theory

1.1 Introduction

The purpose of module FINED is to produce a printed edit of neutronics data catalogued in the FINEGRP data set. Each version of this data set has a specified energy group structure and energy spectrum associated with it. FINEGRP data may also be temperature dependent. All but four of the FINEGRP data records are isotope-dependent. An input option permits edit of data for all or selective isotopes in a version.

1.2 Representation of Data

A detailed description of the FINEGRP data records is given in Volume III of this document. A list of the records is given with brief descriptions below. The version data is listed first and then the isotope-dependent data.

CØNTENTS	-	Names of isotopes in the version.
GRØUPS	-	Group structure table.
SPECTRUM	-	Weighting spectrum.
WEIGHTS	-	Group integrals for spectrum.

Isotope Data:

CØNTENTS	-	Data types and reaction types for each isotope.
XSECTION.RTYPE	-	Smooth (P_0) cross sections for reaction type RTYPE.
PLXSECT.RTYPE.SIZE	-	Highest Legendre component for reaction RTYPE.
PLXSECT.RTYPE.L	-	Legendre cross sections for component L of reaction type RTYPE.
PMATRIX.RTYPE.SIZE	-	P_0 transfer matrix limits of $M_0 = J-I$ for reaction type RTYPE.
PMATRIX.RTYPE.M	-	P_0 transfer matrix for each M and RTYPE.
PLMATRIX.RTYPE.SIZE	-	Highest Legendre order for which transfer matrices are catalogued for RTYPE.

JOSHUA DPSTM-500	Title FINEGRP Editor - Module FINED	Vol. V	Page C.3.4
		Rev.	Date 5/71

- PLMATRIX.RTYPE.L.SIZE - Transfer matrix limiting values of M for a Legendre order and RTYPE.
- PLMATRIX.RTYPE.L.M - Transfer matrix for Legendre order L and a value of M for RTYPE.
- DELAYED.DECAY - Delayed neutron decay constants.
- DELAYED.CHI - Delayed neutron fission spectrum.
- DELAYED.NU - Delayed neutron yield per fission

JOSHUA DPSTM-500	Title FINEGRP Editor - Module FINED	Vol. V	Page C.3.5
		Rev.	Date 5/71

2. Program Logic and Coding Details

2.1 Program Structure and Computational Sequence

Module FINED has only one subroutine, MAIN. The input data record is read and scanned for blank entries. The data are compacted and the number of isotope version pairs is counted. The processing by isotope for a version is performed. The four version records are edited before isotope data. This is done only once for each version. An input option permits the isotope name to be 'ALL'. In this case every isotope in the designated version of FINEGRP is edited.

2.2 Data Storage Allocation

No ~~COMMON~~ blocks are used.

JOSHUA DPSTM-500	Title FINEGRP Editor - Module FINED	Vol. V	Page C.3.6
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title FINEGRP Editor - Module FINED	Vol. V	Page C.3.7
		Rev.	Date 5/71

3. User Information

3.1 Input

Module FINED requires one user-specified input data record:

INPUT.FINEGRP.EDIT

The data in this record consist of up to 21 name pairs for isotopes and versions to be edited. Eight characters are allowed for each name. Blank entries are allowed. If a version name is left blank, ditto from the last previous entry with a specified name is assumed.

A sample input record is shown at the end of this section.

3.2 Output

All the catalogued FINEGRP data for the requested isotope-version names will be edited. This follows an edit of the input data. The FINEGRP.CØNTENTS record is edited for the catalogue version names. Each name encountered in the input will be checked against this list. Edit of the CØNTENTS, GRØUPS, SPECTRUM, and WEIGHTS records for the version is performed and is followed by an edit of the isotope-dependent data.

3.3 Module Execution

Module FINED is referred to by MEMBER=FINED in PDS=AMD.JOSHUA. Required GØSIZE is 140K. No job data set space is required (SPACE=0).

The module cannot be executed from a terminal since FINEGRP is catalogued off-line. When executing FINED by means of the Batch Monitor, use DD card:

//GØ.FT22FOO1 DD DSNAME=JØS.JØSHUA.FINEGRP,DISP=SHR

3.4 Error Stops

An extensive set of error stops and diagnostic messages are programmed. The printed messages are self-explanatory.

[illegible]

JOSHUA DPSTM-500	Title Part D - NUCPARAM Data Processing	Vol. V	Page D.O.1
		Rev.	Date 5/71

Contents

D.1 NUCPARAM Editor - Module NUCED.

JOSHUA DPSTM-500	Title Part D - NUCPARAM Data Processing	Vol. V	Page D.O.2
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title NUCPARAM Editor - Module NUCED	Vol. V	Page D.1.1
		Rev.	Date 5/71

Written By

W. V. Baxter

Work Done By

W. V. Baxter
J. W. Stewart

Contents

	<u>Page</u>
1. <u>Theory</u>	D.1.3
1.1 Introduction	D.1.3
1.2 Representation of Data	D.1.3
2. <u>Program Logic and Coding Details</u>	D.1.5
2.1 Program Structure and Computational Sequence.	D.1.5
2.2 Data Storage Allocation	D.1.5
3. <u>User Information</u>	D.1.7
3.1 Input	D.1.7
3.2 Output	D.1.7
3.3 Module Execution	D.1.7
3.4 Error Stops and Messages	D.1.7

JOSHUA DPSTM-500	Title NUCPARAM Editor - Module NUCED	Vol. V	Page D.1.2
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title NUCPARAM Editor - Module NUCED	Vol. V	Page D.1.3
		Rev.	Date 5/71

1. Theory

1.1 Introduction

The purpose of module NUCED is to produce a printed edit of neutronics data catalogued in the NUCPARAM data set. This data set contains nuclear parameters which are related to FINEGRP, but independent of the MULTIGRP energy structure. Some of these data, such as resonance data and yields, are isotope-dependent. Other records span all the isotopes in a version of NUCPARAM. An input option permits edit of data for all or selective isotopes in a version.

1.2 Representation of Data

A detailed description of the NUCPARAM data records is given in Volume III of this document. In the record names, DISØ is a material name (a material usually consists of one isotope). The material-dependent records are:

CØNTENTS	-	Data types for each isotope.
RESØLVED.CØNTENTS	-	List of isotopes in a material DISØ, with resolved resonance data.
RESØLVED.IZASRI	-	Resolved resonance parameters for each isotope in a material.
UNRESØLV.CØNTENTS	-	List of isotopes in material DISØ with unresolved resonance data.
UNRESØLV.IZASUI	-	Unresolved resonance parameters for each isotope in a material.
YIELDS.SIZE	-	List of fission products for isotopes in material DISØ.
YIELDS.NBND	-	Fission product yields in band NBND for each isotope.

Each version of NUCPARAM has the following general records:

CØNTENTS	-	Names of isotopes (materials) catalogued for the version.
SIGMAP	-	Potential scattering cross sections for all isotopes in the version.
DECAY	-	Decay constants for all isotopes in the version.

JOSHUA DPSTM-500	Title NUCPARAM Editor - Module NUCED	Vol. V	Page D.I.4
		Rev.	Date 5/71

DPDATA.XSECTION - Effective cross sections for
the catalogued isotopes.

DPDATA.QVALUE - Q-values for depletion calcula-
tions.

2. Program Logic and Coding Details

2.1 Program Structure and Calculational Sequence

Module NUCED is a single subroutine, MAIN. The input data record is read and scanned for blank entries. The data are compacted and the number of isotope-version pairs is counted. Then processing by isotope for a version is performed until a new version name is encountered. Then the general data records for the specified version will be edited. Data for the next version will then be processed. An option in the input data permits the isotope name to be 'ALL'. When this is encountered, every isotope in the version is edited and followed by the general data for the version. Processing then continues by stepping to the next isotope name in the input data.

2.2 Data Storage Allocation

No CØMMØN blocks are used.

JOSHUA DPSTM-500	Title NUCPARAM Editor - Module NUCED	Vol. V	Page D.1.6
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title NUCPARAM Editor - Module NUCED	Vol. V	Page D.1.7
		Rev.	Date 5/71

3. User Information

3.1 Input

Module NUCED requires one user-specified input data record:

INPUT.NUCPARAM.EDIT

The data in this record consist of up to 21 name pairs for isotope and version to be edited. Eight characters are allowed for each name. Blank entries are allowed. If a version name is left blank, ditto from the last previous entry with a non-blank version name is assumed.

A sample input record is shown at the end of this section.

3.2 Output

All the catalogued NUCPARAM data for the requested isotope-version names will be edited. This is preceded by an edit of the input data. The isotope-dependent data will be edited before the general data for each version.

3.3 Module Execution

Module NUCED is referred to by MEMBER=NUCED in PDS=AMD.JØSHUA. Required GØSIZE is 140K. No job data set space is required (SPACE=0).

3.4 Error Stops

An extensive set of error stops and warning messages are programmed. Theprinted messages are self-explanatory.

[illegible]

Contents

- E.1 MULTIGRP Compressor - Module PRESS.
- E.2 MULTIGRP Editor - Module MULTED.
- E.3 MULTIGRP Composite Isotope Processor - Module
MULTIMAT.
- E.4 MULTIGRP to Tape Matrix Processor - Module MULTDUMP.
- E.5 Tape to MULTIGRP Matrix Processor - Module MULTREAD.

JOSHUA DPSTM-500	Title Part E - MULTIGRP Data Processing	Vol. V	Page E.O.2
		Rev. 1	Date 9/71

(Blank)

Written By

W. V. Baxter

Work Done By

W. V. Baxter
J. W. Stewart

Contents

	<u>Page</u>
1. <u>Theory</u>	E.1.3
1.1 Introduction	E.1.3
1.2 Cross Section Definitions	E.1.3
1.3 Compression	E.1.3
2. <u>Program Logic and Coding Details</u>	E.1.5
2.1 Overlay Structure and Computational Sequence	E.1.5
2.2 Data Storage Allocation	E.1.8
3. <u>User Information</u>	E.1.9
3.1 Input	E.1.9
3.2 Output	E.1.10
3.3 Module Execution	E.1.10
3.4 Error Stops	E.1.15

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.2
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.3
		Rev.	Date 5/71

1. Theory

1.1 Introduction

The purpose of the module PRESS is to compress neutronics data from an existing MULTIGRP structure into a MULTIGRP structure with a reduced number of energy groups. The original data set is assumed to be of the form generated by CØFIN (C.1) from FINEGRP data, and the resulting MULTIGRP data has the same form, with generally shorter records.

1.2 Cross Section Definitions

The form of MULTIGRP cross sections is discussed in C.1.1.2. There is no alteration of the form of the data produced by the PRESS module.

1.3 Compression

The formulas used in a compression with the PRESS module are essentially those of C.1.1.3. If the word MULTIGRP is substituted for the source of the data, the entire discussion of the above section applies.

A value of 0 is always assigned to the parameter IDBF, to indicate that up-scatter elements in the transfer matrices are catalogued. These elements are developed (for thermal groups only) before the matrices are compressed, since the detailed-balance equations that are used are not valid for broad-group matrix calculations.

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.4
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.5
		Rev.	Date 5/71

2. Program Logic and Coding Details

2.1 Overlay Structure and Computational Sequence

The PRESS module is overlaid to conserve machine core requirements. This overlay structure is shown in Figure 1. PRESS calculations are always of one type. That is, PRESS collapses data from a MULTIGRP format to a reduced energy-structure MULTIGRP format. The computational module is overlaid with the print module, which edits the calculated results.

The root segment of PRESS is composed of a MAIN subroutine, JØS utility routines, IBM FORTRAN and utility routines, and the SRL.FØRTLIB routine CLEAR. Subroutine MAIN sends control to SETUPP, the initial computational routine. After the calculations for all isotopes in the input data are completed, control returns to MAIN. Then MAIN calls MULTED, the first program in the print module. This causes a complete printed edit of all the cross-section and transfer-matrix data. Control returns to MAIN, which writes a termination message and ends the job.

The sequence of calculations is shown in Figure 2, C.1, where the series of boxes on the right is chosen. Table 1 shows the purpose of the routines.

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.6
		Rev.	Date 5/71

Figure 1. Overlay Structure in Module PRESS

* Named CØMMØN blocks. All other references are sub-routine entry points.

