SYN3D: A SINGLE-CHANNEL, SPATIAL FLUX SYNTHESIS CODE FOR DIFFUSION THEORY CALCULATIONS

by

C. H. Adams

Applied Physics Division

July 1976
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Abstract.</strong></td>
<td>1</td>
</tr>
<tr>
<td><strong>I. Introduction.</strong></td>
<td>1</td>
</tr>
<tr>
<td><strong>II. Synthesis Theory and Equations.</strong></td>
<td>2</td>
</tr>
<tr>
<td>A. Diffusion Theory Difference Equations</td>
<td>2</td>
</tr>
<tr>
<td>B. The Single-Channel Synthesis Approximation</td>
<td>5</td>
</tr>
<tr>
<td>C. Derivation of the Synthesis Equations</td>
<td>7</td>
</tr>
<tr>
<td>Weighted Residuals</td>
<td>7</td>
</tr>
<tr>
<td>A Variational Derivation</td>
<td>11</td>
</tr>
<tr>
<td>D. The Solution of the Synthesis Eigenvalue Equations</td>
<td>13</td>
</tr>
<tr>
<td>Fission Source Iteration with Wielandt Acceleration</td>
<td>14</td>
</tr>
<tr>
<td>Solution by Forward Elimination, Backward Substitution</td>
<td>15</td>
</tr>
<tr>
<td>Scaling and Differencing</td>
<td>15</td>
</tr>
<tr>
<td>E. Other Calculations</td>
<td>17</td>
</tr>
<tr>
<td>Multichannel Synthesis</td>
<td>17</td>
</tr>
<tr>
<td>Perturbation Theory</td>
<td>18</td>
</tr>
<tr>
<td><strong>III. User Considerations</strong></td>
<td>18</td>
</tr>
<tr>
<td>A. CCCC Standards and the ARC System</td>
<td>18</td>
</tr>
<tr>
<td>B. Basic Input Requirements</td>
<td>19</td>
</tr>
<tr>
<td>Expansion Functions</td>
<td>19</td>
</tr>
<tr>
<td>Cross Sections</td>
<td>19</td>
</tr>
<tr>
<td>Geometry</td>
<td>21</td>
</tr>
<tr>
<td>Code Dependent Input - SYNFIL</td>
<td>22</td>
</tr>
<tr>
<td>C. Running the ARC System SYNX0 - ARCP016</td>
<td>22</td>
</tr>
<tr>
<td>Input and Output Data Sets</td>
<td>22</td>
</tr>
<tr>
<td>Job Control and Symbolic Parameters</td>
<td>22</td>
</tr>
<tr>
<td>Sample Input</td>
<td>22</td>
</tr>
<tr>
<td>D. Multiple Problems and Restarts</td>
<td>26</td>
</tr>
<tr>
<td>Multiple Problems</td>
<td>26</td>
</tr>
<tr>
<td>Restarts - INTTOC, VOLINT and DIFINT</td>
<td>26</td>
</tr>
<tr>
<td>E. Edits and Output Options</td>
<td>27</td>
</tr>
<tr>
<td>Standard Output</td>
<td>27</td>
</tr>
<tr>
<td>Optional Output</td>
<td>27</td>
</tr>
<tr>
<td>Combining Coefficient Plots</td>
<td>28</td>
</tr>
<tr>
<td>F. Error Messages</td>
<td>28</td>
</tr>
<tr>
<td>G. Storage Requirements and Running Times</td>
<td>28</td>
</tr>
</tbody>
</table>

111
TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>H. Applications of Synthesis</td>
<td>29</td>
</tr>
<tr>
<td>Choosing Expansion Functions.</td>
<td>29</td>
</tr>
<tr>
<td>Blanket Functions - STP016, AJC010 and ARCSP016</td>
<td>31</td>
</tr>
<tr>
<td>IV. CODE STRUCTURE AND PROGRAMMING CONSIDERATIONS</td>
<td>31</td>
</tr>
<tr>
<td>A. The ARC System SYN3D.</td>
<td>31</td>
</tr>
<tr>
<td>STP018.</td>
<td>33</td>
</tr>
<tr>
<td>File Definitions.</td>
<td>33</td>
</tr>
<tr>
<td>B. Overlay Structure</td>
<td>33</td>
</tr>
<tr>
<td>Main Driver</td>
<td>33</td>
</tr>
<tr>
<td>TRANSL.</td>
<td>33</td>
</tr>
<tr>
<td>CARDS</td>
<td>38</td>
</tr>
<tr>
<td>HNG4C</td>
<td>38</td>
</tr>
<tr>
<td>INTEG</td>
<td>39</td>
</tr>
<tr>
<td>SOLVE</td>
<td>40</td>
</tr>
<tr>
<td>EDITS</td>
<td>40</td>
</tr>
<tr>
<td>C. BPOINTER, A Dynamic Storage Allocation Subprogram Package</td>
<td>40</td>
</tr>
<tr>
<td>IBM Allocation.</td>
<td>44</td>
</tr>
<tr>
<td>CDC Allocation.</td>
<td>44</td>
</tr>
<tr>
<td>V. THE ARGONNE CODE CENTER VERSIONS OF SYN3D</td>
<td>44</td>
</tr>
<tr>
<td>A. The SYN3D Package</td>
<td>44</td>
</tr>
<tr>
<td>SYN3D</td>
<td>46</td>
</tr>
<tr>
<td>BPOINTER.</td>
<td>46</td>
</tr>
<tr>
<td>CDFILE.</td>
<td>46</td>
</tr>
<tr>
<td>B. Code Structure.</td>
<td>46</td>
</tr>
<tr>
<td>C. File Number Assignments</td>
<td>47</td>
</tr>
<tr>
<td>D. Running the Code Center SYN3D as a Stand-Alone Program.</td>
<td>47</td>
</tr>
<tr>
<td>CDFILE.</td>
<td>47</td>
</tr>
<tr>
<td>SEEK Table Initialization</td>
<td>50</td>
</tr>
<tr>
<td>The SYN3D BCD Control Data.</td>
<td>50</td>
</tr>
<tr>
<td>Examples of Input Decks with Control Cards.</td>
<td>52</td>
</tr>
<tr>
<td>BPOINTER.</td>
<td>52</td>
</tr>
<tr>
<td>E. Sample Problems</td>
<td>52</td>
</tr>
<tr>
<td>Model Descriptions.</td>
<td>52</td>
</tr>
<tr>
<td>File 4 of the Code Center Tape.</td>
<td>52</td>
</tr>
<tr>
<td>Sample Problem 1.</td>
<td>60</td>
</tr>
<tr>
<td>Sample Problem 2.</td>
<td>60</td>
</tr>
<tr>
<td>Sample Problem 3.</td>
<td>62</td>
</tr>
<tr>
<td>Sample Problem 4.</td>
<td>62</td>
</tr>
</tbody>
</table>
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>F. Suggested Modifications to the Code</strong></td>
<td>65</td>
</tr>
<tr>
<td>TIMER</td>
<td>65</td>
</tr>
<tr>
<td>SEEK Initialization</td>
<td>65</td>
</tr>
<tr>
<td>Eliminating BCD input Data</td>
<td>65</td>
</tr>
<tr>
<td>Matrix Inversion and Multiplication Subroutines</td>
<td>66</td>
</tr>
<tr>
<td>Restrictions on Problem Size in SOLVE</td>
<td>66</td>
</tr>
<tr>
<td><strong>APPENDICES</strong></td>
<td>67</td>
</tr>
<tr>
<td>A. Code Abstract</td>
<td>67</td>
</tr>
<tr>
<td>B. STP018 - The ARC System Standard Path for SYN3D</td>
<td>69</td>
</tr>
<tr>
<td>C. ARCSP018 - The ARC System Catalogued Procedure for SYN3D</td>
<td>72</td>
</tr>
<tr>
<td>D. Input Data Set SYNFIL</td>
<td>80</td>
</tr>
<tr>
<td>E. Code-Dependent Binary Files</td>
<td>101</td>
</tr>
<tr>
<td>COMPSX.</td>
<td>101</td>
</tr>
<tr>
<td>DCOEF.</td>
<td>105</td>
</tr>
<tr>
<td>DIFINT.</td>
<td>109</td>
</tr>
<tr>
<td>INTTOC.</td>
<td>111</td>
</tr>
<tr>
<td>REQFLX.</td>
<td>119</td>
</tr>
<tr>
<td>REQXST.</td>
<td>121</td>
</tr>
<tr>
<td>SYNCON.</td>
<td>123</td>
</tr>
<tr>
<td>VOLINT.</td>
<td>123</td>
</tr>
<tr>
<td>F. SYN3D Error Messages</td>
<td>127</td>
</tr>
<tr>
<td>G. FORTRAN Variable Glossary</td>
<td>132</td>
</tr>
<tr>
<td>H. Sample Problem Output</td>
<td>141</td>
</tr>
<tr>
<td>Problem 1 - Complete Output, with Cross Section Edits</td>
<td>141</td>
</tr>
<tr>
<td>Problem 2 - Complete Output</td>
<td>157</td>
</tr>
<tr>
<td>Problem 3 - Partial Output</td>
<td>163</td>
</tr>
<tr>
<td>Problem 4 - Partial Output</td>
<td>173</td>
</tr>
<tr>
<td>I. CCCC Version III Standard</td>
<td>182</td>
</tr>
<tr>
<td>GEODST.</td>
<td>183</td>
</tr>
<tr>
<td>ISOTXS.</td>
<td>185</td>
</tr>
<tr>
<td>NDXSRF.</td>
<td>194</td>
</tr>
<tr>
<td>ZNATDN.</td>
<td>196</td>
</tr>
<tr>
<td>RTFLUX.</td>
<td>198</td>
</tr>
<tr>
<td>ATRFLUX.</td>
<td>200</td>
</tr>
<tr>
<td>FWINT.</td>
<td>202</td>
</tr>
<tr>
<td>RZFLUX.</td>
<td>203</td>
</tr>
<tr>
<td>J. ARCSP016 - The ARC System Catalogued Procedure for Generating a fixed source for an inhomogeneous calculation of a low-reactivity-plane synthesis expansion function</td>
<td>205</td>
</tr>
</tbody>
</table>
## TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>K. STP016 - The ARC System Standard Path for Generating a fixed source for an inhomogeneous calculation of a low-reactivity-plane synthesis expansion function</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
</tr>
<tr>
<td>REFERENCES</td>
</tr>
</tbody>
</table>
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>No.</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>III-1</td>
<td>ARC System SYN3D Input Deck. The Three-Dimensional Model is Built up from Two-Dimensional A.NIP Data Sets.</td>
<td>24</td>
</tr>
<tr>
<td>III-2</td>
<td>Input for a Blanket Expansion Function Calculation Using ARCSPO16 and ARCSPO3 (DIF2D).</td>
<td>32</td>
</tr>
<tr>
<td>IV-1</td>
<td>SYN3D Program Structure Showing Subroutine Calling Sequences. TRANSL is used only in the ARC SYN3D, HMG4C is used only in the code center version. Routines called by several subroutines in a branch are listed under the branch name. For example, SEEK is used throughout the code, QUIT is called by several subroutines in HMG4C, etc.</td>
<td>34</td>
</tr>
<tr>
<td>IV-2</td>
<td>Example of BPOINTER Use</td>
<td>42</td>
</tr>
<tr>
<td>V-1</td>
<td>Input Deck with control cards for the IBM 370/195. Execute CDFILE to generate interface files, Link edit SYN3D and execute sample problem 4. The source code for CDFILE is in the file &amp;CDFILE and the CDFILE input is in the file &amp;DATA. Interface files not needed in this job are copied to the file &amp;DUMP.</td>
<td>53</td>
</tr>
<tr>
<td>V-2</td>
<td>Input deck with control cards for the CDC 7600. Execute CDFILE to generate interface files, load SYN3D and execute sample problem 4. The relocatable object code for CDFILE is in the file &quot;CDFILE&quot; and the CDFILE input is in the file &quot;DATA&quot;. The file &quot;PROG&quot; contains the relocatable object code for SYN3D. Interface files not needed in this job are copied to the file &quot;TAPE9&quot;.</td>
<td>56</td>
</tr>
<tr>
<td>V-3</td>
<td>The geometry of the 3D test model</td>
<td>57</td>
</tr>
<tr>
<td>V-4</td>
<td>SYN3D BCD input and input interface files for the first sample problem. Each input file is identified by a sequence number from Table V-7 and is assigned a logical unit number consistent with Table V-3. The BCD data consists of the SEEK initialization card, a &quot;CARDS-PER-CARD-TYPE&quot; card and a set of type-numbered cards.</td>
<td>61</td>
</tr>
<tr>
<td>V-5</td>
<td>SYN3D BCD input and input interface files for the second sample problem. Each input file is identified by a sequence number from Table V-7 and is assigned a logical unit number consistent with Table V-3. The BCD data consists of the SEEK initialization card, a &quot;CARDS-PER-CARD-TYPE&quot; card and a set of type-numbered cards.</td>
<td>61</td>
</tr>
<tr>
<td>No.</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>-----</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>V-6</td>
<td>SYN3D BCD input and input interface files for the third sample problem. Each input file is identified by a sequence number from Table V-7 and is assigned a logical unit number consistent with Table V-3. The BCD data consists of the SEEK initialization card, a &quot;CARDS-PER-CARD-TYPE&quot; card and a set of type-numbered cards.</td>
<td>63</td>
</tr>
<tr>
<td>V-7</td>
<td>SYN3D BCD input and input interface files for the fourth sample problem. Each input file is identified by a sequence number from Table V-7 and is assigned a logical unit number consistent with Table V-3. The BCD data consists of the SEEK initialization card, a &quot;CARDS-PER-CARD-TYPE&quot; card and a set of type-numbered cards.</td>
<td>64</td>
</tr>
<tr>
<td>No.</td>
<td>Title</td>
<td>Page</td>
</tr>
<tr>
<td>------</td>
<td>----------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>III-1</td>
<td>ARC System and CCCC Input Files for SYN3D. These are not all required. They are generated by SYN3D or its ARC System Standard Pack and otherwise may never be needed at all.</td>
<td>20</td>
</tr>
<tr>
<td>III-2</td>
<td>ARC SYN3D Input and Output Data Set Names and Job Control Substitution Symbolic Parameters.</td>
<td>23</td>
</tr>
<tr>
<td>III-3</td>
<td>ARC System SYN3D Running Times and Storage Requirements for Two Fast Critical Assembly Models on the IBM 370/195. The GCFR Model was Run at Two Different Container Sizes.</td>
<td>30</td>
</tr>
<tr>
<td>IV-1</td>
<td>SYN3D Subroutines and Common Blocks</td>
<td>35</td>
</tr>
<tr>
<td>IV-2</td>
<td>BPOINTER Subprogram Descriptions.</td>
<td>42</td>
</tr>
<tr>
<td>V-1</td>
<td>Description of SYN3D Tapes and the BCD Files they Contain.</td>
<td>45</td>
</tr>
<tr>
<td>V-2</td>
<td>SYN3D Segments.</td>
<td>45</td>
</tr>
<tr>
<td>V-3</td>
<td>Files Required by the Code Center Version of SYN3D (GEODST through ZNATDN are CCCC files. The rest may be treated as scratch files.)</td>
<td>48</td>
</tr>
<tr>
<td>V-4</td>
<td>CDFILE Input Data</td>
<td>51</td>
</tr>
<tr>
<td>V-5</td>
<td>Examples of CDFILE Modes of Operation. These Examples Assume File 5 is Card Input, File 6 is Printer Output and File 7 is Punch Output.</td>
<td>51</td>
</tr>
<tr>
<td>V-6</td>
<td>Compositions Used in Test Problems.</td>
<td>58</td>
</tr>
<tr>
<td>V-7</td>
<td>Description of the CDFILE BCD Data Blocks in File 4 of the Code Center Tape.</td>
<td>59</td>
</tr>
</tbody>
</table>
SYN3D: A SINGLE-CHANNEL, SPATIAL FLUX SYNTHESIS CODE FOR
DIFFUSION THEORY CALCULATIONS

by

C. H. Adams

ABSTRACT

This report is a user's manual for SYN3D, a computer code which uses single-channel, spatial flux synthesis to calculate approximate solutions to two- and three-dimensional, finite-difference, multigroup neutron diffusion theory equations. SYN3D is designed to run in conjunction with any one of several one- and two-dimensional, finite-difference codes (required to generate the synthesis expansion functions) currently being used in the fast reactor community. The report describes the theory and equations, the use of the code, and the implementation on the IBM 370/195 and CDC 7600 of the version of SYN3D available through the Argonne Code Center.

1. INTRODUCTION

This report is a user's manual for SYN3D, a computer code which uses single-channel, spatial flux synthesis to calculate approximate solutions to two- and three-dimensional, finite-difference, multigroup neutron diffusion theory equations.

SYN3D is designed to run in conjunction with any one of several one- and two-dimensional finite-difference codes (required to generate the synthesis expansion functions) currently being used in the fast reactor community. Most of the data for a calculation must be supplied in the formats of the Standard Interface Files defined by the Committee on Computer Code Coordination (CCCC). Appendix A of this report is the computer code abstract for SYN3D.

SYN3D is operational at Argonne National Laboratory within the ARC System of reactor analysis codes and has been sent, in stand-alone form, to the Argonne Code Center for distribution to other laboratories. Sections II, III, and IV of this report cover material of interest to users of both the ARC System and Code Center versions of the code; Section V deals with the Code Center version only. Users who have just received SYN3D from the Code Center and who are faced with the task of bringing the code up on their machine should read Section V first; it describes the contents of the Code Center tapes and outlines the steps necessary to implement SYN3D in stand-alone form on the IBM 370/195 and CDC 7600.
II. SYNTHESIS THEORY AND EQUATIONS

In this section we develop the synthesis approximation used in SYN3D from the mesh-box-centered, diffusion theory, finite-difference equations and describe the solution of the resulting synthesis equations.

A. Diffusion Theory Difference Equations

There are two finite-difference forms of the neutron diffusion equations currently used in reactor analysis. In the derivation of the first, the fluxes in the differential equations are expanded in Taylor series about the mesh points (i.e. the intersections of the mesh lines in two dimensions and mesh planes in three). The PDQ code contains this formulation.\textsuperscript{1,2} The other difference equations are derived by expanding the fluxes about a point at the center of a mesh block. The codes 2DB, 3DB, DIF2D, CITATION and VENTURE use this approach.\textsuperscript{3-6} SYN3D is designed to calculate an approximate solution for the second type of difference equation, which is the form most frequently used in fast reactor analysis.

In \(x-y-z\) geometry the three-dimensional, finite-difference equations associated with the mesh block defined by \(x\) mesh interval \(i\), \(y\) mesh interval \(j\), and \(z\) mesh interval \(k\) in energy group \(g\) can be written:

\[
\begin{align*}
- d_{i+1,j+1,k}^{xg} [\phi_{i+1,j,k}^{g} - \phi_{i,j,k}^{g}] \Delta y \Delta z_k + d_{i,j+1,k}^{yg} [\phi_{i,j+1,k}^{g} - \phi_{i,j,k}^{g}] \Delta x \Delta z_k + d_{i,j,k+1}^{zg} [\phi_{i,j,k+1}^{g} - \phi_{i,j,k}^{g}] \Delta x \Delta y_j \\
+ \sum_{g} v_{ijk} r_{ijk}^{s,g} \phi_{ijk}^{g} - v_{ijk} s_{ijk}^{g} = 0, \\
i = 1...I, \quad j = 1...J, \quad k = 1...K, \quad g = 1...G.
\end{align*}
\]

\(I\) is the number of \(x\) mesh intervals, \(J\) is the number of \(y\) mesh intervals, and \(K\) is the number of axial mesh intervals. \(G\) is the number of energy groups. No equations are written for those mesh blocks which contain a blackness theory material.

\(\phi_{ijk}^{g}\) is the discrete group flux associated with mesh block \((i,j,k)\). \(\phi_{ijk}^{g}\) is identically zero if mesh block \((i,j,k)\) is in a blackness theory region. If \(i = 0\) or \(i + 1\), \(j = 0\) or \(j + 1\), or \(k = 0\) or \(K + 1\) the \(\phi_{ijk}^{g}\)'s are exterior fluxes and their definition depends on the boundary conditions at the corresponding boundary of the model. For homogeneous boundary conditions of the form...
\[ C_1^g D^g n \cdot v^g + C_2^g C^g = 0 \]  \hspace{1cm} (2)

where \( \mathbf{n} \) is the outward normal unit vector, the exterior fluxes for the boundary are identically zero; the boundary condition is specified by the definition of the d's (see below). Equation (2) includes the cases of zero flux \([C^g = 0]\), zero current \([C^g = 0]\) and logarithmic (or extrapolated) conditions. The exterior fluxes at a boundary with periodic conditions are identically equal to the first interior fluxes on one of the other boundary surfaces of the model.

The three types of d's \([d^{xg}_{ijk}, d^{yg}_{ijk}, d^{zg}_{ijk}]\) are defined similarly, and so it is necessary to discuss only one of them \([d^{xg}_{ijk}]\). If the mesh blocks \((i,j,k)\) and \((i-1,j,k)\) are within the boundaries of the model, and neither contains a blackness theory material,

\[ d^{xg}_{ijk} = \frac{2p^g_{i-1,jk} p^g_{ijk}}{\Delta x_i \Delta y_j \Delta z_k} \text{ } (3) \]

\[ i = 1 \ldots I, \text{ } j = 1 \ldots J, \text{ } k = 1 \ldots K, \text{ } g = 1 \ldots G. \]

\( D^g_{ijk} \) is the group g diffusion coefficient for mesh block \((i,j,k)\) and \(\Delta x_i, \Delta y_j, \text{ and } \Delta z_k\) define its size. If one of the mesh blocks is outside the boundary of the model (the cases \(i = 1\) and \(i = I + 1\)), but that boundary surface has a periodic boundary condition, then Eq. (3) still holds if it is understood that the exterior mesh block is identical to a corresponding interior mesh block on some other boundary.

SYN3D permits anisotropic diffusion coefficients. The value of \( D^g_{ijk} \) appearing in the expression for \( d^{xg}_{ijk} \) may be different from the value of \( D^g_{ijk} \) appearing in \( d^{yg}_{ijk} \).

For the homogeneous boundary condition given by Eq. (2) at \(i = 1\) as well as when there is an internal interface between a blackness theory material in mesh block \((i-1,j,k)\) and an ordinary material in \((i,j,k)\),

\[ d^{xg}_{ijk} = \frac{D^g_{i-1,jk} C^g_{i-1,jk}}{D^g_{ijk} C^g_{ijk} + \frac{1}{2} \Delta x_i C^g_{ijk}} \text{ } (4) \]

In the case of a blackness theory boundary, \( \mathbf{n} \) in Eq. (2) is the unit normal into the blackness region at the interface. When Eq. (2) is the boundary condition at \(i = I\), and for an internal interface between a blackness theory material in mesh block \((i,j,k)\) and an ordinary material in \((i-1,j,k)\),

\[ d^{xg}_{ijk} = \frac{D^g_{i-1,jk} C^g_{i-1,jk}}{D^g_{i-1,jk} C^g_{i-1,jk} + \frac{1}{2} \Delta x_i C^g_{i-1,jk}} \text{ } (5) \]
If both \((i - 1, j, k)\) and \((i, j, k)\) contain blackness theory material \(d^g_{i,j,k}\) is not defined, because the only two equations in which it could appear are never written.

The remaining notation in Eq. (1) is simpler to define. \(v_{i,j,k}\) is the volume of mesh block \((i, j, k)\). \(r_{i,j,k}\) is basically the combination of macroscopic cross sections representing absorption, scattering, and fission. \(s_{i,j,k}\) is an inhomogeneous source. With the proper definitions of \(r_{i,j,k}\) and \(s_{i,j,k}\), Eq. (1) becomes the statement of an eigenvalue problem, an outer iteration of an eigenvalue problem, a distributed source problem, or even one time step of an implicit space-time calculation. To cast Eq. (1) in the form of the eigenvalue problem, for example,

\[
s^g_{i,j,k} = 0,
\]

\[
r^{g}_{i,j,k} = \Sigma^g_{R} + \Sigma^g_{S} - \lambda \chi^g_{f} v^g_{f} ,
\]

where \(\Sigma^g_{R}\) is the total macroscopic cross section for removal from group \(g\), \(\Sigma^g_{S}\) is the cross section for scattering into group \(g\) from \(g'\), \(\chi^g\) is the fission spectrum and \(v^g_{f}\) is the cross section for neutron production by fission. \(\lambda\) is the eigenvalue, and \(\delta_{gg'}\) is the Kronecker delta.

In \(r-z\) and triangular geometries the finite-difference equations are similar to Eq. (1) if one alters some definitions. In \(r-z\) geometry the "\(x\)" dimension is the radial dimension, and the leakage finite-difference coefficient away from external boundaries and blackness regions becomes

\[
\phi_{i,j,k}^{xg} = \frac{2\phi_{i-1,j,k}^{g} - \phi_{i,j,k}^{g}}{\Delta x_i \phi_{i-1,j,k}^{g} + \Delta x_i \phi_{i,j,k}^{g}} \quad \text{(2N}\ \text{N}_{i})
\]

where \(x_i\) is the radial position of the interface between mesh interval \(i-1\) and \(i\). The leakage coefficients at boundaries (Eqs. (4) and (5)) are also multiplied by \(2N_{i}\). Now \(v_{i}\) is the volume of an annulus.

For geometries which have triangular mesh in the plane, each mesh cell couples to 3 other cells in the plane rather than to 4. The coupling between internal mesh triangles \((i-1,j,k)\) and \((i,j,k)\) is

\[
\phi_{i,j,k}^{xg} = \frac{2\sqrt{3}\ \phi_{i-1,j,k}^{g} \phi_{i,j,k}^{g}}{\phi_{i-1,j,k}^{g} + \phi_{i,j,k}^{g}}
\]

and no \(\Delta y_i\) appears multiplying the term (as in Eq. (1)). When one side of a triangle is an exterior (or blackness region) boundary the coefficient is given by
\[ d_{ijk}^{xg} = \frac{D_{ijk}^g C_2^g}{2\sqrt{3} D_{ijk}^g C_1^g + \Delta x C_2^g} \quad (2\sqrt{3} \Delta x) \] (9)

\[ d_{ijk}^{xg} = \frac{C_2^g}{4 C_1^g D_{ijk}} \quad (2\sqrt{3} \Delta x) \] (10)

\[ \Delta x \text{ is not subscripted since in regular triangular mesh there is only one characteristic length, the side of a triangle. When the triangular mesh is bisected by a boundary, and the flux, therefore, is defined for a point on the boundary,} \]

\[ d_{ijk}^{xg} = \frac{C_2^g}{4 C_1^g D_{ijk}} \quad (2\sqrt{3} \Delta x) \] (10)

B. The Single-Channel Synthesis Approximation

The equations and capabilities of SYN3D are based on Kaplan's blending method.\textsuperscript{7} Kaplan suggested that the three-dimensional neutron flux can often be approximated by the trial function

\[ \phi_{ijk}^g = \sum_{n=1}^{N} a_{nk}^g H_{nij}^g \] (11)

The planar expansion functions, \( H_{nij}^g \), are known functions, precalculated by the user. They should be the best practical estimates of the planar flux distributions encountered at various axial positions. The combining coefficients, \( a_{nk}^g \), are the unknowns of the synthesis calculation. Equation (11) states the basic, single-channel synthesis approximation; a more elaborate approximation, multichannel synthesis, is discussed briefly in a later section of this report.

The literature already contains a number of derivations of synthesis difference equations.\textsuperscript{8-11} The derivations shown later in this report borrow features of a number of them and are built around the premise that synthesis should be viewed as an approximate method of solving the finite-difference equations; nowhere in this report do we use the differential form of the diffusion equation. This is a realistic approach, since we know that expansion functions must come from finite-difference calculations, and it avoids the problem of trial function discontinuities which arises when one starts from the differential form.\textsuperscript{9,12,16-17}

The discrete combining coefficients in Eq. (11) represent a potentially large number \((N \times K \times G)\) of unknowns, and SYN3D offers two ways to reduce that number. First, if one of the expansion functions, \( H_{nij}^g \), is characteristic of a limited axial zone of the model, and there is no reason to think its
presence may help the solution away from that zone, one can simply set the corresponding combining coefficients, \( \alpha_{nk} \), equal to zero for ranges of values of \( k \). This is the discrete analog to the treatment of the continuous equations suggested in Ref. 18 and has the effect of excluding particular expansion functions from the trial function in particular axial zones. Carried to the extreme that only one combining coefficient in each group remains unknown for each mesh interval, Eq. (11) reduces to the one-function-per-zone synthesis form proposed by Meyez.19 In a variation of this procedure, setting to zero all those combining coefficients associated with a particular \( n \) and \( g \) offers a mechanism for specifying different numbers of expansion functions for each energy group.

The second way to reduce the number of unknowns is by group collapsing.20 Group-collapsed synthesis is not to be confused with group collapsed cross section; it simply means that the user specifies some prescription for expressing the \( G \) combining coefficients for the \( G \) group fluxes of one expansion function in terms of a smaller number, \( G^* \), of unknowns. In this work we will allow linear transformations of the form

\[
\alpha^g_{nk} = \sum_{b=1}^{G'} u^{gb} \alpha^b_{nk} .
\] (12)

Note that we have limited ourselves to one collapsing scheme for all expansion functions and regions of the model. To see how group collapsing might be applied, consider the example of a four-group model collapsed to two groups. If the matrix \( u \) was given by

\[
u = \begin{pmatrix} 1 & 0 \\ 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{pmatrix},
\] (13)

then the synthesis trial functions [Eq. (11)] for each group flux become

\[
\phi^1_{jk} = \sum_{n=1}^{N} \alpha_{nk}^{1} \Phi_{nj}^1, \quad \phi^2_{jk} = \sum_{n=1}^{N} \alpha_{nk}^{2} \Phi_{nj}^2,
\]

\[
\phi^3_{jk} = \sum_{n=1}^{N} \alpha_{nk}^{3} \Phi_{nj}^3, \quad \phi^4_{jk} = \sum_{n=1}^{N} \alpha_{nk}^{4} \Phi_{nj}^4.
\] (14)

A single set of combining coefficients, \( \alpha^1 \), is used for both the group 1 and group 2 trial functions. Another set, \( \alpha^2 \), is used with groups 3 and 4. In this example the number of unknowns has been reduced by a factor of 2.

One constraint should be applied to the definition of the matrix elements, \( u^{gb} \), to assure that the individual expansion function spectra
are recoverable from the group-collapsed synthesis trial function [Eq. (11)], and that is

\[ \sum_{b=1}^{G'} u^b = \text{the same value for all } g. \tag{15} \]

This constraint is desirable since the user has supplied expansion functions which should have spectra characteristic of the various regions of the model.

With the group-collapsing option added, the synthesis trial function becomes

\[ \phi_{ijk}^g = \sum_{b=1}^{G'} u^b g \sum_{n=1}^{N} a^b_{nk} h^g_{ij}. \tag{16} \]

Note that Eq. (16) reduces to the statement of group-dependent synthesis (Eq. (11)) when \( u^b \) is the unit matrix.

So far we have discussed only the trial function for the direct flux. For a variational derivation of the synthesis equations and for perturbation theory, an adjoint flux is required.

\[ \phi_{ijk}^{a^b} = \sum_{b=1}^{G'} a^b_{gb} \sum_{n=1}^{N} a^b_{nk} h^g_{ij}. \tag{17} \]

\( u^a \) is the adjoint group-collapsing scheme, which does not have to be the same as the direct group-collapsing matrix, \( u \). \( a^a \) is the adjoint combining coefficient. \( h^a_{n} \) is a user-supplied expansion function for the adjoint trial function and is usually called a weighting function. In order to keep the synthesis equations in a form convenient to solve, there must be the same number of nonzero adjoint combining coefficients, \( a^a_{nb} \), as there are direct, \( a^b_{nk} \), at each axial mesh interval, \( k \).

C. Derivation of the Synthesis Equations

There are two methods commonly used to derive synthesis equations. We first outline the most straightforward of them, the weighted residuals (or weight-and-integrate) approach, and next show a derivation from a variational principle.

Weighted Residuals

For the weighted residuals method, start by substituting the synthesis trial function (Eq. (16)) into the finite-difference equation (Eq. (1)). Then multiply each equation for a particular axial mesh interval (i.e., each combination of \( i, j, g \) for one \( k \)) by the factor
and sum the weighted equations over the indices i, j and g. Repeat this procedure for a total number of multipliers (Eq. (18)) equal to the number of combining coefficients for a single value of k, making sure that each time the combination of collapsed group (b') and expansion function (n') for the multiplier is unique. We shall account later for the fact that some of the combining coefficients may be zero (i.e. that certain expansion functions have been dropped where it is felt that they are not needed). For now we will treat the calculation as if all expansion functions are used everywhere.

When the dust settles, the synthesis equations are:

\[
\sum_{b=1}^{G} \sum_{n=1}^{N} a_{nk} \left( \sum_{g=1}^{G} \frac{u_{gb}}{g} \left( \sum_{i=1}^{J} J_{i}^{k} \frac{d_{ijk}^{g}}{g} \right) \right) = 0
\]
for each combination of \( k, b' \) and \( n' \) for which a weighting was performed.

To reduce Eq. (19) to a form which is a convenient basis for a practical computer code we will restrict the axial boundary conditions to the form given in Eq. (2) (zero flux, zero current or extrapolated boundary conditions) and introduce new notation. Periodic conditions will not be allowed at the top and bottom of the model, which means that

\[
\begin{align*}
a_n^{b0} &= a_n^{b} = 0
\end{align*}
\]

for all \( n \) and \( b \) in Eq. (19).

We introduce a new subscript, \( m \), which identifies a unique combination of collapsed group \( (b) \) and expansion function \( (n) \).

\[
a_{mk} = a_{nk} \quad \quad (21)
\]

The maximum value of \( m \), which is the maximum number of combining coefficients associated with a single mesh interval, is \( M \).

\[
M = (i' \times N) \quad \quad (22)
\]

There is a comparable index, \( m' \), associated with the weighting functions. By redefining some of the multiple summations in Eq. (19) we can write the synthesis equations.

\[
0 = - \sum_{m=1}^{M} \left[ \frac{1}{\Delta z_k} D_m^{-mk} a_{mk-1} + \frac{1}{\Delta z_{k+1}} D_{m}^{-mk+1} a_{mk+1} \right] + \sum_{m=1}^{M} \left[ \frac{1}{\Delta z_k} D_{m}^{-mk} R_{m}^{-mk} S_{m}^{-k} \right] a_{mk} - \Delta z_k S_{m}^{-k} \quad \quad (23)
\]

subject to the condition given by Eq. (20). The definitions of the matrices \( D_{m}^{-mk} \) and \( R_{m}^{-mk} \) and the vector \( S_{m}^{-k} \) are:

\[
R_{m}^{-mk} = \sum_{g=1}^{G} u_{g}^{*} g' b' \sum_{i=1}^{I} \sum_{j=1}^{J} h_{i}^{*} h_{1}^{*} \Delta x_{1} \Delta y_{j} r_{ij}^{*} g'_{i1j} - h_{0}^{*} g'_{i1j}
\]

\[
+ \sum_{g=1}^{G} u_{g}^{*} g' b' \sum_{j=1}^{J} \left[ \sum_{i=1}^{I} h_{g}^{*} h_{n}^{*} d_{1} a_{ij} \Delta y_{j} \left( h_{g}^{*} h_{n}^{*} - h_{0}^{*} a_{ij} \right) \right]
\]
In order to define $D_{m'^{mK+1}}$ by a single expression it is necessary to define $\Delta x_k$ in $D_{m'^{mK+1}}$. The definition is arbitrary, since it is cancelled by the inverse mesh spacing when $D_{m'^{mK+1}}$ is used in Eq. (23).

We pointed out earlier that expansion functions can be eliminated from the trial function over ranges of axial mesh intervals by setting their combining coefficients identically equal to zero. We can modify Eq. (23) to imply this if we define the symbol

$$\sum_{m \text{ at } k}$$

(27)

to mean the sum over only those values of $m$ at axial mesh interval $k$ which correspond to nonzero combining coefficients, $a_{mk}$. When the number of unknowns is reduced in this way it becomes necessary to reduce the number of equations. In order that the reduced set of synthesis equations remains in a block-tridiagonal form which is convenient for numerical inversions, we will require that when a combining coefficient is eliminated the corresponding equation that must be dropped be one associated with the same mesh point. Therefore, the "$m'= 1 \ldots m$" in Eq. (23) is replaced by "$m'$ at $k$", implying that Eq. (23) is written only for certain values of $m'$ at each mesh interval.
The final form of the synthesis equations can now be written,

\[ 0 = \sum_{m \text{ at } k-1}^{m \text{ at } k} \frac{1}{\Delta z_k} D_{m'mk} a_{mk-1} - \sum_{m \text{ at } k+1}^{m \text{ at } k} \frac{1}{\Delta z_{k+1}} D_{m'mk+1} a_{mk+1} \]

\[ + \sum_{m \text{ at } k}^{m \text{ at } k+1} \left[ \frac{1}{\Delta z_k} D_{m'mk} + \frac{1}{\Delta z_{k+1}} D_{m'mk+1} + \Delta z_k R_{m'mk} \right] a_{mk} \]

\[ - \Delta z_k S_{m'mk}, \quad m' \text{ at } k, \quad k = 1 \ldots K \]  

(28)

A Variational Derivation

Consider the following functional:

\[ G \sum_{g=1}^{J} \sum_{j=1}^{N} \sum_{k=1}^{K} \Delta y_j \Delta z_k \left( \phi_{1jk} d_{1jk} \left( \phi_{1jk} - \phi_{0jk} \right) - \phi_{0jk} d_{1+1jk} \left( \phi_{1+1jk} - \phi_{1jk} \right) \right) \]

\[ + \sum_{i=2}^{I} \left( \phi_{ijk} - \phi_{i-1jk} \right) d_{ijk} \left( \phi_{ijk} - \phi_{i-1jk} \right) \]

\[ \cdot G \sum_{g=1}^{J} \sum_{i=1}^{L} \sum_{k=1}^{K} \Delta x_i \Delta z_k \left( \phi_{11k} d_{11k} \left( \phi_{11k} - \phi_{10k} \right) - \phi_{10k} d_{1+1k} \left( \phi_{1+1k} - \phi_{1jk} \right) \right) \]

\[ + \sum_{j=2}^{J} \left( \phi_{ijjk} - \phi_{ij-1k} \right) d_{ijjk} \left( \phi_{ijjk} - \phi_{ij-1k} \right) \]

\[ \cdot G \sum_{g=1}^{J} \sum_{i=1}^{L} \sum_{j=1}^{N} \Delta y_j \Delta y_i \left( \phi_{ij1} d_{ij1} \left( \phi_{ij1} - \phi_{ij0} \right) - \phi_{ij0} d_{ijk+1} \left( \phi_{ijk+1} - \phi_{ijk} \right) \right) \]

\[ + \sum_{k=2}^{K} \left( \phi_{ijk} - \phi_{ijk-1} \right) d_{ijk} \left( \phi_{ijk} - \phi_{ijk-1} \right) \]

\[ \cdot G \sum_{g=1}^{J} \sum_{i=1}^{L} \sum_{j=1}^{N} \sum_{k=1}^{K} \phi_{ijk} v_{ijk} r_{ijk} \phi_{ijk} \]

\[ \cdot G \sum_{i=1}^{L} \sum_{j=1}^{N} \sum_{k=1}^{K} v_{ijk} \left( \phi_{ijk} S_{ijk} + s_{ijk} \phi_{ijk} \right) \]  

(29)

\( \phi_{ijk} \) is the discrete adjoint group flux and \( g_{ijk} \) an adjoint source for mesh block \((i,j,k)\). The other quantities were defined when they appeared earlier in Eq. (1). It is straightforward to show that requiring the functional to
be stationary with respect to arbitrary variations of the discrete adjoint fluxes leads to the set of difference equations given in Eq. (1) for any of the allowed types of boundary conditions.

For boundary conditions in the form of Eq. (2) recall that the exterior fluxes \( \phi_{0jk}, \phi_{+1jk}, \phi_{10k}, \) etc. are identically zero. When those terms are dropped from Eq. (29) the symmetry of the remaining expression should make it clear that requiring variations with respect to the direct flux be zero leads to the usual adjoint difference equations. To show that the same is true when there are periodic boundary conditions one must replace the exterior discrete fluxes in Eq. (29) by the appropriate internal fluxes before calculating the variations. For example, if an \( x-y-z \) geometry model has quarter-core symmetry the boundary conditions are periodic at the surfaces \( i = 1 \) and \( j = 1 \). The external fluxes are redefined by

\[
\phi^g_{0jk} = \phi^g_{j1k}, \quad \phi^g_{10k} = \phi^g_{11k},
\]

\[
\phi^*_{0jk} = \phi^*_{j1k}, \quad \phi^*_{10k} = \phi^*_{11k}.
\]

(30)

There are also identities relating the coefficients in the leakage terms at the periodic boundary.

\[
\Delta x_i = \Delta y_i \quad i = 1 \ldots 1, \quad \Delta x_0 = \Delta y_1 = \Delta x_2 = \Delta y_0
\]

\[
\Delta x_i \phi^g_{0jk} + \Delta x_0 \phi^g_{0jk} = \Delta x_i \phi^g_{1jk} + \Delta y_1 \phi^g_{1jk}.
\]

(31)

With the identities shown in Eqs. (30) and (31), one can show that adjoint equations with periodic boundary conditions can be derived from the functional given by Eq. (29).

To derive the synthesis equations, first substitute the trial functions for the direct flux [Eq. (16)] and adjoint flux [Eq. (17)] into the functional [Eq. (29)]. One can reduce the result to

\[
K+1 \sum_{k=1}^{K+1} M \sum_{m=1}^{M} \sum_{m=1}^{M} \frac{1}{\Delta z_k} \left( a^*_{m-k} - a^*_{m-k-1} \right) D_{m-nk} \left( a_{mk} - a_{mk-1} \right)
\]

\[
+ \sum_{k=1}^{K} M \sum_{m=1}^{M} \sum_{m=1}^{M} \Delta z_k a_{m-k} R_{m-nk} a_{mk}
\]

\[
- \sum_{k=1}^{K} M \sum_{m=1}^{M} \Delta z_k a_{m-k} S_{m-k} - \sum_{k=1}^{K} M \sum_{m=1}^{M} \Delta z_k a_{m-k} S_{m-k}
\]

(32)
by using the definitions given in Eqs. (24) - (26) and

\[ S_{mk}^* = \sum_{g'=1}^{G} \sum_{i'j=1}^{I} \sum_{j=1}^{J} H_{nij}^{g*} G_{ij} \Delta x_i \Delta y_j S_{ijk}^{g*}. \]  

(33)

Setting the variations of this functional with respect to nonzero, adjoint combining coefficients equal to zero leads directly to the direct synthesis equations derived in the last section [Eq. (28)]. Following the same procedure with the direct combining coefficients leads to a set of adjoint synthesis equations.

\[ 0 = - \sum_{m} \sum_{k=1}^{K} \left[ \frac{1}{\Delta z_k} D_{m-mk}^{m-k-1} \Delta z_k \Delta z_{k+1} D_{m-mk+1}^{m-k+1} \right] a_{m-k}^{*} - \Delta z_k S_{mk}^*. \]  

(34)

It should be emphasized that the adjoint variables in a synthesis calculation are combining coefficients, \( a_{mk}^* \), and not fluxes. Whether or not the adjoint trial function [Eq. (17)] is a good approximation to the adjoint flux depends on the user's choice of weighting function. It is very common in practical calculations to use the expansion functions for weighting functions also, and this might be a poor choice in situations where the adjoint is required. In particular, when sodium void reactivity distributions are to be calculated by perturbation theory it may be necessary to generate weighting functions from two-dimensional adjoint calculations.

