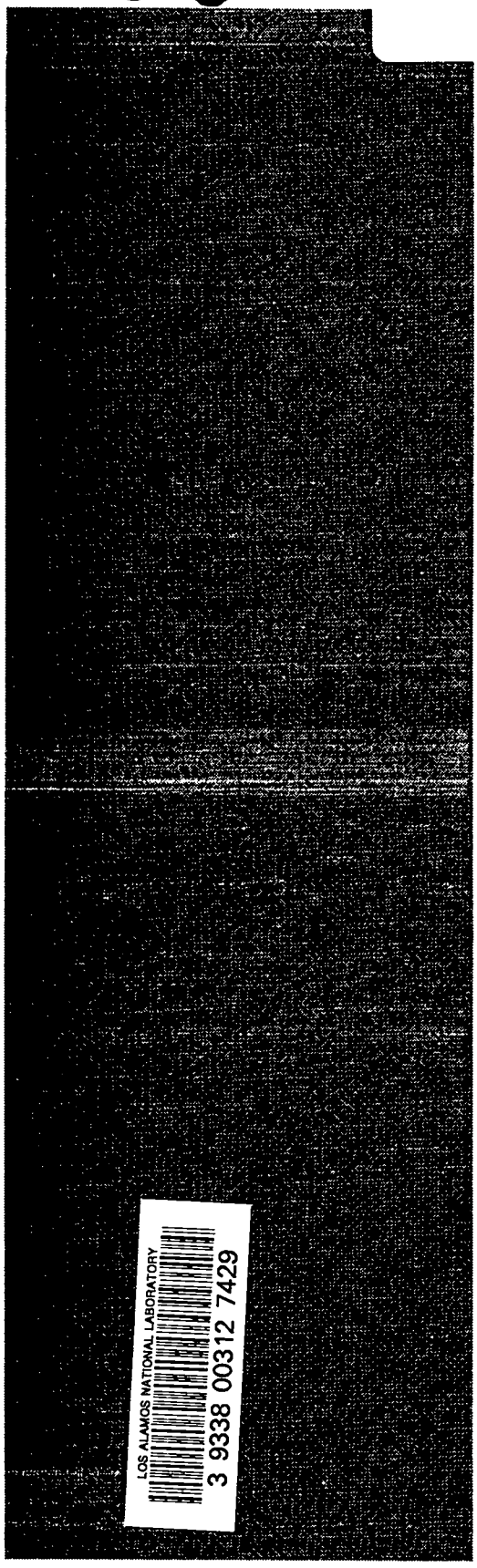
LA-9184-M, Rev.
Manual

c. 3



Revised User's Manual for ONEDANT: A Code Package for One-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport

Los Alamos

Los Alamos National Laboratory is operated by the University of California for the United States Department of Energy under contract W-7405-ENG-36.

Prepared by Helen Byers, Group X-6
Edited by Patricia W. Mendius, Group IS-11

An Affirmative Action/Equal Opportunity Employer

LA-9184-M, Rev.
Manual

UC-705
Issued: December 1989

Revised User's Manual for ONEDANT: A Code Package for One-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport

R. Douglas O'Dell
Forrest W. Brinkley, Jr.
Duane R. Marr
Raymond E. Alcouffe

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

Preface

Since the ONEDANT Code Package was first released in 1982, it has undergone numerous changes in the form of bug fixes, modification to improve the code's robustness, and new features and capabilities. In addition, the current code package contains the TWODANT and TWOHEX Solver Modules, as well as the ONEDANT Solver Module.

This manual represents a complete revision of the original user's manual published as Los Alamos National Laboratory report LA-9184-M, "User's Manual for ONEDANT: A Code Package for One-Dimensional, Diffusion-Accelerated, Neutral-Particle Transport", (Feb. 1982). Several new sections have been added to the new manual, and it has been restructured to improve its readability.

Although this manual's detailed focus is on the one-dimensional, discrete ordinates Solver, there is much general information on the structure of the overall package, the manner in which input is supplied by the user and multigroup cross section libraries accepted, the manner in which nuclides are mixed, how edits are performed, etc. These general features apply to any use of the package and its various Solver Modules. Accordingly, this manual is an essential reference for users of ONEDANT, TWODANT, or TWOHEX.

CONTENTS

REVISED USER'S MANUAL FOR ONEDANT:

A CODE PACKAGE FOR ONE-DIMENSIONAL, DIFFUSION-ACCELERATED, NEUTRAL-PARTICLE TRANSPORT

by

R. Douglas O'Dell, Forrest W. Brinkley, Jr., Duane R. Marr, and Raymond E. Alcouffe

ABSTRACT

Program Identification: ONEDANT

Computer for which Program is designed: Cray X-MP and Cray Y-MP, but the program has been implemented and run on VAX, CDC Cyber 205, large IBM, and Cray-1 computers.

Function: ONEDANT solves the one-dimensional multigroup transport equation in plane, cylindrical, spherical, and two-angle plane geometries. Both regular and adjoint, inhomogeneous and homogeneous (k_{eff} and eigenvalue search) problems subject to vacuum, reflective, periodic, white, albedo, or inhomogeneous boundary flux conditions are solved. General anisotropic scattering is allowed and anisotropic inhomogeneous sources are permitted.

Method of Solution: ONEDANT numerically solves the one-dimensional, multigroup form of the neutral-particle, steady-state form of the Boltzmann transport equation. The discrete-ordinates approximation is used for treating the angular variation of the particle distribution and the diamond-difference scheme is used for phase space discretization. Negative fluxes are eliminated by a local set-to-zero-and-correct algorithm. A standard inner (within-group) iteration, outer (energy-group-dependent source) iteration technique is used. Both inner and outer iterations are accelerated using the diffusion synthetic acceleration method.

Restrictions: The code is thoroughly variably dimensioned with a flexible, sophisticated data management and transfer capability. Originally designed for the CDC-7600 computer, the code is structured for a three-level hierarchy of data storage: a

small, fast core central memory (SCM), a fast-access, peripheral large core memory (LCM), and random-access peripheral storage. (For computing systems based on a two-level hierarchy of data storage - a large fast core and random-access peripheral storage - a portion of fast core is designated as a simulated LCM to mimic the three-level hierarchy). Random-access storage is used only if LCM (or simulated LCM) storage requirements are exceeded. Normally, an SCM of about 25,000 words of storage and an LCM (or simulated LCM) of a few hundred thousand words or less storage is sufficient to eliminate the need for using random-access storage.

Running Time: Running time is directly related to problem size and to the computer's central processor and data transfer speeds. On the Cray X-MP, a 69 energy group [with 40 thermal (upscatter) groups], S_{16}, P_3 scatter, 500 space-point k_{eff} calculation for a light water reactor requires about 48 sec CPU time.

A 30 energy group, S_{16}, P_4 scatter, 30 space-point, fixed surface source, detector efficiency problem requires about 1.4 sec CPU time on the Cray X-MP.

A 42 energy group, S_{32}, P_3 scatter, 200 space-point; coupled neutron/gamma shielding problem requires about 5 sec CPU time on the Cray X-MP.

Generally then, on the Cray X-MP, the running times for ONEDANT will range from a second to about a minute.

Unusual Features of the Program: The code package is modularly structured in a form that separates the input and the output (or edit) functions from the main calculational (or solver) section of the code. Thus, the package consists of an Input Module, an Edit Module, and one or more Solver Modules. Usually when the code package is provided to the user it will contain the Input and Edit Modules together with three distinct Solver Modules. The ONEDANT Solver Module is the one described in this manual. The TWODANT Solver Module performs two-dimensional calculations in (r,z), (x,y), and (r,θ) geometries. The TWOHEX Solver Module performs two-dimensional calculations on an equilateral triangle spatial mesh. The code makes use of binary, sequential data files, called interface files, to transmit data between modules and submodules. Standard interface files whose specifications have been defined by the Reactor Physics Committee on Computer Code Coordination are accepted, used, and created by the code. A free-field card-image input capability is provided for the user. The code provides the user with considerable flexibility in using both card-image or sequential file input and also in controlling the execution of both modules and submodules. Separate versions of the code package exist for short-word and long-word computers.

Programming Languages: The program is written in standard FORTRAN 77 language.

Machine Requirements: For CDC-7600 and similar computers a 50,000-word small core memory (SCM) and large core memory (LCM) are required. For computers with only a single fast core, the fast core size must be sufficiently large to permit

partitioning into an SCM and simulated LCM. Random-access auxiliary storage may occasionally be required if LCM (or simulated LCM) storage is insufficient for the problem being executed.

Material Available: Source deck (about 125,000 card-images), sample problems, and this manual have been submitted to the Radiation Shielding Information Center. The total package containing the three Solver Modules (ONEDANT, TWODANT, and TWOHEX) is collectively called the TWODANT Code Package.

I. INTRODUCTION

The ONEDANT code package is part of a modular computer program package designed to solve the time-independent, multigroup discrete ordinates form of the Boltzmann transport equation in several different geometries. The modular construction of the package separates the input processing, the transport equation solving, and the post processing (or edit) functions into distinct, independently executable code modules: the Input Module, one or more Solver Modules, and the Edit Module, respectively. The Input and Edit Modules are general in nature and are common to any of the Solver Modules that may be provided in the overall package. The ONEDANT Solver Module contains a one-dimensional (slab, cylinder, and sphere), time-independent transport equation solver using the standard diamond-differencing method for space/angle discretization. Also in existence are Solver Modules named TWODANT and TWOHEX. The TWODANT Solver Module solves the time-independent transport equation using the diamond-differencing method for space/angle discretization in (r,z), (x,y), and (r,θ) geometries. The user's guide for TWODANT is provided in Ref. 1. The TWOHEX Solver Module solves the time-independent transport equation on an equilateral triangle spatial mesh. The user's guide for TWOHEX is provided in Ref. 2.

This manual is devoted to the code package consisting of the standardized Input and Edit Modules together with the ONEDANT one-dimensional transport Solver. Throughout this manual we will refer to this package as the ONEDANT code package. When describing the user input for the Input and Edit Modules, we will restrict ourselves only to the user input required for and usable by the ONEDANT Solver Module.

Some of the major features included in the ONEDANT code package are:

(1) a free-field format card-image input capability designed with the user in mind;

(2) highly sophisticated, standardized, data- and file-management techniques as defined and developed by the Committee on Computer Coordination (CCCC) and described in Ref. 3; both sequential file and random-access file handling techniques are used;

(3) the use of a diffusion synthetic acceleration scheme to accelerate the iterative process in the Solver Module;

(4) direct (forward) or adjoint calculational capability;

(5) standard plane, two-angle plane, cylindrical or spherical geometry options;

(6) arbitrary anisotropic scattering order;

(7) vacuum, refiective, periodic, white, albedo, or surface source boundary condition options;

(8) inhomogeneous (fixed) source or k_{eff} calculation options as well as time-absorption (alpha), nuclide concentration, or dimensional search options;

(9) "diamond-differencing" for solution of the transport equation;

(10) user fiexibility in using both card-image or sequential file input;

(11) user fiexibility in controlling the execution of both modules and submodules; and

(12) extensive, user-oriented error diagnostics.

ONEDANT is a large, very fiexible code package. Great effort has been devoted to making the code highly user-oriented. Simple problems can be easily run and many of the code options can be ignored by the casual user. At the same time numerous options for selective and sophisticated executions are available to the more advanced user. In all cases, redundancy of input has been minimized, and default values for many input parameters are provided. The code is designed to be "intelligent" and to do much of the work for the user. The input is designed to be meaningful, easily understood, easily verified, and easy to change. The printed output is well documented with liberal use of descriptive comments and headings. In short, ONEDANT was designed to be fun to use.

Chapter II of this manual provides the user with an overview of the code package. Included are sections on programming practices and standards, code package structure, and functional descriptions of the three principal modules comprising the package.

Chapter III provides a brief development of the multigroup, discrete-ordinates form of the diamond-differenced Boltzmann transport equation in one-dimensional geometries. Also included is a simplified description of the iterative procedure used in solving the transport equations and the application of the diffusion synthetic acceleration method.

Chapter IV presents the card-image input format rules for the user.

Chapter V provides the card-image input specifications for ONEDANT. First is given an overview of the specification of input including descriptive examples. Next is a "mini-specification" sheet on which are listed all the available input arrays arranged by input block. This sheet is very useful to the user in organizing his input. For the more experienced user, the mini-specification sheet is frequently all that is needed for him to specify his input. Following the mini-specification sheet is a moderately detailed description of all the input parameters and arrays.

Chapters II, IV, and V should be read by all first-time users of ONEDANT.

Chapter VI provides extra details on some of the Controls and Dimensions input parameters found in Block I of the code package input.

Chapter VII gives amplified descriptions of some of the Geometry-related input parameters contained in Block II of the input.

Chapter VIII provides information on multigroup cross section data/libraries and their use together with a description of special cross section file-writing capabilities provided by the code as controlled by input parameters in Block III of the code package input.

Chapter IX gives further details on Material Mixing and the Assignment of Materials to Zones as specified in Block IV of the input.

Chapter X is devoted to pertinent details regarding the Solver Module. It contains numerous sections providing specific detailed information needed by the user to more fully understand some of the Block V (Solver) input. It also provides details related to the actual execution of the Solver Module.

Chapter XI presents details related to the Edit Module of the package. Both input and execution-control options for this module are described in detail.

Chapter XII describes the procedure for stacking multiple problems into a single run.

Chapter XIII provides details on some of the more sophisticated options for controlling the execution of modules and submodules in the package.

In Chapter XIV error diagnostics/messages contained in the code package are described. Several examples of errors and the resulting error messages are provided.

Three appendices are also included in the manual. Appendix A provides the file descriptions for the code-dependent, binary, sequential interface files generated by and used in the ONEDANT code package. File descriptions for the CCCC standard interface files are not provided but can be found in Ref. 3. Appendix B provides several sample problems for the user. Appendix C provides file descriptions for the two ASCII files EDTOGX and EDTOUT which contain Edit Module output in eye-readable form.

II. OVERVIEW OF THE ONEDANT CODE PACKAGE

The ONEDANT code package is a computer program designed to solve the one-dimensional, multigroup, discrete-ordinates form of the neutral-particle Boltzmann transport equation. It was developed as a modular code package consisting of three modules: an Input Module, a Solver Module, and Edit Module.

In this chapter is provided a discussion of the general programming practices and standards used in the code package, a description of the code structure, and overviews of the three modules comprising the package.

A. Programming Practices and Standards

In general, the programming standards and practices recommended by the Committee on Computer Code Coordination (CCCC)[3,4] have been followed throughout the development of ONEDANT. By following these practices and standards, problems associated with exporting and implementing the code in different computing environments and at different computing installations are minimized. This section provides a brief summary of the CCCC programming practices and standards used in ONEDANT.

1. Language. The programming language is standard FORTRAN 77 as defined by the ANSI standard X3.9-1978.[5]

2. Structure. The code is structured in a form that separates the input and the output (or edit) functions from the main calculational (or solver) section of the code. A more complete description of the code structure is provided in Section B of this chapter.

3. Standard Interface Files. ONEDANT makes use of interface files to transmit data between and within its modules. These interface files are binary, sequential data files. Standard interface files are interface files whose structure and data-content formats have been standardized by the CCCC. Code-dependent interface files are files whose structure and data-content formats have not been standardized.

The following CCCC standard interface files are accepted, created, or otherwise used in ONEDANT: ISOTXS, GRUPXS, GEODST, NDXSRF, ZNATDN, SNCONS, FIXSRC, RTFLUX, ATFLUX, RZFLUX, RAFLUX, and AAFLUX. File descriptions for these files are provided in Ref. 3.

The following code-dependent interface files are used in ONEDANT: MACRXS, BXSLIB, FISSRC, RMFLUX, RZMFLX, SNXEDT, ADJMAC, SOLINP, EDITIT,

and ASGMAT. File descriptions for these code-dependent interface files are provided in Appendix A. Other special purpose ASC files produced by or usable by the code package are MACBCD, XSLIBB, XSLIBF, XSLIBE, EDTOUT, and EDTOGX. File descriptions for the EDTOUT and EDTOGX files are provided in Appendix C.

The use of the above interface files is described in Section B of this chapter.

4. Data Management and Transfers. ONEDANT is designed with rather sophisticated data-management techniques in order to accommodate, as efficiently as possible, the transfer of the large amounts of data frequently needed for solving large problems. Data management in the code involves the reading and writing of sequential data files, a flexible capability to block data, and if needed, use of multilevel data-management/transfers using random-access files.

The CCCC standardized subroutines SEEK, REED, and RITE are used for data transfers involving binary, sequential data files. A description of these routines is provided in Ref. 3.

For multilevel data transfer using random (direct)-access files, the CCCC procedures have been implemented in ONEDANT. The standardized subroutines DOPC, CRED/CRIT, DRED/DRIT are used to effect multilevel data transfers using random-access files. A description of these procedures and subroutines is provided in Ref. 3.

5. Central Memory Restrictions. ONEDANT is designed to be operable within a 50,000-word central memory. At the same time, it is easily adaptable to a larger amount of central memory for installations having a larger central memory.

6. Word Size. The code is designed to be easily converted from its basic long-word computer form to a form for use on short-word computers. (On a long-word computer, a six-character Hollerith word is a single-precision word, while on a short-word computer, it is a double-precision word.)

B. ONEDANT Code Package Structure

The ONEDANT code package consists of three major, functionally independent modules: an Input Module, a Solver Module, and an Edit Module. These modules are linked by means of binary interface files. The Input Module processes any and all input specifications and data and, if required, generates the binary files for use by the Solver and/or Edit modules. The Solver Module performs the transport calculation and generates flux files for use by the Edit Module.

The Solver Module also generates other interface files for use by other codes or for subsequent calculations by the Solver Module. The Edit Module performs cross-section and response function edits using the flux files from the Solver Module.

The interface files accepted, used, and generated by the modules are shown in Table I.

A segmented structure is used in ONEDANT for implementing the modules. Such a structure involves the use of a main driver together with input, solver, and edit segments.

The main program, DRIVER, controls the calling of the primary segments, together with those service subroutines used by more than one segment.

The first segment constitutes the Input Module. It is structured into a driver routine, INPT10, plus twelve secondary sections as shown in Table II. Each of the secondary sections performs a unique function so that the Input Module itself is constructed in a modular form.

The second segment constitutes the Solver, or calculational, module. It consists of a driver routine, GRND20, plus seven secondary sections and is depicted in Table III.

The third segment is the Edit Module. It currently consists of a driver routine, OUTT30, plus two secondary sections as shown in Table IV.

A fourth segment is used in ONEDANT. This fourth segment provides highlights of the just-executed run as an aid to the user. These highlights are a printed summary of some of the pertinent facts, options, and decisions encountered during the run along with storage and run time information. This segment is not considered to be a module in the sense of the first three segments.

TABLE I.
Card Image Input and Files Read and Produced by ONEDANT Code Package

O = Optional
A = Always

Kind of Information	File or Cards	Input Module		Solver Module		Edit Module	
		Read	Produce	Read	Produce	Read	Produce
Geometry Information	GEODST	O	A	A	O	A	
	Card Images	O					
Cross Sections	ISOTXS	O					
	GRUPXS	O					
	MENDF[a]	O					
	MACRXS/ADJMAC	O	A	A			
	MACBCD	O	O				
	XSLIB	O					
	XSLIBE	O	O				
	XSLIBF	O	O				
	BXSLIB	O	O				
	XSLIBB	O	O				
	SNXEDT		A			O	
	Card Images	O					
Material Mixing	NDXSRF/ZNATDN	O	A	A		O	
	Card Images	O					
Assignment of Mat'ls to Zones	ASGMAT	O	A	A	O	O	
	Card Images	O					
Solver Module Input	SOLINP	O	A	A			
	Card Images	O					
Quadrature	SNCONS	O		O	A		
Inhomog. Sources	FIXSRC	O		O	O		
	Card Images	O					
Edit Module Input	EDITIT	O	O			A	
	Card Images	O					
Other Output Files	RTFLUX/ATFLUX				A	A	O
	RAFLUX/AAFLUX				O		
	RZFLUX						O
	RMFLUX				O		O
	FISSRC				O		
	RZMFLX						O[b]
	EDTOUT						O
	EDTOGX						O

[a] available only at Los Alamos
[b] requires an RMFLUX file from the Solver

TABLE II.
Structure of the Input Module.

Routine	*Function*
INPT10	Input Module Driver; controls the flow of the code by calling one or more of the secondary submodules below.
INPT11	Controls code setup and storage allocation
INPT12	Controls geometry data processing
INPT13	Controls cross section library processing for XSLIB, MENDF, XSLIBB, and MACBCD library forms
INPT14	Controls mixing specification processing
INPT15	Controls GRUPXS cross section library processing
INPT16	Controls ISOTXS cross section library processing
INPT17	Controls BXSLIB cross section library processing
INPT18	Controls Solver Module input data processing
INPT19	Controls Edit Module input data processing
INP110	Controls cross section balancing operation
INP111	Controls adjoint reversals
INP112	Controls GEODST file post processing

TABLE III.
Structure of the Solver Module.

Routine	*Function*
GRND20	Solver Module Driver; controls the flow of the code by calling one or more of the secondary submodules below.
INPT21	Controls module initializations
INPT22	Controls quadrature selection
INPT23	Controls flux guess and inhomogeneous source processing
GRND24	Controls calculational data preparation
GRND25	Controls the outer iterations
OUTT26	Controls final Solver Module printing
OUTT27	Controls binary file preparation

TABLE IV.
Structure of the Edit Module.

Routine	*Function*
OUTT30	Edit Module Driver; controls the flow of the code by calling one or more of the secondary submodules below.
OUTT31	Controls reaction rate calculations
OUTT32	Controls power normalization, edit zone averaging, and output file preparation

C. Input Module

The Input Module performs the necessary activities for processing all input data required for the execution of the Solver and/or Edit Modules. These activities include the reading of input data and the creation of binary interface files. The latter activity may require a certain degree of data processing. Each of these activities is discussed below.

In performing the reading-of-input-data activity, the Input Module accepts standard interface files (binary), code-dependent binary interface files, or card-images for its input. These are listed in Table I. As is indicated in the table, input data to the code can be provided in several different forms and many combinations of forms to provide a great deal of flexibility to the user. Chapters V through XI provide specific information and further details on the specification of input data.

The second major activity in the Input Module is the creation of binary interface files containing all input data. These files are subsequently used as the sole means of transmitting data to either the Solver or Edit Modules. The files emerging from the Input Module are given in Table I and take the form of either CCCC standard interface files or code-dependent interface files. In this file-creation activity, the Input Module is called on to perform several types of tasks. As an example, the only form in which geometry-related information emerges from the Input Module is in the form of a GEODST standard interface binary file. If a user supplies geometry-related input by means of card-image input, the Input Module reads this input, translates the data into a GEODST-compatible form, and creates the resulting GEODST file. On the other hand, if the geometry-related information is supplied by the user through an already existing GEODST file, the Input Module is required to do nothing.

A second, more complex, example of the function of the Input Module involves the mixing of isotopes, or nuclides, to create materials which are subsequently assigned to physical regions in the problem (called zones) to define the macroscopic cross-section data for the zones. For this example, it will be assumed that the user selects card-image input as the form for the Input Module. First, the isotope mixing specifications appropriate for the desired materials are input via card-image. The Input Module reads this data, translates the data and creates the two standard interface files NDXSRF and ZNATDN as shown in Table I. These two files appear as output from the Input Module. Assuming next that the isotope cross sections are provided by the user as a card-image library, the Input Module reads this library (in isotope-ordered form) and also reads the just-created NDXSRF and ZNATDN files. The mixing specifications provided by the latter files are applied to the isotopic cross-section data to generate material cross sections which are written, in group order, to a code-dependent binary file named MACRXS. (A group-ordered file named SNXEDT for use by the Edit Module is also created at this time but will not be considered in this example.) The MACRXS file becomes the sole source of cross-section data to the Solver Module if the Solver calculation is to be a forward, or regular, calculation. If an adjoint calculation is to be performed by the Solver, the Input Module re-reads the MACRXS file, performs the adjoint reversals on

the cross sections, and creates the code-dependent binary file named ADJMAC containing the adjoint-reversed material cross sections for use by Solver.

D. Solver Module

The Solver Module of ONEDANT has the function of effecting numerical solutions of the one-dimensional, multigroup form of the neutral-particle steady-state Boltzmann transport equation. The discrete-ordinates approximation is used for treating the angular variation of the particle distribution and the diamond-difference scheme is used for phase space discretization.[6]

In solving the transport equation numerically, an iterative procedure is used. This procedure involves two levels of iteration referred to as inner and outer iterations. The acceleration of these iterations is of crucial importance to transport codes in order to reduce the computation time involved. The ONEDANT Solver Module employs the diffusion synthetic acceleration method developed by Alcouffe,[7] an extremely effective method for accelerating the convergence of the iterations. A relatively detailed development of the solution method used in the ONEDANT Solver Module is provided in Chapter III.

The Solver Module is essentially a free-standing entity, and input to and output from the module is in the form of binary files together with limited printed output. The binary interface files used as input to the Solver Module are listed in Table I. The files required for execution of the module are a GEODST standard interface file together with the code-dependent interface files MACRXS or ADJMAC, ASGMAT, and SOLINP. Optional files, which may be input to the Solver Module, are the standard interface files SNCONS, RTFLUX or ATFLUX, and FIXSRC.

The output from the Solver Module always consists of the scalar flux standard interface file RTFLUX (or ATFLUX if an adjoint problem were run), the standard interface file SNCONS, and user-selected printed output. If desired by the user, the angular flux standard interface file RAFLUX (or AAFLUX, if an adjoint problem were run) will be produced. If an inhomogeneous source problem were run, a FIXSRC standard interface file would be produced. If desired by the user, the angular flux moments code-dependent interface file RMFLUX would be produced.

E. Edit Module

The function of the Edit Module is to produce the printed edit-output selected by the user. Edit-output refers to information which is obtained from data contained on one or more interface files but which generally requires manipulating or processing of the data. An example of the edit-output is a microscopic reaction-rate distribution, $\sigma\phi$, where σ is a particular multigroup, microscopic cross section for a particular isotope or nuclide and ϕ is the multigroup scalar flux distribution obtained from the Solver Module. In this example, data from both a cross-section interface file and a scalar flux file are required to be recovered and multiplied, and the product printed.

The Edit Module is an essentially free-standing module accepting only interface files as input and producing printed output. The required input files for execution of the Edit Module are the code-dependent binary interface file EDITIT and the standard interface files RTFLUX (or ATFLUX) and GEODST as shown in Table I. Optional input files are the standard interface files NDXSRF and ZNATDN and the code-dependent files SNXEDT and ASGMAT. The code-dependent files are produced by the Input Module.

III. GENERAL DEVELOPMENT OF THE 1-D S_N EQUATIONS

This chapter provides the development of the one-dimensional, multigroup, discrete-ordinates, diamond-differenced form of the time-independent Boltzmann transport equation. This development is followed by a brief description of the iterative procedure used to solve the transport equation using the diffusion synthetic acceleration (DSA) scheme to accelerate the iterations.

A. Development of the Multigroup, Discrete-Ordinates Form of the Transport Equation

The time-independent inhomogeneous Boltzmann transport equation in one space dimension is

$$\begin{aligned} &\nabla \cdot \underline{\Omega}\psi(r,E,\underline{\Omega}) + \sigma(r,E)\,\psi(r,E,\underline{\Omega}) \\ &= \int\int dE'd\Omega'\sigma_s(r,E' \rightarrow E, \underline{\Omega}\cdot\underline{\Omega}')\,\psi(r,E',\underline{\Omega}') \\ &\quad + \frac{1}{4\pi}\int\int dE'd\Omega'\chi(r,E' \rightarrow E)\,\nu\sigma_f(r,E')\,\psi(r,E',\underline{\Omega}') \\ &\quad + Q(r,E,\underline{\Omega}) \quad , \end{aligned} \tag{1}$$

where $\psi(r,E,\underline{\Omega})$ is the particle fiux (particle number density times the particle speed) defined such that $\psi(r,E,\underline{\Omega})\ dE\ d\underline{r}\ d\Omega$ is the fiux of particles in the energy range dE about E, in the volume element d$\underline{r}$ about r, with directions of motion in the solid angle element $d\Omega$ about $\underline{\Omega}$. Similarly, $Q(r,E,\underline{\Omega})\ dE\ d\underline{r}\ d\Omega$ is the rate at which particles are produced in the same element of phase space from sources that are independent of the fiux ψ. The macroscopic total cross section is σ, the macroscopic scattering transfer probability, from energy E' to energy E through a scattering angle $\underline{\Omega}\cdot\underline{\Omega}'$, is σ_s, and the macroscopic fission cross section is σ_f. All of the quantities may be spatially dependent. The number of particles emitted isotropically $\left(\frac{1}{4\pi}\right)$ per fission is ν, and the fraction of these particles appearing in energy dE about E from fissions in dE′ about E′ is $\chi(r,E' \rightarrow E)$.

The homogeneous transport equation is the same as Eq. (11) except that Q is zero and the term representing the fission source is divided by k_{eff}. The inhomogeneous problem is referred to as a source problem and the homogeneous problem will be referred to as an eigenvalue problem. ONEDANT will solve both types of problems.

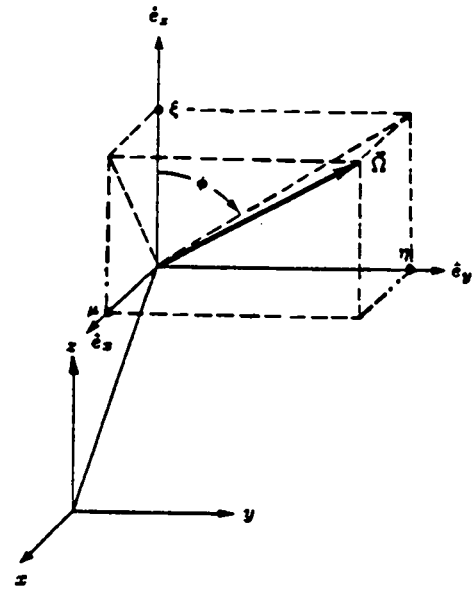


Fig. 1. Coordinates in plane geometry.

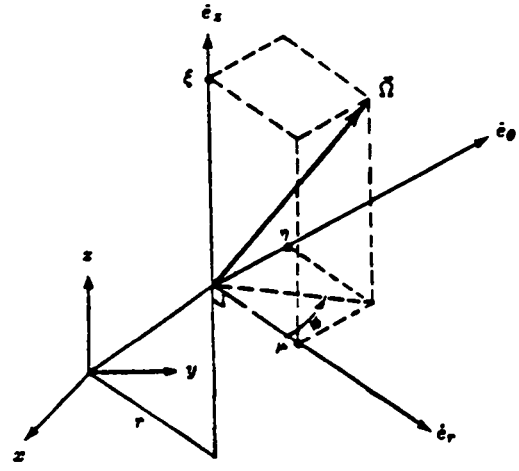


Fig. 2. Coordinates in cylindrical geometry.

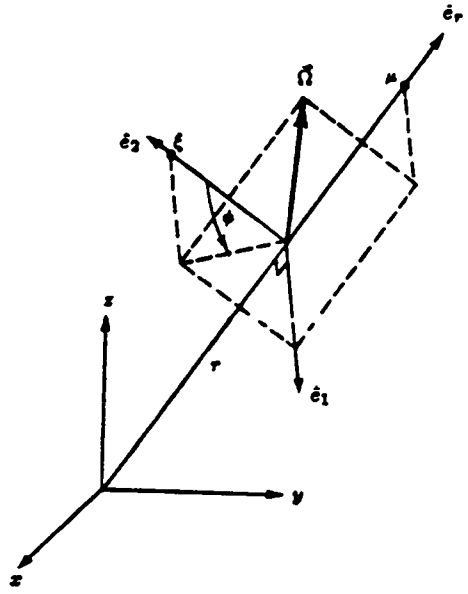


Fig.3. Coordinates in spherical geometry.

TABLE V
FORMS OF $\nabla \cdot \underline{\Omega}\psi$

Geometry	Dependence of	Definition of Variables	$\nabla \cdot \underline{\Omega}\psi$
Plane	$\psi(x,\mu)$	$\mu = \hat{e}_x \cdot \underline{\Omega}$	$\mu\frac{\partial\psi}{\partial x}$
	or	$\xi = (1-\mu^2)^{1/2}\ cos\phi$	
	$\psi(x,\mu,\phi)$	$\eta = (1-\mu^2)^{1/2}\ sin\phi$	
Cylindrical	$\psi(r,\mu,\eta)$	$\mu = \hat{e}_r \cdot \underline{\Omega}$	
		$\xi = \hat{e}_z \cdot \underline{\Omega}$	
			$\frac{\mu}{r}\frac{\partial(r\psi)}{\partial r} - \frac{1}{r}\frac{\partial(\eta\psi)}{\partial\phi}$
		$\eta = (1-\xi^2)^{1/2}\ sin\,\phi$	
		$\mu = (1-\xi^2)^{1/2}\ cos\,\phi$	
Spherical	$\psi(r,\mu)$	$\mu = \hat{e}_r \cdot \underline{\Omega}$	$\frac{\mu}{r^2}\frac{\partial(r^2\psi)}{\partial r} + \frac{1}{r}\frac{\partial[(1-\mu^2)\psi]}{\partial\mu}$

1. Particular Forms of the Divergence Operator. The form of the divergence operator $\nabla \cdot \underline{\Omega}\psi$ or $(\underline{\Omega} \cdot \nabla\psi)$ for the geometries treated by ONEDANT is given in Table V in terms of the coordinate systems shown in Figs. 1-3.

In the standard plane geometry, the angular fiux $\psi(r, E, \underline{\Omega})$ is assumed independent of the azimuthal angle ϕ so that the angular dependence is reduced to the μ interval (-1, +1). ONEDANT also permits the two-angle plane geometry option in which no assumptions of symmetry in angle are imposed. In this case the complete unit sphere of angular directions must be considered.

In cylindrical geometry, the angular fiux is assumed symmetric in the ξ angular cosine and also symmetric about the $\mu - \xi$, (or ϕ=0°- 180°) plane. Thus, only one-fourth of the unit sphere need be considered in the angular dependence.

In spherical geometry, the angular fiux is assumed symmetric in the azimuthal angle ϕ so that the angular dependence is reduced to the μ interval (-1, +1).

2. Spherical Harmonics Expansion of the Scattering Source. The scattering transfer probability in Eq. (1) is represented by a finite Legendre polynominal expansion of order ISCT

$$\sigma_s\left(r, E' \rightarrow E, \underline{\Omega}' \cdot \underline{\Omega}\right) = \sum_{L=0}^{ISCT} \left(\frac{2L+1}{4\pi}\right) \sigma_s^L\left(r, E' \rightarrow E\right) P_L\left(\underline{\Omega}' \cdot \underline{\Omega}\right) \quad . \qquad (2)$$

If this expansion is inserted into Eq. (1) and the addition theorem for spherical harmonics used to expand $P_n\left(\underline{\Omega}' \cdot \underline{\Omega}\right)$, the scattering source becomes

$$\int\int dE' d\Omega' \; \sigma_s\left(r, E' \rightarrow E, \underline{\Omega} \cdot \underline{\Omega}'\right) \psi\left(r, E', \underline{\Omega}'\right) \equiv SS$$

$$= \int_{E'} dE' \sum_{L=0}^{ISCT} \left(\frac{2L+1}{4\pi}\right) \sigma_S^L\left(r, E' \rightarrow E\right)$$

$$\left\{ P_L(\mu) \int_{-1}^{1} d\mu' \int_0^{2\pi} d\phi' P_L\left(\mu'\right) \psi\left(r, E', \mu', \phi'\right) \right. \qquad (3)$$

$$+ 2 \sum_{K=1}^{L} \frac{(L-K)!}{(L+K)!} P_L^K(\mu) \int_{-1}^{1} \int_0^{2\pi} d\phi' P_L^K\left(\mu'\right)$$

$$\left. cos\ K\left(\phi - \phi'\right) \psi\left(r, E', \mu', \phi'\right) \right\} \quad ,$$

where for cylindrical geometry we must replace the μ variable with ξ. Using the relation $cos\ L(\phi - \phi') = cos\ L\phi\, cos\ L\phi' + sin\ L\phi\ sin\, L\phi'$, we can write Eq. (3) as

$$SS = \int_{E'} dE' \sum_{L=0}^{ISCT} (2L+1)\sigma_S^L (r, E' \rightarrow E) \Bigg\{ P_L(\mu)\phi_L (r, E')$$

$$+ \sum_{K=1}^{L} \sqrt{\frac{2(L-K)!}{(L+K)!}} \Big[\phi_{C,L}^K (r, E')\, P_L^K(\mu)\, cos\ K\phi \qquad (4)$$

$$+ \phi_{S,L}^K (r, E')\, P_L^K(\mu)\, sin\ K\phi \Big] \Bigg\} \quad ,$$

where we have defined the moments of the angular fiux as

$$\phi_L (r, E') \equiv \frac{1}{4\pi} \int_{-1}^{1} d\mu' \int_0^{2\pi} d\phi'\ P_L (\mu')\, \psi (r, E', \mu', \phi') \quad , \qquad (5a)$$

$$\phi_{C,L}^K (r, E') \equiv \frac{1}{4\pi} \int_{-1}^{1} d\mu' \int_0^{2\pi} d\phi'\ \psi (r, E', \mu', \phi')\, P_L^K (\mu')\, cos\ K\phi' \quad . \qquad (5b)$$

$$\phi_{S,L}^K (r, E') \equiv \frac{1}{4\pi} \int_{-1}^{1} d\mu' \int_0^{2\pi} d\phi'\ \psi (r, E', \mu', \phi')\, P_L^K (\mu')\, sin\ K\phi' \quad . \qquad (5c)$$

In both standard plane and spherical geometries, due to symmetry in the azimuthal angle ϕ, the fiux moments $\phi_{C,L}^K$ and $\phi_{S,L}^K$ are identically zero. In cylindrical geometry (with ξ, ξ' replacing μ, μ' in Eqs. (4) and (5)), the odd moments (K+L=odd) of $\phi_{C,L}^K$ vanish as do all the sine moments $\phi_{S,L}^K$. In the two-angle plane geometry all moments must be retained.

In all cases the scattering source, SS, can be written in the general form

$$SS = \int_{E'} dE' \sum_{n=1}^{NM} (2L+1)\sigma_S^L (r, E' \rightarrow E)\, R_n(\underline{\Omega})\tilde{\phi}_n (r, E') \quad , \qquad (6)$$

where NM is the total number of spherical harmonics (and fiux moments) required for a given Legendre expansion order, ISCT (as shown in Table VI), the $R_n(\underline{\Omega})$ are the spherical harmonics appropriate to the particular geometry, and the $\tilde{\phi}_n(r, E)$ are the angular fiux moments corresponding to the $R_n(\underline{\Omega})$. The index L in Eq. (6) is the subscript of the Legendre function P_L^K appearing in the appropriate $R_n(\underline{\Omega})$, $0 \leq L \leq ISCT$. The $R_n(\underline{\Omega})$ are listed in Table VII for typical Legendre expansion

orders. For each $R_n(\underline{\Omega})$ in the table is a corresponding flux moment defined by Eq. (5a), (5b), or (5c) as appropriate.

3. Spherical Harmonics Expansion of the Inhomogeneous Source. In a manner similar to that used for the scattering source, the inhomogeneous source $Q(r, E, \underline{\Omega})$ can be represented as a finite expansion using the spherical harmonics $R_n(\underline{\Omega})$ defined in Table VII. First, the inhomogeneous source moments are defined for a Legendre expansion order IQAN:

$$Q_L(r,E) \equiv \frac{1}{4\pi} \int_{-1}^{1} d\mu \int_0^{2\pi} d\phi Q(r,E,\underline{\Omega}) P_L(\mu) \quad , \quad L = 0,\ldots,IQAN \quad , \tag{7a}$$

$$Q^K_{C,L}(r,E) \equiv \frac{1}{4\pi} \int_{-1}^{1} d\mu \int_0^{2\pi} d\phi Q(r,E,\underline{\Omega}) P^K_L(\mu) \cos K\phi \quad , \tag{7b}$$

$$L = 0,\ldots,IQAN \qquad K = 1,\ldots,L \quad ,$$

$$Q^K_{S,L}(r,E) \equiv \frac{1}{4\pi} \int_{-1}^{1} d\mu \int_0^{2\pi} d\phi Q(r,E,\underline{\Omega}) P^K_L(\mu) \sin K\phi \quad . \tag{7c}$$

The inhomogeneous source is represented in the general spherical harmonic expansion

$$Q(r,E,\underline{\Omega}) = \sum_{n=1}^{NMQ} (2L+1) R_n(\underline{\Omega}) \tilde{Q}_n(r,E) \quad , \tag{8}$$

where NMQ is the total number of spherical harmonics (and source moments) required for a given Legendre expansion order, IQAN, as shown in Table VI, the $R_n(\underline{\Omega})$ are the spherical harmonics appropriate to the geometry being used, and the $\tilde{Q}_n(r,E)$ are the angular source moments corresponding the the $R_n(\underline{\Omega})$. The index L is the subscript of the Legendre function P^K_L appearing in the appropriate $R_n(\underline{\Omega})$, $0 \le L \le IQAN$. The $R_n(\underline{\Omega})$ are listed in Table VII for typical Legendre expansion orders. For each of these $R_n(\underline{\Omega})$ is a corresponding source moment defined by Eqs. (7a), (7b), or (7c), as appropriate.

4. Discretization of the Energy Variable–the Multigroup Approximation. The energy domain of interest is assumed to be partitioned into NGROUP intervals of width $\Delta E_g, g = 1,2,\ldots,NGROUP$. By convention, increasing g represents decreasing energy. If Eq. (1) is integrated over ΔE_g using the spherical harmonic expansion of Eqs. (6) and (8), we get

TABLE VI
NUMBER OF SPHERICAL HARMONICS, N, AS A FUNCTION OF LEGENDRE EXPANSION ORDER, L_o

	N (see below)		
L_o	Standard Plane and Spherical Geometries	Cylindrical Geometry	Two-Angle Plane Geometry
0	1	1	1
1	2	2	4
2	3	4	9
3	4	6	10
4	5	9	25
5	6	12	36

$$N = \begin{cases} L_o + 1 & \text{for standard plane and spherical geometry} \\ (L_o + 2)^2 / 4 & \text{for cylindrical geometry} \\ (L_o + 1)^2 & \text{for two-angle plane geometry} \end{cases}$$

TABLE VII
SPHERICAL HARMONICS, R_n ($\underline{\Omega}$), FOR DIFFERENT GEOMETRIES FUNCTION OF LEGENDRE EXPANSION ORDER, L_o

N	*Standard Plane and Spherical Geometries* P_5^a	*Cylindrical Geometry* P_4^a	*Two – Angle Plane* P_3^a
1	$P_o(\mu)$	$P_o(\xi)$	$P_o(\mu)$
2	$P_1(\mu)$	$P_1^1(\xi)\ cos\phi$	$P_1(\mu)$
3	$P_2(\mu)$	$P_2(\xi)$	$P_1^1(\mu)\ cos\phi$
4	$P_3(\mu)$	$\frac{\sqrt{3}}{6}\ P_2^2(\xi)\ cos2\phi$	$P_1^1(\mu)\ sin\phi$
5	$P_4(\mu)$	$\frac{\sqrt{6}}{6}\ P_3^1(\xi)\ cos\phi$	$P_2(\mu)$
6	$P_5(\mu)$	$\frac{\sqrt{10}}{60}\ P_3^3(\xi)\ cos3\phi$	$\frac{\sqrt{3}}{3}\ P_2^1(\mu)\ cos\phi$
7		$P_4(\xi)$	$\frac{\sqrt{3}}{3}\ P_2^1(\mu)\ sin\phi$
8		$\frac{\sqrt{5}}{30}\ P_4^2(\xi)\ cos2\phi$	$\frac{\sqrt{3}}{6}\ P_2^2(\mu)\ cos2\phi$
9		$\frac{\sqrt{35}}{840}\ P_4^4(\xi)\ cos4\phi$	$\frac{\sqrt{3}}{6}\ P_2^2(\mu)\ sin2\phi$
10			$P_3(\mu)$
11			$\frac{\sqrt{6}}{6}P_3^1(\mu)cos\phi$
12			$\frac{\sqrt{6}}{6}P_3^1(\mu)sin\phi$
13			$\frac{\sqrt{15}}{30}P_3^2(\mu)cos2\phi$
14			$\frac{\sqrt{15}}{30}P_3^2(\mu)sin2\phi$
15			$\frac{\sqrt{10}}{60}P_3^3(\mu)cos3\phi$
16			$\frac{\sqrt{10}}{60}P_3^3(\mu)sin3\phi$

[a] P_L denotes L^{th} order Legendre expansion.

$$\nabla \cdot \underline{\Omega}\psi_g(r,\underline{\Omega}) + \sigma_g(r)\psi_g(r,\underline{\Omega}) = \sum_{h=1}^{NGROUP} \sum_{n=1}^{NM} (2L+1)\sigma^L_{s,h\to g} R_n(\underline{\Omega})\tilde{\phi}_{n,h}(r)$$

$$+ \sum_{h=1}^{NGROUP} (\nu\sigma_f)\,\chi_{h\to g}\tilde{\phi}_{l,h}(r) \sum_{n=1}^{NMQ} (2L+1) R_n(\underline{\Omega})\tilde{Q}_{n,g}(r) \quad , \tag{9}$$

for $g = 1, 2, \ldots, NGROUP$. Here the "group fiux"

$$\psi_g(r,\underline{\Omega}) = \int_{\Delta E_g} \psi(r,E,\underline{\Omega})dE \quad , \tag{10}$$

is no longer a distribution in energy nor an average (in energy), but is the total fiux of particles in the energy interval. Because of this, energy integrals in ONEDANT are evaluated by simple sums.

To simplify the notation for the following discussion, the right-hand side of Eq. (9) (the sources due to scattering, fission, and inhomogeneous source) will be denoted by $S_g(r,\underline{\Omega})$. It is recognized that portions of S_g depend on the unknown fiux ψ_g through the fiux moments, but this dependence is treated by iterative procedures and, accordingly, the simplified representation results in no loss of generality. Equation (9) is thus written, with the group index omitted.

$$\nabla \cdot \underline{\Omega}\psi(r,\underline{\Omega}) + \sigma(r)\psi(r,\underline{\Omega}) = S(r,\underline{\Omega}) \quad . \tag{11}$$

5. Discrete-Ordinates Equations. In the discrete-ordinates approximation, the angular-direction domain, characterized by $\underline{\Omega}$, is discretized into a set of quadrature points each with an associated quadrature weight. Although not rigorously correct, the discrete-ordinates approximation is commonly referred to as the S_N method, and the number of quadrature directions, MM, is a function of both the S_N order and the geometry, as shown in Table VIII. In discretizing the angular domain $\underline{\Omega}$, each of the quadrature points, or directions, is characterized by the subscript m and corresponds to direction $\underline{\Omega}_m$. The quadrature weight, w_m, corresponds to the differential area on the unit sphere normalized to unity, i.e., w_m is analogous to $d\Omega_m/4\pi$. In such a manner the weights, w_m, are normalized so that

$$\sum_{m=1}^{MM} w_m = 1 \quad . \tag{12}$$

The angular flux for direction m at space point r is denoted $\psi_m(r)$ and represents the average angular flux in directions $d\Omega_m$ about $\underline{\Omega}_m$ at space point r. The scalar flux at space point $r, \phi_o(r) = \tilde{\phi}_1(r)$, is simply

$$\phi_o(r) = \sum_{m=1}^{MM} w_m \psi_m(r) \quad . \tag{13}$$

a. Standard Plane Geometry. For standard plane geometry (see Table V and Fig. 1) azimuthal symmetry is assumed in ϕ so that $\underline{\Omega}(\mu, \phi)$ becomes $\underline{\Omega}(\mu)$ and $d\underline{\Omega}$ becomes $2\pi d\mu$. The angular interval $\mu\epsilon$ [-1,1] is discretized into MM quadrature points μ_m and associated weights w_m ordered as shown in Fig. 4. Note that the weights, w_m, correspond to $d\mu_m/2$ for this geometry. The angular flux moments, given by Eq. (5a), are approximated by

$$\phi_L(x) \tilde{=} \sum_{m=1}^{MM} w_m P_L(\mu_m) \psi_m(x) \quad . \tag{14}$$

TABLE VIII
NUMBER OF QUADRATURE POINTS, MM, AS A FUNCTION OF S_N ORDER, N

N	MM			
	Standard Plane Geometry	Two-Angle Plane Geometry	Cylindrical Geometry	Spherical Geometry
2	2	8	2	2
4	4	24	6	4
6	6	48	12	6
8	8	80	20	8
12	12	168	42	12
16	16	288	72	16
N	N	$N \cdot (N+2)$	$\frac{N \cdot (N+2)}{4}$	N

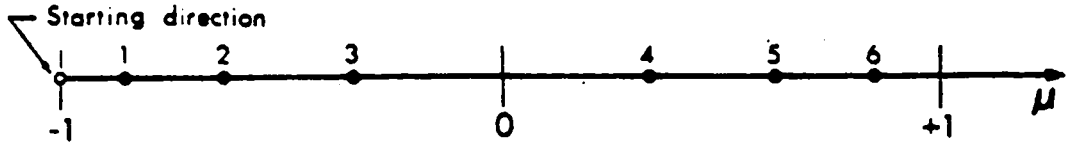


Fig. 4. Ordering of S_6 directions in plane and spherical geometries. The starting direction only applies to spherical geometry.

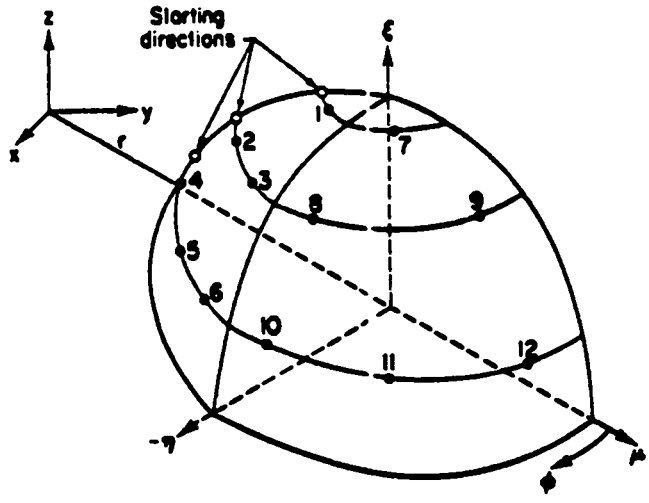


Fig. 5. Ordering of S_6 directions in cylindrical geometry.

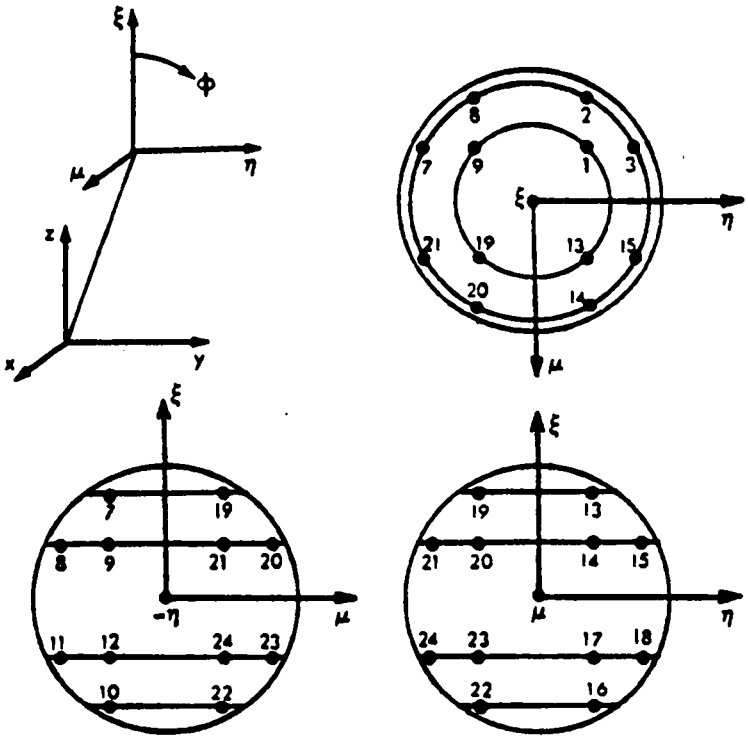


Fig. 6. Ordering of S_4 directions in two-angle plane geometry. The ordinates in the octant $\mu, \xi < 0, \eta > 0$ are not shown.

The discrete-ordinates approximation to the transport Eq. (11) becomes

$$\mu_m \frac{\partial \psi_m(x)}{\partial x} + \sigma(x)\psi_m(x) = S_m(x) \quad . \tag{15}$$

b. Two-Angle Plane Geometry. For two-angle plane geometry the entire unit sphere of directions is discretized into MM quadrature points (μ_m, ϕ_m) and associated weights ordered as shown in Fig. 6. The weights, w_m, correspond to $d\Omega_m/4\pi$ for this option. The angular flux moments, given by Eqs. (5a)-(5c), are approximated by

$$\phi_L(x) \tilde{=} \sum_{m=1}^{MM} w_m P_L(\mu_m)\psi_m(x) \quad , \tag{16a}$$

$$\phi_{C,L}^K(x) \tilde{=} \sum_{m=1}^{MM} w_m \psi_m(x) P_L^K(\mu_m) \cos K\phi_m \quad , \tag{16b}$$

$$\phi_{S,L}^{K}(x) = \sum_{m=1}^{MM} w_m \psi_m(x) P_L^K(\mu_m) \sin K\phi_m \quad , \tag{16c}$$

The discrete-ordinates approximation to the transport equation is the same as for standard plane geometry, i.e.., Eq. (15).

c. Cylindrical Geometry. For cylindrical geometry (see Table V and Fig. 2), the multigroup transport Eq. (11) may be written

$$\mu \frac{\partial(r\psi)}{\partial r} - \frac{\partial(\eta\psi)}{\partial \phi} + r\sigma\psi = rS(r,\underline{\Omega}) \quad , \tag{17}$$

where $\psi = \psi(r,\underline{\Omega})$.