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.7
		Rev.	Date 5/71

TABLE I

Purpose of Routines in Module PRESS

<u>Routine Name</u>	<u>Purpose</u>
MAIN	Calls SETUPP to initiate compression calculations. On return, calls MULTED to produce edit. Issues "Normal Termination" message and terminates.
SETUPP	Calculates origin of variably-dimensioned arrays within COMMON/DATARP/. Checks to see if enough storage has been allocated. Calls COMPUP.
COMPUP	Reads input data. Processes GROUPS and CONTENTS records. Issues a series of calls to routines to complete reading of input data, and processes various types of cross section data.
FREAD	Reads weighting-spectrum data and computes sums of weights over reduced group structure. Writes MULTIGRP 'WEIGHTS' record.
SIGMA	Processes principal, secondary, and delayed cross-section data and computes group limits of non-zero data.
TMATP	Processes scattering matrices for elastic, inelastic, or (n,2n) reactions and computes group limits of non-zero data.
TOTMAP	Processes total scattering matrix data and computes group limits of non-zero data.
MULTED	Reads edit options and controls calls to routine GUTS, isotope by isotope.
GUTS	Produces printed edit for an isotope.

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.8
		Rev.	Date 5/71

2.2 Data Storage Allocation

A detailed description of data storage allocation for the PRESS module would be redundant, since it would repeat a large portion of C.1.2.2 for the CØFIN module. The only significant difference in the compression of data with the CØFIN module as compared to PRESS is that PRESS uses data from a MULTIGRP library. Hence, in order to apply C.1.2.2, Table III to PRESS all references to FINEGRP should be interpreted as descriptions of the initial MULTIGRP data set. The mnemonics are not changed.

Since PRESS always compresses data from a MULTIGRP library, there are only six named-CØMMØN blocks used. They are NAMP, PARP, DATARP, CØNP, DIMNP, and GRØUPS. No temperature interpolation is needed, and therefore the limiting parameters MAXDT and MAXDLT are dropped from CØMMØN/DIMNP/ as discussed in TABLE III, C.1.2.2. Also for the same array the indices KK39 through KK51 and KK130 are not used by the PRESS module. The SETUPP routine was modified to handle these changes.

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.9
		Rev.	Date 5/71

3. User Information

3.1 Input

Two input data records are required when module PRESS is executed. The first of these is

INPUT.PRESS?\$JØB

This record contains both the initial and resulting MULTIGRP version names. The user may select which data types are to be processed, specify up to 21 isotopes to be processed, and up to 21 isotopes to be deleted.

The second input data record is

INPUT.PRESS1?\$JØB

This record contains qualifiers in the LATTICE record in which the Legendre flux-weighting spectrum is stored. This record is of the type

LATTICE.?DLAT.?VLAT.LFLUX.?NCASE

In addition, the new MULTIGRP energy structure is related to the input MULTIGRP energy structure by the specification of

- Number of new MULTIGRP energy groups.
- Index of the first thermal energy group for the new MULTIGRP structure (≤ 31).
- Index of the most energetic initial MULTIGRP group for each of the new MULTIGRP groups.

A sample of INPUT.PRESS.\$JØB and INPUT.PRESS1\$JØB is given on the following pages.

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.10
		Rev.	Date 5/71

3.2 Output

Appropriate self-explanatory messages are printed during data processing to indicate missing input MULTIGRP records. Messages also indicate completion of data processing for each data type by isotope.

After completion of the data processing for all isotopes, a printed edit is produced for all isotopes added to the new MULTIGRP data set. This is followed by a 'Normal Termination' message to end the job.

3.3 Module Execution

Module PRESS may be executed via terminal or batch. The module requires 240K of core and for two or three isotopes the job will run in a minute. Number of lines is 500 - 1,000 per isotope.

The number of blocks (SPACE parameter) used in the job data set is strongly dependent on the number of isotopes and the size of the scattering matrices processed.

[illegible]

1058.INPUT.PRESS.UPDT

*****		ISOTOPES TO BE DELETED		*****	
ISOTOPE	TEMP.	ISOTOPE	TEMP.	ISOTOPE	TEMP. (OC)
-----	-----	-----	-----	-----	-----
:H-H2O	: 20	:	: 0	:	: 0
:	: 0	:	: 0	:	: 0
:	: 0	:	: 0	:	: 0
:	: 0	:	: 0	:	: 0
:	: 0	:	: 0	:	: 0
:	: 0	:	: 0	:	: 0
:	: 0	:	: 0	:	: 0

1058.INPUT.PRESS1.UPDT

LATTICE DATA SET NAME:HALFA

VERSION OF LATTICE DATA SET:CHECK

CASE NO.: 1

NO. OF GROUPS IN NEW STRUCTURE: 42

INDEX OF NEW FIRST THERMAL GROUP: 28

JOSHUA

DPSTM-500

Title MULTIGRP to MULTIGRP Processor - Module
PRESS.

Vol.
V

Page
E.1.13

Rev.

Date
5/71

1058.INPUT.PRESS1.UPDT

GROUP STRUCTURE INDICES ***

NEW	OLD	NEW	OLD	NEW	OLD	NEW	OLD	NEW	OLD	NEW	OLD	NEW	OLD
: 1:	1	: 2:	3	: 3:	5	: 4:	7	: 5:	9	: 6:	11	: 7:	13
: 8:	15	: 9:	17	:10:	19	:11:	21	:12:	23	:13:	25	:14:	27
:15:	29	:16:	31	:17:	33	:18:	35	:19:	37	:20:	39	:21:	41
:22:	43	:23:	45	:24:	47	:25:	49	:26:	51	:27:	53	:28:	55
:29:	57	:30:	59	:31:	61	:32:	63	:33:	65	:34:	67	:35:	69
:36:	71	:37:	73	:38:	75	:39:	77	:40:	79	:41:	81	:42:	83

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.14
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.15
		Rev.	Date 5/71

3.4 Error Stops

An extensive set of error warnings and stops are programmed.
The printed messages are self-explanatory.

JOSHUA DPSTM-500	Title MULTIGRP to MULTIGRP Processor - Module PRESS.	Vol. V	Page E.1.16
		Rev.	Date 5/71

(Blank)

Written By

J. W. Stewart

Work Done By

J. W. Stewart

CONTENTS

	<u>Page</u>
1. <u>Theory</u>	E.2.3
1.1 Introduction	E.2.3
1.2 Cross Section Representation	E.2.3
2. <u>Program Logic and Coding Details</u>	E.2.4
2.1 Program Structure and Calculational Sequence	E.2.4
2.2 Data Storage Allocation	E.2.4
3. <u>User Information</u>	E.2.5
3.1 Input	E.2.5
3.2 Output	E.2.8
3.3 Module Execution	E.2.8
3.4 Error Stops	E.2.8

JOSHUA DPSTM-500	Title MULTIGRP Editor - Module MULTED	Vol. V	Page E.2.2
		Rev.	Date 9/71

(Blank)

1. Theory

1.1 Introduction

The purpose of Module MULTED is to produce a printed edit of neutronics data catalogued in the MULTIGRP data set. All cross section data, the group limits for non-zero data, the group structure, and the cataloguing information contained in CONTENTS records are edited. By option, MULTED will produce a detailed edit of scattering matrix data.

1.2 Cross Section Representation

A detailed description of the MULTIGRP data records is given in Volume III of this document. Data are catalogued by isotope and temperature and by reaction type, including:

- XSECTION - Principle cross section data for all groups
- XSECT2 - Secondary cross section data for all groups
- DELAYED - Delayed neutron data for all groups and delayed families
- PLMATRIX.ELASTIC - Elastic scattering matrix data for all initial groups and Legendre orders. One record per final energy group.
- PLMATRIX.INELAST - Inelastic scattering matrix data for all initial groups and Legendre orders. One record per final energy group.
- PLMATRIX.N2N - (n,2n) scattering matrix data for all initial groups and Legendre orders. One record per final energy group.
- PLMATRIX.TOTAL - Total scattering matrix data for all initial groups and Legendre orders. One record per final energy group.

In addition to the above records the MULTIGRP data set contains catalogues in CONTENTS records, weighting spectra in WEIGHTS records, and energy group structure specifications in GROUPS records.

JOSHUA DPSTM-500	Title MULTIGRP Editor - Module MULTED	Vol. V	Page E.2.4
		Rev.	Date 9/71

2. Program Logic and Coding Details

2.1 Program Structure and Computational Sequence

Module MULTED is composed of two subroutines. Subroutine MAIN reads and processes user-specified input data and produces a printed edit of CONTENTS, ISOTOPES, and GROUPS data. MAIN calls subroutine GUTS for each isotope - temperature pair. GUTS reads and edits neutronics data records.

2.2 Data Storage Allocation

One named common block is used in module MULTED. The structure of the data contained in this block is summarized in Table I.

TABLE I

Structure of Named Common Block in Module Multed

<u>Variable</u>	<u>Type</u>	<u>Description</u>
COMMON /GROUPS/		
E(85)	R4	Upper energy (eV) boundary by group. E(NEG+1) is the lower energy boundary of the thermal-most group.
U(84)	R4	Lower lethargy boundary by group.
RV(84)	R4	Average reciprocal velocity by group.
EBAR(84)	R4	Characteristic energy by group.
NEG	I4	Number of energy groups.
JFTG	I4	Index of the first thermal group.
IDBF	I4	Detailed balance flag. IDBF=1 if only down-scatter elements are stored. IDBF=0 if full scattering matrix is stored.

JOSHUA DPSTM-500	Title MULTIGRP Editor - Module MULTED	Vol. V	Page E.2.5
		Rev.	Date 9/71

3. User Information

3.1 Input

Module MULTED requires one user-supplied input data record:

INPUT.MULTIGRP.EDIT

The data in this record represents a table of twenty entries. For each entry the user specifies an eight-character MULTIGRP version name, an eight-character isotope name, an integer temperature ($^{\circ}\text{C}$), and a matrix edit flag. If the matrix edit flag is zero, only outscatter cross-sections by group are edited; if greater than zero, a detailed matrix edit is produced. If the user leaves any version name blank, ditto from the last previous entry with a non-blank version name is assumed. In place of an isotope name, the user may insert 'ALL' implying that all isotopes in the specified version are to be edited. If 'ALL' is used, the temperature entry is ignored.

A sample input record is shown on the following pages.

3401.INPUT.MULTIGRP.EDIT

***** REQUEST FOR EDIT OF MULTIGRP *****

ISOTOPE MAY BE 'ALL'. TEMPERATURE IN O C. MATRIX=1 FOR DETAIL

___VERSION___	___ISOTOPE___	___TEMP___	___MATRIX___
:VERSION1	:U235	: 0	:1
:	:D-D2O	: 20	:1
:	:H-H2O	: 60	:0
:VERSION2	:H-H2O	: 20	:1
:STANDARD	:ALL	: 0	:0
:	:	: 0	:0
:	:	: 0	:0

PAGE 1

3401.INPUT.MULTIGRP.EDIT

***** REQUEST FOR EDIT OF MULTIGRP *****

ISOTOPE MAY BE 'ALL'. TEMPERATURE IN O C. MATRIX=1 FOR DETAIL

___VERSION___	___ISOTOPE___	___TEMP___	___MATRIX___
:	:	: 0	:0
:	:	: 0	:0
:	:	: 0	:0
:	:	: 0	:0
:	:	: 0	:0
:	:	: 0	:0
:	:	: 0	:0

PAGE 2

3401.INPUT.MULTIGRP.EDIT

***** REQUEST FOR EDIT OF MULTIGRP *****

ISOTOPE MAY BE 'ALL'. TEMPERATURE IN C. MATRIX=1 FOR DETAIL

___VERSION___	___ISOTOPE___	___TEMP___	___MATRIX___
:	:	: 0	:0
:	:	: 0	:0
:	:	: 0	:0
:	:	: 0	:0
:	:	: 0	:0
:	:	: 0	:0

JOSHUA DPSTM-500	Title MULTIGRP Editor - Module MULTED	Vol. V	Page E.2.8
		Rev.	Date 9/71

3.2 Output

All relevant MULTIGRP neutronics, limits, energy structure, and cataloging data are printed. Data associated with a specific version (e.g. isotope catalog, energy structure) are edited only once.

3.3 Module execution

Module MULTED is referenced as MEMBER=MULTED in PDS=AMD.JOSHUA. Required GØSIZE is 240K. Only one block (SPACE=1) of job data set space is required.

3.4 Error Stops

An extensive set of error stops and warning messages are programmed. The printed messages are self-explanatory.