When the groups are not collapsed in a synthesis calculation the adjoint combining coefficients will, as best they can, reproduce the adjoint spectrum. If the groups are partially collapsed and direct flux weighting is used, however, it will be virtually impossible for the adjoint trial function to yield reasonable spectra. It has been found that results can be improved if the direct flux expansion coefficients are scaled before being used as weighting functions.  

\[ H_{nij}^{g*} = \omega g H_{nij}^{g}. \]  

(35)

D. The Solution of the Synthesis Eigenvalue Equations

Equation (28) can be written in the form of an eigenvalue problem if we drop the inhomogeneous source and break up \( R_{m-mk} \) into its fission and non-fission components.
\[
R_{m',mk} = F_{m',mk} - \lambda F_{m',mk}
\] (36)

where \( \lambda \) is the eigenvalue (1/k).

\( F_{m',mk} \) includes the planar leakage, scattering and absorption components of Equation (24). \( F_{m',mk} \) is the fission part of Eq. (24). \( \lambda \) is the eigenvalue (1/k).

**Fission Source Iteration with Wielandt Acceleration**

SRT3D solves Eq. (28), modified by the Eq. (36), by the fission source iteration method with the convergence accelerated by Wielandt (or fractional) iteration. The statement of one iteration of the solution can be written as a matrix equation,

\[
\begin{bmatrix}
\lambda_k & a_{k-1} & a_k & a_{k+1} & \cdots \\
\end{bmatrix}
\begin{bmatrix}
a_{k-1} \\
a_k \\
a_{k+1} \\
\vdots
\end{bmatrix}
= S_k, \quad k = 1 \ldots K
\] (37)

where \( a_k \) is the unknown vector of combining coefficients \( a_{m',m} \) at \( k \) associated with mesh interval \( k \) and \( a_{k-1}, a_k, a_{k+1} \) and \( S_k \) are defined by

\[
(A_k^+)^{m',m} = \frac{1}{\Delta z_k} D_{m',mk}
\] (38)

\[
(A_k^0)^{m',m} = \frac{1}{\Delta z_{k+1}} D_{m',mk+1}
\] (39)

\[
(S_k)^{m',m} = \sum_{m' at k} \Delta z_k \left( 1 - \lambda_k \right) F_{m',mk} a_{m'}^{k+1}
\] (40)

\[
(A_k)^{m',m} = \frac{1}{\Delta z_k} D_{m',mk} + \frac{1}{\Delta z_{k+1}} D_{m',mk+1} - \Delta z_k \left( a_{m',mk} - \lambda_k F_{m',mk} \right)
\] (41)

\( a_{m',mk} \) is the combining coefficient from the previous iteration. \( \lambda_k \) is an estimate of the eigenvalue. When \( \lambda_k \) is a few percent less than the fundamental eigenvalue (the estimated \( k \) is greater than the fundamental \( k \)) most problems converge in a very few iterations. An estimate of \( k \) is one of the input parameters to the code; when none is provided SRT3D falls back on a straight fission source iteration (\( \lambda_k = 0 \)).

SRT3D requires that for each axial mesh interval the number of group-weighting functions used (i.e. the number of equations) be the same as the number of combining coefficients. This means that the matrix \( A_k^0 \) is square. However, since the number of combining coefficients may change from one mesh interval to the next the matrices \( A_k^+ \) and \( A_k^0 \) may be non-square. Note that, because of Eq. (30), the first term of Eq. (37) vanishes for \( k = 1 \), and the second term vanishes for \( k = K \).

The convergence criterion used in SRT3D is that the eigenvalue, \( \lambda \), not change by more than an input criterion on two successive iterations.
Solution by Forward Elimination, Backward Substitution

The solution of the block-tridiagonal source problem (Eq. (37)) is performed by the forward elimination, backward substitution technique (a specialization of Gauss reduction). The first step is to sweep through the axial mesh generating the following set of matrices, $H_k$, and vectors, $Q_k$.

\[
H_1 = [A_1^0]^{-1} A_1^+
\]

at $k = 1$,

\[
Q_1 = [A_1^0]^{-1} S_1
\]

at $k = 2 \ldots K - 1$,

\[
H_k = [A_k^0 - A_k^* H_{k-1}]^{-1} A_k^+
\]

\[
Q_k = [A_k^0 - A_k^* H_{k-1}]^{-1} [A_k^* Q_{k-1} + S_k]
\]

at $k = K$,

\[
Q_k = [A_k^0 - A_k^* H_{k-1}]^{-1} [A_k^* Q_{k-1} + S_k]
\]

The second step, the backward substitution, is

\[
at k = K, \quad a_k = Q_k
\]

\[
at k = K - 1 \ldots 1, \quad a_k = Q_k + H_k a_{k+1}
\]

Scaling and Differencing

The realities of computer arithmetic can lead to two problems with the synthesis equations. The first occurs when numbers generated during the calculation exceed the maximum or minimum magnitudes allowed by the computer. This can happen during the inversion of Eq. (37) when the normalizations of the expansion functions are unusually large (or small) and/or when the spectra of the expansion functions span too many orders of magnitude. Matrix inversion routines which calculate the determinant are especially vulnerable. This is simply a problem with normalization; when it occurs there is probably nothing fundamentally wrong with the synthesis calculation.

The second problem, linear dependence, was very early recognized as a potential nuisance, but to our knowledge it has never caused much difficulty. If a set of expansion functions in one group is "almost" linearly dependent the inversion of Eq. (37) will fail. The best defense against this, of course, is to know enough about the expansion functions to avoid this situation by dropping superfluous group fluxes from one or more expansion functions (an option available in SYN3D).
As insurance against either of these two problems arising, a transformation is performed on Eq. (37) during its solution. The transformation has the effect of a change of variables from the original synthesis approximation

$$\phi_{ijk}^g = \sum_{n=1}^{N} a_{nk}^g H_{nij}^g ,$$

where $H_{nij}^g$ is the input expansion function, to

$$\phi_{ijk}^g = \sum_{n=1}^{N} b_{nk}^g G_{nij}^g ,$$

where the transformed expansion functions, $G_{nij}^g$, are defined

$$G_{lij}^g = H_{lij}^g / s_l^g ,$$

$$G_{nij}^g = \left( H_{nij}^g / s_n^g \right) - \left( H_{lij}^g / s_l^g \right) , \quad n = 2 \ldots N .$$

$s_l^g$ is a scaling factor used to change the normalization. At each axial mesh interval and in each group this transformation has the effect of:

1. renormalizing whichever expansion function has been designated $n = 1$ (Eq. (49)), and
2. renormalizing and differencing the other functions with respect to the first (Eq. (50)). This differencing scheme is a sort of poor man's orthogonalization.

The relationships between the original and transformed combining coefficients are

$$b_l^g = s_l^g a_l^g + \sum_{n=2}^{N} b_n^g ,$$

$$b_n^g = s_n^g a_n^g , \quad n = 2 \ldots N ,$$

$$a_l^g = \left( b_l^g - \sum_{n=2}^{N} b_n^g \right) / s_l^g ,$$

$$a_n^g = b_n^g / s_n^g , \quad n = 2 \ldots N .$$

The transformation of the matrices in Eq. (37) is governed by
\[ \sum_{g=1}^{G} \sum_{n=1}^{N} A_{mn}^g a_n^g = \sum_{g=1}^{G} \sum_{n=1}^{N} B_{mn}^g b_n^g \]  
(55)

where \( A \) represents any of the matrices \( (A^-, A^+ \) or \( A^0 ) \) operating on the combining coefficients and \( B \) represents a transformed matrix. The relationships between \( A \) and \( B \) are

\[ B_{ml}^g = A_{ml}^g / s_1^g, \]  
(56)

\[ B_{mn}^g = (A_{mn}^g / s_n^g) - (A_{ml}^g / s_1^g), \quad n = 2 \ldots N, \]  
(57)

\[ A_{ml}^g = s_1^g A_{ml}^g, \]  
(58)

\[ A_{mn}^g = s_n^g (B_{mn}^g + (A_{ml}^g / s_1^g)), \quad n = 2 \ldots N. \]  
(59)

The scaling factors, \( s_n^g \), used were taken from the set of axial leakage integrals, Eq. (25).

\[ s_1^g = D_{m'mk} \]  
(60)

where \( m' \) is the composite index for the appropriate group and "first" weighting function, \( H_{1}^{g} \), and \( m \) is the composite index for the same group and \( n \)'th function. This choice was made because (1) the integrals \( D_{m'mk} \) were available and (2) the elements of the transformed matrices in the synthesis equations were on the order of unity.

E. Other Calculations

Multichannel Synthesis

Multichannel synthesis is an extension of the single-channel approximation in which different combining coefficients are assigned to an expansion function in separate regions of the \( x \)-\( y \) plane.\textsuperscript{15} The multichannel trial function can be written,

\[ \phi_{ijk}^g = \sum_{n=1}^{N} \sum_{c=1}^{C_n} a_{nck}^g f_{ncij}^{g} H_{nij}^g. \]  
(61)

Equation (61) differs from Eq. (11) by the additional summation over the index \( c \) (for "channel") and the multichannel basis functions \( f_{ncij}^{g} \). Note that Eq. (61) permits the number and definitions of the sets of basis functions to differ from one expansion function to another.
Two constraints should be placed on the choice of basis functions. The first is that the expression on the right in Eq. (61) must be single-valued for each \((i, j)\). The second is the spatial analog to the restriction placed on the group-collapsing matrix (Eq. (15)). Because it is desirable that the planar flux shape of each expansion function, \(H^g_n\), be recoverable from the synthesis trial function for the three-dimensional flux, \(g^h\),

\[
\sum_{c=1}^{C_n} f^g_{ncij} = \text{the same value for all } g, i \text{ and } j. \tag{62}
\]

Our original intention was to offer multichannel synthesis as an option in SYN3D. After some experience with small test problems and large, fast-critical models, we concluded that the advantage of improved accuracy was outweighed by the disadvantage of longer running times and more complex trial functions, and SYN3D was not carried beyond the single-channel stage. For experimental purposes and for special applications it is possible to perform multichannel calculations with SYN3D by factoring in the multichannel basis functions outside the code,

\[
H^g_{niij} = f^g_{ncij} H^g_n, \quad c = 1 \ldots C_n, \tag{63}
\]

and treating each product, \(f^g_{ncij} H^g_n\), as a separate expansion function.

**Perturbation Theory**

The derivation of a perturbation theory expression for the change in an eigenvalue due to a change in the model, carried out within the framework of single-channel flux synthesis, leads to the same result one obtains with derivations based on the original difference equations. Therefore, any perturbation theory code which is based on the finite-difference equations in Eq. (1) can be used with fluxes and adjoints calculated with SYN3D. It is important to remember, however, that a different interpretation must be given the results. A perturbation theory calculation based on synthesis fluxes and adjoints answers the question "what happens to the synthesis calculation when one perturbs the model?" and not necessarily "what happens to the finite-difference calculation?"

### III. USER CONSIDERATIONS

This section is a discussion of the mechanics of running SYN3D from a user's standpoint. It is mostly directed towards the ARC System SYN3D (as opposed to the stand-alone version available through the Argonne Code Center). Users who have the Code Center version should read Section V before Section III.

**A. CCCC Standards and the ARC System**

A number of features of SYN3D will seem strange unless one is aware of the ground rules under which the code was written. In recent years
code development supported by the Division of Reactor Research and Development of ERDA has been subject to a set of standards defined by the Committee on Computer Code Coordination (CCCC).  

A major part of these standards is a set of definitions of a number of binary files, called standard interface files, which contain most of the data required for neutronics calculations. The ARC System of reactor analysis codes developed at Argonne has its own set of standard binary files. SYN3D was written to conform to the CCCC standards but, at the same time, to be compatible with ARC System modules required in synthesis calculations at Argonne.

Table III-1 lists both the ARC System and CCCC input files used by SYN3D. The ARC System SYN3D user should be aware of both sets of interface files; users of the Code Center version of SYN3D need only be concerned with the CCCC files.

The ARC System SYN3D is really the Code Center SYN3D prefaced by a translator (TRANS1) which reads ARC System files and writes the corresponding CCCC standard interface files. Output files generated by both versions of SYN3D are all in the CCCC formats.

B. Basic Input Requirements

Expansion Functions

The expansion functions, $N^k$, in Eq. (11), used in each synthesis calculation must be calculated in separate jobs; SYN3D does not compute expansion functions. Some suggestions of guidelines for choosing expansion functions based on Argonne's experience with LMFBR calculations are given in Section III-M.

The ARC System SYN3D user supplies a number of expansion functions in either ARC System form (FR.D1, FR.D2, PA.D1 and/or PA.DZ), CCCC form (RTFLUX and/or ATFFLUX) or a mix of the two. The several functions are differentiated by the CCCC device of a file version number. The current ARC System Standard Path for SYN3D (STP018) permits up to 10 flux files in the ARC System format (FR.D1, etc.). The ARC System SYN3D and Standard Path does not distinguish between flux and adjoint files (e.g. FR.D2 vs. PA.D2). The Code Center SYN3D permits 10 flux (RTFLUX) files and five adjoint (ATFFLUX) files.

The user controls the version number of the expansion and weighting functions by his assignment of each file to a logical unit number. In the SYN3D input (SYN3FL) he/she refers to a particular expansion or weighting function by its CCCC designation (e.g. RTFLUX, version number five), even though he/she may have supplied the file in an ARC System format (FR.D2, version number five).

Cross Sections

SYN3D follows the practice of most ARC System neutronics calculations of permitting cross section input in either microscopic or macroscopic form. Microscopic cross sections in the ARC System Standard Path STP018 are in 25.180 files, and the compositions are defined by the type 13 and 14 cards of the BCD file A.NIP.  


TABLE III-1. ARC System and COOC Input Files for SYN3D. These are not all required; some are generated by SYN3D or its ARC System Standard Path, and others may never be needed at all.

<table>
<thead>
<tr>
<th>ARC System</th>
<th>ARC System Binary</th>
<th>COOC Binary</th>
<th>Contents</th>
</tr>
</thead>
<tbody>
<tr>
<td>A.NIP</td>
<td>GEOM</td>
<td>GEODST</td>
<td>Model Geometry</td>
</tr>
<tr>
<td></td>
<td>BC</td>
<td>NDXSRF</td>
<td>Composition Definitions</td>
</tr>
<tr>
<td></td>
<td>B.HOMOG</td>
<td>ZNATDN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>XS.ISO</td>
<td>ISOTNS</td>
<td>Microscopic Cross Sections</td>
</tr>
<tr>
<td></td>
<td>XS.C.MIN</td>
<td>COMPXS(^a)</td>
<td>Macroscopic Cross Sections</td>
</tr>
<tr>
<td></td>
<td>COMPXS(^a)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SYNFIL</td>
<td>SYMCNT(^a)</td>
<td>SYMCNT(^a)</td>
<td>SYN3D Control Data</td>
</tr>
<tr>
<td></td>
<td>FR.D1</td>
<td>KTFLUX</td>
<td>Flux Files</td>
</tr>
<tr>
<td></td>
<td>FR.D2</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>FA.D1</td>
<td>ATFLUX</td>
<td>Adjoint Flux Files</td>
</tr>
<tr>
<td></td>
<td>FA.D2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\(^a\) COMPXS and SYMCNT are code-dependent, binary files which are not officially part of either the ARC or COOC Systems.
STP018 uses the module HOMOG to create the macroscopic cross section file XS.C.MIN, and the ARC System-to-CCCC translator in SYN3D, TRANSL, converts XS.C.MIN to COMPXS. Either XS.C.MIN or COMPXS can be input directly, in which case the preceding steps in the cross section generation procedure will be skipped. COMPXS can accommodate directional diffusion coefficients and "power" cross sections; otherwise it is similar to XS.C.MIN.

The Code Center version of SYN3D can run either from the CCCC microscopic cross section and composition files ISOTXS, NDXSRF and ZNATDN or from the macroscopic cross section file COMPXS. In the first case the SYN3D overlay HMG4C will create a COMPXS file.

**Geometry**

SYN3D builds up a three-dimensional model of a reactor by stacking user-supplied two-dimensional planes. This approach was chosen for two reasons:

1. At the time SYN3D was started, Argonne had no user-oriented, three-dimensional geometry input processor. We did have a well established, one- and two-dimensional input processor in the ARC System module GNIP.26

2. The user was going to have to set up two-dimensional geometries for the expansion function calculations anyway, and it would be convenient if he/she could simply reuse that data without changes.

The ARC System input procedure is particularly suited to generating a series of two-dimensional geometry descriptions. STP018 looks for a BLOCK-GEOM card. If it finds one it expects there to be an A.NIP data set specified for a plane. GNIP processes the A.NIP data to a pair of GEOM and BC files (version number 1) and TRANSL converts them to a GEODST file (again, version number 1). STP018 then looks for another BLOCK-GEOM and a corresponding A.NIP for the next planar geometry description (this time producing a GEODST, version number 2). This procedure continues until all the BLOCK-GEOM data blocks are processed. What makes this scheme convenient is that while the first BLOCK-GEOM must contain a complete A.NIP (DATASET=A.NIP), subsequent blocks need only contain the type 15 cards unique to that plane (MODIFY=A.NIP). The manner in which the two-dimensional planes are stacked to form a three-dimensional model is specified in the SYN3D BCD input file SYNFIL.

It is possible to input a three-dimensional GEODST instead of several two-dimensional files. If the SYN3D BCD file SYNFIL does not specify an axial mesh or a stacking of planes (card types 4 and 5) the code interprets the version 1 GEODST to be a description of the full model and breaks it up into a number of two-dimensional GEODST's (version numbers 2, 3, etc.). This will only work when the expansion functions are also input in CCCC formats (RTFLUX and ATFLUX). This restriction is due to the translator TRANSL, and this mode of input only works when there is no geometry or flux file translation required.
In describing the geometry input we have been using the example of three-dimensional models. The same input procedure works for two-dimensional models built up from one-dimensional "planes".

**Code Dependent Input - SYNFIL**

The special BCD input required by SYN3D is defined by the BCD file SYNFIL. Appendix D is a file description for SYNFIL. SYNFIL must be included in the ARC System input in a data block BLOCK=SYN3D.

**C. Running the ARC System SYN3D - ARCP018**

The catalogued procedure for SYN3D on the Argonne 370/195 is ARCP018. The procedure is listed in Appendix C. ARCP018 executes the synthesis Standard Path STP018, which is listed in Appendix B and discussed briefly in Section IV.

**Input and Output Data Sets**

Table III-2 lists all the input and output data sets that are of interest to users. Besides the files already mentioned in Section III-B, the list includes the direct and adjoint combining coefficient files DCCDEF and ACCOEF, the pointwise power density PWDINT, the composition-averaged flux file RZFLUX and the synthesis integral library files INTTOC, VOLINT and DIFINT. Scratch data sets are discussed in Section IV.

File descriptions for the data sets listed in Table III-2 can be found in one of three places:

For the following ARC System Files see ANL-7711 (Ref. 25)
- GEOM, BC, FR.D1, FR.D2, FA.D1, FA.D2, XS.C.MIN, A.NIP,
- XS.ISO

For the following CCCC Standard Interface Files see Appendix I.
- GEODST, RTFLUX, ATFLUX, PWDINT, RZFLUX

For the following code-dependent files see Appendices D and E.
- SYNFIL, COMPXS, INTTOC, VOLINT, DIFINT, DCCDEF, ACCOEF

**Job Control and Symbolic Parameters**

The SYN3D catalogued procedure ARCP018 (see Appendix C for listing) is designed for the convenience of users operating from ARC System data sets. Table III-2 lists the symbolic parameters available and their default values.

**Sample Input**

Figure III-1 is a listing of an input deck. The problem it represents is Sample Problem 3 of the Code Center SYN3D package (see Section V-E), a three-dimensional model with control rods. There are six BLOCK-GEOM data blocks defining rodded and unrodded core, blanket and reflector planes. The four expansion functions are in the FR.D2 format, and the cross sections are generated from an XS.ISO file.
<table>
<thead>
<tr>
<th>Logical Unit Number</th>
<th>Logical Set Name</th>
<th>Volume Number</th>
<th>Symbolic Parameter</th>
<th>Default Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>L.1</td>
<td>1</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
<tr>
<td>L.2</td>
<td>2</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
<tr>
<td>L.3</td>
<td>3</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
<tr>
<td>L.4</td>
<td>4</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
<tr>
<td>L.5</td>
<td>5</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
<tr>
<td>L.6</td>
<td>6</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
<tr>
<td>L.7</td>
<td>7</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
<tr>
<td>L.8</td>
<td>8</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
<tr>
<td>L.9</td>
<td>9</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
<tr>
<td>L.10</td>
<td>10</td>
<td></td>
<td>FILTF</td>
<td>0</td>
</tr>
</tbody>
</table>

Note: All default volume designations are blank, implying old data sets are cataloged.
PIGUPP III-1.  ARC SYSTEM SYN3D INPUT DECK.  THE THREE-DIMENSIONAL MODEL IS BUILT UP FROM TWO-DIMENSIONAL A, NIP DATA SETS.

//SYMPD EXPD ARCSPD010,
//  FLD01=C116.B21006.TEST3D.PF02.COPP2*;
//  FLD02=C116.B21006.TEST3D.PAT7.COPP2*;
//  FLD03=C116.B21006.TEST3D.PFD2.PLANU*;
//  FLD04=C116.B21006.TEST3D.PFD2.REWFP*;
//  FLD05=DISK9A,FLVOL02=DISK9B,FLVOL03=DISK9C,FLYLO04=DISK9D,
//  FLD151=C116.B21006.IST501.TMPFPP*;
//  FLD152=C116.B21006.IST501.TMPFPP2*;

//SYMPD DP
BLOCK=OLD
DATASET=XS
DATASET=FLXN01
DATASET=FLXN02
DATASET=FLXN03
DATASET=FLXN04
DATASET=A,NIP
BLOCK=GRN

DATASET=A,NIP
TIP=15, "WRODED REFLECTOR PLAN"  

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>01</td>
<td>01</td>
<td>3000</td>
<td>-1</td>
<td>3000</td>
</tr>
<tr>
<td>01</td>
<td>01</td>
<td>3000</td>
<td>-1</td>
<td>3000</td>
</tr>
<tr>
<td>06</td>
<td>COPP1</td>
<td>0.0</td>
<td>15.0</td>
<td>0.0</td>
</tr>
<tr>
<td>06</td>
<td>COPP2</td>
<td>0.0</td>
<td>55.0</td>
<td>0.0</td>
</tr>
<tr>
<td>06</td>
<td>COPP3</td>
<td>0.0</td>
<td>55.0</td>
<td>0.0</td>
</tr>
<tr>
<td>06</td>
<td>COPP4</td>
<td>0.0</td>
<td>55.0</td>
<td>0.0</td>
</tr>
<tr>
<td>06</td>
<td>COPP5</td>
<td>0.0</td>
<td>55.0</td>
<td>0.0</td>
</tr>
<tr>
<td>06</td>
<td>COPP6</td>
<td>0.0</td>
<td>55.0</td>
<td>0.0</td>
</tr>
<tr>
<td>06</td>
<td>COPP7</td>
<td>0.0</td>
<td>55.0</td>
<td>0.0</td>
</tr>
<tr>
<td>06</td>
<td>COPP8</td>
<td>0.0</td>
<td>55.0</td>
<td>0.0</td>
</tr>
<tr>
<td>06</td>
<td>COPP9</td>
<td>0.0</td>
<td>55.0</td>
<td>0.0</td>
</tr>
<tr>
<td>06</td>
<td>COPP10</td>
<td>0.0</td>
<td>55.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

BLOCK=GRN

DATASET=A,NIP
TIP=15, "WRODED REFLECTOR PLAN"  

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
</tr>
</tbody>
</table>

MODIFY=4

DATASET=A,NIP
TIP=15, "WRODED REFLECTOR PLAN"  

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
</tr>
</tbody>
</table>
FIGURE 7.1-1. ANG SYSTEM SYM-1 INPUT DECK. CONTINUED.

BLOCK-CORE
MODIFY-A.1
31 TEST10, UNMODDED CORE PLANE
15 ID MOD1 MOD2 MOD3 MOD4 MOD5
15 IF KEEL
15 ID BIAN
15 ID CORE1
15 ID CORE2

BLOCK-CORE
MODIFY-A.110
01 TEST10, UNMODDED CORE PLANE
15 ID MOD1 MOD2 MOD3 MOD4 MOD5
15 IF KEEL
15 ID BIAN
15 ID CORE1
15 ID CORE2

BLOCK-CORE
MODIFY-A.111
01 TEST10, MODDED PLANE PLANE
15 ID MOD1 MOD2 MOD3 MOD4 MOD5
15 IF KEEL
15 ID BIAN
15 ID CORE1
15 ID CORE2

BLOCK-CORE
MODIFY-A.1111
01 TEST10, MODDED REFLECTOR PLANE
15 ID MOD1 MOD2 MOD3 MOD4 MOD5
15 IF KEEL
15 ID BIAN
15 ID CORE1
15 ID CORE2

BLOCK-CORE
MODIFY-A.11111
01 TEST10, 1-D.1ENSIONAL MODEL, PHOX WEIGHTED SYM-THY
01 PLANE PLANE (7 AND 4) NOT USED EVERYWHERE
01 PLANE PLANE BUILT UP FROM 2D GROST FILES
02 1 2
03 1 1
04 2 1 1.11 10.0
05 3
06 4
07 RVALUE 1
08 RVALUE 2
09 RVALUE 3
10 RVALUE 4
11 1 2 3 4
12 1 2 3 4 5 6
13 1 2 3 4 5 6
14 1 2 3 4 5 6
15 1 2 3 4 5 6
16 1 2 3 4 5 6
17 1 2 3 4 5 6
18 1 2 3 4 5 6
19 1 2 3 4 5 6
20 1 2 3 4 5 6
21 1 2 3 4 5 6
D. Multiple Problems and Restarts

For most problems the most expensive part of a SYN3D calculation is generating the integrals (Eqs. (24) and (25)) which form the coefficients of the synthesis equations. If the same integrals can be used for a number of synthesis calculations the user can save a significant amount of time.

Multiple Problems

The simplest way to reuse integrals is to run several problems in a single job. The ARC System SYN3D does this automatically by looping on BLOCK=SYN3D data blocks. A problem defined by a second BLOCK=SYN3D uses integrals generated under the first BLOCK=SYN3D, calculating additional integrals if required. The output flux file from each loop can be saved by specifying different version numbers for the output RTFLUX (see SYNFIL card type 16). Only the last set of combining coefficients can be saved, however, since SYN3D always writes them to DCCOEF version number 1.

This multiple pass feature is not available in the Code Center SYN3D. Code Center SYN3D users will have to use the more formal restart procedure described in the next paragraph.

Restarts - INTTOC, VOLINT and DIFINT

The integrals SYN3D calculates are stored in a library consisting of three files: INTTOC, VOLINT and DIFINT. SYN3D always writes these files with a version number of 2. If these files are input with a version number of 1 the code will merge the old integrals with whatever integrals it still must calculate before writing the output libraries. The version 1 files are not changed.

File descriptions for the library files are in Appendix E. INTTOC is a short file containing the table of contents for VOLINT and DIFINT as well as the general problem data. DIFINT contains the integrals associated with the axial leakage, D_{m'\mu k} (Eq. (25)). VOLINT contains four types of integrals which are referred to in the coding and in the SYN3D output as REM, FIS, POW and FLUX. REM and FIS are, respectively, the removal and fission components of R_{m'\mu k} (see Eqs. (24) and (36)). POW is the planar power integral,

\[ \text{POW}_{nk} = \sum_{g=1}^{G} u_{gb} \sum_{i=1}^{I} \sum_{j=1}^{J} \Delta x_{i} \Delta y_{j} \text{p}_{ijg} \text{H}_{ijg} . \]  

where p_{ijg} is a "power" cross section supplied in the COMPXS file. Eq. (64) is written for an x-y plane; one can obtain the expression for other geometries by replacing \( \Delta x_{i} \Delta y_{j} \) by the appropriate volume element. The FLUX integral is, as one might guess,

\[ \text{FLUX}_{n} = \sum_{g=1}^{G} u_{gb} \sum_{i=1}^{I} \sum_{j=1}^{J} \Delta x_{i} \Delta y_{j} \text{H}_{ijg} . \]  

(65)
The index $b$ refers to a collapsed group. Recall that when group collapsing is not used $\delta_{gb}$ is the Kronecker delta ($\delta_{gb} = 1$ when $g = b$, $0$ otherwise).

One can restart SYN3D to do another synthesis calculation or simply to obtain further edits for a problem run previously. In the latter case DCCOEF and/or ACCOEF must be among the input data sets.

SYN3D cannot be restarted in the middle of the synthesis integral calculation or the middle of the solution routine. If the job terminates before completing one of those two stages, that particular stage must be repeated.

E. Edits and Output Options

SYN3D has some standard edits, over which the user has no control, and some optional edits. Appendix H contains output from the sample problems in the Code Center package (Section V-E).

Standard Output

SYN3D always lists the SYNFIL input data and two pages describing the model and synthesis trial function (in terms of CCCC files). When logarithmic boundary conditions are used the constants $C_1$ and $C_2$ listed in the edits refer to Eq. (2) in Section 16.

\[
C_1^g D^g \hat{n} \cdot \nabla \phi^g + C_2^g \phi^g = 0.
\]  

(2)

In addition, SYN3D edits the integral tables of contents, the eigenvalue and the combining coefficients ($\alpha_{kl}^g$).

The standard output files are:

- INTTOC, 2
- DIFINT, 2
- VOLINT, 2
- DCCOEF, 1 (for a direct solution)
- ACCOEF, 1 (for an adjoint solution)

Optional Output

SYNFIL card type 16 through 21 specify output options. Flux and adjoint edits may be specified by group and plane. Pointwise power density edits may be requested by plane. SYN3D will also edit the composition-averaged group fluxes (the contents of the RZFLUX file) and the perturbation theory denominator,

\[
g^G \sum_{i=1}^{G} \sum_{j=1}^{J} \sum_{k=1}^{K} \phi_{ijk}^g \nu_{gijk}^g \phi_{ijk}^g \phi_{ijk}^g v_{ijk},
\]  

(66)

where the symbols were defined for Eqs. (1) and (6).
The optional output files are:

- KTFLUX, N  (N specified by user)
- ATFLUX, N  (N specified by user)
- PWDFLT, N  (N specified by user)
- RZFLUX, N  (N specified by user)
- GEODST, N  (N specified by user)

The output, three-dimensional (or two-dimensional) GEODST file may be useful when the input geometry has been specified by a set of two-dimensional (or one-dimensional) GEODST files.

Combining Coefficient Plots

SYN3D will edit printer plots of the axial power distribution,

\[
(power)_k = \sum_{n=1}^{N} \sum_{b=1}^{G'} PON_{n,k}^b a_{nk}^b ,
\]

and the axial group fluxes (after group collapsing)

\[
(flux)_k^b = \sum_{n=1}^{N} FLUX_{n,k}^b a_{nk}^b .
\]

\(PON_{n,k}^b\) and \(FLUX_{n,k}^b\) are defined by Eqs. (64) and (65). The plots show the totals (Eqs. (67) and (68)) as well as the modal components (for each \(n\)). The output from Sample Problem 3 in Appendix H shows one of these plots.

F. Error Messages

Errors caught by SYN3D during execution are identified by subroutine name and an error number. Appendix F is a complete list. Non-fatal errors (positive error numbers) are flagged but never cause the termination of a run; they are frequently anomalies in the input which suggest user errors. Fatal errors will not cause the job to terminate immediately; SYN3D will continue as far as it can before forcing an abnormal termination through subroutine ERROR.

G. Storage Requirements and Running times

Data storage during a SYN3D calculation is managed dynamically through the BPOINTER routines (see Section IV-C for a description of BPOINTER). The BPOINTER storage container size required on input card type 3 of data set SYNFIL should be set as large as practical for models of any significant size. SYN3D keeps in-core as much data as it can in order to minimize I/O time and makes use of all the container it is given. A very rough guess at the minimum size required is the largest of the three expressions given below:
or 5000 + 7*(N*G)^2.

(69)

G is the number of groups, I is the number of first-dimension mesh intervals, J is the number of second-dimension mesh intervals, and N is the maximum number of expansion functions used simultaneously at any one axial mesh interval. Users of the CDC version of the Code Center SYN3D may run into trouble at about N*G = 50. Some coding changes are suggested in Section V-F which will alleviate the problem.

Running time can vary dramatically, depending on the complexity of the model and the synthesis scheme used. The time required to do the integrals is proportional to the square of the number of groups (before group collapsing), the number of planar mesh points, the number of different axial zones, and the square of the number of expansion functions. The time for the solution of the synthesis equations is proportional to the number of iterations, the number of axial mesh intervals, the cube of the number of groups (after group collapsing) and the cube of the number of expansion functions.

Table III-3 shows running times for two fast critical assembly models.

H. Applications of Synthesis

Choosing Expansion Functions

The choice of expansion functions is very important to the success of the synthesis approximation, but we will touch on that subject only very briefly in this report. The experiences of many users can be found in the Refs. 7, 9, 16, 19, 20, 27-30. We will summarize here some of Argonne's recent experience with SYN3D calculations of fast critical assemblies.27-29

We have settled on different prescriptions for high-reactivity (core) and low-reactivity (blanket and reflector) planes of a model. For high-reactivity planes the best choices are critically buckled, eigenvalue calculations. It is not necessary to fine-tune the buckling to force the two-dimensional eigenvalue to be exactly that expected for the three-dimensional model; 1% or 2% differences are probably close enough. We have tried expansion functions which are calculated with group- and region-dependent bucklings determined from an r-s model; synthesis calculations based on these expansion functions give better results than calculations based on uniformly buckled expansion functions, but the advantages are probably outweighed by the nuisance of setting up the extra r-s model. Our current practice is to choose a single, constant buckling,

\[ b^2 = (w/L)^2, \]

(70)

based on an effective unreflected core height, L, which is within 10 cm. of the value that would be required to get the expected critical eigenvalue.
**TABLE III-3.** ARC System SYN3D Running Times and Storage Requirements for Two Fast Critical Assembly Models on the IBM 370/195. The GCFR Model was Run at Two Different Container Sizes

<table>
<thead>
<tr>
<th>Model</th>
<th>GCFR</th>
<th>ZPPR3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Groups</td>
<td>11</td>
<td>28</td>
</tr>
<tr>
<td>Spatial Mesh (x-y-z)</td>
<td>25 x 27 x 18</td>
<td>54 x 25 x 22</td>
</tr>
<tr>
<td>Axial Zones</td>
<td>4</td>
<td>3</td>
</tr>
<tr>
<td>Expansion Functions</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>Iterations</td>
<td>5</td>
<td>4</td>
</tr>
<tr>
<td>BPOINTER Container</td>
<td>40,000 8,000*</td>
<td>60,000</td>
</tr>
<tr>
<td>REGION Size (K-bytes)</td>
<td>650 375</td>
<td>800</td>
</tr>
<tr>
<td>CP Time (min)</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>WAIT Time (min)</td>
<td>1.7</td>
<td>2.4</td>
</tr>
<tr>
<td>Total Time (min)</td>
<td>2.6</td>
<td>3.3</td>
</tr>
</tbody>
</table>

*This was the minimum container size, rounded to the next 1000 words, for which the GCFR model would run.
For low-reactivity planes such as a blanket zone, or no-reactivity planes like a reflector, an eigenvalue calculation is a poor (or even impossible) way to generate characteristic planar shapes. Our current practice is to obtain expansion functions from inhomogeneous calculations in which the fixed source is the product of the diffusion coefficient in the zone of interest times the expansion function characteristic of the adjacent, higher-reactivity plane. The philosophy behind this choice is that the most important source of neutrons in a low-reactivity plane is the leakage from a high-reactivity plane, and that leakage source is roughly proportional to:

1. the neutron distribution in the high-reactivity plane, and
2. the ability of neutrons to diffuse into the low-reactivity plane.

We have applied this prescription almost entirely to axial blankets; we are not usually so interested in reflector zones that it is necessary to supply a reflector expansion function.

Blanket Functions - STP016, AJC10 and ARCSP016

An ARC System Catalogued Procedure, ARCSP016, and Standard Path, STP016, have been written to set up a fixed source for a subsequent DIF1D or DIF2D calculation of a low-reactivity expansion function. ARCSP016 is listed in Appendix J, and STP016 is listed in Appendix K. Basically, STP016 calls the usual ARC System geometry and cross section processing modules to generate GEOM and XS.C.MIN files and then calls the module AJC010. AJC010 multiplies an input flux distribution (FR.D1 or FR.D2) by the local diffusion coefficients and writes a fixed source file (ES.D1D or ES.D2D). The symbolic parameters for ARCSP016 are listed in Appendix J.

An ARCSP016 job step is usually followed immediately by an ARCSP001 (DIF1D) or ARCSP003 (DIF2D) job step. Figure III-2 shows an input deck for calculating a blanket function for the sample problems described in Section V-E. Quirks in the DIF2D Standard Path make it necessary to include a type 19 card in A.NIP (even though the data on that card is not used) and to include DATASET=ES.D2SH in the DIF2D BLOCK=OLD (even though no such data set exists). Note that the A.NIP created in the first step is passed to the second.

IV. CODE STRUCTURE AND PROGRAMMING CONSIDERATIONS

This section describes the structure of SYN3D from a programmer's standpoint. It is a survey of the several distinct code blocks making up SYN3D and is intended as an aid to users who wish to make changes to the code or who wish to understand the flow of the calculation.