For the discrete-ordinates approximation in cylindrical geometry, only one quadrant of the unit sphere is discretized into a set of MM quadrature points (μ_m, η_m) and associated quadrature weights w_m. The ordering of these quadrature points is illustrated in Fig. 5 for an S_6 quadrature. As before, $\psi_m(r) \doteq \psi(r, \mu_m, \eta_m)$ represents the average angular flux in $d\Omega_m$ about $\underline{\Omega}_m$ and the angular fiux moments for direction m are given by Eqs. (16a)-(16b). In addition, it is necessary to define angular-cell-edge fiuxes on a given ξ-level as $\psi_{m-1/2}(r)$ and $\psi_{m+1/2}(r)$. The discrete-ordinates approximation to Eq. (17) can then be written:

$$\begin{aligned} \mu_m \frac{\partial(r\psi_m)}{\partial r} &+ \left(\frac{\alpha_{m+1/2}}{w_m}\right)\psi_{m+1/2}(r) \\ &- \left(\frac{\alpha_{m-1/2}}{w_m}\right)\psi_{m-1/2}(r) + r\sigma\psi_m(r) = rS_m(r) \quad , \end{aligned} \tag{18}$$

where the $\alpha_{m-1/2}$ and $\alpha_{m+1/2}$ are angular coupling coefficients. These coefficients satisfy the recursion relation

$$\alpha_{m+1/2} - \alpha_{m-1/2} = -w_m\mu_m \quad , \tag{19}$$

with the requirement that the first $(\alpha_{1/2})$ and last $(\alpha_{M+1/2})$ coefficients on each ξ-level must vanish. It can be shown that Eq. (18) becomes identical to Eq. (17) in the limit of vanishingly small angular intervals. In the output of ONEDANT pertaining to the angular quadrature, the quantities $\left(\frac{\alpha_{m+1/2}}{w_m}\right)$ and $\left(\frac{\alpha_{m-1/2}}{w_m}\right)$ are printed out under the headings BETA PLUS and BETA MINUS, respectively.

d. Spherical Geometry. From Table V the multigroup transport Eq. (1) can be written

$$\mu\frac{\partial\left(r^2\psi\right)}{\partial r}+r\frac{\partial\left[\left(1-\mu^2\right)\psi\right]}{\partial\mu}+r^2\sigma\psi=r^2S(r,\mu) \quad , \tag{20}$$

where azimuthal symmetry in ϕ (see Fig. 3) has been assumed. The angular domain $\mu\epsilon[-1,1]$ is discretized into MM quadrature points μ_m and associated weights w_m. Note that in spherical geometry, like standard plane geometry, the w_m correspond to $d\mu_m/2$. The ordering of the quadrature points is illustrated in Fig. 4. As before, $\psi_m(r)\tilde{=}\psi\left(r,\mu_m\right)$ represents the average angular fiux in $d\Omega_m\left(=d\mu_m\right)$ about $\underline{\Omega}_m$ and the angular fiux moments, given by Eq. (5a), are approximated by Eq. (4). In addition, it is necessary to define angular-cell-edge fiuxes $\psi_{m-1/2}(r)$ and $\psi_{m+1/2}(r)$. The discrete-ordinates approximation to Eq. (20) is then written as

$$\begin{aligned}\mu_m\frac{\partial\left(r^2\psi_m\right)}{\partial r}+\left[\left(\frac{\beta_{m+1/2}}{w_m}\right)\psi_{m+1/2}(r)-\left(\frac{\beta_{m-1/2}}{w_m}\right)\psi_{m-1/2}(r)\right]r \\ +r^2\sigma\psi_m(r)=r^2S_m(r) \quad ,\end{aligned} \tag{21}$$

where the angular coupling coefficients β must satisfy the recursion relation

$$\beta_{m+1/2}-\beta_{m-1/2}=-2w_m\mu_m \quad , \quad m=1,\ldots,MM \quad , \tag{22}$$

with the requirement from particle conservation that the first $\left(\beta_{1/2}\right)$ and last $\left(\beta_{MM+1/2}\right)$ coefficients must vanish. It can be shown[8] that Eq. (21) becomes identical to Eq. (20) in the limit of vanishingly small angular intervals. In the output of ONEDANT pertaining to the angular quadrature, the quantities $\left(\frac{\beta_{m+1/2}}{2w_m}\right)$ and $\left(\frac{\beta_{m-1/2}}{2w_n}\right)$ are printed out under the headings BETA PLUS and BETA MINUS, respectively.

e. **Starting Directions.** For the curved geometries discrete-ordinates Eqs. (18) and (21), there are three variables to be determined at each space position, r: the angular-cell-edge fiuxes $\psi_{m-1/2}(r)$ and $\psi_{m+1/2}(r)$ and the average angular flux $\psi_m(r)$. The $\psi_{m-1/2}(r)$ fiux can be assumed known (except for $\psi_{1/2}(r)$) from the previous angular mesh-cell computation and assuming continuity at the angular mesh-cell boundaries. The standard diamond-difference[9] assumption in angle is made to relate the $\psi_{m+1/2}$ to ψ_m, namely,

$$\psi_m(r)=\frac{1}{2}\left[\psi_{m-1/2}(r)+\psi_{m+1/2}(r)\right] \quad . \tag{23}$$

Using Eq. (23) to solve for $\psi_{m+1/2}$ and substituting the resulting expression into Eq. (18) or (21), there remains but one equation for the one unknown $\psi_m(r)$.

The assumption that $\psi_{m-1/2}$ is known is correct except for m = 1 for which an initial, or starting, condition is required. To achieve this, ONEDANT uses special, zero-weighted starting directions in spherical and cylindrical geometries to calculate $\psi_{1/2}(r)$. For spherical geometry this starting direction is the straight-inward direction $\mu = -1$ for which the term $(1-\mu^2)\,\psi$ in Eq. (20) vanishes. This yields a special form of Eq. (21) which can be solved for $\psi_{1/2}(r)$. For cylindrical geometry, as shown in Fig. 5, starting directions corresponding to ordinates directed towards the cylindrical axis, $\eta = 0, \psi = 180°$, are used for each ξ-level to yield special equations for $\psi_{1/2}(r)$ on each ξ-level.

6. Discretization of the Spatial Variable. The spatial domain of the problem is ultimately partitioned into IT fine-mesh intervals of width $\Delta x_i, i = 1, 2, \ldots,$ IT such that $\Delta x_i \equiv x_{i+1/2} - x_{i-1/2}$. Subscripts with half-integer values denote interval boundaries, and integer subscripts denote interval average, or midpoint, values. It is assumed that $x_{i+1/2} > x_i > x_{i-1/2}$. With such a partitioning, space derivatives are approximated by finite differences and, typically, the resulting equations are cast in forms using interval, or mesh, average fiuxes, sources, etc.

Thus, for group g, direction m, and mesh interval i,

$$\mu \frac{\partial \psi}{\partial x} \cong \mu_m \left(\frac{\psi_{m,i+1/2,g} - \psi_{m,i-1/2,g}}{\Delta x_i} \right) \quad , \tag{24}$$

and the mesh interval average (midpoint) angular fiux is related to the mesh edge angular fiuxes by the diamond-difference expression

$$\psi_{m,i,g} = \frac{1}{2} \left[\psi_{m,i+1/2,g} + \psi_{m,i-1/2,g} \right] \quad . \tag{25}$$

Note that the average scalar fiux for group g in mesh interval i is

$$\phi_{i,g} \equiv \sum_{m=1}^{MM} w_m \psi_{m,i,g} \quad . \tag{26}$$

B. Iteration Procedure

In solving the transport equation numerically, an iterative procedure is used. This procedure involves two levels of iteration referred to as inner and outer iterations. The acceleration of these iterations is of crucial importance to transport codes in order to reduce the computation time involved. The ONEDANT Solver Module employs the diffusion synthetic acceleration method developed by Alcouffe,[7] an extremely effective method for accelerating the convergence of the iterations.

To display the iterative procedure and the application of the diffusion synthetic acceleration method, consider first the inner iteration equation for energy group g and inner iteration ℓ. Isotropic scatter is assumed only for simplicity. The basic inner iteration equation is written

$$\underline{\Omega} \cdot \nabla \tilde{\psi}_g^{\ell}(r,\underline{\Omega}) + \sigma_g(r)\tilde{\psi}_g^{\ell}(r,\underline{\Omega}) = \sigma_{s,g\to g}(r)\phi_g^{\ell-1}(r) + QQ_g(r) \quad . \tag{27}$$

In Eq. (27), $\tilde{\psi}_g^{\ell}(r,\underline{\Omega})$ is the angular fiux for group g at the ℓ^{th} inner iteration using a scalar flux $\phi_g^{\ell-1}(r)$ assumed known at each inner iteration. QQ_g is the group source which remains unchanged for the group throughout the performance of inner iterations. This group source contains scattering and fission contributions to the group together with any inhomogeneous source. The source is computed using the multigroup scalar fiuxes and moments from the previous outer iteration. In the diffusion synthetic method, a corrected diffusion equation is used to determine the scalar flux ϕ_g needed for the next iteration. In actual fact, there are three separate schemes for writing the corrected diffusion equation to be used: the source correction scheme, the diffusion coefficient correction scheme, and the removal correction scheme. For the source correction scheme we write the corrected diffusion equation as

$$-\nabla \cdot D_g(r)\nabla\phi_g^{\ell}(r) + \sigma_{R,g}(r)\phi_g^{\ell}(r) = QQ_g(r) - R_g^{\ell}(r) \quad , \tag{28}$$

where

$$D_g(r) = \frac{1}{3\sigma_{tr,g}}(r) \quad , \quad \sigma_{R,g}(r) = \sigma_g(r) - \sigma_{s,g\to g}(r)$$

and the correction term is

$$R_g^{\ell}(r) = \nabla \cdot \tilde{J}_g^{\ell}(r) + \nabla \cdot D_g(r)\nabla\tilde{\phi}_g^{\ell}(r) \quad . \tag{29}$$

In Eq. (29),

$$\tilde{\phi}_g^{\ell}(r) = \int d\Omega \tilde{\psi}_g^{\ell}(r,\underline{\Omega}) \quad , \quad \tilde{J}_g^{\ell}(r) = \int d\Omega \underline{\Omega} \tilde{\psi}_g^{\ell}(r,\underline{\Omega}) \quad . \tag{30}$$

Note that a tilde is used to indicate quantities calculated using the transport angular flux, $\tilde{\psi}_g^{\ell}$, while the scalar flux calculated from the corrected diffusion equation is without the tilde.

The source correction scheme for the inner iteration proceeds as follows: using $\tilde{\psi}_g^{\ell}$, known from the previous iteration, Eq. (27) is solved for $\tilde{\psi}_g^{\ell}$. This involves one sweep through the space-angle mesh. The correction term, R_g^{ℓ}, is then calculated using Eqs. (29) and (30) and, in turn, used in Eq. (28) to calculate ϕ_g^{ℓ} to complete one cycle or one inner iteration. The steps are repeated until suitable convergence is achieved. Note that for the first inner iteration for a group, a logical first guess for the scalar flux is obtained by solving Eq. (28) with $\ell = 0$ by setting R_g to zero.

It is easy to show that if the iteration converges, it converges to the transport equation solution. Namely, drop all ℓ superscripts and set the transport scalar flux to the corrected diffusion scalar flux, $\tilde{\phi}_g = \phi_g$. Then substituting Eq. (29) into Eq. (30) yields

$$\nabla \cdot \tilde{J}_g(r) + \sigma_{R,g}(r)\tilde{\phi}_g(r) = QQ_g \quad ,$$

which is the converged transport balance equation obtained also by integrating Eq. (27) over all $\underline{\Omega}$.

The second level of iteration, the outer iteration, consists of one pass through the groups using Eqs. (27), (28), and (30) to obtain the group converged correction terms $R_g^k(r)$ and then to solve the multigroup corrected diffusion equation to generate new scalar fluxes consistent with a new fission source, if fission occurs in the problem. That is, the following multigroup diffusion equation is solved following the k^{th} outer iteration:

$$\begin{aligned} &- \nabla \cdot D_g(r)\nabla\phi_g^{k+1}(r) + \sigma_{R,g}(r)\phi_g^{k+1}(r) = Q_g(r) - R_g^k(r) \\ &+ \chi_g \sum_{g'=1}^{G} \nu\sigma_{f,g'}(r)\phi_{g'}^{k+1}(r) + \sum_{g' \neq g} \sigma_{s,g' \to g}(r)\ \phi_{g'}^{k+1}(r) \end{aligned} \tag{31}$$

The source correction scheme outlined above for using the diffusion synthetic method is an effective scheme for inhomogeneous source problems. For eigenvalue problems, Eq. (31) must be homogeneous, and it is necessary to define a different scheme for the diffusion synthetic method. The diffusion coefficient correction scheme is one such scheme. In this scheme we redefine the corrected diffusion coefficient $\underline{\underline{D}}_g(r)$ as

$$\underline{\underline{D}}_g(r) = -\frac{\tilde{J}_g(r)}{\nabla\tilde{\phi}_g(r)} \quad , \tag{32}$$

so that $R_g(r) = 0$ for all r and g. Then with $Q_g(r) = 0$, the inner iteration diffusion equation becomes

$$-\nabla \cdot \underline{\underline{D}}_g^{\ell-1}(r) \cdot \nabla\phi_g^{\ell}(r) + \sigma_{R,g}(r)\phi_g^{\ell}(r) = QQ_g(r) \quad , \tag{33}$$

and the multigroup (outer iteration) diffusion equation becomes

$$\begin{aligned} -\nabla \cdot \underline{\underline{D}}_g^{k}(r) \cdot \nabla\phi_g^{k+1}(r) + \sigma_{R,g}(r)\phi_g^{k+1}(r) &= \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} \nu\sigma_{f,g'}(r)\phi_{g'}^{k+1}(r) \\ &+ \sum_{g' \neq g} \sigma_{s,g' \to g}(r)\ \phi_{g'}^{k+1}(r) \quad , \end{aligned} \tag{34}$$

where k_{eff} is the multiplication factor for the system. The same iteration procedure is used for this diffusion coefficient correction scheme as for the source correction scheme.

For eigenvalue problems, the diffusion correction scheme has been found to accelerate the iterations as readily as the source correction scheme for inhomogeneous source problems. In fact in ONEDANT, the diffusion coefficient correction scheme is used for inhomogeneous source problems in which fission and/or upscatter is present with the source correction scheme used only for inhomogeneous source problems with downscatter and no fission.

One disadvantage to the diffusion coefficient correction scheme is that infinite and negative diffusion coefficients are possible (see Eq. (32)). If this occurs, Eq. (33) cannot be solved using current techniques. To overcome this difficulty, the removal correction scheme is employed. A corrected removal cross section is defined as

$$\tilde{\sigma}_{R,g}^{k}(r) \equiv \sigma_{R,g}(r) + \frac{R_g^k(r)}{\tilde{\phi}_g^k(r)} \quad , \tag{35}$$

where $R_g^k(r)$ is defined by Eq. (29). With this, the diffusion synthetic method is modified and Eq. (28) becomes

$$-\nabla \cdot D_g(r)\nabla\phi_g^{\ell}(r) + \tilde{\sigma}_{R,g}^{\ell-1}(r) = QQ_g(r)$$

and Eq. (31) becomes

$$
\begin{aligned}
&-\nabla \cdot D_g(r)\nabla\phi_g^{k+1}(r) + \tilde{\sigma}_{R,g}^{k+1}(r)\phi_g^{k+1}(r) \\
&= \frac{\chi_g}{k_{eff}} \sum_{g'=1}^{G} \nu\sigma_{f,g'}(r)\phi_{g'}^{k+1}(r) + \sum_{g' \neq g} \sigma_{s,g' \to g}(r)\ \phi_{g'}^{k+1}(r) \quad .
\end{aligned}
\tag{36}
$$

The iteration procedure is entirely analogous to that for the diffusion coefficient correction scheme and again, if it converges, it converges to the transport balance equation solution. This removal correction scheme is employed in eigenvalue problems or source problems with fission and/or upscatter only when the diffusion coefficient correction scheme produces negative or infinite diffusion coefficients.

IV. CARD-IMAGE INPUT FORMAT RULES

This chapter describes the various rules, restrictions, and options available to the user when creating the input for ONEDANT. First are described the details associated with free-field input since most users will likely select this form. Next is presented the information needed for user-specified input forms followed by information for fixed-field FIDO input.

Throughout this document we will use the term *card* or *card-image* to denote a single record or line of characters. Thus, an 80-column card (or card-image) refers to a line containing 80 columns into which ASCII characters may be written one per column.

A. Free-Field Input

1. Card-Image Ground Rules

(a) Eighty (80) columns available.

(b) No special columns; that is, no column is treated any differently than any other column.

2. Delimiters (Separators) and Terminators

(a) Data Item Delimiter (Separator): one or more blanks, a comma, or end of card:

Note: Hereafter, when an item is referred to as being delimited, for example, delimited T, it means that the item must be separated from other data items by a blank, comma, or end of card.

(b) Card Terminator: Slash (/), delimiting not required. All entries on a card beyond the slash are ignored.

(c) Block Terminator: Delimited T. Information beyond the T on the card will be ignored.

(d) Array Terminator: New array name or Block Terminator.

(e) String Delimiter: Semicolon (;), delimiting not required on ;.

(f) String Terminator: Semicolon or new array name or Block Terminator.

(g) Data Item Terminator: Data item delimiter (Separator) or any of the above Terminators.

3. Numerical Data Item Ground Rules

(a) Must not contain embedded blanks.

(b) Must not contain any nonnumeric characters except for E (for exponent), decimal point, or plus and minus signs.

4. Hollerith Data Item Ground Rules

(a) Must begin with alphabetic character [see (c) below for exception] and may contain from one to eight characters.

(b) Must not contain any of the following characters: =, $, *, blank, comma, slash, semicolon, double quote ("). [See (c) below for exception.]

(c) Hollerith data words may be delimited with double quotes to override (a) and (b) restrictions. For example, PU/239 is not allowed, but "PU/239" is allowed.

5. Array Identification and Ordering

(a) To identify an array for which data entries are to be made, one simply enters the appropriate array name, followed by an equal (=) sign (no space between name and =) and then enters the desired data, for example, CHI= 0.95, 0.10 0.05 0.0 .

(b) Within a given Block, arrays may be entered in any order.

6. Block Identification and Ordering

(a) No explicit Block identification is required. Array identification is sufficient to tell the code which Block is involved. Recall, however, that a Block Terminator (delimited T) must be entered when all input arrays for a given Block have been entered.

(b) Blocks must be ordered.

7. Input Data Operators. Several data operators are available to simplify the input. Most of these operators are FIDO operators, but several are new and represent extensions to FIDO.

IMPORTANT NOTE: The following data operators can only be used with arrays containing integer, real, or a combination of integer/real data entries. They are NOT usable with arrays that may contain Hollerith data items.

In free-field the data operators are specified in the general form

n ∅ d

where

n is the "data numerator," an integer, or blank;

∅ is any one of several "data operators" described in Table IX; and

d is a "data entry" (may be blank for some operators).

NOTE: When a "data numerator" is required with a "data operator," there must be no space between the data numerator and the data operator. There may be any number of blanks between the data operator and the "data entry" if the latter is required.

In entering data using data operators, it is convenient to think of an index or pointer that is under the control of the user and which specifies the position in the data string into which the next data item is to go. The pointer is always positioned at string location number 1 when either an array identifier or a string terminator (;) is entered.

B. User-Specified Input Formats

If desired, input data for an array can be provided in a format specified by the user. To specify the format for the input data to an array, the user can use the characters U or V as follows:

U Operator:

(1) Enter the array identification and follow this with a delimited U.

(2) On the next card-image enter the desired format enclosed in parentheses anywhere in columns 1-72.

(3) On the next and succeeding cards enter the data using ordinary FORTRAN rules.

Example: CHI= U

(6E12.5)

data in 6E12.5 format

V Operator:

Has same effect as U except the desired format is not entered; instead the format read in the last preceding U array is used.

In the table below an entry of – for either the data numerator or data entry indicates that the item is not required for the particular data operator.

TABLE IX
FREE-FIELD DATA OPERATORS

Data Numerator	Data Operator	Date Entry	Remarks
n	R	d	REPEAT OPTION: Enter the data entry d n successive times in the current data string. Example: 3R 0.0 → 0.0 0.0 0.0
n	I	d	LINEAR INTERPOLATE OPTION: Enter the value d into the data string followed by n interpolated entries equally spaced between d and the next data entry. Allowed for both real and integer data although the spacing between interpolated integer datapoints must be integer. Example: 3I1, 5 → 1 2 3 4 5 but 3I1, 4 will cause an error if the array data type is integer.
n	L	d	LOGARITHMIC INTERPOLATE OPTION: The effect is the same as that of "I" except that the resulting interpolates are equally separated in log-space.
–	F	d	FILL OPTION: Fill the remainder of the data string with the value d.
n	Z	–	ZERO OPTION: Enter the value zero in the data string n successive times. Example: 4Z → 0 0 0 0
n	S	–	SKIP OPTION: Causes the pointer to skip n positions in the current string leaving the data values in those positions unchanged.
n	A	–	POSITIONING OPTION: Set the pointer in the current data string to the n^{th} data item position in that string. Example: Suppose the array SOURCX is to be a single string of length 100, all values of which are 0.0 except the 15^{th} value which is to be unity. This can be entered simply as SOURCX= 100Z 15A 1.0

TABLE IX (Continued)
FREE-FIELD DATA OPERATORS

Data Numerator	Data Operator	Date Entry	Remarks
n	Q	m	SEQUENCE REPEAT OPTION 1: Enter the last m data entries into the current string in sequence n successive times. Example: 1 2 3 1 2 3 1 2 3 can be input at 1 2 3 2Q3.
n	G	m	SEQUENCE REPEAT OPTION 2: Same effect as "Q" except the sign of each entry in the sequence is reversed each time the sequence is repeated. Example: 1 2 -1 -2 1 2 can be input as 1 2 2G2.
n	N	m	SEQUENCE REPEAT OPTION 3: Same effect as "Q" except the order of the sequence is reversed each time the sequence is entered. For example, 4 5 6 6 5 4 4 5 6 can be input as 4 5 6 2N3.
n	M	m	SEQUENCE REPEAT OPTION 4: Same effect as "N" except the sign of each entry in the sequence is reversed each time the sequence is entered. Example: 1 2 3 -3 -2 -1 1 2 3 can be input as 1 2 3 2M3.
n	Y	m	STRING REPEAT OPTION: Enter the preceding m strings of data into the current array n successive times. (For multistring arrays only.)
n	X	-	COUNT CHECK OPTION: Causes code to check the number of data items entered into the current string to see if the number of items equals n. If count is not correct, an error message will be printed, an attempt will be made to continue processing all remaining input, and then the problem will be halted. (Error diagnostic aid.)
-	E	-	END OF STRING OPTION: Causes pointer to skip to the end of the current data string leaving values of skipped data items unchanged.
n	C	d	SEQUENCE MULTIPLY OPTION: Enter the last n data entries, multiplied by d, into the current data string.

C. Fixed-Field FIDO Input

1. Card-Image Ground Rules

(a) Seventy-two (72) columns available.

(b) Each card divided into six "fields" of 12 columns each.

(c) Each 12-column "field" subdivided into three subfields containing 2, 1, and 9 columns, respectively. Hereafter these subfields will be referred to as Subfield 1 (the first two columns of each field), Subfield 2 (the third column in each field), and the Data Subfield (the remaining nine columns in each field).

2. Delimiters (Separators) and Terminators

(a) Data Item Delimiter (Separator): Field and subfield column boundaries.

(b) Card Terminator: Slash (/) in second subfield with first subfield blank. All entries following the slash on the card are ignored.

(c) Block Terminator: T in second subfield of any field. All entries beyond the T on that card are ignored.

(d) Array Terminator: New array identified in first and second subfields of next field, or a Block Terminator.

(e) String Delimiter: Semicolon (;) in second subfield of any field. Data subfield of that field is ignored.

(f) String Terminator: Semicolon or Array Terminator or Block Terminator.

3. Numerical Data Ground Rules

(a) Standard FORTRAN convention.

(b) Data items entered in third subfield (the data subfield) in each field only.

4. Hollerith Data Item Ground Rules

Hollerith data not allowed.

5. Array Identification and Ordering

(a) To identify an array for which data are to be entered in the fixed-field FIDO format, one simply enters the array number (integer, $\leq$99) in the first subfield of any field followed by the array-type indicator (array purpose character) in the second subfield. If the array data is integer (fixed point), the array-type indicator is the dollar sign ($); if the array

data is real (floating point), the array-type indicator is an *asterisk* (*). The third subfield is left blank.

(b) Arrays may be entered in any order within a given Block.

6. Block Identification and Ordering

(a) No explicit Block identification required. Array identification is sufficient to tell the code which Block is involved. Recall that a Block Terminator (T) must be entered when all input arrays for a given block have been entered.

(b) Blocks must be ordered.

7. Input Data Operators The fixed-field FIDO data operators are the same operators used in the free-field input shown in Table IX. In fixed-field FIDO usage of these operators, however, the following rules must be observed:

(i) the "data numerator," if required, must be entered in the first subfield of a field;

(ii) the "data operator" must be entered in the second subfield of a field; and

(iii) the "data entry," if required, must be entered in the third subfield of a field.

V. ONEDANT CODE PACKAGE INPUT SPECIFICATIONS

Card-image input to the ONEDANT code package consists of (i) Title Card Control Input (always required) and subsequent Title Cards, and (ii) up to six BLOCKS of input data with each BLOCK pertaining to a specific class of input.

This chapter contains a brief overview of the specification of input followed by a Mini-Specification Sheet to be used as a quick reference for the input BLOCKS and their contained data arrays. The third section provides a relatively detailed description of the input specifications.

A. Overview of the Specification of Input

The ONEDANT card-image input consists of a Title Card section followed by six blocks of input. The blocks are:

Block	Name	Purpose
I.	Controls & Dimensions	Provides basic parameters associated with the physical problem model for purposes of data storage requirements. Also provides special code execution controls.
II.	Geometry	Provides a description of the physical problem model geometry.
III.	Cross Sections	Provides the input Isotopic microscopic cross sections.
IV.	Mixing	Provides mixing instructions for forming macroscopic Material cross sections from the input Isotopic microscopic cross sections, and provides instructions for assigning the macroscopic Materials to the Zones of the physical problem model.
V.	Solver Input	Provides input specifications and data for the particular calculation to be effected by the one-dimensional, multigroup, discrete-ordinates, diffusion synthetic accelerated Solver Module.
VI.	Edits	Provides information that allows editing of the fluxes output by the Solver Module and that causes subsequent edit calculations using those fluxes, together with cross sections, response functions, etc., to be performed.

BLOCKS MUST APPEAR IN THE INPUT STREAM IN THE ABOVE ORDER ! !

The Title Card section and BLOCK I input are always required in every run; the remaining blocks are required only as necessary for the particular run desired. For example, the input associated with BLOCK II (Geometry) may be omitted if the geometry specifications are being supplied from a previously created GEODST standard interface file.

BLOCKS II, III, and IV are all directly associated with the INPUT module of ONEDANT. BLOCK V is associated solely with the SOLVER module, which effects the solution of the multigroup, discrete-ordinates, one-dimensional, steady-state, diffusion synthetic accelerated transport equation. BLOCK VI is associated uniquely with the EDIT module of ONEDANT.

BLOCKS consist of one or more ARRAYS and a BLOCK TERMINATOR, the character "T." Each block is identified solely by the arrays comprising the block so that the presence of one or more of these arrays in the input establishes the existence of that block of input. Arrays may be entered in any order within a given block. Blocks, however, must appear in ascending numerical order. If a block is to be omitted, everything associated with that block including the terminal T, is omitted.

ARRAYS are comprised of one or more DATA STRINGS, each of which contains one or more DATA ITEMS. The majority of the ONEDANT input arrays contain only a single string and can be thought of as one-dimensional arrays or vectors. The term STRINGED ARRAY refers to any array containing more than one string of data, that is, a multidimensional array. Arrays are identified by either a unique array Hollerith name (up to six characters) or an array number. Input to a given array is terminated by either (i) the appearance of a new array name or number, or (ii) the appearance of the block terminator, T, in the input stream.

In STRINGED ARRAYS, data strings are delimited by the special STRING DELIMITER, the semicolon (;).

Data items are separated (delimited) by either (i) a blank, (ii) a comma, or (iii) the end of a "card." A data item is terminated by any of the separators above or by the semicolon, an array name (or number), a block terminator, T, or the slash, /.

Card-image input to ONEDANT is processed by a very fiexible input routine that supports a variety of input forms. One of the forms is the FIDO input form used in numerous existing codes. Users who are familiar with this form may thus readily produce input for ONEDANT without having to learn a new input form. Available also is a free-field capability particularly useful for input via remote terminals. Associated with the free-field feature is the extension of the FIDO-like options to include both Hollerith and mixed data-type input. With this capability it is possible to supply input containing real, integer, and Hollerith data in the same string.

For each array there is associated both a Hollerith name (up to six alphanumeric characters) and a number. Either name or number uniquely identifies the array for which input is to be supplied. To distinguish the array name or number from subsequent data items, there is appended to the name or number an array identifier. For free-field input the array identifier suggested is the equal sign (=). This character can be used for any data-type input, real, integer, Hollerith, or any combination, so long as the data is entered in free-field format. If the array's data

is entirely integer data and is to be entered in fixed-field FIDO format, the array identifier for that array is the dollar ($) sign; for a real data array input in fixed-field FIDO format, the asterisk (*) array identifier character is to be used. As an example, consider the input for specifying the coarse-spatial-mesh boundaries. In ONEDANT the array name for this input is XMESH, a BLOCK II array. If the desired mesh boundaries are to be at 0.0, 10.0, 20.0, and 30.0 cm, the data could be entered as:

XMESH= 0, 10, 20, 30.

Options are available to repeat data items, to produce numbers interpolated between two numeric data items, and to perform other more sophisticated operations designed to reduce the volume of input. The interpolate option, for example, can be used in the preceding XMESH example input where two equally spaced interpolation points can be placed between the entries 0.0 and 30.0. Thus, the above example can be input as:

XMESH= 2I 0, 30,

where the data operator I denotes interpolation. Another example of a commonly used operation is the repeat option. Suppose that in the array named LBEDO one wishes to enter the value 0.7 twelve successive times. This can be accomplished by entering

LBEDO= 12R 0.7,

where the data operator R denotes repeated entry.

The full list of available operations is described in Chapter IV.

One additional option for the user available in ONEDANT involves the inputting of data to certain arrays whose data members are a collection of single-word, independent control parameters. For these arrays the user can either input the collective array name followed by the data entries for the control parameters comprising that particular collective array in proper order, or the user can input the individual array members by their unique member names individually. As an example, the collective array named SOLIN in BLOCK V contains the 5 control parameters IEVT, ISCT, ITH, IBL, and IBR. Suppose that IEVT=2, ISCT=3, ITH=1, IBL=1, IBR=0. Input can be provided as either

SOLIN= 2 3, 1, 1, 0 (using collective array name, SOLIN)

or

IEVT= 2, IBR= 0, IBL= 1, ISCT= 3, ITH= 1.

In the input specifications that follow, when the contents of an array have listed named members, input may be effected using either the collective array name or the individual member names.

B. ONEDANT INPUT: Mini-Specification Sheet

Comments:

- Collective arrays, for example, DIMENS in BLOCK I, may be specified in the input by either using the collective array name, followed by =, followed by the requisite number of ordered data entries, or by using the names of the individual data members, each followed by = and the data value, for example, IGEOM= 1, NGROUP= 42.

- Entries within brackets, [ ], indicate the number of entries in the array. If the entry is of the form [I;J], then the array consists of J strings of data each of which contains I data words. [–] means that the number of entries is arbitrary.

- For those arrays amenable to fixed-field FIDO input, the appropriate array number and FIDO array-type indicator is shown in parentheses below the array name.

TITLE CARD CONTROL
(3I6 Format)
NHEAD,NOTTY,NOLIST

TITLE CARDS
(*iff NHEAD* > 0)

BLOCK I: CONTROLS and DIMENS.

DIMENS	[8]	*or*	IGEOM
			NGROUP
			ISN
			NISO
			MT
			NZONE
			IM
			IT
MAXLCM	[1]		
MAXSCM	[1]		
NOEXEC	[3]	*or*	NOFGEN
			NOSOLV
			NOEDIT
NOFILE	[7]	*or*	NOGEOD
			NOMIX
			NOASG
			NOMACR
			NOSLNP
			NOEDTT
			NOADJM

T

BLOCK II: GEOMETRY

XMESH	[IM + 1]
XINTS	[IM]
ZONES	[IM]

T

BLOCK III: CROSS SECTIONS

LIB	[1]		
LNG	[1]		
BALXS	[1]		
NTICHI	[1]		
CHIVEC	[NGROUP]		
WRITMXS	[1]		

Following arrays used only if
LIB= an XSLIB-formatted file

CARDS	[9]	*or*	MAXORD
			IHM
			IHT
			IHS
			IFIDO
			ITITL
			I2LP1
			SAVBXS
			KWIKRD
NAMES	[NISO]		
EDNAME	[IHT-3]		
NTPI	[NISO]		
VEL	[NGROUP]		
EBOUND	[NGROUP+1]		

T

Iff LIB=ODNINP, then insert the ASCII card-image cross sections here.

BLOCK IV : MIXING

PREMIX	[-;-]
MATLS	[-;MT]
ASSIGN	[-;NZONE]
ASGMOD	[-;-]
CMOD	[1]
MATNAM	[MT]
ZONNAM	[NZONE]
MATSPEC	[-]
ATWT	[-]

T

BLOCK V : SOLVER INPUT

SOLIN	[5]	or	IEVT
			ISCT
			ITH
			IBL
			IBR
ITER	[6]	or	EPSI
			EPSO
			IITL
			IITM
			OITM
			ITLIM
TRCOR	[1]		
MISC	[7]	or	BHGT
			BWTH
			NORM
			INFLUX
			INSORS
			IQUAD
			I2ANG

..... continued .....

..... SOLVER INPUT cont. .....

SOLOUT	[8]	or	FLUXP
			XSECTP
			FISSRP
			SOURCP
			GEOMP
			ANGP
			RAFLUX
			BALP
RMFLUX	[1]		
CHI	[NGROUP;M]		
DEN	[IT]		
WGT	[MM]		
MU	[MM]		
LBEDO	[NGROUP]		
RBEDO	[NGROUP]		
SOURCE	[NGROUP]		
SOURCX	[IT;N]		
SOURCF	[IT;M]		
SILEFT	[NGROUP]		
SALEFT	[K;NGROUP]		
SIRITE	[NGROUP]		
SARITE	[K;NGROUP]		
SEARCH	[8]	or	IPVT
			PV
			EV
			EVM
			XLAL
			XLAH
			XLAX
			POD
RM	[IM]		

T

BLOCK VI: EDITS

PTED	[1]
ZNED	[1]
POINTS	[<IT]
EDZONE	[IT]
ICOLL	[NBG] ,NBG < NGROUP
IGRPED	[1]
BYVOLP	[1]
AJED	[1]
POWER	[1]
MEVPER	[1]
EDOUTF	[1]
RZFLUX	[1]
RZMFLX	[1]
EDXS	[<NEDT]
RESDNT	[1]
EDISOS	[<NISO]
EDCONS	[≤NISO]
EDMATS	[≤MT]
RSFNAM	[M] ,M arbitrary
RSFE	[NGROUP;M]
RSFX	[IT;M]
XDF	[IT]
MICSUM	[-]
IRSUMS	[-]

T

C. Input Specifications

CARD 1: TITLE CARD CONTROL
{Always Required}
Format 3I6

Word	Variable Name	Comments
1	NHEAD	Number of title (header) cards to follow.
2	NOTTY	Selected output to on-line user terminal? 0/1=Yes/No (Default=0)
3	NOLIST	Listing of all card-image input with the output? 0/1=Yes/No (Default=0)

CARDS 2 Through NHEAD+1: TITLE CARDS
{Required if NHEAD > 0}
Format 12A6

NHEAD Title Cards containing descriptive comments about the ONEDANT run.

BLOCK I: CONTROLS AND DIMENSIONS
{Always Required}

Array Specification			Array Contents and Descriptive Comments		
Name	Number {Present if}	[Length]	Word	Name	Comments
DIMENS	–	[8]	...	...	Basic parameters for determining storage
	{Always}		1	IGEOM	Sets geometry desired. Enter one of the following integers or Hollerith names: 1 or PLANE or SLAB or SLAB2ANG (plane or slab geometry). If SLAB2ANG is specified, the problem will be run as a 2-angle plane calculation and the BLOCK V parameter I2ANG need not be entered. 2 or CYLINDER or CYL (cylindrical geometry) 3 or SPHERE or SPH (spherical geometry)
			2	NGROUP	Number of energy groups
			3	ISN	S_N Angular quadrature order (even number)
			4	NISO	Number of isotopes on basic input cross-section library
			5	MT	Number of materials to be created
			6	NZONE	Number of geometric zones in problem (each neutronically homogeneous)
			7	IM	Number of coarse spatial mesh intervals
			8	IT	Total number of fine spatial mesh intervals
					Note: A positive integer must be entered for each of the above
MAXSCM	– {Optional}	[1]	...	...	Length of small (fast) core memory (SCM) desired (Default=40000_{10})
MAXLCM	- {Optional}	[1]	...	...	Length of large core memory (LCM) desired (Default=140000_{10})

Note: The following BLOCK I arrays provide specialized options for the more advanced user (See Chapter XIII).

Array Specification			Array Contents and Descriptive Comments		
Name	Number {Present if}	[Length]	Word	Name	Comments
NOEXEC	- {Optional}	[3]	...	...	Execution suppression flags.
			1	NOFGEN	Suppress any further execution of the INPUT Module, i.e., generate no files (all desired files exist): 0/1=No/Yes (Default=0)
			2	NOSOLV	Suppress execution of SOLVER Module: 0/1=No/Yes (Default=0)
			3	NOEDIT	Suppress execution of EDIT Module: 0/1=No/Yes (Default=0)
NOFILE	- {Optional}	[7]	...	...	Interface file suppression flags.
			1	NOGEOD	Suppress the generation of a GEODST file even though BLOCK II card input is present. 0/1=No/Yes (Default=0)
			2	NOMIX	Suppress the generation of the NDXSRF and ZNATDN standard interface files even though mixing specification input exists: 0/1=No/Yes (Default=0)
			3	NOASG	Suppress the generation of the ASGMAT code-dependent interface file even though BLOCK IV material-to-zone assignment input exists: 0/1=No/Yes (Default=0)
			4	NOMACR	Suppress the generation of the MACRXS and SNXEDT code-dependent interface files even though cross-section and mixing specification input exists: 0/1=No/Yes (Default=0)
			5	NOSLNP	Suppress the generation of the SOLINP code-dependent interface files even though BLOCK V input exists: 0/1=No/Yes (Default=0)

BLOCK I: CONTROLS AND DIMENSIONS (Continued)

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word	Name	Comments
			6	NOEDTT	Suppress the generation of the EDITIT code-dependent interface files even though BLOCK VI input exists: 0/1=No/Yes (Default=0)
			7	NOADJM	Suppress the generation of the ADJMAC code-dependent interface files even though an adjoint calculation is called for: 0/1=No/Yes (Default=0)

BLOCK II: GEOMETRY
{ Required} unless existing GEODST file is to be used

Array Specification			Array Contents and Descriptive Comments		
Name	Number {Present if}	[Length]	Word	Name	Comments
XMESH	1* {Always}	[IM+1]	...	...	Coarse spatial mesh interval boundaries.
XINTS	4$, {Always}	[IM]	...	...	Number of equally spaced fine-mesh intervals in each coarse-mesh interval NOTE: $\sum_{I=1}^{IM} XINTS(I) = IT$
ZONES	7$, {Always}	[IM]	...	...	Zone number for each coarse-mesh interval (The number 0 (zero) may be used to specify that a coarse-mesh interval is a pure void. A "0" does not count as a zone.)

BLOCK III: CROSS SECTIONS
{Normally Required}

Array Specification			Array Contents and Descriptive Comments		
Name	Number {Present if}	[Length]	Word	Name	Comments
LIB	– {Always}	[1]	...	...	Source of cross-section data. Enter one of the following Hollerith names:
				GRUPXS	Group-ordered standard interface file.
				ISOTXS	Isotope-ordered standard interface file.
				ODNINP	Card-image ASCII library supplied immediately following this BLOCK III.
				XSLIB	Card-image ASCII library supplied as a separate file named XSLIB.
				BXSLIB	Binary library supplied as a separate file named BXSLIB.
				XSLIBB	ASCII form of BXSLIB file.
				MACRXS	Use existing files named MACRXS for for SOLVER Module, SNXEDT for EDIT Module. Under this option, unless otherwise specified in BLOCK I, any PREMIX and MATLS input in BLOCK IV will be ignored.
				MACBCD	ASCII form of MACRXS file.
				MENDF	A Los Alamos multigroup library based on ENDF data. The code will search the user's local file space for a file named MENDF. If no such file is found, it will use the public file named MENDF5. MENDF5 contains 30-group P_4 cross sections for 99 isotopes. This option only available on Los Alamos mainframe computers.

BLOCK III: CROSS SECTIONS (continued)
{Normally Required}

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word	Name	Comments
				MENDFG	A Los Alamos multigroup gamma-only library. The code will search the user's local file space for a file named MENDFG. If no such file is found, it will use the public file named MENDF5G. MENDF5G contains 12 group, P_4 gamma-only cross sections for 99 isotopes. This option only available on Los Alamos mainframe computers.
				OTHER	If a name other than those listed above is entered, the code will use the file with that name, provided that file exists in the user's file space. Such a file must be structured as an XSLIB file.
LNG	– {Optional}	[1]	...	...	Number of the last neutron group in a coupled neutron-photon library. If LNG=0, a gamma-only calculation will be performed. LNG=0 must be used for gamma-only calculations.
BALXS	– {Optional}	[1]	...	...	-1=balance cross sections by adjusting absorption cross section. 0=do not balance (default) 1=balance cross sections by adjusting self-scatter cross section.
NTICHI	–	[1]	...	...	MENDF fission fraction to use for the problem. 1/2/3=Pu239/U235/U238 {Default=U235}. May be overridden by CHIVEC input (see below) or by zone-dependent CHI input in Biock V.
CHIVEC	–	[NGROUP]	...	...	Fission fraction born into each energy group. Zone dependent CHI input in BLOCK V will override this input.

BLOCK III: CROSS SECTIONS (continued)
{Normally Required}

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word	Name	Comments
WRITMXS	{Present if} – {Optional}	[1]	...	...	Controls the code's preparing and writing certain ASCII cross section files. Enter one of the following Hollerith names:
				MACBCD	Creates the cross-section file named MACBCD, an ASCII image of the MACRXS binary file.
				XSLIBB	Creates the cross-section file named XSLIBB, an ASCII image of the BXSLIB binary file.
				XSLIBE	Creates the cross-section file named XSLIBE, an ASCII XSLIB file derived from, and corresponding to, the MACRXS binary file. XSLIBE is in Los Alamos 6E12 format (IFIDO=0).
				XSLIBF	Creates the cross-section file named XSLIBF, an ASCII XSLIB file derived from, and corresponding to, the MACRXS binary file. XSLIBF is in FIDO fixed-field format. (IFIDO=1). Available only at Los Alamos.

Note: The remaining arrays in Block III are used only if the source of the cross-section data is ODNINP or XSLIB. Exception: If ISOTXS or GRUPXS contain a non-standard treatment of the 2L+1 term in the scattering, see the I2LP1 input below.

CARDS	– {Required}	[9]	...	...	Basic parameters for card-image ODNINP or XSLIB-formatted cross-section libraries. (Refer to Chapter VIII for details.)
			1	MAXORD	Highest Legendre order in scattering tables {Required}
			2	IHM	Number of rows in a cross-section table {Required}
			3	IHT	Row number of total cross-section in table {Required}
			4	IHS	Row number of self-scatter cross-section in table {Default=IHT+1}
			5	IFIDO	Format of cross-section library: 0/1/2=Los Alamos (DTF)/Fixed-Field FIDO/Free-Field FIDO {Default=0}

Array Specification			Array Contents and Descriptive Comments		
Name	Number {Present if}	[Length]	Word	Name	Comments
			6	ITITL	Title card precedes each table? 0/1=No/Yes {Default=0}
			7	I2LP1	Higher order scatter cross sections in the supplied library contain the 2L+1 factor, where L=Legendre scattering order? 0/1=No/Yes {Default=0} Note: For a nonstandard ISOTXS or GRUPXS that contains the 2L+1 terms, set I2LP1=1.
			8	SAVBXS	Create and save binary form of card-image library as file BXSLIB? 0/1=No/Yes {Default=No}
			9	KWIKRD	Process FIDO format, card-image library with fast processor at the sacrifice of error checking? 0/1=No/Yes {Default=Yes}
NAMES	– {Optional}	[NISO]	...	...	One to six character, left-justified Hollerith name for each isotope in the library {Default=ISOn where n is the integer position of the isotope in the library }
EDNAME	– {Optional}	[NISO]	...	...	One to six character, left-justified Hollerith name for the edit positions in the cross-section table (those positions preceding σ_a) {Default=EDIT1, EDIT2, etc. }
NTPI	10\$, {Required if scattering order varies with Isotope}	[NISO]	...	...	Number of Legendre scattering orders for each isotope in the library. {Default=MAXORD+1 }
VEL	11*, {Required if doing α calculation; see IEVT}	[NGROUP]	...	...	Particle speed for each energy group.
EBOUND	12*, {Optional}	[NGROUP+1]	...	...	Energy boundaries for each energy group.

BLOCK IV: MIXING
{Normally Required}

Input in this block describes the nuclide mixing that ultimately creates the macroscopic cross-section sets that are assigned to the zones in the physical problem model.

The general procedure involves first mixing the isotope cross-section data from a basic library (defined in BLOCK III) to create material cross-section sets and then to assign one or more materials to each zone to define the macroscopic data for that zone.

Material cross sections are "permanently stored" in energy-group order and are the only cross-section data available to either the Solver or Edit Modules. These materials are defined by the MATLS array described below.

For the user's convenience, optional temporary mixtures (or premixes) can be created. These temporary mixtures are defined by the PREMIX array described below and can be used as components of materials. The temporary mixtures are indeed temporary; they are not stored and are thus forgotten by the code once the materials have been created.

The assignment of materials to zones is achieved through the ASSIGN and ASGMOD arrays described below. Additional optional input through the arrays MATNAM, ZONNAM, CMOD, MATSPEC, and ATWT is also described.

PREMIX (Temporary Mixture) SPECIFICATIONS {Optional}

FORM: PREMIX= $PREID_a$ $COMP_1$ DEN_1 $COMP_2$ DEN_2 ... ;
$PREID_b$ $COMP_3$ DEN_3 $COMP_4$ DEN_4 ... ;

where

$PREID_a$, $PREID_b$, etc., are unique Premix identifiers. IT IS STRONGLY RECOMMENDED THAT PREMIX IDENTIFIERS BE ONE TO SIX CHARACTER HOLLERITH NAMES BEGINNING WITH AN ALPHABETIC CHARACTER!

$COMP_1$, $COMP_2$, etc., are unique identifiers for the constituents or components of the premix being specified. A component identifier may refer to either an isotope or to another premix (a premix may not be a component of itself). If the component identifier refers to an isotope from the basic input cross-section library, the identifier may either be (i) the Hollerith name of the isotope or (ii) an integer, I($1 \leq I \leq NISO$), in which case I refers to the I^{th} isotope on the basic cross-section library or (iii) a floating point number, called a ZAID, uniquely identifying an isotope on a MENDF or MENDFG library. The ZAID must be of the form ZZAAA.NN where ZZ is the atomic number, AAA is the isotopic mass number, and NN is a two-digit number specifying a particular version of the cross sections for the given isotope. If the component identifier refers to a premix, the identifier should be the same as the premix identifier used when specifying that premix.

DEN_1, DEN_2, etc., are the atom densities, volume fractions, etc., associated with the immediately preceding components. NOTE THAT SEMICOLONS (;) MUST SEPARATE THE SPECIFICATIONS FOR EACH PREMIX!

The mixing of isotopes, nuclides, premixes, etc., to create the materials used in the physical problem model is done via the MATLS array.

FORM 1: MATLS= ISO$\supset$

With this abbreviated form, "isotopes" from the basic input cross-section library (see **BLOCK III**) are directly designated as materials such that the first isotope on the library becomes the first material, the second isotope becomes the second material, etc. Similarly, the Hollerith isotope names are directly used for the Hollerith material names.

NOTE: With this form, the number of materials, MT, will normally be equal to the number of isotopes, NISO. If MT $\neq$ NISO, then MT $<$ NISO is REQUIRED!

FORM 2: MATLS= $MATID_a$ $COMP_1$ DEN_1 $COMP_2$ DEN_2 ... ; $MATID_b$ $COMP_3$ DEN_3 $COMP_4$ DEN_4 ... ; etc.,

where

$MATID_a$, $MATID_b$, etc., are unique identifiers for each of the MT materials to be specified. The MATIDs may be input as either (i) unique Hollerith names (one to six characters beginning with an alphabetic character), or (ii) a unique integer in the range 1 to MT, inclusive. If Hollerith names are used for the material identifiers, the first named material is indexed as material 1, the second named material is indexed as material 2, etc. If integers are used for the material identifiers, the integer denotes the index or position of the material in the ordered list of MT (total number of) materials. Additionally, if integers are used for the MATIDs, Hollerith names for the materials may be optionally provided by the user via the MATNAM array described below. (If names are not provided, the default Hollerith material name MATn will be created by the code where n is the integer material identifier.)

NOTE: All material identifiers (MATIDs) must be of like form (Hollerith name or integer).

$COMP_1$, $COMP_2$, etc., are unique identifiers identifying either isotopes from the basic library or premixes to be used as components or constituents of the

* **see Chapter IX for exceptions**

material identified by the immediately preceding MATID. If the desired component is an isotope, then the component identifier, COMP, may either be (i) the unique Hollerith name associated with the isotope or (ii) a unique integer in the range 1 to NISO denoting the index or position of the desired Isotope in the ordered list of isotopes in the basic library or (iii) a fioating point number, called a ZAID, uniquely identifying an isotope on a MENDF or MENDFG library. The ZAID must be of the form ZZAAA.NN where ZZ is the atomic number, AAA is the isotopic mass number, and NN is a two-digit number specifying a particular version of the cross sections for the given isotope. If the component identifier refers to a premix, the identifier should be the same as the premix identifier used when specifying that premix. If the desired component is a premix, then the component identifier, COMP, should exactly match the premix identifier used in the PREMIX array specifications.

DEN_1, DEN_2, etc., are the atom densities, volume fractions, atomic fractions, etc., associated with the immediately preceding components.

NOTE 1: EACH MATERIAL SPECIFICATION MUST BE SEPARATED FROM THE NEXT BY A SEMICOLON (;).

NOTE 2: THERE MUST BE EXACTLY MT MATERIALS SPECIFIED.

FORM 2 SPECIAL FEATURE: The material specification string characterized above MATID $COMP_1$, DEN_1, $COMP_2$, DEN_2, ... ; can be input in the abbreviated form

MATID COMP ;

if the material identified by MATID contains only a single component (isotope or premix) identified by COMP, the component identifier, with a density of 1.0.

BLOCK IV (continued)

ASSIGNMENT OF MATERIALS TO ZONES
{Normally* Required}

The assignments of one or more materials to each zone in the physical problem model are made via the ASSIGN array. For concentration searches and some other possible uses, additional assignment information can be input via the ASGMOD array.

THE ASSIGN ARRAY {Required if SOLVER or EDIT modules to be executed}

There are two basic forms for using the ASSIGN array.

FORM 1: ASSIGN= MATLS

With this abbreviated form, materials as defined in the MATLS array are directly assigned to zones such that the first material is assigned to the first zone, the second material to the second zone, etc. Similarly, the Hollerith material names are used directly for the Hollerith zone names.

NOTE 1: With this form, the number of zones, NZONE, will normally be equal to the number of materials, MT. If NZONE $\neq$ MT, then NZONE $<$ MT is REQUIRED!

NOTE 2: Form 1 can not be used if the ASGMOD array is used.

FORM 2: ASSIGN= $ZONID_a$, $MATID_1$, $CONC_1$, $MATID_2$, $CONC_2$, ... ; $ZONID_b$, $MATID_3$, $CONC_3$, $MATID_4$, $CONC_4$, ... ; etc.

Here

$ZONID_a$, $ZONID_b$, etc., are unique zone identifiers for each of the NZONE zones. The ZONIDs may be input either as (1) unique Hollerith names (one to six characters beginning with an alphabetic character), or (ii) a unique integer in the range 1 to NZONE, inclusive. If Hollerith names are used for the zone identifiers, the first named zone is given a zone number of 1, the second named zone is given a zone number of 2, etc. If integers are used for the ZONIDs, the integer denotes the zone number, and its value must lie in the range 1 to NZONE, inclusive. Additionally, if integers are used for the ZONIDs, Hollerith names for the zones may be optionally provided by the user via the ZONNAM array described below. (If names are not provided, the default Hollerith name ZONEn will be used, where n is the integer zone identifier, i.e., zone number.)