Written By

W. V. Baxter

Work Done By

W. V. Baxter
J. W. Stewart

CONTENTS

	<u>Page</u>
1. <u>Theory</u>	E.3.3
1.1 Introduction	E.3.3
1.2 Neutronics Data Representation	E.3.3
1.3 Methods and Formulas	E.3.3
2. <u>Program Logic and Coding Details</u>	E.3.5
2.1 Program Structure	E.3.5
2.2 Data Layout	E.3.5
3. <u>User Information</u>	E.3.6
3.1 Input	E.3.6
3.2 Output	E.3.6
3.3 Module Execution	E.3.6
3.4 Error Diagnostics	E.3.6

JOSHUA DPSTM-500	Title MULTIGRP Composite Isotope Processor - Module MULTIMAT	Vol. v	Page E.3.2
		Rev.	Date 9/71

(Blank)

1. Theory

1.1 Introduction

Module MULTIMAT develops MULTIGRP data for a reactor material, or composite 'isotope' with five or less components. Data for the component isotopes already exist in a version of MULTIGRP. Cross sections and transfer matrix elements are formed by combining component data according to the atom fractions of the component isotopes. CONTENTS and LIMITS records are generated for the new 'isotope'. The ISOTOPES record is modified to include the new isotope.

Delayed neutronics data are not treated by MULTIMAT.

1.2 Neutronics Data Representation

The MULTIGRP data set is described in Volume III of this document. All the records generated by MULTIMAT are catalogued in the user-specified MULTIGRP version.

1.3 Methods and Formulas

The number fractions of the component isotopes are denoted by the set W_i , $i=1,2,\dots,N$, normalized such that

$$\sum_{i=1}^N W_i = 1.0 \quad (1)$$

The number of atoms of component isotope i per composite molecule is

$$W_i^* = N W_i \quad (2)$$

MULTIGRP cross sections for the component isotopes are combined by the formula:

$$\bar{\sigma}^g = \sum_{i=1}^N W_i^* \sigma_i^g \quad (3)$$

where the bar indicates the composite, i the isotope index, and g the energy group index. The atomic weight ratio for the composite is computed from

$$\bar{A} = \sum_{i=1}^N W_i^* A_i \quad (4)$$

If M of the N component isotopes are fissionable the fission spectrum is computed from

$$\bar{x}^g = \frac{1}{M} \sum_{i=1}^M x_i^g w_i^* \quad (5)$$

Equation (5) assures proper normalization, i.e.

$$\sum_g \bar{x}^g = 1.0 \quad (6)$$

The number of neutrons produced per fission is computed from

$$\bar{\nu}^g = \frac{\sum_{i=1}^M \nu_i^g \sigma_{fi}^g w_i^*}{\bar{\sigma}_f^g} \quad (7)$$

2. Program Logic and Coding Details

2.1 Program Structure

Module MULTIMAT consists of a main subroutine, a computational subroutine CØMP, and an edit package initiated by the subroutine MULTED. After generating a composite 'isotope', the subroutine CØMP is overlaid by the MULTED package. Briefly, the subroutines are described below:

MAIN: This controls the sequence of calculations. MAIN calls EFTM (a systems subroutine) and then calls CØMP. After return from CØMP, a message is written to indicate completion of the isotope, and then MULTED is called. After the editing is finished, MAIN writes a 'JØB FINISHED' message and terminates.

CØMP: This subroutine combines the MULTIGRP data and modifies LIMITS records to get MULTIGRP records for the composite isotope. No CØMMØN storage is used.

MULTED: This package contains the same programs and storage allocations as Module MULTED, described in Section E.2 of the present volume. No changes are introduced here.

2.2 Data Layout

All the data necessary for building a composite isotope are stored in arrays within the CØMP subroutine. No CØMMØN blocks are used. One named CØMMØN block is used by the MULTED package. The structure of the data contained in this block is shown in Table I, Section E.2 of this volume.

JOSHUA DPSTM-500	Title MULTIGRP Composite Isotope Processor - Module MULTIMAT	Vol. V	Page E.3.6
		Rev.	Date 9/71

3. User Information

3.1 Input

Module MULTIMAT requires one user-supplied input data record:

INPUT.MULTIMAT.?\$JØB

This record contains information about the composite isotope, the component isotopes, and an 8-character name of the MULTIGRP data set being modified. An 8-character name is required for the composite isotope, along with its temperature (°C), its IZAS number, and the number of components. A table of data for the component isotopes is required with 3 parameters for each isotope. These are 1) an 8-character name, 2) the integer temperature of the isotope, and 3) the atom fraction as used in the composite.

A sample input record is shown on a following page.

3.2 Output

A complete edit of all relevant MULTIGRP neutronics, limits, energy structure, and cataloging data is provided. This MULTED output for the new isotope is similar to CØFIN output.

3.3 Module Execution

Module MULTIMAT is referenced as MEMBER=MULTIMAT in PDS=AMD.JØSHUA. Required GØSIZE is 250K. GTIME=1, SPACE=5, and LINES=4 are adequate for any composite isotope.

3.4 Error Diagnostics

Error stops and warning messages are programmed. Printed messages will explain the conditions and actions taken.

JOSHUA
DPSTM-500

Title MULTIGRP Composite Isotope Processor -
Module MULTIMAT

Vol. V	Page E.3.7
Rev.	Date 9/71

1058.INPUT.MULTIMAT.TEST1058

NAME OF MULTIGRP DATA SET : STANDARD

NUMBER OF COMPONENT ISOTOPES : 3 -- ATOMS/MOLECULE : 3

NAME OF COMPOSITE ISOTOPE : HD20

TEMP. OF ABOVE ISOTOPE : 20

IZAS NO. OF NEW ISOTOPE : 180000

INCLUDE UP TO 5 COMPONENT ISOTOPES AS FOLLOWS

NAME	TEMP.	WEIGHT	NAME	TEMP.	WEIGHT
:H-H2O	: 20	: .1700E-02	:O	: 20	: .3333E 00
:D-D2O	: 20	: .6650E 00			

JOSHUA

DPSTM-500

Title MULTIGRP to Tape Matrix Processor -
Module MULTDUMP

Vol.
V

Page
E.4.1

Rev.

Date
9/71

Written By

J. W. Stewart

Work Done By

J. W. Stewart

Contents

	<u>Page</u>
1. Introduction	E.4.3
2. Program Logic	E.4.5
3. User Information	E.4.13
3.1 Input	E.4.13
3.2 Output	E.4.13
3.3 Module Execution	E.4.13
3.4 Error Stops	E.4.13

JOSHUA DPSTM-500	Title MULTIGRP to Tape Matrix Processor - Module MULTDUMP	Vol. V	Page E.4.2
		Rev.	Date 9/71

(Blank)

1. Introduction

The several modules (e.g., CØFIN and PRESS) which generate MULTIGRP nuclear data records process scattering matrices for elastic scattering, inelastic scattering, and (n,2n) scattering. These component scattering matrices are summed to form the total scattering matrix. It is this total scattering matrix which is used in lattice physics codes such as RAHAB. Since the component scattering matrices are not required by lattice physics modules and since these data occupy a large amount of disk space in the JOSHUA data base, it is desirable to be able to move these data to and from tape. Module MULTDUMP was developed to provide the capability of moving component multigroup data from the JOSHUA data base to a sequential file on tape. This module is described in this section. A companion module MULTREAD was developed to copy these data from a sequential file on tape back into the JOSHUA data base and is described in Section E.5.

JOSHUA
DPSTM-500

Title MULTIGRP to Tape Matrix Processor -
Module MULTDUMP

Vol.
V

Page
E.4.4

Rev.

Date
9/71

(Blank)

JOSHUA DPSTM-500	Title MULTIGRP to Tape Matrix Processor - Module MULTDUMP	Vol.	Page
		V	E.4.5
		Rev.	Date
			9/71

2. Program Logic

Module MULTDUMP is a single subroutine. A listing of this subroutine is included as Figure 1. The input data consists of the MULTIGRP version name which is to be processed, a tape mode indicator (IMODE), and a table of up to forty isotope names and temperatures. If IMODE is input as unity a new data tape is written, if IMODE is input as zero an existing tape is updated. The new tape is designated as TAPNEW=90, and the existing tape is designated as unit TAPOLD=91. The input data are processed to determine the number of isotopes to be written to the new tape. If an existing tape is to be updated the existing tape contents record is read and these isotopes are added to the table of contents. The contents record is written to the new tape. If an existing tape is to be updated, the data on the old tape are read and copied to the new tape. Data are then processed for each added isotope in turn in the order elastic data, inelastic data, (n,2n) data. For each of these isotopes the appropriate multigroup contents record is read, the number of records to be written (NREC) for this isotope is computed and the isotope header record is written to tape. The scattering data are then read from the multigroup data set and written to tape. After all data are written to tape an updated tape table of contents is printed, the new tape is rewound and a stop zero is issued.

The structure of the sequential file written to tape is shown in Figure 2.

Figure 1. Source Listing of Module MULTDUMP.

```

MODULE
C  MODULE MULTDUMP
  REAL*8  VERSNI,DISOI(40)/40*'      '//,VERSN/'      '//,
XDISO(200)/200*'      '//,DISOX/'      '//,BL8/'      '//
  REAL*4  XS(10),TM(100,10)
  INTEGER*4  TAPOLD,TAPNEW,ITDCI(40)/40*0/,ITDC(200)/200*0/,
XNISO/0/,IZERO/0/
  INTEGER*2  LP(100)/100*0/,IL(100)/100*0/,IH(100)/100*0/,L1,I1,I2
  READ(5,1000,END=50)VERSNI,IMODE,(DISOI(I),ITDCI(I),I=1,40)
1000  FORMAT(A8,I1/(A8,I8))
C
C  VERSNI  MULTIGRP VERSION TO BE PROCESSED
C  IMODE   TAPE MODE
C          IMODE=0  UPDATE EXISTING TAPE
C          IMODE=1  WRITE NEW TAPE
C  DISOI   ISOTOPE NAME
C  ITDCI   ISOTOPE TEMPERATURE
C
50  CONTINUE
  TAPNEW=90
  TAPOLD=91
C
C  TAPOLD  IS UNIT NUMBER OF OLD TAPE
C  TAPNEW  IS UNIT NO. OF NEW TAPE
C
  WRITE($SYS,2000)  VERSNI,(DISOI(I),ITDCI(I),I=1,40)
2000  FORMAT('1',120('='))/T50,'MULTIGRP MATRIX DUMP TO TAPE'/
XT60,'VERSION ',A8/1X,120('='))/T50,'ISOTOPE',8X,'TEMP'/
XT50,2('-----',5X)/(T50,A8,5X,I8))
  IF(IMODE.EQ.0) WRITE($SYS,2001)
  IF(IMODE.GT.0) WRITE($SYS,2002)
2001  FORMAT('UPDATE EXISTING TAPE')
2002  FORMAT('WRITE NEW TAPE')
  REWIND TAPNEW
  IF(IMODE.EQ.1) GO TO 2
  REWIND TAPOLD
  READ(TAPOLD)  VERSN,NISDX,(DISO(I),ITDC(I),I=1,200)
  NISO=NISDX
  IF(VERSN.NE.VERSNI) GO TO 13
2  CONTINUE
  VERSN=VERSNI
C  CHECK INPUT
DO 11  I=1,40
  IF(DISOI(I).EQ.BL8) GO TO 12
11  CONTINUE
12  NISOI=I-1
  IF(NISOI.GT.0) GO TO 14
  WRITE($SYS,2003)
2003  FORMAT('0***** FIRST ISOTOPE NAME IS BLANK.  ILLEGAL.'
X/'      NO IMBEDDED BLANK ISOTOPE NAMES ARE ALLOWED.'//)