A. The ARC System SYN3D

The differences between the Code Center SYN3D and the ARC System SYN3D are in operating environment; the bulk of the coding (specifically,
FIGURE ITI-2. INPUT FOR A BLANKET EXPANSION FUNCTION CALCULATION USING ARCSP016 AND ARCSP01 (DF2D).

```plaintext
//STE3 EXEC ARCSP016,
// COMPS1='C116.21006.TEST3D.XSCMIN.P1',
// COMPS2='C116.R21006.TEST3D.XSCMIN.P2',
// CSCLSP=(OLD,KEEP),CSVOL=DISK99,
// FLUX2D='C116.921006.TEST3D.PRD2.COMBO',FLUXWOL=DISK99
//SYSIN ND *
BLOCK=OLD
DATASET=FR.D7
DATASET=XS.C.MIN
BLOCK=STP016
DATASET=A.4TP
01 TEST3D, UNRODED BLANKET PLANE
02 0 0 3000 -1 3000 -1 0 1

15 AB BLANK CORP1 CORP2
15 PC FRO1 FRO2 FRO3 FRO4 FRO5
19 XEPF 1.0
/*
//STP3 EXEC ARCSP003,
// COMPS1='C116.21006.TEST3D.XSCMIN.P1',
// COMPS2='C116.21006.TEST3D.XSCMIN.P2',
// CSCLSP=(OLD,KEEP),CSVOL=DISK99,
// VCKS1=NULLFILE,
// MCKS1=NULLFILE,
// MCKS2=NULLFILE,
// RALUX='C116.21006.TEST3D.PRD2.BLANK',
// RALVOL=DISK99,
// FHD3='(OLD,'3FP',
// //FS19001 DC 'SN=SCNT,UNIT=S*SCR,DISP=(OLD,DELETE),
// //FS16001 DD DSN=FRO1,DISP=(OLD,DELETE),VOL=SER=SCR001
// //FS14001 DD DSN=(OLD,DELETE),UNIT=S130,VOL=SER=SCR001
// //SYSTN ND *
BLOCK=OLD
DATASET=A.4TP
DATASET=XS.C.MIN
DATASET=FS.D9D
DATASET=FS.D2S
BLOCK=STP003
DATASET=A.4DF2D
01 TEST3D, UNRODED BLANKET PLANE
01 0 0 0 3 1 5 0 1
01 1 1 1
02 1 1 1 8 3
04 1.0E-5 1.0E-5 1.0E-5 0.0 0.0
05 3000 1 3 3 8 3
06 1.0E-5 0.0 0.0
DATASE=1.DDF2D
0"
the overlays CARDS, INTACG, SOLVE and EDITS) is identical. The ARC System synthesis Standard Path, STP018, calls a number of modules (among them SYN3D) in performing a synthesis calculation; the Standard Path and associated, additional ARC System modules have not been included in the Code Center SYN3D package.

STP018

Appendix B is a listing of the ARC System synthesis Standard Path, STP018. In addition to the SYN3D module, which is designated NUC012 in the ARC System, STP018 may execute one or more of the following modules:

NU1001 The cross section homogenization input processor.
NUC001 Cross section homogenization.
NU1004 Triangular mesh geometry input processor.
NU1002 General geometry input processor.

File Definitions

The ARC System SYN3D uses ARC System files, Version III CCCC files and a number of scratch files. The ARC System and CCCC files are discussed in Section III of this report. The official definitions of the required Version III CCCC files are given in Appendix I along with a discussion of the one instance (in a triangular mesh GEODST file) where the SYN3D requirements and the official definitions differ. Scratch files are defined in Appendix E.

B. Overlay Structure

Figure IV-1 is a diagram of the SYN3D module indicating the calling sequences of the overlays and subroutines. The Code Center version of the code does not include the TRANSL branch. The ARC System SYN3D does not contain the HNG4C branch. The functions of the main driver and each of the six primary overlays are discussed below. Table IV-1 lists each subroutine and common block and includes a brief description of functions performed.

Main Driver

The main drivers of the Code Center and ARC System versions of SYN3D differ somewhat in form but are similar in function. Each is a short routine which sets the values of a number of code dependent parameters and then calls the primary overlays which perform the synthesis calculation. In the Code Center version the main driver handles the SEEK initialization; in the ARC System this function is accomplished in the Standard Path.

TRANSL

TRANSL is a code block required in the ARC System SYN3D as an interface between ARC System files and the CCCC files that SYN3D works with.
FIGURE IV-1. SVEID PROGRAM STRUCTURE SHOWING SUBROUTINE CALLING SEQUENCES.
TRANSL IS USED ONLY IN THE ARC SVEID, McGOC IS USED ONLY IN THE CODE CENTER VERSION. SUBROUTINES CALLED BY SEVERAL SUBROUTINES IN A BRANCH ARE LISTED UNDER THE BRANCH NAME. FOR EXAMPLE, USER IS USED THROUGHOUT THE CODE, QUIT IS CALLED BY SEVERAL SUBROUTINES IN McGOC, ETC.
TABLE IV-1. SYMD SUBROUTINES AND COMMON BLOCKS.

**ROUTINES USED THROUGHOUT SYMD.**

| RNLX | POINTER ROUTINE, INITIALIZES BULK STORAGE (LCM). |
| CLRPM | POINTER ROUTINE, ERRORS ARRAY. |
| ERRPM | ERROR ROUTINE. |
| FRPM | POINTER ROUTINE, RELEASES POINTER CONTAINER. |
| FRPM | POINTER ROUTINE, RETURNS POINTER FOR SUBARRAY. |
| FRPM | POINTER ROUTINE, RETURNS NUMBER OF POINTER ERRORS. |
| FRPM | POINTER ROUTINE, RETURN POINTER FOR ARRAY. |
| LENS | BASE MEANINGS. |
| PNTRM | POINTER ROUTINE, INITIALIZATION. |
| PNTM | POINTER ROUTINE, SHIFTS ARRAY STORAGE TO SQUEEZE OUT BLANKS. |
| PNTM | POINTER ROUTINE, RESERVES STORAGE FOR ARRAY. |
| PNTM | POINTER ROUTINE, CHANGES STORAGE FOR ARRAY. |
| PNTT | "LAST INTO" FILE. |
| REMP | READS BINARY FILES. |
| WRPM | WRITES BINARY FILES. |
| SEMP | CCCC FILE MANAGER. |
| TTRPM | READS FILE, DATE, PTC. |
| TTRPM | POINTER ROUTINE, RELEASES STORAGE FOR ARRAY. |

**COMMON BLOCKS.**

| ARRAY | POINTER CONTAINER. |
| HTHPM | POINTER POINTERS FOR BHC. |
| NUPRM | TO PARAMETERS. |
| LEDM | DATA FOR LINES. |
| LOCAT | POINTERS PARAMETERS. |
| MATER | GENERAL SYMD DATA. |
| OUTPP | TO PARAMETER. |
| PNTT | POINTER POINTERS FOR INPUT. |
| POITM | POINTER POINTERS FOR CARDS. |
| POITM | POINTER POINTERS FOR EDITS. |
| POITM | POINTER POINTERS FOR SOLVE. |
| WNP | GENERAL HGC DATA. |
| SCAT | SCATTERING CROSS SECTION PARAMETERS FOR HGC. |
| TRYM | DATA FOR TRAM. |
TABLE IV-1. SYN3D SUBROUTINES AND COMMON BLOCKS. CONTINUED.

TRANSL - ARC TO CCC CONVERSION (ARC SYSTEM ONLY).

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TPLUX</td>
<td>Reads PR. D1, PR. D2, PA. D1, PA. D2, writes RTPLUX.</td>
</tr>
<tr>
<td>TMX01</td>
<td>Reads TION and BC, writes GEODST.</td>
</tr>
<tr>
<td>TRACR</td>
<td>Translates boundary conditions.</td>
</tr>
<tr>
<td>TISCR</td>
<td>Reads 2-FILE X.R.C.MX, writes COMPS.</td>
</tr>
</tbody>
</table>

CARDS - PROCESSING CONF-DEPENDENT, BCD DATA.

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>RENDP</td>
<td>Breaks 3D MODEL GEOMETRY into several 2D GEODST FILES.</td>
</tr>
<tr>
<td>READCD</td>
<td>Reads BCD CARD INPUT.</td>
</tr>
<tr>
<td>PITSTN</td>
<td>Writes SYNCOND FILE.</td>
</tr>
</tbody>
</table>

PNSAC - CREATES MACROSCOPIC CROSS SECTION FILE, COMPS (CODE CENTER SYN3D ONLY).

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ATDF</td>
<td>Reads Zone Atom Densities.</td>
</tr>
<tr>
<td>DGLPDF</td>
<td>Moves DOUBLE-PRECISION DATA.</td>
</tr>
<tr>
<td>EDITISO</td>
<td>Edits RECORDS 2 AND 3 OF ISOTIS FILE.</td>
</tr>
<tr>
<td>EDITIS1</td>
<td>Edits RECORDS 1 AND 2 OF COMPS.</td>
</tr>
<tr>
<td>EDITIS2</td>
<td>Edits RECORDS 3 AND 4 OF COMPS FOR A SINGLE COMPOSITION.</td>
</tr>
<tr>
<td>FARMET</td>
<td>Translates OPFD and moves SINGLE-PRECISION DATA.</td>
</tr>
<tr>
<td>FT4SPEC</td>
<td>Computes Prompt Fission Spectrum.</td>
</tr>
<tr>
<td>I1MES</td>
<td>Moves INTEGER DATA.</td>
</tr>
<tr>
<td>ISORT1</td>
<td>Reads RECORDS 1-4 OF ISOTIS FILE.</td>
</tr>
<tr>
<td>ISORT2</td>
<td>Reads ISOTOPP-DEPENDENT ISOTIS RECORDS.</td>
</tr>
<tr>
<td>O111</td>
<td>DRIP for INITIAL CROSS SECTION INPUT PROCESSING.</td>
</tr>
<tr>
<td>O112</td>
<td>DRIP for MICROSCOPIC DATA PROCESSING.</td>
</tr>
<tr>
<td>O113</td>
<td>Writes COMPS FILE.</td>
</tr>
<tr>
<td>O114</td>
<td>Edits COMPS FILE.</td>
</tr>
<tr>
<td>O115</td>
<td>Prints ERROR MESSAGE.</td>
</tr>
<tr>
<td>O116</td>
<td>Reads XDISP FILE.</td>
</tr>
<tr>
<td>P1ATCH</td>
<td>Reads 2NATPN FILE.</td>
</tr>
<tr>
<td>S1NGET</td>
<td>Moves SINGLE-PRECISION DATA.</td>
</tr>
<tr>
<td>SYSTSC</td>
<td>Scattering Cross Section Processing.</td>
</tr>
<tr>
<td>SYTIS</td>
<td>Picks up MACROSCOPIC DATA.</td>
</tr>
<tr>
<td>TYP</td>
<td>Communication with the REST OF SYN3D.</td>
</tr>
<tr>
<td>U1PDATE</td>
<td>Builds up MACROSCOPIC DATA.</td>
</tr>
<tr>
<td>W1REC</td>
<td>Writes RECORD TYPE 1 OF COMPS FILE.</td>
</tr>
<tr>
<td>W1REC2</td>
<td>Writes RECORD TYPE 2 OF COMPS FILE.</td>
</tr>
<tr>
<td>W1REC3</td>
<td>Writes RECORD TYPE 3 OF COMPS FILE.</td>
</tr>
<tr>
<td>W1REC4</td>
<td>Writes RECORD TYPE 4 OF COMPS FILE.</td>
</tr>
</tbody>
</table>
**TABLE TV-1. SYN3D SUBROUTINES AND COMMON BLOCKS. CONTINUED.**

**INTFG** - Calculates synthesis integrals.

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DECIDE</td>
<td>Chooses integrals to be done on each pass through INTEG.</td>
</tr>
<tr>
<td>EDMOD</td>
<td>Edits model description.</td>
</tr>
<tr>
<td>EDTOC</td>
<td>Edits integral table of contents.</td>
</tr>
<tr>
<td>INPRO1</td>
<td>Processes code-dependent input from SYNCON.</td>
</tr>
<tr>
<td>INPRO2</td>
<td>Determines what integrals must be calculated.</td>
</tr>
<tr>
<td>INPRO4</td>
<td>Makes data management decisions.</td>
</tr>
<tr>
<td>INTRIT</td>
<td>Writes records of output integral file.</td>
</tr>
<tr>
<td>INT1</td>
<td>Calculation of scattering, absorption and fission integrals.</td>
</tr>
<tr>
<td>INT2</td>
<td>Calculation of planar leakage integrals.</td>
</tr>
<tr>
<td>INT21</td>
<td>Calculation of planar leakage integrals.</td>
</tr>
<tr>
<td>INT22</td>
<td>Calculation of axial leakage integrals.</td>
</tr>
<tr>
<td>INT23</td>
<td>Calculation of axial leakage integrals.</td>
</tr>
<tr>
<td>INT24</td>
<td>Calculation of flux and power integrals.</td>
</tr>
<tr>
<td>INTPLC</td>
<td>Returns data rewrite parameters.</td>
</tr>
<tr>
<td>PLACE</td>
<td>Stores integrals in output arrays.</td>
</tr>
<tr>
<td>REPTIT</td>
<td>Rewrites tying, geometry data and cross sections.</td>
</tr>
<tr>
<td>RITTOC</td>
<td>Writes INTOC file.</td>
</tr>
<tr>
<td>SFPTOP</td>
<td>Opens output integral files.</td>
</tr>
<tr>
<td>WRITEX</td>
<td>Writes process cross sections for REPTIT.</td>
</tr>
</tbody>
</table>

**SOLVF** - Spts up and solves synthesis equations.

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>APITH</td>
<td>Matrix arithmetic.</td>
</tr>
<tr>
<td>CALPOW</td>
<td>Calculates total power.</td>
</tr>
<tr>
<td>DIVVY</td>
<td>Makes data management decisions.</td>
</tr>
<tr>
<td>PILLUP</td>
<td>Builds equation coefficient matrices from INTGLS file.</td>
</tr>
<tr>
<td>INVPRT</td>
<td>Matrix inversion.</td>
</tr>
<tr>
<td>KITIC</td>
<td>Returns axial mesh data.</td>
</tr>
<tr>
<td>ORTH</td>
<td>Change of variable transformation of synthesis equations.</td>
</tr>
<tr>
<td>ONTPRO</td>
<td>Edits, writes combining coefficient file.</td>
</tr>
<tr>
<td>SHIFPT</td>
<td>Shifts integrals in equation coefficient matrices.</td>
</tr>
<tr>
<td>SPROM</td>
<td>Directs solution of source problem.</td>
</tr>
<tr>
<td>INTAT</td>
<td>Writes integrals to INTGLS file.</td>
</tr>
</tbody>
</table>

**EDITS** - Edit package.

<table>
<thead>
<tr>
<th>Subroutine</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PLOT3D</td>
<td>Ptflux, atflux, edinit and rzflux edits and output files.</td>
</tr>
<tr>
<td>PPDEMON</td>
<td>Perturbation denominator.</td>
</tr>
<tr>
<td>PPPLT</td>
<td>Printer plotter.</td>
</tr>
<tr>
<td>PLOTT</td>
<td>Spts up combining coefficient plots.</td>
</tr>
<tr>
<td>REDEN</td>
<td>Input processor for EDITS.</td>
</tr>
<tr>
<td>RTGDR</td>
<td>Writes grodst.</td>
</tr>
<tr>
<td>TWOSDP</td>
<td>Table edits.</td>
</tr>
</tbody>
</table>
When there are input GEOM and BC files TRANSL translates the data into corresponding GEODST files. When there are input FR.D2 (or FR.D1) files TRANSL creates the corresponding RTFLUX files. It distinguishes between FR.D2 and FR.D1 by checking the dimensionality of the GEODST files. Finally, TRANSL rewrites the data in an input XS.C.MIN file into the COMPXS format.

CARDS

The main purpose of CARDS is to read BCD data and write the binary file SY NC. The value of the integer variable IWHERE (defined in the main driver and transmitted to CARDS through the common block HOWIN) identifies the operating environment (ARC System or Code Center) and determines where the BCD data come from.

In the Code Center version (IWHERE = 1) the SYN3D input, starting with the cards-per-card-type data (see section V of this report), is read from the card input file. The logical unit number of the card input file, NFLIN, is set in the main driver and transmitted to CARDS through the common block HOWIN. In the ARC System SYN3D (IWHERE = 2) the BCD input is contained in the ARC System BCD file SYNFIL. In this case the logical unit number is determined through a call to SEEK.

The subroutine READCD reads and stores all the BCD data. The subroutine RITSYN writes the SYNCON file, which is essentially a binary version of SYNFIL (see Appendix E). The first record contains a single number, the largest card type number. The second record contains the number of input cards for each card type. Each subsequent record contains the data on an input card, with the card type number (columns 1-2) omitted.

If the user wishes to install some other input procedure, CARDS can be eliminated entirely, just so long as it is replaced with some other method of generating SY NC.

CARDS performs one other function. SYN3D normally operates with separate GEODST files for each of the planes of a model plus additional data in SY NC specifying how the planes are to be stacked together. When a single GEODST file is input, and the "stacking" instructions (card types 4 and 5) are omitted, subroutine BRKUP converts the single input GEODST to a number of GEODST files of lower dimensionality and constructs the necessary card type 4 and 5 data before writing SY NC.

HMG4C

HMG4C is used in the Code Center version of SYN3D to create macroscopic cross sections from input microscopic data; in the ARC System environment this calculation is performed outside the SYN3D module by the ARC System modules NU001 and NU001, and HMG4C is not used. HMG4C reads the three CCCC files NDXSRF, ZNATDN and ISOTXS and writes the macroscopic cross section file COMPXS (see Appendix E). When a COMPXS file is input to the code, the main driver recognizes the fact and omits the call to HMG4C.

After reading ZNATDN, NDXSRF and the isotope independent data of ISOTXS, into core HMG4C attempts to hold all the macroscopic arrays which
are to be computed in the remaining BPOINTER container space. If this is possible a single pass is made through the ISOTXS file and the contribution of each isotope is added to each macroscopic cross section of each composition containing that isotope. If all the macroscopic data will not fit in the available core, the code determines the maximum number of compositions which will fit in a single pass. As many passes through ISOTXS are then made as are required to process all the data. The results of each pass are written to a scratch file (HFILE) for temporary storage before being written to the output COMPXS.

The COMPXS file has provision for directional diffusion coefficient data. HMG4C only supplies default values for this data - unity for the multipliers and zero for the additive terms (see Appendix E). To use directional diffusion coefficients in SYN3D the user must supply an appropriate COMPXS file from outside the code.

**INTEG**

The integrals required for the synthesis equations (see Section II of this report) are calculated by the INTEG overlay. These integrals are written into the library files VOLINT and DIFINT, with a table of contents and supporting data for the model written to the INTTOC file (see Appendix E).

Subroutine INPRO1 reads input data from SYNCON and checks it for errors. Subroutines INPRO2 and INPRO4 inspect the model, determine what integrals are required and make data management decisions for the calculation of the integrals.

REWRI T rewrites the expansion and weighting function fluxes and the planar geometry descriptions into the REQFLX file (see Appendix E) and rewrites the required macroscopic cross sections into REQXST (see Appendix E). All the point group fluxes and mesh interval data associated with a number of adjacent rows of mesh intervals are combined into each record of REQFLX. The size of the records and, therefore, the number of rows represented in each record are determined at run time and depend on the available BPOINTER container space. The macroscopic cross sections are rewritten with the diffusion coefficients segregated (in separate records) from the removal and fission data. Again, the REQXST record size is determined at run time. Each record of removal/fission data contains cross sections for all groups for as many compositions as the record size allows. Each record of diffusion coefficient data contains coefficients for all compositions for as many groups as the record size allows.

The integrals are calculated inside a nest of loops over records of REQFLX and REQXST. Space is reserved for as many output records of integral library files (VOLINT and DIFINT) as can fit in core with one record of REQFLX and one record of REQXST. The integrals to be saved in those output records are built up during loops over REQFLX records (the inner loop) and REQXST records (the outer loop). The order in which the integrals are to be done is determined in subroutine DECIDE; in an effort to minimize arithmetic, integrals requiring the same two functions are done simultaneously.
The manner in which the input fluxes, geometries and cross section data are rewritten for a particular job is described in the SYN3D output under the heading "DATA MANAGEMENT PARAMETERS FOR INTEGRAL CALCULATION". The table of contents of the output DIFINT and VOLINT files is also edited.

Although the exact size of each VOLINT and DIFINT record depends on the integrals contained, a maximum record size, LENINT, is set in the main driver. The current value of LENINT has been arbitrarily set at 2000 words (REAL*8 words on IBM machines).

**SOLVE**

Overlay SOLVE sets up and solves the synthesis eigenvalue problem. In a series of calls to subroutine FILLUP the code determines the order in which integrals are going to be needed during the sweep through the axial mesh (see section II-D for the solution algorithm). Whenever possible, integrals required to set up equations at one mesh interval are saved and reused at the next. The required integrals from VOLINT and DIFINT are rewritten to a scratch file INTGLS in the order in which they are to be needed. Data management decisions required to set up the integral rewriting and later solution are made in subroutine DIVVY. The integrals are rewritten in WINTGL.

The inhomogeneous problem solution which is the basis of the eigenvalue calculation (see Section II-D) is carried out in subroutine SPROB. SPROB sets up the synthesis equations with the aid of subroutine FILLUP, now operating in a mode in which it retrieves integrals from the INTGLS file, and subroutine SHUFFL, which rearranges elements of the equation matrices in order to reuse integrals during the sweep of the axial mesh. During the forward-elimination part of the sweep the H matrices (see Eqs. 42 and 43) are stored in the file HFILE.

After the eigenvalue iterations subroutine OUTPRO edits the combining coefficients and writes the combining coefficient files, DCCOEF and ACCOEF.

**EDITS**

The EDITS overlay handles all the optional output edits and files.

---

**C. BPOINTER, A Dynamic Storage Allocation Subprogram Package**

BPOINTER is a general, FORTRAN subprogram package which was developed to alleviate bookkeeping chores associated with the use of dynamic storage allocation techniques. Programs which use BPOINTER tend to be structured in subroutine form. A control routine is used to define one or two large blocks of storage (called the container array) and to make the appropriate calls to BPOINTER to control the allocation of storage within the block(s). Calls to calculational subroutines transmit pointers corresponding to appropriate array locations through the calling sequences. All BPOINTER capabilities are accessed through an appropriate call to an entry point, subroutine or function subprogram. The following capabilities are available in the BPOINTER system:
(a) Storage of data in and retrieval of data from the container array, via user defined variable arrays.

(b) Purge of variable arrays stored in the container array.

(c) Automatic "clean up" of the container array when more storage is required.

(d) Re-definition of array sizes without loss of data already stored in the array.

(e) Array dump of selected integer, floating point or BCD arrays in a prescribed format.

(f) Trace dumps of BPOINTER activities.

(g) Status report of the BPOINTER tables.

Detailed program documentation including flow charts, common block information and subprogram descriptions is available in Ref. 25. This section is intended to provide a brief description of how the program package operates. The major differences between the IBM and CDC stand-alone versions of the program package are also noted.

The short example listed in Fig. IV-2 is intended to illustrate the structure of a program using the BPOINTER package. This example shows the manner in which a container is allocated, pointers defined and used, and the container released.

Brief descriptions of all the BPOINTER entry points, subroutines and functions are given in Table IV-2.

All dynamically allocated arrays are addressed relative to the common block /ARRAY/ which contains a single array element, BLK(1). In the IBM version of the code the element must be declared DOUBLE PRECISION. In some versions of the CDC BPOINTER (not the version accompanying the Code Center SYN3D, which does not use LCM) a second common block /ARRAY2/ is used to address arrays allocated to a large core memory container. In versions of BPOINTER which use LCM this common block also contains a single array element, BLKECS(1), which must be declared a LEVEL 2 variable. In the SYN3D BPOINTER package BLKECS appears in the coding but is equivalenced with BLK. The equivalent of the large core memory container on IBM equipment is a second container which may be given a HIARCHY 1 location but is addressed in precisely the same manner as the first (SCM) container. The one word assigned to the container by the source language program provides a reference address. At execution time machine language routines (ALLOC1, ALLOC2 on IBM, MEMGET1, MEMGET2 on CDC) are used to obtain the addresses of core which are available to the program for the allocation of data arrays. These blocks of core are allocated in the following manner:
FIGURE IV-2. EXAMPLE OF BPOINTER USE.

C BPOINTER EXAMPLE
C DEFINE CONTAINER COMMON BLOCK
C
REAL*8 BLK, FLUX, POWER
COMMON/ARRAY/BLK(1)
DIMENSION FLK4(1)
EQUIVALENCE (BLK(1), BLK4(1))
DATA FLUX/1.0, POWER/6.0, MAX/SIZE/10000/
DATA TQ/1.0, TV/1.0, IO/0., NS/27/
C
C ALLOCATE CONTAINER WITH MAXSIZE WORDS OF SIZE AND NO LCM
C
CALL RBLK(IO)
CALL PONTN(BLK, MAXSIZE, IO)
C
C ALLOCATE SPACE FOR ARRAYS POWER, FLUX AND CURRENT
C
CALL RUTH(POWER, IO, NG, IPOWFR)
CALL RUTH(FLUX, IO, 2*NG, IFUX)
C
DETERMINE POINTER FOR SUB-ARRAY CURRENT WHICH FOLLOWS THE
C NG SINGLE PRECISION WORDS FOR THE ARRAY FLUX
C
C NENT=INT2(IFUX, NG, IO)
C
C CHECK ON BPOINTER ERROR
C
IF ( IPEPTN(DUM), GT, 0 ) PRINT 500
500 FORMAT(1HO, 10HBPOINTER ERROR)
C
C CALL SUBROUTINE INIT TO USE THESE ARRAYS
C
CALL INITN(BLK(FLUX), BLK(IPower), BLK4(ICURRENT), NG)
C
C FREE CONTAINER AND RETURN
C
CALL FREE
RETURN
END
SUBROUTINE INITN(PHI, POWER, CURRENT, NG)
C
C USE BPOINTER ARRAYS JUST AS ANY OTHER VARIABLES
C
REAL*8 POWER
DIMENSION PHI(1), POWER(1), CURRENT(1)
DO 10 I=1, NG
PHI(I) = 1.0
POWER(I) = 3.18*06
CURRENT(I) = 333
10 CONTINUE
RETURN
END
<table>
<thead>
<tr>
<th>Subprogram</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>POINTR</td>
<td>Initializes tables of dynamic allocation program package and calls ALLOC1 and ALLOC2 to allocate container(s) for variably dimensioned arrays.</td>
</tr>
<tr>
<td>PUTPNT/PUTBLK</td>
<td>Dummy routine calls PUTM to allocate array storage.</td>
</tr>
<tr>
<td>BULK</td>
<td>Sets number of words of BULK(LCM) core to be allocated.</td>
</tr>
<tr>
<td>FREE</td>
<td>Calls FREE1 and FREE2 to release container allocated by calls from subroutine POINTR.</td>
</tr>
<tr>
<td>WIPOUT/CLEAR</td>
<td>Deletes a named array from BPOINTER tables; zeroes all locations assigned to a named array.</td>
</tr>
<tr>
<td>GETPNT/GETN/DUMP</td>
<td>Returns pointer for a named array; returns index in BPOINTER tables of a named array, controls printing of a named array.</td>
</tr>
<tr>
<td>IGET</td>
<td>Returns pointer for a named array.</td>
</tr>
<tr>
<td>IPT2</td>
<td>Returns pointer to a sub-array relative to a single precision word length container.</td>
</tr>
<tr>
<td>PUTM/PUTB</td>
<td>Enters named arrays into fast and BULK(LCM) containers respectively.</td>
</tr>
<tr>
<td>IPTERR/NNAMSF</td>
<td>Returns number of BPOINTER errors, returns number of named arrays in BPOINTER tables.</td>
</tr>
<tr>
<td>ILAST/ILASTB</td>
<td>Returns word number of first available word in SCM/LCM container.</td>
</tr>
<tr>
<td>REDEF</td>
<td>Dummy routine calls REDEFM to redefine size and/or location of named array.</td>
</tr>
<tr>
<td>REDEFM/REDEFB</td>
<td>Redefine the size and/or location of named array within BPOINTER tables and containers.</td>
</tr>
<tr>
<td>PURGE/PURGEB</td>
<td>Sift storage in SCM/LCM containers to eliminate unused blocks created by WIPOUT calls.</td>
</tr>
<tr>
<td>STATUS</td>
<td>Edits status of BPOINTER tables.</td>
</tr>
<tr>
<td>PRTI1</td>
<td>Prints half word integer array from SCM container.</td>
</tr>
<tr>
<td>PRTI1E</td>
<td>Prints half word integer array from LCM container.</td>
</tr>
<tr>
<td>PRTI2</td>
<td>Prints full word integer array from SCM container.</td>
</tr>
<tr>
<td>PRTI2E</td>
<td>Prints full word integer array from LCM container.</td>
</tr>
<tr>
<td>PRTR1/PRTA1</td>
<td>Prints full word real array from SCM container.</td>
</tr>
<tr>
<td>PRTR1E/PRTA1E</td>
<td>Prints full word real array from LCM container.</td>
</tr>
<tr>
<td>PRTR2/PRTA2</td>
<td>Prints double word real array from SCM container.</td>
</tr>
<tr>
<td>PRTR2E/PRTA2E</td>
<td>Prints double word real array from LCM container.</td>
</tr>
</tbody>
</table>
IBM Allocation

The standard IBM macro instructions GETMAIN and FREEMAIN are used to allocate and free consecutive words of core which are available to the program. The designations subpool 1 and 2 are assigned to the bulk (LCM) and fast (SCM) containers, respectively. Since allocations are performed in units of 256 (eight byte) words, it is most efficient to request blocks of core in such multiples.

CDC Allocation

The COMPASS routine MEMGET uses the standard CDC macro instruction MEMORY to determine the job's SCM and LCM field lengths. The top of the user's SCM field length is used for the BPOINTER SCM container. The user is responsible for providing enough SCM memory to accommodate both the program and the BPOINTER container; there is currently no effective check to make sure that data stored in the BPOINTER container does not overlap code. It is assumed by the CDC version of BPOINTER that the level 2 common block /ARRAY2/ is addressed as the first word of LCM and the entire LCM field length is assumed to be available to BPOINTER for its LCM container.

The letters M and B are used as neumonics within BPOINTER to designate routines which operate on the SCM and LCM containers, respectively. Thus PUTM allocates an array in the SCM container while PUTB (which is not used in SYN3D) allocates an array which must be referenced on CDC equipment as a LEVEL 2 array. On IBM equipment without HIARCHY support (e.g. 370/195) the two containers are equivalent. The distinctions noted above between the two dynamic containers are important on CDC equipment where the containers are addressed quite differently and on IBM equipment with HIARCHY support where access to the BULK container (HIARCHY 1, subpool 1) is significantly slower than access to the MAIN core container (HIARCHY 0, subpool 2).

V. THE ARGONNE CODE CENTER VERSIONS OF SYN3D

SYN3D is available on magnetic tape through the Argonne Code Center for both IBM and CDC machines. This section describes the contents of the tapes and outlines the steps necessary to implement the code in stand-alone form (without understanding very much about synthesis or the code itself). The Code Center package includes several test problems; this section also contains descriptions and the solutions of these test problems.

A. The SYN3D Package

The Code Center SYN3D package consists of this memo and a magnetic tape containing four BCD files. There are separate tapes for IBM and CDC versions; Table V-1 describes the format of each tape and the contents and length of each BCD file.
TABLE V-1. Description of SYN3D Tapes and the BCD Files they Contain

<table>
<thead>
<tr>
<th></th>
<th>IBM Tape</th>
<th>CDC Tape</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>9 trk.</td>
<td>7 trk.</td>
</tr>
<tr>
<td>Density</td>
<td>800 bpi</td>
<td>556 bpi</td>
</tr>
<tr>
<td>Character Code</td>
<td>EBCDIC (029 keypunch)</td>
<td>BCD (026 keypunch)</td>
</tr>
</tbody>
</table>

In both cases there are no internal labels, there are 80 characters per card image and the blocking is 3200 (Forty cards per block).

<table>
<thead>
<tr>
<th>File Number</th>
<th>Contents</th>
<th>Number of Card Images</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>IBM Tape</td>
</tr>
<tr>
<td>1</td>
<td>SYN3D FORTRAN source code</td>
<td>19163</td>
</tr>
<tr>
<td>2</td>
<td>Additional BPOINTER routines</td>
<td>501</td>
</tr>
<tr>
<td>3</td>
<td>CDFILE FORTRAN source code</td>
<td>665</td>
</tr>
<tr>
<td>5</td>
<td>Interface files in CDFILE format</td>
<td>1418</td>
</tr>
</tbody>
</table>

TABLE V-2. SYN3D Segments

<table>
<thead>
<tr>
<th>Overlay Name</th>
<th>Card Numbers</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAIN</td>
<td>1-2154</td>
</tr>
<tr>
<td>CARDS</td>
<td>2155-2971</td>
</tr>
<tr>
<td>HMG4C</td>
<td>2972-6278</td>
</tr>
<tr>
<td>INTEG</td>
<td>6279-12981</td>
</tr>
<tr>
<td>SOLVE</td>
<td>12982-16363</td>
</tr>
<tr>
<td>EDITS</td>
<td>16364-19163</td>
</tr>
</tbody>
</table>
SYN3D

The first two files combine to form the SYN3D code. The first file is SYN3D proper. The IBM and CDC versions are derived from the same source code. Statements that are unique to IBM computers (e.g. REAL*8) are surrounded by pairs of comment cards starting with the characters "CIBM" in columns 1-4 and are commented out in the CDC version. Statements that are unique to CDC computers (e.g. OVERLAY) are surrounded by "CDC*" comment cards and are commented out in the IBM version.

The card number (columns 73-80) of a particular FORTRAN instruction in the IBM version is the same in the CDC version. This was done to avoid the nuisance of maintaining entirely separate versions of the code. Future corrections to the code will be specified in terms of this numbering system.

ADDITIONAL BPOINTER ROUTINES

BPOINTER is a set of dynamic storage allocation subroutines used in SYN3D to manage fast memory. The bulk of the coding is in FORTRAN, is common to both IBM and CDC versions of SYN3D, and is included in the SYN3D source file (cards 902-2154).

A few of the BPOINTER routines, however, are machine dependent, and these are included in file 2 of the Code Center tape. For the IBM version these additional routines are all in assembler language. For the CDC version they are partly FORTRAN (the first 108 cards of file 2) and partly COMPASS (the last 22 cards).

The additional BPOINTER routines can be assembled (and for the CDC version compiled) separately and included in the main overlay of SYN3D (cards 1-2154 of file 1) at load time.

CDFILE

CDFILE (file 3) is a short, stand-alone, FORTRAN code which provides a crude way of generating CCCC binary interface files from BCD card input. File 4 contains the CDFILE BCD input file needed to produce binary interface files for a number of sample problems. If the user has other means of writing CCCC interface files he/she may choose to ignore files 3 and 4.

B. Code Structure

The discussion of the structure of SYN3D included here will just cover those aspects which affect the linking and execution of the code in a straightforward, stand-alone form. The functions of the several overlays of the program are covered in more detail in Section IV.

The simplest overlay scheme (other than none at all) is a division into a main overlay and five primary overlays. The names of the overlays and the corresponding card numbers in the source coding (file 1 of the Code Center tape), are given in Table V-2. The additional BPOINTER routines (file 2) should be included in the main overlay.
The CDC version of SYN3D includes OVERLAY cards at the beginning of each of the segments listed in Table V-2 and, therefore, can be compiled directly from tape to a load file. For those compilers which permit a mix of FORTRAN and assembler language it may be convenient to splice the additional BPOINTER routines (file 2 on the Code Center tape) into the SYN3D source file after card 2154. Alternatively, one can compile SYN3D and the additional BPOINTER routines separately and merge the two either by inserting the additional BPOINTER relocatable object code at the proper place in the SYN3D load file or, for loaders that permit it, by providing the additional routines through a user library.

The IBM version of SYN3D has been run at Argonne by compiling the segments listed in Table V-2 and assembling the additional BPOINTER routines to a partitioned data set and then overlaying at load time.

C. File Number Assignments

All binary files (input and output interface files and scratch files) used by SYN3D are handled through the CCCC standard subroutines SEEK, REED and RITE. The assignment of file numbers to file names and the initialization of the SEEK tables is done from the SYN3D driver routine (cards 1-170 of the source code).

The file assignments in the Code Center version of SYN3D are listed in Table V-3. Multiple versions of some files are required. Printer output is written to file 6; BCD input (if required) is read from file 5. File assignments can be easily rearranged by changing the coding in the main driver; no other routines need be modified.

All files are written sequentially. All but one file (HFILE) is read sequentially. At different stages of the calculation HFILE is read both forwards and backwards. SYN3D does not require direct access files.

D. Running the Code Center SYN3D as a Stand-alone Program

The Code Center version of SYN3D is set up to execute in environments that may never have heard of the CCCC standards. Most of the input data are contained in CCCC standard interface binary files, and a small program, CDFILE, is provided (file 4 of the Code Center tape) to convert input from BCD cards or a card image file to the necessary binary files. Those installations which already can write CCCC files can ignore CDFILE. A small amount of data specifying input binary file numbers and additional, code-dependent data are read from cards. Even these BCD data can be eliminated if desired (see section V-F).

CDFILE

This utility program was written largely as a crude input processor for the Code Center version of SYN3D. It only processes the six CCCC files RTFLUX, ATFLUX, GEODEST, ISOTXS, NDXSRF and ZMATIN.

CDFILE operates in two modes. In the first (MODE=1) it will read a number of CCCC binary files (those types listed above) and write the data in BCD card image form to a single BCD file. In the second (MODE=2) it
TABLE V-3. Files Required by the Code Center Version of SYN3D
(GEODST through ZNATIN are CCCC files. The rest may be treated as scratch files.)

<table>
<thead>
<tr>
<th>File Name</th>
<th>File Version Nos.</th>
<th>Use</th>
<th>Logical Unit No.*</th>
</tr>
</thead>
<tbody>
<tr>
<td>GEODST</td>
<td>1-10</td>
<td>These are the several two-dimensional GEODST files required to describe a three-dimensional model. GEODST,1 can be a three-dimensional file, in which case the code will write two-dimensional files GEODST,2...GEODST,N.</td>
<td>11-21</td>
</tr>
<tr>
<td>RTFLUX</td>
<td>1-11</td>
<td>RTFLUX files containing expansion and/or weighting functions.</td>
<td>21-31</td>
</tr>
<tr>
<td>ATFLUX</td>
<td>1-5</td>
<td>ATFLUX files containing expansion and/or weighting functions.</td>
<td>32-36</td>
</tr>
<tr>
<td>PWDINT</td>
<td>1</td>
<td>Output power density, by mesh volume.</td>
<td>37</td>
</tr>
<tr>
<td>RZFUX</td>
<td>1</td>
<td>Zone average fluxes.</td>
<td>38</td>
</tr>
<tr>
<td>ISOTXS</td>
<td>1</td>
<td>Input microscopic cross sections.</td>
<td>39</td>
</tr>
<tr>
<td>NDXSRF</td>
<td>1</td>
<td>Cross-section reference.</td>
<td>40</td>
</tr>
<tr>
<td>ZNATIN</td>
<td>1</td>
<td>Zone atom densities.</td>
<td>41</td>
</tr>
<tr>
<td>INTTOC</td>
<td>1</td>
<td>Input synthesis integral table of contents (optional).</td>
<td>42</td>
</tr>
<tr>
<td>INTTOC</td>
<td>2</td>
<td>Output synthesis integral table of contents.</td>
<td>43</td>
</tr>
<tr>
<td>VOLINT</td>
<td>1</td>
<td>Input synthesis VOLINT integrals (optional).</td>
<td>44</td>
</tr>
<tr>
<td>VOLINT</td>
<td>2</td>
<td>Output synthesis VOLINT integrals.</td>
<td>45</td>
</tr>
<tr>
<td>DIFINT</td>
<td>1</td>
<td>Input synthesis DIFINT integrals (optional).</td>
<td>46</td>
</tr>
<tr>
<td>DIFINT</td>
<td>2</td>
<td>Output synthesis DIFINT integrals.</td>
<td>47</td>
</tr>
<tr>
<td>DCOEF</td>
<td>1</td>
<td>Direct synthesis combining coefficients.</td>
<td>48</td>
</tr>
<tr>
<td>ACOEF</td>
<td>1</td>
<td>Adjoint synthesis combining coefficients.</td>
<td>49</td>
</tr>
<tr>
<td>COMPXS</td>
<td>1</td>
<td>Macroscopic cross sections.</td>
<td>50</td>
</tr>
</tbody>
</table>
TABLE V-3. Contd.

<table>
<thead>
<tr>
<th>File Name</th>
<th>File Version Nos.</th>
<th>Use</th>
<th>Logical Unit No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>REQFLX</td>
<td>1</td>
<td>Rewritten fluxes.</td>
<td>51</td>
</tr>
<tr>
<td>REQXST</td>
<td>1</td>
<td>Rewritten macroscopic cross sections.</td>
<td>52</td>
</tr>
<tr>
<td>INTGLS</td>
<td>1</td>
<td>Rewritten synthesis integrals.</td>
<td>53</td>
</tr>
<tr>
<td>HFILE</td>
<td>1</td>
<td>Scratch file used during solution. (Also used by HMG4C if multipass mode is used.)</td>
<td>54</td>
</tr>
<tr>
<td>SYNCNO</td>
<td>1</td>
<td>The input data in binary form.</td>
<td>55</td>
</tr>
<tr>
<td>SYNFILE</td>
<td>1</td>
<td>The input data in BCD card images (used with ARC System input routines).</td>
<td>56</td>
</tr>
</tbody>
</table>

*These are the file number assignments made in the main overlay of the Code Center version of SYN3D by the initialization call to SEEK. Since the rest of the code always uses SEEK to determine file numbers, it is a relatively simple job to assign entirely different file numbers by defining new SEEK tables.*
will read BCD card images from a single BCD file and write a number of binary files. File 4 of the Code Center tape is a BCD file written by CDFILE in the first mode and represents the binary files necessary for several test calculations. Section V-E describes these test calculations run by first executing CDFILE with MODE=2 and file 4 of the Code Center tape as input and then executing SYN3D with the resulting binary files as input. Table V-4 describes the contents of the two input cards, and Table V-5 shows examples of the modes in which CDFILE can be run.

CDFILE can process any number of files. The BCD file (LUNBCD) contains blocks (decks) of card images stacked in the order indicated by (LUN(I),I=1,NFILES). There is no form of separator between blocks.

Installations with any sort of a network of reactor analysis codes can find a more efficient procedure for generating CCCC binary interface files than the CDFILE program. We will not encourage the general use of CDFILE by listing in this report the card image formats for those CCCC files the code can handle. Users who have no other way to generate interface files are referred to the CCCC standard file descriptions (see Appendix I) and to the FORTRAN source coding of CDFILE. It suffices to say that the program sweeps sequentially through a binary file reading and writing integer and floating point data as they are encountered.

It is suggested that CDFILE and the test problems supplied in the Code Center SYN3D package be used as a preliminary check of the code when it is initially implemented at any installation. Examples of the use of CDFILE on the ANL IBM-370 and the Lawrence Berkeley Laboratory CDC 7600 are included later in this section.

**SEEK Table Initialization**

SYN3D determines file numbers for all files (except card input and printer output) by calls to the CCCC standard subroutine SEEK. Among its several functions SEEK returns an "existence flag" which specifies whether or not a file "exists" (i.e. has been previously written). Files can exist because they are input to the code or because SYN3D writes them itself. There must be some mechanism, therefore, to flag an input file as "existing" in the tables kept by SEEK.

The Code Center version of SYN3D does this with a single input card, read by the main driver (at card number 134 in the SYN3D source code). Each column of this card represents a logical unit number, and a non-zero integer in a particular column means that the corresponding file is an existing, input file. SYN3D then sets the existence flag for that file through a call to SEEK with an operation code of 1. This SEEK initialization card is the first card read by SEEK.

**The SYN3D BCD Control Data**

Obviously, there is input required by SYN3D that cannot be accommodated by the CCCC standard interface files alone. The Code Center version of SYN3D is set up to read a certain amount of data from cards placed after the SEEK initialization card.
TABLE V-4. CDFILE Input Data

<table>
<thead>
<tr>
<th>Card</th>
<th>Contents</th>
<th>Format</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>NFILES, LUNBCD, MODE</td>
<td>313</td>
</tr>
<tr>
<td>2</td>
<td>(LUN(I), I=1,NFILES)</td>
<td>2413</td>
</tr>
</tbody>
</table>

NFILFS = The number of CCCC binary interface files to be processed.

LUNBCD = The logical unit number of the BCD file to which BCD card images are to be written (MODE=1) or from which BCD card images are to be read (MODE=2).

MODE = 1, data from binary files is written to the BCD card image file (file number LUNBCD). 2, data from the BCD card image file is written to binary files.

LUN(I) = The logical unit number of the CCCC binary interface file from which (MODE=1) or to which (MODE=2) the Ith block of data on the BCD file number LUNBCD is to be transferred.

TABLE V-5. Examples of CDFILE Modes of Operation. These Examples Assume File 5 is Card Input, File 6 is Printer Output and File 7 is Punch Output

<table>
<thead>
<tr>
<th>Mode</th>
<th>LUNBCD</th>
<th>CDFILE Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>Reads binary files and prints contents.</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>Reads binary files and punches contents.</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>Reads cards punched by a previous CDFILE run (MODE=1, LUNBCD=7) and writes binary files. The card deck should be placed immediately after the two cards described in Table V-4.</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>Reads card images from the file on logical unit 10 (e.g. the fourth file on the Code Center tape) and writes binary files.</td>
</tr>
</tbody>
</table>
These data are provided on cards which have numbers in columns 1-2 which identify a particular card type. In many cases several cards of the same type may be required. The cards must be arranged in order of ascending card type number. Immediately after the SEEK initialization card, and before the first numbered card, a card must be provided which defines (with format 2413) the number of cards included for each card type. The number of type 1 cards is punched in columns 1-3, the number of type 2 cards in columns 4-6, etc.

The contents and formats of the numbered cards are given in the file description of the BCD file SYNFIL (see Appendix D). The user is also referred to the sample problem input shown later in this section.

Examples of Input Decks with Control Cards

Figures V-1 and V-2 show input decks set up to execute CDFILE and SYN3D. Figure V-1 is a job run on Argonne's IBM 370/195 which link edits SYN3D from a partitioned data set (C116.B21006.EXPORT.SEVERLIB) containing the segments listed in Table V-2 and BPOINTER. The member BPOINTER contains the additional BPOINTER routines. Figure V-2 is a job run on Berkeley's CDC 7600 which loads SYN3D from a load file (PROG) compiled by FTN4.5.

BPOINTER

Storage for almost all the dimensioned arrays used in SYN3D is managed through the dynamic storage allocation routine BPOINTER. At run time BPOINTER reserves a region of memory, called the "container", the length of which is specified in the input (card type 2). On the IBM 370/195 BPOINTER actually requests space from the system, and if the REGION size is too small to hold both the program and the container an error message is printed. On CDC systems error messages may not occur when there is insufficient field length for the container until the code tries to store data outside the designated field length for the job.

The version of BPOINTER included in the CDC package can be used to allocate storage only in SCM; at present SYN3D only uses SCM.

E. Sample Problems

Model Descriptions

The test calculations included in the Code Center SYN3D package are based on the three-dimensional, simplified LMFBR model shown in Fig. V-3. A set of three-energy group cross sections is supplied, as are geometry and expansion function files required for several SYN3D calculations. The model includes rods which are banked at the midplane. The cross section data files contain microscopic cross sections for seven isotopes and number densities for seven compositions. Table V-6 defines the compositions.

File 4 of the Code Center Tape

The card images supplied in file 4 of the Code Center tape represent the CDFILE input "deck" required to generate 17 separate binary interface files. A description of each file is given in Table V-7. The "sequence
FIGURE V-1. INPUT DECK WITH CONTROL CARDS FOR THE IBM 370/195. EXECUTE CFPILE TO GENERATE INTERFACE FILES, LINK EDIT SYN3D AND EXECUTE SAMPLE PROBLEM 4. THE SOURCE CODE FOR CFPILE IS IN THE FILE CDFILE AND THE CFFILE INPUT IS IN THE FILE DATA. INTERFACE FILES NOT NEEDED IN THIS JOB ARE COPIED TO THE FILE DUMP.

// EXEC PTHCLG
// PTH.SYSTN DD DSN=CDFILE,DISP=(OLD,PASS),UNIT=SASCR
// GO.PT09P001 DD DSN=EDUMP,DISP=(NEW,PASS),UNIT=SASCR
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT10P001 DD DSN=DATA,DISP=(OLD,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.ZT22P001 DD DSN=ATFLUX1,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.ZT23P001 DD DSN=ATFLUX2,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.ZT24P001 DD DSN=ATFLUX3,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.ZT25P001 DD DSN=ATFLUX4,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT21P001 DD DSN=ATFLUX1,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT22P001 DD DSN=ATFLUX2,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT23P001 DD DSN=ATFLUX3,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT24P001 DD DSN=ATFLUX4,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT25P001 DD DSN=ATFLUX1,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT26P001 DD DSN=ATFLUX2,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT27P001 DD DSN=ATFLUX3,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT28P001 DD DSN=ATFLUX4,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT29P001 DD DSN=ATFLUX1,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT30P001 DD DSN=ATFLUX2,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT31P001 DD DSN=ATFLUX3,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT32P001 DD DSN=ATFLUX4,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT33P001 DD DSN=ATFLUX1,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT34P001 DD DSN=ATFLUX2,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT35P001 DD DSN=ATFLUX3,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT36P001 DD DSN=ATFLUX4,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT37P001 DD DSN=ATFLUX1,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT38P001 DD DSN=ATFLUX2,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT39P001 DD DSN=ATFLUX3,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT40P001 DD DSN=ATFLUX4,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT41P001 DD DSN=ZMATTN1,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PT42P001 DD DSN=ZMATTN2,DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BKSIZE=6136)
// GO.PSNT DD *
17 19 ?
22 23 24 25 32 33 11 9 9 9 9 9 9 40 41 39
/*
// EXEC PTHSCLG, ENPOTS='OLY'
// EDT.SYSCLG DD DSN=MODLIB(SYN3D),DISP=(NEW,PASS),UNIT=SASCR,
// SPACE=(TRK,(40,5,1),RLSE),DCB=BLKSIZE=6144
// EDT.SEGLIB DD DSN=C116.R21006.EXPORT.SEGLIB,DISP=(NEW,KEEP)
// PDS.SYSTN DD *
ENTRY MAIN
INCLUDE SEGLIB(MAIN,ASSEM)
OVERLAY LEVEL1
INCLUDE SPGITY(CARDS)
OVERLAY LEVEL1
INCLUDE SEGLIB(HM4C)
OVERLAY LEVEL1
INCLUDE SEGLIB(INTV)
OVERLAY LEVEL1
INCLUDE SEGLIB(SOLV)
OVERLAY LEVEL1
INCLUDE SEGLIB(EDTS)
*/
Figure V-1. Input deck with control cards for the IBM 370/195. (contd.)

// PXRC PGM=SYN3D
//STEPIB DD DSN=MODLIB, DISP=(OLD, PASS), UNIT=SASCR
//FTP05P001 DD DNAME=SYSIN
//FTP06F001 DD SYSOUT=A
//FTP11P001 DD DSN=COBOL1, DISP=(OLD, PASS), UNIT=SASCR
//FTP12P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP13P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP14P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP15P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP16P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP17P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP18P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP19P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP20P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP21P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP22P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP23P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP24P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP25P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP26P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP27P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP28P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP29P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP30P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP31P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP32P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP33P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP34P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP35P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP36P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP37P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP38P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP39P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
//FTP40P001 DD DISP=(NEW, DELETE), UNIT=SASCR, SPACE=(TRK, (1,1)),
// DCB=(RECFM=VPS, LRECL=X, BLKSIZE=6136)
**Figure V-1. Input Deck with Control Cards for the IBM 370/195. (Contd.)**

//PT4P001 DD DSN=6ACC2EP, DISP=(NEW,DELETE), UNIT=SASCR,
// SPACE=(CYL, (1, 1)), DCB=(RECFM=VBS, LRECL=X, BLKSIZE=6136)
//PT5P001 DD DSN=6COM2PS, DISP=(NEW,DELETE), UNIT=SASCR,
// SPACE=(CYL, (1, 1)), DCB=(RECFM=VBS, LRECL=X, BLKSIZE=6136)
//PT51F001 DD DSN=6CP2OLX, DISP=(NEW,DELETE), UNIT=SASCR,
// SPACE=(CYL, (2, 1)), DCR=(RECFM=VBS, LRECL=X, BLKSIZE=13030)
//PT52P001 DD DSN=6RS2YST, DISP=(NEW,DELETE), UNIT=SASCR,
// SPACE=(CYL, (7, 1)), DCR=(RECFM=VBS, LRECL=X, BLKSIZE=13030)
//PT53P001 DD DSN=6INT2LS, DISP=(NEW,DELETE), UNIT=SASCR,
// SPACE=(CYL, (2, 1)), DCR=(RECFM=VBS, LRECL=X, BLKSIZE=13030)
//PT54P001 DD DSN=64P2LE, DISP=(NEW,DELETE), UNIT=SASCR,
// SPACE=(CYL, (7, 1)), DCR=(RECFM=VBS, LRECL=X, BLKSIZE=13030)
//PT55P001 DD DSN=6SY2CON, DISP=(NEW,DELETE), UNIT=SASCR,
// SPACE=(RECFM=VBS, LRECL=X, BLKSIZE=304), SPACE=(TRK, (1, 1))
//SYSTM NO * 1111 11 11 11

4 1 1 3 4 0 0 0 0 0 0 0 1 1 1 1 1 1 1

01 GROUP 3-DIMENSIONAL MODEL, NTXPD PLUX AND ADJOINT WEIGHTING
01 BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
01 GROUP 3 OF BLANKET FUNCTIONS NOT LSFD AT ALL
01 INPUT TO GEOST FLP
02 1 2 1 0 20 3 3 1
03 1 1 10.0
04 21 1 3 2.13
06 24 1 3 2.13
07 RTPLX 1 0.0 200.0
07 RTPLX 2 0.0 200.0
07 RTPLX 3 0.0 100.0
07 RTPLX 4 0.0 100.0
08 RTPLX 1 0.0 200.0
08 RTPLX 2 0.0 200.0
08 RTPLX 3 0.0 100.0
08 RTPLX 4 0.0 100.0
11 RTPLX 1 3 3
11 RTPLX 4 3 3
15 1 1 1 1 1 1 1
17 1 2 3
18 1 2 3
19 1 5 1 1 1 1 2 8 3 8
20 1 5 1 1 1 1 2 8 3 8
21 1 6 4 12 15 18
Figure V-2. Input deck with control cards for the CDC 7600. Execute CDFTLP to generate interface files. Load SYN3D and execute sample problem 4. The relocatable object code for CDFILE is in the file "CDFILE" and the CDFILE input is in the file "DATA". The file "PROG" contains the relocatable object code for SYN3D. Interface files not needed in this job are copied to the file "TAPE9".

COPYB7( DATA, TAPE10, 1 )
REVINH( DATA, TAPE10 )
RE, 150000, 500000 .
LINK ( X, F=CDFILE, P=FTN4LIR )
RE=VNH( TAPE10 )
LINK ( X, F=PROG, P=FTN4LTB, PL=150000 )

789 CARD

17 11 2
22 14 25 32 33 11 9 9 9 9 9 9 9 40 41 39

789 CARD

1 1111 11 111
4 1 1 0 2 4 4 0 0 0 0 2 0 0 1 1 1 1 1 1
01 3 GROUP, 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJOINT WEIGHTING
01 BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
01 GROUP 3 OF BLANKET FUNCTIONS NOT USED AT ALL
01 INPUT 3D GRID FILE
02 15 2 20 3 3 3 1
03 1 1.00 10.0
06 1L 1 3 2.13
06 1M 1 3 2.13
07 DFLUX 1 0.0 200.0
07 DFLUX 2 0.0 200.0
07 DFLUX 3 0.0 100.0
07 DFLUX 4 100.0 200.0
08 DFLUX 1 0.0 200.0
09 DFLUX 2 0.0 200.0
08 DFLUX 3 0.0 100.0
09 DFLUX 4 100.0 200.0
11 DFLUX 3 3 3
12 DFLUX 4 3 3
16 5 1 1 1 1
17 1 2 3
18 1 2 3
19 1 5 1 1 1 1 1 2 9 3 8
20 1 5 1 1 1 1 2 8 3 8
21 1 6 9 12 15 18

6789 CARD
Fig. V-3. The Geometry of the 3D Test Model
ANL Neg. No. 116-75-171.
TABLE V-6. Compositions Used in Test Problems

<table>
<thead>
<tr>
<th>Isotope Number</th>
<th>Isotope Name</th>
<th>Composition</th>
<th>Description</th>
<th>(^{239}\text{Pu})</th>
<th>(^{238}\text{U})</th>
<th>(^{23}\text{Na})</th>
<th>Fe</th>
<th>(^{16}\text{O})</th>
<th>(^{10}\text{B})</th>
<th>(^{12}\text{C})</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>inner core</td>
<td>.0011</td>
<td>.0064</td>
<td>.0104</td>
<td>.0181</td>
<td>.0149</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>outer core</td>
<td>.0015</td>
<td>.0054</td>
<td>.0110</td>
<td>.0181</td>
<td>.0138</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>axial blanket</td>
<td>.0080</td>
<td>.0088</td>
<td>.0244</td>
<td>.0160</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>radial blanket</td>
<td>.0145</td>
<td>.0066</td>
<td>.0173</td>
<td>.0290</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>reflector</td>
<td>.0044</td>
<td>.0691</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>control rod</td>
<td>.0104</td>
<td>.0181</td>
<td>.0149</td>
<td>.0090</td>
<td>.0412</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>rod channel</td>
<td>.0220</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
TABLE V-7. Description of the CDFILE BCD Data Blocks in File 4 of the Code Center Tape

<table>
<thead>
<tr>
<th>Sequence Number</th>
<th>Interface File Name</th>
<th>File Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RTFLUX</td>
<td>A 2D flux solution in the unrodded core plane of the model. A buckled (transverse distance = 120 cm.) eigenvalue calculation (k = .99596).</td>
</tr>
<tr>
<td>2</td>
<td>RTFLUX</td>
<td>A 2D flux solution in the rodded core plane. A buckled (transverse height = 150 cm.) eigenvalue calculation (k = 1.00914).</td>
</tr>
<tr>
<td>3</td>
<td>RTFLUX</td>
<td>A 2D flux solution in the unrodded, axial blanket plane. An inhomogeneous calculation.</td>
</tr>
<tr>
<td>4</td>
<td>RTFLUX</td>
<td>A 2D flux solution in the rodded, axial blanket plane. An inhomogeneous calculation.</td>
</tr>
<tr>
<td>5</td>
<td>ATFLUX</td>
<td>A 2D adjoint solution in the unrodded core plane.</td>
</tr>
<tr>
<td>6</td>
<td>ATFLUX</td>
<td>A 2D adjoint solution in the rodded core plane.</td>
</tr>
<tr>
<td>7</td>
<td>GEODST</td>
<td>A 3D geometry description of the model.</td>
</tr>
<tr>
<td>8</td>
<td>GEODST</td>
<td>A 2D geometry description of the unrodded reflector plane.</td>
</tr>
<tr>
<td>9</td>
<td>GEODST</td>
<td>A 2D geometry description of the unrodded axial blanket plane.</td>
</tr>
<tr>
<td>10</td>
<td>GEODST</td>
<td>A 2D geometry description of the unrodded core plane.</td>
</tr>
<tr>
<td>11</td>
<td>GEODST</td>
<td>A 2D geometry description of the rodded core plane.</td>
</tr>
<tr>
<td>12</td>
<td>GEODST</td>
<td>A 2D geometry description of the rodded axial blanket plane.</td>
</tr>
<tr>
<td>13</td>
<td>GEODST</td>
<td>A 2D geometry description of the rodded reflector plane.</td>
</tr>
<tr>
<td>14</td>
<td>GEODST</td>
<td>A 2D geometry description of a 12 x 12 mesh, uniform plane with zero-current boundary conditions.</td>
</tr>
<tr>
<td>15</td>
<td>NDXSRF</td>
<td>Cross section reference data.</td>
</tr>
<tr>
<td>16</td>
<td>ZNATDN</td>
<td>Atom density data.</td>
</tr>
<tr>
<td>17</td>
<td>ISOTXS</td>
<td>Microscopic cross sections.</td>
</tr>
</tbody>
</table>
number" indicates the order in which the card image blocks for each binary file are arranged in the card image file.

**Sample Problem 1**

The first sample problem is a fundamental mode calculation for the infinite medium eigenvalue and spectrum for composition 1, the material in the inner core of the model (see Table V-6). One of the GEODST files (sequence number 14 in Table V-7) represents a homogeneous, reflected plane containing composition 1. The synthesis calculation uses this GEODST file, a flat expansion function (1.0 everywhere in space and energy) and zero-current axial boundary conditions.

This first sample problem requires four input interface files: the uniform plane GEODST mentioned above, NDXSRF, ZNATDN and ISOTXS. The GEODST file is used only in this calculation; the other three files define the compositions and cross sections and are used in all four sample problems. The SYN3D BCD input for the job is shown in Fig. V-4.

Appendix H-1 is the complete output (16 pages) from sample problem 1. In this one case the macroscopic cross section edits (card type 2) have been turned on to provide a printed record of their values. The infinite medium fundamental eigenvalue for composition 1 is 1.3017, and the eigenvector (from the synthesis combining coefficient edits) is

$$1.000, 1.400, 0.04585$$

**Sample Problem 2**

One of the RTFLUX files in file 5 of the Code Center tape (sequence number 1 in Table V-7) was generated from a buckled, 2D diffusion theory calculation in the unrodded core plane of the 3D model. The buckling corresponded to a transverse slab thickness of 120 cm., and the 2D eigenvalue was .99596. Sample problem 2 is a synthesis calculation of a 3D model which is axially uniform and is described in the plane by the 2D GEODST file defining the unrodded core plane (sequence number 10 in Table V-7). The model is 60 cm. high and has zero current and zero flux boundary conditions on the first and second z boundary planes, respectively.

The unrodded core RTFLUX file is provided as the only expansion function, and the flat (UNIT) weighting option is used. Since the expansion function is "perfect" for the problem, and the flat weighting reduces the synthesis equations to the balance satisfied by the original 2D calculation, SYN3D should reproduce the 2D eigenvalue and yield combining coefficients which are cosine shaped and equal in magnitude in all three groups. Figure V-5 lists the input interface files and shows the BCD input.

Appendix H-2 shows the complete output (6 pages) from this sample problem. The eigenvalue is .99600 and the combining coefficients are essentially equal in all groups. The output shown in Appendix H-2 comes from a job run on an IBM 370/195; the same job run on a CDC 7600 gave an eigenvalue of .99599. Small deviations are to be expected because of the finite axial mesh and the fact that on IBM machines the cross sections are stored in SYN3D in REAL*4 words.
FIGURE V-4. SYN3D BCD input and input interface files for the 1st sample problem. Each input file is identified by a sequence number from Table V-7 and is assigned a logical unit number consistent with Table V-3. The BCD data consists of the seek initialization card, a "cards-per-card-type" card, and a set of type-numbered cards.

INPUT INTERFACE FILES

| SYN3D LOGICAL UNIT NUMBER | 11 39 40 41 |
| FILE SEQUENCE NUMBER (TABLE V-7) | 14 17 15 16 |

BCD INPUT