* see Chapter IX for exceptions

NOTE 1: All zone identifiers (ZONIDs) must be of like form (Hollerith name or integer).

NOTE 2: All zone numbers associated with the ZONIDs are the same numbers used in the ZONES array of BLOCK II.

$MATID_1$, $MATID_2$, etc., are the desired material identifiers for the materials to be used as components or constituents of the zone identified by the immediately preceding ZONID. The MATID can be either the Hollerith material name or the integer material number (see MATLS array).

$CONC_1$, $CONC_2$, etc., are the basic concentrations (densities, volume fractions, etc.) for the materials identified by the immediately preceding MATIDs.

NOTE 3: EACH ZONE'S SPECIFICATION MUST BE SEPARATED FROM THE NEXT BY A SEMICOLON (;).

NOTE 4: THERE MUST BE EXACTLY NZONE ZONES SPECIFIED.

FORM 2 SPECIAL FEATURE: The material-to-zone specification string characterized above by ZONID, $MATID_1$, $CONC_1$, $MATID_2$, $CONC_2$, ... ; can be input in the abbreviated form

ZONID MATID ;

if the zone identified by ZONID consists only of the single material identified by MATID, with a concentration of 1.0.

THE ASGMOD ARRAY {Required for Concentration Search, Optional Otherwise}

NOTE : The ASGMOD array can not be used if ASSIGN=MATLS is used.

In order to understand the use of the ASGMOD Array, it is necessary to understand the manner in which the code uses the material-to-zone assignment information.

Consider a zone, Z, containing one or more materials, M, each of which is characterized by a cross-section set $\sigma_{x,g}$(M), where x denotes a cross-section type (fission, absorption, etc.) and g denotes an energy group. The corresponding macroscopic cross section for the zone, $\Sigma^z_{x,g}$, is generated in the code by the algorithm

$$\Sigma^z_{x,g} = \sum_{M \epsilon Z} [CO(M,Z) + C1(M,Z) * CMOD] * \sigma_{x,g}(M)$$

where the values of CO(M,Z) are the CONC entries in the ASSIGN array (Form 2), and represent the basic concentrations for the materials in a given zone. The values of C1(M,Z) are concentration factors supplied by the user through the ASGMOD array described below and which are multiplied by the concentration modifier, CMOD, where

CMOD = the eigenvalue for a CONCENTRATION SEARCH

CMOD = user supplied value (see other BLOCK IV input below) for ALL OTHER PROBLEMS.

The form for entering the ASGMOD array is as follows:
ASGMOD= $ZONID_a$, $MATID_1$, $C1_1$, $MATID_2$, $C1_2$, ... $ZONID_b$, $MATID_3$, $C1_3$, $MATID_4$, $C1_4$, ... ; etc.

This form is identical to that for the ASSIGN array (except for C1 concentration factors in lieu of the basic concentration, CONC) and will not be further amplified.

NOTE 1: All ZONIDs and MATIDs used in the ASGMOD array should be of the same form (Hollerith name or integer) as those used in the ASSIGN array.

NOTE 2: The ZONIDs and MATIDs used in the ASGMOD array must be a subset of those used in the ASSIGN array, i.e., any MATID used in the ASGMOD array, for a given ZONID must also be used in the ASSIGN array but not conversely. Default values for the C1s for materials used in the ASSIGN array but not in the ASGMOD array are 0.0.

OTHER BLOCK IV INPUT
{Optional}

Array Specification		Array Contents and Descriptive Comments
Name	[Length]	
{Present if}		
CMOD {Optional}	[1]	The value of the concentration modifier used in the general algorithm for assigning materials-to-zones (see discussion under ASGMOD array in BLOCK IV). Only used if (i) ASGMOD array is input and (ii) problem being executed is not a CONCENTRATION SEARCH.
MATNAM {Optional}	[MT]	Hollerith material names for materials. Used only if the MATIDs used in the MATL array were integers. First entry in MATNAM array is Hollerith name for material number 1, second entry is Hollerith name for material number 2, etc.
ZONNAM {Optional}	[NZONE]	Hollerith zone names for zones. Used only if the ZONID entries in the ASSIGN array (and ASGMOD array, if used) were integers. First entry in ZONNAM array is Hollerith name for zone number 1, second entry is Hollerith name for zone number 2, etc.
MATSPEC {Optional}	[≤MT]	Tells code whether material mixing in the MATLS array is to use atomic densities, atomic fractions, and/or weight fractions. The MATSPEC parameter can be entered as a vector parameter with up to MT entries so that different materials can be specified with different types of mixing specifications. If the number of entries is less than MT, the last entry will be applied to all remaining and unspecified entries. The allowable entries for MATSPEC are: ATDENS–use the traditional atom density style of input for mixing specifications {Default}. ATFRAC–use the type of mixing that specifies the isotopes (nuclides) and their atomic fractions in the material together with the material's density. WTFRAC–use the type of mixing that specifies the isotopes (nuclides) and their weight fractions in the material together with the material's density. [See Ch. IX for more details]

OTHER BLOCK IV INPUT (Continued)
{Optional}

Array Specification		Array Contents and Descriptive Comments
Name	[Length]	
{Present if}		
ATWT {Optional}	[≤NISO pairs]	Atomic weights of the isotopes. If using MATSPEC= ATFRAC or WTFRAC, atomic weights must be available to the code. [See Ch. IX for more details). Entries for the ATWT array are made in pairs as follows: ATWT=ISO_1 $ATWT_1$ ISO_2 ATWT2... where ISO_i is the isotope name (identifier) for isotope isotope i on the cross section library and $ATWT_i$ is the atomic weight for isotope i.

BLOCK V: SOLVER INPUT
{Required if Executing Solver Module}

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word	Name	Comments
{Present if}					
SOLIN	–	[5]	...	...	Basic SOLVER input control words
	{Always}		1	IEVT	Type of calculation: -1 Inhomogeneous source with fission or upscatter 0 Inhomogeneous source alone 1 k_{eff} 2 α (time absorption) search 3 concentration search 4 dimension search
			2	ISCT	Legendre order of scattering
			3	ITH	Mode of calculation: 0/1=Direct (forward)/Adjoint (Default=0)
			4	IBL	Left boundary condition: 0 vacuum 1 reflective* 2 periodic 3 white* {Used only for planar geometry}
			5	IBR	Right boundary condition: Same options as for IBL {Used for all geometries}
					*Albedoes may be used with either reflective or white boundary conditions (see LBEDO, RBEDO arrays below).
ITER	–	[6]	...	...	Iteration and convergence control parameters
	{Optional}		1	EPSI	Inner iteration convergence criterion (Default=0.0001).
			2	EPSO	Outer iteration convergence criterion (Default=EPSI).
			3	IITL	Max number of transport inner iterations per group until $\|1.0 - \lambda\|$ <3* EPSO (Default chosen by code). Recommended to use default.

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word	Name	Comments
			4	IITM	Max number of transport inner iterations per group after $\|1.0 - \lambda\|$ <3* EPSO (Default chosen by code). Recommended to use default.
			5	OITM	Max number of outer iterations allowed (Default=20).
			6	ITLIM	Iteration time limit in seconds (Default=no limit).
TRCOR	– {Optional}	[1]	...	...	Apply transport correction to cross sections on MACRXS file. Valid values for this entry are: DIAG use "diagonal" transport correction (generally recommended) BHS use the Bell-Hansen-Sandmeier transport correction CESARO use the Cesaro transport correction {Default=none}
MISC	– {Optional}	[7]	...	...	Miscellaneous control parameters
			1	BHGT	Buckling height (in cm if macroscropic cross sections in cm^{-1}). Used only for plane, cylindrical and two-angle plane geometries. {Default=0.0 → ∞}
			2	BWTH	Buckling width. Used only for plane and two-angle plane geometries. {Default=0.0 → ∞}
			3	NORM	Normalization factor: 0/NORM=No normalization/Normalize fission source rate (IEVT≥1) or inhomogeneous source rate (IEVT<1) to NORM. (Integral of source rate over all angle, space, energy= NORM.) {Default=0}
			4	INFLUX	Input fiux guess: 0/1=None/ RTFLUX standard interface file. {Default=0}
			5	INSORS	Input inhomogeneous source input from FIXSRC standard interface file? 0/1=No/Yes {Default=0}

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word	Name	Comments
			6	IQUAD	Quadrature constants to be used. Valid values are: -3 use SNCONS standard interface file 1 use built-in P_N set 2 use built-in DP_N set 3 use card input via WGT, MU arrays 4 use built-in GQ_N set. (recommended for cylinders) {Default=1, but card input will override.}
			7	I2ANG	Do two-angle plane calculation? 0/1=No/Yes {Default=0} (For IGEOM=1 only.)
SOLOUT	-	[8]	...	...	Basic SOLVER output control parameters
	{Optional}		1	FLUXP	Final flux print: 0/1/2= None/ Isotropic component (scalar flux) only/All flux moments {Default=0}
			2	XSECTP	Macroscopic zone cross sections print: 0/1/2=None/Principal cross sections/All (principal plus scattering matrices). {Default=0}
			3	FISSRP	Final fission source rate density print: 0/1=No/Yes {Default=0}
			4	SOURCP	Inhomogeneous source print: 0/1/2/3=No/As input/ Normalized/Both {Default=0}
			5	GEOMP	Fine-mesh geometry print: 0/1=No/Yes {Default=0}
			6	ANGP	Final angular flux print: 0/1=No/Yes {Default=0}
			7	RAFLUX	Write angular fluxes to standard interface file RAFLUX (AAFLUX if adjoint)? 0/1=No/Yes {Default=0}
			8	BALP	Print balance tables by coarse mesh: 0/1=No/Yes

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word	Name	Comments
RMFLUX	– {Optional}	[1]	...	...	0/1=No/Yes. Prepare RMFLUX fiux moments file. (See file description in App. A)
CHI	20* {Required} Unless IEVT= 0 or unless available from library. See Block III.	[NGROUP;M]	...	...	Fission spectrum by energy group and zone. Data is entered as M strings each NGROUP data words long. M is any integer such that 1≤M≤NZONE. If M<NZONE, the last string entered, i.e., the last NGROUP values of CHI will be used for zones M+1, M+2, ..., NZONE. The SEMICOLON (;) STRING DELIMITER MUST BE USED BETWEEN STRINGS!
DEN	21* {Optional}	[IT]	...	...	Fine spatial mesh density factors to be applied to the zone macroscopic cross sections at each mesh point. (IT=no. of fine spatial mesh intervals, see BLOCK I.)
WGT	22* {Optional}	[MM]	...	...	S_N quadrature weights. For plane and spherical geometry MM=ISN(ISN+2)/4; for 2-angle plane geometry (MM= ISN*(ISN+2). For ordering of weights, see Chapter III.
MU	23* {Optional}	[MM]	...	...	S_N quadrature cosines. For ordering of cosines, see Chapter III.
LBEDO	30* {Optional}	[NGROUP]	...	...	Left boundary albedoes for each group for plane geometry (IGEOM=1) only. Applied as albedoes for either reflective (IBL=1) or white (IBL=3) boundary conditions. {Default=0}
RBEDO	31* {Optional}	[NGROUP]	...	...	Right boundary albedoes for each group for all geometries. Applied as albedoes for either refiective (IBR=1) or white (IBR=3) boundary conditions. {Default=0}

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word Name		Comments
INHOMOGENEOUS (FIXED) SOURCES (particles per unit time per unit volume) {Required if IEVT≤ 0 AND INSORS= 0 AND NO BOUNDARY (SURFACE) SOURCES}					
SOURCE	40* {Present if}	[NGROUP;N]	...	...	Inhomogeneous source energy spectra. The number, N, of strings is the number of angular source moments desired. Each string must contain NGROUP data entries. First string contains the spectrum for the isotropic component of the source. Succeeding strings contain spectra for successively higher source moments (see Ch. III for ordering of higher moments.) THE SEMI-COLON (;) STRING DELIMITER MUST BE USED BETWEEN STRINGS
SOURCX	41*	[IT;N]	...	...	Inhomogeneous source spatial distributions. The number, N, of strings is the number of angular source moments desired. Each string must contain IT (number of fine spatial mesh intervals) data entries. First string contains the spatial distribution for the isotropic component of the source. Succeeding strings contain the distributions for successively higher moments (see Ch. III for ordering of moments). THE SEMI-COLON (;) STRING DELIMITER MUST BE USED BETWEEN STRINGS

Note on use of SOURCE/SOURCX: Inhomogeneous source used by the code is formed by taking the product of energy spectrum times the spatial distribution for each moment. If one of the two arrays SOURCE or SOURCX is input and the other omitted, the omitted array is defaulted to unity.

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word Name		Comments
SOURCF	44*	[IT;M]	...	...	Full inhomogeneous source space-energy distribution. The number of strings, M, is equal to N*GROUP, where N is the desired number of source angular moments. Each string contains IT data entries. First string is group 1, spatial distribution for the isotropic component, second string is group 2 spatial distribution for isotropic component, etc. The NGROUP+1 string (if present) is the group 1 spatial distribution for the 1^{st} angular moment, etc. STRING DELIMITERS (;) MUST BE USED BETWEEN STRINGS

Array Specification					Array Contents and Descriptive Comments
Name	Number	[Length]	Word Name		Comments
BOUNDARY (SURFACE) SOURCES (particles per unit time per unit area) {Optional}					
SILEFT	50*	[NGROUP]	...	...	Fixed, isotropic boundary (surface) source at the left boundary for each energy group. FOR PLANE GEOMETRY ONLY! See Ch. X
SALEFT	51*	[K;NGROUP]	...	...	Fixed, angular boundary (surface) source at the left boundary for each inward-directed direction and each energy group. FOR PLANE GEOMETRY ONLY! K=ISN/2 for standard plane geometry K=ISN*(ISN+2)/2 for 2-angle plane geometry calculation (I2ANG=1). Entered as NGROUP strings of data, each string containing K data entries, beginning with group 1. The ordering of the angular boundary sources (fiuxes) is described in Ch. X. STRING DELIMITERS (;) MUST BE USED BETWEEN STRINGS
SIRITE	52*	[NGROUP]	...	...	Fixed, isotropic boundary (surface) source at the RIGHT boundary for each energy group beginning with group 1.
SARITE	53*	[K;NGROUP]	...	...	Fixed, angular boundary source at the right boundary for each inward-directed angular direction and each energy group. K=ISN/2 for standard plane and spherical geometries (IGEOM=1,3); K=ISN*(ISN+2)/8 for cylindrical geometry (IGEOM=2); K=ISN*(ISN+2)/2 for 2-angle plane geometry calculation (I2ANG=1 and IGEOM=1). Entered as NGROUP strings of data, each string containing K data entries, beginning with group 1. The ordering of the angular boundary sources (fiuxes) is described in Ch. X. STRING DELIMITER (;) MUST BE USED BETWEEN STRINGS!

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word	Name	Comments
SEARCH {Present if} {Always if IEVT≥2}	–	[8]	...	...	Control parameters for searches and time-absorption (α) calculation
			1	IPVT	Parametric value type: 0/1/2= None/k_{eff}/α. IPVT=1 or 2 only valid for IEVT≥2. {Default=1 if IEVT≥2}.
			2	PV	Parametric value of k_{eff} (if IPVT=1) or α (if IPVT=2). PV is a fixed quantity during a search. {Default=1.0 if IPVT=1, 0.0 if IEVT=2}.
			3	EV	Value of eigenvalue at which search is started (used only if IEVT≥2). {Default=0.0 }.
			4	EVM	Eigenvalue modifier to adjust eigenvalue at which search is started (used only if IPVT≥2). REQUIRED INPUT if IEVT≥2!
			5	XLAL	Lambda lower limit for eigenvalue searches. {Default=0.01 }
			6	XLAH	Lambda upper limit for eigenvalue searches. {Default=0.5 }
			7	XLAX	Lambda convergence criterion for second and subsequent search steps, i.e., search convergence criterion. {Default=10*EPSI}
			8	POD	Parameter oscillation damper. See Ch. X {Default=1.0 }

Array Specification			Array Contents and Descriptive Comments	
Name	Number {Present if}	[Length]	Word Name	Comments
RM	27* {Always if IEVT=4}	[IM]	... ...	Radius modifiers for each mesh interval for use with dimension searches only. For coarse-mesh interval K, the coarse-mesh spatial boundary $\tilde{R}_{K+1}$ is computed as $\tilde{R}_{K+1} = \tilde{R}_K + \left(\tilde{R}_{K+1} - \tilde{R}_K\right) * (1 + EV * RM_K) \quad k = 1, \ldots, IM$ where EV is the dimension search eigenvalue being sought and R_K, $\tilde{R}_{K+1}$ are the input coarse-mesh boundaries supplied in the XMESH array of BLOCK II.

Note on Searches: Ch. X has descriptive details on how searches are performed by the code.

BLOCK VI: EDITS
{Required if Executing EDIT Module}

Array Specification			Array Contents and Descriptive Comments		
Name	Number	[Length]	Word	Name	Comments
PTED	{Present if} – {Always}	[1]	...	...	Do fine space-point edits: 0/1=No/Yes
ZNED	– {Always}	[1]	...	...	Do EDIT ZONE edits: 0/1=No/Yes
POINTS	80$ {Optional}	[<IT]	...	...	Fine-mesh point (or interval) numbers for which point edits are desired. Must be in ascending order USED ONLY IF PTED=1. (Default=all points)
EDZONE	83$ {Optional}	[IT]	...	...	EDIT ZONE number for each fine spatial mesh interval. USED ONLY IF ZNED=1. (Default=SOLVER coarse mesh interval numbers, see XMESH ARRAY, BLOCK II)
ICOLL	84$ {Optional}	[NBG]	...	...	Edit Energy-Broad-Group collapsing option. Number of Solver energy groups in each Edit Energy-Broad-Group. NBG is the number of entries in the ICOLL array. It is required that $\sum_{N=1}^{NBG} ICOLL(N) = NGROUP$. (Default=1 energy group per Edit Energy-Broad-Group)
IGRPED	– {Optional}	[1]	...	...	Print option on energy groups: 0 print energy group total only 1 print only Edit Energy-Broad-Groups 2 same as 1 3 Print both Edit Energy-Broad-Groups and group totals {Default = 0}

Array Specification			Array Contents and Descriptive Comments	
Name	Number	[Length]	Word Name	Comments
BYVOLP	– {Optional}	[1]	... ...	Print option on fine space-point edits: Printed point reaction rates will have been multiplied by the mesh interval volume? 0/1=No/Yes {Default = 0}
AJED	– {Optional}	[1]	... ...	0/1=Regular (forward) Edit/Adjoint Edit. Regular edit uses RTFLUX scalar flux file; Adjoint edit uses ATFLUX scalar flux file {Default = 0}
POWER	– {Optional}	[1]	... ...	Normalize edits to POWER megawatts. All printed reaction rates and the fluxes on files RTFLUX and RZFLUX (if requested) will be normalized. Fluxes are normally not printed in the Edit Module although they can be extracted by using a unit response function. Any such fluxes will be normalized to POWER. Note that this normalization is independent of that invoked by the NORM parameter in BLOCK V.
MEVPER	– {Optional}	[1]	... ...	MeV released per fission (default=210 MeV). This value will be used with the calculated fission rate to determine the power. For the power normalization the code needs to know which cross section is the fission cross section (σ_f). It uses the one from the cross section library that has the name "N-FISS." If one uses an ISOTXS, GRUPXS, or MENDF library, that designation will be automatic (see Tables Xa, and Xb). If one uses a card-image library then the fission cross section must exist on the library and the name "N-FISS" must be entered in the proper place in the EDNAMES input array in BLOCK III.

Array Specification					Array Contents and Descriptive Comments
Name	Number	[Length]	Word	Name	Comments
EDOUTF	– {Optional}	[1]	...	...	Edit Module ASCII file preparation control. Permissible values are: -3 EDTOGX (without scalar fluxes) plus EDTOUT files to be produced -2 EDTOGX (without scalar fluxes) to be produced 0 Neither EDTOGX nor EDTOUT files to be produced {Default} 1 EDTOUT file only produced 2 EDTOGX (with scalar fluxes) produced 3 Both EDTOGX (with scalar fluxes) and EDTOUT files produced. (See Ch. XI.F and App. C for more details)
RZFLUX	– {Optional}	[1]	...	...	0/1=No/Yes. Prepare RZFLUX zone-averaged flux file. (See file description in Ref. 3)
RZMFLX	– {Optional}	[1]	...	...	0/1=No/Yes. Prepare RZMFLX zone-averaged flux moments file. (See file description in App. A)
					REACTION RATE/RESPONSE FUNCTION EDIT SPECIFICATIONS
EDXS	– {Optional}	[<NEDT]	...	...	Cross section types to be used in forming reaction rates. May be entered by integer (denoting edit position of desired cross-section type) or by Hollerith name of cross-section type. See Table Xa, Xb. NEDT is the total number of Edit cross-section types available from the input cross-section library. { Default=ALL} NOTE: The cross-section types specified in this array apply to any of all of the following specified edit forms: RESDNT, EDISOS, EDCONS, EDMATS.

Array Specification			Array Contents and Descriptive Comments	
Name	Number	[Length]	Word Name	Comments
RESDNT	–	[1] {Optional}	... ...	Do edits using the Resident Macroscopic cross sections: 0/1=No/Yes. By resident is meant that which actually exists at each mesh point as used in the SOLVER Module. NOTE: If density factors were used in SOLVER, the same density factors should be provided in the XDF array for the EDIT Module. (See Ch. XI)
EDISOS	–	[<NEDT] {Optional}	... ...	Isotope identifiers for isotopes to be used used in forming isotope microscopic reaction rates. Identifiers may be entered as either (i) integers in which case an integer entry of I refers to the I^{th} isotope on the basic input library, or (ii) Hollerith names of the desired isotopes. {Default=none}. (See Ch. XI)
EDCONS	–	[≤NISO] {Optional}	... ...	Isotope identifiers for isotopes in forming Resident Constituent (partial macroscopic) reaction rates. Identifiers may be entered as integers (denoting the ordered position of the isotope on the basic input library) or as Hollerith names of the desired isotopes. {Default=none}. (See Ch. XI)
EDMATS	–	[≤MT] {Optional}	... ...	Material identifiers for materials to be used in forming material (macroscopic) reaction-rate edits. Identifiers may either be integers (material numbers) or Hollerith material names. {Default=none}.
RSFNAM	–	[M] {Optional}	... ...	Hollerith names for the user-input response functions specified below. M is arbitrary but must be <500. {Default=RSFP1, RSFP2, ..., RSFPM}.

Array Specification			Array Contents and Descriptive Comments
Name	Number	[Length]	Comments
RSFE	85* {Required if user-input} {response functions desired}	[NGROUP;M] ...	Response function energy distribution for each of the M different response functions desired. The number of different response functions is arbitrary (but must be fewer than 500). Data are entered as M strings each of NGROUP entries beginning with group 1. THE SEMICOLON(;) STRING DELIMITER SHOULD BE USED.
RSFX	86* {Optional}	[IT;M] ...	Response function spatial distribution by fine mesh for each of the M different functions desired ($1 \leq M < 500$). Data are entered as M strings each of IT entries beginning with mesh point 1. THE SEMICOLON(;) STRING DELIMITER SHOULD BE USED. (Default=1.0 for all entries if RSFE array is entered.) Note: M^{th} response function at space point I, energy group G is computed as RSFX(I,M)*RSFE(G,M).
XDF	90* {Optional}	[IT] ...	Fine spatial mesh density factors used to multiply resident Constituent (see EDCONS), material macroscopic (See EDMATS), and resident macroscopic (see RESDNT) reaction rates only. (Default=all values unity).
MICSUM	- {Optional}	[<500 sums] ...	Cross-section reaction rate summing specifications. The MICSUM array is a packed array with data entered as follows: A set of isotope numbers or names is given, followed by a set of cross-section type position numbers or names (see Tables Xa and Xb). These sets are delimited with 0 an entry of (zero). Reaction rates are calculated for each isotope specified and summed to form the first sum. The next two sets of data are used to form the second sum, etc. Up to 500 sums can be specified. (See Ch. XI)

Array Specification			Array Contents and Descriptive Comments
Name	Number	[Length]	Comments
	{Present if}		
IRSUMS	–	[<500 sums] ...	Response function reactions rate summing specifications. The IRSUMS array is input as follows: A set of response function numbers or names is entered and the set delimited with an entry of 0 (zero). Reaction rates are calculated using these response functions, and the rates are summed to form the first sum. The next set of data is used to form the second sum, etc. Up to 500 sums can be specified. (See Ch. XI)
	{Optional but used only} {if RSFE array is input}		

TABLE Xa
EDIT CROSS-SECTION TYPES BY POSITION AND NAME
(for other than MENDF libraries)

Cross-Section Input Via ISOTXS/GRUPXS			Cross-Section Input Via Card Image Libraries		
	EDIT			EDIT	
Type	Position	Name[a]	Type	Position	Name[a]
χ	1	CHI$\hat{}\hat{}\hat{}$	Not Used	1	CHI$\hat{}\hat{}\hat{}$
$\nu\sigma_f$	2	NUSIGF	$\nu\sigma_f$	2	NUSIGF
σ_t	3	TOTAL	σ_t	3	TOTAL
σ_a	4	ABS	σ_a	4	ABS
$\sigma_{n,p}$	5	N-PROT	1[b]	5	EDIT1[c]
$\sigma_{n,D}$	6	N-DEUT	2[b]	6	EDIT2[c]
$\sigma_{n,T}$	7	N-TRIT	.		
$\sigma_{n,\alpha}$	8	N-ALPH	.		
$\sigma_{n,2n}$	9	N-2N$\hat{}\hat{}$	.		
$\sigma_{n,\gamma}$	10	N-GAMM	N=(IHT-3)[b]	4+N	EDITN$\hat{}$[c]
σ_f	11	N-FISS			
σ_{tr}	12	TRNSPT			

[a] Names are six character Hollerith. $\hat{}$ denotes blank.

[b] Denotes position (row) in the cross-section table. All cross sections in rows (positions) 1 through IHT-3 in the cross-section library are EDIT cross sections chosen by the user.

[c] These are the default names that may be overridden with the user-option names in the EDNAMES array of Block III.

TABLE Xb
MENDF Library EDIT CROSS SECTIONS

When using the Los Alamos MENDF5 cross section library with the S_N codes there are numerous edit cross sections available for use in the Edit Module. Since these come from the MENDF file, they are called upon with special Hollerith names in the Edit Module as part of the EXDS=input. These names are defined below:

Reaction Type	Name	
(elas scat)	MEND1	
(n,n′)	MEND2	
(n,2n)	MEND3	
(n,3n)	MWNS4	
(n,γ)	MEND5	
(n,α)	MEND6	
(n,p)	MEND7	
(n,f)	MEND8	
(n,n′)f	MEND9	
(n,2n)f	MEND10	
(n,F)	N-FISS	[(n,F) = (n,f) + (n,n′)f + (2,2n)f]
$\nu\sigma_f$	NUSIGF	
σ_t	TOTAL	
χ_p	MEND12	(only for fissionable materials)
χ_t	MEND13	(only for fissionable materials)
σ_a	ABS	

Note: σ_a is defined as

$$\sigma_t - \sum_{g'} \sigma(g \rightarrow g')$$

VI. SOME DETAILS ON INPUT OF CONTROLS AND DIMENSION PARAMETERS (BLOCK I)

The input parameters provided in Block I are used by the code to determine storage requirements for the problem being run (the code uses variable dimensioning), to provide error checking on the remaining input, and/or to control the execution-flow of the code.

In this chapter are provided details on certain of the parameters that appear in the Block I input. In some cases, the "details" are simply references to other portions of this manual.

A. Angular Quadrature-Related (ISN)

The numerical value (an even integer) entered for the parameter ISN is simply the value of N in S_N, that is, the angular quadrature order desired for the current calculation. The discrete-ordinates, or S_N, approximation is described in Ch. III and more details on angular quadrature sets supplied in the code are described in Ch. X.

B. Geometry-Related (NZONE, IM, IT)

The parameter NZONE is the number of different zones that are to be defined for the calculation. The concept and meaning of the term Zone is described in more detail in Ch. VII.

The number of coarse spatial mesh intervals for the problem being solved is denoted by the parameter IM. The concept and meaning of the term coarse mesh interval is detailed in Ch. VII.

The total number of fine spatial mesh intervals (or mesh points) for the problem being solved is given by the parameter IT. The fine mesh interval or point is described in Ch. III under Discretization of the Spatial Variable. More is said about fine mesh intervals in Ch. VII.

C. MAXSCM, MAXLCM

Originally designed for the CDC-7600 computer, the code is structured for a three-level hierarchy of data storage: a small, fast core central memory (SCM), a fast-access, peripheral large core memory (LCM), and random-access peripheral storage. (For computing systems based on a two-level hierarchy of data storage – a large fast core and random-access peripheral storage – a portion of fast core is

designated as a simulated LCM to mimic the three-level hierarchy). Random-access storage is used only if LCM (or simulated LCM) storage requirements are exceeded.

The MAXSCM parameter is Block I of the input; it allows the user to specify the size of SCM that is desired. The code requires a certain amount of SCM for execution. The default value of MAXSCM is $40{,}000_{10}$ words, a sufficiently large value to handle the majority of ONEDANT problems. Thus, the user normally need not be concerned with specifying a value of MAXSCM. It should be noted that the value of MAXSCM is the maximum value of SCM storage to be allocated. If the actual problem requires less than MAXSCM, the smaller value will be used.

Through the use of the input parameter, MAXLCM, in Block I of the card-image input, the user can specify the maximum amount of large core memory (LCM) he wishes to use. If unspecified, the value of MAXLCM is defaulted to $140{,}000_{10}$ words.

The modular structure of ONEDANT is such that in the processing of each input Block, as well as the Solver and Edit Modules, each uses LCM storage independently an each such stage requires a different amount of LCM. (In most cases, the cross-section processing stage and the Solver Module require the greatest amount of LCM.) At each stage, the amount of LCM required for that stage is computed with random disk usage, if necessary, such that the stage requires no more than MAXLCM words of LCM. The computing system is instructed to reserve only the actual amount of LCM needed to perform each stage. Thus, if a value of MAXLCM = 100 000 has been input, but a particular stage requires only 30 000 words of LCM, the computing system will reserve only the 30 000 words until that stage is completed. The code will then re-specify the 100 000 allocation and proceed to the next stage. In the Los Alamos time-sharing computing environment, this ability to expand and contract the LCM requirements to the amount actually needed by each stage can be quite advantageous.

The user must be cautioned against specifying too small a value of MAXLCM since the result may be an excessive use of random disk, the access to which is relatively time-consuming. Also, if the user is computing on, say, a virtual memory machine with no random disk, the value of MAXLCM must be large enough so that the problem can be run without random disk.

D. Execution/File Suppression Flags

Included in the Block I input parameters are several flags which control the execution-flow of the code or interface file creation by the code. These flags, included in the collective arrays named NOEXEC and NOFILES are relatively specialized and are normally of interest only to the more advanced user. Accordingly, details on the use of these parameters are not necessary here but is provided in Ch. XIII.

VII. DETAILS ON GEOMETRY-RELATED INPUT (BLOCK II)

Geometry-related information is passed to the ONEDANT SOLVER and EDIT modules solely by means of GEODST standard interface file.[3] If no GEODST file exists prior to the execution of the code package, the user may instruct the Input Module to create the desired GEODST file by (i) providing Block II input data in the card-image input file, and (ii) setting (or defaulting) the BLOCK I input parameter, NOGEOD, to zero. If, on the other hand, a pre-existing GEODST file is to be used, the user may so instruct the code by either (i) omitting all BLOCK II input from the card-image input file or (ii) setting the BLOCK I input parameter NOGEOD to unity.

In the specification of geometry and space-variable related input, the user must be familiar with the nomenclature used by ONEDANT. The terms *fine mesh*, *coarse mesh*, and *zones* are defined below. The term *region* is not used directly by ONEDANT but is used in the GEODST standard file which ONEDANT will accept as input.

The *fine mesh* is the spatial solution-mesh for the problem, as described in Chap. III. Each *fine mesh*, or *fine mesh interval*, is bounded by an adjacent pair of fine-mesh grid-lines $x_{i-1/2}$ and $x_{i+1/2}$ with $x_{i-1/2} < x_{i+1/2}$. There are IT such *fine mesh intervals*. No material discontinuities may occur within a *fine mesh interval*. The specification of the *fine mesh* is accomplished by means of the *coarse mesh*.

The *coarse mesh* is a spatial superset of the *fine mesh* and is formed by partitioning the spatial domain of the problem into a suitable number of "coarse" intervals. There are IM *coarse mesh intervals* spanning the problem. Each *coarse mesh interval* is bounded by an adjacent pair of coarse-mesh boundaries that are specified in the input either as the XMESH array in BLOCK II or an the XMESH array on a GEODST standard interface file. Each *coarse mesh interval* contains one or more *fine mesh intervals*. The number of *fine mesh intervals* per *coarse mesh interval* is specified by means of either the XINTS array in input BLOCK II or the IFINTS array on a GEODST file. All *fine mesh intervals* within a *coarse mesh interval* have equal widths. No material discontinuities may occur within a *coarse mesh interval*.

The *region* is a spatial superset of *coarse mesh intervals* or, conversely, a spatial subset of *zone*. A *region* contains one or more *coarse mesh intervals* and one or more *regions* comprise a *zone*. No material discontinuities occur within a *region*. The concept of the *region* is used only in conjunction with input from a GEODST standard interface file. For input through BLOCK II card images, the term *region* is treated synonymously with the term *coarse mesh interval*.

The *zone* is a spatial superset of *coarse mesh intervals* and is characterized by a single set of multigroup nuclear properties, i.e., cross sections, so that all *fine mesh intervals* within a *zone* have the same cross sections. A *zone* number is assigned to

each *coarse mesh interval* by either (i) the ZONES array in input BLOCK II, or (ii) the NZNR and MR arrays on a GEODST standard file. In the ZONES array input the zone number, n ($1 \leq n \leq$ NZONE), is determined by the order in which zones are specified in the ASSIGN array input in Block IV (see Ch. V), so that the zone number tells the code which macroscopic cross section set is to be used within that zone. Coarse mesh intervals having the same zone number need not be simply connected.

In the ZONES array, the number 0 (zero) can be used to specify that a coarse mesh interval is a pure void (all cross sections are identically zero). A "0" does not count as a zone in determining the value of NZONE.

VIII. INFORMATION ON CROSS-SECTION LIBRARIES (BLOCK III)

A. Input of the Basic Cross-Section Library

The general procedure for generating the macroscopic cross sections appropriate to each zone in the problem is to begin with a basic library containing multigroup cross-section data for isotopes. This section describes the allowable forms that these libraries can take.

1. ISOTXS and GRUPXS Standard Interface Files. Either of the standard interface files ISOTXS or GRUPXS[3] can be used for providing the basic, multigroup cross sections for isotopes. ISOTXS is an isotope-ordered, binary file while GRUPXS is a group-ordered binary file. A complete description of these standard interface files is found in Ref. 3. By default, the file wide vector chi (fission fractions) will be used unless overridden by the zone dependent CHI in BLOCK V.

If the basic library of isotope cross sections is an ISOTXS file, the user enters LIB= ISOTXS in the BLOCK III input; if the library is a GRUPXS file, the user enters LIB= GRUPXS. If LIB=ISOTXS, the cross sections must reside on a file named ISOTXS which must exist at the time of code execution. If LIB=GRUPXS, the cross sections must reside on a file named GRUPXS which must exist at the time of code execution.

2. Isotopic, Card-Image Libraries in the Los Alamos, ANISN or FIDO Format. The basic multigroup cross sections for isotopes can be provided in a card-image library whose form is referred to as Los Alamos, ANISN, or FIDO. This library form consists of a collection of cross-section tables. Each of these cross-section tables contains the full set of multigroup cross sections for one Legendre scattering order for one isotope. The ordering of cross sections within a cross-section table, the ordering of cross-section tables to form the library, and other details and user options are described below.

The user specifies that the library of cross sections is to be such a card-image library by entering either LIB=ODNINP or LIB=XSLIB or LIB=*filename* where *filename* is any name that the user chooses other than any of the following: ISOTXS, GRUPXS, ODNINP, MACRXS, MACBCD, BXSLIB, XSLIBB, XSLIBE, XSLIBF, MENDF, or MENDFG. If LIB= ODNINP, the library card-images are physically located within the input for the ONEDANT code between the input for BLOCK III and the input for BLOCK IV. If LIB= XSLIB (or filename), the library card-images are physically located on a file named XSLIB (or filename), which must exist at the time of code execution.

a. Ordering of Cross Sections Within a Cross-Section Table. The Los Alamos, ANISN or FIDO card-image library form assumes that each cross-section table in the library contains an array of cross sections of IHM rows for each of NGROUP group columns. The cross-section type for each group is determined by its row position as shown in Table XI. Row positions are specified relative to the positions of the total cross section σ_t (row IHT) and the within-group scattering cross section $\sigma_s(g \rightarrow g)$, (row IHS). Note that the values of IHM, IHT, and IHS are input values in BLOCK III.

Each cross-section table contains the cross sections for one Legendre scattering order for one isotope as IHM*NGROUP data entries. A cross-section table begins on a new card-image and the data are entered continuously beginning with IHM entries for group 1, followed by IHM entries for group 2, etc.

b. Card-Image Data Formats. The cross-section data may be entered on the card-images in one of three data formats, the traditional Los Alamos format, the fixed-field FIDO format, or the free-field FIDO format. The user selects the desired format through the IFIDO input parameter in the BLOCK III input.

In the traditional Los Alamos format (IFIDO= 0), also called the DTF format, the data are entered on the card-images in 6E12 format.

In the fixed-field FIDO format (IFIDO= 1), sometimes called the ANISN format, the data are entered on the card-images using the fixed-field FIDO format described in Ch. IV. *When this format is used, each cross-section table must be terminated with the "T" terminator described in Ch. IV.*

In the free-field FIDO format (IFIDO= 2), the data are entered on the card-images in free-field format as described in Ch. IV. *When this format is used, each cross-section table must be terminated with the "T" terminator described in Ch. IV.*

NOTE: For free-or fixed-field FIDO cross sections, neither an array name (or number) nor an array identifier is needed with the cross-section data.

c. Cross-Section Table Title Cards. A single title card may optionally be attached to the front of each cross-section table, if desired. This option is controlled by the input parameter, ITITL in the BLOCK III input.

d. Anisotropic Scattering and the Ordering of Cross-Section Tables. In the ONEDANT code package, it is assumed that the scattering transfer probability can be represented by the finite Legendre polynomial expansion of Eq. (2), which, in multigroup notation, becomes

$$\sigma_s\left(g' \rightarrow g, \mu_o\right) = \sum_{n=0}^{ISCT} \frac{2n+1}{4\pi} P_n\left(\mu_o\right) \sigma_s^n\left(g' \rightarrow g\right) \quad , \tag{37}$$

TABLE XI
CROSS SECTION ORDERING IN CARD-IMAGE LIBRARY

Number of Rows per Group	Row	Cross Section Type, Group g	
	.	.	Edit Positions, (Optional)
	.	.	
	.	.	
	IHT–2	σ_a	Principal Cross Sections
	IHT–1	$\nu\sigma_f$	
	IHT[a]	σ_t	
	.	.	Upscatter Cross Sections
	.	.	
	.	.	
	IHS–2	$\sigma_s(g+2 \rightarrow g)$	
	IHS–1	$\sigma_s(g+1 \rightarrow g)$	
	IHS[a]	$\sigma_s(g \rightarrow g)$	
	IHS+1	$\sigma_s(g-1 \rightarrow g)$	Downscatter Cross Sections
	IHS+2	$\sigma_s(g-2 \rightarrow g)$	
	.	.	
	.	.	
	.	.	

IHM[a]

[a]Input in Block III

where $\mu_o \equiv \underline{\Omega}' \cdot \underline{\Omega}$, the scattering angle and ISCT is the desired Legendre order of anisotropy in the transport calculation (input in BLOCK V). If ISCT>0, additional tables of cross sections must be supplied in order to provide the higher order scattering cross sections $\sigma_s^n(g' \rightarrow g)$ needed for the Legendre expansion.

When using the card image library, the first cross-section table for an isotope contains the P_o, or isotropic, multigroup cross sections ordered as shown in Table XI. The next cross-section table provides the P_1 multigroup cross sections with the same ordering; the next table contains the P_2 multigroup cross sections, etc., all for the same isotope. It should be noted that, for high Legendre order cross-section tables, only the scattering cross sections are used. The first IHT rows for each group are ignored in the $P_L(L > 0)$ tables and the data values in these positions are usually input as 0.0. The number of tables per isotope can vary with each isotope. The number of cross-section tables per isotope is provided in the input array NTPI in BLOCK III. If the NTPI array is not provided, the code will assume that the card-image library contains MAXORD + 1 cross-section tables for each isotope, where MAXORD is an input parameter in BLOCK III.

Note that the library may contain scattering data for up to a MAXORD order of anisotropy, but the actual transport calculation can be performed assuming an ISCT order of anisotropy so long as ISCT $\leq$ MAXORD.

3. Binary Form of Card-Image Libraries (the BXSLIB file). The processing of large, card-image, ASCII libraries can be relatively time-consuming, especially if the library is in FIDO format. The binary form of the card-image library can be processed much more rapidly. By entering LIB= BXSLIB, the user can instruct the code to use the binary form of the card-image, isotopic library (the binary file named BXSLIB) as the input for the basic cross-section data.

Use of LIB= BXSLIB requires that the appropriate binary form of the library exists and is available to the code at the time of execution. To create the BXSLIB file, the user makes his initial execution with the card-image library (LIB= XSLIB or LIB= ODNINP) as previously described. Then, by setting the input parameter SAVBXS= 1 in the BLOCK III input, the user can instruct the code to create the binary BXSLIB file and to retain this file after execution of the Input Module. For Los Alamos users, if LIB= MENDF, (see Sec. A. 7 below) a BXSLIB file is always created and retained. The user can then save this BXSLIB binary file and use it for subsequent runs in place of the BCD library. It should be noted that in addition to the actual cross-section data, the BXSLIB file will contain any and all other information specified in the CARDS, NAMES, EDNAME, NTPI, CHIVEC, VEL, EBOUND, and ATWT arrays of BLOCK III as provided in the originating LIB= ODNINP or LIB= XSLIB run. The file description for the BXSLIB binary file is provided in App. A.

4. XSLIBB Card-Image Library File. By entering LIB=XSLIBB in the Block III input, the user instructs the code to use the specialized isotopic cross-section file named XSLIBB. XSLIBB is an ASCII, card-image version of the BXSLIB file described previously. The principal advantage of an XSLIBB file is that it is an ASCII file and is thus both eye-readable and exportable. Since it can also contain other information such as NAMES, EDNAME, NTPI, CHIVEC, VEL, ATWT, and EBOUND, it provides a very useful form of a cross-section library. Use of LIB=XSLIBB requires that the appropriate form of the card-image file exists with the name XSLIBB and is available to the code at the time of execution.

The creation of an XSLIBB file is controlled by the WRITMXS parameter in Block III as described in Sec. C. of this chapter.

5. MACRXS and SNXEDT Cross-Section Files. By entering LIB= MACRXS in the BLOCK III input, the user can instruct the code to use the code-dependent interface files MACRXS and SNEXDT together with the standard interface files NDXSRF and ZNATDN[3] without referring to a basic library of multigroup isotope cross sections. These four files contain cross sections and other information pertaining to the materials created from the original isotopes. (A more detailed discussion of the MACRXS and SNXEDT file preparation process is provided in Ch. IX and file descriptions for MACRXS and SNXEDT are given in App. A.) This procedure circumvents the sometimes time-consuming process of re-creating these files when a series of code calculations are being made on the same basic problem.

If the user enters LIB= MACRXS, it is understood that the MACRXS, SNXEDT, NDXSRF, and ZNATDN files must have been previously created and saved and, further, that these files must be available to ONEDANT at the time of execution as follows:

(i) MACRXS is required if the SOLVER module is to be executed, and

(ii) SNXEDT, NDXSRF, and ZNATDN are required if the EDIT module is to be executed.

6. The MACBCD Card-Image Cross-Section Library. By entering LIB=MACBCD in the Block III input, the user instructs the code to use the specialized material cross-section file named MACBCD. MACBCD is an ASCII, card-image version of the MACRXS described in the preceding section. Use of LIB=MACBCD requires that the appropriate card-image file exists with the name MACBCD and that it is available to the code at the time of execution.

The creation of a MACBCD file is controlled by the WRITMXS parameter in Block III as described later in the chapter.

7. The Los Alamos MENDF5 Cross-Section Library. At Los Alamos National Laboratory a multigroup, isotopic cross-section library named MENDF5 is maintained as a random access public file available to all users of the Laboratory's mainframe computers. This file is derived from ENDF-V/B nuclear data evaluations. To use this binary library, or a library constructed in the same format

as MENDF5, the user enters LIB=MENDF in the Block III input. The code will seek a file named MENDF in the user's local file space and, if such a file exists, it will use it. If a file named MENDF does not exist in the local file space, the code will extract the MENDF5 public file and will use it.

The appropriate fission fractions may be specified using the NTICHI input parameter in Block III, or by using the CHIVEC input array of Block III, or by using zone-dependent chi's in the Block V input (described in Ch. X).

When using a MENDF file, isotopes are identified by a floating point number called a ZAID. The ZAID is of the form ZZAAA.NN where ZZ is the atomic number, AAA is the (three-digit) mass number, and NN is a two-digit number specifying a particular version of the cross sections for each given isotope. A listing of the current ZAIDs available on MENDF5 can be obtained from Group X-6 at Los Alamos National Laboratory.

8. The Los Alamos MENDF5G Gamma Cross-Section Library. Similar to the MENDF5 library at Los Alamos described in the preceding section, a multigroup, isotopic companion gamma-ray (photon) library is maintained. MENDF5G contains only neutron-induced photon-production and photon-interaction data. To access this library for gamma (photon)-only calculations, the user specifies LIB=MENDFG in Block III of the input and the LNG (Last Neutron Group) parameter of Block III must be set to zero.

Isotopes are referred to by their ZAIDs as described in the preceding section.

9. The XSLIBE and XSLIBF Material Cross-Section Libraries. The code can use card-image material cross-section libraries named XSLIBE and XSLIBF by setting LIB=XSLIBE or XSLIBF in Block III of the input. These two files are ASCII, card-image files containing the material macroscopic cross sections taken from the MACRXS binary file. The format of these card-image files is the Los Alamos or ANISN format described in Section A.2 of this chapter. XSLIBE is formatted in the Los Alamos 6E12 card-image format while XSLIBF is formatted in the fixed-field FIDO format. These files can be created by the code using the WRITMXS parameter in Block III as described later in this chapter.

B. Coupled Neutron-Gamma Cross-Section Sets

The ONEDANT code package can solve coupled neutron-gamma problems in which neutron interactions with matter produce a source of gamma rays (photons). The simultaneous solution of the neutron-gamma transport problem can be effected by simply providing a coupled neutron-gamma cross-section library or set. In such a coupled set the gamma energy groups are treated as if they were the lowest energy neutron groups. For example, a 42-group coupled set (NGROUP = 42) might have 30 neutron groups (LNG = 30) followed by 12 gamma groups. Such coupled sets can be provided in the form of card-image libraries with no upscatter, so that IHS = IHT+1 and IHM = IHT+NGROUP. In this form neutrons appear to "downscatter" into the gamma-ray groups as a result of gamma production resulting from neutron interactions but gamma-rays do not "upscatter" into neutron groups, i.e., neutron

production via photoneutron, or γ-n, reactions is not allowed. Using the card-image form of a coupled library with cross sections ordered as shown in Table XII (for no upscatter), the isotopic cross sections for each Legendre order of scatter carry data arranged as shown in Table XIII. Table XIII shows the contents of a card-image cross-section table for a 7-group coupled set (4 neutron, 3 gamma groups).

C. Creating Cross-Section Files with Different Formats (The WRITMXS Parameter)

It is frequently desirable to produce card-image, ASCII forms of cross section libraries that are eye-readable and exportable. The Input Module provides the user with the capability to produce several card-image library forms through the use of the WRITMXS parameter in Block III of the input.

By setting WRITMXS=XSLIBB, the code is instructed to create the ASCII file XSLIBB described in Section A.4. It can be used if the original cross-section library (as specified in the LIB= instruction) is ODNINP, XSLIB, MENDF, MENDFG, or BXSLIB.

By entering WRITMXS=MACBCD, the code will create the ASCII file MACBCD described in Section A.6. The MACBCD file can be created no matter what the form of the original cross-section library.

By entering WRITMXS=XSLIBE, the code will create the ASCII file XSLIBE containing the material cross sections in Los Alamos 6E12 format as described in Sections A.2 and A.9. Similarly, WRITMXS=XSLIBF instructs the code to create the ASCII file XSLIBF containing the material cross sections in fixed-field FIDO format as described in Sections A.2 and A.9. (NOTE: WRITMXS=XSLIBF is only available in the local, Los Alamos version of the code.) Either file can be created irrespective of the form of the original cross section library (as specified in the LIB= instruction).

TABLE XII
ARRANGEMENT OF DATA IN A COUPLED NEUTRON-GAMMA LIBRARY TABLE

ROW	GROUP 1 2 . . . LNG	LNG+1 . NGROUP
1 2 . . IHT	PRINCIPAL CROSS SECTIONS AND EDIT POSITIONS	
IHS . .	NEUTRON PRODUCTION VIA NEUTRON SCATTER	GAMMA PRODUCTION VIA GAMMA SCATTER
IHS+LNG-1	NOT USED	GAMMA PRODUCTION VIA NEUTRON INTERACTIONS
IHS+LNG . . IHM= IHT+NGROUP		

TABLE XIII
EXAMPLE CROSS-SECTION TABLE FOR A 7-GROUP COUPLED SET WITH 4 NEUTRON, 3 GAMMA GROUPS

NGROUP = 7
LNG = 4
IHT = 3
IHS = 4
IHM = 10

	NEUTRON GROUPS				GAMMA GROUPS		
	1	2	3	4	5	6	7
1	$\sigma_a(1)$	$\sigma_a(2)$	$\sigma_a(3)$	$\sigma_a(4)$	$\sigma_a(5)$	$\sigma_a(6)$	$\sigma_a(7)$
2	$\nu\sigma_f(1)$	$\nu\sigma_f(2)$	$\nu\sigma_f(3)$	$\nu\sigma_f(4)$	0	0	0
3	$\sigma_t(1)$	$\sigma_t(2)$	$\sigma_t(3)$	$\sigma_t(4)$	$\sigma_t(5)$	$\sigma_t(6)$	$\sigma_t(7)$
4	$\sigma(1 \rightarrow 1)$	$\sigma(2 \rightarrow 2)$	$\sigma(3 \rightarrow 3)$	$\sigma(4 \rightarrow 4)$	$\sigma(5 \rightarrow 5)$	$\sigma(6 \rightarrow 6)$	$\sigma(7 \rightarrow 7)$
5	0	$\sigma(1 \rightarrow 2)$	$\sigma(2 \rightarrow 3)$	$\sigma(3 \rightarrow 4)$	$\sigma(4 \rightarrow 5)$	$\sigma(5 \rightarrow 6)$	$\sigma(6 \rightarrow 7)$
6	0	0	$\sigma(1 \rightarrow 3)$	$\sigma(2 \rightarrow 4)$	$\sigma(3 \rightarrow 5)$	$\sigma(4 \rightarrow 6)$	$\sigma(5 \rightarrow 7)$
7	0	0	0	$\sigma(1 \rightarrow 4)$	$\sigma(2 \rightarrow 5)$	$\sigma(3 \rightarrow 6)$	$\sigma(4 \rightarrow 7)$
8	0	0	0	0	$\sigma(1 \rightarrow 5)$	$\sigma(2 \rightarrow 6)$	$\sigma(3 \rightarrow 7)$
9	0	0	0	0	0	$\sigma(1 \rightarrow 6)$	$\sigma(2 \rightarrow 7)$
10	0	0	0	0	0	0	$\sigma(1 \rightarrow 7)$

IX. FURTHER DETAILS ON MATERIAL MIXING AND ASSIGNMENT-OF-MATERIALS-TO-ZONES (BLOCK IV)

A. Review of Terminology

An understanding of the general procedure for mixing nuclides in ONEDANT requires an understanding of the terminology used with the code. A review of five basic terms is provided below.

1. **Fine Mesh** - the Fine Mesh is the spatial solution mesh for the problem.

2. **Coarse Mesh** - the Coarse Mesh is a spatial superset of the Fine Mesh. Each Coarse Mesh contains one or more Fine Mesh intervals. Fine Mesh widths are all the same within a given Coarse Mesh interval. No material discontinuities may occur within a Coarse Mesh interval.

3. **Zone** - the Zone is a spatial superset of Coarse Mesh intervals. Each Zone contains one or more Coarse Mesh intervals. A Zone is characterized by a single set of macroscopic multigroup cross sections so that all Fine Mesh intervals within a Zone have the same cross sections.

4. **Material** - a Material is composed of isotopes (or nuclides or mixes) whose "microscopic" cross sections exist on the cross-section library file to be used. The specification of materials requires knowing which isotopes and how much of each isotope goes into the Material. This information is provided the code through the MATLS= array in BLOCK IV of the ONEDANT/TWODANT input.