```

```

STOP
14 DO 30 IS=1,NISOI
    NISO=NISO+1
    DISO(NISO)=DISOI(IS)
30 ITDC(NISO)=ITDCI(IS)
    WRITE(TAPNEW) VERSN,NISO,(DISO(I),ITDC(I),I=1,270)
    IF(IMODE.EQ.1) GO TO 1
    WRITE($SYS,3001)
    DO 3 IS=1,NISOX
    READ(TAPOLD) DISOX,ITDCX,NREC,JLTME,JHTME,JLTM1,JHTM1,
X JLTMN,JHTMN
    WRITE($SYS,3002) DISOX,ITDCX,NREC,JLTME,JHTME,JLTM1,JHTM1,
XJLTMN,JHTMN
    WRITE(TAPNEW) DISOX,ITDCX,NREC,JLTME,JHTME,JLTM1,JHTM1,
X JLTMN,JHTMN
    IF(JLTME.EQ.0) GO TO 4
    READ(TAPOLD) J1,J2,(LP(J),IL(J),IH(J),J=J1,J2)
    WRITE(TAPNEW) J1,J2,(LP(J),IL(J),IH(J),J=J1,J2)
    DO 5 J=JLTME,JHTME
    READ(TAPOLD) L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
    WRITE(TAPNEW) L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
5 CONTINUE
    WRITE($SYS,3003)
4 CONTINUE
    IF(JLTM1.EQ.0) GO TO 6
    READ(TAPOLD) J1,J2,(LP(J),IL(J),IH(J),J=J1,J2)
    WRITE(TAPNEW) J1,J2,(LP(J),IL(J),IH(J),J=J1,J2)
    DO 7 J=JLTM1,JHTM1
    READ(TAPOLD) L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
    WRITE(TAPNEW) L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
7 CONTINUE
    WRITE($SYS,3004)
6 CONTINUE
    IF(JLTMN.EQ.0) GO TO 8
    READ(TAPOLD) J1,J2,(LP(J),IL(J),IH(J),J=J1,J2)
    WRITE(TAPNEW) J1,J2,(LP(J),IL(J),IH(J),J=J1,J2)
    DO 9 J=JLTMN,JHTMN
    READ(TAPOLD) L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
    WRITE(TAPNEW) L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
9 CONTINUE
    WRITE($SYS,3005)
8 CONTINUE
    WRITE($SYS,3006) DISOX,ITDCX
3 CONTINUE
    WRITE($SYS,3007)
C END OF COPY
1 CONTINUE
    WRITE($SYS,2004) (DISOI(I),ITDCI(I),I=1,NISOI)
2004 FORMAT('OBEGIN ADDING THE FOLLOWING ISOTOPES TO TAPE-'/
X(5X,A8,I8))
    DO 15 IS=1,NISOI
    'MULTIGRP'.VERSN1.DISOI(IS).ITDCI(IS)->NAME
    READ(NAME,'CONTENTS') IZAS,AWP,ICHI,JHCHI,JLTME,JHTME,
XJLTM1,JHTM1,JLTMN,JHTMN
    NREC=0
    IF(JLTME.NE.0) NREC=NREC+JHTME-JLTME+2
    IF(JLTM1.NE.0) NREC=NREC+JHTM1-JLTM1+2
    IF(JLTMN.NE.0) NREC=NREC+JHTMN-JLTMN+2
    WRITE(TAPNEW) DISOI(IS),ITDCI(IS),NREC,JLTME,JHTME,JLTM1,JHTM1,

```

```

XJLT MN, JHT MN
WRITE(6,4000) DISOI(15),ITDCI(15),NREC,JLTME,JHTME,JLTMI,JHTMI,
XJLT MN, JHT MN
4000 FORMAT('O BEGIN PROCESSING ',A8,18/10X,'NREC = ',I10/10X,'ELASTIC='
X,2I5/10X,'INELAST=',2I5/10X,'N2N      =',2I5/)
IF(JLTME.EQ.0) GO TO 16
READ(NAME.'LIMITS'. 'ELASTIC')  J1,J2,(LP(I),IL(I),IH(I),I=J1,J2)
WRITE(TAPNEW)                    J1,J2,(LP(I),IL(I),IH(I),I=J1,J2)
DO 17 J=JLTME,JHTME
READ(NAME.'PLMATRIX'. 'ELASTIC'.J,ERR=18)
XL1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
WRITE(TAPNEW)  L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
GO TO 17
18  L1=1
    I1=1
    I2=1
    ZERO=0.
WRITE(TAPNEW)  L1,I1,I2,ZERO,ZERO
17  CONTINUE
16  IF(JLTMI.EQ.0) GO TO 19
READ(NAME.'LIMITS'. 'INELAST')  J1,J2,(LP(I),IL(I),IH(I),I=J1,J2)
WRITE(TAPNEW)                    J1,J2,(LP(I),IL(I),IH(I),I=J1,J2)
DO 20 J=JLTMI,JHTMI
READ(NAME.'PLMATRIX'. 'INELAST'.J,ERR=21 )
XL1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
WRITE(TAPNEW)  L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
GO TO 20
21  L1=1
    I1=1
    I2=1
    ZERO=0.
WRITE(TAPNEW)  L1,I1,I2,ZERO,ZERO
20  CONTINUE
19  IF(JLT MN.EQ.0) GO TO 22
READ(NAME.'LIMITS'. 'N2N      ')  J1,J2,(LP(I),IL(I),IH(I),I=J1,J2)
WRITE(TAPNEW)                    J1,J2,(LP(I),IL(I),IH(I),I=J1,J2)
DO 23 J=JLT MN,JHT MN
READ(NAME.'PLMATRIX'. 'N2N      '.J,ERR=24)
XL1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
WRITE(TAPNEW)  L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
GO TO 23

```



```

24  L1=1
    I1=1
    I2=1
    ZERO=0.
    WRITE(TAPNEW)  L1,I1,I2,ZERO,ZERO
23  CONTINUE
22  CONTINUE
    WRITE(6,4001)  DISO1(15),ITDC1(15)
4001 FORMAT(7X,'END PROCESSING ',A8,I8//)
15  CONTINUE
C
C
C  END OF DATA WRITING
    WRITE($SYS,3008)  VERSN,NISO,(DISO(1),ITDC(1),I=1,200)
3008 FORMAT('1',120('=')/T50,'UPDATED TAPE CONTENTS'/T40,'MULTIGRP.',
  SA8,10X,'NISO=',I5/1X,120('=')//20X,4(' ISOTOPE    TEMP    ')/
  X20X,4(' ----- ')/(21X,A8,I9,3X,A8,I9,3X,A8,I9,3X,A8,
  XI9))
    REWIND TAPNEW
    STOP  'NORMAL TERMINATION'
13  WRITE($SYS,3000)  VERSN,VERSN1
3000 FORMAT('0***** DATA SET VERSION ON TAPE (' ,A8,') & ON INPUT (' ,
  XA8,') DISAGREE.'//)
    STOP 1
3001 FORMAT('0BEGIN TAPE COPY'//)
3002 FORMAT('0COPY ',A8,I8/10X,'NREC =',I10/10X,'ELASTIC=',2I5/
  X10X,'INELAST=',2I5/10X,'N2N    =',2I5//)
3003 FORMAT(10X,'END ELASTIC DATA COPY')
3004 FORMAT(10X,'END INELAST DATA COPY')
3005 FORMAT(10X,'END N2N    DATA COPY')
3006 FORMAT('0 END COPY OF ',A8,I8//)
3007 FORMAT('0END OF TAPE COPY'//)
    END

```

Figure 2. Structure of Sequential Tape File

TABLE I

STRUCTURE OF SEQUENTIAL TAPE FILE

<u>Variable</u>	<u>Description</u>	<u>Type</u>
<u>Record Type 1. Table of Contents</u>		
VERSN, NISØ, (DISØ(I), ITDC(I), I=1,200)		
VERSN	MULTIGRP Version Name	C8
NISØ	Number of isotopes catalogued.	I4
DISØ	Isotope name.	C8
ITDC	Integer isotope temperature (°C).	I4
<u>Record Type 2. Isotope Header</u>		
DISØT, ITDCT, NREC, JLTME, JHTME, JLTMI, JHTMI, JLTMN, JHTMN		
DISØT	Isotope name.	C8
ITDCT	Integer isotope temperature (°C).	I4
NREC	Number of records to follow for this isotope.	I4
	$NREC = (JHTME - JLTME + 2) * (1. - \delta_{O, JHTME})$ $+ (JHTMI - JLTMI + 2) * (1. - \delta_{O, JHTMI})$ $+ (JHTMN - JLTMN + 2) * (1. - \delta_{O, JHTMN})$	
JLTME	Lower value of final energy group for elastic data.	I4
JHTME	Higher value of final energy group for elastic data.	I4
JLTMI	Lower value of final energy group for inelastic data.	I4
JHTMI	Higher value of final energy group for inelastic data.	I4
JLTMN	Lower value of final energy group for (n,2n) data.	I4
JHTMN	Higher value of final energy group for (n,2n) data.	I4

JOSHUA DPSTM-500	Title MULTIGRP to Tape Matrix Processor - Module MULTDUMP	Vol. V	Page E.4.12
		Rev.	Date 9/71

TABLE I (Cont.)

<u>Variable</u>	<u>Description</u>	<u>Type</u>
-----------------	--------------------	-------------

Record Type 3. Scattering Data Limits

J1, J2, (LP(I), IL(I), IH(I), I = J1,J2)

J1	Lower value of final energy group for scattering data.	I4
J2	Higher value of final energy group for scattering data.	I4
LP	Number of Legendre orders catalogued.	I2
IL	Lower value of initial energy group.	I2
IH	Higher value of initial energy group.	I2

Record Type 4. Scattering Data

L1, I1, I2, (XS(L), (TM(I,L), I = I1, I2), L = 1, L1)

L1	Number of Legendre orders catalogued.	I2
I1	Lower value of initial energy group.	I2
I2	Higher value of initial energy group.	I2
XS	Scattering cross section.	R4
TM	Scattering matrix.	R4

JOSHUA DPSTM-500	Title MULTIGRP to Tape Matrix Processor - Module MULTDUMP	Vol. V	Page E.4.13
		Rev.	Date 9/71

3. User Information

3.1 Input

All input data to module MULTDUMP are input on cards. The header card contains the name of the MULTIGRP data set to be processed in columns 1 through 8 and the tape mode indicator (IMØDE) in column 9. Following the header card as many as forty isotope cards can be included in the input deck. Each of these cards contains the isotope name to be processed in columns 1 through 8 and the integer temperature in °C in columns 9 through 16. All of the isotope cards with non-blank isotope names must precede the first blank card in the data deck.

3.2 Output

The output produced by module MULTDUMP is largely self-explanatory. A printed edit of the input data is included. If an existing tape is to be updated several lines of information are printed as each isotope is copied in turn. A message is printed to indicate the end of the tape copy and the beginning of the processing of additional isotopes. Several lines are printed as each of the additional isotopes is processed in turn. At the end of the job an updated tape table of contents is printed and a message indicating normal termination is printed.

3.3 Module Execution

This module is referenced as MEMBER=MULTDUMP and PDS=AMD. JOSHUA. The module will execute in 200K bytes of core and will process up to forty isotopes in less than one minute CPU time. Two examples of the appropriate JCL cards are shown in Figure 3.

3.4 ERROR STOPS

Several self-explanatory error messages are printed by the module.

To write a new tape:

```
//ABCD1234 JOB (1234,L000,1,1,0,,,8690-00,L5364-02,J,01),  
//          'your name',MSGLEVEL=1,CLASS=M  
//STEP1 EXEC JOSHUA,MEMBER=MULTDUMP,GØSIZE=200K,  
//  BADGE=1234,JOB=AAAA1234,DISP=TMP  
//GØ.FT90FO01 DD DSN='data set name',DISP=(NEW,KEEP),  
//  UNIT=2400,VØLUME=SER=LA0123,DCB=(RECFM=VBS,  
//  DEN=2,BLKSIZE=1804,LRECL=1800)  
//GØ.SYSIN DD *  
    Data  
/*
```

To update existing tape:

Same as above, except insert following card before the //GØ.SYSIN card.

```
//GO.FT91FO01 DD DSN='data set name',DISP=(ØLD,KEEP),  
//  UNIT=2400,VOLUME=SER=LA0222,DCB=(RECFM=VBS,  
//  DEN=2,BLKSIZE=1804,LRECL=1800)
```

Figure 3. JCL for MULTDUMP Execution.

JOSHUA DPSTM-500	Title Tape to MULTIGRP Matrix Processor - Module MULTREAD	Vol. V	Page E.5.1
		Rev.	Date 9/71

Written By

J. W. Stewart

Work Done By

J. W. Stewart

Contents

	<u>Page</u>
1. Introduction	E.5.3
2. Program Logic	E.5.5
3. User Information	E.5.9
3.1 Input	E.5.9
3.2 Output	E.5.9
3.3 Module Execution	E.5.9
3.4 Error Stops	E.5.9

JOSHUA DPSTM-500	Title Tape to MULTIGRP Matrix Processor - Module MULTREAD	Vol.	Page
		V	E.5.2
		Rev.	Date
			9/71

(Blank)

1. Introduction

The several modules (e.g., CØFIN and PRESS) which generate MULTIGRP nuclear data records process scattering matrices for elastic scattering, inelastic scattering, and (n,2n) scattering. These component scattering matrices are summed to form the total scattering matrix. It is this total scattering matrix which is used in lattice physics codes such as RAHAB. Since the component scattering matrices are not required by lattice physics modules and since these data occupy a large amount of disk space in the JOSHUA data base, it is desirable to be able to move these data to and from tape. Module MULTDUMP was developed to provide the capability of moving component multigroup data from the JOSHUA data base to a sequential file on tape. This module is described in Section E.5. A companion module MULTREAD was developed to copy these data from a sequential file on tape back into the JOSHUA data base and is described in this section.