```
 1 1 0 1 1 0 1
 01 THREE GROUP, FUNDAMENTAL MODE CALCULATION, COMPOSITION 1. 111
 02 5 1
 04 3 10.0
 05 1 0.0 10.0
 07 UNIT 0.0 10.0
```

FIGURE V-5. SYN3D BCD input and input interface files for the 2nd sample problem. Each input file is identified by a sequence number from Table V-7 and is assigned a logical unit number consistent with Table V-3. The BCD data consists of the seek initialization card, a "cards-per-card-type" card, and a set of type-numbered cards.

INPUT INTERFACE FILES

| SYN3D LOGICAL UNIT NUMBER | 11 22 39 40 41 |
| FILE SEQUENCE NUMBER (TABLE V-7) | 10 1 17 15 16 |

BCD INPUT

```
 2 1 1 1 1 0 1 1
 01 THREE GROUP, 2D BUCKLED PLANE (THE UNRODDED CORE PLANE FROM THE 111
 01 3D MODEL), HALF CORE SYMMETRY.
 02 5 1 20
 04 30 60.0
 05 1 0.0 60.0
 07 UNIT 0.0 60.0
 08 UNIT 0.0 60.0
```
While of little practical interest, this sort of calculation is useful as a debug tool and as a consistency check between SYN3D and whatever diffusion theory code is used to generate expansion functions. Significant discrepancies might indicate that the two codes are based on different finite-difference formulations.

Sample Problem 3

This is a calculation of the full, three-dimensional model shown in Fig. V-3. The approach used illustrates the way SYN3D has been used to calculate a variety of fast reactor models. The three-dimensional model is described in terms of the six, unique axial zones (reflector, blanket and core planes, each of which may be either rodded or unrodded). The six input GEODST files are sequence numbers 8-13 of Table V-7. Four expansion functions (sequence numbers 1-4) are provided. Two are buckled, eigenvalue calculations in the unrodded and rodded core planes. Two are from inhomogeneous calculations in the axial blankets in which the fixed source is the product of the expansion function associated with the adjacent core zone and the blanket diffusion coefficient distribution. Typically, the solution in the reflector is of little interest, and reflector expansion functions are not included.

Figure V-6 lists the input interface files and shows the BCD input. The core expansion functions are used everywhere, but each blanket function is used only over 40% of the height of the model. It is unlikely that a rodded (top) blanket function could contribute significantly to the unrodded (bottom) axial blanket. Economies of this sort can significantly reduce running times with little sacrifice of accuracy.

Appendix H-3 shows selected pages of the output for this job. The eigenvalue is .97506 (v. .97532 for a 3D finite-difference calculation). Samples of the flux, power density and zone average flux edits are included, as is a sample printer-plot of the contributions from each expansion function to the axial flux distribution (i.e. the 3D flux integrated over the x and y dimensions). Selective use of these plots can help identify poor choices of expansion functions and anomalous solutions.

Sample Problem 4

This last calculation is intended to illustrate some of the other options available in SYN3D. Figure V-7 lists the input interface files and shows the BCD input. The model geometry is the same as that of sample problem 3, but now there is a single, three-dimensional input GEODST file (sequence number 7 in Table V-7). At the top and bottom boundaries the boundary conditions are now logarithmic. The same four expansion functions used in problem 3 are used in problem 4, but each blanket function is used over half the height of the core, and the group 3 flux of each blanket function is not used at all. Two of the four weighting functions are now adjoint fluxes (sequence numbers 5 and 6 in Table V-7), and both direct and adjoint synthesis problems are solved. Sample problem 4 is intended as an illustration of some of the more exotic options in the code and should not be taken as a recommendation. Sample problem 3 is a more typical example of an application of SYN3D.
FIGURE V-6. SYN3D BCD INPUT AND INPUT INTERFACE FILES FOR THE 3rd SAMPLE PROBLEM. EACH INPUT FILE IS IDENTIFIED BY A SEQUENCE NUMBER FROM TABLE V-7 AND IS ASSIGNED A LOGICAL UNIT NUMBER CONSISTENT WITH TABLE V-3. THE BCD DATA CONSISTS OF THE SET initialization CARD, A "CARDS-PER-CARD-TYPE" CARD AND A SET OF TYPE-NUMBERED CARDS.

INPUT INTERFACE FILES

SYN3D LOGICAL UNIT NUMBER 11 12 13 14 15 16 22 23 24 25 39 40 41
FILE SEQUENCE NUMBER (TABLE V-7) 8 9 10 11 12 13 1 2 3 4 17 15 16

PCD INPUT

<table>
<thead>
<tr>
<th>SEQUENCE NUMBER</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>111111</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
</tr>
<tr>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>1</td>
<td>00.0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

GROUP 3-DIMENSIONAL MODEL, FLUX WEIGHTED SYNTHESIS
PLANET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
3D GEOMETRY BUILT UP FROM 2D GEOST FILE
Figure V-7. SYN3D BCD input and input interface files for the 4th sample problem. Each input file is identified by a sequence number from Table V-7 and is assigned a logical unit number consistent with Table V-3. The BCD data consists of the seek initialization card, a "cards-per-card-type" card and a set of type-numbered cards.

Input Interface Files

<table>
<thead>
<tr>
<th>SYN3D Logical Unit Number</th>
<th>11 22 23 24 25 32 33 39 40 41</th>
</tr>
</thead>
<tbody>
<tr>
<td>PVIP Sequence Number (Table V-7)</td>
<td>7 1 2 3 4 5 6 17 15 16</td>
</tr>
</tbody>
</table>

BCD Input

```
01 3 GROUP, 3-DIMENSIONAL MODEL
02 15 2 20
03 1.00 10.0
07 STFLUX 1 0.0 200.0
07 STFLUX 2 0.0 200.0
07 STFLUX 3 0.0 100.0
07 STFLUX 4 100.0 200.0
08 STFLUX 1 0.0 200.0
08 STFLUX 2 0.0 200.0
08 STFLUX 3 0.0 100.0
09 STFLUX 4 100.0 200.0
16 5 3 1 1
17 1 2 3
18 1 2 3
19 1 5 1 8 1 11 2 9 3 8
20 1 5 1 8 1 11 2 8 3 8
21 1 1 6 9 12 15 18 20
```
Appendix H-4 shows selected pages of output for this job. The eigenvalue is .97520 (vs. .97532 for a 3D finite-difference calculation). The input decks shown in Figs. V-1 and V-2 are set up to run this sample problem.

F. Suggested Modifications to the Code

**TIMER**

TIMER is a CCCC subroutine with a variety of options for returning elapsed and clock time, job identification, data, etc. Every computer installation has its own set of FORTRAN callable system routines for providing some or all of this information, and so no universal TIMER can offer all options. The version of TIMER included in both IBM and CDC versions of the Code Center package returns elapsed CP time; in addition the IBM version returns the date and wall clock time. SYN3D uses TIMER to determine the time spent in the several parts of the calculation and to provide the date and clock time for file identification and page headers.

**SEEK Initialization**

Those installations which have their own SEEK and an established SEEK initialization procedure will want to make some changes in the main driver of SYN3D. In the Code Center SYN3D the correspondence between file name and logical unit number is set by the DATA statements defining the elements of the arrays DSNNAME and NREF. The file named DSNNAME(N) is assigned to logical unit number NREF(N).

On the first call to SEEK (with an operation code of 3) the entire DSNNAME array (up to the $), as well as the NREF array, is transferred to a table in the subroutine. Files with the same names are assigned version numbers in ascending order. Blank file names are ignored. Existence flags are initialized to 0 (the file has not been written).

The main driver then reads the SEEK initialization card and looks for columns which contain a number greater than zero. It interprets column numbers as logical unit numbers and for a non-zero entry in a column requests the corresponding file name from SEEK (with an operation code of 5). When it has the file name and version number it calls SEEK with an operation code of 1 to set the existence flag.

Changes to any part or all of this initialization procedure can be made by recoding the main driver and SEEK. No other routines are affected.

**Eliminating BCD Input Data**

Some installations may prefer to run SYN3D in a mode that completely eliminates direct BCD card input. LASL, for example, uses a general input processor which converts BCD cards to binary interface files; ANL has the ARC System input processors SCAN and STUFF which convert BCD cards to BCD files.

The first overlay in SYN3D, CARDS, reads BCD data either from the BCD card input file or the BCD disk file SYNFIL, depending on the value of the
sentinel IWHERE set in the main driver. CARDS then rewrites the input data to the binary file SYNCON (see the Appendix E for the file description). SYNCON is a binary version of the BCD card input; the data on each card is rewritten to an unformatted record. The user can write SYNCON outside of SYN3D and drop the call to CARDS from the main driver. The rest of the code is unaffected.

CARDS provides one function which is lost if the overlay is eliminated. For historical reasons, SYN3D was set up to build 3D (or 2D) geometries from 2D (or 1D) GEODST files. After the code was written an option was added to permit an input 3D (or 2D) file. CARDS was coded to break up an input 3D (or 2D) GEODST into the required number of 2D (or 1D) files and to create the additional, required BCD input data (cards types 4 and 5) before writing SYNCON. If CARDS is eliminated the user must specify 3D (or 2D) models in terms of 2D (or 1D) GEODST files.

**Matrix Inversion and Multiplication Subroutines**

The matrix inversion routine INVERT and matrix arithmetic routine ARITH which are supplied with the Code Center SYN3D are coded to minimize running time for large problems (many groups and/or expansion functions) when the code is compiled under the IBM FORTRAN H Extended compiler and run on the 370/195. There may be more efficient routines available for other computers.

**Restrictions on Problem Size in SOLVE**

It is unlikely that the problem size limitations (see Section III-G) will affect users who have large IBM computers; more than enough fast core should be available. CDC users (with a 50K core limitation), however, may reach the problem size limit when the number of energy groups (after group collapsing) times the number of expansion functions used concurrently at any one axial mesh interval is about 50.

The difficulty is with the solution overlay SOLVE. As it is currently written SOLVE must have enough storage to keep six matrices in-core at the same time. These matrices are $A_{\mu}$, $A_{\sigma}$, $A_{\tau}$, $H_{\tau}$, REM and FIS (see Eqs. (38), (39), (41), (42), (43) and Section III-D). With a modest investment in programming some or most of these matrices could be kept in LCM.
APPENDIX A

Code Abstract

1. Name or Designation - SYN3D

2. Computer for which Program is Designed and Others upon which it is Operable - IBM 370/195, CDC 7600.

3. Description of Problem or Function - SYN3D solves the direct and adjoint, diffusion theory, static eigenvalue equations in two and three dimensions. The geometries available are x-y, r-z, x-y-z and triangular-z.

4. Method and Solution - SYN3D uses single-channel spatial flux synthesis to calculate approximate solutions to the three-dimensional (or two-dimensional) diffusion theory difference equations. Synthesis expansion functions must be supplied by the user from two-dimensional (or one-dimensional) finite-difference calculations performed by some other code. SYN3D sets up the synthesis equations and solves them by power iteration with Wielandt acceleration. Each iteration is an exact inversion of the block-tridiagonal synthesis equations by forward-elimination, backward-substitution.

5. Restrictions on the Complexity of the Problem - For the most part, SYN3D uses variably dimensioned arrays and disk scratch files to manage data for any size problem in the available fast core. The only serious limitation is on the product of the number of groups and the maximum number of expansion functions used at any particular axial elevation; for CDC users (with a 64K machine) this product is limited to about 50.

6. Typical Running Time - Exclusive of the time required to generate expansion functions SYN3D will solve an 11-group, 12,000 mesh point model with two expansion functions in .9 minutes (CPU). A 28-group, 30,000 mesh point model with three expansion functions requires 8.9 minutes (CPU). These times are for jobs run on an IBM 370/195. For large problems the PP time is less than the CPU time.

7. Unusual Features of the Program - The difference equations SYN3D solves are the mesh-interval-centered type. Expansion functions should be generated using diffusion theory codes solving the same equations (e.g. 2DB, CITATION, VENTURE). The code is designed with restart capabilities which reduce, on the average, the running times for individual problems when a series of similar problems is to be run. SYN3D requires input cross sections, expansion functions and geometry descriptions in the Version III formats defined by the Committee on Computer Code Coordination (CCCC). Special options include group collapsing and the use of different expansion functions in different axial zones of the model.
8. Related and Auxiliary Programs - SYN3D requires binary input files containing cross sections, expansion functions and geometry descriptions. Although a small program is provided with the code to read these data from cards, the user may wish to use other programs which generate CCCC files. A finite-difference, diffusion theory program is required to generate expansion functions.

9. Status -


11. Machine Requirements - The code requires at least a 35 K full-word core to execute small problems and runs more efficiently with larger storage. SYN3D does not use BULK (IBM) or LCM (CDC) storage. Depending on the complexity of the problem, SYN3D may require up to 45 logical units.

12. Programming Language Used - FORTRAN IV. Both IBM and CDC version of SYN3D contain a few routines written in assembler language.

13. Operating System or Monitor under which Program is Executed - The IBM version of SYN3D has been compiled and executed under OS-37U with the FORTRAN H Extended compiler. The CDC SYN3D has been compiled and executed at Berkeley Laboratory under the COKE System with the FORTRAN F4N4.5 compiler.

14. Any Other Programming or Operating Information or Restrictions -

15. Name and Establishment of Author -

C. H. Adams
Applied Physics Division
Argonne National Laboratory
Argonne, Illinois 60439

16. Material Available - Separate tapes are available for the IBM and CDC versions of SYN3D. The SYN3D package includes:

Source decks for SYN3D and auxiliary input program,
Input for sample problems,
Reference report.

17. Category - C

Keywords - two-dimensional, three-dimensional, diffusion equations, synthesis, x-y, r-z, x-y-z, triangular-z.
APPENDIX B. STP018 - THE ARC SYSTEM STANDRAD PATH FOR SYN3D.

DOUBLE PRECISION DSNAME
REAL*8 STFNAM, BLKNAME
REAL*8 NCNTIP, GEOM, CHANGE, BC, XSCMIN, XHOMG, HOMOG, SYN3D,
1 SYNBLK, TRGNTIP, ANIP, X, COMP3X
COMMON / STP3ARC / STFNAM, BLKNAME(50), IFLTAB(3,50), NRLOCK, NSET
COMMON / IOPT / NIN, NOUT, NOUT2
DIMENSION DSNAME(87)
DATA DSNAME / 6HGEOM01, 6HGEOM02, 6HGEOM03, 6HGEOM04, 6HGEOM05,
1 6HGEOM06, 6HGEOM07, 6HGEOM09, 6HGEOM10,
2 6HGEOM01, 6HGEOM02, 6HGEOM03, 6HGEOM04, 6HGEOM05, 6HGEOM06,
3 6HGEOM07, 6HGEOM08, 6HGEOM09, 6HGEOM10,
4 4H3C01, 4H3C02, 4H4C03, 4HBC04, 4HBC05, 4HBC06, 4HBC07,
5 4HBC08, 4HBC09, 4HBC10,
6 6HFLUX01, 6HFLUX02, 6HFLUX03, 6HFLUX04, 6HFLUX05, 6HFLUX06,
7 6HFLUX07, 6HFLUX09, 6HFLUX10, 6HFLUX12, 6HFLUX12,
8 6HRTFL01, 6HRTFL02, 6HRTFL03, 6HRTFL04, 6HRTFL05, 6HRTFL06,
9 6HRTFL07, 6HRTFL08, 6HRTFL09, 6HRTFL09,
1 6HINTC1, 6HINTC2, 6HINTC3, 6HINTC4, 6HINTC5, 6HINTC6,
2 6HINTC7, 6HINTC8, 6HINTC9, 6HINTC10, 6HINTC11,
3 6HINTC12, 6HINTC13, 6HINTC14, 6HINTC15, 6HINTC16,
4 6HINTC17, 6HINTC18, 6HINTC19, 6HINTC20, 6HINTC21,
5 6HINTC22, 6HINTC23, 6HINTC24, 6HINTC25, 6HINTC26,
6 6HINTC27, 6HINTC28, 6HINTC29, 6HINTC30, 6HINTC31,
7 6HINTC32, 6HINTC33, 6HINTC34, 6HINTC35, 6HINTC36,
8 6HINTC37, 6HINTC38, 6HINTC39, 6HINTC40, 6HINTC41,
9 6HINTC42, 6HINTC43, 6HINTC44, 6HINTC45, 6HINTC46,
0 6HINTC47, 6HINTC48, 6HINTC49, 6HINTC50, 6HINTC51,
1 6HINTC52, 6HINTC53, 6HINTC54, 6HINTC55, 6HINTC56,
2 6HINTC57, 6HINTC58, 6HINTC59, 6HINTC60, 6HINTC61,
3 6HINTC62, 6HINTC63, 6HINTC64, 6HINTC65, 6HINTC66,
4 6HINTC67, 6HINTC68, 6HINTC69, 6HINTC70, 6HINTC71,
5 6HINTC72, 6HINTC73, 6HINTC74, 6HINTC75, 6HINTC76,
6 6HINTC77, 6HINTC78, 6HINTC79, 6HINTC80, 6HINTC81,
7 6HINTC82, 6HINTC83, 6HINTC84, 6HINTC85, 6HINTC86,
8 6HINTC87, 6HINTC88, 6HINTC89, 6HINTC90, 6HINTC91,
9 6HINTC92, 6HINTC93, 6HINTC94, 6HINTC95, 6HINTC96,
0 6HINTC97, 6HINTC98, 6HINTC99, 6HINTC100, 6HINTC101,
1 6HINTC102, 6HINTC103, 6HINTC104, 6HINTC105, 6HINTC106,
2 6HINTC107, 6HINTC108, 6HINTC109, 6HINTC110, 6HINTC111,
3 6HINTC112, 6HINTC113, 6HINTC114, 6HINTC115, 6HINTC116,
4 6HINTC117, 6HINTC118, 6HINTC119, 6HINTC120, 6HINTC121,
5 6HINTC122, 6HINTC123, 6HINTC124, 6HINTC125, 6HINTC126,
6 6HINTC127, 6HINTC128, 6HINTC129, 6HINTC130, 6HINTC131,
7 6HINTC132, 6HINTC133, 6HINTC134, 6HINTC135, 6HINTC136,
APPENDIX B. STP018. CONTINUED.

STPNAM=GEOM
CALL STUFF
IF( NRET.LE.0 ) GO TO 10
L=L+1
K=K+1

C CREATE A MACROSCOPIC CROSS SECTION FILE ON THE FIRST LOOP IF
C NONE EXISTS. ICALL=4 FOR NO OUTPUT, ICALL=1 FOR OUTPUT.
C
ICALL=4
IF( L.GT.1 ) GO TO 4
CALL SNIPP( COMPS, NIS, I0)
IF( NXS.GT.0 ) GO TO 4
CALL SNIPP( XSFCTN, NIS, I0)
IF( NXS.GT.0 ) GO TO 4
TXS=1
CALL LINK(XSFCTN)
CALL LINK( HOMOG, ICALL, TO, TO, TO, IXS)

C CREATE GEOM AND BC FILES.
C
4 CONTINUE
CALL SNIPP( DSNAM(L), J, TO)
IF( J.GT.0 ) GO TO 2
CALL SNIPP( GEOM, I, I1)
CALL SNIPP( DSNAM(L), J, I1)
CALL SNIPP( CHANGE, J, J)
CALL SNIPP( GEOM, I, I2)
CALL SNIPP( BC, I, I2)
CALL SNIPP( DSNAM(K), J, I1)
CALL SNIPP( CHANGE, I, J)
CALL SNIPP( PC, I, I1)
CALL SNIPP( ANIP, NIP, TO)
"?*?"( LIP, LR, 0 ) GO TO 2
READ(NIP,500) V, NPE
READ(NIP,501) (ISCR(I),I=1,NPE)
V=0
IF( TSCPL0,J,0 ) GO TO 6
CALL LINK("PCHIP")
6 CONTINUE
CALL LINK(GHIP,T)
CALL SNIPP( GEOM, J, I1)
CALL SNIPP( ANIP, NIP, J1)
CALL SNIPP( CHANGE, I, 1)
CALL SNIPP( GEOM, J, 12)
CALL SNIPP( PC, I, 11)
CALL SNIPP( DSNAM(K), J, I1)
CALL SNIPP( CHANGE, J, J)
CALL SNIPP( BC, J, I2)
GO TO 2
APPENDIX R. STP018. CONTINUED.

C
C LINK TO SYN3D.
C
10 CONTINUE
   STFNAM=SYNRLK
   CALL STOPP
   IF( NRET.LE.0 ) GO TO 100
   CALL LINK(SYN3D)
   GO TO 10
100 RETURN
END
APPENDIX C. ARCSPO18 - THE ARC SYSTEM CATALOGUED PROCEDURE FOR SYN3D.

//ARCSPO18 PROC ACCOEF='&ACCOEF',ACCDSP='(NEW,DELETE)',ACCVOL=,
// BC01='&BC01',BC02='&BC02',BC03='&BC03',
// BC04='&BC04',BC05='&BC05',BC06='&BC06',
// BC07='&BC07',BC08='&BC08',BC09='&BC09',BC10='&BC10',
// BCDSP01='(NEW,DELETE)',BCDSP10='(NEW,DELETE)',
// BCDSP02='(NEW,DELETE)',BCDSP03='(NEW,DELETE)',
// BCDSP04='(NEW,DELETE)',BCDSP05='(NEW,DELETE)',
// BCDSP06='(NEW,DELETE)',BCDSP07='(NEW,DELETE)',
// BCDSP08='(NEW,DELETE)',BCDSP09='(NEW,DELETE)',
// BCVOL01=',BCVOL02=',BCVOL03=',BCVOL04=',BCVOL05=',
// BCVOL06=',BCVOL07=',BCVOL08=',BCVOL09=',
// COMPS1='&EXSCMIN1',COMPS2='&EXSCMIN2',
// CXDSIP='(NEW,DELETE)',CXSVDLM=',
// CXSBK1=1028,CXSBK2=6136,
// DCCOEF='&DCCOEF',DCCDSP='(NEW,DELETE)',DCCVOL=,
// DITFNT1=NULLFILE, DITFNT2=NULLFILE,
// DMTVOL1=NULLFILE, DMTVOL2=NULLFILE,
// DMTDSP01='(NEW,DELETE)', DMTDSP02='(NEW,DELETE)',
// DMTDSP03='(NEW,DELETE)', DMTDSP04='(NEW,DELETE)',
// DMTDSP05='(NEW,DELETE)', DMTDSP06='(NEW,DELETE)',
// DMTDSP07='(NEW,DELETE)', DMTDSP08='(NEW,DELETE)',
// DMTDSP09='(NEW,DELETE)', DMTDSP10='(NEW,DELETE)',
// FLUX01=NULLFILE, FLUX02=NULLFILE,
// FLUX04=NULLFILE, FLUX05=NULLFILE,
// FLUX07=NULLFILE, FLUX08=NULLFILE,
// FLUX09=NULLFILE, FLUX10=NULLFILE,
// FLVC01=',FLVC02=', FLVC03=',FLVC04=',
// FLVC05=',FLVC06=', FLVC07=',FLVC08=',
// FLVC09=',FLVC10=, FULLRLK=12280,
// G/DSP01='(NEW,DELETE)', G/DSP10='(NEW,DELETE)',
// GEDSP02='(NEW,DELETE)', GEDSP03='(NEW,DELETE)',
// GEDSP04='(NEW,DELETE)', GEDSP05='(NEW,DELETE)',
// GEDSP06='(NEW,DELETE)', GEDSP07='(NEW,DELETE)',
// GEDSP08='(NEW,DELETE)', GEDSP09='(NEW,DELETE)',
// G/EOM01='&GEOM01', G/EOM02='&GEOM02',
// G/EOM04='&GEOM04', G/EOM05='&GEOM05',
// G/EOM07='&GEOM07', G/EOM08='&GEOM08',
// G/EOM10='&GEOM10', G/EOM11='&GEOM11',
// GEOL01=',GEOL02=', GEOL03=', GEOL04=',
// GEOL05=',GEOL06=', GEOL07=', GEOL08=',
// GEOL09=',GEOL10=', GEOL11=', GEOL12=',
// H/SF1K=6136,
// ISOXS=NULLFILE, ISOXS='(OLD,KEEP)',
// INTT0C1=NULLFILE, INTT0C2='&INTT0C2',
// LIBBLK2=12280,
// MICRS1=NULLFILE, MICRS2=NULLFILE,
// MICRSV1=NULLFILE, MICRSV2=NULLFILE,
// MXVOL='&MXVOL',
// NDXSRF=NULLFILE, NDXDSP='(OLD,KEEP)',
// NDXVOL=',
APPENDIX C. ARCSP018. CONTINUED.