5. **Material Assignments to Zones** - the material composition of a Zone is made by assigning one or more Materials in specified amounts to each Zone through the ASSIGN= input array in BLOCK IV of the ONEDANT/TWODANT input. This assignment thus links a *spatial* portion of the physical problem model (the Zone) to the *macroscopic cross sections* that are to be used in that spatially-defined Zone.

B. The Basic Method for Creating Materials as Mixtures of Nuclides

If "isotopic" cross sections are provided on a cross-section library, it is necessary to mix these isotopes, or nuclides, to create materials. The mixing instructions are provided either by (i) card-image input in BLOCK IV by means of the MATLS array and, optionally, the PREMIX array, whose specifications are described in Ch. V, or by (ii) the standard interface files NDXSRF and ZNATDN.[3,4] If the

NDXSRF and ZNATDN files are used, the term "zone" in the file descriptions of Refs. 3 and 4 must be replaced with the word "material" to be consistent with ONEDANT terminology.

In this section we will provide a basic description of how to mix isotopes (or nuclides) to form materials using the MATLS array specification in BLOCK IV of the input. For now we will assume that it is desired to create materials by specifying the isotopes and their atom densities in each material. Later in this chapter will be described some alternatives to using atom densities.

Let us consider an example of a common way of specifying materials. Suppose we have a multigroup library of cross sections (in barns) for several nuclides including those whose identifiers are U235, U238, PU239, PU240, IRON, CHROME, NICKEL, OXYGEN, and SODIUM. It is desired to mix these nuclides into the following materials:

(i) Stainless Steel (SS) at 8 gm/cm^3 [74% Iron, 18% Chromium, and 8% Nickel (percents are weight percents)],

(ii) Uranium Oxide (U02) at 10 gm/cm^3 [0.3% U235 and 99.7% U238 (percents are atomic percents)],

(iii) Plutonium Oxide (PU02) at 10 gm/cm^3 [80% PU239 and 20% PU240 (percents are atomic percents)],

(iv) Sodium (NA) at 0.8 gm/cm^3.

Translating the above information into atom densities (atoms/barn-cm^2) gives the values shown in Table XIV.

TABLE XIV
Example Specification of Materials

Material Identifier	Isotope Identifier	Atom Density
SS	CHROME	1.6676E-2
"	NICKEL	6.566E-3
"	IRON	6.3832E-2
UO2	U235	6.7E-5
"	U238	2.2234E-2
"	OXYGEN	4.4602E-2
PUO2	PU239	1.7761E-2
"	PU240	4.440E-3
"	OXYGEN	4.4402E-2
NA	SODIUM	2.096E-2

To input this information to ONEDANT, we enter in BLOCK IV the following

```
MATLS= SS, CHROME 1.6676-2, NICKEL 6.566-3, IRON 6.3832-2; U02, U235
       6.7-5, U238 2.2234-2, OXYGEN 4.4602-2; PU02, PU239 1.7761-2, PU240
       4.44-3, OXYGEN 4.4402-2; NA, SODIUM 2.096-2
```

Note the use of semicolons (;) to delimit the different materials (see Ch. V). Also note that the use of commas is optional. Blanks also serve as delimiters (see Ch. IV).

C. Assignment of Materials to Zones

The macroscopic cross sections for the zones in the physical problem being analyzed are created from the material cross sections by assigning materials to zones with appropriate material concentrations, volume fractions, or densities, as desired. This assignment is accomplished either by means of the ASSIGN array card-image input in BLOCK IV or by means of a pre-existing code-dependent binary interface file ASGMAT.

As an example of the material assignments to zones, suppose the following materials have been created: Stainless Steel (SS), Coolant (NA), U-238 Oxide (U802), U-235 Oxide (U502), and PU-239 Oxide (PU902). It is desired to assign these three materials to create the correct macroscopic zone sections for the three zones named CORE, BLKT, and REFL whose compositions are as follows:

Zone	Material	Material Volume Fraction
CORE	SS	0.25
CORE	NA	0.40
CORE	U802	0.20
CORE	PU902	0.15
BLKT	SS	0.25
BLKT	NA	0.40
BLKT	U802	0.349
BLKT	U502	0.001
REFL	SS	0.30
REFL	NA	0.70

The above specifications can be provided via the ASSIGN array of BLOCK IV of the input by entering the card-image input:

ASSIGN=	CORE	SS	0.25	NA	0.40	U802	0.20	PU902	0.15;
	BLKT	SS	0.25	NA	0.40	U802	0.349	U502	0.001;
	REFL	SS	0.3	NA	0.7				

The card-image input for the assignment-of-materials-to-zones is written to a code-dependent, binary interface file named ASGMAT for use by both the Solver and Edit Modules. The file description for ASGMAT is given in Appendix A.

If it is desired to use a previously created ASGMAT file for specification of the assignment-of-materials-to-zones, the user should

(i) omit the ASSIGN array specifications in the BLOCK IV card-image input or, alternatively, set the BLOCK I input parameter NOASG to unity, and

(ii) ensure that the binary ASGMAT file exists and is available to ONEDANT at the time of code execution.

D. Alternative Forms of Mixing (the MATSPEC and ATWT Arrays)

Historically, ONEDANT and TWODANT have been oriented toward specifying materials by requiring input that gives the atom densities of isotopes comprising the material. This is typical of reactor-oriented computer codes. There is a nuclear analysis community, however, that specifies mixes by identifying materials by the density of the material together with the isotopes and their *atomic* or *weight fractions* in that material.

ONEDANT and TWODANT will accept the latter type of material specification in addition to the traditional atom density type of specification. A description follows.

1. Mixing of Materials Using Atomic Fractions or Weight Fractions of Isotopes (the MATSPEC Parameter)

In BLOCK IV of the ONEDANT/TWODANT input (the mixing specifications) is the parameter MATSPEC. There are three allowable input values for MATSPEC:

MATSPEC= ATDENS tells the code you are using the traditional atom density style of input for mixing specifications as described in Sec. B of this chapter. This is the DEFAULT.

MATSPEC= ATFRAC tells the code you are using the type of mixing that specifies the density (gm/cc) of each material together with the isotopes (nuclides) and their *atomic fractions* in each material.

MATSPEC= WTFRAC tells the code you are using the type of mixing that specifies the density (gm/cc) of each material together with the isotopes (nuclides) and their *weight fractions* in each material.

The MATSPEC parameter can be entered as a vector parameter with up to MT entries so that different materials can be specified with different types of mixing specifications. (Recall that MT is the number of materials to be created, as specified in BLOCK I of the input.) If the number of entries is less than MT, the last entry will be applied to all remaining and unspecified entries.

Following is a description of how to input information in each of the above styles.

ASIDE: In the following

MAT_m denotes the name (or identifier) of *material* m. It is usually a hollerith name, e.g., fuel, Tu, steel, etc., for ease of reading.

$ISO_{i,m}$ denotes the name (or identifier) of *isotope* i that exists in *material* m. $ISO_{i,m}$ is usually the ZAID identifier if one is using a MENDF cross section library.

$AF_{i,m}$ denotes the *atomic fraction* of isotope i that exists in material m.

$WF_{i,m}$ denotes the *weight fraction* of isotope i that exists in material m.

$ZONID_j$ denotes the name (or identifier) of *Zone* j. It is usually a hollerith name, e.g., core, shell, driver, etc., for ease of reading and identifying.

ρ_m denotes the *density* [gm/cc] of material m as it exists in a Zone.

END ASIDE

- MATSPEC= ATDENS
  If one makes the entry MATSPEC=ATDENS or one omits the MATSPEC= entry, the mixing is done exactly as described in Sec. B. of this chapter.

- MATSPEC= ATFRAC
  (Step 1) One defines Materials, *each with a density of 1 gm/cc,* by specifying (using the MATLS= input array of BLOCK IV) the name (or identifier) of the Material followed by the Isotope names (or identifiers) paired with the Atomic Fractions of those isotopes which define the Material. This is done by inputting

  MATLS= MAT_1 $ISO_{1,1}$ $AF_{1,1}$ $ISO_{2,1}$ $AF_{2,1}$ ...; MAT_2 $ISO_{1,2}$ $AF_{1,2}$ $ISO_{2,2}$ $AF_{2,2}$ ...; etc.

  NOTE: *Each Material's specification must be separated from the next by a semi-colon!*

  (Step 2) Then one specifies how the above materials will be mixed and at what density into each Zone. This is done by inputting

  ASSIGN=$ZONID_1$ MAT_i ρ_i ... ; $ZONID_2$ MAT_k ρ_k MAT_ℓ ρ_ℓ ... ; etc.

NOTE: *Each Zone's specification must be separated from the next by a semicolon!*

- MATSPEC= WTFRAC
  (Step 1) One defines Materials, *each with a density of 1 gm/cc,* by specifying (using the MATLS= input array of BLOCK IV) the name (or identifier) of the Material followed by the Isotope ;names (or identifiers) paired with the Weight Fractions of those isotopes which define the Material. This is done by inputting

  MATLS= MAT_1 $ISO_{1,1}$ $WF_{1,1}$ $ISO_{2,1}$ $WF_{2,1}$ ... ; MAT_2 $ISO_{1,2}$ $WF_{1,2}$ $ISO_{2,2}$ $WF_{2,2}$ ... ; etc.

  NOTE: *Each Material's specification must be separated from the next by a semicolon!*

  (Step 2) Then one specifies how the above materials will be mixed and at what density into each Zone. This is done by inputting

  ASSIGN=$ZONID_1$ MAT_i ρ_i ... ; $ZONID_2$ MAT_k ρ_k MAT_ℓ ρ_ℓ ... ; etc.

  NOTE: *Each Zone's specification must be separated from the next by a semicolon!*

2. Providing Atomic Weights to the Code (the ATWT Array)

When using MATSPEC=ATFRAC or MATSPEC=WTFRAC the code must have the atomic weights of the isotopes.

If the cross-section library is *not* a MENDF, MENDFG, or a BXSLIB file created with atomic weights on it, the atomic weights must be supplied in the BLOCK IV input. This is done as follows:

In BLOCK IV enter

ATWT=ISO_1 $ATWT_1$ ISO_2 $ATWT_2$ ... ISO_N $ATWT_N$

where N $\leq$ NISO,

ISO_i is the isotope name (identifier) for isotope i on the cross section library, and $ATWT_i$ is the atomic weight for isotope i.

If using LIB=MENDF for neutrons or LIB=MENDFG for gammas in BLOCK III of the input, the atomic weights are automatically provided and nothing more need be done by the user.

NOTE: Whenever the atomic weights are provided, either in the card-image input or from a MENDF or MENDFG file, the BXSLIB file that the code creates will contain the atomic weight data automatically. By saving this BXSLIB file (see the SAVBXS parameter in BLOCK III) and using it as the cross-section library file in subsequent runs, there will be no need to re-enter the atomic weights in the card-image input.

E. The Creation/Use of Interface Files in Mixing and Assigning Materials

1. Material Mixing and the Creation of Interface Files. In the material mixing operation in the Input Module of ONEDANT, the following four binary interface files are produced: MACRXS, SNXEDT, NDXSRF, and ZNATDN. These, and only these, files are used by subsequent portions of the code; the basic isotope cross-section library is "forgotten" once these four files are created.

The MACRXS code-dependent binary interface file is described in Appendix A and contains material cross sections in energy-group order. The MACRXS file is the only cross-section file available to the Solver Module. If a large isotope-ordered, basic cross-section library is used, the mixing and group-ordering process used in creating the MACRXS file can be quite time-consuming. If several calculations are to be performed, for example, parametric studies, on a particular nuclear system, it is advantageous to create a basic MACRXS material file one time only and save this file for use in subsequent runs involving the Solver Module. By use of the assignment-of-materials-to-zones specification, described in BLOCK IV of Ch. V and in Section C of this chapter, a single set of materials, that is, a single MACRXS file can be used for calculating numerous different problems in which the problem zone compositions consist of different proportions of materials. The manner in which the code is instructed to use an existing MACRXS file is described in Sec. E.2 below.

The SNXEDT code-dependent, binary interface file produced by the Input Module contains group-ordered cross-section data for use by the Edit Module. Contained in the file are the principal cross sections and edit position data for all isotopes on the basic input cross-section library. Scattering, or transfer, matrices are not included on the SNXEDT file. This file is used directly by the Edit Module for providing microscopic and constituent edits described in Chapter XI. The SNXEDT file description is given in Appendix A.

The NDXSRF and ZNATDN standard interface files are used by the Edit Module together with the SNXEDT file to mix the isotopes into the materials used by the Solver Module. The Edit Module uses these materials in providing the macroscopic (or material) edits described in Chapter XI. It is again noted that in using the NDXSRF and ZNATDN files, the term "zone" in the file descriptions of Refs. 3 and 4 must be replaced with the word "MATERIAL" to be consistent with ONEDANT terminology.

As with the MACRXS file discussed above, it is frequently advantageous to save the SNXEDT, NDXSRF, and ZNATDN files created in one run for use in subsequent runs, if possible. This procedure eliminates the need to continually repeat the often time-consuming process of re-creating the group-ordered code-dependent SNXEDT file. Parametric studies on variations of material compositions in the zones of the physical problem can be accomplished simply by changing the assignment-of-materials-to-zones specifications described in BLOCK IV of Ch. V and in Sec. C of this chapter.

The manner in which the code is instructed to use existing SNXEDT, NDXSRF, and ZNATDN files is described below. It should be noted that the use of an

SNXEDT file by the Edit Module is ually accompanied by the use of the associated NDXSRF and ZNATDN files, id it is wise to treat these three files as a single triumvirate.

2. Using Existing MACRXS, SNXEDT, NDXSRF, ZNATDN Interface Files. If an existing pair of NDXSRF and ZNATDN standard interface files is to be used to specify the material mixing instructions in conjunction with a basic isotope cross-section library, the user should

(i) omit the specification of the MATLS array in BLOCK IV card-image input or, alternatively, set the BLOCK I input parameter NOMIX to unity, and

(ii) ensure that the NDXSRF and ZNATDN binary files exist and are available to ONEDANT at the time of execution.

If an existing quartet of MACRXS, SNXEDT, NDXSRF, and ZNATDN binary interface files is to be used, the user should

(i) omit BLOCK II and the MATLS array in BLOCK IV in the card-image input or, alternatively, set LIB=MACRXS in the BLOCK III input or, alternatively, set the BLOCK I input parameters NOMIX and NOMACR both in unity, and

(ii) ensure that the MACRXS, SNXEDT, NDXSRF, and ZNATDN binary files exist and are available to ONEDANT at the time of execution. Note: only the MACRXS file is needed for execution of the Solver Module, and only the SNXEDT, NDXSRF, and ZNATDN files are needed for execution of the Edit Module.

X. DETAILS RELATED TO ONEDANT SOLVER MODULE INPUT AND EXECUTION (BLOCK V)

A. Iteration Strategy

As described in Chapter III.B of this report, the ONEDANT Solver Module employs the diffusion synthetic method to accelerate the iterative procedure used in solving the transport equation. In this section is described the iteration strategy used in the execution of the Solver Module and reflected in the iteration monitor printout supplied as printed output.

The basic features of the iteration strategy are shown in the simplified flow diagram of Fig. 7. As indicated, there are two different iterative procedures, one for problems containing fissionable material and/or energy-group upscattering and one for problems with neither fissions nor upscattering.

The iterative strategy is divided into two parts: inner iterations and outer iterations. The inner iterations are concerned with the convergence of the pointwise scalar fluxes in each group for a given source distribution. The outer iterations are concerned with the convergence of the eigenvalue, the fission source distribution and the energy-group upscatter source if any or all are present.

For problems containing fissionable material, the iterative procedure begins with the calculation of a diffusion coefficient for each space-energy point using

$$D(x,g) = \begin{cases} 1/\left[3\sum_{trans}(x,g)\right], & \text{If } \sum_{trans} \text{ (x,g) is available.} \\ & \text{If not,} \\ 1/\left[3\sum_{t}(x,g)\right], & \text{Isotropic scatter} \\ 1/\left\{3\left[\sum_{t}(x,g) - \sum_{s1}(x,g\rightarrow g)\right]\right\}, & \text{Anisotropic scatter} \end{cases} \tag{38}$$

where D(x,g) is the diffusion coefficient at position x for energy group g, $\sum_t(x,g)$ is the macroscopic total cross section at the space energy point in question, $\sum_{trans}(x,g)$ is the macroscopic transport cross section (transport is normally provided on the ISOTXS or GRUPXS files), and $\sum_{sl}(x,g\rightarrow g)$ is the P_1 anisotropic self-scatter cross section. It should be noted that $\sum_t(x,g)$ is formed from the isotope cross sections contained in the total cross-section position in the cross-section library. The data provided in this position may, in fact, contain the transport cross section in transport-corrected cross-section libraries for isotropic scatter.

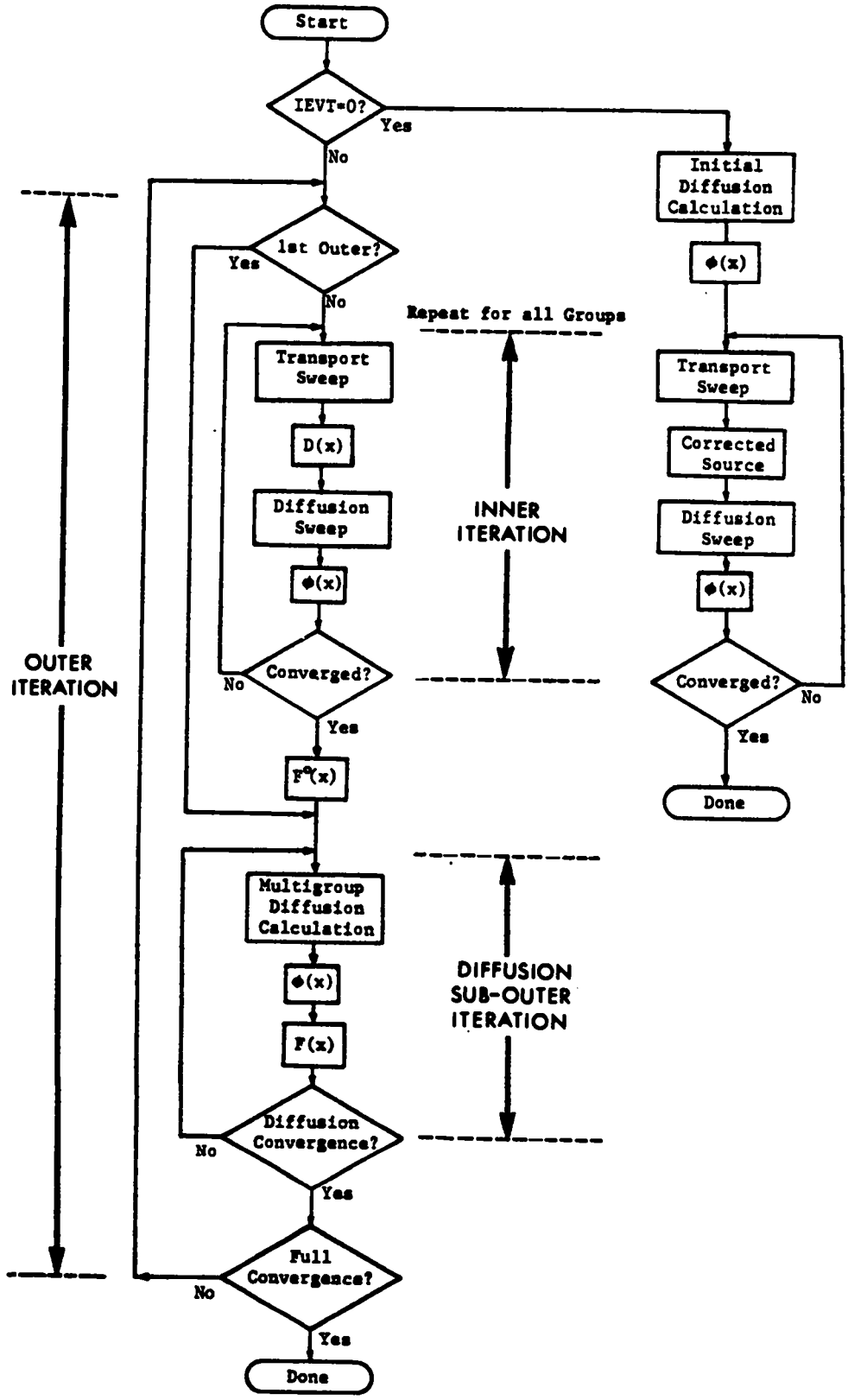


Fig. 7. Simplified flow diagram of SOLVER iteration strategy.

Using the above diffusion coefficients, a standard diffusion calculation is performed. Beginning with a fiat fission source guess, the diffusion fluxes are calculated for each energy group. With the fiuxes for all groups, a new fission source rate distribution, F(x), is calculated. This new F(x) is then used to generate new diffusion fiuxes. The process is repeated until both F(x) and the pointwise fiuxes are converged. Each such recalculation of F(x) is called a DIFFUSION SUB-OUTER ITERATION.

Next, using the diffusion-converged F(x) and using the first energy-group diffusion scalar fiuxes to fix the within-group scattering sources, a single discrete-ordinates transport sweep through the spatial mesh is made for the first energy group. In this sweep angular fiuxes are generated. Using Fick's Law, Eq. (32), these angular fiuxes are used to calculate an effective diffusion coefficient, $\underline{\underline{D}}(x)$, at each mesh point. With these effective diffusion coefficients, a diffusion sweep for the group is performed to determine the group scalar fiux, $\phi(x)$, at each point. This transport sweep, followed by a calculation of $\underline{\underline{D}}(x)$, followed by a diffusion sweep is called an INNER ITERATION. Since the new diffusion-calculated group scalar flux, $\phi(x)$, changes the within-group scattering source terms, the user may opt to perform another (or several more) inner iterations before proceeding to the next energy group. The IITL and IITM input parameters in BLOCK V of the input control the maximum number of inner iterations to be performed for each group. Normally, however, the user need not enter these parameters since the code will use default values that have been found to be reliable.

When the inner iterations for the first energy group are completed, the group scalar fiuxes and fiux moments are used to calculate the scattering source for the next group. One or more inner iterations are performed for the next group and the process is repeated until all energy groups have been completed.

When all energy groups have been calculated via inner iterations, the group fiuxes are used to calculate a new fission source rate distribution, F°(x). Following this, a series of diffusion sub-outer iterations is performed. In these diffusion sub-outer iterations, however, the effective diffusion coefficients from the last-completed inner iteration for each group are used, thus making the sub-outer iteration calculation a synthetic diffusion calculation.

Each completion of the diffusion sub-outer iteration process based on the current set of diffusion coefficients defines the completion of an OUTER ITERATION (see Fig. 7). By the initial definition of D(x), Eq. (38), the first outer iteration is seen to be a pure diffusion calculation, while all subsequent outer iterations are synthetic diffusion in nature. Outer iterations continue until convergence (as described in Sec. B of this chapter) is achieved. The above procedure, of course, employs the diffusion coefficient correction scheme described in III.B.

For problems containing an inhomogeneous source (either distributed in space or localized as a surface source, or both), the iteration solution strategy may take one of two forms. The form used is controlled by the input parameter IEVT in BLOCK V of the input. When IEVT= -1, the iterative procedure used is the same as described above. IEVT= -1 should be used if the inhomogeneous source problem contains either fissionable material or material capable of producing energy-group upscattering of particles. If the inhomogeneous source problem contains neither

fission nor upscattering, a value of IEVT= 0 should be selected. In this case the source correction scheme is used in the diffusion synthetic acceleration and the iterative strategy is as shown in Fig. 7. No outer iteration is performed and problem convergence is determined by the inner iteration.

B. Convergence Criteria

The convergence of the iterations is monitored at both the inner and the outer iteration level. The input parameters that control the number of iterations are EPSI, EPSO, IITL, IITM, and OITM found in BLOCK V of the Solver Module input.

1. Inner Iteration Convergence. The inner iterations for a given energy group are said to be converged when the pointwise scalar fluxes from one inner iteration to the next satisfy the condition:

$$max\left|\left(\phi_{i,g}^{\ell}-\phi_{i,g}^{\ell-1}\right)\right| < EPSI \quad , \tag{39}$$

where $\phi_{i,g}^{\ell}$ is the scalar fiux for mesh point i, Group g, and inner iteration ℓ, and where EPSI is the user-input inner iteration convergence criterion.

2. Diffusion Sub-Outer Iteration Convergence. The convergence of the diffusion sub-outer iterations requires the satisfaction of two criteria. Let us use the index ν to denote the outer iteration number and the index p to denote the diffusion sub-outer iteration number. Convergence of the diffusion sub-outers is then satisfied when both

$$max\left|\left(\phi_{i,g}^{p,\nu}-\phi_{i,g}^{p-1,\nu}\right)\Big/\phi_{i,g}^{p,\nu}\right| < 0.95 * EPSX \quad , \tag{40}$$

and

$$\left|1-\lambda_{D}^{p,\nu}\right| < EPSO \quad . \tag{41}$$

In the above

$$EPSX \equiv EPSI * \left[1 + NGROUP * exp(-100 * EPSI)\right] \quad , \tag{42}$$

where NGROUP is the number of energy groups, and

$$\lambda_{D}^{p,\nu} \equiv \left(F^{p,\nu},1\right)\Big/\left(F^{p-1,\nu},1\right) \quad . \tag{43}$$

The notation (F,G) denotes the inner product, or volume integral, of the product F*G.

3. Full Convergence. Full, or overall problem, convergence is achieved for problems requiring outer iterations when the fiux changes represented in Eqs. (39) and (40) are less than EPSX together with the additional requirements that

$$max\left|\left(\phi_{i,g}^{1,\nu}-\phi_{i,g}^{0,\nu}\right)\Big/\phi_{i,g}^{0,\nu}\right|<EPSX \quad , \tag{44}$$

and

$$\left|1-\lambda^{\nu}\right|<EPSO \quad , \tag{45}$$

where

$$\phi_{i,g}^{1,\nu} \equiv \textit{scalar flux at point } i, \textit{ group } g \textit{ from the first diffusion sub-outer iteration for outer iteration } \nu.$$

$$\phi_{i,g}^{0,\nu} \equiv \textit{scalar flux at point } i, \textit{ group } g \textit{ from the last diffusion inner iteration of outer iteration } \nu.$$

and

$$\lambda^{\nu} \equiv \left(F^{o,\nu},1\right)\Big/\left(F^{o,\nu-1},1\right) \quad . \tag{46}$$

In Eq. (46), $(F^{o,\nu},1)$ denotes the volume integral of the fission source rate distribution calculated at the end of the inner iteration cycle but before the performance of diffusion sub-outer iterations for outer iteration ν (see Fig. 7).

4. Iterative Loop Termination. Each of the iterative loops (inner iterations, diffusion sub-outer iterations, and outer iterations) is terminated when either the convergence criteria for that loop are met or when a specified maximum number of iterations have been attained.

For inner iterations the number of iterations is limited by the user input parameter IITL. If the user elects to omit this quantity, the code chooses an appropriate default value.

In problems where outer iterations are not required, that is, fixed-source problems with IEVT= 0, the value of IITL is usually chosen to be large, say 20-50, in order that the pointwise fiuxes be allowed to meet the convergence criterion before the number of inner iterations reaches IITL.

For eigenvalue problems (IEVT > 0), the usual procedure is to allow only one inner iteration per group until the fissions, upscatter sources, and diffusion scalar

fiuxes have neared full convergence. When this is achieved, the allowable number of inner iterations is increased to IITM (a user input quantity) which typically is in the range of 10-20 in order to permit full convergence of the transport fiuxes. The assumption here is that it is most efficient to first converge the fission/upscatter sources and then to converge the pointwise fiuxes. The code will switch the inner iteration limit from IITL to IITM when both

$$\left|1 - \lambda_D^{p,\nu}\right| < 3 * EPSO \quad , \tag{47}$$

and

$$max\left| \left(\phi_{i,g}^{p,\nu} - \phi_{i,g}^{p-1,\nu}\right) \Big/ \phi_{i,g}^{p,\nu} \right| < 10 * EPSX \quad , \tag{48}$$

where $\lambda_D^{p,\nu}$ is defined by Eq. (43) for diffusion sub-outer iteration p, outer iteration ν and EPSX is given by Eq. (42).

For fixed-source problems with fission and/or upscatter (IEVT= -1), a procedure similar to that for eigenvalue problems is followed except that IITL is usually chosen to be in the range of 2-5 instead of 1.

Diffusion sub-outer iterations are terminated when either the convergence criteria are met or when 100 sub-outers have been performed. The maximum allowable number of sub-outers is built into the code and is not under user control.

Outer iterations are terminated when either the full convergence criteria are met or when the number of outer iterations reaches OITM, a user-input quantity. If not supplied by the user, the code will default the value of OITM to 20.

C. Iteration Monitor Print

In the printed output from the Solver Module, an iteration monitor print is supplied for the user. The user should always inspect this monitor print to determine whether or not the problem has successfully converged.

1. General Aspects of the Monitor Print. At the end of each outer iteration the monitor provides the elapsed computer time in seconds, the outer iteration number, and the number of diffusion sub-outer iterations required. A number of sub-outer iterations of 100 implies that the diffusion sub-outer iteration did not converge to the criterion of Eqs. (40) and (41) before reaching the maximum allowable number of sub-outer iterations. Also provided is a message as to whether or not the inner iterations satisfied their convergence criterion, Eq. (39). Finally are included the values of $\lambda^{\nu} - 1$ and the maximum pointwise flux error corresponding to the values used in the test for full convergence given by Eqs. (45) and (44), respectively.

In addition to the basic outer iteration information described above, the monitor print provides an inner iteration monitor for certain outer iterations. This inner iteration monitor is normally provided for a fixed-source problem without fission or

upscatter (IEVT= 0) since only one outer iteration is required. The inner iteration monitor print is triggered by IITL being equal to IITM; this condition is assured by the defaults for these quantities when IEVT=0. The user may thus suppress this portion of the print by setting IITL to some number different from IITM. For other problems (IEVT$\neq$ 0) the inner iteration monitor is only provided for outer iterations following the satisfaction of the "nearly converged" conditions of Eqs. (47) and (48) at which time IITL is set equal to IITM by the code. In the inner iteration monitor are included the group number; the number of inner iterations taken, the maximum pointwise scalar flux error (see Eq. (39)), and the spatial mesh point where this maximum error occurred.

2. Warning Messages and Their Meanings. In the inner iteration monitor several warning messages are provided for the user if the calculation encountered some difficulty.

A message "ACCELERATION DISABLED" is printed when the transport correction to the diffusion coefficient, diffusion source, or diffusion removal term is such that the synthetic diffusion equation cannot be applied to accelerate that inner iteration. The presence of the message does not necessarily make the answers suspect if convergence is achieved; it merely tells the user that the inner iteration could not be accelerated.

The message "TRANSPORT FLUXES BAD" is a more serious warning. It is provided when nonpositive transport scalar fluxes exist following the last inner iteration. The presence of nonpositive scalar fluxes causes the diffusion inner iteration acceleration to be disabled. Although such a condition is not fatal, it does usually indicate that the spatial mesh is too coarse and that the results are suspect.

The message "NEG. SOURCE" is printed when a negative angular source term is calculated for one or more mesh cells. This situation can only occur when an anisotropic scattering source is being used, that is, when the parameter ISCT in BLOCK V is greater than zero. A negative source for a given angular direction is usually the result of the truncation of the Legendre expansion of the scattering source term in mesh cells where scattering dominates the source term. Both the scattering cross sections and the angular flux must be quite anisotropic for this condition to occur.

If the message "NEG. SOURCE - ACCELERATION DISABLED" appears, it means that the presence of negative angular sources has occurred to such a level that the synthetic diffusion equation could not be used to accelerate that inner iteration. The presence of this message does not necessarily make the answers suspect if convergence is achieved; it merely tells the user that the inner iteration could not be accelerated.

The message "NEG. SOURCE - TRANSPORT FLUXES BAD" is the most serious ramification of the presence of negative angular sources. It means that nonpositive transport scalar fluxes existed following the last inner iteration in conjunction with negative angular sources. Nonpositive scalar fluxes are not necessarily fatal, but they usually indicate that the truncated Legendre scattering expansion was quite poor and that any results are suspect.

The possible remedy for any of the "NEG. SOURCE" occurrences is to increase the Legendre scattering expansion order (increase the value of ISCT in BLOCK V) if the cross-section library contains data for the higher order scattering. The other remedy is to use the Cesàro transport correction described in Sec. I. below.

D. Boundary Conditions

Several boundary condition options are available to the user of ONEDANT as follows:

- Vacuum boundary condition – the angular flux on the boundary is identically zero for all incoming directions

- Reflective boundary condition – the incoming angular flux on the boundary is set equal to the outgoing angular flux in the direction corresponding to specular reflection.

- Periodic boundary condition – the incoming angular flux on one boundary is set equal to the outgoing angular flux in the same direction on the opposite boundary.

- White boundary condition – the incoming angular fluxes on the boundary are each set equal to the single value chosen such that the net flow across the boundary is zero, that is,

$$\psi_{incoming}(m) = \frac{\sum w_n \mu_n \psi(\mu_n)_{outgoing}}{\sum w_n \mu_n} \quad ,$$

  where the sums range over all outgoing directions. This condition is used primarily for cell calculations in cylindrical and spherical geometries where it is applied to the right (outer radial) boundary.

The above boundary conditions are controlled by the BLOCK V input parameters, IBL (left boundary), and IBR (right boundary). For planar geometries (IGEOM=1), both IBL and IBR must be specified. For curvilinear geometries (IGEOM=2 or 3), only IBR need be specified since the left boundary is assumed by the code to be at the radial origin (r=0), for which the curvilinear geometry, r=0 boundary condition is the only physical condition possible.

Note: Use of a reflective boundary condition (IBL or IBR=1) requires the S_N quadrature set to be symmetric about $\mu = 0$.

Two additional boundary conditions, not controlled uniquely by IBL/IBR, are the albedo and surface source conditions defined as follows:

- Albedo condition – the incoming angular flux on the boundary is set equal to a user-supplied albedo times the value it would have without the albedo. It may be used in conjunction with either the reflective or white boundary condition described above. The use of albedoes is controlled solely by the presence of the LBEDO and/or RBEDO array specifications in the BLOCK V card-image input.

- Surface source boundary conditions – the incoming angular fluxes on the boundary are set equal to the user-supplied values as specified in the SILEFT or SALEFT and/or SIRITE or SARITE input arrays in BLOCK V. See discussion of inhomogeneous sources in Sec. G of this chapter.

E. Input of Quadrature Sets

The ONEDANT code package has the option of obtaining the discrete-ordinates angular quadrature coefficients from a SNCONS standard interface file,[3,4] from one of three built-in sets in subroutine SNCON, or from card-image input. The input parameter IQUAD in BLOCK V of the card-image input specifies the source of these coefficients. The number of quadrature coefficients, MM, is determined from the input S_N order parameter ISN and the geometry specification input parameter IGEOM, both found in input BLOCK I. Values of MM are shown in Table VIII in Ch. III.

The built-in constants provided in the code are

(i) the Gaussian P_N constants (IQUAD=1) for $S_2, S_4, S_6, S_8, S_{12}, S_{16}, S_{20}, S_{24}, S_{32}$, and S_{48};

(ii) the double Gaussian DP_N constants (IQUAD=2) for $S_4, S_8, S_{12}, S_{16}, S_{24}, S_{32}$, S_{40}, S_{48}, S_{64}, and S_{96}; and

(iii) generalized quadrature, GQ_N, constants (IQUAD=4) for S_4, S_6, S_8, S_{12}, and S_{16}.

For most problems the P_N set is satisfactory. For thin-slab problems in which the angular representation for the leakage flux is important, the DP_N set is recommended. For cylindrical or two-angle plane calculations with anisotropic scattering, the GQ_N set is recommended. The GQ_N set for cylinders and two-angle planes is a generalized even-moment fully symmetric quadrature set.

For problems with anisotropic scattering, it is important that the S_N order be chosen sufficiently large such that the spherical harmonic polynomials described in Section A of Ch. III are correctly integrated. Otherwise, the numerical quadrature error may introduce nonphysical contributions to the neutron balance, preventing convergence of the problem to the desired precision.

For user card-image input of S_N quadrature sets through the WGT and MU arrays in BLOCK V, it is necessary that the sets be correctly ordered as illustrated in Figs. 4, 5, and 6. In addition, if the sums $1 - \sum_m w_m$, $\sum_m \mu_m$, and $\sum_m w_m \mu_m$ exceed 10^{-5}, an error message is printed. It should be noted that if the user provides the card-image input arrays WGT and MU, the code will use these arrays for the quadrature constants irrespective of the value of IQUAD entered in the input, that is, the WGT and MU input arrays will override any other source of quadrature constants.

F. Zone-Dependent Fission Fractions (the CHI Array)

For problems containing fissile or fissionable nuclides it is necessary to provide the code with the groupwise fission fractions, χ_g, g=1,2 ...,NGROUP. From certain of the cross-section libraries (ISOTXS, GRUPXS, MENDF, etc.) a single χ vector characterizing the dominant fissionable isotope (U235, U238, Pu239, etc.) can be extracted automatically from the library and used for the problem. Alternatively, a single χ vector can be provided by the user in BLOCK III through the card-image CHIVEC input array. Again, this is a single χ vector characterizing the dominant fissionable isotope in the problem.

In BLOCK V of the input is provided the card-image CHI input array that can optionally be used to provide multiple χ vectors so that each zone in the problem can use the fission fraction characteristic of the dominant fissionable isotope *in that zone*. The availability of such zone-dependent χ's is particularly useful for problems in which one zone may have a different dominant fissile isotope than another zone. The CHI array is input as N distinct χ vectors ($1 \leq N \leq$ NZONE) with each vector containing NGROUP χ values. The first χ vector is applied to the first zone, where the first zone is that defined by the first entry in the ASSIGN array of BLOCK IV; the second χ vector is applied to zone number 2 (define by the second entry in the ASSIGN array), etc. If the total number of χ vectors in the CHI array is less than NZONE (the total number of zones) then the last χ vector in the CHI array is used for all remaining but unspecified zones.

Note that if the CHI array is used in BLOCK V, its entries will override the χ's provided either from the cross-section library or from the CHIVEC array in BLOCK III.

G. Input of Inhomogeneous Sources

The Solver Module of ONEDANT will solve the inhomogeneous form of the transport equation, Eq. (11), using the multigroup, discrete-ordinates approximation outlined in Section A of this chapter. The user specifies this type of calculation by setting the input control word IEVT to 0 or -1 (IEVT is found in the collective input array SOLIN in BLOCK V). IEVT=0 is used when there is neither fissionable material nor upscattering in the problem and IEVT=-1 is used when either fissionable material is present (but not in sufficient amount to make the system nuclearly critical or supercritical) or upscattering is present.

The user must supply the specifications for the inhomogeneous sources either in the input or from a FIXSRC[3] standard interface file. The inhomogeneous sources may be spatially distributed on the interior of the problem (distributed source) and/or may be external boundary (surface) sources. If the sources are to be input via FIXSRC standard interface file, the user sets the input control word INSORS to 1 (INSORS is found in the collective input array MISC in BLOCK V). If INSORS is not input with value of unity, the user must supply the source specifications in the input of BLOCK V as described below.

1. Distributed Source Input. As described in Section III.A.3, the inhomogeneous distributed source must be represented by the spherical harmonic expansion, Eq. (8) in multigroup form:

$$Q_g(r,\Omega) = \sum_{n=1}^{NMQ} (2L+1) R_n(\Omega) \tilde{Q}_{n,g}(r) \quad , \quad g = 1,\ldots,NGROUP \quad . \tag{49}$$

Through the SOURCF or the SOURCE and/or SOURCX input arrays in BLOCK V of the input, the user inputs the $\tilde{Q}_{n,g}(r)$ of Eq. (49). If input is via the SOURCF array, the input values are used directly as $\tilde{Q}_{n,g}(r)$. If input is via either SOURCE or SOURCX (or both) arrays, the input must be supplied such that SOURCE (g,n)* SOURCX $(r,n) = \tilde{Q}_{n,g}(r)$. The number of moments, NMQ, in Eq. (49) is determined solely from the number of moments supplied in the input arrays. The proper number of moments for a given Legendre order of anisotropy of the distributed source is shown in Table VI of Ch. III for each geometry. For example, if one wishes to enter a P_3 inhomogeneous source in cylindrical geometry, Table VI shows that six spherical harmonics are required for P_3 in cylindrical geometry. Table VII in Ch. III shows that source moments for the spherical harmonics $P_o(\xi), P_1^1(\xi)cos\phi, P_2(\xi), \frac{\sqrt{3}}{6}P_2^2(\xi)cos2\phi, \frac{\sqrt{6}}{6}P_3^1(\xi)cos\phi$, and $\frac{\sqrt{10}}{60}P_3^3(\xi)cos3\phi$ are needed. These moments are defined by Eq. (7) using multigroup notation and recalling that for cylindrical geometry μ is replaced by ξ. The six moments to be supplied in the input are thus:

$$\tilde{Q}_{1,g}(r) = \frac{1}{4\pi}\int_{-1}^{1} d\xi \int_{0}^{2\pi} d\phi P_0(\xi) Q_g(r,\xi,\phi) = Q_{0,g}(r) \quad , \tag{50a}$$

$$\tilde{Q}_{2,g}(r) = \frac{1}{4\pi}\int_{-1}^{1} d\xi \int_{0}^{2\pi} d\phi P_1^1(\xi)cos\phi \; Q_g(r,\xi,\phi) = Q_{c,1,g}^1(r) \quad , \tag{50b}$$

$$\tilde{Q}_{3,g}(r) = \frac{1}{4\pi}\int_{-1}^{1} d\xi \int_{0}^{2\pi} d\phi P_2(\xi) Q_g(r,\xi,\phi) = Q_{2,g}(r) \quad , \tag{50c}$$

$$\tilde{Q}_{4,g}(r) = \frac{1}{4\pi}\int_{-1}^{1} d\xi \int_{0}^{2\pi} d\phi \frac{\sqrt{3}}{6} P_2^2(\xi)cos2\phi \; Q_g(r,\xi,\phi) = Q_{c,2,g}^2(r) \quad , \tag{50d}$$

$$\tilde{Q}_{5,g}(r) = \frac{1}{4\pi}\int_{-1}^{1} d\xi \int_{0}^{2\pi} d\phi \frac{\sqrt{6}}{6} P_3^1(\xi)cos\phi \; Q_g(r,\xi,\phi) = Q_{c,3,g}^1(r) \quad , \qquad (50e)$$

and

$$\tilde{Q}_{6,g}(r) = \frac{1}{4\pi}\int_{-1}^{1} d\xi \int_{0}^{2\pi} d\phi \frac{\sqrt{10}}{60} P_3^3(\xi)cos3\phi \; Q_g(r,\xi,\phi) = Q_{c,3,g}^3(r) \quad , \qquad (50f)$$

for n=1,...,NGROUP. It should be recognized that the source moments above are not input as continuous variables in space, r, but are input by fine spatial mesh interval i, i=1,...,IT.

It is worth noting that most inhomogeneous distributed sources are assumed to be isotropic, so that NMQ in Eq. (49) is unity and the only source moment entered is the zeroth moment.

$$\tilde{Q}_{1,g}(r) = \frac{1}{4\pi}\int_{-1}^{1} d\xi \int_{0}^{2\pi} d\phi Q_g(r,\xi,\phi) = Q_{0,g}(r) \quad , \qquad (51)$$

which, in fact, is simply the scalar source distribution.

The units on the input source moments $\tilde{Q}_{n,g}(r)$ are [particles per unit time and unit volume].

2. Surface (Boundary) Source Input. With a surface (boundary) source present, the incoming angular fiux on the surface is set equal to a user-supplied source, Q_m:

$$\psi(\mu_m)_{incoming} = Q_m \quad .$$

The units on the surface source are the same as those for angular flux.

The user-supplied source is group-dependent and may either be angularly isotropic or angle-dependent. The user-supplied sources may be input either by BLOCK V card-image input or via a FIXSRC[3,4] standard interface file.

For card-image input the left boundary surface sources are input via the SILEFT array (angularly isotropic) or the SALEFT array (for angle-dependent sources). Similarly, right boundary surface sources are input via the SIRITE or SARITE arrays. Note that surface sources may only be input at either the left or right (or both) external boundaries of the physical problem. For the angle-dependent surface sources, only the *incoming* directions are required, but they must be ordered as described below.

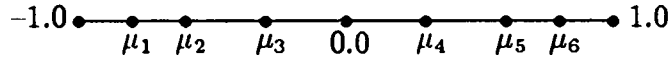
For input of surface sources via a FIXSRC standard interface file, the user-input parameter INSORS in BLOCK V must be set to unity and the appropriate FIXSRC file must be available to ONEDANT at the time of code execution. Note that ONEDANT will only accept surface sources at either left or right (or both) external boundaries of the physical problem.

Angle-dependent surface sources can be input to ONEDANT through the SARITE and SALEFT input arrays. Currently, SALEFT can only be used in slab (plane) geometry. The surface sources are actually angular fluxes, $\psi_{m,g}$, at the right or left surfaces [m denotes the order index (with m correlated to specific quadrature directions), g denotes energy group]. SARITE and SALEFT entries are made as follows:

$$\psi_{1,1}, \psi_{2,1}, \psi_{3,1}, \ldots \; ; \; \psi_{1,2}, \psi_{2,2}, \psi_{3,2}, \ldots \; ; \; \ldots$$

- Slab Geometry (IGEOM= 1 or SLAB or PLANE):

  The ordering of the angles is as shown below (S_6 used for illustration).



  with $|\mu_1| = \mu_6, |\mu_2| = \mu_5$, and $|\mu_3| = \mu_4$.

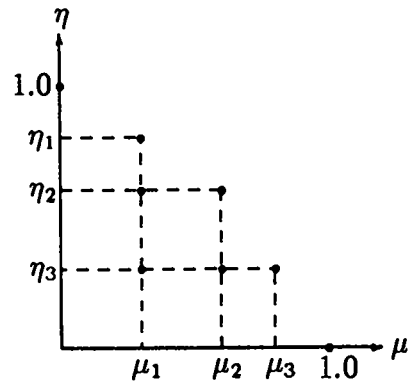
  Using the S_6 quadrature illustrated above, the correspondence between the SARITE and SALEFT ordering index, m, and the angular directions (shown above) is as follows:

m	SARITE μ	SALEFT μ
1	μ_1	μ_4
2	μ_2	μ_5
3	μ_3	μ_6

  NOTE: The SALEFT ordering differs from the SARITE ordering in terms of the $|\mu|$ associated with the ordered angular fluxes. For example, the first entry in SARITE (for each group) corresponds to μ_1 while the first entry in SALEFT (for each group) corresponds to μ_4 and $|\mu_1| \neq \mu_4$.

- Cylindrical Geometry (IGEOM= 2 or CYLINDER or CYL):

  Using S_6 for illustrative purposes, the principal octant ($\mu > 0, \eta > 0$) of the unit sphere of angular directions is as shown below



  The order, m, in which the SARITE angular fluxes are entered for each energy group is correlated to the (μ, η) angular directions (shown above) in the table below:

m	μ	η
1	$-\mu_1$	η_1
2	$-\mu_2$	η_2
3	$-\mu_1$	η_2
4	$-\mu_3$	η_3
5	$-\mu_2$	η_3
6	$-\mu_1$	η_3

H. Normalization of the Calculation (the NORM Parameter)

In BLOCK V of the input is the parameter NORM. Through the use of NORM the user may specify whether or not the calculation should be normalized to a particular *particle production rate* as described below.

If NORM= 0 or omitted from BLOCK V no normalization is done.

If NORM= XX, where XX is positive, and inhomogeneous (fixed) sources exist (IEVT$\leq$ 0), then the calculation will be conducted and normalized such that the *total rate at which particles are introduced into the system from inhomogeneous sources* is XX. Thus, all fluxes that are in the calculation are normalized to the values they would have if the *total* rate at which particles are introduced in the system from inhomogeneous sources (distributed and/or surface sources) is XX particles/unit time (or particles if fluxes are to be interpreted as fluences).

If NORM= XX where XX has a positive value, and an eigenvalue calculation is being conducted (IEVT> 0), then the calculation is performed and normalized such that the *total rate at which neutrons are produced by fission reactions* is XX. Thus, all fiuxes that are computed in the calculation are normalized to the values they would have if the *total* rate at which neutrons are produced by fissions in the system is XX neutrons/unit time (or neutrons if fiuxes are to be interpreted as fiuences).

Note that NORM applies normalization only to *particle production rates.* Normalization of edit results to a desired fission *power* level can be cone through the use of the POWER and MEVPER parameters in BLOCK VI as described in Ch. V. These latter two parameters effect normalization in only the Edit Module.

I. Applying Transport Corrections to the Cross Sections (the TRCOR Parameter)

In BLOCK V of the input is the parameter TRCOR which permits the user to have the Solver Module perform its calculation using transport-corrected cross sections. Below is provided a little background on the use of the transport correction in ONEDANT/TWODANT.

As described in Ch. III, a truncated spherical harmonics, or Legendre polynomial, expansion of the scattering sources is made per Eq. (2). The scattering order ISCT (an input parameter in BLOCK V) determines where the expansion is to be truncated. Any value of ISCT (0≤ ISCT ≤ MAXORD) is allowed. Recall that MAXORD is the highest Legendre order for which scattering matrix cross sections exist in the cross-section library being used. The principal reason for choosing to truncate the expansion to an order less than MAXORD is because of computer storage. Each term in the spherical harmonics (Legendre) expansion requires computing and storing an angular fiux moment for every energy group and spatial mesh cell, and the number of such flux moments increases with scattering order (see Table VI in Ch. III).

If ISCT is less than MAXORD, it is usually advantageous to apply a transport correction to the truncated Legendre scattering cross sections. Transport corrections are designed to account approximately for terms in the expansion being omitted by the truncation. There are three different types of transport corrections allowed in the code. They are selected through the use of the parameter TRCOR in BLOCK V of the input.

- If TRCOR= NO or if TRCOR is omitted, no transport correction is applied.
- If TRCOR= DIAG the diagonal transport correction is applied. With this correction the material macroscopic total cross section for each group and the material macroscopic within-group scattering cross section for each group and each scattering order is modified as shown below:

$$\hat{\sigma}_{t,g} = \sigma_{t,g} - \sigma_s^{ISCT+1}\ (g \rightarrow g)$$

and

$$\hat{\sigma}_s^{\ell}\,(g \rightarrow g) = \sigma_s^{\ell}\,(g \rightarrow g) - \sigma_s^{ISCT+1}\,(g \rightarrow g),\ \ell = 0, 1, \ldots, ISCT \quad .$$

where the notation $\hat{\sigma}$ denotes "transport corrected" cross section. The diagonal transport correction is normally recommended.

- If TRCOR= BHS, the Bell-Hansen-Sandmeier transport correction is applied. With this correction the macroscopic total cross section for each material and group and the macroscopic within-group scattering cross sections for each scattering order, material, and group are modified as follows:

$$\hat{\sigma}_{t,g} = \sigma_{t,g} - \sum_{g'}^{\Gamma} \sigma_s^{ISCT+1}\,(g \rightarrow g')$$

and

$$\hat{\sigma}_s^{\ell}\,(g \rightarrow g) = \sigma_s^{\ell}\,(g \rightarrow g) - \sum_{g'} \sigma_s^{ISCT+1}\,(g \rightarrow g')\ ,\ \ell = 0,\ 1,\ \ldots,\ ISCT \quad .$$

- If TRCOR=CESARO, the n^{th}-Cesàro-mean-of-order 2 transport correction is applied where n=ISCT. With this correction the macroscopic group-to-group scattering cross section for each scattering order, zone, and group is modified as shown below:

$$\hat{\sigma}_s^{\ell}\,(g \rightarrow g') = \frac{(ISCT+2-\ell)(ISCT+1-\ell)}{(ISCT+2)(ISCT+1)}\sigma_s^{\ell}(g \rightarrow g')\ ,\ \ell = 1, \ldots, ISCT \quad .$$

This Cesàro correction ensures that the truncated Legendre expansion for the scattering cross section from group g to group g' is (i) positive, (ii) preserves the $\ell = 0$ (P_o) moment of the scattering cross section, and (iii) converges to the same value as $\sigma_s^{\ell}(g \rightarrow g)$ for large values of ISCT (see Ref. 10).

Note that if ISCT= 0, the Cesàro correction does *nothing* to the cross sections – the multiplier of $\sigma_s^{\ell}(g \rightarrow g')$ is unity. Generally, the Cesàro transport correction should only be used with ISCT≥2.

At the time of the writing of this manual, little experience has been had with the Cesàro correction in terms of its accuracy. As a result the user is cautioned

regarding its use. However, using this transport correction will eliminate the possibility of negative sources, as discussed in Sec. C.2 of this chapter.

Please note that the transport corrections described above are only applied to the macroscopic cross sections used in the Solver Module. None of the cross-section files are altered by the use of the transport correction in the Solver Module.

J. Buckling Corrections

Leakage from the transverse dimension(s) of a multidimensional system may be simulated by using a user-specified buckling height (BHGT) and/or buckling width (BWTH) in the BLOCK V card-image input. For plane and 2-angle plane geometries (IGEOM=1), both BHGT and BWTH may be specified. For cylindrical geometry only the buckling height, BHGT, may be specified. The buckling dimensions are in units consistent with the units on cross section, for example, in cm if macroscopic cross sections are in cm^{-1}. If diffusion theory is assumed adequate, then the fiux shape in the transverse directions, say z, is of the form $cos\pi z/\tilde{h}$ so that the fiux shape function vanishes at the extrapolated system half-heights $\pm\tilde{h}/2$. Applying this to the transport equation the transverse leakage appears as a buckling absorption with a buckling absorption cross section

$$\sigma_{a,BHGT} = \frac{\sigma}{3}\left[\frac{\pi}{\sigma * BHGT + 1.4209}\right]^2 \quad ,$$

where σ is the macroscopic zone total cross section, BHGT (or similarly BWTH) is the buckling height (or buckling width), and $1.4209/\sigma$ is twice the Milne planar extrapolation distance.