JOSHUA DPSTM-500	Title Tape to MULTIGRP Matrix Processor - Module MULTREAD	Vol. V	Page E.5.4
		Rev.	Date 9/71

(Blank)

JOSHUA DPSTM-500	Title Tape to MULTIGRP Matrix Processor - Module MULTREAD	Vol. V	Page E.5.5
		Rev.	Date 9/71

2. Program Logic

Module MULTREAD is a single subroutine. A listing of this subroutine is included as Figure 1. The input data consists of the MULTIGRP version name which is to be processed and a table of up to forty isotope names and temperatures. The tape is designated as TAPNEW=90. The input data are processed to determine the number of isotopes to be processed. The table of contents record is read from tape and the problem input table of isotopes is sorted to correspond to tape isotope sequence. Data are processed, isotope by isotope, by searching the tape and writing MULTIGRP records to the job data set. The job ends with a stop zero and 'NORMAL TERMINATION' message.

The structure of the sequential file written to tape is shown in Figure 2, page E.4.10.

Figure 1. Source Listing of Module MULTREAD.

```

MODULE
C  MODULE MULTREAD
  REAL*8  VERSNI/'',DISOI(40)/40*'','',
XVER SN/'',DISOI(200)/200*'','',DISOX/'',
XBL8/'',DISOT,STYPE(3)/'ELASTIC','INELAST','N2N'/
  REAL*4  XS(10),TM(100,10)
  INTEGER*4  TAPNEW ,ITDCI(40)/40*0/,ITDC(200)/200*0/,NISO(0)/,
XIZEPO(0)/,INDEX(40)/40*0/,JLO(3),JHI(3)
  INTEGER*2  LP(100),IL(100),IH(100),L1,I1,I2
  EQUIVALENCE(JLO(1),JLTME),(JLO(2),JLTMI),(JLO(3),JLTMN),
X(JHI(1),JHTME),(JHI(2),JHTMI),(JHI(3),JHTMN)
  TAPNEW=90
  READ(5,1000,END=99)  VERSNI,(DISOI(I),ITDCI(I),I=1,40)
1000  FORMAT(A8/(A8,I8))
99  WRITE($SYS,2000)  VERSNI,(DISOI(I),ITDCI(I),I=1,40)
2000  FORMAT('1',120('='))/T50,'MULTIGRP MATRIX RETRIEVAL FROM TAPE'/
XT60,'VERSION ',A8/1X,120('='))/T50,'ISOTOPE          TEMP'/T50,
X2('-----',5X)/(T50,A8,5X,I8))

C
C  CHECK INPUT
DO 1  I=1,40
  IF(DISOI(I).EQ.BL8)  GO TO 2
1  CONTINUE
2  NISOI=I-1
  IF(NISOI.GT.0)  GO TO 4
  WRITE($SYS,2001)
2001  FORMAT('0***** FIRST ISOTOPE NAME IS BLANK. ILLEGAL.'/
X'  NO IMBEDDED BLANK ISOTOPE NAMES ALLOWED'//)
  STOP
4  REWIND TAPNEW
  READ(TAPNEW)  VERSN,NISO,(DISOI(I),ITDCI(I),I=1,200)
  WRITE($SYS,2004)  VERSN,NISO,(DISOI(I),ITDCI(I),I=1,200)
2004  FORMAT('1',120('='))/T50,'TAPE CONTENTS'/1X,120('=')//
XT50,'VERSION = ',A8/T50,'NISO = ',I4//
X(2X,A8,I5,2X,A8,I5,2X,A8,I5,2X,A8,I5,2X,A8,I5))
  IF(VERSN.NE.VERSN1)  GO TO 5

C
C  REORDER INPUT ISOTOPE LIST TO CORRESPOND TO TAPE SEQUENCE
C
DO 6  IN=1,NISOI
DO 7  IT=1,NISO
  IF(DISOI(IN).EQ.DISO(IT).AND.ITDCI(IN).EQ.ITDC(IT))  GO TO 8
7  CONTINUE
  INDEX(IN)=0
  WRITE($SYS,2003)  DISOI(IN),ITDCI(IN)
2003  FORMAT('0***** ISOTOPE ',A8,'.',I8,' NOT ON TAPE. PROCESSING OMITT
XED.')
8  INDEX(IN)=I T

```

```

6      CONTINUE
C
C      DELETE ZERO INDEX ISOTOPES
      NX=NISOI
      DO 9  IN=1,NISOI
      IF(INDEX(IN).GT.0) GO TO 9
      NX=NX-1
      NM1=NISOI-1
      DO 10 I=IN,NM1
      DISOI(I)=DISOI(I+1)
10     ITDCI(I)=ITDCI(I+1)
9      CONTINUE
      NISOI=NX
C
C      REORDER ISOTOPES
      DO 11 IN=1,NISOI
      DO 12 I=1,NISOI
      IF(INDEX(I).LE.INDEX(IN)) GO TO 12
      IND=INDEX(I)
      DISOX=DISOI(I)
      ITDCX=ITDCI(I)
      INDEX(I)=INDEX(IN)
      DISOI(I)=DISOI(IN)
      ITDCI(I)=ITDCI(IN)
      INDEX(IN)=IND
      ITDCI(IN)=ITDCX
      DISOI(IN)=DISOX
12     CONTINUE
11     CONTINUE
C
C      WRITE($SYS,4000) NISOI,(DISOI(I),ITDCI(I),I=1,NISOI)
4000  FORMAT('1BEGIN RETRIEVING DATA FOR ',I5,' ISOTOPES: '/(2X,A8,I8))
C
C      START PROCESSING ISOTOPES
      DO 14 IS=1,NISOI
      DISOX=DISOI(IS)
      ITDCX=ITDCI(IS)
      'MULTIGRP'.VERSNI.DISOX.ITDCX->NAME
      READ(TAPNEW) DISOT,ITDCT,NREC,JLTME,JHTME,JLTM1,JHTM1,
XJLTMN,JHTMN
      WRITE(6,3000) DISOT,ITDCT,NREC,JLTME,JHTME,JLTM1,JHTM1,
XJLTMN,JHTMN
      IF(DISOT.EQ.DISOX.AND.ITDCT.EQ.ITDCX) GOTO15
      DO 16 I=1,NREC
16     READ(TAPNEW)
      GO TO 14

```

```

15  DO 17  IT=1,3
    IF(JLO(IT).EQ.0) GO T 017
    READ(TAPNEW)  J1,J2,(LP(J),IL(J),IH(J),J=J1,J2)
    WRITE(6,3001)  VERSNI,DISOX,ITDCX,
X   STYPE(IT),(J,LP(J),IL(J),IH(J),J=J1,J2)
    WRITE(NAME.'LIMITS'.STYPE(IT))
X   J1,J2,(LP(J),IL(J),IH(J),J=J1,J2)
    DO18  J=J1,J2
    READ(TAPNEW)  L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
    IF(L1.EQ.1.AND.I1.EQ.1.AND.I2.EQ.1.AND.XS(1).EQ.0..AND.
XTM(1,1).EQ.0.) GO TO 18
    WRITE(NAME.'PLMATRIX'.STYPE(IT).J)
X   L1,I1,I2,(XS(L),(TM(I,L),I=I1,I2),L=1,L1)
18  CONTINUE
    WRITE($SYS,3002)  STYPE(IT)
17  CONTINUE
14  CONTINUE
    WRITE($SYS,3003)  DISOX,ITDCX
3000 FORMAT('OBEGIN RETRIEVING DATA FOR ',A8,I8/10X,'NREC =',I10/
X10X,'ELASTIC=',2I5/10X,'INELAST=',2I5/10X,'N2N    =',2I5)
3001 FORMAT('O      MULTIGRP.',A8,'.',A8,'.',I8,'.LIMITS.',A8//
X12X,5('  J  L  I1 I2  ')/12X,5(' -- -- -- -- ')/
X(12X,4I3,3X,4I3,3X,4I3,3X,4I3,3X,4I3))
3002 FORMAT('O      END PROCESSING ',A8,' DATA.')
3003 FORMAT('OEND PROCESSING ',A8,I8/)
    REWIND TAPNEW
    STOP 'NORMAL TERMINATION'
5   STOP 'MULTIGRP VERSION NAMES ON TAPE AND INPUT DO NOT AGREE.'
    END

```

3. User Information

3.1 Input

All input data to module MULTREAD are input on cards. The header card contains the name of the MULTIGRP data set to be processed in columns 1 through 8. Following the header card, as many as forty isotope cards can be included in the input deck. Each of these cards contains the isotope name to be processed in columns 1 through 8 and the integer temperature in °C in columns 9 through 16. All of the isotope cards with non-blank isotope names must precede the first blank card in the data deck.

3.2 Output

The output produced by module MULTDUMP is largely self-explanatory. A printed edit of the input data is included. Several lines are printed as each of the isotopes is processed in turn. At the end of the job an updated tape table of contents is printed and a message indicating normal termination is printed.

3.3 Module Execution

This module is referenced as MEMBER=MULTREAD and PDS=AMD. JOSHUA. The module will execute in 200K bytes of core and will process up to forty isotopes in less than one minute CPU time. An example of the appropriate JCL cards are shown in Figure 2.

3.4 ERROR STOPS

Several self-explanatory error messages are printed by the module.

JOSHUA

DPSTM-500

Title Tape to MULTIGRP Matrix Processor -
Module MULTREAD

Vol.
V

Page
E.5.10

Rev.

Date
9/71

```
//ABCD1234 JOB (1234,LOOO,1,1,0,,,8690-00,L5364-02,J,01),  
//          'your name',MSGLEVEL=1,CLASS=M  
//STEP1 EXEC JOSHUA,MEMBER=MULTREAD,GØSIZE=200K,  
//  BADGE=1234,JØB=AAAA1234,DISP=TMP  
//GØ.FT90FOO1 DD DSN='data set name',DISP=ØLD,KEEP),  
//  UNIT=2400,VOLUME=SER=LA0123,DCB=(RECFM=VBS,  
//  DEN=2,BLKSIZE=1804,LRECL=1800)  
//GØ.SYSIN DD *  
    Data  
/*
```

Figure 2. JCL for MULTREAD Execution.

JOSHUA
DPSTM-500

Title Part F - FEWGRP Data Processing

Vol.
V

Page
F.O.1

Rev.

Date
5/71

Contents

F.1 FEWGRP Editor - Module FEWED.

F.2 FEWGRP Homogenization - Module HØMØCELL.

JOSHUA DPSTM-500	Title Part F - FEWGRP Data Processing	Vol. V	Page F.O.2
		Rev.	Date 5/71

(Blank)

JOSHUA

DPSTM-500

Title FEWGRP Editor - Module FEWED

Vol.
V

Page
F.1.1

Rev.

Date
5/71

Written By

J. M. Sandler

Work Done By

J. M. Sandler
J. W. Stewart

Contents

	<u>Page</u>
1. <u>Theory</u>	F.1.3
1.1 Introduction and Theory	F.1.3
2. <u>Program Logic and Coding Details</u>	F.1.5
2.1 Program Organization	F.1.5
2.2 Symbol Correspondence	F.1.5
2.3 Data Organization and Restrictions	F.1.6
2.4 Special Coding Techniques	F.1.6
3. <u>User Information</u>	F.1.7
3.1 Input	F.1.7
3.2 Output	F.1.7
3.3 Module Execution	F.1.7
3.4 Error Stops	F.1.7

JOSHUA
DPSTM-500

Title FEWGRP Editor - Module FEWED

Vol.
V

Page
F.1.2

Rev.

Date
5/71

(Blank)

1. Theory

1.1 Introduction and Theory

The purpose of module FEWED is to produce a printed edit of neutronics data catalogued in the FEWGRP data set. The FEWGRP data are produced by the JOSHUA lattice physics subsystem. These data are cell spectrum- and composition-weighted. The format of the FEWGRP data set is described in Volume III of this document.