```
// PATH=ST018,
// POSTLIB='SYS1.DUMMYLIB',PPLIB='SYS1.DUMMYLIB',
// QRTSPBLK=3064,
// TINLTM='(600,0)',
// TOCDSP1='(OLD,KEEP)',TOCDSP2='(NEW,DELETE)',
// TOCVCL1=,TOCVOL2=,
// UNITS=BATCHDSK,
// VOLINT1=NULLFILE,VNTVOL1='(OLD,KEEP)',
// VOLINT2='&WCLTNT2',VNTVOL2='(NEW,DELETE)',
// ZNATDN=NULLFILE,ZNAVOL='(OLD,KEEP)',

="/*****
** ******************************************************
*
CATALOGUED PROCEDURE FOR SYND3D FLUX
SYNTESTS CALCULATION
*
**
**
**
**
**
**
**
**
****
*** Parameter Default Value Usage
****
****
ACCOEP GACCOEP DSN OF ACCOEP FILE 75
ACCNOP (NEW,DELETE) DISPOSITION OF ACCOEP 75
ACCVOL ------ VOLUME FOR ACCOEP 75
BC01- 6BC01- DSN OF BC FILPS 31-40
BC10 &BC10 ----
RCDSPO1- (NEW,DELETE) DISPOSITION OF BC FILES 31-40
RCDSPO10 ----
RCVOL01- ------ VOLUMFS FOR BC FILES 31-40
RCVOL10 ----
COMPS1 EKSCMNT1 DSN OF XS.C.MIN FILP 1 67
COMPS2 EKSCMNT2 DSN OF XS.C.MIN FILP 2 67
CXSDFSP (NEW,DELETE) DISPOSITION OF XS.C.MIN 67
CXSVDL ---- VOLUME FOR XS.C.MIN 67
CXSXBLK1 1028 BIRSIZE FOR XS.C.MIN FILE 1 67
CXSXBLK2 6136 EIRSIZE FOR XS.C.MIN File 2 67
DCCSDF (NEW,DELETE) DISPOSITION OF DCCSDF 74
DCCOEP EDCOEP DSN OF DCCOEP FILE 74
DCCVOL ------ VOLUME FOR DCCOEP 74
DTPINT1 NULLFILE DSN OF INPUT DIFINT FILF 70
DTPINT2 EDTPTNT2 DSN OF OUTPUT DIFINT FILE 71
DPSPRT ? ROUTE DUMP TO FICHE SYSUDUMP
DNTDSP1 (OLD,KEEP) DISPOSITION OF INPUT DTPINT 70
DNTDSP2 (NEW,DELETE) DISPOSITION OF OUTPUT DTPINT 71
DNTVOL1 ------ VOLUME FOR INPUT DTPINT 70
DNTVOL2 ------ VOLUME FOR OUTPUT DTPINT 71
PLDSP01- (OLD,KEEP) DISPOSITION OF INPUT FLUXPS 41-50
PLDSP10 ----
PLUX1- NULLFILE DSN OF INPUT FLUX FILPS 41-50
```

APPENDIX C. ARCSP018. CONTINUED.

/*
** PLVOL01  ------  VOLUMES FOR INPUT FLUXES  41-50
** PLVOL10
** FULLRLK  12280  FULL TRACK BLKSIZE
** GERDSP01  (NEW, DELETE)  DISPOSITION OF GFOM FILES  11-20
** GERDSP10
** GEOM01  6GFOM01  DSN OF GEOM FILES  11-20
** GEOM10  &GEOM01
** GPVOL01  ------  VOLUMES FOR GEOM FILES  11-20
** GPVOL10
** HALPYLK  6136  HALF TRACK BLKSIZE
** INTTOC1 NULLFILE  DSN OF INPUT INTTOC FILE  65
** INTTOC2 &INTTOC2  DSN OF OUTPUT INTTOC FILE  66
** ISDSP  (OLD, KEEP)  DISPOSITION OF INPUT ISOTXS  94
** ISOX1 NULLFILE  DSN OF INPUT ISOX1  94
** ISOXVOL  ------  VOLUME FOR ISOTXS  94
** LBRKLK2  12280  ELKSIZE, OUTPUT INTTOC & VOLTNT  69, 71
** MICRXS1 NULLFILE  DSN OF XS.ISO FILE 1  79
** MICRXS2 NULLFILE  DSN OF XS.ISO FILE 2  79
** NICMULV  ------  VOLUME FOR XS.ISO  79
** NDXSP  (OLD, KEEP)  DISPOSITION OF INPUT NDXSRF  95
** NDXSRF NULLFILE  DSN OF INPUT NDXSRF  95
** NDXVOL  ------  VOLUME FOR NDXSRF  95
** PTH  STE01A  PROGRAM NAME EXEC
** POSTLIB  SYS1.DUMMYLIB  ADDITIONAL LIBRARY DSN STEPLIB
** PRELIB  SYS1.DUMMYLIB  ADDITIONAL LIBRARY DSN STEPLIB
** QRTBLK  3064  QUARTER TRACK BLKSIZE
** TIME  (600,0)  STMP TIME LIMIT EXEC
** TOCSP1  (OLD, KEEP)  DISPOSITION OF INPUT INTTOC  65
** TOCSP2  (NEW, DELETE)  DISPOSITION OF OUTPUT INTTOC  66
** TOCVOL1  ------  VOLUME FOR INPUT INTTOC  65
** TOCVOL2  ------  VOLUME FOR OUTPUT INTTOC  66
** UTCBATCH  BATCHDSK  DEFAULT UNIT PARAMETER
** UNTDSP1  (OLD, KEEP)  DISPOSITION OF INPUT VOLINT  68
** UNTDSP2  (NEW, DELETE)  DISPOSITION OF OUTPUT VOLINT  69
** UNTVOL1  ------  VOLUME FOR INPUT VOLINT  68
** UNTVOL2  ------  VOLUME FOR OUTPUT VOLINT  69
** VOLINT1 NULLFILE  DSN OF INPUT VOLINT FILE  68
** VOLINT2 &VOLINT2  DSN OF OUTPUT VOLINT FILE  69
** ZNATDN NULLFILE  DSN OF INPUT ZNATDN  96
** ZNAVOL  ------  VOLUME FOR ZNATDN  96
** ZNADSP  (OLD, KEEP)  DISPOSITION OF INPUT ZNADSP  96
** */

*********************************************************************************

/*
** GO FYEC PM=SPATH,TIMR=TIMLIM
** STEPLIB  DD DSN=PRELIB,DISP=SHR
** DD DSN=C116.A21006.MODLIB,DISP=SHR
** DD DSN=C116.abc.MODLIB,DISP=SHR
** DD DSN=POSTLIB,DISP=SHR
** FTSPO031 DD DSNAM=SYS1
** FT06PO01 DD SYSOUT=A
** SYSUT4P DD SYSOUT=6MPDEST
*/
APPENDIX C. ARCSPO18. CONTINUED.

//FT09F001 DD UNIT=2314, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=3364)
//FT11F001 DD DSN=6GEO901, VOL=SER=6GEOVOL01, DISP=GEDSPO1,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
/* FILES 11 thru 20 ARE GEOF FILES
//FT12F001 DD DSN=6GEO902, VOL=SER=6GEOVOL02, DISP=GEDSPO2,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT13F001 DD DSN=6GEO903, VOL=SER=6GEOVOL03, DISP=GEDSPO3,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT14F001 DD DSN=6GEO904, VOL=SER=6GEOVOL04, DISP=GEDSPO4,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT15F001 DD DSN=6GEO905, VOL=SER=6GEOVOL05, DISP=GEDSPO5,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT16F001 DD DSN=6GEO906, VOL=SER=6GEOVOL06, DISP=GEDSPO6,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT17F001 DD DSN=6GEO907, VOL=SER=6GEOVOL07, DISP=GEDSPO7,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT18F001 DD DSN=6GEO908, VOL=SER=6GEOVOL08, DISP=GEDSPO8,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT19F001 DD DSN=6GEO909, VOL=SER=6GEOVOL09, DISP=GEDSPO9,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT20F001 DD DSN=6GEO910, VOL=SER=6GEOVOL10, DISP=GEDSPO10,
// UNIT=UNITS, SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT21F001 DD DSN=6GEO911, DISP=(NEW,DFLTF), SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT22F001 DD DSN=6GEO912, DISP=(NEW,DFLTF), SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT23F001 DD DSN=6GEO913, DISP=(NEW,DFLTF), SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT24F001 DD DSN=6GEO914, DISP=(NEW,DFLTF), SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT25F001 DD DSN=6GEO915, DISP=(NEW,DFLTF), SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT26F001 DD DSN=6GEO916, DISP=(NEW,DFLTF), SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT27F001 DD DSN=6GEO917, DISP=(NEW,DFLTF), SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
//FT28F001 DD DSN=6GEO918, DISP=(NEW,DFLTF), SPACE=(CYL, (1,1)),
// DCB=(RECFM=VB,S,LRECL=84, BLKSIZE=HALFBLK)
APPENDIX C. ARCSPO18. CONTINUED.

//FT29P001 DD DSN=&GEODS09, DISP=(NEW, DELETE), SPACE=(CYL, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=6HALPBK), UNIT=SASCR
//FT30P001 DD DSN=&GEODS10, DISP=(NEW, DELETE), SPACE=(CYL, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=6HALPBK), UNIT=SASCR
/* FILES 21 THRU 30 ARE CCCC GEODST FILES
//FT31P001 DD DSN=&BCO1, VOL=SER=BCVOL01, DISP=&BCDSP01,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
/* FILES 31 THRU 40 ARE THE BC FILES ASSOCIATED WITH THE GEOM FILES
/* FILES (11 THRU 20)
//FT32P001 DD DSN=&BCO2, VOL=SER=BCVOL02, DISP=&BCDSP02,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
//FT33P001 DD DSN=&BCO3, VOL=SER=BCVOL03, DISP=&BCDSP03,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
//FT34P001 DD DSN=&BCO4, VOL=SER=BCVOL04, DISP=&BCDSP04,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
//FT35P001 DD DSN=&BCO5, VOL=SER=BCVOL05, DISP=&BCDSP05,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
//FT36P001 DD DSN=&BCO6, VOL=SER=BCVOL06, DISP=&BCDSP06,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
//FT37P001 DD DSN=&BCO7, VOL=SER=BCVOL07, DISP=&BCDSP07,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
//FT38P001 DD DSN=&BCO8, VOL=SER=BCVOL08, DISP=&BCDSP08,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
//FT39P001 DD DSN=&BCO9, VOL=SER=BCVOL09, DISP=&BCDSP09,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
//FT40P001 DD DSN=&BC10, VOL=SER=BCVOL10, DISP=&BCDSP10,
// UNIT=SUNITS, SPACE=(TRK, (1, 1)),
// DCB=(RECFM=VBS, LRECL=X, BLKSIZE=304)
/* FILES 31 THRU 40 ARE THE BC FILES ASSOCIATED WITH THE GEOM FILES
/* FILES (11 THRU 20)
//FT41P001 DD DSN=FLUX01, UNIT=SUNITS, VOL=SER=FLVOL01, DISP=FLDSP01
// FT42P001 DD DSN=FLUX02, UNIT=SUNITS, VOL=SER=FLVOL02, DISP=FLDSP02
// FT43P001 DD DSN=FLUX03, UNIT=SUNITS, VOL=SER=FLVOL03, DISP=FLDSP03
// FT44P001 DD DSN=FLUX04, UNIT=SUNITS, VOL=SER=FLVOL04, DISP=FLDSP04
// FT45P001 DD DSN=FLUX05, UNIT=SUNITS, VOL=SER=FLVOL05, DISP=FLDSP05
// FT46P001 DD DSN=FLUX06, UNIT=SUNITS, VOL=SER=FLVOL06, DISP=FLDSP06
// FT47P001 DD DSN=FLUX07, UNIT=SUNITS, VOL=SER=FLVOL07, DISP=FLDSP07
// FT48P001 DD DSN=FLUX08, UNIT=SUNITS, VOL=SER=FLVOL08, DISP=FLDSP08
// FT49P001 DD DSN=FLUX09, UNIT=SUNITS, VOL=SER=FLVOL09, DISP=FLDSP09
// FT50P001 DD DSN=FLUX10, UNIT=SUNITS, VOL=SER=FLVOL10, DISP=FLDSP10
/* FILES 41 THRU 50 ARE INPUT PR.D2, FA.D2, FP.D1 AND FA.D1 FILES
//FT41P001 DD DSN=FLUX01, UNIT=SUNITS, VOL=SER=FLVOL01, DISP=FLDSP01
// FILES 41 THRU 50 ARE INPUT PR.D2, FA.D2, FP.D1 AND FA.D1 FILES
//FT42P001 DD DSN=FLUX02, UNIT=SUNITS, VOL=SER=FLVOL02, DISP=FLDSP02
//FT43P001 DD DSN=FLUX03, UNIT=SUNITS, VOL=SER=FLVOL03, DISP=FLDSP03
//FT44P001 DD DSN=FLUX04, UNIT=SUNITS, VOL=SER=FLVOL04, DISP=FLDSP04
//FT45P001 DD DSN=FLUX05, UNIT=SUNITS, VOL=SER=FLVOL05, DISP=FLDSP05
//FT46P001 DD DSN=FLUX06, UNIT=SUNITS, VOL=SER=FLVOL06, DISP=FLDSP06
//FT47P001 DD DSN=FLUX07, UNIT=SUNITS, VOL=SER=FLVOL07, DISP=FLDSP07
//FT48P001 DD DSN=FLUX08, UNIT=SUNITS, VOL=SER=FLVOL08, DISP=FLDSP08
//FT49P001 DD DSN=FLUX09, UNIT=SUNITS, VOL=SER=FLVOL09, DISP=FLDSP09
//FT50P001 DD DSN=FLUX10, UNIT=SUNITS, VOL=SER=FLVOL10, DISP=FLDSP10
/* FILES 41 THRU 50 ARE INPUT PR.D2, FA.D2, FP.D1 AND FA.D1 FILES
APPENDIX C. ARCSPO18. CONTINUED.