The buckling absorption correction is applied to both the total cross section and absorption cross section for each group and zone in the physical problem. Consequently, the absorption rate printed in the output Solver Module coarse-mesh balance table contains this buckling absorption.

K. Eigenvalue Searches

It is possible in ONEDANT to perform an eigenvalue search on material concentration (concentration search), system dimensions (dimension search), or the time absorption (alpha search) to achieve a desired value of k_{eff}. The type of search is controlled by the input parameter IEVT supplied in BLOCK V of the card-image input as follows:

IEVT*	Type of Eigenvalue Search
2	Time absorption (alpha)
3	Concentration
4	Critical size (dimension)

*Not included here are the options IEVT=–1 for inhomogeneous source problems with upscatter and/or fission, IEVT=0 for inhomogeneous source problems without upscatter or fissions, and IEVT=1 for k_{eff} calculations.

For time-absorption calculations, the time-dependent angular flux is assumed to be separable in time and space, viz.,

$$\psi(r,\underline{\Omega},t) = \psi(r,\underline{\Omega})\, e^{\alpha t} \quad .$$

If this assumption is inserted into the time-dependent transport equation, the exponentials cancel and a fictitious cross-section term of the form α/v_g appears as a correction to the total and absorption cross sections. Here v_g is the neutron speed associated with energy group g. The exponential factor α is then the eigenvalue sought in the time-absorption eigenvalue search. Obviously, $\alpha = 0$ for an exactly critical system, and $\alpha > 0$ for a supercritical system.

For concentration searches, the material concentrations are modified in accordance with the description provided under the ASGMOD array in BLOCK IV of the card-image input (see Ch. V).

For dimension searches, the coarse-mesh boundaries can be modified selectively to obtain a critical system. The modified coarse-mesh boundaries, $\tilde{R}_k$, are calculated from the initial input boundaries, R_k, by

$$\tilde{R}_{k+1} = \tilde{R}_k + (R_{k+1} - R_k) * (1 + EV * RM_k), \quad k = 1,2,\ldots,IM \quad , \qquad (52)$$

where EV is the eigenvalue sought in the search. The factors RM_k are the coarse-mesh radii modifiers which are input by the user via the RM array in the BLOCK V card-image input, and control how the coarse-mesh boundaries are modified. Clearly, if RM_k is zero, the thickness of the k^{th} zone is not altered. If all RM_k are unity, the system dimensions are uniformly expanded (EV>0) or contracted (EV<0). Many sophisticated changes can be made, limited only by the ingenuity of the user. For example, an interface between two zones may be moved while the remainder of the system is left unchanged.

In all three types of searches the appropriate system parameter may be adjusted to achieve the desired value of k_{eff}. This value is taken to be unity (criticality) unless the input parametric value type (IPVT in BLOCK V of the card-image input) is set to unity. If IPVT=1, the desired parametric value of k_{eff} is input by the user as PV (in BLOCK V).

For concentration searches (IEVT=3) and dimension searches (IEVT=4), it is also possible to adjust the appropriate system parameter to achieve a system whose neutral particle fiux is changing exponentially in time at the rate $e^{\alpha t}$ by setting the input parametric value type, IPVT, to 2. In this case the user enters the desired exponential factor α as the parametric value PV in the input. Note that an α of 0.0 corresponds to a normal concentration or dimension search on a k_{eff} of unity.

It is important to recognize that the value of PV input by the user remains fixed throughout the search process.

Regardless of the parameter being adjusted, the search is executed by performing a sequence of k_{eff}-type calculations, each sequence for a different value of the parameter being treated as the eigenvalue. The search is for a value of the parameter that makes the value of λ unity where λ is defined as

$$\lambda = \frac{(Fission\ source)^k + Inhomogeneous\ source}{(Fission\ source)^{k-1} + Inhomogeneous\ source} \quad , \tag{53}$$

for the k^{th} outer iteration. The search is controlled by the subroutine NEWPAR in the Solver Module.

In the following description of NEWPAR, it is helpful to refer to Fig. 8 in which the deviation of λ from unity for each outer iteration is plotted.

For the initial system, NEWPAR continues the outer iteration until two successive values of λ differ by less than EPSO. For subsequent sequences of λ values, a different convergence precision, XLAX, is used. After the first converged λ sequence is obtained, the initial value of the eigenvalue (EV) is altered by EVM, an input value. If $\lambda > 1$ (multiplying system), the new eigenvalue is equal to EV+EVM if $\lambda < 1$ (decaying system), the new value is equal to EV-EVM. Thus, the sign and value of EVM should be chosen such that the use of EV+EVM will reduce the reactivity of the system. Conversely, the use of EV-EVM should increase the reactivity of the system.

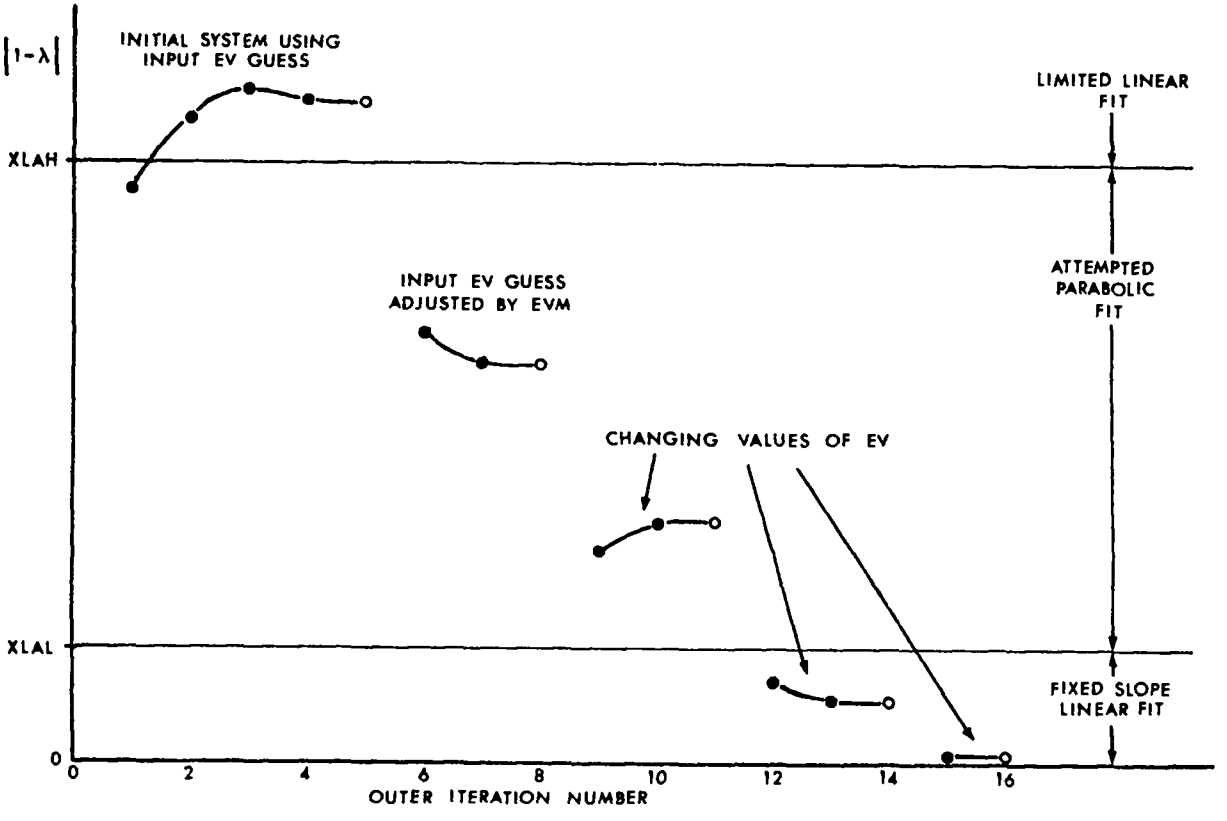


Fig. 8. Variation of λ during a hypothetical eigenvalue search.

Basically, after two converged values of λ are obtained for two different system configurations, subroutine NEWPAR attempts to fit a curve through the most recent values to extrapolate or interpolate to a value of unity. Depending on the amount of information available and the size of $|1-\lambda|$, this fit proceeds in different ways. A parabolic fit cannot be made until three converged values of λ are available, and is not attempted unless $|1-\lambda|$ is greater than an input search lower limit (XLAL) and less than an input search upper limit (XLAH). If a parabolic fit is tried and the roots are imaginary, a straight-line fit is used. If the roots are not imaginary, the closest root is used as the new value of EV. Once a bracket is obtained (change of sign of $\lambda-1$), the fit procedure is not allowed to move outside the region of the bracket. Should a parabolic fit select an eigenvalue outside the bracket region, this value is rejected and the new value is taken to be one-half the sum of the previous value and the value previous to that.

Whenever the parabolic fit is not used, a linear fit is used and the new eigenvalue is computed from

$$(EV)_{new} = (EV)_{old} + POD * EVS * (1-\lambda) \quad , \tag{54}$$

where POD is an input "parameter oscillation damper" that may be used to restrict the amount of change in the eigenvalue. In Eq. (54), EVS is a measure of the slope of the curve. When $|1-\lambda| > XLAH, (1-\lambda)$ in Eq. (54) is replaced by XLAH (with the correct sign) to prevent too large a change in EV. After $|1-\lambda| < XLAL$, the value of EVS is fixed and kept constant until convergence to prevent numerical difficulty in the approximation of the derivative when λ is close to unity.

Because parametric search problems represent sequences of k_{eff} calculations, it behooves the user to study the use of subroutine NEWPAR in order to optimize his calculations. It also behooves the user to pose soluble problems. That is, there are many problems, especially concentration searches, for which solutions are not possible, and discovering this by trial and error is the hard way. Ideally, the user will have some estimate of the critical parameter available from a lower order computation.

Convergence in time-absorption calculations is typically one-sided. If EV (the eigenvalue α) is negative, then there is a possibility that the corrected removal cross section will become negative. If this happens, the automatic search procedure may fail dramatically. For this reason POD=0.5 or less is frequently used in such searches.

L. Adjoint Computations

The ONEDANT code package solves the adjoint transport equation by transposing (in energy) the matrices of scattering cross sections and inverting the group order of the problem. The transposition of the scattering matrix converts a downscatter problem to an upscattering problem so that by inverting the group order the problem will execute in a downscatter-like mode. In addition to transposing the scattering matrices, the fission source term in the transport equation is transposed so that instead of $\chi_g \sum (\nu\sigma_f)_h \phi_h$, one has $(\nu\sigma_f)_g \sum \chi_h \phi_h$. The code does

not transpose the angular direction matrix associated with the leakage terms in the transport equation. Instead, the adjoint calculation of the leakage operator proceeds as in the direct (forward) calculation, but the results of the adjoint calculation for direction $\underline{\Omega}$ must be identified as the adjoint solution for direction $-\underline{\Omega}$. For example, the vacuum boundary condition at a surface (no incoming angular flux) in an adjoint calculation must be interpreted as a condition of no outgoing flux. Likewise, the adjoint leakage at a surface must be interpreted as incoming instead of outgoing.

All group-order inversions and fission source and scattering matrix transpositions are performed by the code; the user need only set the input parameter ITH in BLOCK V to unity to effect an adjoint calculation. (If the problem contains inhomogeneous sources, these source must quantitatively be, of course, the adjoint sources.)

The printed output from the Solver Module in an adjoint calculation indicates the correct group ordering and need not be inverted by the user. The adjoint fluxes from the Solver Module are written to a binary ATFLUX standard interface file. The ATFLUX file description is shown in Ref. 3.

In performing the adjoint reversals of the scattering matrices and the group inversions, the code prepares a binary, code-dependent interface file ADJMAC. This ADJMAC file contains the adjoint-reversed material cross sections to be used by the Solver Module. ADJMAC is essentially the adjoint-reversed counterpart to the MACRXS file described in Ch. IX, and the rules for saving and using an existing ADJMAC file are the same as for an existing MACRXS file.

The performance of adjoint edits is described in Chapter XI.

XI. DETAILS RELATED TO EDIT MODULE INPUT AND EXECUTION (BLOCK VI)

The basic function of the Edit Module is to perform postprocessing, or edit, operations using multigroup, pointwise scalar fluxes generated in a previous execution of the solver module or, perhaps, in some other neutronics code. The Edit Module uses the scalar fluxes, multiplies them by suitable quantities hereafter called response functions, calculates sums of these products over space and/or energy (if desired), and produces printed output of the results. Recall, from Ch. II of this manual, that the Edit Module is essentially a free-standing code module accepting only interface files as input. Most of these interface files are general in nature in that they apply both to the Solver and the Edit Modules (see Table I in Ch. II). Included in these general files are the geometry specifications (GEODST file), the material mixing and cross-section specifications (NDXSRF, ZNATDN, and SNEXDT files), and the assignment of materials to zones specifications (ASGMAT file). Another general file required by the Edit Module is a standard scalar flux interface file, either regular (forward) scalar fluxes (RTFLUX file), or adjoint scalar fluxes (ATFLUX file). Either an RTFLUX or an ATFLUX file is automatically provided by the Solver Module when it is executed. The specific edit operations to be performed using the information from the above general files are provided to the Edit Module by means of an EDITIT interface file. This file is created by the Input Module solely from user card-image input in BLOCK VI of the input data.

Because of the structure and interface file linkage of the ONEDANT code, several different Edit Module runs can be performed using the same set of general files. For example, once the Solver Module is executed and its scalar flux interface file saved, the Edit Module can be repeatedly executed without re-execution of the Solver Module. Only the Edit Module card-image input need be changed so that a new EDITIT file is created between runs.

The remainder of this chapter provides details pertinent to the editing options available to the user in the Edit Module card-image input (BLOCK VI).

A. Spatial Options for Edits

Edits can be performed on the fine spatial mesh points (as specified in BLOCK II geometry input) or on integrals over specified spatial intervals (called Edit Zones).

The fine space-point option is chosen by setting the BLOCK VI input parameter PTED to unity. In this form the edit quantity, denoted by ρ, for the i^{th} spatial mesh point is computed as

$$\rho_{i,g'} = \sum_{g \epsilon g'} \phi_{i,g} R_{i,g} \quad , \tag{55}$$

where

$\phi_{i,g}$ = scalar fiux for mesh point i, energy group g.

$R_{i,g}$ = a response function which may be either input directly. via the RSFE and RSFX arrays (below) or formed from input cross sections.

g' denotes an Edit energy-broad-group (See XI.B) consisting of one or more Solver energy groups.

With the BLOCK VI input parameter BYVOLP set to unity, the above edit quantity will be multiplied by the mesh interval "volume" V_i. The user may also select those points, or intervals, for which he wishes point edits by use of the POINTS input array in BLOCK VI. If PTED=1 and the POINTS array is not specified, the code will provide output for all mesh points (default).

To obtain edit quantities that are integrals over desired spatial intervals, the input quantity ZNED is set to unity. The desired spatial intervals, called Edit Zones, are specified by the user through the EDZONE array in input BLOCK VI. In specifying the Edit Zones through the EDZONE array the following rules must be observed:

(i) each and every fine-mesh interval (point) must be assigned to an Edit Zone, that is, given an Edit Zone number,

(ii) Edit Zone boundaries are arbitrary, that is, they are independent of coarse-mesh or material boundaries,

(iii) Edit Zone numbers must be positive integers in the range 1,2...,N where N is the total number of Edit Zones desired.

Example: Given a problem with 30 mesh intervals. It is desired that edit quantities be produced that are integrals over the first 10-mesh intervals, the second 10-mesh intervals, and the remaining 10-mesh intervals. There are thus 3 Edit Zones each comprising 10-mesh intervals. Using the free-field repeat option of Table IX, the EDZONE specification could be provided as EDZONE= 10R1, 10R2, or 10R3 to specify that the first 10-space intervals are in Edit Zone 1, the second 10 in Edit Zone 2, and the third 10 in Edit Zone 3. It should be noted that the ordering of the Edit Zones 1, 2, 3 with the first, second, and third set of 10-mesh points is not required.

Thus, with ZNED=1, the EDIT module will produce edit quantities, ρ, for Edit Zone Z_m as

$$\rho_{Zm,g'} = \sum_{g \epsilon g'} \sum_{i \epsilon Z_m} \phi_{i,g} R_{i,g} V_i \quad . \tag{56}$$

If Edit Zone edits are requested (ZNED=1) and the EDZONE array is not specified, the code will assume a default specification of the Edit Zones equal to the Coarse-Mesh intervals (see XMESH input array in BLOCK II).

IMPORTANT NOTE: In order to get printed output from the EDIT module, either point edits (PTED=1) or edit zone edits (ZNED=1) or both must be specified.

B. Energy-Group Options for Edits

The user may select the energy-group structure desired for the edit output by means of the ICOLL input array in BLOCK VI. Through this input array the user can collapse the energy-group structure used in the Solver Module down to fewer (broader) groups for edit purposes.

Example: Consider a 24 energy-group structure used by the Solver Module in which the first 12 groups are considered "fast" groups, groups 13 through 21 are "epithermal" groups, and groups 22 through 24 are "thermal" groups. If it is desired that quantities be calculated as integrals (sums) over the three that edit broad groups denoted fast, epithermal, and thermal, the ICOLL array would be specified as ICOLL=12, 9, 3 to collapse the first 12 groups into Edit Energy-Broad-Group 1 (the "fast" broad group), the next 9 groups (groups 13 through 21) into Edit Energy-Broad-Group 2 (epithermal), and the last 3 groups (groups 22 through 24) into Edit Energy-Broad-Group 3(thermal).

If the ICOLL array is not specified, the code will assume the default condition of one Solver Module energy group per Edit Energy-Broad-Group.

The IGRPED input parameter in BLOCK VI is used to control the printed output with respect to the Edit Energy-Broad-Groups. With IGRPED=0 only the energy-group total (sum over all groups) of the edit quantities is printed. With IGRPED=1 or =2 edit quantities for each of the Edit Energy-Broad-Groups are printed. With IGRPED=3 edit quantities for each Edit Broad Group plus the energy-group total are printed.

C. Forms of Response Functions

As indicated in the preceding sections, edit quantities all involve taking the product of the scalar fiux, $\phi_{i,g}$, and a response function, $R_{i,g}$, (for spatial mesh point i and energy group g). In this section are described the various forms that the response function $R_{i,g}$ can take.

1. Cross-Section Response Functions: EDXS Input Array. Response functions can be formed directly from cross-section data. In this case it is necessary to specify the particular type, or types, of cross sections to be used, that is, (n,γ), (n,α), total, absorption, etc. The cross-section data provided to the Edit Module on the SNXEDT file will contain a particular cross-section type in a unique position within the cross-section data table as indicated in Table Xa, Xb. Through the EDXS input array in the BLOCK VI input, the user specifies which cross-section types are desired using either the integer edit position numbers or the Hollerith names as given in Table Xa, Xb.

Example: Consider a problem in which isotope cross sections were supplied by means of an ISOTXS binary file. It is desired that edits be performed using both the n,α and n,γ cross sections. Using Table Xa, the EDXS array would be input as EDXS= "N-ALPHA," "N-GAMM" or, alternatively, as EDXS= 8, 10.

The specific forms of cross-section-based response functions available in the EDIT module are the resident macroscopic, isotope microscopic, constituent, and material forms. Each of these is described below.

a. Resident Macroscopic Cross-Section Response Functions: RESDNT Input Parameter. The resident macroscopic cross section, $\sum_{i,g}^{RES}$, at mesh point i, energy group g is defined as the actual macroscopic cross section that was used by the Solver Module. To obtain this response function, namely

$$R_{i,g} = \Sigma_{i,g}^{RES} \quad ,$$

the Block VI input parameter RESDNT is set to unity.

b. Isotope Microscopic Cross-Section Response Functions: EDISOS Input Array. Isotope microscopic cross section, σ_g^{ISO}, may be used for the response functions by identifying the isotopes desired through the EDISOS BLOCK VI input array. In this edit the cross sections are taken directly from the EDIT module file SNEXDT, which themselves originally came from the basic cross-section library. Note that the response function

$$R_{i,g} = \sigma_g^{ISO} \quad ,$$

is spatially constant so that the edit quantity $\sigma_g^{ISO}\ \phi_{i,g}$ will be calculated at mesh point i even if the isotope was not physically present at that location.

c. Resident Constituent Cross-Section Response Functions: EDCONS Input Array. Resident constituent cross sections, $\sum_{i,g}^{ISO}$, can be used for response functions by identifying the isotopes desired through the EDCONS BLOCK VI input array. The resident constituent or simply, constituent, cross section is a partial macroscopic cross section given by the product of the isotope microscopic cross section times the actual atom density associated with that isotope at the spatial location as seen by the Solver Module. Thus, for a constituent cross-section edit the response function $R_{i,g}$ is

$$R_{i,g} = N_i^{ISO}\ \sigma_g^{ISO} = \Sigma_{i,g}^{ISO} \quad ,$$

for spatial mesh interval i, group g.

d. Material Cross-Section Response Functions: EDMATS Input Array. Material macroscopic cross sections, $\sum_{g,}^{MATL}$, can be used for the response function by identifying the desired materials through the EDMATS array in the BLOCK VI input. In this material edit the macroscopic cross sections for the materials specified in the MATLS array in the mixing input block (BLOCK IV) are reformed using the microscopic cross sections on file SNXEDT together with the mixing instructions stored on the NDXSRF and ZNATDN standard interface files. Thus, for a material cross-section edit the response functions are of the form

$$R_{i,g} = \Sigma_g^{MATL} \quad .$$

Note that these response functions are spatially constant so that the edit quantity $\sum_g^{MATL}\ \phi_{i,g}$ will be calculated at each mesh point, i, even if the material was not physically present at that location.

2. User-Input Response Functions: The RSFE and RSFX Input Arrays. In addition to response functions based on cross-section data, the user may directly input response functions in a space-energy separable form through the BLOCK VI input arrays $RSFE_g$ and $RSFX_i$ for energy-group g and space-point (interval) i. Thus, for user-input response functions,

$$R_{i,g} = RSFE_g * RSFX_i \quad .$$

NOTE: The RSFE array is required if user-input response functions are desired. The RSFX input array is optional. The RSFE input array can be used, for example, to obtain groupwise fluxes (or sums of groupwise fluxes) by using RSFE array entries of 1.0 in the groups of interest. Fluxes can similarly be renormalized by use of the appropriate normalization factor in either the RSFE or RSFX arrays.

D. Response Function Summing Options

Certain response summing operations are available to the user by means of the BLOCK VI input arrays MICSUM and IRSUMS. The MICSUM array provides for the specification of cross-section response function summing, while the IRSUMS array provides for the specification of user-input response function summing. Each of these is described below.

1. Cross-Section Response Functions Sums: MICSUM Input Array. Through the use of MICSUM input array in BLOCK VI, either isotope microscopic edit sums or resident constituent edit sums, but not both, will be computed. (Recall that isotope microscopic edits are invoked by means of the EDISOS input array and resident constituent edits are invoked by means of the EDCONS input array.)

The MICSUM input array is a packed array with data entered as follows: a set of isotope numbers or Hollerith names (from the basic isotope input library) is given followed by a set of cross-section type position numbers of Hollerith names (see Table Xa, Xb in Ch. V). These sets are delimited with an entry of 0 (zero). Reaction rates (edit quantities) are calculated for each isotope specified in the set for each cross-section type specified and summed to form the first sum. The next two sets of data are used to define the second sum, etc.

The MICSUM array is only used in conjunction with either the EDCONS array or the EDISOS array as follows:

- If the EDCONS array is specified, the summing defined by the MICSUM array applies to the resident constituent (partial macroscopic) cross sections. Isotopes used in the MICSUM array must have been used in the EDCONS array.

- If the EDCONS array is not specified and the EDISOS array is specified, the summing defined by the MICSUM array applies to the isotope microscopic cross sections. Isotopes used in the MICSUM array must have been used in the EDISOS array.

Example: Suppose the EDCONS array were specified as
EDCONS= PU239, PU240, PU241, U238

and the MICSUM array were specified as
MICSUM= PU239, PU241, 0, "N-GAMM," "N-FISS,"
0, PU240, U238, 0, ABS.

For mesh point i, energy group g, the two sums specified in the MICSUM array would be

$$SUM1: \left\{ \left[N_i \left(\sigma^{n,\gamma} + \sigma^f \right)_g \right]^{PU239} + \left[N_i \left(\sigma^{n,\gamma} + \sigma^f \right)_g \right]^{PU241} \right\} * \phi_{i,g}$$

$$SUM2: \left[\left(N_i \sigma_g^a \right)^{PU240} + \left(N_i \sigma_g^a \right)^{U238} \right] * \phi_{i,g} \quad .$$

2. User-Input Response Functions Sums: IRSUMS Input Array. Through the use of the IRSUMS input array in BLOCK VI, user-input response function edit sums can be computed. The input to the IRSUMS array is supplied as follows: a set of user-input response function numbers or names is entered and the set is delimited with an entry of 0 (zero). Edit quantities are calculated for each response function specified and the edit quantities summed to form the first sum. The next set of data is used to form the second sum, etc. Only user-input response functions that have been provided through the RSFE input array (and, optionally, the RSFX input array) can be used in the IRSUMS array.

E. Adjoint Edits

The EDIT module will normally perform regular (forward) edits using regular (forward) scalar fluxes from an RTFLUX standard interface file. If adjoint edits are to be performed, the user need only set the BLOCK VI input parameter AJED to unity and ensure that the appropriate adjoint scalar flux file (ATFLUX file) exists and is available to the EDIT module at the time of execution. (Chapter X.L describes adjoint calculations in the SOLVER module.) In the adjoint mode, the EDIT module performs all adjoint reversals and provides the correct group ordering in the printed output.

F. Edit Module ASCII File Output Capabilities (the EDOUTF Parameter)

Because of the desirability of presenting output quantities from a calculation in graphics form (plots, tables, etc.) it is valuable to have output in eye-readable ASCII files. Such files are easy to use for extracting information and data and putting it in forms usable by graphics packages, text-editors, etc. The Edit Module can optionally create two such ASCII files for the user. The EDTOUT file is an ASCII file which contains certain geometric information (such as problem geometry, number of coarse mesh intervals, the coarse mesh boundary positions, and number of fine mesh intervals per coarse mesh) together with the edit quantities produced by the Edit Module. The EDTOGX file is an ASCII file containing geometric information, group-total fission source (if problem had fissionable material), and, optionally, the multigroup scalar fluxes at each fine mesh point. File descriptions for these two files are provided in Appendix C of this manual.

The Edit Module (BLOCK VI) parameter EDOUTF controls the creation of the EDTOUT and EDTOGX files as described in Ch. V.

XII. STACKED RUNS

It is possible to run more than one problem in a single execution of the code by stacking the problem-specification input "decks." In the context of this discussion the term "deck" refers to all card-image input necessary for a problem. To run more than one problem, the input file is created with two or more problem decks separated from one another by a single card-image record containing the entry

]eof

beginning in column 1. Thus, a single input file containing the specification decks for three separate problems would be constructed as follows:

Problem 1 "deck"
]eof
Problem 2 "deck"
]eof
Problem 3 "deck".

The code output for each of these problems will appear consecutively in a single output file.

Caution: Nonunique interface files created in one problem (for example, an RT-FLUX scalar flux file) will be overwritten and lost when the next problem is executed.

XIII. CONTROLLING THE EXECUTION OF MODULES AND SUBMODULES

As described in Chapter II, the ONEDANT code package is comprised of three major functionally independent modules: the Input Module, the Solver Module, and the Edit Module. The modules are linked solely by means of binary interface files. The Input Module processes any and all card-image input and, if required, generates the binary interface files for use by the Solver and/or Edit Modules. The Input Module itself is constructed in a modular form and thus is comprised of submodules, each of which performs a unique function related to the generation of certain binary interface files. The Solver Module accepts the appropriate interface files produced by the Input Module (or any other computer code capable of producing such interface files), performs the transport calculation, and generates standard interface fiux files for use by the Edit Module (or other computer codes). The Edit Module accepts the appropriate standard and code-dependent interface files and performs cross-section and user-input response function edits.

With the modular construction of the code package and the interface file linkage between modules and submodules, there is a great deal of fiexibility provided in the execution fiow of a particular computer run. For example, the processing of the input, the execution of the transport solution, and the editing of the results of the solution can be effected as three separate and distinct computer runs and not as a single (perhaps expensive) run. All that need be done is to save the appropriate interface files from each partial execution run and to make these files available to the module to be executed in the next partial execution. This mode of operation enables the user, for example, to process his problem input specification (mixing of nuclides, cross-section preparation, geometry specification, etc.,) and to analyze his input before committing it to the Solver Module. If errors are discovered in, say, the geometry specification, the user can correct the errors in the card-image input and simply rerun the geometry-related submodule of the Input Module. When certain that the input is correct, the user can then execute the Solver Module. Following the successful running of the Solver Module, one or more executions of the Edit Module can then be independently made.

In this chapter are provided details for controlling the execution of selected modules and submodules in the ONEDANT code package.

A. Module Execution Control

The execution of each of the three major modules in the ONEDANT code package (Input, Solver, and Edit Modules) can be independently controlled as described below.

1. Input Module Execution Control. The Input Module may be thought of as an interface file generating module. It processes card-image input and creates binary interface files as shown in Table I of Chapter II. Accordingly, if any BLOCK II through BLOCK VI (see Chapter V) card-image input is provided and the BLOCK I input execution suppression flag NOFGEN is zero, the INPUT module will be executed and the appropriate interface files created.

The execution of the Input Module will be suppressed if either of the following conditions is met: (I) the BLOCK I input parameter NOFGEN is set to unity, or (ii) there is no card-image input provided other than BLOCK I input. If the Input Module is not executed, none of its interface files will be created in that execution of ONEDANT.

2. Solver Module Execution Control. Execution of the Solver Module will be attempted if both the following conditions are met: (i) a SOLINP binary interface file exists and is available to the Solver Module, and (ii) the BLOCK I input parameter NOSOLV is zero.

The Solver Module will not be executed if the BLOCK I input parameter NOSOLV is set to unity.

Alternatively, since the Input Module creates the SOLINP interface file solely from card-image input provided in BLOCK V of the input, the user can suppress the execution of the Solver Module by simply omitting all BLOCK V data from the card-image input.

3. Edit Module Execution Control. Execution of the Edit Module will be attempted if both of the following conditions are met: (i) an EDITIT binary interface file exists and is available to the Edit Module and (ii) the BLOCK I input parameter NOEDIT is zero.

The Edit Module will not be executed if the BLOCK I input parameter NOEDIT is set to unity.

Alternatively, since the Input Module creates the EDITIT interface file solely from card-image input provided in BLOCK VI of the input, the user can suppress the execution of the Edit Module by simply omitting all BLOCK VI data from the card-image input.

B. Input Submodule Execution Control (File Generation Suppression)

The Input Module of ONEDANT is constructed in submodular form. Each submodule has a unique interface file-creation function, and each has its associated card-image input. Also associated with each submodule is a BLOCK I input flag to

turn off, or suppress, the execution of that submodule. The control of the execution of the Input Module submodules is described below.

1. Geometry Submodule Execution Control. The geometry submodule creates a GEODST standard interface file[3] from BLOCK II card-image input data as described in Chapter V. This submodule will be executed and a GEODST file created by (i) setting (or defaulting) the BLOCK I input parameters NOGEOD to zero and (ii) providing BLOCK II input data in the card-image input "deck" or file.

The geometry submodule will not be executed (no GEODST file will be created) if (i) the BLOCK I input parameter NOGEOD is set to unity or (ii) all BLOCK II input is omitted from the card-image input "deck."

2. Mixing Submodule Execution Control. The mixing submodule creates the standard interface files NDXSRF and ZNATDN[3] from the BLOCK IV card-image input data found in the MATLS array and, optionally, the PREMIX array as described in Chapter V.

The mixing submodule will be executed and the NDXSRF and ZNATDN files created by both (i) setting (or defaulting) the BLOCK I input parameter NOMIX to zero and (ii) providing card-image input through the MATLS array in BLOCK IV.

The mixing submodule will not be executed in NOMIX is set to unity or (ii) the MATLS input array is omitted from the BLOCK IV card-image input or (iii) LIB= MACRXS or LIB= MACBCD in BLOCK III.

3. Assignment-of-Materials-to-Zones Submodule Execution Control. The assignment-of-materials-to-zones submodule creates the code-dependent interface file ASGMAT from the BLOCK IV card-image data found in the ASSIGN array. Details on the assignment of materials to zones are given in Chapter V.

This submodule will be executed and the ASGMAT file created by both (i) setting (or defaulting) the BLOCK I input parameter NOASG to zero and (ii) providing card-image input through the ASSIGN array in BLOCK IV.

The submodule will not be executed (no ASGMAT file created) if either (i) the BLOCK I input parameter NOASG is set to unity or (ii) the ASSIGN input array is omitted from the BLOCK IV card-image input.

4. Working-Cross-Section-File Submodule Execution Control. The working-cross-section-file submodule creates the code-dependent interface files MACRXS and SNXEDT.

The working-cross-section-file submodule will be executed and the MACRXS and SNXEDT files created if both the following conditions are met: (i) the BLOCK I input parameter NOMACR is set (or defaulted) to zero, and (ii) the BLOCK III input parameter LIB is not specified as LIB= MACRXS or LIB= MACBCD.

The submodule will not be executed (no MACRXS and SNXEDT files created) if either (i) the BLOCK I input parameter NOMACR is set to unity or (ii) the BLOCK III input parameter LIB is specified as LIB= MACRXS or LIB= MACBCD.

Since the formation of the working cross-section files MACRXS and SNXEDT can be quite time-consuming for large multigroup cross-section libraries, it is frequently advantageous to save the MACRXS and SNXEDT files created in one run for use in subsequent runs. Through the use of the NOMACR parameter in BLOCK I or the LIB= MACRXS parameter in BLOCK III of the input, the user can easily suppress the re-execution of the working-cross-section-file submodule in subsequent code executions.

5. SOLVER-Input-File Submodule Execution Control. The Solver-input-file submodule processes the BLOCK V card-image input and creates the code-dependent interface file SOLINP for use by the Solver Module.

This submodule will be executed and the SOLINP file created if both (i) the BLOCK I input parameter NOSLNP is set (or defaulted) to zero and (ii) BLOCK V card-image input is supplied.

The Solver-input-file submodule will not be executed (no SOLINP file created) if either (i) the BLOCK I input parameter NOSLNP is set to unity or (ii) all BLOCK V card-image input is omitted from the input "deck."

6. Edit-input-File Submodule Execution Control. The EDIT-input-file submodule processes the BLOCK VI card-image input and creates the code-dependent interface file EDITIT for use by the EDIT module of ONEDANT.

The EDIT-input-file submodule will be executed and the EDITIT file created if both (i) the BLOCK I input parameter NOEDTT is set (or defaulted) to zero and (ii) BLOCK VI card-image input is supplied.

This submodule will not be executed (no EDITIT file created) if either (i) the BLOCK I input parameter NOEDTT is set to unity or (ii) all BLOCK VI card-image input is omitted from the input "deck."

7. Adjoint-Reversal Submodule Execution Control. The adjoint-reversal submodule processes the MACRXS code-dependent cross-section interface file and creates the code-dependent interface file ADJMAC, the adjoint-reversed counterpart to the MACRXS file. This is described in Chapter X.L.

The adjoint-reversal submodule will be executed if both (i) the BLOCK I input parameter NOADJM is set (or defaulted) to zero and (ii) the BLOCK V input quantity ITH (found in the collective array SOLIN) is set to unity.

The submodule will not be executed (no ADJMAC file created) if either (i) the BLOCK I input parameter NOADJM is set to unity or (ii) the BLOCK V input quantity ITH (found in the collective array SOLIN) is set to zero.

XIV. ERROR DIAGNOSTICS

A comprehensive error-checking capability has been provided in the ONEDANT code package. Most of the checks are in the Input Module to ensure that the input data are correct, insofar as the code can determine, before execution of the problem commences. Other checks are made in the Solver and Edit Modules to ensure that the modules are executing the desired problem properly.

One important feature of the error diagnostics in the Input Module is that an error will normally not cause an immediate termination of execution. Instead, the code will attempt to process remaining data in the offending input BLOCK and/or in remaining input BLOCKS. Once all remaining input has been processed (if possible) the run will be terminated.

Error messages are normally provided in at least two places in the output. The first error message is printed at the time that the error was detected by the code. Such messages will be imbedded in the printed output, but they are clearly marked for easy spotting. The second error message will normally occur in the RUN HIGHLIGHTS provided at the end of the printed output. These RUN HIGHLIGHTS provide a printed summary of the code package execution. The user is encouraged to always check the RUN HIGHLIGHTS following a run to quickly ascertain if the completed run did what it was supposed to.

A. Examples of Errors and Resulting Messages

Several examples of common input errors and the resulting error message printouts are provided below.

Sample Error 1. Misspelled Input Array Name

A common input error is that of misspelling the name of an input array. In this example, the BLOCK II card-image input array XMESH has been misspelled as XMESSH. The Input Module is thus presented with an unrecognizable and undefined array name resulting in the following error message:

```
*error* card        7 *2 4 6 8(1)2 4 6 8(2)2 4 6 8(3)2 4 6 8(4)2
                        xmessh=0.0  5.0  xints=5  zones=1    t


*error* array name xmessh array number      -1

*error* columns  2 -  8


undefined array name
```

The first line of the error message indicates that an error was found on input card 7. This is followed by the card-image column numbers. Directly below this, the card-image is reproduced. The third line indicates that the array name XMESSH is in error and that this array has been given a number -1. (Acceptable arrays are given positive integer identification numbers by the code.) The next line says that the error occurred in columns 2 through 8 on the card-image. Finally, the message that the array name is undefined is provided.

The RUN HIGHLIGHTS indicate that the GEODST interface file has not been created due to the error in the array name. Note, however, that the HIGHLIGHTS indicate that all remaining input data have been processed satisfactorily following the detection of the error and all other interface files were properly created before the run was terminated.

```
                    run highlights


*****************************************
*                                       *
*    all modules are tentatively go.    *
* **unable to write good geodst file**  *
* cross sections from cards via bxslib. *
*    interface mixing files written.    *
*     interface file asgmat written     *
*    xs files macrxs.snxedt written.    *
*  solver module execution suppressed.  *
* neither solinp nor solver cards exist *
*   edit module execution suppressed.   *
*  neither editit nor edit cards exist. *
*         ** fatal input errors **      *
*                                       *
*****************************************
```

Sample Error 2. Input Block Terminator Omitted

As described in Chapter IV.A., each card-image input BLOCK must be terminated with a delimited T, the BLOCK terminator. In this example this terminal "T" has been omitted from the end of the BLOCK I card-image input. The offending portion actual card-image input for this case is shown below.

```
*****************************************************************************************
*                                                                                       *
*                    ...listing of cards in the input stream...                         *
*                                                                                       *
*    1.      2           0                                                              *
*    2.  Sample ONEDANT input to display error messages                                 *
*    3.  Error 2  Terminal "t" omitted from Block I                                     *
*    4. /*************************B L O C K  I  ***********************************     *
*    5.  igeom=slab ngroup=30 isn=8 niso=16 mt=1 nzone=1 im=1 it=5                      *
*    6. /************************* B L O C K  II ***************************************  *
*    7.  xmesh=0.0  5.0  xints=5  zones=1    t                                          *
*    8. /************************** B L O C K III ***************************************  *
*    9.  lib= bxslib      t                                                             *
*   10. /************************** B L O C K  IV ***************************************  *
*   11.  matls=matt 92235.50    assign=matls     t                                     *
*                                                                                       *
*****************************************************************************************
```

Note that there is no delimited T at the end of line 5. As a result of this omission the following message is printed:

```
**********************
*key end   block 1 read*
**********************



**************************************************************************

***error** current block contains arrays belonging to other blocks

**************************************************************************



*
*  no. of    from
*  arrays   block
*
*     8        1
*     3        11
*     0        111
*     0        iv
*     0         v
*     0        vi



**************************************************************************

***error** in block identification

**************************************************************************
```

The first line of the message indicates that the Input Module has finished reading the card-image input that it thinks belongs in BLOCK I. (The code has actually read the 8 array entries on line 5 of the input but has continued reading until it found the terminal T following the three array entries belonging to BLOCK II on line 7 of the input.) The second line of the error message indicates that arrays that do not belong in BLOCK I have been found. Next is printed a table indicating that eight arrays from BLOCK I and three from BLOCK II were discovered in the BLOCK I card-image reading process. The final error message of an error in BLOCK identification is self-explanatory.

It should be noted that when arrays from other BLOCKS are found in any given BLOCK, the code will terminate execution immediately and no RUN HIGHLIGHTS are provided.

Sample Error 3. Invalid Entry for BLOCK I Input Parameter.

As described in Chapter IV, certain card-image input is always required in BLOCK I for a ONEDANT code execution. Specifically, the eight parameters in the collective array DIMENS, (IGEOM, NGROUP, ISN, NISO, MT, NZONE, IM, and IT) are required to be entered as positive integers. In this example, one of these parameters, IM, has been incorrectly entered with a value of zero. It should be noted that if one of these parameters is omitted altogether, the code will default its value to zero.

The code prints the following fatal error message:

```
**********************************************************************

***error** block 1 entry .le. zero

**********************************************************************



 * im
```

The message is self-explanatory. No RUN HIGHLIGHTS are provided when BLOCK I input errors are encountered.

Sample Error 4. Incorrect Number of Entries in an Input Array.

Several input arrays available as input to ONEDANT require a predetermined number of entries. In this example the XINTS array in the BLOCK II card-image input was provided with only four entries instead of the five it should have had. The error message provided by ONEDANT is shown below.

```
**********************************************************************

***error** fine mesh specs.ne.it

**********************************************************************
```

The error message is self-explanatory.

In this case the remaining blocks of input data were successfully processed and the RUN HIGHLIGHTS are provided as shown below.

```
****************************************
*                                      *
*    all modules are tentatively go.   *
* **unable to write good geodst file** *
* cross sections from cards via bxslib.*
*    interface mixing files written.   *
*     interface file asgmat written    *
*    xs files macrxs,snxedt written.   *
*  solver module execution suppressed. *
* neither solinp nor solver cards exist*
*   edit module execution suppressed.  *
*  neither editit nor edit cards exist.*
*         ** fatal input errors **      *
*                                      *
****************************************
```

Note that the fatal XINTS error prevented the code from creating the necessary GEODST interface file. The run was thus terminated after the remaining input data were processed.

Sample Error 5. Misplaced Array Identifier.

As discussed in Chapters IV.A.5., IV.C.5, and V.A., arrays in the card-image input are identified by a Hollerith name or a number immediately followed by an array identifier - an equals (=) sign, a dollar ($) sign, or an asterisk (*). The array identifier is required so that the code can recognize that the array name or number is indeed an array name or number and not an ordinary data item. In this sample a blank was inadvertently placed between the array name (ASSIGN) and the array identifier (=). The resulting message is shown below.

The first line of the message indicates that an error was found on input card 11 and is followed by the card-image column numbers. Directly below this, the card-image is reproduced. The third line indicates that the error was detected in column 32 of the card-image and the fourth line provides the self-explanatory message that a blank was found preceding the array identifier.

```
*error* card       11 *2 4 6 8(1)2 4 6 8(2)2 4 6 8(3)2 4 6 8(4)2 4 6 8
                      matls=matt 92235.50    assign *matls     t

*error* columns 32 - 32


**error**blank preceding an array identifier(=, $, or *)
```

B. Comments Regarding Multiple Errors

As a result of the Input Module's attempt to continue processing card-image input after a fatal error has been detected, it is possible for multiple errors to be diagnosed and for multiple error messages to be printed.

When multiple error messages are printed, the user should check to see if one or more of the errors was due to a preceding error. In other words, a particular input error may cause a chain reaction of other errors. For example, suppose that the entry IT were inadvertently omitted from the BLOCK I input. The code will thus record a value of IT= 0. An otherwise correct entry for the XINTS array in BLOCK II, however, will now appear incorrect to the code since the code checks to see if

$$\sum_{I=1}^{IM} XINTS(I) = IT \quad ,$$

and a message to the effect that the fine-mesh specifications (XINTS array) is not equal to IT will be printed. The user is thus advised to review multiple error messages starting with the first message printed in order to determine which errors are independent of other errors and which are results of a preceding error.

APPENDIX A

CODE-DEPENDENT INTERFACE FILE DESCRIPTIONS

ONEDANT makes use of interface files to transmit data between and within its modules. These interface files are binary, sequential data files. Standard interface files are interface files whose structure and data-content formats have been standardized by the CCCC. Code-dependent interface files are files whose structure and data-content formats have not been standardized.

The following CCCC standard interface files are accepted, created, or otherwise used in ONEDANT: ISOTXS, GRUPXS, GEODST, NDXSRF, ZNATDN, SNCONS, FIXSRC, RTFLUX, ATFLUX, RAFLUX, AAFLUX and RZFLUX. File descriptions for these files are provided in Ref. 3.

The following code-dependent binary interface files are used in ONEDANT: MACRXS, SNXEDT, ADJMAC, ASGMAT, SOLINP, and EDITIT. An additional code-dependent binary file, the BXSLIB file, is also optionally used by ONEDANT, although not as an interface file in the strict sense. Other code-dependent files provided as output from ONEDANT are the RMFLUX, RZMFLX, and FISSRC files. In this appendix are provided the file descriptions for these code-dependent files. The file descriptions follow the format used for the standard interface file descriptions in Ref. 3.