For each FEWED edit the data contained in the record FEWGRP.CØNTENTS are printed. For each FEWGRP version (?VERSN) for which an edit is requested the data contained in

FEWGRP.?VERSN.CØNTENTS

FEWGRP.?VERSN.GRØUPS

are printed. Within a specified version the macroscopic data, microscopic data, delayed neutron data, and anisotropic scattering data are edited for each requested combination of FEWGRP lattice name (?LATT), assembly name (?ASSY), and case identifier (?NCASE). In addition to editing all catalogued data the value of K_{∞} is calculated from macroscopic data and edited. The infinite medium diffusion equation is

$$(\Sigma_c^g + \Sigma_f^g) \phi^g + \sum_{g'} \Sigma_s^{g \rightarrow g'} \phi^{g'} - \sum_{g'} \Sigma_s^{g' \rightarrow g} \phi^{g'} =$$

$$\frac{1}{K_{\infty}} \sum_{g'} \chi^{g' \rightarrow g} \nu \Sigma_f^{g'} \phi^{g'}$$

where:

Σ_c^g = macroscopic capture cross section, group g

Σ_f^g = macroscopic fission cross section, group g

$\Sigma_s^{g' \rightarrow g}$ = macroscopic scattering cross section, group g' to group g

$\chi^{g' \rightarrow g}$ = fraction of fission neutrons appearing in group g due to fission events produced by neutrons in group g'

$\nu \Sigma_f^g$ = macroscopic production cross section,
group g

ϕ^g = neutron flux, group g

In matrix form this equation becomes

$$\underline{A} \underline{\phi} = \frac{1}{K_{\infty}} \underline{P} \underline{\phi}$$

where:

$$[\underline{A}]_{g',g} = \delta_{gg'} [\Sigma_c^g + \Sigma_f^g + \sum_n \Sigma_s^{g \rightarrow n}] - \Sigma_s^{g \rightarrow g'}$$

$$[\underline{P}]_{g',g} = \chi^{g' \rightarrow g} \nu \Sigma_f^{g'}$$

The coefficient matrices are set up in subroutine KCAL and the eigenvalue-eigenvector are calculated in subroutine DALMAT, using a standard "QR - transformation" algorithm.

2. Program Logic and Coding Details

2.1 Program Organization

The purpose of each subroutine of module FEWED is listed in Table I.

2.2 Symbol Correspondence

The mnemonics used for cross section arrays correspond to the FEWGRP record-list descriptions contained in Volume III of this manual. The input data arrays are named RVERSN, RLAT, RASSY, and RNCASE in Subroutine INFED and VERN, LATT, ASSY, and NCASE in other subroutines (for the FEWGRP version, lattice, assembly, and case identification, respectively). Subroutine ISORT uses the mnemonic ID rather than RNCASE.

TABLE I

Purpose of Routines in Module FEWED

<u>Routine Name</u>	<u>Purpose</u>
MAIN	Controls call to other subroutines.
CNTNTS	Edits data from FEWGRP.CØNTENTS.
INFED	Reads user-supplied input data from INPUT.FEWGRP.EDIT and packs input data arrays. Calls ALFIL and ISORT.
ALFIL	Extends input data arrays if user elects 'ALL' option in any column.
ISORT	Sorts input data arrays by version, lattice, assembly, and case identification, in order.
VERCØN	Edits FEWGRP.?VERSN.CØNTENTS
RGRPS	Edits FEWGRP.?VERSN.GRØUPS
LATNTS	Edits FEWGRP.?VERSN.?LATT.?ASSY.CØNTENTS
RMACRØ	Edits macroscopic data. Calls KCAL.
RMICRØ	Edits microscopic data.
RDELD	Edits delayed neutron data.
RANISØ	Edits anisotropic scattering data.

TABLE I (Cont.)

Purpose of Routines in Module FEWED

<u>Routine Name</u>	<u>Purpose</u>
KCAL	Generates removal and production matrices from macroscopic data. Calls DALMAT.
DALMAT	Solves eigenvalue-eigenvector equation for K_{∞} and group fluxes using a standard "QR - transformation" algorithm.

2.3 Data Organization and Restrictions

Most of the data in FEWED are contained in two common blocks. Common block 'IN' contains the arrays RVERSN, RLAT, RASSY, and RNCASE corresponding to FEWGRP version, lattice, assembly, and case identifier, respectively. The first three of these arrays are typed REAL*8, dimensioned 1000, and equivalenced to the REAL*8 array C8(1000,3). RNCASE is typed REAL*8 and dimensioned 1000.

The blank common block contains the scratch array DUM4; typed REAL*4 and dimensioned 10000. This array is partitioned differently in various parts of the module, and is used to transmit data from one routine to another only in the K_{∞} calculation. Each partition may be regarded as internal to the particular subroutine using the common block. These particular uses are described by comment cards in the program labelled "SPECIAL SPECIFICATION STATEMENTS".

2.4 Special Coding Techniques

Subroutine DALMAT and its associated routines is used as a generalized computing package to obtain a solution to the eigenvalue-eigenvector equation in the calculation of K_{∞} . A standard QR transformation technique is employed.

Subroutine ALFIL scans the input data arrays for the characters 'ALL'. If this option is employed the data arrays are extended using data from either FEWGRP.CONTENT, FEWGRP.?VERSN.CONTENT, or FEWGRP.?VERSN.?LATT.?ASSY.CONTENT. The scan for 'ALL' and array extension is processed by version, lattice, assembly, and case identifier, in order. The data is sorted in the same hierarchical structure. If the appropriate contents record is not locatable (i.e., does not exist) the character 'DELETED' are inserted and this entry is not processed.

3. User Information

3.1 Input

Module FEWED requires one user-supplied input data record:

INPUT.FEWGRP.EDIT

This record contains the integer number of table entries and a table of four eight-character names corresponding to FEWGRP version, lattice, assembly, and case identifier for which edits are desired. Any column-entry which contains eight blank characters but has a non-blank entry in any column to its right is assumed to imply ditto from the last previous non-blank entry. The characters 'ALL' (left-adjusted) may be used in any column to imply that all records whose other qualifiers match those specified are to be edited.

A sample input record is shown on the following page.

3.2 Output

All FEWGRP data characterized by a specific version name, lattice name, assembly name, and case identifier are edited, along with appropriate cataloging data. Data associated with a specific version name, or with a specific version/lattice/assembly name set, are edited only once.

3.3 Module Execution

Module FEWED is referenced as MEMBER=FEWED in PDS=AMD.JØSHUA. Required GØSIZE is 240K. No job data set space is required.

3.4 Error Stops

FEWED contains warning messages and one error stop. Messages are self-explanatory.

3401.INPUT.FEWGRP.EDIT

SPECIFICATIONS FOR EDIT OF FEWGRP DATA

PLEASE FILL IN THE FOLLOWING TABLE. 'ALL' MAY BE USED IN ANY COLUMN.

IF ANY LINE IN A COLUMN IS LEFT BLANK, BUT HAS A NON-BLANK ENTRY TO ITS RIGHT, THE LAST NON-BLANK ENTRY IN THE COLUMN WILL BE INSERTED.

INJUDICIOUS USE OF 'ALL' CAN RESULT IN AN UNMANAGEABLE MOUNTAIN OF DATA.

NUMBER OF LINES IN TABLE = : 5

3401.INPUT.FEWGRP.EDIT

SPECIFICATIONS FOR EDIT OF FEWGRP DATA

VERSION	LATTICE	ASSEMBLY	CASE ID
*****	*****	*****	*****
:FG	:LAT	:CELL	:1
:	:	:	:2
:	:	:	:3
:	:	:	:4
:STD4GRP	:MK-14JMS	:MK-14	:ALL

Work Done By

M. M. Anderson
J. W. Stewart

Report Written By

M. M. Anderson

Contents

	<u>Page</u>
1. <u>Theory</u>	F.2.3
1.1 Introduction	F.2.3
1.2 Few-group Flux-Volumes	F.2.3
1.3 Macroscopic Cross Section Data	F.2.3
1.4 Microscopic Cross Section Data	F.2.5
1.5 Delayed Neutron Data	F.2.7
2. <u>Program Logic</u>	F.2.9
2.1 Interfaces	F.2.9
2.2 Calculational Sequence	F.2.9
3. <u>Coding Details</u>	F.2.11
3.1 CØMMØN/ID/	F.2.11
3.2 CØMMØN/LATCEL/	F.2.11
3.3 CØMMØN/FLUX/	F.2.12
3.4 CØMMØN/TEMPØ/	F.2.12
3.5 Main Program HØMØCELL	F.2.12
3.6 Subroutine CØLFUX	F.2.12
3.7 Subroutine HØMMAC	F.2.12
3.8 Subroutine HØMMIC	F.2.14
3.9 Subroutine HØMDEY	F.2.14
4. <u>User Information</u>	F.2.15
4.1 Input	F.2.15
4.2 Output	F.2.19
4.3 Restrictions	F.2.19
4.4 Core Size and Timing	F.2.19
4.5 Error Stops	F.2.21

JOSHUA DPSTM-500	Title FEWGRP Homogenization - Module HØMØCELL	Vol. V	Page F.2.2
		Rev.	Date 5/71

(Blank)

1. Theory

1.1 Introduction

The HØMØCELL module homogenizes cell-averaged few-group macroscopic cross section data, microscopic cross section data, and delayed neutron data for several cells into supercell-averaged data. In general, the homogenization is accomplished by conserving reaction rates.

1.2 Few-group Flux-Volumes

In order to homogenize the cell data, the few-group flux-volumes, must be obtained for each cell. The HØMØCELL module reads the multigroup flux volumes, multigroup energy structure, and few-group energy structure; then collapses the multigroup flux-volumes into few-group flux-volumes. The equation used to collapse the flux-volumes is

$$\phi_{fg,c}^i = \sum_{j \in i} \phi_{mg,c}^j WT_c \quad (1)$$

where:

$j \in i$ means multigroup j contained in few-group i

$\phi_{fg,c}^i$ = few-group i flux-volume in cell c

$\phi_{mg,c}^j$ = multigroup j flux-volume in cell c

WT_c = cell weighting factor which is generally equal to number of type c cells in a supercell.

1.3 Macroscopic Cross Section Data

The few-group macroscopic cross section records contain the group diffusion coefficients, macroscopic cross sections, group-to-group transfer matrices, and the fission spectra. The following formula is used to homogenize the diffusion coefficients and macroscopic cross sections

$$\Sigma_{\text{homo}}^i = \sum_{\text{cells}} \Sigma_c^i \phi_{fg,c}^i / \sum_{\text{cells}} \phi_{fg,c}^i \quad (2)$$

where:

Σ_{homo}^i = homogenized group i diffusion coefficient or macroscopic cross section

Σ_c^i = group i diffusion coefficient or macroscopic cross section in cell c.

The following formula is used to homogenize the group-to-group transfer matrices

$$T_{\text{homo}}^{i \rightarrow j} = \sum_{\text{cells}} T_c^{i \rightarrow j} \phi_{fg,c}^i / \sum_{\text{cells}} \phi_{fg,c}^i \quad (3)$$

where: $T_{\text{homo}}^{i \rightarrow j}$ = homogenized transfer cross section from group i to group j

$T_c^{i \rightarrow j}$ = transfer cross section from group i to group j in cell c.

The following formula is used to homogenize the fission spectra

$$\chi_{\text{homo}}^{i \rightarrow j} = \frac{\sum_{\text{cells}} \chi_c^{i \rightarrow j} (\nu \Sigma_f)_c^i \phi_{fg,c}^i}{\sum_{\text{cells}} (\nu \Sigma_f)_c^i \phi_{fg,c}^i} \quad (4)$$

where:

$\chi_{\text{homo}}^{i \rightarrow j}$ = homogenized fraction of fission neutrons ultimately appearing in group j which result from a fission event produced by a neutron of incident energy i

$\chi_c^{i \rightarrow j}$ = fraction of cell c fission neutrons ultimately appearing in group j which result from a fission event produced by a neutron of incident energy i in cell c

$(\nu \Sigma_f)_c^i$ = product of group i fission cross section and neutron yield in cell c.

A power fraction per cell type is obtained from the fission cross sections and flux-volumes. The formula used to calculate the power fractions per cell type is

$$P_a = \sum_{\text{groups}} \Sigma_{f,a}^i \phi_{fg,a}^i / \sum_{\text{cells}} \sum_{\text{groups}} \Sigma_{f,c}^i \phi_{fg,c}^i \quad (5)$$

where:

P_a = power fraction of type a cells

$\Sigma_{f,c}^i$ = group i fission cross section in cell c.

1.4 Microscopic Cross Section Data

The few-group microscopic cross section data records contain the concentrations, microscopic cross sections, neutron yields, and self-shielding factors by isotope. The following formula is used to homogenize the isotope concentrations

$$N_{k,homo} = \sum_{\text{cells}} N_{k,c} WT_c / \sum_{\text{cells}} WT_c \quad (6)$$

where:

$N_{k,homo}$ = homogenized concentration of isotope k

$N_{k,c}$ = concentration of isotope k in cell c.