```plaintext
//FT51F001 DD DSN=&RTFL01,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&HALFBLK)
// FILE 51 THRU 60 ARE CCCC RTFLUX AND ATFLUX FILES
//FT52F001 DD DSN=&RTFL02,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&HALFBLK)
//FT53F001 DD DSN=&RTFL03,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&HALFBLK)
//FT54F001 DD DSN=&RTFL04,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&HALFBLK)
//FT55F001 DD DSN=&RTFL05,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&HALFBLK)
//FT56F001 DD DSN=&RTFL06,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&HALFBLK)
//FT57F001 DD DSN=&RTFL07,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&HALFBLK)
//FT58F001 DD DSN=&RTFL08,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&HALFBLK)
//FT59F001 DD DSN=&RTFL09,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&HALFBLK)
//FT60F001 DD DSN=&RTFL10,UNIT=SASCR,DISP=(NEW,DELETE),
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&FULLBLK)
// FILE 51 THRU 60 ARE CCCC RTFLUX AND ATFLUX FILES
//FT61F001 DD DSN=&SYNFIL,DISP=(NEW,DELETE),UNIT=SASCR,
  DCB=(RECFM=VBS,LRECL=84,BLKSIZE=3156),SPACE=(TRK,(1,1))
/* THE BCD INPUT FILE
//FT62F001 DD DSN=&XSCLN,DISP=(NEW,DELETE),UNIT=SASCR,
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6447)
/* A ONE-FILE XS.C.MTN
//FT63F001 DD DSN=&RTFPSL,DISP=(NEW,DELETE),UNIT=SASCR,
  SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&FULLBLK)
/* REWRITTEN FLUXES AND GEOMETRIES
//FT64F001 DD DSN=&REQXS1,DISP=(NEW,DELETE),UNIT=SASCR,
  SPACE=(CYL,(2,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&FULLBLK)
/* REWRITTEN CROSS SECTIONS
//FT65F001 DD DSN=FTNTTOC1,VOL=SER=BTOCV01,DISP=EIOCDSP1,UNIT=SASCR
/* THE INPUT TOTOC FILE
//FT66F001 DD DSN=FTNTTOC2,VOL=SER=BTOCVOL2,DISP=EIOCDSP2,
  SPACE=(TRK,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&RTBLK),
  UNIT=SASCR
/* THE OUTPUT TOTOC FILE
//FT67F001 DD DSN=COMPS1,UNIT=SASCR,VOL=SER=CTVOS1,DISP=CSDISP,
  SPACE=(TRK,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&CSRLK1)
/* THE FIRST FILE OF THE INPUT, TWO-FILE XS.C.MTN
//FT68F002 DD DSN=COMPS2,UNIT=SASCR,VOL=SER=CTVOS2,DISP=CSDISP,
  SPACE=(TRK,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=&CSRLK2)
/* THE SECOND FILE OF THE INPUT, TWO-FILE XS.C.MTN
//FT69F001 DD DSN=VOLINT1,VOL=SER=ETVOL1,DISP=EVTDSP1,UNIT=SASCR
/* THE INPUT VOLINT FILE
```
APPENDIX C. ARCSPO1R. CONTINUED.

//FT85P001 DD DSN=SPCRIT,DISP=(NEW,DELETE),UNIT=SASCR,
//SPACE=(TRK,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=388)
//FT86P001 DD DSN=XSCAUX1,DISP=(NEW,DELETE),UNIT=SASCR,
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=516)
//FT87P001 DD DSN=XSCAUX2,DISP=(NEW,DELETE),UNIT=SASCR,
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT88P001 DD DSN=SCR003A,DISP=(NEW,DELETE),UNIT=SASCR,
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT89P001 DD DSN=XSMAUX,DISP=(NEW,DELETE),UNIT=SASCR,
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT90P001 DD DSN=XSMMIN,DISP=(NEW,DELETE),UNIT=SASCR,
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT91P001 DD DSN=ATFL01,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(TRK,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT92P001 DD DSN=ATFINT,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT93P001 DD DSN=EISOTXS,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(TRK,(5,2)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT94P001 DD DSN=ENTNUTS,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(TRK,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT95P001 DD DSN=FR003N,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(TRK,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT96P001 DD DSN=FRZFLUY,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(TRK,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT97P001 DD DSN=XSMMIN,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT98P001 DD DSN=XSMAUX,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT99P001 DD DSN=ENGNTS,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)
//FT100P001 DD DSN=:init,UNIT=SASCR,DISP=(NEW,DELETE),
//SPACE=(CYL,(1,1)),DCB=(RECFM=VBS,LRECL=X,BLKSIZE=6HALFBLK)

* FILES 80 THRU 84 ARE NEEDED IN THE CROSS SECTION GENERATION

* THE PRIMARY CONTROLL FILE FOR SYN3D

* A CCCC FLUXES BY ZONE, CCCC FORMAT

* AVERAGE FLUXES BY ZONE, CCCC FORMAT

* INPUT MICROSCOPIC CROSS SECTIONS IN CCCC FORMAT

* CROSS SECTIION REFERENCE FILE, CCCC FORMAT

* ATOM DENSITIES, CCCC FORMAT

* FILE 1 OF COMPOSITION CROSS SECTION SET XS.M.W1

* FILE 2 OF COMPOSITION CROSS SECTION SET XS.M.W1

* ANYONE EXPERIENCING DIFFICULTY WITH THIS PROCEDURE

* SEE C. H. ADAMS, BLDG 209, ROOM W-117
APPENDIX D. INPUT DATA SET SYNPI.

PREPARED 09/19/74 AT ANL

THIS IS A USPR SUPPLIED BCD DATA SET.

THE LIST FOR EACH RECORD IS GIVEN IN TERMS OF THE BCD FORMAT OF THAT DATA CARD.

COLUMNS 1-2 CONTAIN THE CARD TYPE NUMBER.

BLANK FIELDS PRODUCE THE DEFAULT VALUES.

CARD TYPES 01 THROUGH 06 CONTAIN THE MODEL GEOMETRY SPECIFICATIONS. CARD TYPES 07 THROUGH 21 DESCRIBE THE SYNTHESIS SCHEME.

-----------------------------------------------------------------------

PROBLEM TITLE (TYPE 01)

FORMAT---- (T2, 4X, 11A6)

COLUMNS CONTENTS...IMPLICATIONS, IF ANY

1-2 01

7-72 ANY ALPHANUMERIC CHARACTERS.

AS MANY TYPE 01 CARDS MAY BE USED AS ARE NECESSARY.

-----------------------------------------------------------------------

GENERAL MODEL GEOMETRY SPECIFICATIONS (TYPE 07)

FORMAT---- (T2, 4X, 11I6)

COLUMNS CONTENTS...IMPLICATIONS, IF ANY

1-2 02

7-12 MAXST7, SIZE OF MAIN COPE STORAGE ARRAY IN REAL*8

WORDS (SINGLE WORDS ON CDC SYSTEMS), SPECIFIED IN THOUSANDS OF WORDS (DEFAULT=10, I.F., 10000 WORDS).
APPENDIX D: INPUT DATA SET SYNFIL. (CONT'D.)

CD 13-18 MCALC, CALCULATION OPTIONS.
CD 0...DO NOT SOLVE SYNTHESIS EQUATIONS, ONLY UPDATE
CD INTEGRALS TO INCLUDE WHAT IS NEEDED FOR FLUX AND
CD POWER EDITS AND  GO TO EDITS. IF THERE ARE ANY
CD CARDS OF TYPES 16 THRU 21 A COMBINING COEFFICIENT
CD (DCCOEF OR ACCOEF) MUST BE AVAILABLE
CD (DEFAULT).
CD 1...UPDATE INTEGRALS AND SOLVE A DIRECT EIGENVALUE
CD PROBLEM.
CD 2...UPDATE INTEGRALS AND SOLVE BOTH A DIRECT AND
CD ADJOINT EIGENVALUE PROBLEM.
CD 3...UPDATE INTEGRALS AND SOLVE AN ADJOINT EIGENVALUE
CD PROBLEM.
CD 19-24 MAXTTR, MAXIMUM NUMBER OF EIGENVALUE ITERATIONS
CD ALLOWED (DEFAULT=10).
CD 25-30 IBCXL, BOUNDARY CONDITION FOR THE LOWER "X" BOUNDARY
CD (X=0.) OF THE THREE-DIMENSIONAL MODEL.
CD 31-36 IBCXU, BOUNDARY CONDITION FOR THE UPPER "X" BOUNDARY
CD OF THE THREE-DIMENSIONAL MODEL.
CD 37-42 IBCET, BOUNDARY CONDITION FOR THE LOWER "Y" BOUNDARY
CD OF THE THREE-DIMENSIONAL MODEL.
CD 43-48 IBCEU, BOUNDARY CONDITION FOR THE UPPER "Y" BOUNDARY
CD OF THE THREE-DIMENSIONAL MODEL.
CD 49-54 IPCZL, BOUNDARY CONDITION FOR THE LOWER "Z" BOUNDARY
CD OF THE THREE-DIMENSIONAL MODEL.
CD 55-60 IPCZU, BOUNDARY CONDITION FOR THE UPPER "Z" BOUNDARY
CD OF THE THREE-DIMENSIONAL MODEL.
CD THE POSSIBLE BOUNDARY CONDITIONS ARE:
CD 1...ZERO FLUX.
CD 2...REFLECTIVE.
CD 3...EXTRAPOLATED (C*D*DEL PHI + PHI = 0).
CD THE CONSTANTS C FOR THE EXTRAPOLATED BOUNDARY
CD CONDITION ARE SPECIFIED CN CARD TYPE 06.
CD 4...PERIODIC WITH OPPOSITE BOUNDARY.
CD 5...PERIODIC WITH NEXT BOUNDARY GOING IN THE ORDER
CD LOWER X, UPPER Y, UPPERX, LOWER Y.
CD 6...PERIODIC WITH NEXT BOUNDARY GOING IN THE ORDER
CD LOWER X, UPPER Y, UPPERX, LOWER Y.
CD COUNTERWISE.
CD 7...PERIODIC, INVERTED ALONG THE SAME BOUNDARY.
CD PERTURBATION CONDITIONS (CONDITIONS 4 THROUGH 7) CAN ONLY
CD APPLY TO "X" AND "Y" BOUNDARIES.
APPENDIX D. INPUT DATA SET SYN3D. (CONT'D.)

CD 61-66 LGROUP, THE NUMBER OF ENERGY GROUPS AFTER GROUP COLLAPSING. THIS NUMBER MUST BE PROVIDED WHEN A GENERAL-GROUP COLLAPSING SCHEME IS EMPLOYED (CARD TYPES 11 AND 12 PRESENT IN INPUT), OTHERWISE, MAY BE IGNORED.

CD 67-72 IEDIT, EDIT OPTIONS FOR HMG4C OVERLAY. IGNORED IN THE ARC SYSTEM SYN3D.

0...NO EDIT DESIRED.
1...NORMAL EDIT SHOWING CORE USED AND ELAPSED TIME.
2...EDIT 1 PLUS COMPLETE EDIT OF MACROSCOPIC CROSS SECTION FILE WRITTEN BY HMG4C (COMPXS).
3...EDIT 2 PLUS RUNNING EDIT OF ISOTXS (I.E. ONLY THAT DATA FROM ISOTXS WHICH IS ACTUALLY USED IS EDITED).
4...EDIT 3 PLUS RPINTER TRACE PRINTS.

WHEN THERE IS NO INPUT INTTOC FILE THE "X" AND "Y" BOUNDARY CONDITIONS ARE PICKED UP FROM THE FIRST GEOSTR FILE ON THE TYPE 05 CARD. WHEN THERE IS NO INPUT INTTOC FILE THE OLD BOUNDARY CONDITIONS ARE ASSUMED. THE DEFAULT "Z" BOUNDARY CONDITIONS ARE ZERO CURRENT.

IF THE "X" AND "Y" BOUNDARY CONDITIONS ARE EXTRAPOLATED THE CODE DOES NOT PICK UP THE CONSTANTS FROM THE GEOSTFILE CARD TYPE 06 IS REQUIRED.

FOR TWO-DIMENSIONAL (XY OR PZ) PROBLEMS THE SECOND DIMENSION IS TREATED AS THE "Z" DIMENSION. THE SECOND DIMENSION BOUNDARY CONDITIONS GO IN COLS. 49-54 AND 55-60.

C-------------------------------------------------------------------------
C-------------------------------------------------------------------------

C GENERAL PROBLEM CONSTANTS (TYPE 03)

FORMAT-----(12,10Y,5E12.5)

COLUMNS CONTENTS...IMPLICATIONS, IF ANY

1-2 03

13-24 GUESS, K (EIGENVALUE) ESTIMATE FOR "IELANDT ITERATION (DEFAULT=NC WITH LANDT ITERATION). THE BEST CHOICE IS A NUMBER SLIGHTLY HIGHER THAN THE EXPECTED K.

25-36 CONVRG, EIGENVALUE CONVERGENCE CRITERION (DEFAULT=1.2-5).

37-42 TOTAL POWER IN WATTS (DEFAULT = 1 WATT).
APPENDIX D. INPUT DATA SET SYNFIL. (CONT'D.)

CR

AXIAL MESH DESCRIPTION (TYPE 04)

C

CL

FORMAT----(I2,10X,3(I6,E12.5))

C

CD

COLUMNS CONTENTS...IMPLICATIONS, IF ANY

CD

1-2 04

CD

13-18 N, NUMBER OF INTERVALS INTO WHICH A PORTION OF THE "Z" AXIS IS TO BE DIVIDED (DEFAULT=2).

CD

19-30 Z, UPPER COORDINATE OF THAT PORTION (DEFAULT=.5CM EACH).

CD

31-36 N, NUMBER OF INTERVALS INTO WHICH A PORTION OF THE "Z" AXIS IS TO BE DIVIDED.

CD

37-48 Z, UPPER COORDINATE OF THAT PORTION.

CD

49-54 N, NUMBER OF INTERVALS INTO WHICH A PORTION OF THE "Z" AXIS IS TO BE DIVIDED.

CD

55-66 Z, UPPER COORDINATE OF THAT PORTION.

C

THE AXIAL MESH DESCRIPTION STARTS AT Z=0.

CN

AS MANY TYPE 04 CARDS ARE USED AS ARE NECESSARY TO SPECIFY THE AXIAL MESH DESCRIPTION.

CN

WHEN THERE ARE NO TYPE 04 OR 05 CARDS THE CODE WILL EXPECT TO FIND A GEOMETRY FILE (GEODST,1) FROM WHICH TO EXTRACT THE DATA.

C

AXIAL GEOMETRY DESCRIPTION (TYPE 05)

C

CL

FORMAT----(I2,10X,16,6X,4E12.5)

C

CD

COLUMNS CONTENTS...IMPLICATIONS, IF ANY

CD

1-2 05

CD

13-18 IV, THE VERSION NUMBER FOR A GEODST FILE DESCRIBING A TWO-DIMENSIONAL GEOMETRY.

CD

25-36 ZBOT, THE LOWER AXIAL LIMIT OF AN AXIAL ZONE CHARACTERIZED BY THE TWO-DIMENSIONAL GEOMETRY IN GEODST, IV.".

CD

37-48 ZTOP, THE UPPER AXIAL LIMIT OF THE SAME ZONE.
APPENDIX D. INPUT DATA SET SYMPL. (CONT'D.)

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>49-60</td>
<td>ZBOT, the lower axial limit of an axial zone characterized by the two-dimensional geometry in GEOST, IVER.</td>
</tr>
<tr>
<td>61-72</td>
<td>ZTOP, the upper axial limit of the same zone.</td>
</tr>
<tr>
<td></td>
<td>As many TYPE 05 cards are used as are necessary to specify the axial geometry description.</td>
</tr>
<tr>
<td></td>
<td>When there are no TYPE 04 or 05 cards the code will expect to find a geometry file (GEOST,1) from which to extract the data.</td>
</tr>
<tr>
<td></td>
<td>The user may overlay geometries in building a model.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Columns</th>
<th>CONTENTS...IMPLICATIONS, IF ANY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>B, the boundary designator identifying the boundary of one of the three dimensions.</td>
</tr>
<tr>
<td>11-12</td>
<td>X, Y, Z lower.</td>
</tr>
<tr>
<td></td>
<td>X, Y, Z upper.</td>
</tr>
<tr>
<td></td>
<td>Y, Z lower.</td>
</tr>
<tr>
<td></td>
<td>Y, Z upper.</td>
</tr>
<tr>
<td></td>
<td>Zlower.</td>
</tr>
<tr>
<td></td>
<td>Zupper.</td>
</tr>
<tr>
<td>13-19</td>
<td>IGH, the highest energy group which uses the constant C.</td>
</tr>
<tr>
<td>19-24</td>
<td>IGLO, the lowest energy group which uses the constant C.</td>
</tr>
<tr>
<td>25-36</td>
<td>C, the constant for IRC=3 on the type 07 card.</td>
</tr>
<tr>
<td>37-42</td>
<td>IGH, the highest energy group which uses the constant C.</td>
</tr>
<tr>
<td>43-44</td>
<td>IGLO, the lowest energy group which uses the constant C.</td>
</tr>
</tbody>
</table>
Appendix D. Input Data Set Summary (Contd.)

49-60 C, THE CONSTANT FOR IRC=3 ON THE TYPE 02 CARD

AS MANY TYPE 06 CARDS ARE USED AS ARE NECESSARY TO
SPECIFY THE REQUIRED VALUES FOR THE CONSTANT C.

WHEN AN EXTRAPOLATED BOUNDARY CONDITION IS SPECIFIED
ON THE TYPE 2 CARD, AND THE CORRESPONDING CONSTANTS
ARE NOT SPECIFIED ON A TYPE 6 CARD, A DEFAULT VALUE
OF C=2.13 IS USED (FOR THE PARTICULAR BOUNDARY AND
GROUP RANGE NOT SPECIFIED).

EXPANSION FUNCTION SPECIFICATION (TYPE 07)

FORMAT-----(12,6X,5X,6X,5X,5X,6X,5X,6X,5X,6X,5X,5X,5X,5X,5X)

COLUMNS CONTENTS...IMPLICATIONS, IF ANY
1-2 07
7-12 TITFNP, FLUX FILE NAME. TITFNP MAY BE "UNIT" WHICH
IMPLIEDLY A UNIT FLUX IN ALL GROUPS EVEN THOUGH SUCH A
FILE DOES NOT ACTUALLY EXIST. "UNIT" MUST BE ENTERED
IN COLS. 7-10.
13-18 IVER, FLUX FILE VERSION NUMBER. IN THE CASE OF A UNIT-
FLUX IVER IS UNNECESSARY.
25-30 ZBOT, THE LOWER LIMIT OF AN AXIAL ZONE WHERE THE
FUNCTION IS USED.
31-44 ZTOP, THE UPPER LIMIT OF THAT ZONE.
49-60 ZBOT, THE LOWER LIMIT OF AN AXIAL ZONE WHERE THE
FUNCTION IS USED.
61-72 ZTOP, THE UPPER LIMIT OF THAT ZONE.
(DEFAULT: A "UNIT FLUX WILL BE USED EVERYWHERE AS THE
ONLY EXPANSION FUNCTION).

AS MANY TYPE 07 CARDS ARE USED AS ARE NECESSARY TO
SPECIFY THE EXPANSION FUNCTION DATA.
APPENDIX D. INPUT DATA SET SYNFIL. (CONT'D.)

WEIGHTING FUNCTION SPECIFICATION (TYPE 09)

FORMAT-----(I2,4X,A6,I6,6X,4E12.5)

<table>
<thead>
<tr>
<th>COLUMNS</th>
<th>CONTENTS...IMPLICATIONS, IF ANY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>08</td>
</tr>
<tr>
<td>7-12</td>
<td>TITFN, FLUX FILE NAME. TITFN MAY BE &quot;UNIT&quot; WHICH IMPLIES A UNIT FLUX IN ALL GROUPS EVEN THOUGH SUCH A FILE DOES NOT ACTUALLY EXIST. &quot;UNIT&quot; MUST BE ENTERED IN COLS. 7-10.</td>
</tr>
<tr>
<td>13-18</td>
<td>IVFR, FLUX FILE VERSION NUMBER. IN THE CASE OF A FLUX IVFR IS UNNECESSARY.</td>
</tr>
<tr>
<td>25-36</td>
<td>ZBOT, THE LOWER LIMIT OF AN AXIAL ZONE WHERE THE FUNCTION IS USED.</td>
</tr>
<tr>
<td>37-48</td>
<td>ZTOP, THE UPPER LIMIT OF THAT ZONE.</td>
</tr>
<tr>
<td>49-60</td>
<td>ZBOT, THE LOWER LIMIT OF AN AXIAL ZONE WHERE THE FUNCTION IS USED.</td>
</tr>
<tr>
<td>61-72</td>
<td>ZTOP, THE UPPER LIMIT OF THAT ZONE.</td>
</tr>
</tbody>
</table>

(DEFAULT = IF NO TYPE 08 CARDS ARE PROVIDED THE EXPANSION FUNCTIONS WILL BE USED AS WEIGHTING FUNCTIONS). |

AS MANY TYPE 08 CARDS ARE USED AS ARE NECESSARY TO SPECIFY THE WEIGHTING FUNCTION DATA.)
### SPECIAL GROUP COLLAPSING SCHEME FOR EXPANSION FUNCTION (TYPE 09)

**FORMAT**: (12, 41, 1116)

<table>
<thead>
<tr>
<th>COLUMN</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>00</td>
</tr>
<tr>
<td>7-12</td>
<td>LGP, LOWEST ENERGY GROUP IN COLLAPSED GROUP 1.</td>
</tr>
<tr>
<td>13-14</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1.</td>
</tr>
<tr>
<td>19-21</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+2.</td>
</tr>
<tr>
<td>25-29</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+3.</td>
</tr>
<tr>
<td>31-36</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+4.</td>
</tr>
<tr>
<td>47-48</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+6.</td>
</tr>
<tr>
<td>49-53</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+7.</td>
</tr>
<tr>
<td>55-60</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+8.</td>
</tr>
<tr>
<td>61-66</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+9.</td>
</tr>
<tr>
<td>67-72</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+10.</td>
</tr>
</tbody>
</table>

(DEFAULT = NO GROUP COLLAPSING).

**AS MANY TYPE 09 CARDS MAY BE USED AS NECESSARY TO SWEET THE NUMBER OF COLLAPSED GROUPS. THE TYPE 09 CARDS MUST BE IN ORDER, STARTING WITH THE HIGHEST ENERGY GROUP. THE LAST GROUP NUMBER BY THE LAST CARD MUST BE EQUAL TO THE NUMBER OF GROUPS AFFECTED GROUP COLLAPSING.**
APPENDIX D.  INPUT DATA SET SYMFIL.  (CONT'D.)

<table>
<thead>
<tr>
<th>COLUMNS</th>
<th>CONTENTS...IMPLICATIONS, IF ANY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>10</td>
</tr>
<tr>
<td>7-12</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP I.</td>
</tr>
<tr>
<td>13-18</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+1.</td>
</tr>
<tr>
<td>19-24</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+2.</td>
</tr>
<tr>
<td>25-30</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+3.</td>
</tr>
<tr>
<td>31-36</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+4.</td>
</tr>
<tr>
<td>37-42</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+5.</td>
</tr>
<tr>
<td>43-48</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+6.</td>
</tr>
<tr>
<td>49-54</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+7.</td>
</tr>
<tr>
<td>55-60</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+8.</td>
</tr>
<tr>
<td>61-66</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+9.</td>
</tr>
<tr>
<td>67-72</td>
<td>LGP, LOWEST ENERGY GROUP NUMBER IN COLLAPSED GROUP 1+10.</td>
</tr>
</tbody>
</table>

(DEFAULT = NO GROUP COLLAPSING IF NO TYPE 09 CARD.
IF TYPE 09 CARD IS PRESENT, BUT NO TYPE 10 CARD, THE
EXPANSION FUNCTION COLLAPSING SCHEME IS REPEATED FOR
THE WEIGHTING FUNCTION.)

AS MANY TYPE 10 CARDS MAY BE USED AS ARE NECESSARY TO
SUPPLEMENT THE NUMBER OF COLLAPSED GROUPS. THE TYPE 10 CARDS
MUST BE IN ORDER, STARTING WITH THE HIGHEST ENERGY
GROUP. THE LAST GROUP NUMBER ON THE LAST CARD MUST BE
EQUAL TO THE NUMBER OF GROUPS BEFORE GROUP COLLAPSING.)
APPENDIX D. INPUT DATA SET SYNFIL. (CONT'D.)

C-------------"-----------------------------------------------------------
C7 F-INNAT GROUP
C7 COLLAPSING SCHEME FOR EXPANSION FUNCTION
C7 (TYPE 11)
C
C FORMAT------(T2,19X,5E12.4)
C
C COLUMNS CONTENTS...IMPLICATIONS, IF ANY
C
C 1-2 11
C
C 13-16 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 17-20 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 21-24 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 25-28 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 29-32 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 33-36 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 37-40 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 41-44 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 45-48 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 49-52 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 53-56 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C 57-60 M(I,J), GROUP COLLAPSING MATRIX FOR EXPANSION FUNCTION.
C
C EXPANSION FUNCTION DATA IS SPECIFIED IN THE ORDER
C (((I,J),1=1,NGROUP,1=1,NGROUP), " BEING EQUAL TO THE
C GROUP COLLAPSING MATRIX, IGROUP BEING EQUAL TO THE
C NUMBER OF EMPTY GROUPS AFTER GROUP COLLAPSING AND
C NGROUP BEING EQUAL TO THE NUMBER OF EMPTY GROUPS
C BEFORE GROUP COLLAPSING.
C
C AS MANY TYPE 11 CARDS ARE USED AS ARE NECESSARY TO
C SPECIFY THE EXPANSION FUNCTION DATA.
APPENDIX C. INPUT DATA SET SYMPI. (CONT'D.)

C-------------------------------
C GENERAL GROUP COLLAPSING SCHEME FOR WEIGHTING FUNCTION
C (TYPE 12)
C
C FORMAT-----(I2,10X,5E12.4)
C
C COLUMNS CONTENTS...IMPLICATIONS, IF ANY
C
1-2  12

13-24  U(I,J), GROUP COLLAPSING MATRIX FOR WEIGHTING FUNCTION.

25-36  U(I,J), GROUP COLLAPSING MATRIX FOR WEIGHTING FUNCTION.

37-49  M(I,J), GROUP COLLAPSING MATRIX FOR WEIGHTING FUNCTION.

49-60  U(J,I), GROUP COLLAPSING MATRIX FOR WEIGHTING FUNCTION.

61-72  M(I,J), GROUP COLLAPSING MATRIX FOR WEIGHTING FUNCTION.

(C-LIST OF STRIPED TERMS

(D-E'AL'T = NO GROUP COLLAPSING IF NO TYPE 11 CARD.

IF TYPE 11 CARD IS PRESENT, BUT NO TYPE 12 CARD, THE

EXPANSION FUNCTION COLLAPSING SCHEME IS APPLIED FOR

THE WEIGHTING FUNCTION).

WEIGHTING FUNCTION DATA IS SPECIFIED IN THE ORDER

((U(I,J), I=1, LGROUP, J=1, NGROUP), U BEING EQUAL TO THE

GROUP COLLAPSING MATRIX, LGROUP BEING EQUAL TO THE

NUMBER OF ENERGY GROUPS AFTER GROUP COLLAPSING AND

NGROUP BEING EQUAL TO THE NUMBER OF ENERGY GROUPS

BEFORE GROUP COLLAPSING.

AS MANY TYPE 12 CARDS ARE USED AS ARE NECESSARY TO

SPECIFY THE WEIGHTING FUNCTION DATA.

C-------------------------------
**APPENDIX D. INPUT DATA SRT SYNFIL. (CONT'D.)**

GROUP FLUX ELIMINATION FOR INPUT FUNCTIONS (TYPE 13)

<table>
<thead>
<tr>
<th>CR</th>
<th>CL</th>
<th>CD</th>
</tr>
</thead>
<tbody>
<tr>
<td>(I2,4X,A6,9I6)</td>
<td>COLUMNS</td>
<td>CONTENTS...IMPlications, IF ANY</td>
</tr>
<tr>
<td>1-2</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>7-12</td>
<td>TITPUN, FLUX FILE NAMP. TITPUN MAY BE &quot;UNIT&quot; WHICH MUST BE ENTERED IN COIS. 7-10.</td>
<td></td>
</tr>
<tr>
<td>13-19</td>
<td>IVER, FLUX FILE VERSION NUMBER.</td>
<td></td>
</tr>
<tr>
<td>19-24</td>
<td>IGHl, HIGHEST ENERGY GROUP OF A SEQUENCE OF GROUP FLUXES (AFTER COLLAPSING, IF ANY) TO BE EXCLUDED FROM THE SYNTHESIS CALCULATION.</td>
<td></td>
</tr>
<tr>
<td>25-30</td>
<td>IGLO, LOWEST ENERGY GROUP OF THE SEQUENCE.</td>
<td></td>
</tr>
<tr>
<td>31-36</td>
<td>IGHl, HIGHEST ENERGY GROUP OF A SEQUENCE OF GROUP FLUXES (AFTER COLLAPSING, IF ANY) TO BE EXCLUDED FROM THE SYNTHESIS CALCULATION.</td>
<td></td>
</tr>
<tr>
<td>37-42</td>
<td>IGLO, LOWEST ENERGY GROUP OF THE SEQUENCE.</td>
<td></td>
</tr>
<tr>
<td>43-48</td>
<td>IGHl, HIGHEST ENERGY GROUP OF A SEQUENCE OF GROUP FLUXES (AFTER COLLAPSING, IF ANY) TO BE EXCLUDED FROM THE SYNTHESIS CALCULATION.</td>
<td></td>
</tr>
<tr>
<td>49-54</td>
<td>IGLO, LOWEST ENERGY GROUP OF THE SEQUENCE.</td>
<td></td>
</tr>
<tr>
<td>55-60</td>
<td>IGHl, HIGHEST ENERGY GROUP OF A SEQUENCE OF GROUP FLUXES (AFTER COLLAPSING, IF ANY) TO BE EXCLUDED FROM THE SYNTHESIS CALCULATION.</td>
<td></td>
</tr>
<tr>
<td>61-66</td>
<td>IGLO, LOWEST ENERGY GROUP OF THE SEQUENCE.</td>
<td></td>
</tr>
<tr>
<td>(DEFAULT - ALL GROUP FLUXES ARE USED).</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

AS MANY TYPE 13 CARDS ARE USED AS ARE NECESSARY TO SPECIFY THE ENERGY GROUPS TO BE EXCLUDED FROM THE SYNTHESIS CALCULATION.
APPENDIX D. INPUT DATA SET SYNPIF. (CONT.)

C-------------------------------------------------------------
C EXPANSION FUNCTION SCALING FACTORS (TYPE 1G)
C-------------------------------------------------------------
C
C FORMAT---(12,1X,A6,16,2(216,E12.4))
C
C COLUMNS CONTENTS...IMPLICATIONS, IF ANY
C 1-2 14
C 7-12 TITFUN, FLUX FILE NAME. TITFUN MAY BE "UNIT" WHICH
C MUST BE ENTERED IN COLS. 7-10.
C 13-19 IVFR, FLUX FILE VERSION NUMBER.
C 19-24 IGLO, THE FIRST ENERGY GROUP OF A SEQUENCE OF GROUP
C Fluxes to be scaled by ESCLAE when used as expansion
C functions.
C 25-30 TGH1, THE LAST ENERGY GROUP OF THAT SEQUENCE.
C 31-42 ESCLAE, THE EXPANSION FUNCTION SCALING FACTOR
C (DEFAULT = 1.0).
C 43-49 IGLO, THE FIRST ENERGY GROUP OF A SEQUENCE OF GROUP
C Fluxes to be scaled by ESCLAE when used as expansion
C functions.
C 49-54 TGH1, THE LAST ENERGY GROUP OF THAT SEQUENCE.
C 55-66 ESCLAE, THE EXPANSION FUNCTION SCALING FACTOR
C (DEFAULT = 1.0).
C
C IF THE FLUX FILE VERSION NUMBER IS BLANK, THE LAST
C NAMED FLUX FILE IS AUTOMATICALLY ASSUMED.
C
C AS MANY TYPE 1G CARDS ARE USED AS ARE NECESSARY TO
C SPECIFY THE REQUIRED SCALING FACTOR DATA.
C
C-------------------------------------------------------------
APPENDIX D. INPUT DATA SET SYMFIL. (CONT'D.)

WEIGHTING FUNCTION SCALING FACTORS (TYPE 15)

FORMAT---(12,4X,A6,16,2(216,E12.4))

COLUMNS

CONTENTS...IMPLICATIONS, IF ANY

1-2 15

7-12 TITFUN, FLUX FILE NAME. TITFUN MAY BE "UNIT" WHICH MUST BE ENTERED IN COLS. 7-10.

13-18 PVC, FLUX FILE VERSION NUMBER.

19-24 IGLO, THE FIRST ENERGY GROUP OF A SEQUENCE OF GROUP FLUXES TO BE SCALCED BY ESCELE WHEN USED AS WEIGHTING FUNCTIONS.

25-30 IGH1, THE LAST ENERGY GROUP OF THAT SEQUENCE.

31-42 ESCLL, THE WEIGHTING FUNCTION SCALING FACTOR.

DEFNULT = 1.0.

43-44 IGLO, THE FIRST ENERGY GROUP OF A SEQUENCE OF GROUP FLUXES TO BE SCALCED BY ESCELE WHEN USED AS WEIGHTING FUNCTIONS.

49-54 IGH1, THE LAST ENERGY GROUP OF THAT SEQUENCE.

55-66 ESCLL, THE WEIGHTING FUNCTION SCALING FACTOR.

DEFNULT = 1.0.

IF THE FLUX FILE VERSION NUMBER IS BLANK, THE LAST NAMED FLUX FILE IS AUTOMATICALLY ASSUMED.

AS MANY TYPE 15 CARDS ARE USED AS APP NECESSARY TO SPECIFY THE REQUIRED SCALING FACTOR DATA.
### SYNTHESIS OUTPUT OPTIONS (TYPE 16)

<table>
<thead>
<tr>
<th>COLUMNS</th>
<th>CONTENTS...IMPLICATIONS, IF ANY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>16</td>
</tr>
<tr>
<td>7-12</td>
<td>NDENOM, PERTURBATION DENOMINATOR OPTION.</td>
</tr>
<tr>
<td>0...</td>
<td>DO NOT CALCULATE PERTURBATION DENOMINATOR (DEFAULT).</td>
</tr>
<tr>
<td>1...</td>
<td>CALCULATE PERTURBATION DENOMINATOR.</td>
</tr>
<tr>
<td>13-18</td>
<td>NDFLUX, OUTPUT RTFLUX OPTION.</td>
</tr>
<tr>
<td>0...</td>
<td>DO NOT CONSTRUCT AN OUTPUT RTFLUX FILE (DEFAULT).</td>
</tr>
<tr>
<td>.GT.0</td>
<td>THE VERSION NUMBER FOR THE OUTPUT RTFLUX FILE. BE CAREFUL NOT TO ACCIDENTALLY DESTROY AN INPUT RTFLUX FILE.</td>
</tr>
<tr>
<td>19-24</td>
<td>NDADJ, OUTPUT ATFLUX OPTION.</td>
</tr>
<tr>
<td>0...</td>
<td>DO NOT CONSTRUCT AN OUTPUT ATFLUX FILE (DEFAULT).</td>
</tr>
<tr>
<td>.GT.0</td>
<td>THE VERSION NUMBER FOR THE OUTPUT ATFLUX FILE. BE CAREFUL NOT TO ACCIDENTALLY DESTROY AN INPUT ATFLUX FILE.</td>
</tr>
<tr>
<td>25-30</td>
<td>NPDINT, OUTPUT PDINT OPTION.</td>
</tr>
<tr>
<td>0...</td>
<td>DO NOT CONSTRUCT AN OUTPUT PDINT FILE (DEFAULT).</td>
</tr>
<tr>
<td>.GT.0</td>
<td>THE VERSION NUMBER FOR THE OUTPUT PDINT FILE.</td>
</tr>
<tr>
<td>31-36</td>
<td>NDZFLX, OUTPUT RZFLUX OPTION.</td>
</tr>
<tr>
<td>0...</td>
<td>DO NOT CONSTRUCT AN OUTPUT RZFLUX (DEFAULT).</td>
</tr>
<tr>
<td>.GT.0</td>
<td>THE VERSION NUMBER FOR THE OUTPUT RZFLUX FILE.</td>
</tr>
<tr>
<td>37-42</td>
<td>NGEOST, OUTPUT GEOST OPTION.</td>
</tr>
<tr>
<td>0...</td>
<td>DO NOT CONSTRUCT AN OUTPUT GEOST FILE (DEFAULT).</td>
</tr>
<tr>
<td>.GT.0</td>
<td>THE VERSION NUMBER FOR THE OUTPUT GEOST FILE. BE CAREFUL NOT TO ACCIDENTALLY DESTROY AN INPUT GEOST FILE.</td>
</tr>
<tr>
<td>43-49</td>
<td>NPREZF, FLUX INTEGRAL EDIT OPTION.</td>
</tr>
<tr>
<td>0...</td>
<td>DO NOT EDIT THE AVERAGE FLUXES BY ZONE (DEFAULT).</td>
</tr>
<tr>
<td>.GT.0</td>
<td>EDIT THE AVERAGE FLUXES.</td>
</tr>
</tbody>
</table>
### Appendix D. Input Data Set SYNPI. (Contd.)

**Direct Combining Coefficient Plots (Type 17)**

<table>
<thead>
<tr>
<th>Columns</th>
<th>Contents...Implications, If Any</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-2</td>
<td>17</td>
</tr>
<tr>
<td>7-12</td>
<td>TPLDCC(1), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>13-19</td>
<td>TPLDCC(2), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>19-20</td>
<td>TPLDCC(3), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>25-30</td>
<td>TPLDCC(4), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>31-36</td>
<td>TPLDCC(5), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>37-42</td>
<td>TPLDCC(6), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>43-48</td>
<td>TPLDCC(7), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>49-54</td>
<td>TPLDCC(8), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>55-60</td>
<td>TPLDCC(9), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>61-66</td>
<td>TPLDCC(10), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
<tr>
<td>67-72</td>
<td>TPLDCC(11), an energy group (after group collapsing, if any) for which the modal components of the total axial direct flux are to be printer-plotted.</td>
</tr>
</tbody>
</table>
APPENDIX D. INPUT DATA SET SYNFIL. (CONTD.)

CD
CD IF NO TYPE 17 CARDS ARE PRESENT, THERE WILL BE NO
CD PLOTS (DEFAULT).
CD
CN DIRECT COMBINING COEFFICIENT PLOTS DATA IS SPECIFIED
CN SUCH THAT (IPLDCC(I), I=1,NPLDCC), NPLDCC BEING THE
CN NUMBER OF GROUPS FOR WHICH PLOTS ARE TO BE OUTPUT
CN (NPLDCC.LF.LGROUP).
CN
CN AS MANY TYPE 17 CARDS ARE USED AS ARE NECESSARY TO
CN SPECIFY THE DATA FOR THE REQUIRED PLOTS.
C
C-----------------------------------------------------------------------
C----------------------------------------------------------------------
CR ADJOINT COMBINING COEFFICIENTS PLOTS (TYPE 18)
C
CL FORMAT----(I2,4X,11T6)
C
CD COLUMNS CONTENTS...IMPLICATIONS, IF ANY
CD =========== =========================================================
CD 1-2 18
CD 7-12 IPLACC(1), AN ENERGY GROUP (AFTER GROUP COLLAPSING, IF
CD ANY) FOR WHICH THE MODAL COMPONENTS OF THE TOTAL AXIAL
CD ADJOINT FLUX ARE TO BE PRINTER-PLOTTED.
CD 13-19 IPLACC(2), AN ENERGY GROUP (AFTER GROUP COLLAPSING, IF
CD ANY) FOR WHICH THE MODAL COMPONENTS OF THE TOTAL AXIAL
CD ADJOINT FLUX ARE TO BE PRINTER-PLOTTED.
CD 19-24 IPLACC(3), AN ENERGY GROUP (AFTER GROUP COLLAPSING, IF
CD ANY) FOR WHICH THE MODAL COMPONENTS OF THE TOTAL AXIAL
CD ADJOINT FLUX ARE TO BE PRINTER-PLOTTED.
CD 25-30 IPLACC(4), AN ENERGY GROUP (AFTER GROUP COLLAPSING, IF
CD ANY) FOR WHICH THE MODAL COMPONENTS OF THE TOTAL AXIAL
CD ADJOINT FLUX ARE TO BE PRINTER-PLOTTED.
CD 31-36 IPLACC(5), AN ENERGY GROUP (AFTER GROUP COLLAPSING, IF
CD ANY) FOR WHICH THE MODAL COMPONENTS OF THE TOTAL AXIAL
CD ADJOINT FLUX ARE TO BE PRINTER-PLOTTED.
CD 37-42 IPLACC(6), AN ENERGY GROUP (AFTER GROUP COLLAPSING, IF
CD ANY) FOR WHICH THE MODAL COMPONENTS OF THE TOTAL AXIAL
CD ADJOINT FLUX ARE TO BE PRINTER-PLOTTED.
CD 43-49 IPLACC(7), AN ENERGY GROUP (AFTER GROUP COLLAPSING, IF
CD ANY) FOR WHICH THE MODAL COMPONENTS OF THE TOTAL AXIAL
CD ADJOINT FLUX ARE TO BE PRINTER-PLOTTED.
APPENDIX D. INPUT DATA SET SYNFIL. (CONTD.)

IPLACC(8), an energy group (after group collapsing, if any) for which the modal components of the total axial adjoint flux are to be printer-plotted.

IPLACC(9), an energy group (after group collapsing, if any) for which the modal components of the total axial adjoint flux are to be printer-plotted.

IPLACC(10), an energy group (after group collapsing, if any) for which the modal components of the total axial adjoint flux are to be printer-plotted.

IPLACC(11), an energy group (after group collapsing, if any) for which the modal components of the total axial adjoint flux are to be printer-plotted.

If no type 1P cards are present, there will be no plots (default).

Adjoint combining coefficient plots data is specified such that (IPLACC(I), I=1,NPLACC), NPLACC being the number of groups for which plots are to be output (NPLACC<=LGROUP).

As many type 1B cards are used as are necessary to specify the data for the required plots.
APPENDIX D. INPUT DATA SET SYNPI.  (CONT.)

C-----------------------------------------------------------------------
C
C TWO-DIMENSIONAL DIRECT FLUX EDITS (TYPE 19)
C
C FORMAT-----(I2,4X,10I6)
C
C COLUMNS CONTENTS...IMPLICATIONS, IF ANY
C
C 1-2 19
C
C 7-12 IPRRTP(1,1), A GROUP NUMBER USED IN IDENTIFYING A PLANE OF DIRECT FLUXES TO BE EDITED.
C
C 13-18 IPRRTP(2,1), AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLANE OF DIRECT FLUXES TO BE EDITED.
C
C 19-24 IPRRTP(1,2), A GROUP NUMBER USED IN IDENTIFYING A PLAN CF DIRECT FLUXES TO BE EDITED.
C
C 25-30 IPRRTP(2,7), AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLANE OF DIRECT FLUXES TO BE EDITED.
C
C 31-36 IPRRTP(1,3), A GROUP NUMBER USED IN IDENTIFYING A PLAN CF DIRECT FLUXES TO BE EDITED.
C
C 37-42 IPRRTP(2,3), AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLAN CF DIRECT FLUXES TO BE EDITED.
C
C 43-48 IPRRTP(1,4), A GROUP NUMBER USED IN IDENTIFYING A PLAN CF DIRECT FLUXES TO BE EDITED.
C
C 49-54 IPRRTP(2,4), AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLAN CF DIRECT FLUXES TO BE EDITED.
C
C 55-60 IPRRTP(1,5), A GROUP NUMBER USED IN IDENTIFYING A PLAN CF DIRECT FLUXES TO BE EDITED.
C
C 61-66 IPRRTP(2,5), AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLAN CF DIRECT FLUXES TO BE EDITED.
C
C (DEFAULT = NO EDITS).
C
C THE TWO-DIMENSIONAL DIRECT FLUX EDIT DATA IS SPECIFIED SUCH THAT (IPRRTP(I,N),I=1,2),N=1,NPPRTP), NPPRTP BEING
C THE NUMBER OF PLAN CF EDITS TO BE DONE, WHEN THE PROBLEM IS TWO-DIMENSIONAL ALL AXIAL MESH INTERVALS ARE EDITED WHEN IPRRTP(2,N).GT.0.
C
C AS MANY TYPE 19 CARDS ARE USED AS ARE NECESSARY TO SPECIFY THE DATA REQUIRED FOR THE DIRECT FLUX EDITS.

C-----------------------------------------------------------------------
APPENDIX D. INPUT DATA SET SYNFIL. (CONT'D.)

**TWO-DIMENSIONAL ADJOINT FLUX EDITS (TYPE 20)**

**FORMAT**: *(12,4X,10I6)*

**COLUMNS**

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>I</strong>PRATF**(1,1)**, A GROUP NUMBER USED IN IDENTIFYING A PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td>20</td>
<td>IPRATF**(2,1)**, AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td>IPRATF**(1,2)**, A GROUP NUMBER USED IN IDENTIFYING A PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td>IPRATF**(2,2)**, AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td>IPRATF**(1,3)**, A GROUP NUMBER USED IN IDENTIFYING A PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td>IPRATF**(2,3)**, AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td>IPRATF**(1,4)**, A GROUP NUMBER USED IN IDENTIFYING A PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td>IPRATF**(2,4)**, AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td>IPRATF**(1,5)**, A GROUP NUMBER USED IN IDENTIFYING A PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td>IPRATF**(2,5)**, AN AXIAL MESH INTERVAL USED IN IDENTIFYING THE PLANE OF ADJOINT FLUXES TO BE EDITED.</td>
<td></td>
</tr>
<tr>
<td><strong>(DEFAULT = NO EDITS)</strong>.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The two-dimensional adjoint flux edit data is specified such that if **I**PRATF**(1,N), I=1,2, N=1,NPRAT, **I**PRATF being the number of planar edits to be done. When the problem is two-dimensional all axial mesh intervals are edited when **I**PRATF**(2,N).GT.0. As many type 20 cards are used as are necessary to specify the data required for the adjoint flux edits.
APPENDIX D. INPUT DATA SET SYNFIL. (CONT'D.)

TWO-DIMENSIONAL PLANAR POWER EDITS (TYPE 21)

COLUMN CONTENTS...IMPLICATIONS, IF ANY

1-2 21

7-12 IPRPPWD(1), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

13-18 IPRPPWD(2), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

19-24 IPRPPWD(3), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

25-30 IPRPPWD(4), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

31-36 IPRPPWD(5), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

37-42 IPRPPWD(6), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

43-49 IPRPPWD(7), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

49-54 IPRPPWD(8), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

55-60 IPRPPWD(9), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

61-66 IPRPPWD(10), AN AXIAL MESH INTERVAL IDENTIFYING A PLANE OF POWER EDITS

(DEFAULT = NO EDITS).

THE TWO-DIMENSIONAL PLANAR POWER EDIT DATA IS SPECIFIED SUCH THAT (IPRPWD(I), I = 1, NRPWD), NRPWD BEING THE NUMBER OF PLANAR EDITS ENTERED, WHEN THE PROBLEM IS TWO-DIMENSIONAL ALL AXIAL MESH INTERVALS ARE EDITED IF IPRPPWD(1).GT.0.