I. MACRXS File

The MACRXS code-dependent interface file is the working cross-section file for the Solver Module. On the MACRXS file are the material macroscopic cross sections arranged in energy-group order. The contents of this file are described below:

```
C*****************************************************************
C                        OATE 05/12/83                           -
C                                                                -
CF          MACRXS                                               -
CE          CODE DEPENOENT MACROSCOPIC MULTIGROUP CROSS SECTION FILE -
CE          FOR USE IN ONEDANT SOLVER MOOULE                     -
C                                                                -
C                                                                -
C*****************************************************************
C                                                                -
CN               THIS FILE PROVIOES A BASIC BROAO GROUP          -
CN               LIBRARY. ORDERED BY GROUP                       -
C                                                                -
C                ORDER OF GROUPS IS ACCORDING TO DECREASING ENERGY -
C                                                                -
C----------------------------------------------------------------
CS          FILE STRUCTURE                                       -
CS                                                               -
CS             RECORD TYPE                          PRESENT IF   -
CS             ****************************      ************** -
CS             FILE IDENTIFICATION                  ALWAYS       -
CS             FILE CONTROL                         ALWAYS       -
CS             FILE OATA                            ALWAYS       -
CS                                                               -
CS   ************(REPEAT FOR ALL GROUPS)                        -
CS   *         PRINCIPAL CROSS SECTIONS            ALWAYS       -
CS   *         SCATTERING CONTROL DATA             NORD.NE.O    -
CS   *         SCATTERING MATRIX                   NORD.NE.O    -
CS   ************                                               -
C                                                                -
C----------------------------------------------------------------
C                                                                -
C                                                                -
C----------------------------------------------------------------
CR          FILE IDENTIFICATION                                  -
C                                                                -
CL    HNAME.(HUSE(I).I=1.2).IVERS                                -
C                                                                -
CW    1+3*MULT*NUMBER OF WORDS                                   -
C                                                                -
CD    HNAME          HOLLERITH FILE NAME - MACRXS - (A6)         -
CO    HUSE(I)        HOLLERITH USER IDENTIFICATIDN (A6)          -
CD    IVERS          FILE VERSION NUMBER                         -
CO    MULT           DOUBLE PRECISION PARAMETER                  -
CO                       1- A6 WORD IS SINGLE WORD               -
CD                       2- A6 WORD IS DOUBLE PRECISION WORD     -
C                                                                -
C----------------------------------------------------------------
C----------------------------------------------------------------
CR          FILE CONTROL                                         -
C                                                                -
CL    NGROUP.NMAT.NORD.NED.IDPF.LNG.MAXUP.MAXDN.NPRIN.I2LP1      -
C                                                                -
CW    10*NUMBER OF WORDS                                         -
C                                                                -
CD    NGROUP         NUMBER OF ENERGY GROUPS IN FILE             -
CD    NMAT           NUMBER OF MATERIALS IN FILE                 -
CO    NORD           NUMBER OF LEGENDRE SCATTERING ORDERS        -
CD    NED            NUMBER OF EXTRA EDIT CROSS SECTIONS (IN ADDITION -
CO                   TO THE BASIC PRINCIPAL CROSS SECTIONS)      -
CD    IDPF           O/1 ND/YES CROSS SECTIDN DATA ARE DOUBLE PRECISION -
CD    LNG            NUMBER OF THE LAST NEUTRON GROUP(FOR COUPLED SETS) -
CD    MAXUP          MAXIMUM NUMBER OF UPSCATTER GROUPS          -
CO    MAXDN          MAXIMUM NUMBER OF OOWNSCATTER GRDUPS        -
CD    NPRIN          NUMBER OF PRINCIPAL CROSS SECTIONS          -
CD    I2LP1          O/1 = NO/YES  2L+1 TERM WAS INCLUDED IN LIBRARY -
C                                                                -
C----------------------------------------------------------------
```

```
C-----------------------------------------------------------------------
CR          FILE DATA                                                  -
C                                                                      -
CL    (HMAT(I),I=1,NMAT),(HED(J),J=1,NEDT),(VEL(N),N=1,NGROUP),        -
CL   1(EMAX(N),N=1,NGROUP),EMIN                                        -
C                                                                      -
CW    NMAT+NEDT+2*NGROUP+1=NUMBER OF WORDS                             -
C                                                                      -
CO    HMAT(I)        HOLLERITH MATERIAL LABEL FOR MATERIAL I (A6)      -
CD    HED(J)         HOLLERITH LABEL FOR J-TH CROSS SECTION POSITION(A6)-
CD    VEL(N)         MEAN NEUTRON VELOCITY IN GROUP N (CM/SEC)         -
CD    EMAX(N)        MAXIMUM ENERGY BOUND OF GROUP N (EV)              -
CD    EMIN           MINIMUM ENERGY BOUND OF SET (EV)                  -
CD    NEDT           NED+NPRIN                                         -
C                                                                      -
CN                   THE FOUR BASIC PRINCIPAL CROSS SECTIONS           -
CN                   ALWAYS PRESENT ARE:                               -
CN                                                                     -
CN                         HED(1) = 3HCHI                              -
CN                         HED(2) = 6HNUSIGF                           -
CN                         HED(3) = 5HTOTAL                            -
CN                         HED(4) = 3HABS                              -
C                                                                      -
CN                   ALSO PRESENT WHEN NPRIN=5 IS:                     -
C                                                                      -
CN                         HED(5) = 5HTRANS                            -
C                                                                      -
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
CR          PRINCIPAL CROSS SECTIONS FOR GROUP N                       -
C                                                                      -
CL    ((C(I,J),I=1,NMAT),J=1,NEDT)                                     -
C                                                                      -
CW    NMAT*NEDT=NUMBER OF WORDS                                        -
C                                                                      -
CO    C(I,J)         PRINCIPAL CROSS SECTIONS                          -
C                                                                      -
CN                   BASIC PRINCIPAL CROSS SECTIONS ALWAYS PRESENT ARE: -
CN                                                                     -
CN                        J=1  FISSION SPECTRUM                        -
CN                        J=2  FISSION NU*FISSION CROSS SECTION        -
CN                        J=3  TOTAL CROSS SECTION                     -
CN                        J=4  ABSORPTION CROSS SECTION                -
C                                                                      -
CN                   ALSO PRESENT WHEN NPRIN=5 IS:                     -
C                                                                      -
CN                        J=5  TRANSPORT CROSS SECTION                 -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          SCATTERING CONTROL BLOCK FOR GROUP N                       -
C                                                                      -
CC        PRESENT IF NORD.GT.0                                         -
C                                                                      -
CL    ((NGPB(L,J),L=1,NORD),J=1,NMAT)                                  -
CL    ((IFSG(L,J),L=1,NORD),J=1,NMAT)                                  -
C                                                                      -
CW    2*NORD*NMAT=NUMBER OF WORDS                                      -
C                                                                      -
CO    NGPB(L,J)    NUMBER OF SOURCE GROUPS THAT CAN SCATTER INTO GROUP N-
CD    IFSG(L,J)    GROUP NUMBER OF THE FIRST SOURCE GROUP             -
CD    L            LEGENDRE ORDER NUMBER                             -
CD    J            MATERIAL NUMBER                                   -
C                                                                      -
C-----------------------------------------------------------------------
```

```
C-------------------------------------------------------------------
CR          SCATTERING SUB-BLOCK FOR GROUP N                       -
C                                                                  -
CC        PRESENT IF NDRD.GT.O                                     -
C                                                                  -
CL    (SCAT(I),I=1,NTAB)                                           -
C                                                                  -
CW    NTAB=NUMBER OF WORDS                                         -
C                                                                  -
CD    SCAT(I)        SCATTERING CROSS SECTION                      -
C                                                                  -
CD    NTAB           TABLE LENGTH OF THE CROSS SECTIONS FOR SCATTERING -
CD                   INTO GROUP N.  THIS IS FOR ALL MATERIALS AND ALL -
CD                   LEGENDRE ORDERS. THUS IT IS THE SUM OF NGPB(L,J) -
CD                   OVER L FROM 1 TO NDRD AND OVER J FROM 1 TO NMAT. -
C                                                                  -
CN                   THE SCATTERING CROSS SECTIONS ARE PACKED IN BANDS. -
CN                   ONE FOR EACH LEGENDRE ORDER AND MATERIAL. EACH BAND-
CN                   CONTAINS THE NGPB GROUPS WHICH SCATTER INTO GROUP -
CN                   N.  THE FIRST SOURCE GROUP NUMBER IS IFSG AND    -
CN                   THE LAST IS IFSG-NGPB+1. THE NORD BANDS FOR THE  -
CN                   FIRST MATERIAL APPEAR FIRST (P0, P1, ....) FOLLOWED-
CN                   BY THE NORD BANDS FOR THE SECOND, ETC.           -
C                                                                  -
CN                   HIGHER LEGENDRE ORDER SCATTERING CROSS SECTIONS -
CN                   INCLUDE A 2*L+1 FACTOR WHERE L IS THE LEGENDRE -
CN                   ORDER.                                         -
C                                                                  -
C-------------------------------------------------------------------
CEOF                                                               -
```

II. SNXEDT File

The SNXEDT file is the working cross-section file for the Edit Module. On the SNXEDT file are the isotope microscopic cross sections arranged in energy-group order. Although the Edit Module will read any SNXEDT file constructed as described below, all SNXEDT files created by the Input Module will have the parameter NORD set to zero so that scattering cross sections will not appear on the created SNXEDT file.

```
C*****************************************************************
C                          DATE 05/12/83                         -
C                                                                -
CF          SNXEDT                                               -
CE          CODE DEPENDENT MICROSCOPIC MULTIGRDUP CRDSS SECTIDN FILE -
CE          FDR USE IN ONEDANT EDITS                             -
C                                                                -
C*****************************************************************
C                                                                -
C                                                                -
CN                THIS FILE PROVIDES A BASIC BROAD GROUP         -
CN                LIBRARY, ORDERED BY GROUP                      -
C                                                                -
C                                                                -
C-----------------------------------------------------------------
CS          FILE STRUCTURE                                       -
CS                                                               -
CS             RECORD TYPE                        PRESENT IF     -
CS             ***********                        **********     -
CS             FILE IDENTIFICATIDN                ALWAYS         -
CS             FILE CDNTRDL                       ALWAYS         -
CS             FILE DATA                          ALWAYS         -
CS                                                               -
CS    ************(REPEAT FOR ALL GROUPS)                       -
CS    *              (GRDUP 1 IS FIRST)                         -
CS    *           PRINCIPAL CROSS SECTIDNS        ALWAYS         -
CS    *           SCATTERING CONTROL DATA         NORD.NE.0      -
CS    *           SCATTERING MATRIX               NORD.NE.0      -
CS    ************                                               -
C                                                                -
C-----------------------------------------------------------------
C                                                                -
C                                                                -
C-----------------------------------------------------------------
CR          FILE IDENTIFICATION                                  -
C                                                                -
CL     HNAME,(HUSE(I),I=1,2),IVERS                               -
C                                                                -
CW     1+3*MULT=NUMBER OF WORDS                                  -
C                                                                -
CD     HNAME          HOLLERITH FILE NAME - SNXEDT - (A6)        -
CD     HUSE(I)        HOLLERITH USER IDENTIFICATION (A6)         -
CD     IVERS          FILE VERSION NUMBER                        -
CD     MULT           DOUBLE PRECISIDN PARAMETER                 -
CD                        1- A6 WORD IS SINGLE WORD              -
CD                        2- A6 WORD IS DOUBLE PRECISION WORD    -
C                                                                -
C-----------------------------------------------------------------
```

```
C-----------------------------------------------------------------------
CR          FILE CONTROL                                               -
C                                                                      -
CL    NGROUP,NISO,NORD,NED,IDPF,LNG,MAXUP,MAXON,NPRIN,I2LP1            -
C                                                                      -
CW    10*NUMBER OF WORDS                                               -
C                                                                      -
CO    NGROUP         NUMBER OF ENERGY GROUPS IN FILE                   -
CD    NISO           NUMBER OF ISOTOPES IN FILE                        -
CO    NORD           NUMBER OF LEGENDRE SCATTERING ORDERS              -
CD    NED            NUMBER OF EXTRA EDIT CROSS SECTIONS (IN ADDITION  -
CD                   TO THE BASIC PRINCIPAL CROSS SECTIONS)            -
CD    IOPF           0/1 NO/YES CROSS SECTION DATA ARE DOUBLE PRECISION -
CD    LNG            NUMBER OF THE LAST NEUTRON GRDUP(FOR COUPLED SETS) -
CD    MAXUP          MAXIMUM NUMBER OF UPSCATTER GRDUPS                -
CD    MAXDN          MAXIMUM NUMBER OF DOWNSCATTER GROUPS              -
CD    NPRIN          NUMBER DF PRINCIPAL CRDSS SECTIDNS (4 FDR SNXEDT) -
CD    I2LP1          0/1 = NO/YES  2L+1 TERM WAS INCLUDED IN LIBRARY   -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C                                                                      -
C-----------------------------------------------------------------------
CR          FILE DATA                                                  -
C                                                                      -
CL    (HISO(I),I=1,NISO),(HED(J),J=1,NEDT),(VEL(N),N=1,NGROUP),        -
CL   1(EMAX(N),N=1,NGROUP),EMIN                                        -
C                                                                      -
CW    NISO+NEDT+2*NGROUP+1=NUMBER OF WORDS                             -
C                                                                      -
CD    HISD(I)        HOLLERITH ISOTDPE LABEL FOR ISDTOPE (A6)          -
CD    HED(J)         HOLLERITH LABEL FDR EDIT NUMBER J (A6)            -
CO    VEL(N)         MEAN NEUTRON VELOCITY IN GROUP N (CM/SEC)         -
CD    EMAX(N)        MAXIMUM ENERGY BOUND OF GROUP N (EV)              -
CD    EMIN           MINIMUM ENERGY BOUND OF SET (EV)                  -
CD    NEDT           NED+NPRIN                                         -
C                                                                      -
CN                   THE FOUR BASIC PRINCIPAL CROSS SECTIONS           -
CN                   ALWAYS PRESENT ARE:                               -
CN                                                                     -
CN                         HED(1) = 3HCHI                              -
CN                         HED(2) = 6HNUSIGF                           -
CN                         HED(3) = 5HTOTAL                            -
CN                         HED(4) = 3HABS                              -
C                                                                      -
CN                   ALSO PRESENT WHEN NPRIN=5 IS:                     -
C                                                                      -
CN                         HED(5) = 5HTRANS                            -
C                                                                      -
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
CR          PRINCIPAL CROSS SECTIONS FOR GROUP N                       -
C                                                                      -
CL    ((C(I,J),I=1,NISD),J=1,NEDT)                                     -
C                                                                      -
CW    NISO*NEDT=NUMBER OF WORDS                                        -
C                                                                      -
CD    C(I,J)         PRINCIPAL CROSS SECTIONS                          -
C                                                                      -
CN                   BASIC PRINCIPAL CROSS SECTIONS ALWAYS PRESENT ARE: -
CN                                                                     -
CN                        J=1  FISSION SPECTRUM                        -
CN                        J=2  FISSION NU*FISSION CROSS SECTION        -
CN                        J=3  TOTAL CROSS SECTION                     -
CN                        J=4  ABSORPTION CRDSS SECTIDN                -
C                                                                      -
CN                   ALSO PRESENT WHEN NPRIN=5 IS:                     -
C                                                                      -
CN                        J=5  TRANSPORT CRDSS SECTION                 -
C                                                                      -
C-----------------------------------------------------------------------
```

```
C-------------------------------------------------------------------
CR          SCATTERING CONTROL BLOCK FOR GROUP N                    -
C                                                                   -
CC        PRESENT IF NORO.GT.O                                      -
C                                                                   -
CL    ((NGPB(L,J),L=1,NORD),J=1,NISO)                               -
CL    ((IFSG(L,J),L=1,NORO),J=1,NISO)                               -
C                                                                   -
CW    2*NORD*NISO=NUMBER OF WORDS                                   -
C                                                                   -
CD    NGPB(L,J)   NUMBER OF SOURCE GROUPS THAT CAN SCATTER INTO GROUP N-
CD    IFSG(L,J)   GROUP NUMBER OF THE FIRST SOURCE GROUP            -
CD    L           LEGENDRE ORDER NUMBER                             -
CD    J           ISDTOPE NUMBER                                    -
C                                                                   -
C-------------------------------------------------------------------
C-------------------------------------------------------------------
CR          SCATTERING SUB-BLOCK FOR GRDUP N                        -
C                                                                   -
CC        PRESENT IF NORD.GT.O                                      -
C                                                                   -
CL    (SCAT(I),I=1,NTAB)                                            -
C                                                                   -
CW    NTAB=NUMBER OF WORDS                                          -
C                                                                   -
CD    SCAT(I)       SCATTERING CROSS SECTION                        -
C                                                                   -
CD    NTAB          TABLE LENGTH OF THE CROSS SECTIONS FOR SCATTERING -
CD                  INTO GRDUP N.  THIS IS FOR ALL ISDTDPES AND ALL -
CD                  LEGENDRE ORDERS, THUS IT IS THE SUM DF NGPB(L,J) -
CD                  OVER L FROM 1 TO NORD AND OVER J FROM 1 TO NISO. -
C                                                                   -
CN                  THE SCATTERING CROSS SECTIONS ARE PACKED IN BANDS. -
CN                  ONE FOR EACH LEGENORE ORDER AND ISOTOPE. EACH BAND -
CN                  CONTAINS THE NGPB GROUPS WHICH SCATTER INTO GROUP -
CN                  N.  THE FIRST SOURCE GROUP NUMBER IS IFSG AND   -
CN                  THE LAST IS IFSG-NGPB+1. THE NORD BANDS FOR THE -
CN                  FIRST ISOTOPE APPEAR FIRST (PO, P1, ....) FOLLOWED -
CN                  BY THE NORD BANDS FOR THE SECOND. ETC.          -
C                                                                   -
CN                  HIGHER LEGENDRE ORDER SCATTERING CROSS SECTIONS  -
CN                  INCLUDE A 2*L+1 FACTOR WHERE L IS THE LEGENDRE  -
CN                  ORDER.                                          -
C                                                                   -
C-------------------------------------------------------------------
CEDF                                                                -
```

III. ADJMAC File

The ADJMAC file is the adjoint-reversed counterpart to the MACRXS interface file.

```
C*****************************************************************-
C                          DATE 05/12/83                          -
C                                                                 -
CF          ADJMAC                                                -
CE          CODE OEPENDENT MACROSCOPIC MULTIGROUP CROSS SECTION FILE -
CE          USED IN ONEDANT SOLVER MODULE FOR ADJOINT CALCULATIONS -
C                                                                 -
C                                                                 -
C*****************************************************************-
C                                                                 -
CN                THIS FILE PROVIDES A BASIC BRDAD GROUP          -
CN                LIBRARY, ORDERED BY GROUP                       -
C                                                                 -
C                 DRDER OF GRDUPS IS ACCORDING TD INCREASING ENERGY -
C                                                                 -
C-----------------------------------------------------------------
CS          FILE STRUCTURE                                        -
CS                                                                -
CS             RECORD TYPE                          PRESENT IF    -
CS             ****************************      ************** -
CS             FILE IDENTIFICATION                  ALWAYS        -
CS             FILE CONTROL                         ALWAYS        -
CS             FILE DATA                            ALWAYS        -
CS                                                                -
CS   ************(REPEAT FOR ALL GROUPS)                         -
CS   *         PRINCIPAL CROSS SECTIONS             ALWAYS        -
CS   *         SCATTERING CONTROL DATA              NORD.NE.0     -
CS   *         SCATTERING MATRIX                    NORD.NE.0     -
CS   ************                                                -
C                                                                 -
C-----------------------------------------------------------------
C                                                                 -
C                                                                 -
C-----------------------------------------------------------------
CR          FILE IDENTIFICATION                                   -
C                                                                 -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                 -
C                                                                 -
CW    1+3=MULT=NUMBER OF WORDS                                    -
C                                                                 -
CD    HNAME          HDLLERITH FILE NAME - ADJMAC - (A6)          -
CD    HUSE(I)        HOLLERITH USER IDENTIFICATION (A6)           -
CD    IVERS          FILE VERSION NUMBER                          -
CO    MULT           OOUBLE PRECISION PARAMETER                   -
CD                       1- A6 WORD IS SINGLE WORO                -
CD                       2- A6 WORD IS DOUBLE PRECISIDN WORD      -
C                                                                 -
C-----------------------------------------------------------------
C-----------------------------------------------------------------
CR          FILE CONTROL                                          -
C                                                                 -
CL    NGROUP,NMAT,NDRD,NED,IDPF,LNG,MAXUP,MAXDN,NPRIN,I2LP1       -
C                                                                 -
CW    10=NUMBER OF WDRDS                                          -
C                                                                 -
CD    NGROUP         NUMBER OF ENERGY GROUPS IN FILE              -
CD    NMAT           NUMBER OF MATERIALS IN FILE                  -
CD    NORD           NUMBER OF LEGENDRE SCATTERING ORDERS         -
CD    NED            NUMBER OF EXTRA EDIT CROSS SECTIONS (IN ADDITION -
CD                   TO THE BASIC PRINCIPAL CROSS SECTIONS)       -
CD    IDPF           O/1 NO/YES CROSS SECTION DATA ARE DOUBLE PRECISION -
CO    LNG            NUMBER OF THE LAST NEUTRON GROUP(FOR COUPLED SETS) -
CD    MAXUP          MAXIMUM NUMBER OF UPSCATTER GROUPS           -
CD    MAXDN          MAXIMUM NUMBER OF DDWNSCATTER GROUPS         -
CD    NPRIN          NUMBER OF PRINCIPAL CROSS SECTIONS           -
CD    I2LP1          O/1 = NO/YES  2L+1 TERM WAS INCLUDED IN LIBRARY -
C                                                                 -
C-----------------------------------------------------------------
```

```
C-----------------------------------------------------------------------
CR          FILE DATA                                                  -
C                                                                      -
CL    (HMAT(I),I=1,NMAT),(HED(J),J=1,NEDT),(VEL(N),N=1,NGROUP),        -
CL   1(EMAX(N),N=1,NGROUP),EMIN                                        -
C                                                                      -
CW    NMAT+NEDT+2*NGROUP+1=NUMBER OF WORDS                             -
C                                                                      -
CD    HMAT(I)        HOLLERITH MATERIAL LABEL FOR MATERIAL I (A6)      -
CD    HED(J)         HOLLERITH LABEL FOR J-TH CROSS SECTION POSITION(A6)-
CD    VEL(N)         MEAN NEUTRON VELOCITY IN GROUP N (CM/SEC)         -
CD    EMAX(N)        MAXIMUM ENERGY BOUND OF GROUP N (EV)              -
CD    EMIN           MINIMUM ENERGY BOUND OF SET (EV)                  -
CD    NEDT           NED+NPRIN                                         -
C                                                                      -
CN                   THE FOUR BASIC PRINCIPAL CROSS SECTIONS           -
CN                   ALWAYS PRESENT ARE:                               -
CN                                                                     -
CN                         HED(1) = 3HCHI                              -
CN                         HED(2) = 6HNUSIGF                           -
CN                         HED(3) = 5HTOTAL                            -
CN                         HED(4) = 3HABS                              -
C                                                                      -
CN                   ALSO PRESENT WHEN NPRIN=5 IS:                     -
C                                                                      -
CN                         HED(5) = 5HTRANS                            -
C                                                                      -
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
CR          PRINCIPAL CROSS SECTIONS FOR GROUP N                       -
C                                                                      -
CL    ((C(I,J),I=1,NMAT),J=1,NEOT)                                     -
C                                                                      -
CW    NMAT=NEDT=NUMBER OF WORDS                                        -
C                                                                      -
CD    C(I,J)         PRINCIPAL CRDSS SECTIONS                          -
C                                                                      -
CN                   BASIC PRINCIPAL CROSS SECTIONS ALWAYS PRESENT ARE: -
CN                                                                     -
CN                        J=1  FISSION SPECTRUM                        -
CN                        J=2  FISSION NU=FISSION CROSS SECTION        -
CN                        J=3  TOTAL CROSS SECTION                     -
CN                        J=4  ABSORPTION CROSS SECTION                -
C                                                                      -
CN                   ALSO PRESENT WHEN NPRIN=5 IS:                     -
C                                                                      -
CN                        J=5  TRANSPORT CROSS SECTION                 -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          SCATTERING CONTROL BLOCK FOR GROUP N                       -
C                                                                      -
CC        PRESENT IF NORD.GT.O                                         -
C                                                                      -
CL    ((NGPB(L,J),L=1,NORD),J=1,NMAT)                                  -
CL    ((IFSG(L,J),L=1,NORD),J=1,NMAT)                                  -
C                                                                      -
CW    2=NORD=NMAT=NUMBER OF WORDS                                      -
C                                                                      -
CD    NGPB(L,J)    NUMBER OF SOURCE GROUPS THAT CAN SCATTER INTO GROUP N-
CD    IFSG(L,J)    GROUP NUMBER OF THE FIRST SOURCE GRDUP             -
CD    L            LEGENDRE ORDER NUMBER                             -
CD    J            MATERIAL NUMBER                                   -
C                                                                      -
C-----------------------------------------------------------------------
```

```
C-------------------------------------------------------------------
CR          SCATTERING SUB-BLOCK FOR GROUP N                       -
C                                                                  -
CC        PRESENT IF NORD.GT.O                                     -
C                                                                  -
CL    (SCAT(I),I=1,NTAB)                                           -
C                                                                  -
CW    NTAB=NUMBER OF WOROS                                         -
C                                                                  -
CD    SCAT(I)       SCATTERING CROSS SECTION                       -
C                                                                  -
CD    NTAB          TABLE LENGTH OF THE CROSS SECTIONS FOR SCATTERING -
CD                  INTO GROUP N.  THIS IS FOR ALL MATERIALS AND ALL -
CO                  LEGENDRE ORDERS, THUS IT IS THE SUM OF NGPB(L,J) -
CD                  OVER L FROM 1 TO NORD AND OVER J FROM 1 TO NMAT. -
C                                                                  -
CN                  THE SCATTERING CROSS SECTIONS ARE PACKED IN BANDS. -
CN                  ONE FOR EACH LEGENDRE ORDER AND MATERIAL. EACH BAND-
CN                  CONTAINS THE NGPB GROUPS WHICH SCATTER INTO GROUP -
CN                  N.  THE FIRST SOURCE GROUP NUMBER IS IFSG AND    -
CN                  THE LAST IS IFSG-NGPB+1. THE NORD BANDS FOR THE  -
CN                  FIRST MATERIAL APPEAR FIRST (PO, P1, ....) FOLLOWED-
CN                  BY THE NORD BANDS FOR THE SECOND, ETC.           -
C                                                                  -
CN                  HIGHER LEGENDRE ORDER SCATTERING CROSS SECTIONS  -
CN                  INCLUDE A 2*L+1 FACTOR WHERE L IS THE LEGENDRE   -
CN                  ORDER.                                           -
C                                                                  -
C-------------------------------------------------------------------
CEOF                                                               -
```

IV. ASGMAT File

The ASGMAT interface file contains the information needed by the Solver and Edit Modules to assign materials to zones to create the zone macroscopic cross sections.

```
C*****************************************************************
C                          DATE 09/18/81                         -
C                                                                -
CF          ASGMAT                                               -
CE          CODE DEPENDENT FILE ASSIGNING MATERIALS TO ZONES      -
C                                                                -
C                                                                -
C*****************************************************************
C                                                                -
CN               THIS FILE CONTAINS THE INFORMATION FROM THE INPUT -
CN               ARRAYS ASSIGN= AND ASGMOD=                      -
C                                                                -
C----------------------------------------------------------------
CS          FILE STRUCTURE                                       -
CS                                                               -
CS             RECORD TYPE                        PRESENT IF     -
CS             ****************************    **************   -
CS             FILE IDENTIFICATION                ALWAYS         -
CS             FILE CONTROL                       ALWAYS         -
CS             COMPATABILITY CODE                 ALWAYS         -
CS             MATERIAL NAMES                     ALWAYS         -
CS             ZONE NAMES                         ALWAYS         -
CS             NUMBER OF MATERIALS PER ZONE       MPZTOT.NE.O    -
CS             MATERIAL LIST FOR ALL ZONES        MPZTOT.NE.O    -
CS             MATERIAL CONCENTRATIONS            MPZTOT.NE.O    -
CS             MATERIAL CONCENTRATION FACTORS     MPZTOT.NE.O    -
CS             CONCENTRATION MODIFIER             MPZTOT.NE.O    -
C                                                                -
C----------------------------------------------------------------
C                                                                -
C----------------------------------------------------------------
CR          FILE IDENTIFICATION                                  -
C                                                                -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                -
C                                                                -
CW    1+3=MULT=NUMBER OF WORDS                                   -
C                                                                -
CD    HNAME          HOLLERITH FILE NAME - ASGMAT - (A6)         -
CD    HUSE(I)        HOLLERITH USER IDENTIFICATION (A6)          -
CO    IVERS          FILE VERSION NUMBER                         -
CD    MULT           DOUBLE PRECISION PARAMETER                  -
CD                       1- A6 WORD IS SINGLE WORD               -
CD                       2- A6 WORD IS DOUBLE PRECISION WORD     -
C                                                                -
C----------------------------------------------------------------
C----------------------------------------------------------------
CR          FILE CONTROL                                         -
C                                                                -
CL    MT, NZONE, MPZTOT                                          -
C                                                                -
CW    3=NUMBER OF WORDS                                          -
C                                                                -
CD   1  MT        NUMBER OF MATERIALS                            -
CD   2  NZONE     NUMBER OF ZONES                                -
CD   3  MPZTOT    IN-SOLVER MIXING TABLE LENGTH                  -
C                                                                -
C----------------------------------------------------------------
C                                                                -
C----------------------------------------------------------------
CR          COMPATABILITY CODE WORDS                             -
C                                                                -
CL    CODE1, CODE2                                               -
C                                                                -
CW    2=NUMBER OF WORDS                                          -
C                                                                -
CD   1  CODE1     DATE OF THE MACRXS FILE TO WHICH ASGMAT APPLIES -
CD   2  CODE2     TIME OF THE MACRXS FILE TO WHICH ASGMAT APPLIES -
C                                                                -
C----------------------------------------------------------------
```

```
C-----------------------------------------------------------------------
CR          MATERIAL NAMES                                             -
C                                                                      -
CL    (MATNAM(I),I=1,MT)                                               -
C                                                                      -
CW    MT=NUMBER OF WDRDS                                               -
C                                                                      -
CD      MATNAM(I) HOLLERITH NAME FOR THE I-TH MATERIAL                 -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          ZDNE NAMES                                                 -
C                                                                      -
CL    (ZDNNAM(I),I=1,NZONE)                                            -
C                                                                      -
CW    NZONE=NUMBER OF WDRDS                                            -
C                                                                      -
CO      ZONNAM(I) HOLLERITH NAME FOR THE I-TH ZONE                     -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          NUMBER OF MATERIALS PER ZONE                               -
C                                                                      -
CC     PRESENT IF MPZTOT.NE.O                                          -
C                                                                      -
CL    (NUMZON(I),I=1,NZONE)                                            -
C                                                                      -
CW    NZONE=NUMBER OF WORDS                                            -
C                                                                      -
CD      NUMZDN(I) NUMBER OF MATERIALS IN THE I-TH ZDNE                 -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          MATERIAL LIST FOR ALL ZDNES                                -
C                                                                      -
CC     PRESENT IF MPZTOT.NE.O                                          -
C                                                                      -
CL    (MATLST(I),I=1,MPZTOT)                                           -
C                                                                      -
CW    MPZTOT=NUMBER OF WORDS                                           -
C                                                                      -
CD      MATLST    PACKED LIST OF MATERIAL NUMBERS. MATERIALS FOR ZONE 1-
CD                FDLLOWED BY THDSE FOR ZONE 2. THEN ZONE3. ETC.     -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          MATERIAL CONCENTRATIONS                                    -
C                                                                      -
CC     PRESENT IF MPZTOT.NE.O                                          -
C                                                                      -
CL    (CONC(I),I=1,MPZTOT)                                             -
C                                                                      -
CW    MPZTOT=NUMBER OF WORDS                                           -
C                                                                      -
CD      CDNC(I)    CDNCENTRATION OF THE I-TH MATERIAL IN THE MATLST    -
CD                 ARRAY                                              -
C                                                                      -
C-----------------------------------------------------------------------
```

```
C-------------------------------------------------------------------------
CR          MATERIAL CONCENTRATION FACTORS                              -
C                                                                       -
CC     PRESENT IF MPZTOT.NE.O                                           -
C                                                                       -
CL    (C1(I),I=1,MPZTOT)                                                -
C                                                                       -
CW    MPZTOT=NUMBER OF WORDS                                            -
C                                                                       -
CD      C1(I)     CONCENTRATION FACTOR FOR THE I-TH MATERIAL IN THE     -
CO                MATLST ARRAY                                          -
C                                                                       -
C-------------------------------------------------------------------------
C                                                                       -
C-------------------------------------------------------------------------
CR          CONCENTRATION MODIFIER                                      -
C                                                                       -
CC     PRESENT IF MPZTOT.NE.O                                           -
C                                                                       -
CL    CMDD                                                              -
C                                                                       -
CW    1=NUMBER OF WORDS                                                 -
C                                                                       -
CD      CMOD      INPUT VALUE OF THE CONCENTRATION MODIFIER             -
C                                                                       -
C-------------------------------------------------------------------------
CEOF                                                                    -
```

V. SOLINP File

The SOLINP code-dependent interface file contains information specific to the Solver Module, mainly the information from BLOCK V of the card-image input.

```
C*****************************************************************************
C                          DATE 10/14/89                                      -
C                                                                             -
CF          SOLINP                                                            -
CE          CODE DEPENDENT FILE OF INFORMATION SPECIFIC TO THE                -
CE          ONEDANT SOLVER MODULE                                             -
C                                                                             -
C                                                                             -
C*****************************************************************************
C-----------------------------------------------------------------------------
CS          FILE ,STRUCTURE                                                   -
CS                                                                            -
CS              RECORD TYPE                          PRESENT IF               -
CS              ===========                          ==========               -
CS              FILE IDENTIFICATION                  ALWAYS                   -
CS              TITLE CARD COUNT                     ALWAYS                   -
CS      ***********(REPEAT FOR UP TO 10 CDS)                                  -
CS      *       TITLE CARD                           NHEAD.GT.0               -
CS      ***********                                                           -
CS              DIMENSION                            ALWAYS                   -
CS              RAW CONTROLS AND DIMENSIONS          ALWAYS                   -
CS              RAW FLOATING INPUT DATA              ALWAYS                   -
CS              DEFAULTED CONTROLS AND DIMENSIONS    ALWAYS                   -
CS              DEFAULTED FLOATING INPUT DATA        ALWAYS                   -
CS              BNDRY TRANSFER FIRST SOURCE GROUP    IBL.EQ.4.OR.IBR.EQ.4     -
CS              BNDRY TRANSFER VECTOR LENGTHS        IBL.EQ.4.OR.IBR.EQ.4     -
CS      ***********(REPEAT FOR ALL GROUPS)                                    -
CS      *       LEFT BDRY TRANSFER VECTOR            IBL.EQ.4                 -
CS      *       RIGHT BDRY TRANSFER VECTOR           IBR.EQ.4                 -
CS      ***********                                                           -
CS              FINE MESH DENSITY FACTORS            IDEN.EQ.1                -
CS              RADIUS MODIFIERS                     IEVT.EQ.4                -
CS              LEFT ALBEDOES                        IBL.EQ.4                 -
CS              RIGHT ALBEDOES                       IBR.EQ.4                 -
CS              SINGLE CHI ARRAY(FISSION SPECTRA)    INCHI.EQ.1               -
CS      ***********(REPEAT FOR ALL ZONES)                                     -
CS      *       CHI ARRAY(FISSION SPECTRA)           INCHI.EQ.2               -
CS      ***********                                                           -
CS              QUADRATURE WEIGHTS                   IQUAD.EQ.3               -
CS              QUADRATURE COSINES                   IQUAD.EQ.3               -
CS      ***********(REPEAT FOR ALL MOMENTS)                                   -
CS      *       SOURCE SPECTRUM                      IQOPT.EQ.1               -
CS      *       SOURCE SPATIAL DISTRIBUTION          IQOPT.EQ.2               -
CS      * *********(REPEAT FOR ALL GROUPS)                                    -
CS      * *     SOURCE SPATIAL DISTRIBUTION          IQOPT.EQ.3               -
CS      * *********                                                           -
CS      *       SOURCE SPECTRUM                      IQOPT.EQ.4               -
CS      *       SOURCE SPATIAL DISTRIBUTION          IQOPT.EQ.4               -
CS      ***********                                                           -
CS              LEFT BOUNDARY ISOTROPIC SOURCES      IQL.EQ.-*                -
CS              RIGHT BOUNDARY ISOTROPIC SOURCES     IQR.EQ.-1                -
CS      ***********(REPEAT FOR ALL GROUPS)                                    -
CS      *       LEFT BOUNDARY ANISOTROPIC SOURCE     IQL.EQ.+1                -
CS      *       RIGHT BOUNDARY ANISOTROPIC SOURCE    IQR.EQ.+1                -
CS      ***********                                                           -
C                                                                             -
C-----------------------------------------------------------------------------

C-----------------------------------------------------------------------------
CR          FILE IDENTIFICATION                                               -
C                                                                             -
CL      HNAME,(HUSE(I),I=1,2),IVERS                                           -
C                                                                             -
CW      1+3*MULT=NUMBER OF WORDS                                              -
C                                                                             -
CD      HNAME           HOLLERITH FILE NAME - SOLINP - (A6)                   -
CD      HUSE(I)         HOLLERITH USER IDENTIFICATION (A6)                    -
CD      IVERS           FILE VERSION NUMBER                                   -
CD      MULT            DOUBLE PRECISION PARAMETER                            -
CD                          1- A6 WORD IS SINGLE WORD                         -
CD                          2- A6 WORD IS DOUBLE PRECISION WORD               -
C                                                                             -
C-----------------------------------------------------------------------------
```

```
C-----------------------------------------------------------------------
CR          TITLE CARD COUNT                                           -
C                                                                      -
CL    NHEAD                                                            -
C                                                                      -
CW    1=NUMBER OF WORDS                                                -
C                                                                      -
CD    NHEAD     NUMBER OF TITLE CARDS TO FOLLOW                        -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          TITLE CARD                                                 -
C                                                                      -
CC      PRESENT IF NHEAD.GT.0                                          -
C                                                                      -
CL    (TITLE(I),I=1,12)                                                -
C                                                                      -
CW    12=NUMBER OF A6 WORDS                                            -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          SPATIAL DIMENSION                                          -
C                                                                      -
CL    IDIMEN                                                           -
C                                                                      -
CW    1=NUMBER OF WORDS                                                -
C                                                                      -
CD    IDIMEN=1 FOR ONEDANT                                             -
C                                                                      -
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
CR          RAW CONTROLS AND DIMENSIONS                                -
C                                                                      -
CL    IEVT, ITH,   ISCT,  ISN,   IQUAD, ISTART,ICSM, INCHI,IBL, IBR,  -
CL  1 IDENX, IPVT,  I2ANG, IQOPT, IQAN,  IQL,   IQR,  OITM, IITL,ITM,  -
CL  2 ITLIM,I1,    FLUXP, XSECTP,FISSRP,SOURCP,GEOMP,IANG, IACC,IRMFLX,-
CL  3 IGRPSN, IAFLUX,ISBEDO,IBALP, DUM3,  IBB,   IBT,  IITLO,IOT, IOB, -
CL  4 IXM,  IYM,   IZM,   IDENY, IDENZ, DUM4, ITMPMX, MFGACC,JUPDCH,  -
CL  5 JACC,  IBF,  IBA,                                                -
CL  6 (DUM5(I),I=1,48)                                                 -
C                                                                      -
CW    100=NUMBER OF WORDS                                              -
C                                                                      -
CD    1  IEVT     TYPE OF CALCULATION                                  -
CD    2  ITH      0/1 - DIRECT/ADJOINT CALCULATION                     -
CD    3  ISCT     LEGENDRE ORDER OF SCATTERING                         -
CD    4  ISN      ANGULAR QUADRATURE ORDER                             -
CD    5  IQUAD    SOURCE OF QUADRATURE SET                             -
CD    6  ISTART   FLUX GUESS FLAG (ZERO FOR ONEDANT)                   -
CD    7  ICSM     0/1 - NO/YES  IN-SOLVER MIXING(FROM ASSIGN= )        -
CD    8  INCHI    0/1/2 - NONE/ONE CHI/ZONEWISE CHI                    -
CD    9  IBL      0/1/2/3/4 - LEFT BDRY CONDITION                      -
CD   10  IBR      0/1/2/3/4 - RIGHT BDRY CONDITION                     -
CD                                                                     -
CD   11  IDENX    0/1 - NO/YES FINE MESH DENSITY FACTORS               -
CD   12  IPVT     0/1/2 - NONE/K-EFF/ALPHA  PARAMETRIC EIGENVALUE TYPE -
CD   13  I2ANG    0/1 - NO/YES  DO 2 ANGLE SLAB CALCULATION            -
CD   14  IQOPT    0/1/2/3/4/5 - INHOMOGENEOUS SOURCE OPTION            -
CD   15  IQAN     INHOMOGENEOUS SOURCE LEGENDRE ORDER                  -
CD   16  IQL      -1/0/1 - ISOTROPIC/NONE/ANGULAR  LEFT BOUNDARY SOURCE -
CD   17  IQR      -1/0/1 - ISOTROPIC/NONE/ANGULAR  RIGHT BOUNDARY SOURCE-
CD   18  OITM     OUTER ITERATION LIMIT                                -
CD   19  IITL     EARLY INNER ITERATION LIMIT                          -
CD   20  IITM     NEAR CONVERGENCE INNER ITERATION LIMIT               -
CD                                                                     -
CD   21  ITLIM    TIME LIMIT IN SECONDS                                -
CD   22  I1       NOT USED                                             -
CD   23  FLUXP    0/1/2 - NONE/ISOTROPIC/ALL MOMENTS  FLUX PRINT       -
CD   24  XSECTP   0/1/2 - NONE/PRINCIPAL/ALL  CROSS SECTION PRINT      -
CD   25  FISSRP   0/1 - NO/YES  FISSION RATE PRINT                     -
CD   26  SOURCP   0/1/2/3 - NO/AS READ/NORMALIZED/BOTH  SOURCE PRINT   -
CD   27  GEOMP    0/1 - NO/YES  FINE MESH GEOMETRY PRINT               -
CD   28  ANGP     0/1 - NO/YES  ANGULAR FLUX PRINT                     -
CD   29  IACC     ACCELERATION TYPE (FIXED AT 2 FOR ONEDANT)           -
CD   30  IRMFLX   WRITE CODE-DEPENDENT ZONE FLUXES                     -
```

```
CD                                                                      -
CO  31  IGRPSN   NOT USED BY ONEDANT                                    -
CD  32  RAFLUX   0/1 - NO/YES  WRITE ANGULAR FLUX FILE RAFLUX           -
CD  33  ISBEDO   ALBEDO PRESENCE CODE                                   -
CD  34  IBALP    0/1 - NO/YES  PRINT BALANCES BY ZONE                   -
CD  35  DUM3     NOT USED                                               -
CD  36  IBB      NOT USED BY ONEDANT                                    -
CO  37  IBT      NOT USED BY ONEDANT                                    -
CD  38  IITLD    NOT USED BY ONEDANT                                    -
CD  39  IQT      NOT USED BY ONEDANT                                    -
CD  40  IQB      NOT USED BY ONEDANT                                    -
CD                                                                      -
CD  41  IXM      NOT USED BY ONEDANT                                    -
CD  42  IYM      NOT USED BY ONEDANT                                    -
CD  43  IZM      NOT USED BY ONEDANT                                    -
CD  44  IDENY    NOT USED BY ONEDANT                                    -
CD  45  IDENZ    NOT USED BY ONEDANT                                    -
CD  46  DUM4     NOT USED                                               -
CO  47  ITMPMX   NOT USED BY ONEDANT(RESERVED)                          -
CD  48  MFGACC   NOT USED BY ONEDANT(RESERVED)                          -
CD  49  JUPDCH   NOT USED BY ONEDANT(RESERVED)                          -
CO  50  JACC     NOT USED BY ONEDANT(RESERVED)                          -
CO                                                                      -
CD  51  IBF      NOT USED BY ONEDANT(RESERVED)                          -
CD  52  IBA      NOT USED BY ONEDANT(RESERVED)                          -
CD                                                                      -
CD  53  DUM5     VECTOR NOT USED                                        -
C-----------------------------------------------------------------------
C                                                                       -
C                                                                       -
C-----------------------------------------------------------------------
CR          RAW FLOATING DATA                                           -
C                                                                       -
CL    EV,   NORM, EPSO, EPSI, BHGT, BWTH, EVM, PV, XLAL, XLAH,          -
CL  1 XLAX, POD,  EPSR, EPSX, EPST,                                     -
CL  2 (DUM(I),I=1,10),                                                  -
CL  3 EFACT, TO, TS, EOM, SIGTH, TRCOR, PLANET, FCSRC,                  -
CL  4 (DUM1(I),I=1,57),                                                 -
CL  5 (EXTRAS(I),I=1,110)                                               -
C                                                                       -
CW    200*MULT=NUMBER OF WORDS                                          -
C                                                                       -
CO   1  EV       EIGENVALUE GUESS                                       -
CD   2  NORM     NORMALIZATION CONSTANT                                 -
CO   3  EPSO     OUTER ITERATION CONVERGENCE CRITERION                   -
CD   4  EPSI     INNER ITERATION CONVERGENCE CRITERION                   -
CD   5  BHGT     BUCKLING HEIGHT                                        -
CO   6  BWTH     BUCKLING WIDTH                                         -
CD   7  EVM      EIGENVALUE MODIFIER                                    -
CD   8  PV       PARAMETRIC VALUE                                       -
CD   9  XLAL     LAMBDA LOWER LIMIT FOR SEARCHES                        -
CO  10  XLAH     LAMBDA UPPER LIMIT FOR SEARCHES                        -
CD                                                                      -
CD  11  XLAX     SEARCH CONVERGENCE CRITERION                           -
CD  12  POD      PARAMETER OSCILLATION DAMPER                           -
CD  13  EPSR     DIFFUSION PERIODIC BDRY ITERATION CONV. CRITERION       -
CD  14  EPSX     MAX FRACTIONAL POINTWISE CHANGE CRITERION               -
CD  15  EPST     NOT USED BY ONEDANT                                    -
CO                                                                      -
CD  16  DUM      VECTOR NOT USED                                        -
CD                                                                      -
CD  26  EFACT    NOT USED BY ONEDANT(RESERVED)                          -
CD  27  TO       NOT USED BY ONEDANT(RESERVED)                          -
CD  28  TS       NOT USED BY ONEDANT(RESERVED)                          -
CO  29  EOM      NOT USED BY ONEDANT(RESERVED)                          -
CO  30  SIGTH    NOT USED BY ONEDANT(RESERVED)                          -
CD                                                                      -
CD  31  TRCOR    TRANSPORT CORRECTION INDICATOR                         -
CD  32  PLANET   PLANET INDICATOR                                       -
CD  33  FCSRC    NOT USED BY ONEDANT(RESERVED)                          -
CD                                                                      -
CD  34  DUM1     VECTOR NOT USED                                        -
CD                                                                      -
CD  91  XETRAS   VECTOR USED BY INDIVIDUAL SOLVERS                      -
C                                                                       -
C-----------------------------------------------------------------------
```

```
C-------------------------------------------------------------------
CR          DEFAULTED CONTROLS AND DIMENSIONS                      -
C                                                                  -
CN   THIS RECORD IS THE SAME FORMAT AS THE RAW CONTROLS AND DIMENSION -
CN   RECORD ABOVE, BUT IT CONTAINS THE DEFAULTED VALUES FOR EACH    -
CN   VARIABLE                                                      -
C                                                                  -
C-------------------------------------------------------------------
C-------------------------------------------------------------------
CR          DEFAULTED FLOATING DATA                                -
C                                                                  -
CN   THIS RECORD IS THE SAME FORMAT AS THE RAW FLOATING DATA        -
CN   RECORD ABOVE, BUT IT CONTAINS THE DEFAULTED VALUES FOR EACH    -
CN   VARIABLE                                                      -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          BOUNDARY TRANSFER FIRST SOURCE GROUP                   -
C                                                                  -
CC      PRESENT IF IBL.EQ.4 .OR. IBR.EQ.4                          -
C                                                                  -
CL    (IFSGL(N),N=1,NGROUP),(IFSGR(N),N=1,NGROUP)                  -
C                                                                  -
CW    2*NGROUP*NUMBER OF WORDS                                     -
C                                                                  -
CD      IFSGL     FIRST SOURCE GROUP FOR LEFT BOUNDARY             -
CD      IFSGR     FIRST SOURCE GROUP FOR RIGHT BOUNDARY            -
CD      NGROUP    NUMBER OF ENERGY GROUPS                          -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          BOUNDARY TRANSFER VECTOR LENGTHS                       -
C                                                                  -
CC      PRESENT IF IBL.EQ.4 .OR. IBR.EQ.4                          -
C                                                                  -
CL    (LENL(N),N=1,NGROUP),(LENR(N),N=1,NGROUP)                    -
C                                                                  -
CW    2*NGROUP*NUMBER OF WORDS                                     -
C                                                                  -
CD      LENL      LENGTH OF THE VECTOR OF TRANSFERS INTO GRP N     -
CD                AT THE LEFT BOUNDARY                             -
CD      LENR      LENGTH OF THE VECTOR OF TRANSFERS INTO GRP N     -
CD                AT THE RIGHT BOUNDARY                            -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          LEFT BOUNDARY TRANSFER VECTOR                          -
C                                                                  -
CC      PRESENT IF IBL.EQ.4                                        -
C                                                                  -
CL    (TRL(I),I=1,NW)                                              -
C                                                                  -
CW    NW=NUMBER OF WORDS                                           -
C                                                                  -
CD      TRL(I)    I-TH TRANSFER INTO GROUP N FOR LEFT BOUNDARY     -
CD                THE FIRST VALUE IS THE TRANSFER FROM GROUP IFSGL(N) -
CD                INTO GROUP N. NEXT IS FROM GROUP IFSGL(N)-1 TO N. ETC.-
CD      NW        LENL(N)                                          -
CD      N         NUMBER OF THE RECEIVING GROUP                    -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
```

```
C-------------------------------------------------------------------
CR          RIGHT BOUNDARY TRANSFER VECTOR                         -
C                                                                  -
CC      PRESENT IF IBR.EQ.4                                        -
C                                                                  -
CL    (TRR(I),I=1,NW)                                              -
C                                                                  -
CW    NW=NUMBER OF WOROS                                           -
C                                                                  -
CO      TRR(I)    I-TH TRANSFER INTO GROUP N FOR RIGHT BOUNOARY    -
CO                THE FIRST VALUE IS THE TRANSFER FROM GROUP IFSGR(N)  -
CO                INTO GROUP N. NEXT IS FROM GROUP IFSGR(N)-1 TO N. ETC.-
CO      NW        LENR(N)                                          -
CD      N         NUMBER OF THE RECEIVING GROUP                    -
C                                                                  -
C-------------------------------------------------------------------

C-------------------------------------------------------------------
CR          FINE MESH DENSITY FACTORS                              -
C                                                                  -
CC     PRESENT IF IDENX.EQ.1                                       -
C                                                                  -
CL    (DEN(I),I=1,IT)                                              -
C                                                                  -
CW    IT=MULT=NUMBER OF WORDS                                      -
C                                                                  -
CD      IT        NUMBER OF FINE MESH INTERVALS                    -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          RADIAL MOOIFIERS                                       -
C                                                                  -
CC       PRESENT IF IEVT.EQ.4                                      -
C                                                                  -
CL    (RM(I),I=1,IM)                                               -
C                                                                  -
CW    IM=MULT=NUMBER OF WORDS                                      -
C                                                                  -
CO      IM        NUMBER OF COARSE MESH INTERVALS                  -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          LEFT ALBEDOES                                          -
C                                                                  -
CC      PRESENT IF IBL.EQ.4                                        -
C                                                                  -
CL    (LBEDO(N),N=1,NGROUP)                                        -
C                                                                  -
CW    NGROUP=MULT=NUMBER OF WORDS                                  -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          RIGHT ALBEDOES                                         -
C                                                                  -
CC      PRESENT IF IBR.EQ.4                                        -
C                                                                  -
CL    (RBEDO(N),N=1,NGROUP)                                        -
C                                                                  -
CW    NGROUP=MULT=NUMBER OF WORDS                                  -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          CHI - FISSION SPECTRA                                  -
C                                                                  -
CC      PRESENT IF INCHI.NE.O                                      -
C                                                                  -
CL    (CHI(N),N=1,NGROUP)                                          -
C                                                                  -
CW    NGROUP=MULT=NUMBER OF WORDS                                  -
C                                                                  -
C-------------------------------------------------------------------
```

```
C
C
C-----------------------------------------------------------------------
CR          QUADRATURE WEIGHTS
C
CC      PRESENT IF IQUAD.EQ.3
C
CL    (WGT(M),M=1,MM)
C
CW    MM=MULT=NUMBER OF WOROS
C
CD      MM        NUMBER OF ANGLES IN QUADRATURE SET
C
C-----------------------------------------------------------------------
C
C-----------------------------------------------------------------------
CR          QUAORATURE COSINES
C
CC      PRESENT IF IQUAD.EQ.3
C
CL    (MU(M),M=1,MM)
C
CW    MM=MULT=NUMBER OF WOROS
C
C-----------------------------------------------------------------------
C
C-----------------------------------------------------------------------
CR          SOURCE SPECTRUM
C
CC      PRESENT IF IQOPT.EQ.1 .OR. IQOPT.EQ.4
C
CL    (SOURCE(N),N=1,NGRDUP)
C
CW    NGROUP=MULT=NUMBER OF WORDS
C
C-----------------------------------------------------------------------
C
C-----------------------------------------------------------------------
CR          SOURCE SPATIAL DISTRIBUTION
C
CC      PRESENT IF IQOPT.EQ.2 .OR. IQOPT.EQ.3 .OR. IQOPT.EQ.4
C
CL    (SOURCX(I),I=1,IT)
C
CW    IT=MULT=NUMBER OF WOROS
C
C-----------------------------------------------------------------------
C
C
C-----------------------------------------------------------------------
CR          LEFT BOUNDARY ISOTROPIC SOURCE
C
CC      PRESENT IF IBL.EQ.-1
C
CL    (SILEFT(N),N=1,NGROUP)
C
CW    NGROUP=MULT=NUMBER OF WORDS
C
C-----------------------------------------------------------------------
C
C-----------------------------------------------------------------------
CR          RIGHT BOUNDARY ISOTROPIC SOURCE
C
CC      PRESENT IF IBL.EQ.-1
C
CL    (SIRITE(N),N=1,NGROUP)
C
CW    NGROUP=MULT=NUMBER OF WOROS
C
C-----------------------------------------------------------------------
```

```
C
C-----------------------------------------------------------------------
CR          LEFT BOUNDARY ANISOTROPIC SOURCE                           -
C                                                                      -
CC       PRESENT IF IBL.EQ.+1                                          -
C                                                                      -
CL     (SALEFT(M),M=1,MMHALF)                                          -
C                                                                      -
CW     MMHALF=MULT=NUMBER OF WOROS                                     -
C                                                                      -
CD       SALEFT    LEFT BOUNDARY SOURCE FOR GROUP N                    -
CO       MMHALF    NUMBER OF ANGLES IN THE SOURCE(=MM/2)               -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          RIGHT BOUNDARY ANISOTROPIC SOURCE                          -
C                                                                      -
CC       PRESENT IF IBL.EQ.+1                                          -
C                                                                      -
CL     (SARITE(M),M=1,MMHALF)                                          -
C                                                                      -
CW     MMHALF=MULT=NUMBER OF WORDS                                     -
C                                                                      -
CO       SARITE    RIGHT BOUNDARY SOURCE FOR GROUP N                   -
CD       MMHALF    NUMBER OF ANGLES IN THE SOURCE(=MM/2)               -
C                                                                      -
C-----------------------------------------------------------------------
CEOF                                                                   -
```