The following formula is used to homogenize the microscopic cross sections

$$\sigma_{m,k,homo}^i = \frac{\sum_{\text{cells}} \sigma_{m,k,c}^i N_{k,c} \phi_{fg,c}^i}{N_{k,homo} \sum_{\text{cells}} \phi_{fg,c}^i} \quad (7)$$

where:

$\sigma_{m,k,homo}^i$ = homogenized group i microscopic cross section for reaction type m, isotope k

$\sigma_{m,k,c}^i$ = group i microscopic cross section for reaction type m, isotope k in cell c.

The following formula is used to homogenize the neutron yields

$$v_{k,homo}^i = \frac{\sum_{cells} v_{k,c}^i \sigma_{f,k,c}^i N_{k,c} \phi_{fg,c}^i}{\sum_{cells} \sigma_{f,k,c}^i N_{k,c} \phi_{fg,c}^i}$$

where:

$v_{k,homo}^i$ = homogenized group i neutron yield for isotope k

$v_{k,c}^i$ = group i neutron yield for isotope k in cell c

$\sigma_{f,k,c}^i$ = group i microscopic fission cross section (smooth plus resonance) for isotope k in cell c.

The following formula is used to homogenize the self-shielding factors

$$SF_{k,homo}^i = \frac{\sum_{cells} SF_{k,c}^i (\sigma_{rc,k,c}^i + \sigma_{rf,k,c}^i) N_{k,c} \phi_{fg,c}^i}{\sum_{cells} (\sigma_{rc,k,c}^i + \sigma_{rf,k,c}^i) N_{k,c} \phi_{fg,c}^i} \quad (9)$$

where:

$SF_{k,homo}^i$ = homogenized group i self-shielding factor for isotope k

$SF_{k,c}^i$ = group i self-shielding factor for isotope k in cell c

$G_{rc,k,c}^i$ = group i resonance capture cross section for isotope k in cell c

$G_{rf,k,c}^i$ = group i resonance fission cross section for isotope k in cell c.

1.5 Delayed Neutron Data

The few-group delayed neutron records contain the delayed neutron time constants, inverse group velocities, values of the product of the prompt neutron yield and fission cross section, and the delayed fission spectra.

The product of the prompt neutron yield and fission cross section is homogenized by the same flux-volume weighting used to homogenize the macroscopic cross sections (see Section 1.3). The delayed neutron fission spectra are homogenized by a weighting with the total neutron production rate. The following formula is used to accomplish this homogenization

$$x_{j,n,homo} = \frac{\sum_{\text{cells}} x_{j,n,c} \sum_{\text{groups}} \phi_{fg,c}^i (\nu \Sigma_f)_c^i}{\sum_{\text{cells}} \sum_{\text{groups}} \phi_{fg,c}^i (\nu \Sigma_f)_c^i}$$

where:

$x_{j,n,homo}$ = homogenized fraction of total fissions neutrons which appear in delayed family n, energy group j

$x_{j,n,c}$ = fraction of total cell c fission neutrons which appear in delayed family n, energy group j in cell c

$(\nu \Sigma_f)_c^i$ = product of group i fission cross section and neutron yield in cell c.

JOSHUA DPSTM-500	Title FEWGRP Homogenization - Module HØMØCELL	Vol. V	Page F.2.8
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title FEWGRP Homogenization - Module HØMØCELL	Vol. V	Page F.2.9
		Rev.	Date 5/71

2. Program Logic

2.1 Interfaces

HØMØCELL reads the following records ('FEWGRP'.FGVER.DLAT(I) →V):

```

'INPUT'. 'HØMØCELL'. 'JØB'. $JØB
'INPUT'. 'HØMØCELL'. 'CASES'. CASNAM(I)
V.DCEL(I). 'CØNTENTS'
'LATTICE'. DLAT(I). VLAT(I). 'FLUX'. DCEL(I). VCEL(I). NSØ(I)
'LATTICE'. DLAT(I). VLAT(I). 'INPUT'
'MULTIGRP'. VMULT. 'GRØUPS'
'FEWGRP'. FGVER. 'GRØUPS'
V.DCEL(I). 'MACRØ'. NCASE(I)
V.DCEL(I). 'MICRØ'. NCASE(I)
V.DCEL(I). 'DELAYED'. NCASE(I).

```

HØMØCELL writes the following records:

```

V.CASNAM(I). 'MACRØ'. 'HØMØCELL'
V.CASNAM(I). 'MICRØ'. 'HØMØCELL'
V.CASNAM(I). 'DELAYED'. 'HØMØCELL'
'OUTPUT'. 'HØMØCELL'. 'PFRAC'. FGVER.DLAT(I). CASNAM(I).

```

2.2 Calculational Sequence

The HØMØCELL module consists of a main program and four subroutines. The main program reads the number of cases to be performed (number of homogenizations), identifies the cells to be homogenized in each case, and calls the following subroutines in sequence for each cell that is to be homogenized.

- 1) Subroutine CØLFUX
- 2) Subroutine HØMMAC
- 3) Subroutine HØMMIC
- 4) Subroutine HØMDEY

During the last call to the subroutine for each case, the homogenized data is written into the FEWGRP data set.

JOSHUA DPSTM-500	Title FEWGRP Homogenization - Module HØMØCELL	Vol. V	Page F.2.10
		Rev.	Date 5/71

(Blank)

3. Coding Details

The HØMØCELL module is divided into the following parts:

1. CØMMØN/ID/
2. CØMMØN/LATCEL/
3. CØMMØN/FLUX/
4. CØMMØN/TEMPØ/
5. Main Program HØMØCELL
6. Subroutine CØLFUX
7. Subroutine HØMMAC
8. Subroutine HØMMIC
9. Subroutine HØMDEY.

3.1 CØMMØN/ID/

This area contains the following data:

- ICASE - Index of case being processed
- NCELLS - Number of cells in the case being processed
- ICELL - Cell being processed
- NØUT - Output data set reference number
- CASNAM(25) - Array of case names
- FGVER - Few-group energy version of case being processed.

3.2 CØMMØN/LATCEL/

This area contains the following data:

- DLAT(10) - Array of lattice names for case being processed
- VLAT(10) - Array of lattice versions for case being processed
- DCEL(10) - Array of cell names for case being processed
- VCEL(10) - Array of cell versions for case being processed
- NSØ(10) - Array of search indices for case being processed
- NCASE(10) - Array of few-group case identification numbers for case being processed.

JOSHUA DPSTM-500	Title FEWGRP Homogenization - Module HØMØCELL	Vol.	Page
		V	F.2.12
		Rev.	Date
			5/71

3.3 CØMMØN/FLUX/

This area contains the following data:

FGFLUX(10)	-	Few-group flux-volumes for cell being processed
TFGFUX(10)	-	Total few-group flux-volumes for case being processed
SIGFNU(10)	-	Product of neutron yield and macroscopic fission cross section for cell being processed
WT(10)	-	Cell weighting factors which are generally equal to the number of cells of a given type in a supercell.

3.4 CØMMØN/TEMPØ/

This area is used for temporary storage.

3.5 Main Program HØMØCELL

The flow diagram of module HØMØCELL is given in Figure 1.

3.6 Subroutine CØLFUX

This subroutine reads the multigroup fluxes, collapses them into few-group fluxes, calculates the total few-group fluxes (i.e., sum of few-group fluxes over the cells), and outputs the few-group fluxes.

3.7 Subroutine HØMMAC

This subroutine reads the macroscopic cross section data, homogenizes the data, calculates a power fraction per cell, and outputs the homogenized macroscopic data and power fractions.

Figure 1. Flowchart for HØMØCELL.

3.8 Subroutine HØMMIC

This subroutine reads the microscopic cross section data, homogenizes the data, and outputs the homogenized microscopic data.

3.9 Subroutine HØMDEY

This subroutine reads the delayed neutron data, homogenizes the data, and outputs the homogenized delayed neutron data.

4. User Information

4.1 Input

HØMØCELL homogenizes the few-group cross section cell data that have been generated by other modules and stored in the FEWGRP data set. In addition to the FEWGRP data set, HØMØCELL reads multigroup cell flux-volumes and multigroup version names from the LATTICE data set, and the multigroup energy structures from the MULTIGRP data set.

If the information on the aforementioned data sets is available, the user only needs to supply the following two types of input records:

- 1) Name: 'INPUT'. 'HØMØCELL'. 'JØB'. \$JØB
 List: NCASES, (CASNAM(I), I=1, NCASES)
 \$JØB = JØSHUA job name R*8
 NCASES = number of cases performed (a case I*4
 is a single homogenization of
 several cells)
 CASNAM(I) = array of case names. R*8

- 2) Name: 'INPUT'. 'HØMØCELL'. 'CASES'. CASNAM(I)
 List: NCELLS, IMACRØ, IMICRØ, IDELAY,
 FGVER, (DLAT(J), DCEL(J), NCASE(J),
 NSØ(J), WT(J), J=1, NCELLS)
 CASNAM(I) = name of Ith case R*8
 NCELLS = number of cells in Ith case I*4
 IMACRØ = control flag equal to 1 if I*4
 homogenization of macroscopic
 records is desired.
 IMICRØ = control flag equal to 1 if I*4
 homogenization of microscopic
 records is desired
 IDELAY = control flag equal to 1 if I*4
 homogenization of delay neutron
 records is desired
 FGVER = few-group version name for Ith R*8
 case
 DLAT(J) = lattice name of Jth cell in Ith R*8
 case.

DCEL(J) = cell name of Jth cell in Ith case R*8
NCASE(J) = few-group case identification number I*4
 of Jth cell in Ith case
NSØ(J) = lattice search index of Jth cell in I*4
 Ith case
WT(J) = weighting factor of Jth cell in Ith R*4
 case.

One type - 1 input record and NCASES type - 2 input records
are required for each execution of HØMØCELL. Sample
input records are attached.

JOSHUA
DPSTM-500

Title FEWGRP Homogenization - Module HOMOCCELL

Vol.
V

Page
F.2.17

Rev.

Date
5/71

3401.INPUT.HOMOCCELL.JOB.TEST3947

JOB INPUT RECORD FOR THE MODULE HOMOCCELL

NUMBER OF CASES IN THIS JOB IS : 1 (<26)

3401.INPUT.HOMOCCELL.JOB.TEST3947

SUPPLY A CASENAME FOR EACH CASE IN THIS JOB BELOW

CASE NUMBER

CASE NAME

: 1

: TEST-ONE

3401.INPUT.HCMOCELL.CASES.TEST-ONE

THE NUMBER OF CELLS TO BE HOMOGENIZED IN THIS CASE

IS : 2 (<11).

IF A HOMOGENIZATION OF THE MACRO DATA IS DESIRED ENTER 1 :1.

IF A HOMOGENIZATION OF THE MICRO DATA IS DESIRED ENTER 1 :1.

IF A HCMOGENIZATION OF THE DELAYED DATA IS DESIRED ENTER 1 :1.

FEWGROUP VERSION NAME IS :FG .

3401.INPUT.HCMOCELL.CASES.TEST-ONE

FOR EACH CELL ENTER THE DATA LISTED BELOW

(WEIGHT NORMALLY SPECIFIES THE # OF CELL TYPES IN A SUPERCELL):

DLAT	DCEL	NCASE	NSO	WEIGHT
----	----	-----	----	-----
:LATTICE	:CELL	: 1	: 1	: 1.0E 00
:LATTICE	:CELL	: 2	: 2	: 2.0E 00

4.2 Output

The homogenized few-group data is written into the FEWGRP data set. The lattice names are retained for lattice identification and the case names (CASNAM) are used to identify the homogenized supercell. The homogenized data is then written on the following records:

```
('FEWGRP'.FGVER.DLAT(J).CASNAM(J)→Q)
```

```
Q.'MACRØ'. 'HØMØCELL'
```

```
Q.'MICRØ'. 'HØMØCELL'
```

```
Q.'DELAYED'. 'HØMØCELL'.
```

The power fractions per cell type are written on the following record:

```
'ØUTPUT'. 'HØMØCELL'. 'PFRAC'. FGVER.DLAT(J).CASNAM(J).
```

4.3 Restrictions

The following bounds on problem size are set by the storage allocations in HØMØCELL:

Maximum number of cases = 25

Maximum number of cells per case = 10

Maximum number of isotopes per case = 50

Maximum number of reaction types = 15.

Because of the normalization of the flux-volumes performed in the module RAHAB which outputs the FEWGRP data set, all cells homogenized in a case must have the same lattice name. In addition, all cells to be homogenized in a case must have the same few-group version name.

4.4 Core Size and Timing

The HØMØCELL module requires 200K of core, 1 block of job data set space, and a GTIME of 1 minute. The size of the subroutines and timings for one case with three cells is given in Table I (the case has 10 multigroups and 4 few-groups).