AS MANY TYPE 21 CARDS ARE USED AS ARE NECESSARY TO SPECIFY THE DATA REQUIRED FOR THE PLANAR POWER EDITS.
APPENDIX E.  CODE-DEPENDENT BINARY FILES.  COMPXS.

PREPARED 9/26/75 AT ANL.

COMPXS
MACROSCOPIC COMPOSITION CROSS SECTIONS

FILE STRUCTURE

<table>
<thead>
<tr>
<th>RECORD TYPE</th>
<th>PRESENT IF</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPECIFICATIONS</td>
<td>ALWAYS</td>
</tr>
<tr>
<td>SPT CHI</td>
<td>ISCHI=ST.0</td>
</tr>
<tr>
<td>*(REPEAT FOR ALL COMPOSITIONS)</td>
<td></td>
</tr>
<tr>
<td>COMPOSITION SPECIFICATIONS</td>
<td>ALWAYS</td>
</tr>
<tr>
<td>*(REPEAT FOR ALL ENERGY GROUPS)</td>
<td></td>
</tr>
<tr>
<td>** IN THE ORDER OF DECREASING ENERGY</td>
<td></td>
</tr>
<tr>
<td>** COMPOSITION MACROSCOPIC GROUP</td>
<td>ALWAYS</td>
</tr>
<tr>
<td>** CROSS SECTIONS</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IGROUP</th>
<th>NUMBER OF ENERGY GROUPS.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICHI</td>
<td>PROMPT FISSION SPECTRUM FLAG FOR THIS COMPOSITION. ICHI=1 IF COMPOSITION USES THE SPF-WIDE PROMPT CHI GIVEN IN SPT CHI RECORD (BELOW). ICHI=0 IF COMPOSITION IS NOT FISSIONABLE, ICHI=1 FOR COMPOSITION PROMPT CHI VECTOR, ICHI=NGROUP FOR COMPOSITION PROMPT CHI MACRIT.</td>
</tr>
<tr>
<td>NGP</td>
<td>NUMBER OF GROUPS OF UPSCATTERING FOR THIS COMPOSITION.</td>
</tr>
<tr>
<td>NGN</td>
<td>NUMBER OF GROUPS OF DOWNSCATTERING FOR THIS COMPOSITION.</td>
</tr>
<tr>
<td>ISCHI</td>
<td>PROMPT FISSION SPECTRUM FLAG, ISCHI=0 IF THERE IS NO SET-WIDE PROMPT CHI, ISCHI=1 IF THERE IS A SET-WIDE PROMPT CHI VECTOR.</td>
</tr>
<tr>
<td>ISCHI-NGROUP</td>
<td>IF THERE IS A SET-WIDE PROMPT CHI MATRIX.</td>
</tr>
<tr>
<td>MALT</td>
<td>3 FOR IBM MACHINES, 1 OTHERWISE.</td>
</tr>
</tbody>
</table>
APPENDIX E.  CODE-DEPENDENT BINARY FILES (CONTD.). COMPS.

C----------------- SPECIFICATIONS (TYPE 1) -----------------
C
CL  NCMP,NGROUP,ISCHI,NPCMP,MAXUP,MAIDN,NDOM1,NDOM2,NDOM3,NDOM4
C
CV  10
C
CD  NCMP  NUMBER OF COMPOSITIONS.
CD  NPCMP  NUMBER OF PSSIONABLE COMPOSITIONS.
CD  MAXUP  MAXIMUM NUMBER OF GROUPS OF UPSCATTERING FOR
CD  MAIDN  MAXIMUM NUMBER OF GROUPS OF DOWNSCATTERING
CD  NDOM1  RESERVED.
CD  NDOM2  RESERVED.
CD  NDOM3  RESERVED.
CD  NDOM4  RESERVED.
C

C----------------- SET CHI (TYPE 2) -----------------
C
CC  PRESENT IF ISCHI.GT.0
C
CL  ((CHI(I,J),I=1,ISCHI),J=1,NGROUP)
C
CW  MULT*ISCHI*NGROUP
C
CD  CHI  PRECPT FISSION FRACTION INTO GROUP J FROM
CD  GROUP I. IF ISCHI=1, THE LIST REDUCES TO
CD  (CHI(J),J=1,NGROUP), WHERE CHI(J) IS THE
CD  FISSION FRACTION INTO GROUP J.
C

C----------------- COMPOSITION SPECIFICATIONS (TYPE 3) -----------------
C
CC  ALWAYS PRESENT
C
CL  (ISCHI,(NUP(I),I=1,NGROUP), (NDN(I),I=1,NGROUP)
C
CV  1+2*NGROUP
C
C
APPENDIX E.  CODE-DEPENDENT BINARY FILES (CONT'D.).  COMPXS.

--------------------
CR  COMPOSITION MACROSCOPIC GROUP CROSS SECTIONS (TYPE 4)
CC  ALWAYS PRESENT
C
CL  XA,XTOT,XXRM,XXR,XXF,XXN, (CHI(I),I=1,ICHI),
CL  (XSCATU(I),I=1,NUMUP),XSCATJ,(XSCATD(I),I=1,NUMDN),
CL  2PC,A1,R1,A2,B2,A3,B3
C
CC  NUMUP = NUP FOR THE CURRENT GROUP.
CC  NUMDN = NDN FOR THE CURRENT GROUP.
C
C4  MUL5*(14+TCHI+NUMUP+NUMDN) IF ICHI.GT.0
C7  MUL5*(14+NUMUP+NUMDN) IF ICHI.EQ.-1
CV  MUL5*(12+NUMUP+NUMDN) IF ICHI.EQ.0
C
CD  XA  ABSORPTION CROSS SECTION.
CD  XTOT  TOTAL CROSS SECTION.
CD  XXRM  REMOVAL CROSS SECTION, TOTAL CROSS SECTION
CD  FOR REMOVING A NEUTRON FROM GROUP J DUE TO ALL
CD  PROCESSES.
CD  XXR  TRANSPORT CROSS SECTION.
CD  XXF  FISSION CROSS SECTION, PRESENT ONLY IF
CD  ICHI.NE.0.
CD  XXN  TOTAL NUMBER OF NEUTRONS EMITTED PER FISSION
CD  TIMES XF, PRESENT ONLY IF ICHI.NE.0.
CD  CHI  PROMPT FISSION FRACTION INTO GROUP J FROM
CD  GROUP I, PRESENT ONLY IF ICHI.GT.0. IF ICHI=1,
CD  THE LIST REDUCES TO THE SINGLE NUMBER CHI,
CD  WHICH IS THE PROMPT FISSION FRACTION INTO
CD  GROUP J.
CD  XSCATU  TOTAL SCATTERING CROSS SECTION INTO GROUP J
CD  FROM GROUPS J+NUM(J),J+NUM(J)-1,...,J+2,J+1,
CD  PRESENT ONLY IF NUM(J).GT.0.
CD  XSCATJ  TOTAL SELF-SCATTERING CROSS SECTION FROM
CD  GROUP J TO GROUP J.
CD  XSCATD  TOTAL SCATTERING CROSS SECTION INTO GROUP J
CD  FROM GROUPS J-1,J-2,...,J-NDN(J), PRESENT
CD  ONLY IF NDN(J).GT.0.
CD  PC  PC TIMES THE GROUP J REGION INTEGRATED
CD  FLUX FOR THE REGIONS CONTAINING THE CURRENT
CD  COMPOSITION YIELDS THE POWER IN JATTS IN THOSE
CD  REGIONS AND ENERGY GROUP J DTP TO FISSIONS
CD  AND NON-FISSION ABSORPTIONS.
CD  A1  FIRST DIMENSION DIRECTIONAL DIFFUSION
CD  COEFFICIENT MULTIPLIER.
CD  B1  FIRST DIMENSION DIRECTIONAL DIFFUSION
CD  COEFFICIENT ADDITIVE TERM.
APPENDIX E. CODE-DEPENDENT BINARY FILES (CONTD.).

| CD | A2     | SECOND DIMENSION DIRECTIONAL DIFFUSION COEFFICIENT MULTIPLIER. |
| CD | A2     | SECOND DIMENSION DIRECTIONAL DIFFUSION COEFFICIENT ADDITIVE TERM. |
| CD | A3     | THIRD DIMENSION DIRECTIONAL DIFFUSION COEFFICIENT MULTIPLIER. |
| CD | B3     | THIRD DIMENSION DIRECTIONAL DIFFUSION COEFFICIENT ADDITIVE TERM. |

C-----------------------------------------------------------------------

CEOFP
APPENDIX E. CODE-DEPENDENT BINARY FILES (CONT'D.), DCCOEF.

PREPARED 1/27/76 AT ANL

DCCOEF (OR ACCOEF)
DIRECT (OR ADJACENT) SYNTHESIS COMBINING COEFFICIENT FILE

NUMBER OF ENERGY GROUPS BEFORE COLLAPSING.
NUMBER OF ENERGY GROUPS AFTER COLLAPSING.
NUMBER OF FUNCTION FILE NAMES IN TOC.
THE NUMBER OF CHANNEL STRUCTURE SCHEMES.
THE MAXIMUM NUMBER OF WEIGHTING FUNCTIONS USED IN A SINGLE ZONE, OR THE MAXIMUM NUMBER OF EXPANSION FUNCTIONS, WHICHER IS LARGER.
MAXKNM.LE.MAXUSE.
NCOLGP = 0 FOR GROUP DEPENDENT SYNTHESIS.
= 1 FOR GENERAL GROUP COLLAPSING.
= 2 FOR SPECIAL GROUP COLLAPSING.
THE NUMBER OF A6 WORDS USED TO STORE THE TITLE.
THE NUMBER OF AXIAL ZONES IN THE MODEL. THERE IS A NEW ZONE WHEN EITHER THE PLANAR GEOMETRY OF THE FUNCTION SETS CHANGE.

FILE STRUCTURE

FILE IDENTIFICATION
ALWAYS
SPECIFICATIONS
ALWAYS
FIXED POINT ARRAYS
ALWAYS
FLOATING POINT ARRAYS
ALWAYS
COMBINING COEFFICIENTS
ALWAYS

FILE IDENTIFICATION

HNAME, (HNAME(I),I=1,2),TVERS

103MUL
APPENDIX E. CODE-DEPENDENT BINARY FILES (CONT'D.). DCCOFF.

CD 
HNAMF  HOLLERITH FILE NAME - DCCOFF (OR ACCORP) - (A6) -
CD 
HUSE  HOLLERITH USER IDENTIFICATION (A6).
CD 
IVRS  FILE VERSION NUMBER.
CD 
MULT  WORD LENGTH PARAMETER.
CD 
   MULT = 1, A6 WORD IS SINGLE WORD.
CD 
   = 2, A6 WORD IS DOUBLE PRECISION WORD.
C

---------------------------------------------------------------------

CR

SPECIFICATIONS

CR

NGROUP, LGROUP, MAXFUN, MAXCHN, NCHNST, KNAX, MAXKNM, NCOLGP, NTITLE,
1 NUMFIN

C

10

C

MAXCHN  THE MAXIMUM NUMBER OF CHANNELS IN ANY CHANNEL

C

STRUCTURE SCHEME. MAXCHN EQ. THE LARGEST

C

VALUE OF NUMCHN(I), I = 1, NCHNST.

C

KNAX  NUMBER OF AXIAL FLUX INTERVALS.

C

---------------------------------------------------------------------

CR

FIXED POINT ARRAYS

CR

((TGPUSE(N,J), N=1, LGROUP), J=1, MAXFUN),

1 ((INT(J,K), J=1, MAXK4P), K=1, NNMKN),

2 ((TPI(J,K), J=1, MAXKNM), K=1, NNMKN),

3 ((NCHAN(N,J), N=1, LGROUP), J=1, MAXFUN), (NUMCHN(I), I=1, NCHNST),

4 (KCHAN(T), T=1, NNMKN), (KDTM(T), T=1, NNMKN),

5 (TVRFUN(J), J=1, MAXFIN)

C

2 = LGROUP*MAXFIN + 2*MAXKNM*NMKN + NCHNST + 2*NMKN + MAXFIN

C

TGPUSE  A PROJECTION MATRIX (ELEMENTS ARE 0 OR 1)

C

SHOWING WHICH GROUP PLAXFS OF EACH FUNCTION

C

ARE TO BE USED (1) OR OMMITTED (0).

C

INT  THE EXPANDING FUNCTIONS USED IN ZONE K, IF

C

THERE ARE FEWER THAN MAXKNM, THE OTHER ENTRIES

C

FOR A PARTICULAR ZONE ARE ZERO.

C

TPI  THE EXPANSION FUNCTIONS USED IN ZONE K, IF

C

THERE ARE FEWER THAN MAXKNM, THE OTHER ENTRIES

C

FOR A PARTICULAR ZONE ARE ZERO.

C

NCHAN  CHANNEL SCHEME ASSIGNMENT FOR EACH GROUP-

C

COLLAPSED GROUP FLUX.

C

NUMCHAN  NUMBER OF CHANNELS IN CHANNEL STRUCTURE

C
APPENDIX E. CODE-DEPENDENT BINARY FILES (CONT'D.).

CD KCHANG
THE TOPMOST MESH INTERVAL IN EACH ZONE.

CD KCHANG(NUMKZN)=KMAX.

CD KDTM
THE NUMBER OF UNKNOWN COMBINING COEFFICIENTS
AND THEREFORE EQUATIONS ASSOCIATED WITH ONE
MESH INTERVAL IN ZONE I.

CD IVRFUN
VERSION NUMBER OF FILE FOR FUNCTION J.

CD SEE FUNNAM.

-----------------------------------------------------------------------

C floating point arrays

CL ((FUNNAM(I,J),I=1,3),J=1,MAXFUN), (TITLE(I),I=1,MTITLE),

CL ((N(I,J,K),I=1,LGROUP),J=1,NGROUP),K=1,2), EIGEN

C

CW 3*MULT*MAXFUN+MULT*MTITLE+1 IF NCOLGP.EQ.0

CW 3*MULT*MAXFUN+MULT*MTITLE+2*GROUP+NGROUP+1 IF NCOLGP.GT.0

C

CD FUNNAM
NAME OF FILE FOR FUNCTION J.

CD TITLE
USER INPUT TITLE.

CD T
GENERAL GROUP COLLAPSING MATRIX FOR THE

CD PXFANSION FUNCTIONS (K=1) AND WEIGHTING

CD PFUNCTIONS (K=2).

CD EIGEN
SYNTHESIS EIGENVALUE.

C

CW N IS OMITTED WHEN NCOLGP=0.
COMBINING COEFFICIENTS

SYNTHESIS COMBINING COEFFICIENTS, PACKED TO ELIMINATE ZEROS.

THE TOTAL NUMBER OF NONZERO COMBINING COEFFICIENTS.

THE DIRECT COMBINING COEFFICIENTS, CC(N,I,K), ASSOCIATED WITH EACH COLLAPSED GROUP (N), EXPANSION FUNCTION (I) AND AXIAL MESH INTERVAL (K) CAN BE UNPACKED FROM VECTOR A IN THE FOLLOWING MANNER.

DIMENSION CC(LGROUP,MAXFUN,KMAX)

NPT=0
K2=0
DO 20 L=1,NMKZN K1=K2+1 K2=K2+KCHANG(L) DO 20 K=K1,K2 DO 20 M=1,MAXKNM I=TPL(N,L) IF (I.EQ.0) GO TO 20 DO 10 N=1,LGROUP CC(N,I,K)=0 IF (IGPUSE(N,I).EQ.0) GO TO 20 NPT=NPT+1 CC(N,I,K)=A(NPT)

10 CONTINUE

FOR THE ADJOINT COMBINING COEFFICIENT (IN THE A'COEF FILE) USE IWT(N,L) INSTEAD OF TPL(N,L).
APPENDIX E. CODE-DEPENDENT BINARY FILES (CONT'D.). DIPINT.

C**-----------------------------------------------------------------------**
C** FILE STRUCTURE **
C** RECORD TYPE PRESENT IF **
C** (REPEAT FOR NRCDTF RECORDS) ALWAYS **
C**
C**-----------------------------------------------------------------------**

C** INTEGRALS **
C
C (X(I), I=1, N1)
C
C MULT*LEN
C
X A DIP INTEGRAL. SEE THE INTEGRAL TABLE OF
C CONTENTS IDIP1B IN THE INTTOC FILE.
C LEN RECORD LENGTH. LFN=LEN DP(I) FOR RECORD I
C LENDIP SEE FIXED POINT ARRAYS IN INTTOC FILE.
C
C THE DIP INTEGRALS ARE STORED IN THE FOLLOWING FORMS.
C IF NSTDIP = 0 (ONLY IF THE DIP MATRIX IS
C BLOCK DIAGONAL).
C
C (((((DIP(I,J,K), I=1, NCHAN(K,1W))),
C J=1, NCHAN(K,IE)), K=1, LGROUP))
C
C IF NSTDIP = 1
C
C (((((DIP(I,J,K,L), I=1, NCHAN(K,1W))),
C J=1, LGROUP, K=1, NCHAN(I,IE)),
C L=1, LGROUP))
APPENDIX E.  CODE-DEPENDENT BINARY FILES (CONTD.).  DIFINT.

CD IW  WEIGHTING FUNCTION IDENTIFIER
       (1.LE.IW.LE.MAXFUN).
CD IE  EXPANSION FUNCTION IDENTIFIER
       (1.LE.IE.LE.MAXFUN).
CD IP  PLANAR GEOMETRY IDENTIFIER (1.LE.IP.LE.MAXPLN).
C
CN    SEE THE INTTOC FILE DESCRIPTION FOR
CN    DEFINITIONS OF THE OTHER VARIABLES.
C
C-- ----------------------------------------------------------
C
CEND
APPENDIX E. CODE-DEPENDENT BINARY FILES (CONT'D.). INTTOC.

PREPARED 1/27/76 AT ANL

INTTOC

TABLE OF CONTENTS (TOC) FOR SYNTHESIS INTEGRAL
DATA SETS VOLINT AND DIFINT

CD

\*MAX
NUMBER OF AXIAL MESH INTERVALS.

CD LDIPTB
NUMBER OF ENTRIES IN THE IDIPTB TABLE.

CD LNEDLZ
LENGTH OF DLZMAT ARRAY.

CD LGROUP
NUMBER OF ENERGY GROUPS AFTER COLLAPSING.

CD LVOLTB
NUMBER OF ENTRIES IN THE IVOLTB TABLE.

CD MAXFD
NUMBER OF FUNCTION FILE NAMES IN TOC.

CD MAXHM
THE MAXIMUM NUMBER OF WEIGHTING FUNCTIONS USED
IN A SINGLE ZONE, OR THE MAXIMUM NUMBER OF
EXPANSION FUNCTIONS, WHICHER IS LARGER.

CD MAXPLM
NUMBER OF PLANAR GEOMETRY FILE NAMES IN TOC.

CD MCHNST
THE NUMBER OF CHANNEL STRUCTURE SCHEMES.

CD MCOLGP
MCOLGP = 0 FOR GROUP DEPENDENT SYNTHESIS.
= 1 FOR GENERAL GROUP COLLAPSING.
= 2 FOR SPECIAL GROUP COLLAPSING.

CD NGROUP
NUMBER OF ENERGY GROUPS BEFORE COLLAPSING.

CD NRCDTF
NUMBER OF RECORDS IN THE DIFINT FILE.

CD NRIVOL
NUMBER OF RECORDS IN THE VOLINT FILE.

CD NSCALE
NSCALE = 1 IF USER SUPPLIED SCALING FACTORS
ESCAPE AND "SCALE" ARE TO BE
APPLIED.
= 0 IF NOT TO BE APPLIED.

CD NTTL
THE NUMBER OF A6 WORDS USED TO STORE THE
TITLE.

CD NUMZM
THE NUMBER OF AXIAL ZONES IN THE MODEL. THERE
IS A NEW ZONE WHEN EITHER THE PLANAR GEOMETRY
OF THE FUNCTION SETS CHANGE.

FILE STRUCTURE

<table>
<thead>
<tr>
<th>RECORD TYPE</th>
<th>PRESENT IF</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILE IDENTIFICATION</td>
<td>ALWAYS</td>
</tr>
<tr>
<td>SPECIFICATIONS</td>
<td>ALWAYS</td>
</tr>
<tr>
<td>FIXED POINT ARRAYS</td>
<td>ALWAYS</td>
</tr>
<tr>
<td>FLOATING POINT ARRAYS</td>
<td>ALWAYS</td>
</tr>
<tr>
<td>VOLINT TOC</td>
<td>ALWAYS</td>
</tr>
<tr>
<td>DIFINT TOC</td>
<td>LDIPTB.GT.0</td>
</tr>
<tr>
<td>SCALING FACTORS</td>
<td>NSCALE.EQ.1</td>
</tr>
<tr>
<td>BOUNDARY CONDITION CONSTANTS</td>
<td>ALWAYS</td>
</tr>
</tbody>
</table>
APPENDIX E.  CODE-DEPENDENT BINARY FILES (CONTD.).  INTTOC.

FILE IDENTIFICATION

CL HNAME, (HUSE(I), I=1,2), IVERS
C
CV 1+3*MULT
C
CD HNAME  HOLLERITH FILE NAME - INTTOC - (A6).
CD HUSE  HOLLERITH USER IDENTIFICATION (A6).
CD IVERS  FILE VERSION NUMBER.
CD MULT  WORD LENGTH PARAMETER.
   MULT=1, A6 WORD IS SINGLE WORD.
   =2, A6 WORD IS DOUBLE PRECISION WORD.

SPECIFICATIONS

CL TOTPOW, EPS, POWFST, XEPS, IBCLX, IBCXU, IPCYL, IBCYU, IBCZL, IBCZU,
CL 1BNDCH, IGEOM, IMAX, JMAX, KMAX, LDIPTB, LENA, LENLZ, LENPLU, LENINT,
CL 2LENPOW, LGROUP, LVOLTB, MAXCHW, MAXCYP, MAXDIM, MAXDN, MAXPW, MAXNM,
CL 3MAXD, MAXPLN, MAXREG, MAXSIZ, MAXUP, MAUXSE, NCAIC, NCHNST, NCOLGP,
CL 4NFLUNI, NGROUP, NRCDIF, NRCVOL, NSCALE, NSTDIF, NTITLE, NTRIA,
CL 5NTRIPT, NGROUP, NSTDIF TITLE, NTRIAG,
C
CV 48
C
CD TOTPOW  TOTAL POWER (WATTS).
CD EPS  A SMALL NUMBER USED IN TESTS FOR EQUALITY OF MESH INTERVALS.
CD POWFST  NOT USDF.
CD XEPS  A SMALL NUMBER USED IN TESTS FOR ZERO CROSS SECTIONS.
CD IBCLX  THE LOWER X BOUNDARY CONDITION.
   1  - ZERO FLUX.
   2  - REFLECTIVE.
   3  - EXTRAPOLATED, C*DEL PHI + PHI = 0.
   4  - PERIODIC WITH OPPOSITE BOUNDARY.
   5  - PERIODIC WITH NEXT BOUNDARY GOING CLOCKWISE.
   6  - PERIODIC WITH NEXT BOUNDARY GOING COUNTERCLOCKWISE.
   7  - PERIODIC, INVERTED ALONG SAME BOUNDARY.
CD IBCXU  THE UPPER X BOUNDARY CONDITION.
CD IPCYL  THE LOWER Y BOUNDARY CONDITION.
CD IBCYU  THE UPPER Y BOUNDARY CONDITION.
CD IBCZL  THE LOWER Z BOUNDARY CONDITION.
CD IBCZU  THE UPPER Z BOUNDARY CONDITION.
APPENDIX E. CODE-DEPENDENT BINARY FILES (CONTD.). INTOC.

C) IBNDCH   LOWEST ENERGY GROUP (AFTER GROUP COLLAPSING) FOR WHICH THERE IS A NONZERO CHI.
C) IGEOM    PLANAR GEOMETRY TYPE SENTINEL.
C) IMAX     THE NUMBER OF MESH INTERVALS IN THE FIRST (X) DIRECTION.
C) IMAY     THE NUMBER OF MESH INTERVALS IN THE SECOND (Y) DIRECTION.
C) LENA     THE STORAGE REQUIRED FOR THE COMBINING COEFFICIENTS.
C) LENFLX   LENGTH OF THE ARRAY FLUX IN OVERLAY SOLVE.
C) LENINT   THE MAXIMUM RECORD LENGTH FOR VOLINT AND DIFTINT INTEGRAL FILES.
C) LENPOW   LENGTH OF THE ARRAY POW IN OVERLAY SOLVE.
C) MAXCHN   THE MAXIMUM NUMBER OF CHANNELS IN ANY CHANNEL STRUCTURE SCHEME. MAXCHN.EQ.THE LARGEST VALUE OF NUMCHN(I),I=1,NCHNST.
C) MAXCMP   THE NUMBER OF MATERIALS REPRESENTED IN THE INPUT CROSS SECTION FILE.
C) MAXDIM   THE LARGEST DIMENSION OF ANY MATRIX USED IN THE SOLUTION.
C) MAXDN    MAXIMUM NUMBER OF DOWNSCATTERING GROUPS.
C) MAXMD    THE LARGEST DIMENSION OF AN INPUT MATRIX (THE LARGEST NUMBER OF GROUP-CHANNEL COMBINATIONS FOR A SINGLE FUNCTION). MAXMD**2 IS THE MINIMUM SPACE REQUIRED FOR A RECORD OF REWRITTEN INTEGRALS.
C) MAXREG   THE MAXIMUM NUMBER OF REGIONS ASSOCIATED WITH ANY OF THE PLANAR GEOMETRIFS.
C) MAXSIZ   THE LENGTH OF THE BPOINTER CONTAINER ARRAY.
C) MAXUP    MAXIMUM NUMBER OF UPSCATTERING GROUPS.
C) MAXUSP   MAXIMUM NUMBER OF INPUT FUNCTIONS USED ANYWHERE IN THE MODEL AS EXPANSION FUNCTIONS OR WEIGHTING FUNCTIONS, WHICHEVER IS LARGER.
C) MAXUSE.LP.MAXFUN.
C) MTCALC   CALCULATION TYPE SENTINEL. SEE TYPE 2 CARD OF SYNPIL.
C) NPLUNI   A BOGUS FILE NUMBER FOR THE UNIT PLUNI.
C) NSTDTP   FORMAT SENTINEL FOR DIFTINT FIPIL.
C) NTAG     TRIANGULAR MESH SENTINEL.
C) NTRIPT   SECONDARY TRIANGULAR MESH SENTINEL.
APPENDIX F. CODE-DEPENDENT BINARY FILES (CONT'D.). TWTTCG.

C-----------------------------------------------
C FIXED POINT ARRAYS
C-----------------------------------------------
C
C (TBND$1(I), I=1, LGROUP), (TBND$2(I), I=1, LGROUP),
C 1 (LNDPG(I), I=1, MAXPLN), (LENVL(I), I=1, NRCVOL),
C 2 (LNDIF(I), I=1, NRCDTP), (TGPNGE(N, J), N=1, LGROUP), 1=1, MAXFUN),
C 3 ((JWT(J, K), J=1, MAXKCN), K=1, NUKZ4),
C 4 ((TDL(J, K), J=1, MAXKCN), K=1, NUKZ4),
C 5 ((NCHAN(N, J), N=1, LGROUP), J=1, MAXFUN), (NCHPN(I), I=1, NCHNST),
C 6 (KCHANG(I), I=1, NUKZ4), (KCHON(I), I=1, NUKZ4),
C 7 (IVRPLN(J), J=1, MAXFUN), (IVRPNJ(J), J=1, MAXPLN)
C
C 2*GROUP+2*MAXLN+NRCVOL+NRCDTF+2*GROUP+MAXLN+2*MAXKCN+NUKZ4
C +NCHNST+4*NUKZ4+MAXFUN
C
C IBND$1 THE POSITION IN COLUMN I OF THE GROUP
C COLLAPSED SCATTERING MATRIX OF THE TOPMOST,
C NONZERO ELEMENT, EQUIVALENT TO LOG-SCH WHEN
C THERE IS NO GROUP COLLAPSED.
C IBND$2 THE POSITION IN COLUMN I OF THE GROUP
C COLLAPSED SCATTERING MATRIX OF THE BOTTOM,
C NONZERO ELEMENT.
C IBND$7 THE LOWEST ENERGY GROUP (AFTER GROUP COLLAPSING)
C FOR WHICH THERE IS A NONZERO PION CROSS
C SECTION FOR PLANE I. A 2" means no pions.
C LENVL LENGTH OF RECORD I OF THE VOLUNT "T1".
C LENDLP LENGTH OF RECORD I OF DPINT FILE.
C TGROUP A PROJECTION MATRIX (ELEMENTS ARE 0 OR 1)
C SHOWING WHICH GROUP PLAYS OF EACH FUNCTION
C ARE TO BE USED (1) OR OMITTED (0).
CJWT THE WEIGHTING FUNCTION USED IN ZONE K. IF
C THERE ARE LESS THAN MAXCN, THE OTHER ENTRIES
C FOR A PARTICULAR ZONE ARE ZERO.
C TPL THE EXPANSION FUNCTIONS USED IN ZONE K. IF
C THERE ARE LESS THAN MAXCN, THE OTHER ENTRIES
C FOR A PARTICULAR ZONE ARE ZERO.
CNCHAN CHANNEL SCHEME ASSIGNMENT FOR EACH GROUP-
C COLLAPSED GROUP FILE.
CNCHN0 NUMBER OF CHANNELS IN CHANNEL STRUCT.
C KCHANG THE TOPMOST PION VALUE IN EACH ZONE.
C KGEOM THE GEOMETRY FOR PION VALUE KCHANG(I).
C IVRPNJ VERSO NUMBER OF FILE FOR FUNCTION J.
C IVRPLN VERSO NUMBER OF FILE FOR PLANE GROUP OF J.
C
C-----------------------------------------------

APPENDIX E.  CODE-DEPENDENT BINARY FILES (CCN10e.).  "ITTOC."

FLOATING POINT ARRAYS

((PUNHAR(I,J), I=1,3), J=1,MAXPON),
((PUNHAR(I,J), I=1,3), J=1,MAXPLH), (TITLE(I), I=1,4TITLE),
((U(I,J,K), I=1,1), LGROUP), J=1,2),
((ULTRA(I), I=1,1), LGROUP), (EMPS(K), K=1,KNAX)

10041*0=MAXPIN*0MULT0AXPLH=MULT0TITLE=LEVELZ+KNAX
IP NCOLGP.EQ.0.
10041*0=MAXPIN*0MULT0AXPLH=MULT0TITLE=LEVELZ+KNAX*2*LGROUP
IMGROUP IP NCOLGP.GT.0.

PUNHAR NAME OF FILE FOR FUNCTION J.
PUNHAR NAME OF FILE FOR PLANAR GEOMETRY J.
TITLE TITLE.
" GENERAL GROUP COLLAPSE MATRIX FOR THE
ULTRA EXPANSION FUNCTIONS (K=1) AND WEIGHTING
ULTRA NOST OF 2 ADJACENT AXIAL MESH INTERVALS.
(TOPF/DUPF)"""

UPF  POSITION OF UPPERS BOUNDARY OF AXIAL MESH
UPF INTERVAL K.
F
PUNHAR, PUNHAR AND TITLE ARE DOUBLE PRECISION
AUXAAYS ON MUL-T2 MACHINES.
F
IP NCOLGP=0, 0 IS NOT INCLUDED.
F
APPENDIX E.  CODE-DEPENDENT BINARY FILES (CONTD.).  TNTTOC.

C........................................................................................................
C VOLNT TOC
C
C ((TVOLTB(I,K),I=1,7),K=1,LVOLTB)
C
C IVOLTB TABLE OF CONTENTS FOR REM AND FIS INTEGRALS
C IN THE VOLINT FILE.
C
C IVOLTB(1,K)  =  1 FOR REM INTEGRAL.
C =  2 FOR A FIS INTEGRAL.
C =  3 FOR A PO INTEGRAL.
C =  4 FOR A FLUX INTEGRAL.
C IVOLTB(2,K)  THE RECORD OF THE VOLNT FILE
C CONTAINING THE INTEGRAL.
C =  0 IF THE INTEGRAL COULD NOT BE
C D orn.
C = -1 IF THE INTEGRAL COULD HAVE
C BEEN D orn BUT FOR SOME REASON
C WAS NOT.
C IVOLTB(3,K)  POINTER TO THE FIRST WORD OF THE
C INTEGRAL BLOCK WITHIN THAT RECORD.
C IVOLTB(4,K)  PLANAR GEOMETRY IDENTIFIED.
C IVOLTB(5,K)  WEIGHT FUNCTION IDENTIFIED.
C IVOLTB(6,K)  EXPANSION FUNCTION IDENTIFIED.
C IVOLTB(7,K)  NUMBER OF RECORDS REQUIRED FOR ALL
C THE INTEGRALS OF THIS COMBINATION.
C
C........................................................................................................
APPENDIX F.  CODE-DEPENDENT BINARY FILES (CONT.).  INTROC.

C------------------------------------------------
C              DIFINT TOC
C------------------------------------------------
C (DIFTH(I,K),I=1,N),K=1,LDIFTH)
C 10*LDIFTH
C
C TDIFTH
C TABLE OF CONTENTS FOR DIF INTEGRALS IN
C DIFINT FILE.
C
C TDIFTH(I,K) RECORD OF THE DIFINT FILE
C CONTAINING THE INTEGRALS.
C TDIFTHS(2,K) POINTER TO THE FIRST WORD OF THE
C INTEGRAL BLOCK WITHIN THAT RECORD.
C TDIFTHB(3,K) LOWER PLANAR GEOMETRY IDENTIFIER.
C = 0 FOR THE BOTTOM BOUNDARY
C INTEGRAL.
C TDIFTHB(4,K) UPPER PLANAR GEOMETRY IDENTIFIER.
C = 0 FOR THE TOP BOUNDARY INTEGRAL.
C TDIFTHR(5,K) POINTER TO A MESH INTERVAL RATIO
C IN THE DLZAR ARRAY.
C TDIFTHR(6,K) WEIGHT FUNCTION IDENTIFIER.
C TDIFTHR(7,K) EXPANSION FUNCTION IDENTIFIER.
C TDIFTHR(8,K) NUMBER OF RECORDS REQUIRED FOR ALL
C THE INTEGRALS OF THIS COMBINATION.
C------------------------------------------------
C
C------------------------------------------------
C            SCALING FACTORS
C------------------------------------------------
C (SCALF(I,J),I=1,N,SCALF),J=1,MAXPI)
C 1 (SCALF(I,J),I=1,N,SCALF),J=1,MAXPI)
C 9*GROUP*MAXPI
C
C RSCALE
C A USER SUPPLIED SCALING FACTOR FOR EACH GROUP
C OF FUNCTION J WHEN FUNCTION J IS USED AS AN
C EXPANSION FUNCTION. IT CAN ONLY HAVE AN EFFECT
C WHEN GROUP COLLAPSING IS EMPLOYED.
C
C SSCALE
C A USER SUPPLIED SCALING FACTOR FOR EACH GROUP
C OF FUNCTION J WHEN FUNCTION J IS USED AS A
C WEIGHTING FUNCTION. IT CAN ONLY HAVE AN EFFECT
C WHEN GROUP COLLAPSING IS EMPLOYED.
C
C 9*GROUP*MAXPI
C
C PRESENT ONLY IF NSCALE.LG.1.
APPENDIX P.  CODE-DEPENDENT PINARY FILES (CCNTD.). TTTCOC.

C-----------------------------------------------
CR  BOUNDARY CONDITION CONSTANTS
C
CL  ((C (*, J), I=1, 6), J=1, 6GROUP)
C
CV  6GROUP
C
CD  C  CONSTANT IN THE HOMOGENEOUS BOUNDARY CONDITION -
CD  EXPRESSION DEL PHI*C*PHI = 0.
C
CD-----------------------------------------------
C
COP
APPENDIX F. CODE-DEPENDENT BINARY FILES (CONT.). RECPX.

PREPARED 1/27/76 AT ANL.

RECPX

THIS IS A SCRATCH FILE CONTAINING THE GROUP FLUXES,
MESH SPACING AND COMPOSITION MAPS FOR RANGES OF THE
X AND Y MESH. ALL THE GROUP FLUXES AND PLANEAR
COMPOSITION MAPS REQUIRED FOR THE SYNTHESIS INTEGRAL
CALCULATION ARE INCLUDED.

LEPREP

NUMBER OF RECORDS IN THE FILE RECPX.
LELEN

RECORD SIZE OF RECPX.

FILE STRUCTURE

RECORD TYPE               PRESENCE

* * * * * * * (REPEAT FOR LEPREP RECORDS) * * * * * * *

DATA

* * * * * * *

DATA

((((FLUX(I,J,Y,Y,JOPLN(L)),I=1,12),J=J1,J2),N=1,NGROUP),
N=1,NUMPLN),Y(J),I=1-1,12),,Y(J),J=J1,J2),
J=1,JOPLN(L)),T=1,1,2),J=J1,J2),L=1,NUMPLN)

* * * * * * * (NGROUP * NUMPLN * NUMPLN) * * * * * * *

FLUX

THE FLUX FOR MESH INTERVAL (I,J), GROUP N,
OF FUNCTION NUMBER JDOPEN(L). THESE FLUXES
ARE NOT INCLUDED IN THE FIIT.

Y

THE Y POSITION OF THE MESH LINE SEPARATING
INTERVALS I AND I+1.

Y

THE Y POSITION OF THE MESH LINE SEPARATING
INTERVALS J AND J+1.

ICOMP

THE COMPOSITION NUMBER IN MESH INTERVAL (T,J)
OF FLANAR GEOMETRY NUMBER JDOPEN(L).

IJ,IT

THE FIRST AND LAST COLUMNS ASSOCIATED WITH A
RECORD OF FLUXES. I2-I1=IAND-1 EXCEPT,
PERHAPS, WHEN I1 IS THE LAST COLUMN. FOR THE
FIRST RECORD I1=0.

IJ,J2

THE FIRST AND LAST PUNKS ASSOCIATED WITH A
RECORD OF FLUXES. J2-J1=JAND-1 EXCEPT,
PERHAPS, WHEN J1 IS THE LAST PUNK. FOR THE
FIRST RECORD J1=0.
CD IBAND THE NUMBER OF COLUMNS OF PLINES IN ONE RECORD
CD IBAND OF THE REFLX FILE.
CD IBAND THE NUMBER OF ROWS OF PLINES IN ONE RECORD
CD IBAND OF THE REFLX FILE.
CD JDOFIN THE LIST OF FUNCTIONS REQUIRED TO CALCULATE
CD THE INTEGRALS, IN THE ORDER IN WHICH THEY
CD APPEAR IN THE REFLX FILE. WHEN IT IS PRESENT,
CD THE UNIT FUNCTION IS LAST IN THE LIST.
CD JDOFIN THE LIST OF PLANAR GEOMETRIES REQUIRED TO
CD CALCULATE THE INTEGRALS.
CD MUPLY THE NUMBER OF PLANE COMPLETES REQUIRED TO
CD CALCULATE THE INTEGRALS (MUPLY = JDOFIN).
CD MUPLY THE NUMBER OF FUNCTIONS REQUIRED TO CALCULATE
CD THE INTEGRALS. IF THE UNIT FUNCTION IS
CD REQUIRED, IT IS INCLUDED IN MUPLY, EVEN
CD THOUGH NO SUCH PLUX FILE IS EVER USED
CD (MUPLY = JDOFIN).
CD THE BLOCKS OF DATA OVERLAP ONE WASH INTERVAL
CD IN THE X DIRECTION BUT DO NOT OVERLAP IN THE
CD Y DIRECTION. THE RANGES OF X AND Y WASH
CD ASSOCIATED WITH EACH RECORD OF REFLX ARE
CD DEFINED IN THE FOLLOWING SEQUENCE.

<table>
<thead>
<tr>
<th>RECORD</th>
<th>X1</th>
<th>X2</th>
<th>Y1</th>
<th>Y2</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>TBAND-1</td>
<td>0</td>
<td>TBAND-1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>TBAND-1</td>
<td>0</td>
<td>2*TBAND-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>TBAND-1</td>
<td>(TBAND-1)</td>
<td>0</td>
<td>TBAND-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
APPENDIX E. CODE-DEPENDENT BINARY FILES (CONTD.). REQXST.

PREPARED 1/27/76 AT ANL

THIS IS A SCRATCH FILE CONTAINING THE MACROSCOPIC CROSS SECTIONS REQUIRED FOR DOING THE SYNTHESIS INTEGRALS

**FILE STRUCTURE**

<table>
<thead>
<tr>
<th>RECORD TYPE</th>
<th>PRESENT IF</th>
</tr>
</thead>
<tbody>
<tr>
<td>*** (REPEAT FOR LREQX1 RECORDS)</td>
<td></td>
</tr>
<tr>
<td>* FISSION AND REMOVAL DATA</td>
<td>LREQX1.GT.0</td>
</tr>
<tr>
<td>***</td>
<td></td>
</tr>
<tr>
<td>*** (REPEAT FOR LREQX2 RECORDS)</td>
<td></td>
</tr>
<tr>
<td>* DIFFUSION COEFFICIENTS</td>
<td>LREQX2.GT.0</td>
</tr>
<tr>
<td>***</td>
<td></td>
</tr>
</tbody>
</table>

**FISSION AND REMOVAL DATA**

\[
\begin{align*}
&\text{LXST1} = \text{LENST} + 3 \times \text{NGROUP} \\
&\text{LENST} = \text{MAXUP} \times (2 \times \text{NGROUP} - \text{MAXUP} - 1) + \text{MAXDN} \times (2 \times \text{NGROUP} - \text{MAXDN} - 1) / 2 \\
&\text{LXST2} = \text{MAXDN} \times \text{MAXUP} \\
&\end{align*}
\]

THE NUMBER OF CROSS SECTION DATA PER COMPOSITION, FOR RECORDS OF REQXST CONTAINING FISSION AND REMOVAL DATA.
The matrix is stored in Packed form, excluding entries outside the scattering band.

THE NUMBER OF DATA IN THE REMOVAL CROSS SECTION MATRIX. THE MATRIX IS STORED IN PACKED FORM, EXCLUDING ENTRIES OUTSIDE THE SCATTERING BAND.

THE NUMBER OF COMPOSITIONS PER RECORD, FOR RECORDS OF REQXST CONTAINING FISSION AND REMOVAL DATA.

MAXIMUM NUMBER OF DOWNSCATTERING GROUPS.

MAXIMUM NUMBER OF UPSCATTERING GROUPS.
APPENDIX R.  CODE-DEPENDENT BINARY FILES (CONT.).

FISSION AND REMOVAL DATA FOR REQUIRED
COMPOSITIONS, REQUIRED COMPOSITIONS 1 THROUGH
LST2 ARE IN THE FIRST RECORD OF REQUEST,
REQUIRED COMPOSITIONS LST2+1 THROUGH 2*LST2
IN THE SECOND, ETC. THE ACTUAL (COMPSX)
COMPOSITION NUMBER FOR REQUIRED COMPOSITION
NUMBER K IS JOCMP(K). FOR EACH COMPOSITION,
THE REMOVAL MATRIX IS STORED FIRST (SEE THE
DEFINITION OF LENSCT), FOLLOWED BY VECTORS
CONTAINING NU SIGMA P, CHI AND THE POWER CROSS
SECTION SCATTERING CROSS SECTIONS INTO A
GROUP APPEAR AS NEGATIVE NUMBERS IN THE
REMOVAL MATRIX.
THE LIST OF COMPOSITIONS REQUIRED TO
CALCULATE THE INTEGRALS.

DIFFUSION COEFFICIENTS

THE NUMBER OF GROUPS PER RECORD FOR RECORDS
CONTAINING ONLY DIFFUSION COEFFICIENTS.
THE NUMBER OF MATERIALS FOR WHICH CROSS
SECTIONS ARE REQUIRED WHEN THE MATERIALS ARE
CALCULATED (NUMCMP.LE.MAXCMP).
THE DIFFUSION COEFFICIENT FOR EACH OF THE
DIRECTIONS FOR ALL REQUIRED COMPOSITIONS, ALL
THE DATA FOR GROUPS 1 THROUGH LST3 ARE STORED
IN THE FIRST DIFFUSION COEFFICIENT RECORD,
GROUPS LST3+1 THROUGH 2*LST3 IN THE SECOND,
ETC.
APPENDIX D.  CODE-DEPENDENT BINARY FILES (CONT'D.).  SYNCON.

PREPARED 1/27/76 AT ANL

SYNCON
A BINARY FILE CONTAINING THE DATA IN THE OCD FILE
SYN1FL.  THIS IS A SCRATCH FILE WRITTEN IN OVERLAY
CARDS OF SYN1FL

FILE STRUCTURE

RECORD TYPE  PRESENT IF

MAXIMUM CARD TYPE  ALWAYS
CARDS PER CARD TYPE  ALWAYS

****** (REPEAT FOR EACH INPUT
SYN1FL CARD)
SYN1FL CARD DATA  ALWAYS

MAXIMUM CARD TYPE

MAXREC

1

MAXREC  MAXIMUM CARD TYPE NUMBER OF SYN1FL INPUT.

CARDS PER CARD TYPE

(MAXRECI-I+1,MAXREC)

MAXREC

NUMBER OF RECORDS (CARDS) OF CARD TYPE I
INCLUDED IN SYN1FL INPUT.
APPENDIX E. CODE-DEPENDENT BINARY FILES (CONT'D). SYNCON.

C-------------------------------
CR SYNPIL CARD DATA
C
CN EACH RECORD OF THIS TYPE CONTAINS THE DATA ON
CN ONE CARD OF THE SYNPIL INPUT, EXCLUDING THE
CN CARD TYPE NUMBER. THE CARDS MUST BE IN ORDER
CN OF ASCENDING CARD TYPE NUMBER.
C-------------------------------

CPDP
APPENDIX F. CODE-DEPENDENT BINARY FILES (CONTD.). VOLINT.

PREPARED 1/27/76 AT ANL

VOLINT, FISSTON, POWER AND FLUX INTEGRALS FOR SYN3D

NRCVOL = NUMBER OF RECORDS IN VOLINT FILE. (SEE SPECIFICATIONS RECORD OF INTTOC FILE).

FILE STRUCTURE

RECORD TYPE PRESENT IF

**** (REPEAT FOR NRCVOL RECORDS) ALWAYS

INTEGRALS

MULTFLX

A PTS. PFN, POF OR FLUX INTEGRAL. SEE THE INTEGRAL TABLE OF CONTENTS IVOLTP IN THE INTTOC FILE.

LEN LENVOL SEE FIXED POINT AREAS IN INTTOC FILE.

FOR INTEGRALS ARE STORED

(((FIT (I,J,K,L), I=1, NUMCHN (NCHAN (J,IP))), J=1, TBNDF (IP)),
 L=1, LENDFLX (IP))

THE other INTEGRALS ARE STORED

(((PFN (I,J,K,L), I=1, NUMCHN (NCHAN (J,IP))), J=I, TBNDS1 (L), TBNDS2 (L)),
 K=1, NUMCHN (NCHAN (1,IP)), L=1, LGROUP)

THE POF INTEGRALS ARE STORED

(((POF (I,J), I=1, NUMCHN (NCHAN (J,IP))),
 J=1, TBNDF (IP))

THE FLUX INTEGRALS ARE STORED

(((FLUX (I,J), I=1, NUMCHN (NCHAN (J,TE))),
 J=1, LGROUP)
APPENDIX R. CODE-DEPENDENT BINARY FILES (CONTD.).

CODE-DEPENDENT BINARY FILES (CONTD.).

- WEIGHTING FUNCTION IDENTIFIER
  - (1.0F1W.LP.MAXFUN).
- EXPANSION FUNCTION IDENTIFIER
  - (1.0FTE.LE.MAXFUN).
- PLANAR GEOMETRY IDENTIFIER (1.0F1P.LE.MAX2LN).

SEE THE INTTOC FILE DESCRIPTION FOR DEFINITIONS OF THE OTHER VARIABLES.

CEOF
APPENDIX F. SYN3D ERROR MESSAGES. ERRORS ARE IDENTIFIED BY SUBROUTINE NAME AND ERROR NUMBER.

<table>
<thead>
<tr>
<th>SYM</th>
<th>NUMB</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>BRKUP</td>
<td>10</td>
<td>THE CODE CANNOT HANDLE THE GEOMETRY TYPE SPECIFIED IN THE INPUT GEOST FILE.</td>
</tr>
<tr>
<td>BRKUP</td>
<td>152</td>
<td>AN OUTPUT GEOST FILE CANNOT BE &quot;WRITTEN&quot;.</td>
</tr>
<tr>
<td>CARDS</td>
<td>12</td>
<td>THERE IS NO INPUT SYN32 FILE.</td>
</tr>
<tr>
<td>CARDS</td>
<td>14</td>
<td>THERE ARE NO INPUT CARDS.</td>
</tr>
<tr>
<td>CARDS</td>
<td>222</td>
<td>THERE ARE NEITHER TYPE 4 AND 5 CARDS NOR AN INPUT GEOST FILE.</td>
</tr>
<tr>
<td>CARDS</td>
<td>410</td>
<td>AN OUTPUT SYNCON FILE CANNOT BE &quot;WRITTEN&quot;.</td>
</tr>
<tr>
<td>DECSIE</td>
<td>50</td>
<td>CODE BUG. THE OUTPUT INTEGRALS FOR A SINGLE COMBINATION ARE TOO BIG FOR THE SPECIFIED RECORD LENGTH.</td>
</tr>
<tr>
<td>DECSIE</td>
<td>52</td>
<td>CODE BUG.</td>
</tr>
<tr>
<td>DECSIE</td>
<td>130</td>
<td>SAME AS 50.</td>
</tr>
<tr>
<td>DECSIE</td>
<td>134</td>
<td>CODE BUG.</td>
</tr>
<tr>
<td>DECSIE</td>
<td>232</td>
<td>CODE BUG.</td>
</tr>
<tr>
<td>DIVVY</td>
<td>48</td>
<td>THERE IS NOT ENOUGH SPACE IN THE BPOINTER CONTAINER TO HOLD THE MATRICES GENERATED IN THE SOLUTION ROUTINE.</td>
</tr>
<tr>
<td>DIVVY</td>
<td>110</td>
<td>CODE BUG.</td>
</tr>
<tr>
<td>ENTS5</td>
<td>16</td>
<td>NO COMBINING COEFFICIENT DATASETS EXIST.</td>
</tr>
<tr>
<td>ENTS5</td>
<td>20</td>
<td>THERE IS NO INTEGRAL DATASET AVAILABLE.</td>
</tr>
<tr>
<td>ENTS5</td>
<td>26</td>
<td>THERE IS NO INTRPOC DATASET AVAILABLE.</td>
</tr>
<tr>
<td>ENTS5</td>
<td>60</td>
<td>TO CALCULATE THE PERTURBATION DENOMINATOR BOTH ECCOEF AND ACCOEF FILES MUST BE AVAILABLE.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SUCH IS NOT THE CASE.</td>
</tr>
<tr>
<td>ENTS5</td>
<td>82</td>
<td>THE BPOINTER CONTAINER IS NOT LARGE ENOUGH TO HOLD ALL THE DATA REQUIRED TO CONSTRUCT AN OUTPUT PLUX INTEGRAL FILE.</td>
</tr>
<tr>
<td>ENTS5</td>
<td>102</td>
<td>THE BPOINTER CONTAINER IS NOT LARGE ENOUGH TO HOLD ALL THE DATA REQUIRED TO CONSTRUCT AN OUTPUT POWER FILE.</td>
</tr>
<tr>
<td>FILLUP</td>
<td>40</td>
<td>CODE BUG. MISSING DIFINT INTEGRAL.</td>
</tr>
<tr>
<td>FILLUP</td>
<td>130</td>
<td>CODE BUG. MISSING DIFINT INTEGRAL.</td>
</tr>
<tr>
<td>FILLUP</td>
<td>160</td>
<td>CODE BUG. MISSING DIFINT INTEGRAL.</td>
</tr>
<tr>
<td>FILLUP</td>
<td>220</td>
<td>CODE BUG. MISSING VOLINT INTEGRAL.</td>
</tr>
<tr>
<td>FILLUP</td>
<td>260</td>
<td>CODE BUG. MISSING VOLINT INTEGRAL.</td>
</tr>
<tr>
<td>FILLUP</td>
<td>340</td>
<td>CODE BUG. MISSING DIFINT INTEGRAL.</td>
</tr>
<tr>
<td>FLUX3D</td>
<td>10</td>
<td>AN OUTPUT FILE CANNOT BE &quot;WRITTEN&quot;.</td>
</tr>
<tr>
<td>FLUX3D</td>
<td>12</td>
<td>THE CROSS SECTION FILE XSMCHN IS NOT AVAILABLE.</td>
</tr>
<tr>
<td>FLUX3D</td>
<td>62</td>
<td>A REQUIRED EXPANSION FUNCTION FILE IS NOT AVAILABLE.</td>
</tr>
<tr>
<td>HMG4C</td>
<td>1</td>
<td>A FATAL ERROR HAS BEEN ENCOUNTERED DURING THE CROSS SECTION PROCESSING.</td>
</tr>
<tr>
<td>INPRO1</td>
<td>10</td>
<td>THERE IS A SYN32 CARD TYPE NUMBER GREATER THAN THE LARGEST DEFINED.</td>
</tr>
</tbody>
</table>
APPENDIX F. SY3D ERROR MESSAGES. CONTINUED.

INPRO1 28  THERE ARE MORE THAN ONE TYPE 2 CARDS. ALL BUT THE FIRST ARE IGNORED.

INPRO1 34  THE BOUNDARY CONDITIONS HAVE BEEN CHANGED SINCE THE INPUT INTEGRALS WERE CALCULATED. SOME OF THE INTEGRALS MAY NO LONGER BE CORRECT.

INPRO1 44  THERE ARE MORE THAN ONE TYPE 3 SYNFIL CARDS. ALL BUT THE FIRST ARE IGNORED.

INPRO1 62  THERE IS NO TYPE 5 SYNFIL CARD. IT IS REQUIRED.

INPRO1 68  A REQUIRED GEOMETRY FILE IS MISSING.

INPRO1 72  A REQUIRED GEOMETRY FILE IS MISSING.

INPRO1 74  THE LABEL ON A GEOMETRY FILE DOES NOT MATCH THE LABEL ON THE FILE WHEN THE INPUT INTEGRALS WERE CALCULATED. IF THE GEOST FILE HAS BEEN RECONSTRUCTED ARE YOU SURE NO CHANGES HAVE BEEN MADE IN THE CONTENT.

INPRO1 76  THE LOWER X BOUNDARY CONDITION READ FROM AN INPUT GEOST FILE IS NOT ALLOWED OR, IN THE CASE OF PERIODIC CONDITIONS, IS INCONSISTENT.

INPRO1 78  THE UPPER X BOUNDARY CONDITION READ FROM AN INPUT GEOST FILE IS NOT ALLOWED OR, IN THE CASE OF PERIODIC CONDITIONS, IS INCONSISTENT.

INPRO1 80  THE LOWER Y BOUNDARY CONDITION READ FROM AN INPUT GEOST FILE IS NOT ALLOWED OR, IN THE CASE OF PERIODIC CONDITIONS, IS INCONSISTENT.

INPRO1 82  THE UPPER Y BOUNDARY CONDITION READ FROM AN INPUT GEOST FILE IS NOT ALLOWED OR, IN THE CASE OF PERIODIC CONDITIONS, IS INCONSISTENT.

INPRO1 104  THE BOUNDARY SYMBOL ON A TYPE 6 SYNFIL CARD IS UNRECOGNIZABLE.

INPRO1 108  THE GROUP NUMBERS ON A TYPE 6 SYNFIL CARD ARE EITHER OUT OF SEQUENCE OR EXCEED THE TOTAL NUMBER OF GROUPS.

INPRO1 140  THERE IS NO FUNCTION NAME ON THE FIRST SYNFIL TYPE 7 CARD.

INPRO1 148  A REQUIRED EXPANSION FUNCTION FLUX FILE IS MISSING.

INPRO1 154  A REQUIRED EXPANSION FUNCTION FLUX FILE IS MISSING.

INPRO1 160  THE LABEL ON A FLUX FILE DOES NOT MATCH THE LABEL ON THE FILE WHEN THE INPUT INTEGRALS WERE CALCULATED. IF THE RTFLUX FILE HAS BEEN RECONSTRUCTED ARE YOU SURE NO CHANGES HAVE BEEN MADE IN THE CONTENT.

INPRO1 182  THERE IS NO FUNCTION NAME ON THE FIRST SYNFIL TYPE 8 CARD.

INPRO1 188  A REQUIRED WEIGHTING FUNCTION FLUX FILE IS MISSING.

INPRO1 194  A REQUIRED WEIGHTING FUNCTION FLUX FILE IS MISSING.

INPRO1 200  THE LABEL ON A FLUX FILE DOES NOT MATCH THE LABEL ON THE FILE WHEN THE INPUT INTEGRALS WERE CALCULATED. IF THE RTFLUX FILE HAS BEEN RECONSTRUCTED ARE YOU SURE NO CHANGES HAVE BEEN MADE IN THE CONTENT.
APPENDIX P. SYN3D ERROR MESSAGES. CONTINUED.

TPROC 243  PLANAR GEOMETRY FILES HAVE NOT BEEN DEFINED FOR ALL AXIAL MESH INTERVALS.
TPROC 256  WHEN THERE ARE INPUT INTEGRALS THE OLD GROUP COLLAPSING SCHEME IS AUTOMATICALLY USED.
TPROC 262  WHEN THERE ARE NO TYPE 9 CARDS THERE CANNOT BE ANY TYPE 10 CARDS.
TPROC 278  THE GROUP NUMBERS ON A TYPE 9 CARD ARE EITHER OUT OF SEQUENCE OR EXCEED THE TOTAL NUMBER OF GROUPS.