VI. EDITIT File

The EDITIT code-dependent interface file contains information specific to the Edit Module, mainly information from BLOCK VI of the card-image input.

```
C*****************************************************************
C                        DATE 06/13/89                           -
C                                                                -
CF          EDITIT                                               -
CE          CODE OEPENDENT FILE OF INFORMATION SPECIFIC TO THE    -
CE          ONEOANT EDIT MODULE                                  -
C                                                                -
C                                                                -
C*****************************************************************
C                                                                -
CN               THIS FILE CONTAINS THE CARD INPUT INFORMATION   -
CN               FROM BLOCK VI                                   -
C                                                                -
C----------------------------------------------------------------
CS          FILE STRUCTURE                                       -
CS                                                               -
CS             RECORD TYPE                          PRESENT IF   -
CS             ****************************       **********   -
CS             FILE IDENTIFICATION                  ALWAYS       -
CS             RAW CONTROLS AND DIMENSIONS          ALWAYS       -
CS             DEFAULTED CONTROLS AND DIMENSIONS    ALWAYS       -
CS             RAW FLOATING INPUT DATA              ALWAYS       -
CS             FINE GROUPS PER BROAD GROUP          ALWAYS       -
CS             ZONE NUMBERS                         NZNS.NE.O    -
CS             POINT NUMBERS                        NIPE.NE.O    -
CS             CRDSS SECTION POSITIONS              NPOS.NE.O    -
CS             ISOTOPE NUMBERS TO EDIT              NISO.NE.O    -
CS             MATERIAL NUMBERS TO EDIT             MACRO.NE.O   -
CS             CONSTITUENT NUMBERS TO EDIT          NCONS.NE.O   -
CS             RESPONSE FUNCTION NAMES              IOOSE.NE.O   -
CS    **********(REPEAT FOR ALL RESPONSE FUNCTIONS)              -
CS    *        RESPONSE FUNCTION ENERGY VECTOR      IOOSE.NE.O   -
CS    *        RESPONSE FUNCTION SPATIAL VECTOR     IDOSE.NE.O   -
CS    **********                                                 -
CS             CROSS SECTION SUMMING ARRAY -        IXSUM.NE.O   -
CS             RESPONSE FUNCTION SUMMING ARRAY      IRSUM.NE.O   -
CS             FINE MESH DENSITY FACTORS            IDEN.NE.O    -
C                                                                -
C----------------------------------------------------------------
C                                                                -
C----------------------------------------------------------------
CR          FILE IDENTIFICATION                                  -
C                                                                -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                -
C                                                                -
CW    1+3*MULT=NUMBER OF WORDS                                   -
C                                                                -
CD    HNAME          HOLLERITH FILE NAME - EDITIT - (A6)         -
CD    HUSE(I)        HOLLERITH USER IDENTIFICATION (A6)          -
CD    IVERS          FILE VERSION NUMBER                         -
CD    MULT           DOUBLE PRECISION PARAMETER                  -
CD                       1- A6 WORD IS SINGLE WORD               -
CD                       2- A6 WORD IS DOUBLE PRECISION WORD     -
C                                                                -
C----------------------------------------------------------------
C                                                                -
```

```
C-------------------------------------------------------------------
CR          RAW CONTROLS AND DIMENSIONS                            -
C                                                                  -
CL    IEDOPT,PTED,  NIPE,  IKND,   ZNED, NZNS,IXSUM,IRSUM, NPOS, NISO, -
CL  1 NCONS, MACRO, IDOSE, IGRPED, LNG,  NBG, IDEN, NGROUP, AJED,  -
CL  2 ITED, JTED, KTED, IRZFLX, IRZMFX, EDOUTF                     -
C                                                                  -
CW    25=NUMBER OF WORDS                                           -
C                                                                  -
CD   1  IEDOPT   NOT USED                                          -
CD   2  PTED     0/1 - NO/YES  DO POINT EDIT                       -
CD   3  NIPE     NUMBER OF POINTS TO EDIT                          -
CD   4  BYVOLP   0/1 - NO/YES  MULTIPLY REACTION RATES BY MESH VOLUME -
CD   5  ZNED     0/1 - NO/YES  DO ZONE EDIT                        -
CD   6  NZNS     NUMBER OF EDIT ZONES                             -
CD   7  IXSUM    LENGTH OF CROSS SECTION SUMMING TABLE            -
CD   8  IRSUM    LENGTH OF RESPONSE FUNCTION SUMMING TABLE         -
CD   9  NPOS     NUMBER OF CROSS SECTION POSITIONS TO EDIT         -
CD  10  NISO     NUMBER OF ISOTOPES TO EDIT                       -
CD                                                                 -
CD  11  NCONS    NUMBER OF ISOTOPES TO EDIT AS CONSTITUENTS        -
CD  12  MACRO    NUMBER OF MATERIALS TO EDIT                      -
CD  13  IDOSE    NUMBER OF RESPONSE FUNCTIONS TO EDIT             -
CD  14  IGRPED   0/1/2/3 - ENERGY GROUP PRINT OPTIONS             -
CD  15  LNG      NUMBER OF THE LAST NEUTRON GROUP                 -
CO  16  NBG      NUMBER OF BROAD ENERGY GROUPS                    -
CD  17  IDEN     0/1 - NO/YES  THERE ARE FINE MESH DENSITY FACTORS -
CD  18  NGROUP   NUMBER OF FINE ENERGY GROUPS                     -
CD  19  AJED     0/1 - NO/YES  THIS IS AN ADJOINT EDIT            -
CD  20  ITED     NUMBER OF FINE RADIAL MESH                       -
CD  21  JTED     NUMBER OF FINE AXIAL MESH                        -
CO  22  KTED     NUMBER OF FINE Z DIRECTION MESH                  -
CO  23  IRZFLX   WRITE CCCC RZFLUX FILE                           -
CO  24  IRZMFX   WRITE ZONE MOMENTS FILE                          -
CO  25  EDOUTF   -3/-2/0/1/2/3 ASC EDIT FILE PREPARATION INDICATOR -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          DEFAULTED CONTROLS AND DIMENSIONS                      -
C                                                                  -
CN   THIS RECORD IS THE SAME FORMAT AS THE RAW CONTROLS AND        -
CN   DIMENSION RECORD ABOVE, BUT IT CONTAINS THE DEFAULTED         -
CN   VALUES FOR EACH VARIABLE                                      -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          RAW FLOATING INPUT DATA                                -
C                                                                  -
CL    POWER, MEVPER                                                -
C                                                                  -
CW    2=NUMBER OF WORDS                                            -
C                                                                  -
CD   1  POWER    NORMALIZE TO POWER                                -
CO   2  MEVPER   MEV RELEASED PER FISSION                          -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
C-------------------------------------------------------------------
CR          FINE GROUPS PER GROAD GROUP                            -
C                                                                  -
CL    (ICOLL(G),G=1,NBG)                                           -
C                                                                  -
CW    NBG=NUMBER OF WORDS                                          -
C                                                                  -
CD      ICOLL(G) NUMBER OF FINE GROUPS IN BROAD GROUP G            -
C                                                                  -
C-------------------------------------------------------------------
C                                                                  -
```

```
C-----------------------------------------------------------------------
CR          ZONE NUMBERS                                               -
C                                                                      -
CC     PRESENT IF NZNS.GT.0                                            -
C                                                                      -
CL    (EDZONE(I),I=1,IT)                                               -
C                                                                      -
CW    IT=NUMBER OF WORDS                                               -
C                                                                      -
CD      EDZONE(I) EDIT ZONE NUMBER FOR THE I-TH FINE MESH              -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          POINTS TO EDIT                                             -
C                                                                      -
CC     PRESENT IF NIPE.GT.0                                            -
C                                                                      -
CL    (POINTS(I),I=1,NIPE)                                             -
C                                                                      -
CW    NIPE=NUMBER DF WORDS                                             -
C                                                                      -
CD      POINTS(I) NUMBER OF THE I-TH POINT TO EDIT                     -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          CROSS SECTION PDSITIDNS TO EDIT                            -
C                                                                      -
CC     PRESENT IF NPOS.GT.0                                            -
C                                                                      -
CL    (EDXS(I),I=1,NPDS)                                               -
C                                                                      -
CW    NPDS=NUMBER DF WDRDS                                             -
C                                                                      -
CD      EDXS(I)   POSITIDN NUMBER TO EDIT(IN NUMERIC FORM)             -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          ISOTOPE NUMBERS TO EDIT                                    -
C                                                                      -
CC     PRESENT IF NISO.GT.0                                            -
C                                                                      -
CL    (EDISOS(I),I=1,NISO)                                             -
C                                                                      -
CW    NISD=NUMBER OF WORDS                                             -
C                                                                      -
CD      EDISOS(I) ISOTOPE NUMBER TO EDIT(IN NUMERIC FORM)              -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          MATERIAL NUMBERS TO EDIT                                   -
C                                                                      -
CC     PRESENT IF MACRO.GT.0                                           -
C                                                                      -
CL    (EDMATS(I),I=1,MACRO)                                            -
C                                                                      -
CW    MACRO=NUMBER OF WORDS                                            -
C                                                                      -
CD      EDMATS(I) MATERIAL NUMBER TO EDIT(IN NUMERIC FORM)             -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          CONSTITUENT NUMBERS TO EDIT                                -
C                                                                      -
CC     PRESENT IF NCONS.GT.0                                           -
C                                                                      -
CL    (EDCONS(I),I=1,NCONS)                                            -
C                                                                      -
CW    NCONS=NUMBER OF WORDS                                            -
C                                                                      -
CD      EDCONS(I) CONSTITUENT NUMBER TO EDIT(IN NUMERIC FORM)          -
C                                                                      -
C-----------------------------------------------------------------------
```

```
C-----------------------------------------------------------------------
CR          RESPONSE FUNCTION NAMES                                    -
C                                                                      -
CC     PRESENT IF IDOSE.GT.O                                           -
C                                                                      -
CL    (RSFNAM(I),I=1,IDOSE)                                            -
C                                                                      -
CW    IDOSE=NUMBER OF WORDS                                            -
C                                                                      -
CO      RSFNAM(I) HOLLERITH NAME FOR THE I-TH RESPONSE FUNCTION        -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          RESPONSE FUNCTION ENERGY VECTOR                            -
C                                                                      -
CC     PRESENT IF IDOSE.GT.O                                           -
C                                                                      -
CL    (RSFE(I),I=1,NGROUP)                                             -
C                                                                      -
CW    NGRDUP=NUMBER OF WORDS                                           -
C                                                                      -
CD      RSFE(I)   RESPONSE FOR GROUP I                                 -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          RESPONSE FUNCTIDN SPATIAL VECTOR                           -
C                                                                      -
CC     PRESENT IF IDOSE.GT.O                                           -
C                                                                      -
CL    (RSFX(I),I=1,IT)                                                 -
C                                                                      -
CW    IT=NUMBER DF WDRDS                                               -
C                                                                      -
CD      RSFX(I)   RESPONSE FUNCTIDN FDR FINE MESH I                    -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          CROSS SECTION SUMMING ARRAY                                -
C                                                                      -
CC     PRESENT IF IXSUM.GT.O                                           -
C                                                                      -
CL    (MICSUM(I),I=1,IXSUM)                                            -
C                                                                      -
CW    IXSUM=NUMBER OF WORDS                                            -
C                                                                      -
CD      MICSUM    INPUT SUMMING ARRAY IN NUMERIC FDRM                  -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          RESPDNSE FUNCTION SUMMING ARRAY                            -
C                                                                      -
CC     PRESENT IF IRSUM.GT.O                                           -
C                                                                      -
CL    (IRSUMS(I),I=1,IRSUM)                                            -
C                                                                      -
CW    IRSUM=NUMBER OF WORDS                                            -
C                                                                      -
CD      IRSUMS    INPUT SUMMING ARRAY IN NUMERIC FORM                  -
C                                                                      -
C-----------------------------------------------------------------------
C                                                                      -
C-----------------------------------------------------------------------
CR          FINE MESH DENSITY ARRAY                                    -
C                                                                      -
CC     PRESENT IF IDEN.GT.O                                            -
C                                                                      -
CL    (XDF(I),I=1,IT)                                                  -
C                                                                      -
CW    IT=NUMBER OF WORDS                                               -
C                                                                      -
CD      XDF(I) DENSITY FACTOR FOR THE I-TH FINE MESH                   -
C                                                                      -
C-----------------------------------------------------------------------
CEOF                                                                   -
```

VII. BXSLIB File

The BXSLIB code-dependent file contains, in binary form, the cross sections and other information as described in Ch. VIII.A.3.

```
C-----------------------------------------------------------------------
C                       DATE 09/22/88                                  -
C                                                                      -
CF          BXSLIB                                                     -
CE          MICROSCOPIC GRDUP NEUTRON CROSS SECTIONS FRDM CARDS        -
C                                                                      -
CN                      THIS FILE CONTAINS IN BINARY FORM THE          -
CN                      BLOCK III ONEDANT INPUT TOGETHER WITH THE      -
CN                      CROSS SECTIONS FROM THE ORIGINAL CARD LIBRARY. -
CN                      THE FILE ALSO MAY CONTAIN ISOTOPE/ATOMIC WEIGHT-
CN                      PAIRS FROM THE BLOCK IV ONEDANT INPUT          -
C                                                                      -
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
CS          FILE STRUCTURE                                             -
CS                                                                     -
CS             RECORD TYPE                          PRESENT IF         -
CS             ****************************      **************      -
CS             FILE IDENTIFICATION                  ALWAYS             -
CS             FILE CONTROL                         ALWAYS             -
CS             FILE OATA                            ALWAYS             -
CS                                                                     -
CS    ************(REPEAT FOR ALL ISOTOPES)                           -
CS    *  **********(REPEAT FDR ALL LEGENDRE ORDERS)                   -
CS    *  *       CRDSS SECTIDN SET                   ALWAYS           -
CS    *  **********                                                   -
CS    ************                                                   -
CS             ISOTOPE LABEL/ATOMIC WEIGHT PAIR     NISOAW.GT.O        -
C                                                                      -
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
CR          FILE IDENTIFICATION                                        -
C                                                                      -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                      -
C                                                                      -
CW    1+3*MULT*NUMBER OF WORDS                                         -
C                                                                      -
C                                                                      -
CD    HNAME          HOLLERITH FILE NAME - BXSLIB - (A6)               -
CD    HUSE(I)        HOLLERITH USER IDENTIFICATION (A6)                -
CD    IVERS          FILE VERSION NUMBER                               -
CD    MULT           OOUBLE PRECISIDN PARAMETER                        -
CD                       1- A6 WORD IS SINGLE WORD                     -
CD                       2- A6 WORD IS DOUBLE PRECISION WORD           -
C                                                                      -
C-----------------------------------------------------------------------
C-----------------------------------------------------------------------
CR          FILE CONTRDL                                               -
C                                                                      -
CL    NGRDUP,NISD,IHM,IHT,IHS,MAXT,NISOAW,NXSREC,I2LP1                 -
C                                                                      -
CW    9*NUMBER DF WORDS                                                -
C                                                                      -
C                                                                      -
CD    NGROUP         NUMBER OF ENERGY GROUPS IN FILE                   -
CD    NISO           NUMBER OF ISOTOPES IN FILE                        -
CD    IHM            TABLE LENGTH (NUMBER OF CROSS SECTIONS FOR        -
CD                   ONE ISDTOPE, FOR ONE GRDUP, AND FDR ONE LEGENDRE  -
CD                   ORDER                                             -
CD    IHT            TOTAL CROSS SECTIDN POSITIDN IN THE TABLE         -
CD    IHS            SELF SCATTER CROSS SECTION POSITION               -
CD    MAXT           MAXIMUM NUMBER OF LEGENDRE MOMENTS                -
CD    NISDAW         NUMBER OF ISDTOPE LABEL/ATOMIC WEIGHT PAIR        -
CD    NXSREC         TOTAL NUMBER OF CROSS SECTION SETS                -
CD    I2LP1          O/1 - NO/YES  SCATTERING CRDSS SECTIONS CONTAIN   -
CD                                 2L+1 FACTOR                       -
C                                                                      -
C-----------------------------------------------------------------------
```

```
C-------------------------------------------------------------------
CR          FILE DATA                                              -
C                                                                  -
CL    (HSETID(I),I=1,12),(HISONM(I),I=1,NISO),(EDNAME(I),I=1,IHT-3),  -
CL   1(CHI(J),J=1,NGROUP),(VEL(J),J=1,NGROUP),                    -
CL   2(EMAX(J),J=1,NGROUP),EMIN,(NSPI(I),I=1,NISO)                -
C                                                                  -
CW    (NISO+12+IHT-3)*MULT+(3*NGROUP+1)*MULT+NISO*NUMBER OF WORDS  -
C                                                                  -
C                                                                  -
CD    HSETID(I)      HOLLERITH IDENTIFICATION OF FILE (A6)         -
CD    HISONM(I)      HOLLERITH ISOTOPE LABEL FOR ISOTOPE I (A6)    -
CD    EDNAME(I)      HOLLERITH NAME FOR EDIT POSITIONS             -
CD                   PRECEDING SIGMA ABSORPTION FOR POSITION I (A6)-
CD    CHI(J)         FILE-WIDE FISSION SPECTRUM(ZEROES IN ONEDANT)  -
CO    VEL(J)         MEAN NEUTRON VELOCITY IN GROUP J (CM/SEC)      -
CD    EMAX(J)        MAXIMUM ENERGY BOUND OF GROUP J (EV)           -
CD    EMIN           MINIMUM ENERGY BOUND OF SET (EV)               -
CO    NSPI(I)        NUMBER OF LEGENDRE ORDERS FOR ISOTOPE I        -
C                                                                  -
C-------------------------------------------------------------------
C-------------------------------------------------------------------
CR          CROSS SECTION SET FOR ISOTOPE I AND LEGENDRE ORDER M    -
C                                                                  -
CL    ((C(I,J),I=1,IHM),J=1,NGROUP)                                -
C                                                                  -
CW    IHM*NGROUP*MULT*NUMBER OF WORDS                              -
C                                                                  -
C-------------------------------------------------------------------
C-------------------------------------------------------------------
CR          HOLLERITH ISOTOPE LABEL/ATOMIC WEIGHT PAIR SET         -
C                                                                  -
CC   PRESENT IF NISOAW.GT.0                                        -
C                                                                  -
CL    (ATWTP(I),I=1,2*NISOAW)                                      -
C                                                                  -
CW    2*NISOAW*MULT*NUMBER OF WORDS                                -
C                                                                  -
C-------------------------------------------------------------------
CEOF                                                               -
```

VIII. RMFLUX File

The RMFLUX code-dependent file contains, in binary form, the spherical harmonics angular flux moments for all spatial fine mesh points and all energy groups. It is optionally produced by the Solver Module.

```
C*****************************************************************************
C                          DATE 04/01/85                                      -
C                                                                             -
CF          RMFLUX-IV                                                       -
CE          REGULAR FLUX MOMENTS                                            -
C                                                                             -
C*****************************************************************************
C-----------------------------------------------------------------------------
CR          FILE IOENTIFICATION                                             -
C                                                                             -
CL    HNAME,(HUSE(I),I=1,2),IVERS                                           -
C                                                                             -
CW    1+3*MULT*NUMBER OF WORDS                                              -
C                                                                             -
CD    HNAME               HOLLERITH FILE NAME - RMFLUX - (A6)               -
CD    HNAME               HOLLERITH FILE NAME -        -(A6)                -
CO    HUSE(I)             HOLLERITH USER IOENTIFICATION (A6)                -
CD    IVERS               FILE VERSION NUMBER                               -
CD    MULT                DOUBLE PRECISION PARAMETER                        -
CD                            1- A6 WORD IS SINGLE WORD                     -
CD                            2- A6 WORD IS DOUBLE PRECISION WORO           -
C                                                                             -
C-----------------------------------------------------------------------------
C-----------------------------------------------------------------------------
CR          SPECIFICATIONS     (10 RECORO)                                  -
C                                                                             -
CL    NDIM,NGROUP,NINTI,NINTJ,NINTK,NORD,EFFK,POWER                         -
C                                                                             -
CW    8*NUMBER OF WORDS                                                     -
C                                                                             -
CD    NDIM                NUMBER OF DIMENSIONS                              -
CD    NGROUP              NUMBER OF GROUPS                                  -
CD    NINTI               NUMBER OF FIRST DIMENSION INTERVALS               -
CD    NINTJ               NUMBER OF SECOND OIMENSION INTERVALS              -
CD                        NINTJ.EQ.1 IF NOIM.EQ.1                           -
CO    NINTK               NUMBER OF THIRO OIMENSION INTERVALS               -
CO                        NINTK.EQ.1 IF NOIM.LE.2                           -
CO    NORO                NUMBER OF LEGENDRE MOMENTS                        -
CD    EFFK                EFFECTIVE MULTIPLICATION FACTOR                   -
CO    POWER               POWER IN WATTS TO WHICH FLUX IS NORMALIZED        -
C                                                                             -
C-----------------------------------------------------------------------------
C-----------------------------------------------------------------------------
CR          REGULAR MOMENTS FLUXES ON MULTIDIMENSIONAL INTERVALS              -
C                           (2D RECORD)                                       -
C                                                                             -
CL    ((FLUX(M,I),M=1,NORO),I=1,NINTI)----NOTE STRUCTURE BELOW---             -
C                                                                             -
CW    NORD*NINTI*NUMBER OF WORDS                                            -
C                                                                             -
C     DO 1 L=1,NGROUP                                                       -
C     DO 1 K=1,NINTK                                                        -
C     DO 1 J=1,NINTJ                                                        -
C   1 READ (N) *LIST AS ABOVE*                                              -
C                                                                             -
CD    FLUX(M,I)           REGULAR FLUX MOMENTS ON FIRST OIMENSION           -
CD                        INTERVALS.  ORDER OF GROUPS IS ACCORDING          -
CD                        TO DECREASING ENERGY.                             -
C                                                                             -
C-----------------------------------------------------------------------------
CEOF                                                                          -
```

IX. RZMFLX File

The RZMFLX file is a binary, code-dependent file containing the spherical harmonics angular fiux moments averaged over each zone for each energy group. The zones over which the fiuxes are averaged are the zones used in the Solver Module and *not* the Edit Zones optionally used in the Edit Module.

```
C*****************************************************************
C                        DATE 09/29/89                           -
C                                                                -
CF          RZMFLX-IV                                           -
CE          REGULAR ZONE FLUX MOMENTS BY GRDUP, AVERAGED OVER EACH ZDNE-
C                                                                -
C*****************************************************************
C-----------------------------------------------------------------
CR          FILE IDENTIFICATION                                 -
C                                                                -
CL     HNAME,(HUSE(I),I=1,2),IVERS                              -
C                                                                -
CW     1+3*MULT*NUMBER OF WORDS                                  -
C                                                                -
CD     HNAME              HOLLERITH FILE NAME - RZMFLX - (A6)    -
CD     HNAME              HOLLERITH FILE NAME -     .    -(A6) -
CD     HUSE(I)            HOLLERITH USER IDENTIFICATION (A6)     -
CD     IVERS              FILE VERSION NUMBER                    -
CD     MULT               DOUBLE PRECISION PARAMETER             -
CD                            1- A6 WORD IS SINGLE WORD           -
CD                            2- A6 WORD IS OOUBLE PRECISION WORD -
C                                                                -
C-----------------------------------------------------------------
C-----------------------------------------------------------------
CR          SPECIFICATIONS     (1D RECORD)                       -
C                                                                -
CL     NDIM,NGRDUP,NZONE,OUM,DUM,NORD,EFFK,POWER                 -
C                                                                -
CW     8*NUMBER OF WORDS                                         -
C                                                                -
CD     NDIM               NUMBER OF DIMENSIONS                   -
CD     NGROUP             NUMBER OF GRDUPS                       -
CD     NZONE              NUMBER OF GEOMETRIC ZONES              -
CD     OUM                DUMMY, NOT USED                        -
CD     DUM                DUMMY, NOT USED                        -
CD     NORD               NUMBER OF LEGENDRE MOMENTS             -
CD     EFFK               EFFECTIVE MULTIPLICATION FACTOR        -
CD     POWER              POWER IN WATTS TO WHICH FLUX IS NORMALIZED -
C                                                                -
C-----------------------------------------------------------------
C-----------------------------------------------------------------
CR          REGULAR FLUX MOMENTS AVERAGED OVER EACH ZONE         -
C                          (2D RECORD)                           -
C                                                                -
CL     ((FLUX(M,I),M=1,NORD),I=1,NZONE)----NOTE STRUCTURE BELOW--- -
C                                                                -
CW     NORD*NZONE*NUMBER OF WORDS                                -
C                                                                -
C      DO 1 L=1,NGROUP                                           -
C    1 READ (N) *LIST AS ABOVE*                                  -
C                                                                -
CD     FLUX(M,I)          REGULAR FLUX MOMENT AVERAGES FOR EACH  -
CD                        ZONE.  ORDER OF GROUPS IS ACCORDING    -
CD                        TO DECREASING ENERGY.                  -
C                                                                -
C-----------------------------------------------------------------
CEOF                                                             -
```

X. FISSRC File

The FISSRC file is a binary, code-dependent file containing the energy-group total fission source at each spatial fine-mesh point, that is,

$$\sum_g \left(\nu\Sigma_f\right)_{g,i} \phi_{g,i} \quad , \quad i = 1,2,\ldots,IT \quad .$$

The FISSRC file is automatically produced by the Solver Module whenever fissions are present in the problem.

```
C*****************************************************************-
C                          DATE 07/13/87                          -
C                                                                 -
CF          FISSRC                                                -
CE          CODE DEPENDENT FISSION SOURCE                         -
C                                                                 -
C*****************************************************************-

C-----------------------------------------------------------------
C                                                                 -
C                                                                 -
CN                THIS FILE PROVIDES THE FISSION SOURCE           -
C                                                                 -
C                                                                 -
C-----------------------------------------------------------------

C-----------------------------------------------------------------
C                                                                 -
C                                                                 -
CD                        NOTE THAT DOUBLE PRECISION FISSIONS     -
CO                        ARE GIVEN WHEN MULT=2                   -
C                                                                 -
C                                                                 -
C-----------------------------------------------------------------

C-----------------------------------------------------------------
CR          FILE IDENTIFICATION                                   -
C                                                                 -
CL     HNAME,(HUSE(I),I=1,2),IVERS                                -
C                                                                 -
CW     1+3*MULT=NUMBER OF WORDS                                   -
C                                                                 -
CO     HNAME               HOLLERITH FILE NAME - FISSRC - (A6)    -
CD     HUSE(I)             HOLLERITH USER IDENTIFICATION (A6)     -
CD     IVERS               FILE VERSION NUMBER                    -
CO     MULT                DOUBLE PRECISION PARAMETER             -
CO                             1- A6 WORD IS SINGLE WORD           -
CD                             2- A6 WORD IS DOUBLE PRECISION WORD -
C                                                                 -
C-----------------------------------------------------------------
```

```
C-------------------------------------------------------------------------
CR          SPECIFICATIONS      (1D RECORD)                               -
C                                                                         -
CL    NDIM,NGROUP,NINTI,NINTJ,NINTK,ITER,EFFK,POWER,NBLOK                 -
C                                                                         -
CW    9 *NUMBER OF WORDS                                                  -
C                                                                         -
CO    NDIM              NUMBER OF DIMENSIONS                              -
CO    NGROUP            NUMBER OF ENERGY GROUPS                           -
CO    NINTI             NUMBER OF FIRST DIMENSION FINE MESH INTERVALS     -
CO    NINTJ             NUMBER OF SECOND DIMENSION FINE MESH INTERVALS    -
CO    NINTK             SET TO 1                                          -
CD    ITER              OUTER ITERATION NUMBER AT WHICH FISSION WAS       -
CD                      WRITTEN                                           -
CO    EFFK              EFFECTIVE MULTIPLICATION FACTOR                   -
CO    POWER             POWER IN WATTS TO WHICH FISSION IS NORMALIZED     -
CD    NBLOK             SET TO 1                                          -
C                                                                         -
C-------------------------------------------------------------------------


C-------------------------------------------------------------------------
CR          FISSION SOURCE      (2D RECORD)                               -
C                                                                         -
CL    ((FISS(I,J),I=1,NINTI),J=1,NINTJ)                                   -
C                                                                         -
CW    NINTI*NINTJ*MULT=NUMBER OF WORDS                                    -
C                                                                         -
CO    FISS(I,J)         FISSION SOURCE WITHOUT VOLUME AT FINE MESH        -
CD                      POINT (I,J). I.E., NUSIGF*FLUX                    -
C                                                                         -
C-------------------------------------------------------------------------
CEOF
```

APPENDIX B

ONEDANT SAMPLE PROBLEMS

In this appendix are presented the printed output from two sample problems. The first sample problem is a standard k_{eff} calculation with all input by means of card-images. The second sample problem is an edit-only problem in which edits are performed using the scalar fluxes and cross sections from the first sample problem.

I. Sample Problem 1: Standard k_{eff} Calculation

Sample Problem 1 is a standard k_{eff} calculation for a one-dimensional cylindrical reactor. Two energy-group cross sections are used and the scattering is assumed isotropic. The Edit Module is not executed in this sample.

The reactor model consists of a central core of radius 40 cm surrounded by an annular blanket 30 cm thick followed by a shield 30 cm thick. The core consists of 35 volume percent (v/o) fuel, 40 v/o sodium, and 25 v/o steel. The blanket contains 35 v/o blanket fuel, 40 v/o sodium, and 25 v/o steel. The shield consists of 70 v/o sodium and 30 v/o steel.

The first page of the ONEDANT output lists the entire card-image input "deck" supplied to the ONEDANT code for this sample problem. The code provides this card-image input listing unless the third entry on card 1, the entry NOLIST, is set to unity by the user. Note that numerous "comment cards" have been used in the card-image input using the slash (/) as described in Chapter IV.

On page 2 of the problem output are a descriptive summary of the Title Card Control Parameters and the printout of the two title cards provided. This is followed by the message KEY END BLOCK I READ, which indicates that all BLOCK I input has been successfully read and is ready for processing. Next appears the BLOCK I input summary followed by messages that both the BLOCK II and BLOCK III input card-images were successfully read.

On page 3 of the output is a descriptive summary of the BLOCK III card-image input pertaining to cross sections. Included in this summary is a listing of the cross-section types from the card-image library that can be used for edit purposes. These edit cross sections are written to the SNXEDT group-ordered cross-section interface file for use by the Edit Module, if desired. (See Tables Xa and Xb in Ch. V.) The card-image cross-section library, provided directly in the card-image input, is read and the header cards that were included in the library are printed for the user. For this sample problem cross sections for seven isotopes have been provided. Hollerith names have been provided through the NAMES array in BLOCK III and these are listed under the column labelled Isotope Name. The scattering is specified to be isotropic and this is indicated by the entries "p0" under the column labelled Order.

(The label "Order" refers to the Legendre order of expansion for the scattering and, since it is isotropic, only the P_0 Legendre polynomial term appears.)

Page 4 of the output provides the user with a listing of all Nuclide and Material Mixing instructions provided in BLOCK IV of the card-image input. For this problem the nuclides FE (iron), CR (chromium), and NI (nickel) are mixed with atom densities 0.05, 0.016, and 0.01, respectively, to create the material named STEEL. The mixed-oxide, (U-238, PU-239)O_2, material named FUEL is then created using the isotopes PU-239, U-238, and O-16 with atom densities of 0.0051, 0.0206, and 0.0412, respectively. The depleted uranium oxide material named BLKT and the material SODIUM are also created as shown in the output. These specifications are provided in the card-image input through the MATLS= input in BLOCK IV. Through the ASSIGN= input in BLOCK IV the four materials STEEL, FUEL, BLKT, and SODIUM are suitably mixed to create the actual macroscopic mixtures assigned to each of the three ZONES in the sample problem: the core zone (named CORE), the blanket zone (named BLANKT), and the outer shield zone (named SHIELD). The CORE consists of the material FUEL with a volume fraction (density) of 0.35, SODIUM with a volume fraction 0.40, and STEEL with a volume fraction 0.25. The zone BLANKT is identical to the CORE except that the material FUEL is replaced by the material BLKT. The SHIELD zone consists only of the materials SODIUM and STEEL. The subsequent message KEY START MIX CARD XS indicates that the ONEDANT Input Module is to begin creating the working cross-section files MACRXS and SNXEDT and the standard interface files NDXSRF and ZNATDN as described in Chapter II. The last three KEY END messages on the page indicate that the cross-section mixing and processing operation was completed, the BLOCK V Solver Module input was read (and the SOLINP interface file created), and all Input Module operations were completed.

Page 5 of the sample problem output begins the printed output provided by the Solver Module. Pages 5 and 6 present a summary of the input parameters related to, or required by the Solver Module as provided (or defaulted). Note that for the input parameters two columns are provided: one labelled RAW INPUT and one labelled AS DEFAULTED. The RAW INPUT column presents the actual input values provided by the user. If no entry is made in the input, a RAW INPUT value of zero is listed. The AS DEFAULTED column lists the values of the input parameters that the Solver Module actually uses. For example, on page 5 of the output, under the heading CONVERGENCE CONTROLS, the RAW INPUT value for the parameters EPSI is listed as 0. (In the actual card-image input, no entry for EPSI has been provided.) The default value of EPSI (0.0001) is, accordingly, assumed by the Solver Module and this value is provided under the AS DEFAULTED column.

On page 6 of the output are listed the BLOCK I input parameters that are carried over for use by the Solver Module. Here, for example, is indicated that the problem is cylindrical geometry (IGEOM= 2), two energy groups (NGROUP= 2), and S_4 quadrature is to be used (ISN= 4), etc.

Page 7 of the output provides a recap of the assignment of materials to zones in terms of the algorithm described in Chapter V. under the ASGMOD ARRAY description in BLOCK IV. Following this is a map of the problem geometry showing

the coarse-mesh boundary locations, the zone number assigned to each coarse-mesh interval, and other pertinent information. The data storage requirements for the Solver Module are shown next. Below this is a summary of the discrete-ordinates quadrature quantities used for the calculation. For this problem the values printed are built-in S_4 Gaussian quadrature values. Column headings generally refer to quantities depicted in Fig. 2 in Chapter III. The column labelled LI refers to the ξ-level index. The terms BETA PLUS and BETA MINUS refer, respectively to the terms $\alpha_{m+1/2}/w_m$ and $\alpha_{m-1/2}/w_m$ in Eq. (18) of Chapter III. (For spherical geometry BETA PLUS and BETA MINUS refer, respectively, to one-half the value of the terms $\beta_{m+1/2}/w_m$ and $\beta_{m-1/2}/w_m$ in Eq. (21) of Chapter III.)

Page 8 lists the material names of materials for which cross-section data exist on the MACRXS interface file being used by the Solver Module. Next is provided a listing of the ZONE macroscopic cross sections used by the Solver Module. This print is optional and is controlled by the XSECTP entry in the BLOCK V input. In this sample the full table of ZONE macroscopic cross sections has been requested by setting XSECTP= 2. The PRINCIPAL CROSS SECTIONS are defined as the ZONE macroscopic values of χ (fission fraction), $\nu\sigma_f, \sigma_t$, and σ_a. The scattering matrix terms correspond to the term $\sigma^n_{s,h\rightarrow g}$ in Eq. (9)* of Chapter III. The superscript n denotes the Legendre expansion order for the term; the value of n is provided under the column labelled ORDER in the printout. The actual scatter matrix terms for scatter from energy-group h to energy-group g are listed across the page in the sequence

$$\sigma_{s,h\rightarrow g} \quad \sigma_{s,h-1\rightarrow g} \quad \sigma_{s,h-2\rightarrow g}, etc.$$

The entries in the column labelled FIRST GRP in the printout give the value of the energy-group h, namely the first group in the listing which scatters into group g. For downscatter only problems, the value of h is the same as the group number g. For upscatter problems the value of h will not be the same as the value of g. At the bottom of page 8 of the output is geometry and spatial mesh information.

Page 9 of the output provides a summary description of iteration control criteria followed by the iteration monitor print. These items are fully described in Chapter X. It is noted that for this type of problem, a k_{eff} calculation, the eigenvalue is the value of k_{eff}. For the sample problem, then, $k_{eff} = 0.993\ 402$ is provided in the monitor print for outer iteration 4 under the column labelled K-EFF EIGENVALUE.

Page 10 of the output provides a final system edit and balance table print for each energy group and the sum of the groups. The group-dependent quantities are defined and computed as follows:

(1) SOURCE = total inhomogeneous source = QG_g =

* Note that in this discussion, as in Chapter III., a lower case sigma is used to represent a macroscopic cross section.

$$\sum_{i=1}^{IT} Q_i V_i + \sum_{\mu_m<0} w_m|\mu_m| A_{IT+1/2} QR_m + \sum_{\mu_m>0} w_m \mu_m A_{1/2} QL_m \quad ,$$

where Q_i is the inhomogeneous distributed source, QL_m is the left boundary (surface) source, QR_m is the right boundary (surface) source, V_i is the "volume" of spatial mesh interval i, $A_{IT+1/2}$ is the surface area at the rightmost boundary of the system, and $A_{1/2}$ is the surface area at the leftmost boundary of the system;

(2) FISSION SOURCE = total fission source to the group g = FG_g =

$$\frac{1}{k_{eff}} \sum_{h=1}^{NGROUP} \sum_{i=1}^{IT} \chi_{g,i} \left(\nu\sigma_f\right)_{h,i} \phi_{h,i} V_i \quad ;$$

(3) IN SCATTER = in scatter source to group g from other groups =

$$SIN_g = \sum_{\substack{h=1 \\ h\neq g}}^{NGROUP} \sum_{i=1}^{IT} \left(\sigma_{s,h\to g}\right)_i \phi_{h,i} V_i \quad ;$$

(4) SELF SCATTER = self-scatter (within group scatter) in group g =

$$SS_g = \sum_{i=1}^{IT} \left(\sigma^o_{s,g\to g}\right)_i \phi_{g,i} V_i \quad ;$$

(5) OUT SCATTER = out-scatter from group g to all other groups =

$$SOUT_g = \sum_{i=1}^{IT} \left(\sigma'_{t,g}\right)_i \phi_{g,i} V_i - ABG_g - SS_g \quad ;$$

where $\sigma'_{t,g}$ is the total cross section for group g plus any buckling "absorption" plus any "time absorption" $\left(\alpha/v_g\right)$;

(6) ABSORPTION = absorption in group g =

$$ABG_g = \sum_{i=1}^{IT} \left(\sigma'_{a,g}\right)_i \phi_{g,i} V_i \quad ;$$

where $\sigma'_{a,g}$ is the absorption cross section for group g plus any buckling "absorption" plus any "time absorption" $\left(\alpha / v_g\right)$;

(7) RIGHT LEAKAGE = net flow out of system right boundary =

$$RL_g = \sum_{\mu_m>0} w_m \mu_m A_{IT+1/2} \psi_{m,IT+1/2} - \sum_{\mu_m<0} w_m |\mu_m| A_{IT+1/2} \psi_{m,IT+1/2} \quad ;$$

(8) NET LEAKAGE = net flow from system (both boundaries) =

$$NL_g = RL_g + \sum_{\mu_m<0} w_m |\mu_m| A_{1/2} \psi_{m,1/2} - \sum_{\mu_m>0} w_m \mu_m A_{1/2} \psi_{m,1/2} \quad ;$$

and

(9) PARTICLE BALANCE =

$$BAL_g = 1 - \frac{NL_g + ABG_g + SOUT_g}{QG_g + FG_g + SIN_g}$$

Page 11 of the sample problem provides two optional pointwise quantity printouts. The isotropic flux print is provided when the input parameter FLUXP is BLOCK V is set to a value of 1 or 2 (a value of unity is used in this sample problem). The flux values printed are the mesh-interval average fluxes, commonly referred to as the cell-centered flux values. The fission source rate print is provided when the input parameter FISSRP in BLOCK V is set to unity (as in this sample problem). The fission source rate for energy group g and mesh point i is simply the quantity $\left(\nu\sigma_f\right)_{g,i} \phi_{g,i}$, having units of particles per unit time and volume.

Page 12 of the output, the final page, provides the RUN HIGHLIGHTS for the sample problem execution. This is followed by a storage and timing history of the run.

It is to be noted that no Edit Module output appears in the output of this sample problem. The reason for this is that no Edit Module input (BLOCK VI of the card-image input) is provided in the input "deck" and no EDITIT binary interface file (containing previously created Edit Module input) was in existence at the time of the sample problem execution.

```
                GENERALIZED INPUT MODULE RUN ON  09/29/89 WITH VERSION  09-30-89

                              ...LISTING OF CARDS IN THE INPUT STREAM...

  1.      2     0     0
  2.  SAMPLE PROBLEM 1 FOR USER'S MANUAL
  3.  STANDARD K CALCULATION.  ALL INPUT BY MEANS OF CARD-IMAGES
  4.  /   GEOMETRY          - CYLINDRICAL
  5.  /   CROSS SECTIONS    - 2 GROUP, ISOTROPIC SCATTER
  6.  /                       ISOTOPE DATA ON CARDS,  LOS ALAMOS (OTF) FORMAT
  7.  /   MIXING            - ISOTOPES MIXED TO MAKE MATERIALS NAMED STEEL,
  8.  /                       FUEL, BLKT, AND SODIUM
  9.  /                     - MATERIALS ASSIGNED TO MAKE ZONES NAMED CORE,
 10.  /                       BLANKT, AND SHIELD.
 11.  /   SOLVER           - CARD INPUT SUPPLIED
 12.  /   EDITS            - NONE
 13.  /
 14.  /
 15.  / • • • • BLOCK I • • • •
 16.      IGEOM=2, NGROUP=2, ISN=4 NISO=7  MT=4 NZONE=3  IM=3 IT=50   T
 17.  /
 18.  /
 19.  / • • • • BLOCK II (GEOMETRY) • • • •
 20.      XMESH=0.0,40.70,100      XINTS= 20, 2R15
 21.      ZONES= 1 2 3       T
 22.  /
 23.  /
 24.  / • • • • BLOCK III (CROSS SECTIONS) • • • •
 25.      LIB= ODNINP
 26.      MAXORD=0  IHM=8  IHT=4  IHS=5  IFIDO=0  ITITL=1
 27.      NAMES= 'O-16'  'NA-23'  FE  CR  NI  'PU-239'  'U-238'
 28.      EDNAME= FISS
 29.  /
 30.  / •••••• SINCE LIB=ODNINP, THE CROSS SECTION LIBRARY IN CARD-IMAGES
 31.  /  WILL BEGIN IMMEDIATELY FOLLOWING THE BLOCK III TERMINAL 'T'.
 32.  /   NOTE THAT A TITLE CARD PRECEDES EACH CROSS SECTION
 33.  /   BLOCK (SINCE ITITL=1). ••••••
 34.  /
 35.    T
 36.  OXYGEN-16  (O-16)    SAMPLE 2 GROUP LMFBR CROSS SECTIONS
 37.        0.000       0.010       0.000       2.000       1.600       0.000 O16/1
 38.        0.000       0.000       0.000       3.600       3.600       0.390 O16/2
 39.  SODIUM     (NA-23)   SAMPLE 2 GROUP LMFBR CROSS SECTIONS
 40.        0.000       0.002       0.000       1.900       1.500       0.000 NA23/1
 41.        0.000       0.005       0.000       4.000       3.995       0.398 NA23/2
 42.  IRON       (FE)      SAMPLE 2 GROUP LMFBR CROSS SECTIONS
 43.        0.000       0.008       0.000       2.100       1.700       0.000 FE/1
 44.        0.000       0.010       0.000       4.500       4.490       0.392 FE/2
 45.  CHROMIUM   (CR)      SAMPLE 2 GROUP LMFBR CROSS SECTIONS
 46.        0.000       0.013       0.000       2.450       2.150       0.000 CR/1
 47.        0.000       0.020       0.000       5.000       4.980       0.287 CR/2
 48.  NICKEL     (NI)      SAMPLE 2 GROUP LMFBR CROSS SECTIONS
 49.        0.000       0.080       0.000       2.400       2.000       0.000 NI/1
 50.        0.000       0.030       0.000       8.000       7.970       0.320 NI/2
 51.  PLUTONIUM  (PU-239)  SAMPLE 2 GROUP LMFBR CROSS SECTIONS
 52.        1.900       1.950       6.270       4.800       2.000       0.000 PU239/1
 53.        1.800       2.500       4.800      12.000       8.500       0.890 PU239/2
 54.  URANIUM    (U-238)   SAMPLE 2 GROUP LMFBR CROSS SECTIONS
 55.        0.300       0.400       0.900       4.700       3.000       0.000 U238/1
 56.        0.000       0.500       0.000      13.000      12.500       1.300 U238/2
 57.  /
 58.  / ••••••• END OF CROSS SECTION DATA ••••••
 59.  / •••• NOTE THAT THERE IS NO TERMINAL 'T' SINCE THE CROSS SECTIONS ARE
 60.  /       IN LOS ALAMOS (OTF) FORMAT (IFIDO=0) ••••
 61.  /
 62.  /
 63.  / • • • • BLOCK IV (MIXING) • • • •
 64.      MATLS= STEEL, FE .05, CR .018,  NI 0.01;
 65.             FUEL  'PU-239' .005',  'U-238' .0158    'O-16' .0412;
 66.             BLKT  'U-238' 0.0208,  'O-16' .0412;
 67.             SODIUM  'NA-23' .025
 68.      ASSIGN= CORE    FUEL .35, SODIUM .4, STEEL .25;
 69.              BLANKT  BLKT .35, SODIUM .4, STEEL .25;
 70.              SHIELD  SODIUM .7, STEEL .3                T
 71.  /
 72.  /
 73.  / • • • • BLOCK V (SOLVER) • • • •
 74.      IEVT=1   ISCT=0  IBR=0
 75.      NORM=1     FLUXP=1  XSECTP=2  FISSRP=1
 76.      CHI=0.8,0.4; 0.7, 0.3      T
 77.  /
 78.  /
 79.  / • • • • BLOCK VI (EDITS) • • • •
 80.  /  NO INFORMATION SUPPLIED
```

```
*KEY START CASE INPUT  *

                                CASE TITLE

                    2  NHEAD   NUMBER OF TITLE CARDS TO FOLLOW
                    0  NOTTY   0/1  NO/YES  SUPPRESS ON-LINE TERMINAL OUTPUT
                    0  NOLIST  0/1  NO/YES  SUPPRESS INPUT LISTING

                  SAMPLE PROBLEM 1 FOR USER'S MANUAL
                  STANDARD K CALCULATION.  ALL INPUT BY MEANS OF CARD-IMAGES

*KEY END   BLOCK I READ*
```

```
                          ...BLOCK I - CONTROLS AND DIMENSIONS...

                              ...DIMENSIONS (ARRAY NAME = DIMENS)...

      2  IGEOM   1/2/3  PLANE/CYLINDER/SPHERE
      2  NGROUP  NUMBER OF ENERGY GROUPS
      4  ISN     ANGULAR QUADRATURE ORDER
      7  NISO    NUMBER OF INPUT ISOTOPES (FROM ISOTXS, GRUPXS, OR CARDS)
      4  MT      NUMBER OF PERMANENT MATERIALS
      3  NZONE   NUMBER OF ZONES
      3  IM      NUMBER OF COARSE MESH X INTERVALS
     50  IT      NUMBER OF FINE MESH INTERVALS

                                 ...STORAGE...

                                MAXLCM=  140000
                                MAXSCM=   40000

*KEY END   BLOCK II READ-GEOM*

                    2  IGEOM    1/2/3/6/7/8/9/11/14

*KEY END   BLOCK III READ-XS *
```

```
...BLOCK III - CROSS SECTION LIBRARY...

...LIBRARY SOURCE...
LIB=DONINP

...CARD LIBRARY PARAMETERS (ARRAY NAME = CARDS)...

0  MAXORD  MAXIMUM LEGENDRE ORDER TO BE FOUND IN INPUT CROSS SECTIONS
8  IHM     LAST POSITION IN CROSS SECTION TABLE
4  IHT     POSITION OF TOTAL CROSS SECTION
5  IHS     POSITION OF SELF SCATTER CROSS SECTION
0  IFIDO   0/1/2 - OTF/FIXED FIDO/FREE FIDO LIBRARY
1  ITITL   0/1 - NO/YES THERE IS A TITLE CARD BEFORE EACH TABLE
0  I2LP1   0/1 - NO/YES LIBRARY HIGHER ORDER SCATTERING CONTAINS 2L+1 FACTOR
0  SAVBXS  0/1 - NO/YES SAVE BINARY XSLIB FILE (FILENAME=BXSLIB)
1  KWIKRD  0/1 - FULL FIDO READ/QUICK FIDO READ (DEFAULT=QUICK)

...ENERGY STRUCTURE...

GROUP      CHI          VEL          LOWER BOUND   UPPER BOUND   GROUP      CHI          VEL          LOWER BOUND   UPPER BOUND
-----  -----------  -----------  -----------   -----------   -----  -----------  -----------  -----------   -----------
    1  0.00000E+00  0.00000E+00  0.00000E+00   0.00000E+00       2  0.00000E+00  0.00000E+00  0.00000E+00   0.00000E+00

LAST NEUTRON GROUP(LNG) IS NUMBER   2

0  BALXS   -1/0/1 - ADJUST ABSORPTION/NO/ADJUST SELF SCATTER TO FORCE XS BALANCE

...EDIT POSITION NAMES...

                                  CARD
POSITION EDNAME POSITION
-------- ------ --------
       1  CHI
       2  NUSIGF       3
       3  TOTAL        4
       4  ABS          2
       5  FISS         1

*KEY START CARD LIBE READ    *

...HEADER CARDS FROM THE CARD LIBRARY...

ISOTOPE ISOTOPE
NUMBER   NAME   ORDER          HEADER CARD
------  ------- -----      ----------------------
     1. O-16    PO    -   OXYGEN-16  (O-16)    SAMPLE 2 GROUP LMFBR CROSS SECTIONS
     2. NA-23   PO    -   SODIUM     (NA-23)   SAMPLE 2 GROUP LMFBR CROSS SECTIONS
     3. FE      PO    -   IRON       (FE)      SAMPLE 2 GROUP LMFBR CROSS SECTIONS
     4. CR      PO    -   CHROMIUM   (CR)      SAMPLE 2 GROUP LMFBR CROSS SECTIONS
     5. NI      PO    -   NICKEL     (NI)      SAMPLE 2 GROUP LMFBR CROSS SECTIONS
     6. PU-239  PO    -   PLUTONIUM  (PU-239)  SAMPLE 2 GROUP LMFBR CROSS SECTIONS
     7. U-238   PO    -   URANIUM    (U-238)   SAMPLE 2 GROUP LMFBR CROSS SECTIONS

*KEY END   CARD LIBE READ    *

*KEY END   BLOCK IV READ-MATS*
```

```
*****************************************************************************************************************
*****************************************************************************************************************
*
*                                        ... MIXING INSTRUCTIONS ...
*****************************************************************************************************************
*
*           MIX     COMP        DENSITY     COMP        DENSITY ETC.
*           ---     ----        -------     ----        -------
*
*     MATLS
*     ------
*
*       1. STEEL   FE        5.00000E-02. CR        1.60000E-02. NI        1.00000E-02.
*       2. FUEL    PU-239    8.10000E-02. U-238     1.55000E-02. O-16      4.12000E-02.
*       3. BLKT    U-238     2.06000E-02. O-16      4.12000E-02.
*       4. SODIUM  NA-23     2.50000E-02.
*
*     ASSIGN
*     ------
*
*       1. CORE    FUEL      3.50000E-01. SODIUM    4.00000E-01. STEEL     2.50000E-01.
*       2. BLANKT  BLKT      3.50000E-01. SODIUM    4.00000E-01. STEEL     2.50000E-01.
*       3. SHIELD  SODIUM    7.00000E-01. STEEL     3.00000E-01.
*************************
*KEY START MIX CARD XS *
*************************
*************************
*KEY END   MIX CARD XS *
*************************
*****************************************************************************************************************
*****************************
*KEY END   BLOCK V READ-SOLVR*
*****************************
*************************
*KEY END   INPUT MODULE*
*************************
*
*
*****************************************************************************************************************
*****************************************************************************************************************
```

```
                                THIS ONEDANT PROBLEM RUN ON  08/29/89 WITH VERSION  09-30-89      MACHINE 7

*KEY START SOLVER EXECUTION  *

                                          ...TITLE...

                                SAMPLE PROBLEM 1 FOR USER'S MANUAL
                                STANDARD K CALCULATION.  ALL INPUT BY MEANS OF CARD-IMAGES

                                     ...BLOCK V -- SOLVER INPUT...

          RAW     AS
         INPUT  DEFAULTED
         -----  --------

                                     ...REQUIRED INPUT(ARRAY NAME = SOLIN)...

          1         1       IEVT   =1/0/1/2/3/4 - TYPE OF CALCULATION
                                      -1 INHOMOGENEOUS SOURCE WITH FISSION AND/OR UPSCATTER
                                       0 INHOMOGENEOUS SOURCE ALONE
                                       1 K-EFFECTIVE
                                       2 ALPHA OR TIME ABSORPTION SEARCH
                                       3 CONCENTRATION SEARCH
                                       4 DELTA(I.E. DIMENSION) SEARCH
          0         0       ISCT   LEGENDRE ORDER OF SCATTERING
          0         0       ITH    0/1 - DIRECT/ADJOINT - MODE OF CALCULATION (DEFAULT=DIRECT)
          0         1       IBL    0/1/2/3 - LEFT BOUNDARY CONDITION
                                      VACUUM/REFLECTIVE/PERIODIC/WHITE
          0         0       IBR    0/1/2/3 - RIGHT BOUNDARY CONDITION
                                      VACUUM/REFLECTIVE/PERIODIC/WHITE


                                     ...CONVERGENCE CONTROLS(ARRAY NAME = ITER)...

     0.000E+00 1.000E-04  EPSI   INNER ITERATION CONVERGENCE CRITERION (DEFAULT=0.0001)
     0.000E+00 1.000E-04  EPSO   OUTER ITERATION CONVERGENCE CRITERION (DEFAULT=EPSI)
          0         1     IITL   MAXIMUM NUMBER OF INNER ITERATIONS PER GROUP UNTIL (1-LAMBDA).LT.3*EPSO (DEFAULT CALCULATED)
          0        20     IITM   MAXIMUM NUMBER OF INNER ITERATIONS PER GROUP AFTER (1-LAMBDA).LT.3*EPSO (DEFAULT CALCULATED)
          0        20     OITM   MAXIMUM NUMBER OF OUTER ITERATIONS (DEFAULT=20)
          0         0     ITLIM  ITERATION TIME LIMIT (SECONDS)
```

```
...BLOCK V -- SOLVER INPUT (CONTINUED)...

  RAW     AS
INPUT  DEFAULTED
-----  --------

                          ...MISCELLANEOUS PARAMETERS(ARRAY NAME = MISC)...

0.000E+00 0.000E+00  BHGT   BUCKLING HEIGHT
0.000E+00 0.000E+00  BWTH   BUCKLING WIDTH(PLANE GEOM ONLY)
1.000E+00 1.000E+00  NORM   NORMALIZATION FACTOR

    0         0      INFLUX 0/1 NO/YES - READ INPUT FLUX FROM FILE RTFLUX (ATFLUX FOR ADJOINT)
    0         0      INSORS 0/1 NO/YES - READ INPUT SOURCE FROM FILE FIXSRC
    0         0      I2ANG  0/1 NO/YES - DO 2 ANGLE CALCULATION
    0         1      IQUAD  -3/1/2/3 - SOURCE OF QUADRATURE CONSTANTS (DEFAULT=1)
                        FILE SNCONS/BUILT IN P-N/BUILT IN DP-N/CARDS

                             ...OUTPUT CONTROLS(ARRAY NAME = SOLOUT)...

                  1  FLUXP  0/1/2   NONE/ISOTROPIC/ALL MOMENTS - FLUX PRINT
                  2  XSECTP 0/1/2   NONE/PRINCIPAL/ALL - MACROSCOPIC CROSS SECTION PRINT
                  1  FISSRP 0/1     NO/YES - PRINT FINAL FISSION SOURCE RATE
                  0  SOURCP 0/1/2/3 NO/AS READ/NORMALIZED/BOTH - PRINT INHOMOGENEOUS SOURCE
                  0  GEOMP  0/1     NO/YES - PRINT FINE MESH GEOMETRY
                  0  ANGP   0/1     NO/YES - PRINT ANGULAR FLUXES
                  0  RAFLUX 0/1     NO/YES - WRITE ANGULAR FLUXES TO FILE RAFLUX(AAFLUX IF ADJOINT)
                  0  RMFLUX 0/1     NO/YES - WRITE FLUX MOMENTS TO FILE RMFLUX
                  0  BALP   0/1     NO/YES - PRINT COARSE MESH BALANCES

                                ...PARAMETERS INFERRED FROM INPUT ARRAYS...

                               2  INCHI  0/1/2 NONE/ONE CHI/2ONEWISE CHI
                               0  IDEN   0/1 NO/YES - USE DENSITY FACTORS
                               0  IDAN   SOURCE ANISOTROPY
                               0  ISORSE NUMBER OF SOURCE VECTORS INPUT
                               0  ISORSX NUMBER OF SOURCX VECTORS INPUT
                               0  ISORSF NUMBER OF SOURCF VECTORS INPUT
                               0  IDL    -1/0/1 ISOTROPIC/NONE/ALL ANGLES - LEFT BOUNDARY SOURCE
                               0  IDR    -1/0/1 ISOTROPIC/NONE/ALL ANGLES - RIGHT BOUNDARY SOURCE
                               0  ISBEDO 0/N NO/YES - THERE ARE ALBEDOES

                                   ...PARAMETERS FROM BLOCK I...

                             2  IGEOM   1/2/3  PLANE/CYLINDER/SPHERE
                             2  NGROUP  NUMBER OF ENERGY GROUPS
                             4  ISN     ANGULAR QUADRATURE ORDER
                             4  MT      NUMBER OF PERMANENT MATERIALS
                             3  NZONE   NUMBER OF ZONES
                             3  IM      NUMBER OF COARSE MESH X INTERVALS
                            50  IT      NUMBER OF FINE MESH X INTERVALS
```

```
                                    ...MATERIAL ASSIGNMENTS TO ZONES...

*KEY START MATLS TO ZONES    *

                ZONE CROSS SECTION = SUM OVER MATLS IN THE ZONE OF (MATL CROSS SECTION)*( C0 + C1*CMOD )
                                          WITH
                                       CMOD = 0.000000E+00

                          ENTRY   ZONE       MATERIAL        C0              C1
                                  NO. NAME   NO. NAME

                            1     1 CORE     2 FUEL       3.500000E-01   0.000000E+00
                            2     1 CORE     4 SODIUM     4.000000E-01   0.000000E+00
                            3     1 CORE     1 STEEL      2.500000E-01   0.000000E+00
                            4     2 BLANKT   3 BLKT       3.500000E-01   0.000000E+00
                            5     2 BLANKT   4 SODIUM     4.000000E-01   0.000000E+00
                            6     2 BLANKT   1 STEEL      2.500000E-01   0.000000E+00
                            7     3 SHIELD   4 SODIUM     7.000000E-01   0.000000E+00
                            8     3 SHIELD   1 STEEL      3.000000E-01   0.000000E+00


                                       ...ZONE ASSIGNMENT MAP...

*KEY START MAP    *

          R*-----------------*
          R     .     .     V
          R  1  .  2  .  3  V
          R     .     .     V
          R*-----------------*
    R     0.    40.   70.   100.
        0000  0000  0000  0000

 XINTS       20    18    18
 COARSE MESH  1     2     3

                                  ...COMMENTS FROM ONEDANT ON STORAGE REQUIREMENTS...

*KEY START STORAGE NOTE*

                                       STORAGE      REQUIRED   ALLOWED
                                       SMALL CORE       1311     40000
                                       LARGE CORE        870    140000

                                    ...ANGULAR QUADRATURE DATA FOR ONEDANT...

                                          XI-LEVEL  STARTING COSINE
                                              1      -8.083741E-01
                                              2      -9.404323E-01

                                       MU-LEVEL  LEVEL WEIGHT  LEVEL COSINE
                                           1     1.739274E-01  -8.611363E-01
                                           2     3.260726E-01  -3.399810E-01
                                           3     3.260726E-01   3.399810E-01
                                           4     1.739274E-01   8.611363E-01

  M   REFL M   LI   POINT WEIGHT   MU COSINE      WGT*MU       BETA PLUS     BETA MINUS       XI          PHI(DEG)
   1      4     1   1.630363E-01 -3.399810E-01 -5.542925E-02  3.399810E-01  0.000000E+00  0.88113631    131.9719
   2      8     2   1.739274E-01 -8.611363E-01 -1.497752E-01  8.611363E-01  0.000000E+00  0.33998104    156.3027
   3      5     2   1.630363E-01  3.399810E-01  5.542925E-02  1.258643E+00  9.188819E-01  0.33998104    111.1933
   4      1     1   1.630363E-01  3.399810E-01  5.542926E-02  0.000000E+00  3.399810E-01  0.88113631     48.0285
   5      3     2   1.630363E-01  3.399810E-01  5.542925E-02  9.188819E-01  1.258643E+00  0.23998104     68.8067
   6      2     2   1.739274E-01  8.611363E-01  1.497752E-01  0.000000E+00  8.611363E-01  0.33998104     23.6873

                                    ...FISSION GUESS-FLUX ZEROED IN ALL GROUPS...
```

```
...CROSS SECTION RELATED DATA FROM FILE MACRXS  123459082989 VERSION  1 ...

                              ...MATLS AVAILABLE FROM FILE MACRXS...

 1 STEEL     2 FUEL      3 BLKT      4 SODIUM
```

```
                              ...CROSS SECTIONS FOR LEGENDRE ORDERS UP TO PO...

*KEY START MAC CROSS SECTIONS*

                                        G R O U P   1

                              ...PRINCIPAL CROSS SECTIONS...

                    ZONE          CHI          NU*FISSION       TOTAL        ABSORPTION
                   NO.  NAME
                    1   CORE     8.0000E-01    1.6074E-02    1.2398E-01    8.1689E-03
                    2   BLANKT   7.0000E-01    8.4890E-03    1.2378E-01    3.4002E-03
                    3   SHIELD   7.0000E-01    0.0000E+00    8.3710E-02    4.8740E-04

                              ...SCATTERING MATRICES...
                                 (2L+1 NOT INCLUDED)

  ZONE   ORDER  FIRST GRP    CROSS SECTIONS
    1      0       1          9.2787E+02
    2      0       1          9.4882E-02
    3      0       1          8.8070E-02

                                        G R O U P   2

                              ...PRINCIPAL CROSS SECTIONS...

                    ZONE          CHI          NU*FISSION       TOTAL        ABSORPTION
                   NO.  NAME
                    1   CORE     4.0000E-01    8.5880E-03    2.8011E-01    7.5050E-03
                    2   BLANKT   3.0000E-01    0.0000E+00    2.8189E-01    3.9350E-03
                    3   SHIELD   3.0000E-01    0.0000E+00    1.8550E-01    4.2350E-04

                              ...SCATTERING MATRICES...
                                 (2L+1 NOT INCLUDED)

  ZONE   ORDER  FIRST GRP    CROSS SECTIONS
    1      0       2          2.7280E+01    2.5022E-02
    2      0       2          2.7794E+01    2.5825E-02
    3      0       2          1.8508E-01    1.5183E-02
```

```
                              ...GEOMETRY INFORMATION AS EDITED BY ONEDANT...

                                        ...COARSE MESH GEOMETRY...

                          NO. OF INTERVALS      WIDTH        FINE MESH SIZE   LEFT BOUNDARY

                    1           20        4.00000000E+01  2.00000000E+00  0.00000000E+00
                    2           15        3.00000000E+01  2.00000000E+00  4.00000000E+01
                    3           15        3.00000000E+01  2.00000000E+00  7.00000000E+01
                                                                          1.00000000E+02
```

```
...ITERATION CONTROLS AND CRITERIA...

                                   ***ITERATION CRITERIA***

                                        TRANSPORT INNERS

CRITERION    QUANTITY TO TEST                                  VALUE      ACTION TAKEN IF VALUE EXCEEDED
---------    ----------------                                  -----      ------------------------------
   IITL   - INNER ITERATION COUNT                                 1        TERMINATES INNERS
            UNTIL (1.-LAMBDA).LT.3*EPSD
   IITM   - INNER ITERATION COUNT                                20        TERMINATES INNERS
            AFTER (1.-LAMBDA).LT.3*EPSD
   EPSI   - FRACTIONAL PTWISE FLUX CHANGE                  1.00E-04        DOES ANOTHER INNER
            PER INNER

                                     DIFFUSION SUB-OUTERS

CRITERION    QUANTITY TO TEST                                  VALUE      ACTION TAKEN IF VALUE EXCEEDED
---------    ----------------                                  -----      ------------------------------
   DITMD  - SUB-OUTER ITERATION COUNT                           100        TERMINATES SUB-OUTERS
   EPSD   - DIFFUSION LAMBDA-1.0                           1.00E-04        DOES ANOTHER SUB-OUTER
   EPSX   - FRACTIONAL LAMBDA CHANGE PER SUB-OUTER         2.98E-04        DOES ANOTHER SUB-OUTER
   EPSA   - FRACTIONAL PTWISE FLUX CHANGE                  0.99*EPSX       DOES ANOTHER SUB-OUTER
            PER SUB-OUTER

                                   FINAL CONVERGENCE CRITERIA

CRITERION    QUANTITY TO TEST                                  VALUE      ACTION TAKEN IF VALUE EXCEEDED
---------    ----------------                                  -----      ------------------------------
   DITH   - OUTER ITERATION COUNT                                20        QUITS WITH ERROR MESSAGE
   EPSD   - LAMBDA-1.0                                     1.00E-04        DOES ANOTHER OUTER
   EPSX   - MAX FRACTIONAL PTWISE FLUX CHANGE              2.98E-04        DOES ANOTHER OUTER
```

```
...FLUX AND EIGENVALUE CONVERGENCE AS MONITORED BY ONEDANT...

*KEY START ITERATION MONITOR *

 TIME    OUTER          DIFFUSION      K-EFF                          MAX PTWISE    INNERS
 (SEC)    NO.   INNERS SUB-OUTERS  EIGENVALUE    LAMBDA-1           FLUX CHANGE CONVERGED
  0.02      0
  0.03      1       0        8     9.87880E+01  2.71041E-05  8.71213E-04   YES
  0.04      2       2        3     9.83480E+01  4.57919E-03  8.75653E-02  **NO**
  0.04      3       2        2     9.93410E+01 -8.25958E-06  1.83735E+02  **NO**

                        -- INNER ITERATION SUMMARY FOR OUTER ITERATION NO.   4 --

                                              ITER PER MAX FLUX  AT
                                            GROUP  GROUP  CHANGE  MESH
                                              1      3   0.57E-04   47
                                              2      2   0.87E-04   90

                    @@@@@@ ALL CONVERGENCE CRITERIA SATISFIED @@@@@@

 TIME    OUTER          DIFFUSION      K-EFF                          MAX PTWISE    INNERS
 (SEC)    NO.   INNERS SUB-OUTERS  EIGENVALUE    LAMBDA-1           FLUX CHANGE CONVERGED
  0.05      4       8        1     9.93402E+01 -2.96675E-06  1.47891E-04   YES

 PARTICLE BALANCE * -1.81984E-06     TOTAL INNERS ALL OUTERS *    8
```

```
                                    ...GROUP EDIT AND BALANCES UPON CONVERGENCE...

                    ...TITLE--- SAMPLE PROBLEM 1 FOR USER'S MANUAL

                                    ...SYSTEM BALANCE TABLES...(NEUTRONS ONLY)

*KEY START BALANCE TABLES    *
     GP          SOURCE      FISSION SOURCE        IN SCATTER        SELF SCATTER       OUT SCATTER
      1   0.0000000E+00       6.0245109E-01     -3.5527137E-15       1.8227508E+00     4.9157276E-01
      2   0.0000000E+00       3.9754891E-01      4.9157276E-01       3.9595871E+01     2.2737368E-13
  TOTAL   0.0000000E+00       1.0000000E+00      4.9157276E-01       4.1418422E+01     4.9157276E-01

     GP       ABSORPTION      PARTICLE BALANCE     RIGHT LEAKAGE      NET LEAKAGE
      1   1.0871255E-01      -1.4068484E-07       2.1659568E-03      2.1659568E-03
      2   8.3207468E-01      -2.8216069E-08       5.7049228E-02      5.7049228E-02
  TOTAL   9.4078723E-01      -1.8196350E-08       5.9215185E-02      5.9215185E-02
```

```
INTEGRAL SUMMARY INFORMATION
SUMMARY INTEGRAL-K-EFF      9.9340172E-01
INTEGRAL-SOURCE-I             NEUTRON  0.0000000E+00
INTEGRAL-FISSION-I            NEUTRON  1.0000000E+00
INTEGRAL-IN-SCAK-I            NEUTRON  4.9157278E-01
INTEGRAL-SELF-SCAK-I          NEUTRON  4.1418422E+01
INTEGRAL-OUT-SCAK-I           NEUTRON  4.9157278E-01
INTEGRAL-ABSORPTION-I         NEUTRON  9.4078723E-01
INTEGRAL-RIGHT LKAGE-I        NEUTRON  5.9215185E-02
INTEGRAL-NET LKAGE-I          NEUTRON  5.9215185E-02

                                    EXECUTION TIME        0.08 SECONDS
```