JOSHUA DPSTM-500	Title FEWGRP Homogenization - Module HØMØCELL	Vol. V	Page F.2.20
		Rev.	Date 5/71

TABLE I

SIZE AND TIMINGS FOR HØMØCELL SUBROUTINES

(Size in bytes; time in CPU seconds.)

<u>Subprogram</u>	<u>Size</u>	<u>Total Time</u>	<u>Average Time</u>	<u>Entries</u>
MAIN	5664	.36664	.36664	1
CØLFUX	4926	.19999	.06666	3
HØMMAC	5752	.21665	.07222	3
HØMMIC	35832	.33331	.11110	3
HØMDEY	6534	.08333	.02778	3
/ID/	224			
/LATCEL/	400			
/FLUX/	160			
/TEMPØ/	42000			
JØS	<u>93068</u>			
Totals	194560	<u>1.19992</u>		<u>13</u>

4.5 Error Stops

All error stops are handled by the JOSHUA operating system and are self-explanatory. When errors occur that force the termination of a case, a message is written and the next case is started.

JOSHUA DPSTM-500	Title FEWGRP Homogenization - Module HØMØCELL	Vol. V	Page F.2.22
		Rev.	Date 5/71

(Blank)

JOSHUA DPSTM-500	Title Part G - RILIB Data Processing	Vol. V	Page G.O.1
		Rev.	Date 5/71

Contents

G.1 RILIB Editor - Module RILED.

JOSHUA DPSTM-500	Title Part G - RILIB Data Processing	Vol. V	Page G.O.2
		Rev.	Date 5/71

(Blank)

JOSHUA

DPSTM-500

Title Resonance Integral Library (RILIB)
Editor - Module RILED.

Vol. V	Page G.1.1
Rev.	Date 5/71

Written By

J. W. Stewart

Work Done By

J. W. Stewart

Contents

	<u>Page</u>
1. <u>Theory</u>	G.1.3
1.1 Introduction	G.1.3
1.2 Data Representation	G.1.3
2. <u>Program Logic and Coding Details</u>	G.1.5
2.1 Program Structure	G.1.5
2.2 Data Arrays	G.1.5
3. <u>User Information</u>	G.1.7
3.1 Input	G.1.7
3.2 Output	G.1.7
3.3 Module Execution	G.1.7
3.4 Error Stops	G.1.7

JOSHUA DPSTM-500	Title Resonance Integral Library (RILIB) Editor - Module RILED.	Vol. V	Page G.1.2
		Rev.	Date 5/71

(Blank)

1. Theory

1.1 Introduction

The purpose of Module RILED is to produce a printed edit of neutronics data catalogued in the RILIB (resonance integral library) data set.

1.2 Data Representation

A description of the RILIB data set is contained in Volume III of this manual. Cataloguing information is contained in the record RILIB.?DCEL.?VCEL.CØNTENTS, and resonance integrals in the record RILIB.?DCEL.?VCEL.?DISØ.?DREG.RESULTS.

JOSHUA DPSTM-500	Title Resonance Integral Library (RILIB) Editor - Module RILED	Vol.	Page
		V	G.1.4
		Rev.	Date
			5/71

(Blank)

2. Program Logic and Coding Details

2.1 Program Structure

Module RILED is a single routine (Subroutine MAIN). The user-specified control data record (INPUT.RILIB.EDIT) is read and non-blank entries are packed. A printed edit of the cell names and versions for which resonance integral edits are requested is produced. RILED then loops over each cell name and version, reading resonance data records and editing the contents thereof.

2.2 Data Arrays

The cell names and versions to be edited are stored in the REAL*8 arrays DCEL and VCEL, respectively; each dimensioned to length 20. For each requested edit for a cell name/version the eight-character isotope names and region names are stored in the REAL*8 arrays DISØ and DREG, respectively; and temperatures (°C) and concentrations (atoms/barn-cm) are stored in the REAL*4 arrays TDC and CØNC, respectively. There may be up to 50 isotope/region pairs in any cell edit. For each isotope/region pair there are NRES resonances, the data for which are stored in the REAL*4 arrays ER (resonance energy, eV), RA (absorption resonance reaction rate for one resonance, isotope, and region based on a unit slowing down density in the cell), RF (fission resonance reaction rate), and SF (resonance absorption self-shielding factor). There may be up to 400 resonances per isotope.

JOSHUA DPSTM-500	Title Resonance Integral Library (RILIB) Editor - Module RILED	Vol. V	Page G.1.6
		Rev.	Date 5/71

3. User Information

3.1 Input

Only one user-specified input data record is required, namely:

INPUT.RILIB.EDIT

This record contains a table of up to 20 cell names and cell versions which are to be processed. Each entry is a left-adjusted, eight-character name. Blank entries are ignored. A sample input record is attached.

The resonance data is read from two types of RILIB records as in Section 1.2.

3.2 Output

Printed output includes a table of edits requested and a full printout of tabular data contained in RILIB.

3.3 Module Execution

Module RILED is referenced as MEMBER=RILED in PDS=AMD.JØSHUA. Required GØSIZE is 120K bytes. No job data set space is required.

3.4 Error Stops

Several self-explanatory error stops are programmed.

JOSHUA
DPSTM-500

Title Resonance Integral Library (RILIB)
 Editor - Module RILED

Vol. V

Page
G.I.8

Rev.

Date
5/71

3401.INPUT.RILIB.EDIT

***** RILIB EDIT INPUT *****

__CASE__	__CELL__	_VERSN__	__CASE__	__CELL__	_VERSN__
: 1	: MARK5R	: SE-X	: 2	: MK-30A	: SERE1
: 3	:	:	: 4	:	:
: 5	:	:	: 6	:	:
: 7	:	:	: 8	:	:
: 9	:	:	:10	:	:
:11	:	:	:12	:	:
:13	:	:	:14	:	:
:15	:	:	:16	:	:

3401.INPUT.RILIB.EDIT

***** RILIB EDIT INPUT *****

__CASE__	__CELL__	_VERSN__	__CASE__	__CELL__	_VERSN__
:17	:	:	:18	:	:
:19	:	:	:20	:	:

JOSHUA
DPSTM-500

Title Part I - Engineering Data Processing

Vol.
V

Page
I.O.1

Rev.

Date
5/71

Contents

I,1 PRØPERTY Data Processing: Routine PRØPTY

JOSHUA DPSTM-500	Title Part I - Engineering Data Processing	Vol. V	Page I.O.2
		Rev.	Date 5/71

(Blank)

JOSHUA

DPSTM-500

Title PROPERTY Data Processing: Routine PRØPTY

Vol.
V

Page
I.1.1

Rev.

Date
5/71

Written By

M. R. Buckner

Work Done By

M. M. Anderson
M. R. Buckner
J. H. Hightower

Contents

	<u>Page</u>
1. <u>Theory</u>	I.1.3
1.1 Introduction	I.1.3
1.2 Representation of Data	I.1.3
2. Program Logic and Coding Details	I.1.5
3. User Information	I.1.7
3.1 Argument List	I.1.7
3.2 Error Conditions	I.1.7

JOSHUA DPSTM-500	Title PRØPERTY Data Processing: Routine PRØPTY	Vol. V	Page I.1.2
		Rev.	Date 5/71

(Blank)

1. Theory

1.1 Introduction

The purpose of the routine PRØPTY is to process correlation parameters contained in the PRØPERTY data sets and return the value of a property to a calculational module. The returned property value is in the units specified in Volume III of this document and also in the PRØPERTY data records.

1.2 Representation of Data

A detailed description of the PRØPERTY data records is given in Volume III of this document. The property data are represented by polynomial correlations in one or more engineering state variables. Correlations in a single engineering variable are catalogued by property name. Within a particular property record, the correlations are listed by material name. Multi-variable correlations are catalogued by both the property name and material name.

JOSHUA DPSTM-500	Title PRØPERTY Data Processing: Routine PRØPTY	Vol. V	Page I.1.4
		Rev.	Date 5/71

(Blank)

2. Program Logic and Coding Details

The engineering data processing routine PRPTY is a collection of subroutines which work together to process the PROPERTY data. These subroutines are described below:

- PRPTY - This subroutine searches the PROPERTY records for the proper correlation data and returns a value for the engineering property desired. In order to minimize disk accesses, the property correlation data for a material is held in core after each call to PRPTY for a different engineering property. These data are then readily available for use in subsequent calls for a property value. Core space for twenty single variable correlations and five multi-variable correlations is reserved. If additional correlations are required for a calculation, the data is simply read in and processed. A check is made on the value of the engineering variables to determine if they are within the range of applicability of the available correlations for a material. If the variables are not within the proper range, a message stating this condition is supplied and the property value at the range extreme closest to the expansion variables is returned.
- VARCON - This subroutine is called from PRPTY to check the units on the engineering variables to be used in the calculation of the property value. The variables are returned with the proper units.
- ESP - This subroutine is called from PRPTY to compute the weight composition or enrichment of an element or isotope in a material. The definition of the variable is determined from the record 'PROPERTY'. ?VERSN.'VARIABLE'. The value of the variable is determined from the record 'OUTPUT'. 'CHECK'. 'MATERIAL'. ?DCEL.?VCEL.?MATNAM where DCEL is the name of the assembly for which the calculation is being performed, VCEL is the version name and MATNAM is the material name. The module CHECK is an assembly processing program which produces the above records as output. A record for a particular material contains the concentrations of the elements and isotopes that make up the material. This subroutine automatically executes the CHECK module unless a record for the assembly and material already exists. The input records required by CHECK are the ASSEMBLY records for the assembly in question, the 'NUCPARAM'. 'ISOTOPES' record, and

JOSHUA DPSTM-500	Title PRØPERTY Data Processing: Routine PRØPTY	Vol.	Page
		v	I.1.6
		Rev.	Date
			5/71

and the record 'INPUT'. 'CHECK'. 'ASP' which is generated by this subroutine. It is important to note that CHECK requires approximately 60K bytes of core for execution. Thus, it will be necessary to overlay this module in most applications. If the ASSEMBLY records are not used in a calculation, the user can physically create the above records for any materials whose properties depend on enrichment or weight composition. Or, by specifying the assembly name to be 'TEMP', the call to this routine is bypassed and the enrichment or weight composition should be passed through the argument list to PRØPTY.

- CHAR - This subroutine is called from ESP to perform character interpretation on the isotope and element names in the CHECK output records.

3. User Information

3.1 Argument List

The calling statement for PRØPTY is:

CALL PRØPTY(DCEL,VCEL,PRØP,MAT,VERN,VARNAM,VARUT,VAR,VALUE,ICC). The arguments passed are described in Table I. The material name (MAT) corresponds to physics designation for materials as specified in the ASSEMBLY records. The corresponding engineering material names by which the PRØPERTY data is catalogued are found in the record 'ASSEMBLY'.DCEL.VCEL.'MATERIAL'. 'DENSITY'.

3.2 Error Conditions

The condition code ICC is normally equal to zero. For a situation in which the information needed to determine a property is not available, the condition code is set equal to one. A message is supplied explaining the difficulty. It is the option of the user as to what action should be taken when this condition arises.

JOSHUA DPSTM-500	Title PRØPERTY Data Processing: Routine PRØPTY	Vol. V	Page I.1.8
		Rev.	Date 5/71

TABLE I

DESCRIPTION OF ARGUMENTS PASSED TO PRØPERTY

<u>ARGUMENT</u>	<u>DESCRIPTION</u>	<u>TYPE</u>
DCEL	Assembly name.	R*8
VCEL	Assembly version name.	R*8
PRØP	Property name. Allowed names are given in Volume III.	R*8
MAT	Material name.	R*8
VERSN	PRØPERTY Data Set Version name.	R*8
VARNAM	Names of engineering variables passed (<6). Allowed names are given in Table II.	R*8
VARUT	Units of each engineering variable . Allowed units are given in Table II.	R*8
VAR	Value of each engineering variable.	R*4
VALUE	Value of property to be returned.	R*4
ICC	Condition code.	I*4

TABLE II

CORRELATION VARIABLES AND ALLOWED UNITS

<u>VARIABLE</u>	<u>NAME</u>	<u>ALLOWED UNITS</u>	<u>NAME</u>
Temperature	TEMP	°C	DEG C
		°F	DEG F
		°K	DEG K
		°R	DEG R
Pressure	PRESS	psia	PSIA
		psig	PSIG
		psfa	PSFA
		psfg	PSFG
		atm	ATM
		dyne/cm ²	D/CM2
Enrichment	ENRICH	newton/m ²	N/M2
		atom percent	ATM PCNT
Weight Composition	WGT CØMP	atom fraction	ATM FRAC
		weight percent	WGT PCNT
		weight fraction	WGT FRAC