...GROUP FLUXES AND FISSION SOURCE RATES BY SPATIAL MESH...

KEY START FLUXES

GROUP 1

...ISOTROPIC FLUX COMPONENT...

ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT	
1	1	4.7570E-03	1	11	3.6007E-03	2	21	1.2723E-03	2	31	1.5403E-04	3	41	3.0295E-05
1	2	4.7424E-03	1	12	3.6190E-03	2	22	9.9187E-04	2	32	1.2865E-04	3	42	2.6381E-05
1	3	4.7051E-03	1	13	3.4237E-03	2	23	7.8768E-04	2	33	1.0420E-04	3	43	2.2860E-05
1	4	4.6499E-03	1	14	3.2181E-03	2	24	6.3301E-04	2	34	8.5762E-05	3	44	1.9760E-05
1	5	4.5771E-03	1	15	2.9933E-03	2	25	5.1254E-04	2	35	7.0604E-05	3	45	1.6993E-05
1	6	4.4874E-03	1	16	2.7679E-03	2	26	4.1705E-04	3	36	5.9743E-05	3	46	1.4505E-05
1	7	4.3813E-03	1	17	2.5077E-03	2	27	3.4048E-04	3	37	5.2212E-05	3	47	1.2248E-05
1	8	4.2590E-03	1	18	2.2295E-03	2	28	2.7863E-04	3	38	4.5821E-05	3	48	1.0185E-05
1	9	4.1212E-03	1	19	1.9466E-03	2	29	2.2841E-04	3	39	3.9842E-05	3	49	8.2077E-06
1	10	3.9682E-03	1	20	1.8148E-03	2	30	1.8748E-04	3	40	3.4785E-05	3	50	6.3103E-06

...FISSION SOURCE RATE...

ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT	
1	1	7.6488E-05	1	11	6.1084E-05	2	21	8.2582E-06	2	31	9.9952E-07	3	41	0.0000E+00
1	2	7.6231E-05	1	12	5.8174E-05	2	22	6.4350E-06	2	32	8.2185E-07	3	42	0.0000E+00
1	3	7.5833E-05	1	13	5.5035E-05	2	23	5.1112E-06	2	33	6.7815E-07	3	43	0.0000E+00
1	4	7.4748E-05	1	14	5.1882E-05	2	24	4.1078E-06	2	34	5.5851E-07	3	44	0.0000E+00
1	5	7.3579E-05	1	15	4.8116E-05	2	25	3.3259E-06	2	35	4.5815E-07	3	45	0.0000E+00
1	6	7.2133E-05	1	16	4.4332E-05	2	26	2.7062E-06	3	36	0.0000E+00	3	46	0.0000E+00
1	7	7.0428E-05	1	17	4.0309E-05	2	27	2.2084E-06	3	37	0.0000E+00	3	47	0.0000E+00
1	8	6.8481E-05	1	18	3.5999E-05	2	28	1.8081E-06	3	38	0.0000E+00	3	48	0.0000E+00
1	9	6.6246E-05	1	19	3.1291E-05	2	29	1.4821E-06	3	39	0.0000E+00	3	49	0.0000E+00
1	10	6.3787E-05	1	20	2.5954E-05	2	30	1.2185E-06	3	40	0.0000E+00	3	50	0.0000E+00

GROUP 2

...ISOTROPIC FLUX COMPONENT...

ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT	
1	1	2.4258E-02	1	11	1.9897E-02	2	21	9.8559E-03	2	31	3.3104E-03	3	41	1.2978E-03
1	2	2.4171E-02	1	12	1.8855E-02	2	22	8.9004E-03	2	32	2.9585E-03	3	42	1.1331E-03
1	3	2.3990E-02	1	13	1.7960E-02	2	23	8.0172E-03	2	33	2.6410E-03	3	43	1.0115E-03
1	4	2.3726E-02	1	14	1.7021E-02	2	24	7.2089E-03	2	34	2.3607E-03	3	44	8.9298E-04
1	5	2.3378E-02	1	15	1.6043E-02	2	25	6.4878E-03	2	35	2.1138E-03	3	45	7.7713E-04
1	6	2.2949E-02	1	16	1.5033E-02	2	26	5.7988E-03	3	36	1.9288E-03	3	46	6.8379E-04
1	7	2.2441E-02	1	17	1.4001E-02	2	27	5.1893E-03	3	37	1.7890E-03	3	47	6.5245E-04
1	8	2.1856E-02	1	18	1.2955E-02	2	28	4.6415E-03	3	38	1.6515E-03	3	48	4.4221E-04
1	9	2.1204E-02	1	19	1.1907E-02	2	29	4.1467E-03	3	39	1.5170E-03	3	49	3.3118E-04
1	10	2.0482E-02	1	20	1.0867E-02	2	30	3.7068E-03	3	40	1.3857E-03	3	50	2.1498E-04

...FISSION SOURCE RATE...

ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT	
1	1	2.0784E-04	1	11	1.6878E-04	2	21	0.0000E+00	2	31	0.0000E+00	3	41	0.0000E+00
1	2	2.0709E-04	1	12	1.6155E-04	2	22	0.0000E+00	2	32	0.0000E+00	3	42	0.0000E+00
1	3	2.0556E-04	1	13	1.5388E-04	2	23	0.0000E+00	2	33	0.0000E+00	3	43	0.0000E+00
1	4	2.0328E-04	1	14	1.4583E-04	2	24	0.0000E+00	2	34	0.0000E+00	3	44	0.0000E+00
1	5	2.0030E-04	1	15	1.3745E-04	2	25	0.0000E+00	2	35	0.0000E+00	3	45	0.0000E+00
1	6	1.9662E-04	1	16	1.2881E-04	2	26	0.0000E+00	3	36	0.0000E+00	3	46	0.0000E+00
1	7	1.9228E-04	1	17	1.1996E-04	2	27	0.0000E+00	3	37	0.0000E+00	3	47	0.0000E+00
1	8	1.8728E-04	1	18	1.1100E-04	2	28	0.0000E+00	3	38	0.0000E+00	3	48	0.0000E+00
1	9	1.8167E-04	1	19	1.0202E-04	2	29	0.0000E+00	3	39	0.0000E+00	3	49	0.0000E+00
1	10	1.7549E-04	1	20	9.3108E-05	2	30	0.0000E+00	3	40	0.0000E+00	3	50	0.0000E+00

GROUP SUM

...ISOTROPIC FLUX COMPONENT...

ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT	
1	1	2.9015E-02	1	11	2.3488E-02	2	21	1.1128E-02	2	31	3.4844E-03	3	41	1.2881E-03
1	2	2.8913E-02	1	12	2.2474E-02	2	22	9.8821E-03	2	32	3.0832E-03	3	42	1.1595E-03
1	3	2.8694E-02	1	13	2.1384E-02	2	23	8.8048E-03	2	33	2.7452E-03	3	43	1.0344E-03
1	4	2.8378E-02	1	14	2.0238E-02	2	24	7.8399E-03	3	34	2.4464E-03	3	44	9.1272E-04
1	5	2.7956E-02	1	15	1.9036E-02	2	25	6.9803E-03	2	35	2.1844E-03	3	45	7.9413E-04
1	6	2.7438E-02	1	16	1.7781E-02	2	26	6.2138E-03	3	36	1.9883E-03	3	46	6.7829E-04
1	7	2.6822E-02	1	17	1.6509E-02	2	27	5.5298E-03	3	37	1.8412E-03	3	47	5.8470E-04
1	8	2.6117E-02	1	18	1.5198E-02	2	28	4.9201E-03	3	38	1.6971E-03	3	48	4.9238E-04
1	9	2.5329E-02	1	19	1.3883E-02	2	29	4.3771E-03	3	39	1.5568E-03	3	49	3.9837E-04
1	10	2.4450E-02	1	20	1.2482E-02	2	30	3.8939E-03	3	40	1.4205E-03	3	50	2.9130E-04

...FISSION SOURCE RATE...

ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT		ZONE	PT	
1	1	2.8431E-04	1	11	2.2888E-04	2	21	8.2582E-06	2	31	9.9952E-07	3	41	0.0000E+00
1	2	2.8332E-04	1	12	2.1972E-04	2	22	6.4350E-06	2	32	8.2185E-07	3	42	0.0000E+00
1	3	2.8118E-04	1	13	2.0892E-04	2	23	5.1112E-06	2	33	6.7815E-07	3	43	0.0000E+00
1	4	2.7803E-04	1	14	1.9752E-04	2	24	4.1078E-06	2	34	5.5851E-07	3	44	0.0000E+00
1	5	2.7388E-04	1	15	1.8587E-04	2	25	3.3259E-06	2	35	4.5815E-07	3	45	0.0000E+00
1	6	2.6878E-04	1	16	1.7314E-04	2	26	2.7062E-06	3	36	0.0000E+00	3	46	0.0000E+00
1	7	2.6270E-04	1	17	1.6027E-04	2	27	2.2084E-06	3	37	0.0000E+00	3	47	0.0000E+00
1	8	2.5574E-04	1	18	1.4700E-04	2	28	1.8081E-06	3	38	0.0000E+00	3	48	0.0000E+00
1	9	2.4792E-04	1	19	1.3331E-04	2	29	1.4821E-06	3	38	0.0000E+00	3	49	0.0000E+00
1	10	2.3927E-04	1	20	1.1908E-04	2	30	1.2185E-06	3	40	0.0000E+00	3	50	0.0000E+00

...INTERFACE FILE SNCONS WRITTEN...

...INTERFACE FILE RTFLUX WRITTEN...

RUN HIGHLIGHTS

*KEY START RUN HIGHLIGHTS *

```
    ALL MODULES ARE TENTATIVELY GO.
     INTERFACE FILE GEODST WRITTEN.
       CROSS SECTIONS FROM CARDS.
    INTERFACE MIXING FILES WRITTEN.
     INTERFACE FILE ASGMAT WRITTEN
    XS FILES MACRXS,SNXEOT WRITTEN.
 *LEFT BOUNDARY CONDITION OVERRIDDEN*
     INTERFACE FILE SOLINP WRITTEN.
  EDIT MODULE EXECUTION SUPPRESSED.
NEITHER EDITIT NOR EDIT CARDS EXIST.
       START SOLVER EXECUTION.
     ALL CONVERGENCE CRITERIA MET.
     INTERFACE FILE SNCONS WRITTEN.
     INTERFACE FILE RTFLUX WRITTEN.
```

STORAGE AND TIMING HISTORY

MODULE	SCM WORDS	SCM LIMIT	LCM WORDS	LCM LIMIT	CPU SECONDS	CHARGE SECONDS
0	0		0	0	1.8	4.8
100	3851	40000	0	0	1.4	3.7
101	0		0	0	0.4	0.7
102	30	40000	5	140000	0.0	0.2
103	49	40000	0	0	0.0	0.2
104	2073	40000	0	0	0.1	0.8
105	0		0	0	0.0	0.0
106	0		0	0	0.0	0.0
107	188	40000	92	140000	0.0	0.3
108	299	40000	7	140000	0.0	0.2
109	0		0	0	0.0	0.0
112	0		0	0	0.0	0.2
200	1311	40000	870	140000	0.1	1.1
201	0		0	0	0.0	0.1
202	281	40000	0	0	0.0	0.0
203	281	40000	0	0	0.0	0.0
204	1311	40000	39	140000	0.0	0.0
205	1288	40000	0	0	0.0	0.0
1205	0		0	0	0.0	0.0
206	1297	40000	0	0	0.0	0.2
207	182	40000	0	0	0.0	0.8
400	0		0	0	0.0	0.0

...EXECUTION TERMINATED...

II. Sample Problem 2: Edit-Only Run

Sample Problem 2 is an edit calculation for the problem specified in the first sample problem. The edits are performed using the scalar fluxes produced during the execution of the Solver Module in Sample Problem 1.

Sample Problem 2 illustrates the way in which the modular construction of ONEDANT can be used to execute the Edit Module independently and separately from a previous Solver Module execution. The card-image input is shown on the first page of the printed output provided by ONEDANT for Sample Problem 2. Only BLOCK I and BLOCK VI input data are present in the card-image input. The geometry, cross-section, material mixing, and Solver portions of the code are thus not executed. Instead, the binary interface files GEODST (geometry), NDXSRF and ZNATDN (mixing), SNXEDT (cross sections for edits), and RTFLUX (scalar fluxes), which were created during prior execution of Sample Problem 1, were saved and made available to ONEDANT at the time of execution of the second sample problem. This procedure is described in Chapter XIII.

It should be noted that the execution of the Edit Module could have been included in Sample Problem 1 simply by including the BLOCK VI input in the input "deck" for that problem.

Page 1 of the output displays the card-image input for this sample problem. Note the use of comment cards as denoted by the slash(/) as the first entry on each such card-image. Also provided on page 1 are a summary of the Title Card Control Parameters and the printout of the two title cards. Next appears the BLOCK I input summary. Following this appears the message "KEY END BLOCK VI READ-EDIT." This message is written after the BLOCK VI card-image input has been successfully read and processed. The final message, "KEY END INPUT MODULE" indicates that all Input Module operations are completed.

Page 2 of the output lists the Edit Module input as provided in the BLOCK VI card-image input. Chapter VII provides a detailed description of the BLOCK VI input parameters and the edit quantities produced. Both "point" and "zone" edits are requested. Referring to the card-image input, the points at which edits are desired are provided in the POINTS array input where the mesh points 1 through 10 and 46 through 50 are specified (note the use of the linear interpolate operator described in Chapter IV in specifying the POINTS array input). Since no edit zones are explicitly specified (no EDZONE array input is specified), the code will assume that the edit zones are the same as the coarse-mesh intervals specified in Sample Problem 1. Since the parameter IGRPED is input with a value of zero, only the energy-group totals for each edit quantity are to be printed. Resident macroscopic, resident constituent, and material cross-section response functions are specified using both the $\nu\Sigma_f$ and Σ_f cross-section types.

Page 3 provides the desired edit output, or reaction rates, for the material "FUEL," $(U,Pu)O_2$, specified in Sample Problem 1 and also for the resident macroscopic cross sections at each spatial mesh point requested. Also provided are the edit zone (defaulted to coarse-mesh interval) sums requested.

Page 4 provides the desired edit output, or reaction rate information, for the "CONSTITUENTS" PU-239 and U-238 both at the desired space-points and as sums (integrals) over the edit zones (coarse-mesh intervals).

Page 5 shows the RUN HIGHLIGHTS and STORAGE/TIMING HISTORY for this sample problem.

```
GENERALIZED INPUT MODULE RUN ON  09/29/89 WITH VERSION  09.30.89

                    ...LISTING OF CARDS IN THE INPUT STREAM...

    1.      2     0     0
    2.  SAMPLE PROBLEM 2 FOR USER'S MANUAL
    3.  EDITS (ONLY) ON PREVIOUS SAMPLE PROBLEM 1
    4.  /   GEOMETRY        - FROM PREVIOUS GEODST STANDARD INTERFACE FILE
    5.  /   CROSS SECTIONS  - FROM PREVIOUS MACRXS AND SNXEDT INTERFACE FILES
    6.  /   MIXING          - FROM PREVIOUS NDXSRF AND ZNATON INTERFACE FILES
    7.  /   SOLVER          - NO INFORMATION SUPPLIED.  USING PREVIOUS RTFLUX
    8.  /                     FILE FROM SOLVER RUN.
    9.  /   EDITS           - CARD INPUT SUPPLIED
   10.  /
   11.  /
   12.  / * * * * BLOCK I * * * *
   13.      IGEOM=2, NGROUP=2. ISN=4 NISO=7  MT=4 NZONE=3  IM=3 IT=50   T
   14.  /
   15.  /
   16.  / * * * * BLOCK VI (EDITS) * * * *
   17.      PTED=1  ZNED=1  POINTS=811.10 3148.50   IGRPED=0
   18.      EDXS= NUSIGF FISS   RESONT=1   EDMATS=FUEL
   19.      EDCONS= 'PU-239'  'U-238'      T

                                CASE TITLE

*KEY START CASE INPUT  *
                          2  NMEAD   NUMBER OF TITLE CARDS TO FOLLOW
                          0  NOTTY   0/1  NO/YES  SUPPRESS ON-LINE TERMINAL OUTPUT
                          0  NOLIST  0/1  NO/YES  SUPPRESS INPUT LISTING

                    SAMPLE PROBLEM 2 FOR USER'S MANUAL
                    EDITS (ONLY) ON PREVIOUS SAMPLE PROBLEM 1
*KEY END   BLOCK I READ*

                        ...BLOCK I - CONTROLS AND DIMENSIONS...

                              ...DIMENSIONS (ARRAY NAME = DIMENS)...

                    2  IGEOM   1/2/3  PLANE/CYLINDER/SPHERE
                    2  NGROUP  NUMBER OF ENERGY GROUPS
                    4  ISN     ANGULAR QUADRATURE ORDER
                    7  NISO    NUMBER OF INPUT ISOTOPES (FROM ISOTXS, GRUPXS, OR CARDS)
                    4  MT      NUMBER OF PERMANENT MATERIALS
                    3  NZONE   NUMBER OF ZONES
                    3  IM      NUMBER OF COARSE MESH X INTERVALS
                   50  IT      NUMBER OF FINE MESH INTERVALS

                                   ...STORAGE...

                                  MAXLCM=  140000
                                  MAXSCM=   40000

*KEY END   BLOCK VI READ-EDIT*
*KEY END   INPUT MODULE*
```

```
...EDIT OUTPUT...

...BLOCK VI - EDIT SPECIFICATION DATA...

*KEY START EDIT OUTPUT *

CROSS SECTION BALANCING (BALXS*1)
      OR
TRANSPORT CORRECTION (TRCOR*DIAG, CESARO, OR BHS)
WILL NOT BE REFLECTED IN EDITS

...INPUT CONTROL INTEGERS...

      1  PTED      0/1     NO/YES - POINT EDITS DESIRED
      1  ZNED      0/1     NO/YES - ZONE EDITS DESIRED

      0  AJED      0/1     DIRECT/ADJOINT EDIT(USE RTFLUX/ATFLUX FILE)

      0  IGRPED    0/1/2/3 PRINT TOTALS ONLY/PRINT BROAD GROUPS ONLY/SAME AS 1/PRINT ALL GROUPS AND TOTALS
      0  BYVOLP    0/1     NO/YES - MULTIPLY POINT REACTION RATES BY MESH VOLUMES
      0  RZFLUX    0/1     NO/YES - WRITE THE RZFLUX FILE (ZONE AVERAGE FLUX FILE)
      0  RZMFLX    0/1     NO/YES - WRITE THE RZMFLX FILE (ZONE AVERAGE FLUX MOMENTS FILE)

...FLOATING PARAMETERS...

0.000000E+00  POWER     0/P     NO/NORMALIZE ALL RESULTS, INCLUDING FLUX FILES, TO P MEGAWATTS
2.100000E+02  MEVPER    MEV PER FISSION (DEFAULT: 210 MEV)


...ENERGY RELATED EDIT INFORMATION...

   2  NUMBER OF FINE NEUTRON GROUPS
   0  NUMBER OF FINE GAMMA GROUPS
   2  TOTAL NUMBER OF FINE GROUPS
   2  TOTAL NUMBER OF BROAD GROUPS


...SPACE RELATED EDIT INFORMATION...

  18  NUMBER OF POINTS TO EDIT
   3  NUMBER OF ZONES TO EDIT
   0  0/1  NO/YES  DENSITY FACTORS WERE INPUT
```

```
...EDIT OUTPUT...

...EDIT SPECIFICATION DATA(CONTINUED)...

...DESCRIPTION OF CROSS SECTION EDITS...

ISOTOPE  NO.      MATERIAL  NO.      CONSTITUENT  NO.
-------  ---      --------  ---      -----------  ---

-NONE-            FUEL       2          PU-239     6
                  RESONT                U-238      7

     REACTION RATES WILL BE FORMED FOR EACH
          OF THE ABOVE
     USING THE CROSS SECTION TYPES SHOWN BELOW

                        POSITION
               TYPE      NO.
              ------    ---
              NUSIGF      2
              FISS        5
```

*KEY START MATERIALS *

REACTION RATES
FROM
MATERIALS

...POINT EDIT FOR THE SUM OF THE NEUTRON GROUPS ...

... PERMANENT MATERIAL 2 - FUEL ...

POINT	AV RAD	NUSIGF	FISS
1	1.0000	8.12311E-04	2.66181E+04
2	3.0000	8.09497E-04	2.65237E+04
3	5.0000	8.03380E-04	2.63234E+04
4	7.0000	7.94363E-04	2.60282E+04
5	9.0000	7.82500E-04	2.56398E+04
6	11.0000	7.67678E-04	2.51811E+04
7	13.0000	7.50577E-04	2.45947E+04
8	15.0000	7.30689E-04	2.39438E+04
9	17.0000	7.08341E-04	2.32120E+04
10	19.0000	6.83642E-04	2.24035E+04
48	91.0000	1.69187E-05	5.82481E-06
47	93.0000	1.40884E-05	4.68360E-06
48	95.0000	1.12922E-05	3.75422E-06
49	97.0000	8.48378E-06	2.81997E-06
50	99.0000	5.58269E-06	1.84478E+06

... RESIDENT MACROSCOPIC ...

POINT	AV RAD	NUSIGF	FISS
1	1.0000	2.84309E-04	9.31563E-05
2	3.0000	2.83324E-04	9.28330E-05
3	5.0000	2.81163E-04	9.21319E-05
4	7.0000	2.78027E-04	9.10887E-05
5	9.0000	2.73875E-04	8.97393E-05
6	11.0000	2.68757E-04	8.80638E-05
7	13.0000	2.62702E-04	8.60814E-05
8	15.0000	2.55743E-04	8.38033E-05
9	17.0000	2.47918E-04	8.12421E-05
10	19.0000	2.39275E-04	7.84124E-05
48	91.0000	0.00000E+00	0.00000E+00
47	93.0000	0.00000E+00	0.00000E+00
48	95.0000	0.00000E+00	0.00000E+00
49	97.0000	0.00000E+00	0.00000E+00
50	99.0000	0.00000E+00	0.00000E+00

...ZONE EDIT FOR THE SUM OF THE NEUTRON GROUPS ...

... PERMANENT MATERIAL 2 - FUEL ...

ZONE	VOLUME	NUSIGF	FISS
1	8.0265E+03	2.76872E+00	8.07889E-01
2	1.0367E+04	1.38890E+00	4.58664E-01
3	1.6022E+04	4.04088E-01	1.34308E-01
SUM	3.1416E+04	4.85971E+00	1.50063E+00

... RESIDENT MACROSCOPIC ...

ZONE	VOLUME	NUSIGF	FISS
1	8.0265E+03	9.89083E-01	3.17751E-01
2	1.0367E+04	2.43492E-02	8.11838E-03
3	1.6022E+04	0.00000E+00	0.00000E+00
SUM	3.1416E+04	9.93402E-01	3.25887E-01

*KEY START CONSTITUENTS *

REACTION RATES
FROM
CONSTITUENTS

...POINT EDIT FOR THE SUM OF THE NEUTRON GROUPS ...

... CONSTITUENT ISOTOPE 6 - PU-238 ...

POINT	AV RAD	NUSIGF	FISS
1	1.0000	2.61063E-04	8.54143E-05
2	3.0000	2.60189E-04	8.51148E-05
3	5.0000	2.58210E-04	8.44743E-05
4	7.0000	2.58324E-04	8.45310E-05
5	9.0000	2.51527E-04	8.22901E-05
6	11.0000	2.48847E-04	8.07605E-05
7	13.0000	2.41310E-04	7.89508E-05
8	15.0000	2.34948E-04	7.68717E-05
9	17.0000	2.27798E-04	7.45348E-05
10	19.0000	2.19900E-04	7.19541E-05
48	91.0000	0.00000E+00	0.00000E+00
47	93.0000	0.00000E+00	0.00000E+00
48	95.0000	0.00000E+00	0.00000E+00
49	97.0000	0.00000E+00	0.00000E+00
50	99.0000	0.00000E+00	0.00000E+00

... CONSTITUENT ISOTOPE 7 - U-238 ...

POINT	AV RAD	NUSIGF	FISS
1	1.0000	2.32259E-06	7.74197E-06
2	3.0000	2.31546E-06	7.71820E-06
3	5.0000	2.29726E-06	7.65761E-06
4	7.0000	2.27031E-06	7.56770E-06
5	9.0000	2.23478E-06	7.44927E-06
6	11.0000	2.19100E-06	7.30332E-06
7	13.0000	2.13915E-06	7.13051E-06
8	15.0000	2.07947E-06	6.93156E-06
9	17.0000	2.01217E-06	6.70722E-06
10	19.0000	1.93749E-06	6.45829E-06
48	91.0000	0.00000E+00	0.00000E+00
47	93.0000	0.00000E+00	0.00000E+00
46	95.0000	0.00000E+00	0.00000E+00
49	97.0000	0.00000E+00	0.00000E+00
50	99.0000	0.00000E+00	0.00000E+00

...ZONE EDIT FOR THE SUM OF THE NEUTRON GROUPS ...

... CONSTITUENT ISOTOPE 6 - PU-239 ...

ZONE	VOLUME	NUSIGF	FISS
1	8.0269E+03	8.93230E-01	2.93476E-01
2	1.0367E+04	0.00000E+00	0.00000E+00
3	1.6022E+04	0.00000E+00	0.00000E+00
SUM	3.1418E+04	8.93230E-01	2.82476E-01

... CONSTITUENT ISOTOPE 7 - U-238 ...

ZONE	VOLUME	NUSIGF	FISS
1	8.0269E+03	7.58228E-02	2.52743E-02
2	1.0367E+04	2.43492E-02	8.11639E-03
3	1.6022E+04	0.00000E+00	0.00000E+00
SUM	3.1418E+04	1.00172E-01	3.33907E-02

```
*KEY START RUN HIGHLIGHTS    *
```

RUN HIGHLIGHTS

```
ALL MODULES ARE TENTATIVELY GO.
INTERFACE FILE EOITIT WRITTEN.
SOLVER MODULE EXECUTION SUPPRESSED.
NEITHER SOLINP NOR SOLVER CARDS EXIST
START EOIT EXECUTION.
EOITS COMPLETED.
```

STORAGE AND TIMING HISTORY

MODULE	SCM WORDS	SCM LIMIT	LCM WORDS	LCM LIMIT	CPU SECONDS	CHARGE SECONDS
0	0	0	0	0	0.7	1.8
100	3681	40000	0	0	0.8	1.3
101	0	0	0	0	0.3	0.8
102	0	0	0	0	0.0	0.0
103	0	0	0	0	0.0	0.0
104	0	0	0	0	0.0	0.0
105	0	0	0	0	0.0	0.0
106	0	0	0	0	0.0	0.0
107	0	0	0	0	0.0	0.0
108	0	0	0	0	0.0	0.0
109	3687	40000	0	0	0.0	0.2
300	0	0	0	0	0.1	0.2
301	828	40000	269	140000	0.1	0.2
302	0	0	0	0	0.0	0.0
400	0	0	0	0	0.0	0.0

```
...EXECUTION TERMINATED...
```

APPENDIX C

FILE DESCRIPTIONS FOR SPECIAL EDIT MODULE ASCII OUTPUT FILES

In this appendix are presented file descriptions for the two ASCII files EDTOUT and EDTOGX optionally produced by the Edit Module of ONEDANT/TWODANT. These two files are especially suited to the user for preparing data file input for graphics packages.

I. Description of the EDTOUT File.

EDTOUT is a special ASCII file optionally prepared by the Edit Module of ONEDANT/TWODANT containing geometric and edit information which can be selectively processed by the user.

This description of the EDTOUT file describes the format and construct of the file. The term "section" shall be used to refer to grouped data. The term "card image" or "card" shall be used in the same context that was described in Ch. IV of this manual.

1. NUMBER-OF-TITLE-RECORDS (CARDS) SECTION (Format I6)

This section (card) contains the single word NTITLE, where NTITLE is the number of title records included in the file.

2. TITLE CARD SECTION (Format 10A8)

The title cards from the problem are given as individual records. This section is read as follows:

```
      DIMENSION HTITLE (10, NTITLE)
      DO 10 N=1,NTITLE
      READ (NINP,20) (HTITLE(I,N), I=1,10)
10    CONTINUE
20    FORMAT(10A8)
```

3. EDIT SPECIFICATION SECTION (Format 12I6)

The edit specification section is a single card-image containing those parameters needed to process the edit-related data sections. The entries in this section are ordered as follows:

(1)	IZNED	Zone Edits present? 0/1=no/yes
(2)	NZNS	Number of Edit Zones
(3)	IPTED	Point Edits present? 0/1=no/yes
(4)	NIPES	Number of points for point edits. -1/0/n = all points / no points / selected points
(5)	NEDISO	Number of isotopes selected for edits. (Corresponds to EDISOS edits in ONEDANT)
(6)	MACRO	Number of mixed macroscopic cross sections selected for edits. This corresponds to number of EDMATS entries PLUS the number of RESDNT edits in ONEDANT.
(7)	NCONS	Number of constituent cross sections selected for edits. This corresponds to the number of EDCONS entries in ONEDANT
(8)	NXSTYP	The number of cross-section types (positions) e.g. ABS, NUSIGF, in the edits. This corresponds to the number of EDXS entries in ONEDANT
(9)	NSRF	The number of response functions in the edits. This corresponds to the number of RSFNAM values in ONEDANT.
(10)	NBG	The number of Edit Broad Groups in the edits.

4. GEOMETRIC SPECIFICATION SECTION (Format 12I6)

The geometric specification section contains those parameters needed to process the geometry-related data blocks. The 12 entries in this section are ordered as follows:

(1)	IDIMEN	The geometry-dimension of the problem 1/2 = one-dimensional/two-dimensional
(2)	IGEOM	Geometry of the problem. 1=slab, 2=cylinder, etc.
(3)	IM	Number of coarse radial mesh intervals
(4)	IT	Total number of fine mesh intervals in radial direction
(5)	JM	Number of coarse axial mesh intervals (=1 for IDIMEN=1)
(6)	JT	Total number of axial fine mesh intervals (=1 for IDIMEN=1)
(7)	NDUM1	Not used by ONEDANT/TWODANT
(8)	NDUM2	Not used by ONEDANT/TWODANT
(9)	NDUM3	Not used
(10)	NDUM4	Not used
(11)	NDUM5	Not used
(12)	NDUM6	Not used

ASIDE: In order to simplify the dimensions of some of the following data blocks, let us define the following internal parameters in terms of the preceding parameters found on the EDTOUT file :

(1) NBGP1 = NBG + 1
(2) NXSTOT = NEDISO + MACRO + NCONS. Note that an index I that runs from 1 to NXSTOT runs in the order indicated, namely isotopes, then mixed macroscopic, then constituents.
(3) IMP1 = IM + 1
(4) JMP1 = JM + 1
(5) ITP1 = IT + 1
(6) JTP1 = JT + 1
(7) NIPE = a parameter which reflects the actual number of edit points. It assumes the following value as determined by the parameter NIPES:
NIPE = IT*JT, if NIPES .LT. 0,
NIPE = 0, if NIPES .EQ. 0,.
NIPE=NIPES, if NIPES .GT. 0

DATA BLOCKS
In reading the following data blocks, the user must assign data to be read to its own storage as defined by the parameters just defined. For the following data blocks we will give only a generic name for the data, the number of words in the data block, and the format-type of the block. REAL, INTEGER, and CHARACTER*8 data blocks are required. The presence or absence of a block will be indicated below by the IFF (if and only if) notation.

END ASIDE

5. GROUP ENERGY BOUNDS SECTION

ENERGY (NBGP1), 6E12

Using the data block ENERGY as an example, the block is to be read as follows:

```
      DIMENSION ENERGY(NBGP1)
      READ (NINP,400) ENERGY
400   FORMAT(6E12.5)
```

6. COARSE RADIAL MESH BOUNDARIES SECTION
XMESH (IMP1) , 6E12

7. NUMBER OF FINE RADIAL MESHES PER COARSE MESH SECTION
   IHX (IM) , 12I6

8. COARSE AXIAL MESH BOUNDARIES SECTION [IFF (IDIMEN .GT. 1)]
   YMESH (IMP1) , 6E12

9. NUMBER OF FINE AXIAL MESHES PER COARSE MESH SECTION
   [IFF (IDIMEN .GT. 1)]
   IHY (IM) , 12I6

10. POINTS EDITED SECTION [IFF (NIPES .GT. 0)]
    KPT (NIPE) , 12I6

11. ZONE VOLUMES SECTION [IFF (NZNS .GT. 0)]
    VZ (NZNS) , 6E12

12. NAMES OF ISOTOPES SECTION [IFF (NEDISO .GT. 0)]
    HISO (NEDISO) , 9A8

13. NAMES OF MACROSCOPIC-EDITS SECTION [IFF (MACRO .GT. 0)]
    HMACR (MACRO) , 9A8

14. NAMES OF CONSTITUENTS SECTION [IFF (NCONS .GT. 0)
    HCONS (NCONS) , 9A8

15. NAMES OF CROSS SECTION EDIT-TYPES [IFF (NXSTYP .GT. 0)]
    HXSTY (NXSTYP) , 9A8

16. NAMES OF RESPONSE-FUNCTIONS SELECTED [IFF (NRSF .GT. 0)]
    HRSF (NRSF) , 9A8

17. ZONE-DEPENDENT DATA SECTIONS [IFF (IZNED .GT. 0)]
    A. CROSS-SECTIONS-BY-ZONE SECTION
       [IFF (NXSTOT*NXSTYP .GT. 0)]

       XSZN (NBGP1,NZNS,NXSTYP,NXSTOT) , 6E12

       This section is to be read as follows:

```
       DIMENSION XSZN (NBGP1,NZNS,NXSTYP,NXSTOT)
       DO 40 L=1,NXSTOT
       DO 30 K=1,NXSTYP
       DO 20 J=1,NZNS
       READ (NINP,400) (XSZN(I,J,K,L), I=1,NBGP1)
20     CONTINUE
30     CONTINUE
40     CONTINUE
400    FORMAT(6E12.5)
```

B. RESPONSE-FUNCTIONS-BY-ZONE SECTION[IFF (NRSF .GT. 0)]

RSZN (NBGP1,NZNS,NRSF) , 6E12
This section is to be read as follows:

```
      DIMENSION RSZN (NBGP1,NZNS,NRSF)
      DO 30 K=1,NRSF
      DO 20 J=1,NZNS
      READ (NINP,400) (RSZN(I,J,K), I=1,NBGP1)
20    CONTINUE
30    CONTINUE
400   FORMAT(6E12.5)
```

18. POINT-DEPENDENT DATA SECTIONS [IFF (NIPE .GT. 0)]

A. CROSS-SECTIONS-BY-POINT SECTION
[IFF (NXSTOT*NXSTYP .GT. 0)]

XSPT (NBGP1,NIPE,NXSTYP,NXSTOT) , 6E12

This section is to be read as follows:

```
      DIMENSION XSPT (NBGP1,NIPE,NXSTYP,NXSTOT)
      DO 40 L=1,NXSTOT
      DO 30 K=1,NXSTYP
      DO 20 J=1,NIPE
      READ (NINP,400) (XSPT(I,J,K,L), I=1,NBGP1)
20    CONTINUE
30    CONTINUE
40    CONTINUE
400   FORMAT(6E12.5)
```

B. RESPONSE-FUNCTIONS-BY-POINT SECTION
[IFF (NRSF .GT. 0)]

RSPT (NBGP1,NIPE,NRSF) , 6E12

This section is to be read as follows:

```
      DIMENSION RSPT (NBGP1,NIPE,NRSF)
      DO 30 K=1,NRSF
      DO 20 J=1,NIPE
      READ (NINP,400) (RSPT(I,J,K), I=1,NBGP1)
20    CONTINUE
30    CONTINUE
400   FORMAT(6E12.5)
```

II. Description of the EDTOGX File.

EDTOGX is a special ASCII file optionally prepared by the Edit Module of ONEDANT/TWODANT containing geometric, fission source, and scalar flux information which can be selectively processed by the user.

This description of the EDTOGX file describes the format and construct of the file. The term "section" shall be used to refer to grouped data. The term "card image" or "card" shall be used in the same context that was described in Ch. IV of this manual.

1. NUMBER-OF-TITLE-RECORDS (CARDS) SECTION (Format I6)

This section (card) contains the single word NTITLE, where NTITLE is the number of title cards included in the file.

2. TITLE CARD SECTION (Format 10A8)

The title cards from the problem are given as individual records. This section is read as follows:

```
      DIMENSION HTITLE (10, NTITLE)
      DO 10 N=1,NTITLE
      READ (NINP,20) (HTITLE(I,N), I=1,10)
10    CONTINUE
20    FORMAT(10A8)
```

3. SPECIFICATION SECTION (Format 12I6)

The specification section is a single card-image containing those parameters needed to process the data sections. The 12 entries in this section are ordered as follows:

(1)	IDIMEN	The geometry dimension of the problem 1/2 = one-dimensional/two-dimensional
(2)	ISADJ	Adjoint problem? , 0/1 = no/yes
(3)	NGROUP	Number of energy groups
(4)	IM	Number of radial coarse mesh intervals
(5)	IT	Total number of radial fine mesh intervals.
(6)	JM	Number of axial coarse mesh intervals (=1 for IDIMEN=1)
(7)	JT	Total number of axial fine mesh intervals (=1 for IDIMEN=1)
(8)	NDUM1	Not used by ONEDANT/TWODANT
(9)	NDUM2	Not used by ONEDANT/TWODANT
(10)	IFISS	Fission source array present? 0/1 = no/yes
(11)	IGEOM	Geometry of the problem. 1=slab, 2=cylinder, etc.
(12)	IADDFX	Scalar fluxes present? 0/1 = yes/no

ASIDE: In order to simplify the dimensions of some of the following data blocks, the following four internal parameters are defined in terms of the preceding parameters found on the EDTOGX file:

(1) IMP1 = IM + 1
(2) JMP1 = JM + 1
(3) IMJM = IM*JM (Total number of coarse mesh intervals)
(4) ITJT = IT*JT (Total number of fine mesh intervals)

DATA BLOCKS

In reading the following data blocks, the user must assign data to be read to its own storage as defined by the parameters just defined. For the following data blocks we will give only a generic name for the data, the number of words in the data block, and the format-type of the block. REAL, INTEGER, and CHARACTER*8 data blocks are required. The presence or absence of a block will be indicated below by the IFF (if and only if) notation.

END ASIDE

4. RADIAL DATA INFORMATION

A. FINE MESH CELL-AVERAGE-RADIUS SECTION

RDAVG (IT) , 6E12
This section is to be read as follows:

```
      DIMENSION RDAVG (IT)
      READ (NINP,400) RDAVG
400   FORMAT(6E12.5)
```

B. NUMBER OF RADIAL-FINE-MESHES-PER-COARSE MESH SECTION
IHX (IM) , 12I6

C. COARSE MESH RADIAL BOUNDARIES SECTION
XMESH (IMP1) , 6E12

5. AXIAL DATA INFORMATION [IFF (IDIMEN .GT. 1)]

   A. FINE MESH CELL-AVERAGE-AXIAL-POSITION SECTION
      ADAVG (JT) , 6E12

   B. NUMBER OF AXIAL-FINE-MESHES-PER-COARSE MESH SECTION
      IHY (JM) , 12I6

   C. COARSE-MESH-AXIAL-BOUNDARIES SECTION
      YMESH (JMP1) , 6E12

6. ZONE NUMBERS-BY-COARSE-MESH-INTERVAL SECTION
   IDCS (IMJM) , 12I6

7. FISSION-SOURCE-RATE SECTION [IFF (IFISS .GT. 0)]
   FISRT (ITJT) , 6E12

8. SCALAR FLUX SECTION [IFF (IADDFX .EQ. 0)]

   FLUX (ITJT,NGROUP) , 6E12

   This section is to be read as follows:

```
      DIMENSION FLUX (ITJT,NGROUP)
      DO 10 J=1,NGROUP
      READ (NINP,400) (FLUX(I,J) , I=1,ITJT)
10    CONTINUE
400   FORMAT(6E12.5)
```

REFERENCES

1. R. E. Alcouffe, F. W. Brinkley, D. R. Marr, and R. D. O'Dell, "User's Guide for TWODANT: A Code Package for Two-Dimension, Diffusion-Accelerated, Neutral-Particle Transport," Los Alamos National Laboratory manual LA-10049-M, Rev. 1, (October 1984).

2. W. F. Walters, F. W. Brinkley, and D. R. Marr, "User's Guide for TWOHEX: A Code Package for Two-Dimensional, Neutral-Particle Transport in Equilateral Triangular Meshes," Los Alamos National Laboratory manual LA-10258-M, (October 1984).

3. R. D. O'Dell, "Standard Interface Files and Procedures for Reactor Physics Codes, Version IV," Los Alamos Scientific Laboratory report LA-6941-MS (September 1977).

4. B. M. Carmichael, "Standard Interface Files and Procedures for Reactor Physics Codes, Version III," Los Alamos Scientific Laboratory report LA-5486-MS (February 1974).

5. American National Standard Programming Language FORTRAN, ANSI X3.9-1978, American National Standards Institute, Inc., New York, NY 10018.

6. G. I. Bell and S. Glasstone,"Discrete Ordinates and Discrete S_N Methods," in *Nuclear Reactor Theory* (Van Nostrand Reinhold, New York, 1970), Chap. 5, pp. 232-235.

7. R. E. Alcouffe, "Diffusion Synthetic Acceleration Methods for the Diamond-Difference Discrete-Ordinates Equations," *Nucl. Sci. Eng.* 64, 344 (1977).

8. B. G. Carlson and K. D. Lathrop, "Transport Theory-Method of Discrete Ordinates," in Computing Methods in Reactor Physics, H. Greenspan, C. N. Kelber and D. Okrent, Eds. (Gordon and Breach, New York, 1968), Chap. III, p. 185.

9. Ref. 6, p. 211.

10. T. E. Albert and P. Nelson, "Computation of Azimuthally Dependent Albedo Data by Invariant Embedding," in Proc. of Sixth Intl. Conf. on Radiation Shielding, May 16-20, 1983, Tokyo, Vol. I, pp. 283-293.

☆ U.S. GOVERNMENT PRINTING OFFICE: 1989-773-034/00236

Page Range	NTIS Price Code	Page Range	NTIS Price Code	Page Range	NTIS Price Code	Page Range	NTIS Price Code
001–025	A02	151–175	A08	301–325	A14	451–475	A20
026–050	A03	176–200	A09	326–350	A15	476–500	A21
051–075	A04	201–225	A10	351–375	A16	501–525	A22
076–100	A05	226–250	A11	376–400	A17	526–550	A23
101–125	A06	251–275	A12	401–425	A18	551–575	A24
126–150	A07	276–300	A13	426–450	A19	576–600	A25
						601-up*	A99

*Contact NTIS for a price quote.

Los Alamos National Laboratory
Los Alamos,New Mexico 87545