TPROC 294  SAME AS 278, NOT FOR A TYPE 10 CARD.
TPROC 298  THE NUMBER OF COLLAPSED GROUPS IS DIFFERENT FOR EXPANSION AND WEIGHING FUNCTIONS.
TPROC 302  MORE THAN ONE TYPE OF GROUP COLLAPSING IS SPECIFIED. IT CAN BE SPECIAL OR GENERAL, BUT NOT BOTH.
TPROC 304  A TYPE 2 CARD MUST BE PROVIDED WITH THE PARAMETER 1 GROUP SPECIFIED.
TPROC 312  THERE IS A WRONG AMOUNT OF DATA ON A TYPE 11 CARD.
TPROC 328  GENERAL GROUP COLLAPSING IS SPECIFIED FOR THE WEIGHING FUNCTIONS BUT NOT FOR THE EXPANSION FUNCTIONS.
TPROC 334  THERE IS A WRONG AMOUNT OF DATA ON A TYPE 12 CARD.
TPROC 344  THE FUNCTION NAME ON A SYNPLT TYPE 13 CARD DOES NOT MATCH ANY FUNCTION SPECIFIED ON CARD TYPES 7 AND 8.
TPROC 354  THE GROUP NUMBERS ON A TYPE 13 CARD ARE ENTERED INCORRECTLY.
TPROC 392  THERE ARE INPUT INTEGRALS, AND SCALING IS SPECIFIED ON TYPE 14 AND/OR 15 CARDS. MAKE SURE THE SCALING IS THE SAME AS WHEN THE INPUT INTEGRALS WERE CALCULATED.
TPROC 394  THE FUNCTION NAME ON A TYPE 14 OR 15 CARD DOES NOT MATCH ANY FUNCTIONS SPECIFIED ON CARD TYPES 7 AND 8.
TPROC 398  SCALING IS SPECIFIED FOR FUNCTIONS ASSOCIATED WITH THE INPUT INTEGRALS. THE ORIGINAL SCALING FACTORS WILL BE USED IF MORE INTEGRALS ARE DONE.
TPROC 392  A GROUP NUMBER IS OUT OF RANGE ON A TYPE 14 OR 15 CARD.
TPROC 610  THE SCATTERING RADIUS HAS CHANGED SINCE THE INPUT INTEGRALS WERE CALCULATED. THEY CANNOT BE USED.
TPROC 62  A REQUIRED FLUX FILE IS NOT AVAILABLE.
TPROC 64  A REQUIRED GEOMETRY FILE IS NOT AVAILABLE.
TPROC 60  THERE IS NOT ENOUGH CONTAINER STORAGE TO HOLD AN INPUT FLUX FILE.
APPENDIX F. SYNTH ERROR MESSAGES. CONTINUED.

INTEG 12
THESE IS NEITHER AN INPUT INTTCC DATASET NOR A
SYNFL DATASET.

INTEG 14
THESE IS NO MACROSCOPIC CROSS SECTION DATASET.

INTEG 34
THE INITIAL STORAGE ALLOCATIONS ASSIGNED BY THE
CODE TO SEVERAL ARRAYS ARE FAR TOO SMALL. THIS IS
UNUSUAL. CHECK YOUR INPUT.

INTEG 116
NOT ENOUGH STORAGE IS AVAILABLE TO DO THE INTEGRATION.

INTERT 10100
A MATRIX IS SINGULAR. POSSIBLE LINEAR DEPENDENCE
PROBLEM.

INT1 110
THE CODE CANNOT YET HANDLE THIS PARTICULAR
GEOMETRY TYPE.

INT1 147
CODE RUG.

INT1 149
CODE RUG.

INT1 190
CODE BUG.

INT2 21
CODE RUG.

INT2 74
CODE MUG.

INT2 10
CODE BUG.

INT21 1
THE CODE CANNOT YET HANDLE THIS PARTICULAR
GEOMETRY TYPE.

INT21 14
CODE MUG. THERE ARE MISSING COMPOSITIONS IN THE
MATERIAL MAP.

INT21 40
SAME AS 14.

INT21 90
SAME AS 14.

INT21 106
SAME AS 14.

INT21 132
SAME AS 14.

INT21 148
SAME AS 14.

INT21 166
SAME AS 14.

INT21 180
SAME AS 14.

INT21 198
SAME AS 14.

INT3 110
THE CODE CANNOT YET HANDLE THIS PARTICULAR GEOMETRY
TYPE.

INT3 142
CODE RUG.

INT3 148
CODE RUG.

INT3 180
CODE RUG.

INT3 194
CODE RUG.

INT4 14
CODE RUG.

INT4 104
CODE PIC.

DRITH 110
NO FUNCTIONS HAVE BEEN SPECIFIED FOR ONE OF THE
ENERGY GROUPS AT SOME MESH POINT.

OUTPRO 10
A COMBINING COEFFICIENT FILE (ACCOFF OR DCCOFF)
CANNOT BE WRITTEN.

PDENOM 04
A REQUIRED INTEGRAL IS NOT AVAILABLE.

RECMOD 14
THE CARDS-PER-CARD-TYPE DATA IS TOO LONG.

REDN 16
A DEFECT COMBINING COEFFICIENT PLOT IS REQUIRED
FOR A GROUP NUMBER LARGER THAN THE TOTAL
NUMBER OF GROUPS (AFTER COLLAPSING).

REDN 26
AN ADJOINT COMBINING COEFFICIENT PLOT IS REQUIRED
FOR A GROUP NUMBER LARGER THAN THE TOTAL
NUMBER OF GROUPS (AFTER COLLAPSING).

RPPP 9
THE PROBLEM FILE CANNOT BE WRITTEN.

REXIT 224
CODE RUG.

REXIT 222
CODE BUG. LENST, THE RECORD LENGTH FOR REXIT, IS TOO SMALL.

REXIT 110
THE REXIT FILE CANNOT BE WRITTEN.
APPENDIX F. SYN3D ERROR MESSAGES. CONTINUED.

<table>
<thead>
<tr>
<th>CODE</th>
<th>MESSAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>RTGEO 10</td>
<td>A REQUIRED INPUT GEOST FILE CANNOT BE FOUND.</td>
</tr>
<tr>
<td>RTGEO 40</td>
<td>THE CODE CANNOT HANDLE THE GEOMETRY TYPE SPECIFIED IN AN INPUT GEOST FILE.</td>
</tr>
<tr>
<td>RTGEO 96</td>
<td>SAME AS 10.</td>
</tr>
<tr>
<td>RTGEO 100</td>
<td>TWO INPUT GEOST FILES ARE INCOMPATIBLE.</td>
</tr>
<tr>
<td>RTGEO 102</td>
<td>THE OUTPUT GEOST FILE CANNOT BE WRITTEN.</td>
</tr>
<tr>
<td>SETUP 10</td>
<td>THE VOLT,2 FILE CANNOT BE WRITTEN.</td>
</tr>
<tr>
<td>SETUP 14</td>
<td>THE DIPINT,2 FILE CANNOT BE WRITTEN.</td>
</tr>
<tr>
<td>SETUP 20</td>
<td>THE VOLT,1 FILE CANNOT BE FOUND.</td>
</tr>
<tr>
<td>SETUP 30</td>
<td>THE DIPINT,1 FILE CANNOT BE FOUND.</td>
</tr>
<tr>
<td>SHUFFL 26</td>
<td>CODE BUG.</td>
</tr>
<tr>
<td>SHUFFL 56</td>
<td>CODE BUG.</td>
</tr>
<tr>
<td>SHUFFL 114</td>
<td>CODE BUG.</td>
</tr>
<tr>
<td>SHUFFL 126</td>
<td>CODE 37G.</td>
</tr>
<tr>
<td>SOLVE 10</td>
<td>THE INTTOC,2 FILE CANNOT BE READ.</td>
</tr>
<tr>
<td>SOLVE 66</td>
<td>CODE BUG. INTV,np.1 AFTER DIVVY.</td>
</tr>
<tr>
<td>SOLVE 72</td>
<td>THE INTSLS FILE CANNOT BE WRITTEN.</td>
</tr>
<tr>
<td>SOLVE 74</td>
<td>THE VOLT,2 FILE CANNOT BE READ.</td>
</tr>
<tr>
<td>SOLVE 76</td>
<td>THE DIPINT,2 FILE CANNOT BE READ.</td>
</tr>
<tr>
<td>SOLVE 86</td>
<td>THE HFILE FILE CANNOT BE WRITTEN.</td>
</tr>
<tr>
<td>TRANS! 1</td>
<td>A GEOST FILE CANNOT BE WRITTEN.</td>
</tr>
<tr>
<td>TRNINT 1</td>
<td>AN RSSLX FILE CANNOT BE WRITTEN.</td>
</tr>
<tr>
<td>TRNINT 14</td>
<td>THE TRANSLATION CANNOT YET HANDLE THIS TYPE OF GEOMETRY.</td>
</tr>
<tr>
<td>TRNINT 60</td>
<td>THE TRANSLATOR CANNOT HANDLE THE PLANAR BOUNDARY CONDITIONS.</td>
</tr>
<tr>
<td>TRGFO1 70</td>
<td>SAME AS 60.</td>
</tr>
<tr>
<td>TRXSCM 12</td>
<td>THE XS.C.MIN FILE CANNOT BE FOUND.</td>
</tr>
<tr>
<td>UNINTL 18</td>
<td>CODE BUG. MULTICHANNEL SYNTHESIS NOT AVAILABLE.</td>
</tr>
<tr>
<td>UNINTL 110</td>
<td>CODE BUG. MULTICHANNEL SYNTHESIS NOT AVAILABLE.</td>
</tr>
<tr>
<td>UNINTL 126</td>
<td>NO FUNCTIONS HAVE BEEN SPECIFIED FOR ONE OF THE ENERGY GROUPS AT SOME MESH POINT.</td>
</tr>
<tr>
<td>UPITXS 20</td>
<td>THERE IS SOMETHING WRONG WITH THE XSCHM DATAFILE. THE CODE EXPECTED TO FIND THE GROUP NUMBER IN THE RECORD BEING READ.</td>
</tr>
</tbody>
</table>
APPENDIX G.  FORTRAN VARIABLE GLOSSARY.

\[(A('',K),I=1,KDIM(T')),K=1,KMAX) =  IN OVERLAY SOLVE, EITHER 
THE DIRECT OR ADJOINT COMBINING COEFFICIENTS.  IZ IS THE 
SYNTHESIS ZONE ASSOCIATED WITH MESH INTERVAL K.
\]
\[(ACOEF(I,K),I=1,KDIM(IZ)),K=1,KMAX) = ADJOINT COMBINING 
COEFFICIENTS.  IZ IS THE SYNTHESIS ZONE ASSOCIATED WITH 
MESH INTERVAL K.
\]
\[(AMNS(I,J),I=1,KDIM(IZ)),J=1,KDIM(IZM)) = THE SUBBLOCK (FOR THE 
CURRENT AXIAL MESH INTERVAL) BELOW THE DIAGONAL OF THE 
BLOCK-TRODIAGONAL SYSTEM OF SYNTHESIS EQUATIONS.  IZ IS THE 
SYNTHESIS ZONE ASSOCIATED WITH THE CURRENT MESH INTERVAL. 
IZM IS THE SYNTHESIS ZONE ASSOCIATED WITH THE PREVIOUS 
AXIAL MESH INTERVAL.
\]
\[(PLUS(I,J),I=1,KDIM(IZ)),J=1,KDIM(IZP)) = THE SUBBLOCK (FOR THE 
CURRENT AXIAL MESH INTERVAL) ABOVE THE DIAGONAL OF THE 
BLOCK-TRODIAGONAL SYSTEM OF SYNTHESIS EQUATIONS.  IZ IS THE 
SYNTHESIS ZONE ASSOCIATED WITH THE CURRENT MESH INTERVAL. 
IZP IS THE SYNTHESIS ZONE ASSOCIATED WITH THE NEXT 
AXIAL MESH INTERVAL.
\]
\[(ARRAY(T),I=1,36) = THE LIST OF NAMES OF DIMENSIONED VARIABLES 
DEFINED THROUGH BPOINTER.
\]
\[(AZERO('',I),I=1,KDIM(IZ)),J=1,KDIM(IZ)) = THE SUBBLOCK (FOR THE 
CURRENT AXIAL MESH INTERVAL) ON THE DIAGONAL OF THE 
BLOCK-TRODIAGONAL SYSTEM OF SYNTHESIS EQUATIONS.  IZ IS THE 
SYNTHESIS ZONE ASSOCIATED WITH THE CURRENT MESH INTERVAL.
\]
\[(BLC(T),I=1,MAXSIZ) = THE BPOINTER CONTAINER.
\]
\[(CARD(I,J),I=1,11),J=1,NREC) = THE CONTENTS OF EACH INPUT 
CARD, EXCLUDING THE CARD TYPE NUMBER.  NREC = THE NUMBER 
OF DATA CARDS.
\]
\[(CMC(I,J,K),I=1,NGROUP),J=1,6),K=1,2) = THE BOUNDARY CONDITION 
CONSTANTS C1 AND C2 FOR EACH GROUP AND EACH OF THE 6 
BOUNDARY PLANES.
\]
CONVUC = EIGENVALUE CONVERGENCE CRITERION.

\[(DCOEF(T,K),T=1,KDIM(IZ)),K=1,KMAX) = DIRECT COMBINING 
COEFFICIENTS.  IZ IS THE SYNTHESIS ZONE ASSOCIATED WITH "TH 
MESH INTERVAL K.
\]
\[(DI.7r AT(I),I= 1, LNML?) = RATIO OF 2 ADJACENT AXIAL MESH 
INTERVALS. (LOWER/UPPER)
\]
EIGEN = EIGENVALUE

EPS = A SMALL NUMBER USED IN TESTS FOR EQUALITY OF MESH INTERVALS.

\[(ESCAL('',J),I=1,NGROUP),J=1,MAXFUN) = A USER SUPPLIED SCALING 
FACTOR FOR EACH GROUP OF FUNCTION J WHEN FUNCTION 1 IS USED 
AS A EXPANSION FUNCTION.  IT CAN ONLY HAVE AN EFFECT WHEN 
GROUP COLLAPSING IS EMPLOYED.
\]
\[(FIS(T,K),I=1,KDIM(IZ)),J=1,KDIM(IZ)) = THE SUBBLOCK (FOR THE 
CURRENT AXIAL MESH INTERVAL) CONTAINING THE INTEGRALS OVER 
THE IONIZATION CROSS SECTIONS.  IZ IS THE SYNTHESIS ZONE 
ASSOCIATED WITH THE CURRENT AXIAL MESH INTERVAL.
\]
\[(FLUX(I),I=1,MAXFUN*GROUP) = AN ARRAY CONTAINING FUNCTION 
INTEGRALS.  THE INTEGRALS ARE STORED IN THE SAME "WAY AS THE 
COMBINING COEFFICIENTS, WITH PLANKS OMITTED.  POINTERS TO 
THE FIRST NONZERO GROUP ARE GIVEN IN THE ARRAY TPLOC.
APPENDIX G. FORTRAN VARIABLE GLOSSARY. (CONT'D.)

(FUX1(I),I=1,LENFLX) = THE ARRAY INTO WHICH A RECORD OF REQFLX IS READ. FUX2 AND FUX3 SERVE THE SAME PURPOSE.

(FLUX1(I),I=1,LENFLX) = THE ARRAY INTO WHICH A RECORD OF REQFLX IS READ. FLUX2 AND FLUX3 SERVE THE SAME PURPOSE.

(Guess) = EIGENVALUE ESTIMATE.

(((FNNAM(IJ),I=1,3),(IVRFUN(J)),J=1,MAXFUN) = THE MATRIX FORMED DURING THE FORWARD ELIMINATION OF THE SYNTHESIS EQUATIONS FOR THE CURRENT MESH INTERVAL. IT IS THE SYNTHESIS ZONE ASSOCIATED WITH THE CURRENT AXIAL MESH INTERVAL, IZP IS THE ZONE FOR THE NEXT INTERVAL.

(TBND) = THE NUMBER OF COLUMNS OF FLUXES IN ONE RECORD OF THE REQFLX FILE.

(TBXL) = THE LOWER X BOUNDARY CONDITION.

1 = ZERO FLUX

2 = REFLECTIVE

3 = EXTRAPOLATED C*D*DEL PHI + PHI = 0

4 = PERIODIC WITH OPPOSITE BOUNDARY

5 = PERIODIC WITH NEXT BOUNDARY GOING CLOCKWISE

6 = PERIODIC WITH NEXT BOUNDARY GOING COUNTERCLOCKWISE

7 = PERIODIC, INVERTED ALONG SAME BOUNDARY.

(TBXR) = THE UPPER X BOUNDARY CONDITION.

(TBCy) = THE LOWER Y BOUNDARY CONDITION.

(TBCy) = THE UPPER Y BOUNDARY CONDITION.

(TBCZI) = THE LOWER Z BOUNDARY CONDITION.

(TBCZU) = THE UPPER Z BOUNDARY CONDITION.

(IBNDCH) = LOWEST ENERGY GROUP (AFTER GROUP COLLAPSING) FOR WHICH THERE IS A NONZERO CHI.

((TRNSPT(I),I=1,MAXPEN) = LOWEST ENERGY GROUP (AFTER GROUP COLLAPSING) FOR WHICH THERE IS A NONZERO FISSION CROSS SECTION FOR PLANE I. A ZERO MEANS NO FISSION.

((TRNS2(I),I=1,ICOLPEN) = THE POSITION IN COLUMN T OF THE GROUP COLLAPSED SCATTERING MATRIX OF THE TOPMOST, NONZERO ELEMENT. EQUIVALENT TO LCCST WHEN THERE IS NO GROUP COLLAPSING.

((TRNS3(I),I=1,ICOLPEN) = THE POSITION IN COLUMN T OF THE GROUP COLLAPSED SCATTERING MATRIX OF THE BOTTOM, NONZERO ELEMENT.

(IDIFTR(F,K),F=1,8,K=1,LDIFTR) = TABLE OF CONTENTS FOR DIFTR INTEGRALS IN DIFTR FILE.

IDIFTR(1,K) = RECORD OF THE DIFTR FILE CONTAINING THE INTEGRALS.

IDIFTR(2,K) = POINTER TO THE FIRST WORD OF THE INTEGRAL BLOCK WITHIN THAT RECORD.

IDIFTR(3,K) = LOWER PLANAR GEOMETRY IDENTIFIER.

IDIFTR(4,K) = LOWER PLANAR GEOMETRY IDENTIFIER.

IDIFTR(5,K) = MESH INTERVAL RATIO IN THE NLZ RAT ARRAY.

IDIFTR(6,K) = WEIGHTING FUNCTION IDENTIFIER.

IDIFTR(7,K) = EXPANSION FUNCTION IDENTIFIER.

IDIFTR(9,K) = NUMBER OF RECORDS REQUIRED FOR ALL THE INTEGRALS OF THIS COMBINATION.
APPENDIX G. FORTRAN VARIABLE GLOSSARY. (CONT'D.)

(1DIM (I), I=1, MAXFUN) = THE NUMBER OF COMBINING COEFFICIENTS (OR EQUATIONS) ASSOCIATED WITH AN EXPANSION FUNCTION (OR WEIGHTING FUNCTION).

((IPL (J, I), J=1, MAXKNM), I=1, NUMK7N) = THE EXPANSION FUNCTIONS USED IN ZONE I. IF THERE ARE FEWER THAN MAXKNM, THE OTHER ENTRIES FOR A PARTICULAR ZONE ARE ZERO.

(TFLOC (I), I=1, MAXFUN) = POINTERS TO THE FIRST GROUP FLUX INTEGRAL FOR FUNCTION I IN THE ARRAY FLUX. A ZERO MEANS THAT THERE ARE NO INTEGRALS FOR A FUNCTION.

IGEOM = PLANAR GEOMETRY TYPE SENTINEL.

((TGUSE(N, J), N=1, LGROUP), J=1, MAXFUN) = A PROJECTION MATRIX (ELEMENTS ARE 0 OR 1) SHOWING WHICH GROUP FLUXES OR EACH FUNCTION ARE TO BE USED (1) OR OMITTED (0).

TMAX = THE NUMBER OF MESH INTERVALS IN THE X DIRECTION.

(INIGLS (I), I=1, NCONRC*MAXNT) = STORAGE USED FOR RECORDS OF THE INIGLS FILE. DURING THE SOLUTION NCONRC=1.

(TPLACC (I), I=1, NPLACC) = THE GROUP FOR WHICH THE ADJOINT COMBINING COEFFICIENTS ARE TO BE PLOTTED.

(TPLDC (I), I=1, NPLDC) = THE GROUP FOR WHICH THE DIRECT COMBINING COEFFICIENTS ARE TO BE PLOTTED.

((TPRATF (I, J), I=1, 2), J=1, NRPATF) = GROUP NUMBERS (I=1) AND AXIAL MESH INTERVALS (I=2) FOR WHICH THE PLANAR ADJOINT IS TO BE EDITED. FOR TWO DIMENSIONAL MODELS J=1 WILL EDIT ALL FLUXES.

TPRINT = POINTERS PRINT EDIT SENTINEL.

((TPRWT (I, J), I=1, 2), J=1, NRPWT) = AXIAL MESH INTERVALS FOR WHICH THE PLANAR POWER IS TO BE EDITED. FOR TWO DIMENSIONAL MODELS I=1 WILL EDIT ALL POWERS.

((TPRPWT (I, J), I=1, 2), J=1, NRPWT) = GROUP NUMBERS (I=1) AND AXIAL MESH INTERVALS (I=2) FOR WHICH THE PLANAR Flux IS TO BE EDITED. FOR TWO DIMENSIONAL MODELS J=1 WILL EDIT ALL FLUXES.

((TT (I, J), I=1, LGROUP), J=1, 2) = THE NUMBER OF POIS OF THE KN MARİX STORED IN THE OUTPUT INTEGRAL FILE FOR THE COLUMNS ASSOCIATED WITH COLLAPSED GROUP I. IT IS THE SUM OF THE NUMBERS OF CHANNELS ASSOCIATED WITH EACH COLLAPSED GROUP BETWEEN TBINDS1(I) AND TBINDS2 (I). J=1 OR 2 DEPENDING ON WHETHER THE WEIGHTING/EXPANSION FLUX CONTINUATION BEING CONSIDERED IS IN THE PRECEP ORDER FOR THE OUTPUT INTEGRAL DESTINED OR TRANSPOSED.

((TFLOC (I, J), I=1, MAXFUN), J=1, MAXPLN) = POINTERS TO THE FIRST GROUP POWER INTEGRAL FOR FUNCTION I AND PLANE J IN THE ARRAY POL. A ZERO MEANS THAT THERE ARE NO INTEGRALS.

TPCNP = DIFF. PILF RECORD COUNTER.

TPCNT = INIGLS PILF RECORD COUNTER.

TPCUTC = INITOC PILF RECORD COUNTER.

TPCVOL = VOLTINT PILF RECORD COUNTER.

TSCNT = SET-WTDE PROMPT CH pt SENTINEL FROM XSCMIN PILF.
APPENDIX G. FORTRAN VARIABLE GLOSSARY. (CONT'D.)

((IVOLTB(I,K),T=1,7),K=1,IVOLTB) = TABLE OF CONTENTS FOR REM AND FTS INTEGRALS IN THE VOLINT FILE.

IVOLTR(1,K) = 1 FOR A REM INTEGRAL, 2 FOR A FTS INTEGRAL, 3 FOR A POW INTEGRAL, 4 FOR A FLUX INTEGRAL.

IVOLTB(2,K) = THE RECORD OF THE VOLINT FILE CONTAINING THE INTEGRAL. = 0 IF THE INTEGRAL COULD NOT BE DONE AND -1 IF THE INTEGRAL COULD HAVE BEEN DONE BUT FOR SOME REASON WAS NOT.

IVOLTR(3,K) = POINTER TO THE FIRST WORD OF THE INTEGRAL BLOCK WITHIN THAT RECORD.

IVOLTB(4,K) = PLANNAR GEOMETRY IDENTIFIER.

IVOLTB(5,K) = WEIGHTING FUNCTION IDENTIFIER.

IVOLTB(6,K) = EXPANSION FUNCTION IDENTIFIER.

IVOLTB(7,K) = NUMBER OF RECORDS REQUIRED FOR ALL THE INTEGRALS OF THIS COMBINATION.

(TVOLTN(J),J=1,MAXVOL) = VERSION NUMBER OF FILE FOR FUNCTION J. SEE VNNAM.

(TVOLTPN(J),J=1,MAXPLP) = VERSION NUMBER OF FILE FOR PLANAR GEOMETRY J. SEE FNNAM.

THERE = 1 IF THE INPUT DATA IS TO BE READ DIRECTLY FROM CARDS.

= 2 IF THE CARD INPUT IS TO BE PROCESSED THRU THE ARC SYSTEM ROUTINES.

((IP1(J,1),J=1,MAXVOL),T=1,N1) = THE WEIGHTING FUNCTIONS USED IN ZONE I. IF THERE ARE FEWER THAN MAXVOL, THE OTHER ENTRIES FOR A PARTICULAR ZONE ARE ZERO.

(TLLEN(I),I=1,53) = THE WORD LENGTH PARAMETER FOR EACH DIMENSIONED VARIABLE DEFINED THROUGH BTOFINP. THIS ARRAY IS OF DIFFERENT LENGTH IN EACH OVERLAY.

JBRND = THE NUMBER OF ROWS OF FLUXES IN ONE RECORD OF THE REQFLX FILE.

(J1TR(I,K),I=1,7),K=1,IVOLTB) = TABLE OF CONTENTS FOR FTS INTEGRALS IN THE INTCLS FILE. THE DEFINITIONS ARE THE SAME AS FOR IDIPT. EXCEPT THAT THE LAST ENTRY IS OMITTED.

(JDCOLM(T),T=1,NUMCMF) = THE LIST OF COMPOSITIONS REQUIRED TO CALCULATE THE INTEGRALS.

(JDCOL(M,T),T=1,NUMCM) = THE LIST OF FUNCTIONS REQUIRED TO CALCULATE THE INTEGRALS, IN THE ORDER IN WHICH THEY APPEAR IN THE REQFLX FILE. WHEN IT IS PRESENT, THE UNIT FUNCTION IS LAST IN THE LIST.

(JDCOLP(T),T=1,NUMPN) = THE LIST OF PLANAR GEOMETRIES REQUIRED TO CALCULATE THE INTEGRALS.

JMXY = THE NUMBER OF MESH INTERVALS IN THE Y DIRECTION.

(((JPT(I,J,K),I=1,IVOLBP),K=1,MAXPN),J=1,2) = A POINTER IN A COLUMN OF THE GROUP CCLAPSF D FTS INTEGRAL FOR PLAN K TO THE FIRST ELEMENT ASSOCIATED WITH WEIGHTING FUNCTION GROUP I. THE DIFFERENCE BETWEEN JPT(I,J,K) AND JPT(I+1,J,K) IS THE NUMBER OF CHANNELS ASSOCIATED WITH GROUP I OF THE WEIGHTING FUNCTION. J = 1 OR 2 DEPENDING ON WHETHER THE WEIGHTING/EXPANSION FLUX COMBINATION BEING CONSIDERED IS IN THE PROPER ORDER FOR THE OUTPUT INTEGRAL DESIRED OR IS TRANPOSED.
APPENDIX G. PORT RAN VARTARLF GLOSSARY. (CONT'D.)

(JVOLTB(I,K),I=1,6,K=1,NVOLTB) = TABLE OF CONTENTS FOR REM & PIS
  INTEGRALS IN THE INTGLS FILE. THE DEFINITIONS ARE THE SAME
  AS FOR JVOLTB, EXCEPT THAT THE LAST ENTRY IS OMITTED.
(KCHANG(I),I=1,NUMKZN) = THE TOP MOST MESH INTERVAL IN EACH ZONE.
(KNMB(I),I=1,NUMKZN) = THE NUMBER OF UNKNOWN COMBINING
  COEFFICIENTS (AND THEREFORE EQUATIONS) ASSOCIATED WITH
  ONE MESH INTERVAL IN ZONE I.
(KGEO(I),I=1,NUMKZN) = THE GEOMETRY FOR MESH INTERVALS
  KCHANG(I-1) TO KCHANG(I). KCHANG(0) IS DEFINED TO BE 1.
(KMAX) = NUMBER OF AXIAL MESH INTERVALS.
(KMMPT(I),I=1,NUMKZN) = THE NUMBER OF FLUX FILES REFERENCED
  AS EXPANSION FUNCTIONS FOR ZONE I.
(KMMPT(I),I=1,NUMKZN) = THE NUMBER OF FLUX FILES REFERENCED
  AS WEIGHTING FUNCTIONS FOR ZONE I.
(((KPT(I,J,K),I=1,LGROUP),J=1,LGROUP),K=1,2) = A POINTER IN
  THE 1D ARRAY CONTAINING THE 2D, GROUP-COLLAPSED REM INTEGRAL
  TO THE FIRST STORED ELEMENT OF THE INTEGRALS ASSOCIATED
  WITH WEIGHTING FUNCTION GROUP I AND EXPANSION FUNCTION GROUP
  J. THE DIFFERENCE BETWEEN KPT(I,J,K) AND KPT(I,J,K+1) IS THE
  NUMBER OF CHANNELS FOR WEIGHTING FUNCTION GROUP I. K=1 OR 2
  DEPENDING ON WHETHER THE WEIGHTING/EXPANSION FLUX
  COMBINATION BEING CONSIDERED IS IN THE PROPER ORDER FOR THE
  OUTPUT INTEGRAL DESIRED OR IS TRANSPOSED.
(IPDMTB) = NUMBER OF ENTRIES IN THE IDITFB TABLE.
(LENA) = THE STORAGE REQUIREMENTS FOR THE VARIOUS COEFFICIENTS.
(LENAPR(I),I=1,50) = THE LENGTH OF EACH DIMENSIONED VARTARLF
  DEFINED THROUGH BPRINTER. THIS ARRAY IS OF DIFFERENT LENGTH
  IN EACH OVERLAY.
(LENCRD(I),I=1,MPREC) = THE LENGTH (IN REAL WORDS) OF EACH DATA
  RECORD OF SYMON. NREC = THE NUMBER OF DATA RECORDS.
(LENDIF(I),I=1,MRECIF) = LENGTH OF RECORD I OF DIFT11T FILE.
(LENSIZ) = LENGTH OF DIZNAT ARRAY.
(LENFLX) = RECPFD SIZE FOR PFOFLX.
(LENFLX) = LENGTH OF THE ARRAY FLUX IN OVERLAY SOLVE.
(LENN) = MAXIMUM RECORD SIZE FOR THE NPFLX FILE.
(LENT) = THE MAXIMUM RECORD LENGTH FOR VOLTNT AND DIFT11T INTEGRAL
  FILES.
(LENPST) = THE MAXIMUM LENGTH OF THE LIST OF INTEGRALS, LSTIV.
(LENT) = LENGTH OF THE ARRAY POT IN OVERLAY SOLVE.
(LENERCH(I),I=1,NRECCH) = THE LENGTH OF EACH记录 OF THE WTFLX FILE.
(LENERC(I),I=1,NRECCH) = THE LENGTH OF EACH RECORD OF THE INTGLS
  FILE.
(LENERC(I),I=1,NREC) = LENGTH OF RECORD I OF THE VOLTNT FILE.
(LENX) = RECPFD SIZE FOR CROSS SECTION FILE PM0ST.
(LEN1) = IRAND*JBAND = THE MAXIMUM NUMBER OF MESH BLOCKS
  REPRESENTED IN EACH RECORD OF PFOFLX.
(LEN2) = LEN1*IRANGL
(LEN) = LYST1.
(LGROUP) = NUMBER OF ENERGY GROUPS AFTER COLLAPSING.
(LYST1) = A LIST OF ENTRIES IN JVOLTB AND/OR IDITFB
  DESIGNATING INTEGRALS TO BE DONE AND THE ORDER IN WHICH THEY
  ARE TO BE DONE. THE JVOLTB ENTRIES ARE LISTED FIRST, THEN
  THE IDITFB ENTRIES. ALSO SEE NT2YV AND NT2YD.
APPENDIX C. PORTRAN VARIABLE GLOSSARY. (CONT'D.)

(LCSCCT(I),I=1,NGROUP) = THE POSITION IN COLUMN I OF THE SCATTERING MATRIX OF THE TOPMOST, NONZERO ELEMENT. EQUIVALENT TO IVBDS1 WHEN THERE IS NO GROUP COLLAPSING.

(((LPT(I,J,K),I=1,NGROUP),K=1,MAXPLN),J=1,2) = A POINTER TO THE 1D ARRAY CONTAINING THE 2D, GROUP-COLLAPSED FIS INTEGRAL FOR PLANE K TO THE TOP OF THE FIRST COLUMN ASSOCIATED WITH EXPANSION FUNCTION GROUP J. THE DIFFERENCE BETWEEN LPT(I,J,K) AND LPT(I'*1,J,K) IS NPT(J,F) TIMES THE NUMBER OF CHANNELS ASSOCIATED WITH GROUP I OF THE EXPANSION FUNCTION, J=1 OR 2 DEPENDING ON WHETHER THE WEIGHTING/EXPANSION FLUX COMBINATION BEING CONSIDERED IS IN THE PROPER ORDER FOR THE OUTPUT INTEGRAL DESIRED OR IS TRANSPOSED.

LREQQ = NUMBER OF RECORDS IN THE REQUEST FILE.
LREQQ1 = NUMBER OF RECORD OF THE REQUEST FILE EXCLUDING DIFFUSION COEFFICIENTS.
LREQQ2 = NUMBER OF RECORDS OF THE REQUEST FILE CONTAINING ONLY DIFFUSION COEFFICIENTS.
LVOLTB = NUMBER OF ENTRIES IN THE IVOLTB TABLE.
LXSST1 = THE NUMBER OF CROSS SECTION DATA PER COMPOSITION, FOR RECORDS OF REQUEST EXCLUDING DIFFUSION COEFFICIENTS.
LXSST2 = THE NUMBER OF COMPOSITIONS PER RECORD OF REQUEST, FOR RECORDS EXCLUDING DIFFUSION COEFFICIENTS.
LXSST3 = THE NUMBER OF GROUPS PER RECORD FOR RECORDS CONTAINING ONLY DIFFUSION COEFFICIENTS.
MAXCHN = THE MAXIMUM NUMBER OF CHANNELS IN ANY CHANNEL STRUCTURE.
MAXC = THE LARGEST VALUE OF VNCHN(I),I=1,NCHNST.
MACP = THE NUMBER OF MATERIALS REPRESENTED IN THE INPUT CROSS SECTION FILE.
MAXD = THE LARGEST DIMENSION OF ANY MATRIX USED IN THE SOLUTION.
MAXG = MAXIMUM NUMBER OF ONSCATTERING GROUPS.
MAXPN = NUMBER OF FUNCTION FILE NAMPS IN TIC.
MAXPNS = THE MAXIMUM RECORD LENGTH FOR THE INTEGRAL TEMPORAL FILE.
MAXPS = MAXIMUM NUMBER OF ITERATIONS ALLOWED.
MAXPN = THE MAXIMUM NUMBER OF WEIGHTING FUNCTIONS USED IN A SINGLE ZONE, OR THE MAXIMUM NUMBER OF EXPANSION FUNCTIONS, WHICHEVER IS LARGER. MAXPN.LE.MAXUSE.
MAXD = THE LARGEST DIMENSION OF AN INPUT MATRIX (THE LARGEST NUMBER OF GROUP/CHANNEL COMBINATIONS FOR A SINGLE FUNCTION).
MAXD2 = THE MAXIMUM SPACE REQUIRED FOR A RECORD OF PRINTED INTEGRALS.
MAXPLN = NUMBER OF PLANAR GEOMETRY FILE NAMPS IN TIC.
MAXREG = THE MAXIMUM NUMBER OF REGIONS ASSOCIATED WITH ANY OF THE PLANAR GEOMETRIES.
MAXS = THE LENGTH OF THE POINTED CONTAINER ARRAY.
MAXT = MAXIMUM ALLOWED NUMBER OF INPUT TITLE CARDS.
MAXU = MAXIMUM NUMBER OF ONSCATTERING GROUPS.
MAXUS = MAXIMUM NUMBER OF INPUT FUNCTIONS USED ANYWHERE IN THE MODEL AS EXPANSION FUNCTIONS OR WEIGHTING FUNCTIONS, WHICHEVER IS LARGER. MAXUS.LE.MAXPN.
MAXUS = NUMBER OF VARIABLES IN THE JOUSSTB TABLE.
(MAXI(I),I=1,MAXPN) = THE NUMBER OF COMBINING COEFFICIENTS (OR EQUATIONS) ASSOCIATED WITH FUNCTION I.
APPENDIX G. FORTRAN VARTABLE GLOSSARY. (CONT.)

\[ ((\text{NPT}(I,J), I=1, \text{MAXPN}), J=1, 2) = \text{the number of rows in the group collapsed flux integral for plane } I. \]
\[ \text{It is the sum of the number of channels associated with each collapsed group down to the last nonzero chf. } t=1 \text{ or 2 depending on whether the weighting/expansion flux combination being considered is in the proper order for the output integral desired or is transposed.} \]
\[ \text{NULT} = \text{the word length parameter for input/output.} \]
\[ \text{MVOLTS} = \text{number of entries in the JVOLTS table.} \]
\[ \text{NCALC} = \text{calculation type sentinel. see type 2 card of SYMFIL.} \]
\[ ((\text{NCAN}(N,J), N=1, \text{GROUP}), J=1, \text{MAXPN}) = \text{channel scheme assignment for each group-collapsed group flux.} \]
\[ \text{NCUST} = \text{the number of channel structure schemes.} \]
\[ \text{NCOGDP} = 0 \text{ for group dependent synthesis, 1 for general group collapsing, 2 for special group collapsing.} \]
\[ \text{NCVOC} = \text{the number of concurrent input files that can be held in core during the integral compile.} \]
\[ \text{NFCOM} = 0, \text{do not calculate perturbation denomenator.} \]
\[ \text{1, calculate perturbation denominator.} \]
\[ \text{MFDIF} = \text{the file number for the DIFF file.} \]
\[ \text{FLNLY} = \text{REMOTE file number.} \]
\[ (\text{NFLPN}(I), I=1, \text{MAXPN}) = \text{the file number for each flux file.} \]
\[ \text{NFLT} = \text{eight file number.} \]
\[ \text{NFLNT} = \text{integer file number.} \]
\[ \text{NFLTC} = \text{the file number for the IMTOC file.} \]
\[ (\text{NFLML}(I), I=1, \text{MAXPN}) = \text{the file number for each planar geometry file.} \]
\[ \text{NFLSY} = \text{SYNCH file number.} \]
\[ \text{NFLPT} = \text{a foreign file number for the unit flux.} \]
\[ \text{NFLUH} = \text{the file number for the VOLINT file.} \]
\[ \text{NFLVSC} = \text{XSVCH file number.} \]
\[ \text{NFLVST} = \text{PVOUT file number.} \]
\[ \text{NFLVOL} = \text{the number of functions represented in input input primal files.} \]
\[ \text{NGGQHT} = 0, \text{do not construct an output GEOQHT file.} \]
\[ \text{NGGQH} = \text{the version number for the output GGROUP file.} \]
\[ \text{NGC} = \text{number of energy groups OFF COLLAPSING.} \]
\[ \text{NQPT} = \text{output file number.} \]
\[ \text{NQPTC} = \text{the number of adjoint combining coefficient plots.} \]
\[ \text{NPLCO} = \text{the number of direct combining coefficient plots.} \]
\[ \text{NQLO} = \text{the number of planar geometries represented in input integral files.} \]
\[ \text{NWAPG} = \text{the number of planar adjoint group fluxes to be edited.} \]
\[ \text{NPRED} = \text{the number of planar power distributions to be edited.} \]
\[ \text{NPRED} = 0, \text{do not edit the average fluxes by zone.} \]
\[ \text{NPRED} = 0, \text{do not edit the average fluxes by zone (NPRED must be zero).} \]
\[ \text{NPTAKE}(I), I=1, 36) = \text{the pointer to each dimensioned variable defined through pointer.} \]
APPENDIX G. FORTRAN VARIABLE GLOSSARY. (CONT'D.)

(NPTOUT(I),I=1,2*LENLST) = A SCRATCH ARRAY CONTAINING POINTERS WITHIN THE OUTPUT RECORDS FOR EACH ENTRY OF LISTV.
(NPTXNS) = A POINTER TO THE LOCATION JUST BEFORE THE START OF THE MESH INTERVAL DATA IN A RECORD OF REQFLX.
(NPTXST(I),I=1,LENLST) = A SCRATCH ARRAY USED TO STORE POINTERS TO COMPOSITIONS FOR EACH ENTRY IN LISTV.
(NPTWDNT) = 0, DO NOT CONSTRUCT AN OUTPUT PWDINT FILE.
"GT. 0, THE VERSION NUMBER FOR THE OUTPUT PWDINT FILP.
(NPCDIF) = NUMBER OF RECORDS IN THE DIFINT FILE.
(NRLCH) = THE NUMBER OF RECORDS IN THE HFILE FILE.
(NRCINT) = THE NUMBER OF RECORDS IN THE INTGIS FILE.
(NRCVOL) = NUMBER OF RECORDS IN THE VOLINT FILE.
(NREC) = THE NUMBER OF RECORDS OF THE DIFINT FILE BEING CONSTRUCTED ON A PARTICULAR PASS THRU THE INTEGRATION LOOP.
(NREC2) = THE NUMBER OF RECORDS OF THE VOLINT FILE BEING CONSTRUCTED ON A PARTICULAR PASS THRU THE INTEGRATION LOOP.
(NREC3) = 0, DO NOT CONSTRUCT AN OUTPUT RTFLUX FILP.
"GT. 0, THE VERSION NUMBER FOR THE OUTPUT RTFLUX FILP.
(NSCALE) = 1 IF USER SUPPLIED SCALING FACTORS FSSCALE AND "SCALE ARE TO BE APPLIED, 0 IF NOT.
(NSEDIF) = FORMAT SENTINEL FOR DIFINT FILE.
(NSTTLE) = THE NUMBER OF 6W WORDS USED TO STORE THE TITLE.
(NTRANG) = TRIANGULAR MESH SENTINEL.
(NTRSCT) = SECONDARY TRIANGULAR MESH SENTINEL.
(NTPYD) = THE NUMBER OF ENTRIES IN LISTV WHICH PERTAIN TO THE IDPTS TABLE OF CONTENTS.
(NTPYV) = THE FIRST NTYV ENTRIES IN LISTV PERTAIN TO THE TVOLTB TABLE OF CONTENTS.
(NTPYXS) = MAXIMUM ALLOWED NUMBER OF INPUT CARD TYPES.
(NUMREC) = THE NUMBER OF OUTPUT INTEGRAL RECORDS THAT CAN BE CONTAINED IN CORF.
(INUMCHN(I),I=1,NCHNST) = NUMBER OF CHANNELS IN CHANNEL STRUCTURE SCHEME 1.
(NUMCM) = THE NUMBER OF MATERIALS FOR WHICH CROSS SECTIONS ARE REQUIRED WHEN THE INTEGRALS ARE CALCULATED (NUMCM.LE.MAXCM)
(NUMFN) = THE NUMBER OF FUNCTIONS REQUIRED TO CALCULATE THE INTEGRALS. IF THE UNIT FUNCTION IS REQUIRED, IT IS INCLUDED IN NUMFUN, EVEN THOUGH NO SUCH FLUX FILP IS EVER BOTTEN (NUMFN.LE.MAXFN).
(NUMKZ) = THE NUMBER OF AXIAL ZONES IN THE MODEL. THERE IS AN AXIAL ZONE WHEN EITHER THE PLANAR GEOMETRY OF THE FUNCTION CHANGES.
(NUMPLN) = THE NUMBER OF PLANAR GEOMETRIES REQUIRED TO CALCULATE THE INTEGRALS. (NUMPLN.LE.MAXPLN)
(N3DADJ) = 0, DO NOT CONSTRUCT AN OUTPUT ATFLUX FILP.
"GT. 0, THE VERSION NUMBER FOR THE OUTPUT ATFLUX FILP.
(N3DFLX) = 0, DO NOT CONSTRUCT AN OUTPUT RTFLUX FILP.
"GT. 0, THE VERSION NUMBER FOR THE OUTPUT RTFLUX FILP.
((CMFC(I,J),I=1,MAXDIM),J=1,NUMKZ) = THE FACTORS FOR EACH SYNTHETICS ZONE USED IN THE NORMALIZATION/DIFFRENCING TRANSFORMATION APPLIED TO THE SYNTHETICS EQUATIONS. FOR EACH ZONE THE FACTORS ARE STORED IN THE SAME MANNER AS THE CORRESPONDING COMBINING COEFFICIENT.
APPENDIX G. FORTRAN VARIABLE GLOSSARY. (CONT'D.)

(((PLNNAM(I,J),I=1,3),TVQPLN(J)),J=1,MAXPLN) = FULL NAME OF
FILE FOR PLANE GEOMETRY J.

(PWT(I),I=1,MAXFT1*LGROUP*MAXPLN) = AN ARRAY CONTAINING POWER
INTEGRALS FOR EACH FUNCTION AND PLAN. THE INTEGRALS ARE
STORED IN THE SAME WAY AS THE COMBINING COEFFICIENTS, WITH
PLANKS OMITTED. POINTERS TO THE FIRST NONZERO GROUP ARE
GIVEN IN THE ARRAY IP4TOC.

PWTPTS = WATTS PER FISSION.

(PP(T),T=1,LPN1) = A SCRATCH ARRAY USED IN CALCULATING THE
INTEGRALS. FOR A MULTICHANNEL CALCULATION 3*LEN1 IS RESERVED.

(((REXT(I,J),I=1,KDIM(I2)),J=1,KDIM(I2)) = THE SUBBLOCK (FOR THE
CURRENT AXIAL MESH INTERVAL) CONTAINING THE INTEGRALS OVER
THE REMOVAL CROSS SECTIONS. IZ IS THE SYNTHESIS ZONE
ASSOCIATED WITH THE CURRENT AXIAL MESH INTERVAL.

((RESULT(I,J),I=1,LENWT),J=1,NMREC) = ARRAYS FOR OUTPUT
INTEGRALS.

(TMP(T),T=1,10) = THE ARRAY IN THE ARGUMENT OF SUBROUTINE 
TMR.

(TITLE(I),I=1,NTITLE) = USER INPUT TITLE.

(((ST(I,J,K),I=1,LGROUP),J=1,NGROUP),K=1,2) = GENERAL GROUP
COLLAPSE MATRIX FOR THE EXPANSION FUNCTIONS (K=1) AND
WEIGHTING FUNCTIONS (K=2).

(VOL(I),I=1,LFN1) = A SCRATCH ARRAY FOR MESH BLOCK VOLUMES.

((SCALE(I,J),I=1,NGROUP),J=1,MAXFUN) = A USER SUPPLIED SCALING
FACTOR FOR EACH GROUP OF FUNCTION J WHEN FUNCTION J IS USED
AS A WEIGHTING FUNCTION. IT CAN ONLY HAVE AN EFFECT WHEN
GROUP COLLAPSE IS EMPLOYED.

XEPS = A SMALL NUMBER USED IN TESTS FOR ZERO CROSS SECTIONS.

(XSECT(I),I=1,LENXS1) = THE ARRAY INTO WHICH A RECORD OF REQIST 
IS READ. XSECT2 SERVES THE SAME PURPOSE.

(ZMESH(K),K=1,MAX) = POSITION OF UPPER BOUNDARY OF AXIAL MESH
INTERVAL K.
APPENDIX H. Sample Problem Output. Problem 1 - Complete Output, with Cross Section Edits

SYT3D, OVERLAY CARDS

THREE GROUP, FUNDAMENTAL MODE CALCULATION

LISTING OF SYT3D INPUT

CARDS-PFP-CARD-TYPE DATA

<table>
<thead>
<tr>
<th>CARD</th>
<th>TYPE</th>
<th>THREE GROUP, FUNDAMENTAL MODE CALCULATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>0.10000E+02 0.0 0.0</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>0.10000E+02 0.0 0.0 0.0</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>0.10000E+02 0.0 0.0 0.0</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>0.10000E+02 0.0 0.0 0.0</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>0.10000E+02 0.0 0.0 0.0</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>0.10000E+02 0.0 0.0 0.0</td>
</tr>
</tbody>
</table>

1/20/, 04.43., PAGE 1
SYN3D, OVERLAY HMG4C
THREE GROUP, FUNDAMENTAL MODE CALCULATION

* * * HMG4C - CCCC TO ARC SYSTEM CROSS SECTION HOMOGENIZATION * * *

FILE WBISBF IS LUN 40 AND CONTAINS THE USER ID -3D,3GP MODEL-
FILE 2MATCH IS LUN 41 AND CONTAINS THE USER ID -3D,3GP MODEL-
FILE ISOTIS IS LUN 39 AND CONTAINS THE USER ID -3D,3GP MODEL-
FILE COMPS IS NOW BE WRITTEN ON LUN 50
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>MCNP, Number of Compositions</td>
<td>7</td>
</tr>
<tr>
<td>NGPOMP, Number of Energy Groups</td>
<td>3</td>
</tr>
<tr>
<td>ISCH1, Prompt Fission Spectrum Flag</td>
<td>1</td>
</tr>
<tr>
<td>NFSCMP, Number of Fissionable Compositions</td>
<td>4</td>
</tr>
<tr>
<td>MACKUP, Maximum Number of Groups of Upscattering</td>
<td>0</td>
</tr>
<tr>
<td>MAAMD, Maximum Number of Groups of Downscattering</td>
<td>2</td>
</tr>
<tr>
<td>NDUP1, Reserved Variable</td>
<td>0</td>
</tr>
<tr>
<td>NDUP2, Reserved Variable</td>
<td>0</td>
</tr>
<tr>
<td>NDUP3, Reserved Variable</td>
<td>0</td>
</tr>
<tr>
<td>NDUP4, Reserved Variable</td>
<td>0</td>
</tr>
</tbody>
</table>

**Prompt Fission Spectrum (Set CHI Vector)**

1. 9.6731730-01  
2. 1.2664190-02  
3. 1.9472010-05
### STU3D, OVERLAY MNGAC

**THREE GROUP, FUNDAMENTAL NODE CALCULATION**

**FDIT OF (FISSIONABLE) COMPOSITION NO. 1**

**ICHX = -1**

<table>
<thead>
<tr>
<th>GROUP</th>
<th>ABSORPTION</th>
<th>TOTAL</th>
<th>REMOVAL</th>
<th>TRANSPORT</th>
<th>FISSION</th>
<th>NUP*FISSION</th>
<th>CHI</th>
<th>NUP</th>
<th>NDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.523270D-03</td>
<td>1.466634D-01</td>
<td>1.221477D-02</td>
<td>1.630343D-01</td>
<td>2.502560D-03</td>
<td>7.417706D-03</td>
<td>9.67317D-01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>5.771064D-03</td>
<td>2.573538D-01</td>
<td>6.52605D-03</td>
<td>2.508233D-01</td>
<td>1.908718D-03</td>
<td>5.708737D-03</td>
<td>3.266419D-02</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>2.428450D-02</td>
<td>3.129400D-01</td>
<td>2.428450D-02</td>
<td>3.556871D-01</td>
<td>7.002667D-03</td>
<td>2.235440D-02</td>
<td>1.947201D-05</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GROUP</th>
<th>POWER CONVERSION</th>
<th>DIRECTIONAL DIFFUSION COEFF. MULTIPLIER</th>
<th>DIRECTIONAL DIFFUSION COEFF. ADDITIVE TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FACTOR</td>
<td>DIMENSION 1</td>
<td>DIMENSION 2</td>
</tr>
<tr>
<td>1</td>
<td>8.000853D-14</td>
<td>1.00000D+00</td>
<td>1.00000D+00</td>
</tr>
<tr>
<td>2</td>
<td>6.349367D-14</td>
<td>1.00000D+00</td>
<td>1.00000D+00</td>
</tr>
<tr>
<td>3</td>
<td>2.496052D-13</td>
<td>1.00000D+00</td>
<td>1.00000D+00</td>
</tr>
</tbody>
</table>

**TOTAL SCATTERING CROSS SECTION**

<table>
<thead>
<tr>
<th>INTO GROUP/FROM GROUP CROSS SECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
</tbody>
</table>

**144**
### EDIT OF (FISSIONABLE) COMPOSITION NO. 2

**ICHI = -1**

<table>
<thead>
<tr>
<th>GROUP</th>
<th>ABSORPTION</th>
<th>TOTAL</th>
<th>REMOVAL</th>
<th>TRANSPORT</th>
<th>FISSION</th>
<th>NBF</th>
<th>FISSION</th>
<th>CHI</th>
<th>MUP</th>
<th>RDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.9477E+10</td>
<td>1.2276E+09</td>
<td>1.5700E+08</td>
<td>3.0516E+07</td>
<td>9.3945E+05</td>
<td>9.6731E+03</td>
<td>0</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>6.2897E+09</td>
<td>9.0069E+08</td>
<td>7.7645E+07</td>
<td>2.7056E+06</td>
<td>7.3844E+04</td>
<td>3.2664E+02</td>
<td>0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4.0077E+09</td>
<td>3.2618E+08</td>
<td>1.0000E+07</td>
<td>3.0397E+06</td>
<td>1.0000E+05</td>
<td>1.9672E+03</td>
<td>0</td>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GROUP</th>
<th>POWER CONVERSION FACTOR</th>
<th>DIRECTIONAL DIFFUSION COEFF. MULTIPLIER</th>
<th>DIRECTIONAL DIFFUSION COEFF. ADDITIVE TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DIMENSION 1</td>
<td>DIMENSION 2</td>
<td>DIMENSION 3</td>
</tr>
<tr>
<td>1</td>
<td>9.7658E+05</td>
<td>1.0000E+00</td>
<td>1.0000E+00</td>
</tr>
<tr>
<td>2</td>
<td>8.6582E+05</td>
<td>1.0000E+00</td>
<td>1.0000E+00</td>
</tr>
<tr>
<td>3</td>
<td>4.0077E+05</td>
<td>1.0000E+00</td>
<td>1.0000E+00</td>
</tr>
</tbody>
</table>

**TOTAL SCATTERING CROSS SECTION**

<table>
<thead>
<tr>
<th>INTO GROUP/FROM GROUP, CROSS SECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>Group</td>
</tr>
<tr>
<td>-------</td>
</tr>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Group</th>
<th>TOTAL SCATTERING CROSS SECTION INTO GROUP/FRONT GROUP CROSS SECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.000000D+00</td>
</tr>
<tr>
<td>2</td>
<td>1.000000D+00</td>
</tr>
<tr>
<td>3</td>
<td>1.000000D+00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Directional Diffusion Coeff. Multiplier</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.000000D+00</td>
</tr>
<tr>
<td>2</td>
<td>1.000000D+00</td>
</tr>
<tr>
<td>3</td>
<td>1.000000D+00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Dimension</th>
<th>Directional Diffusion Coeff. Additive Term</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.0</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
</tr>
<tr>
<td>3</td>
<td>0.0</td>
</tr>
</tbody>
</table>
## EDIT OF (FISSIONABLE) COMPOSITION NO. 4

**ICHI** = -1

<table>
<thead>
<tr>
<th>GROUP</th>
<th>ABSORPTION</th>
<th>TOTAL</th>
<th>REMOVAL</th>
<th>TRANSPORT</th>
<th>FISSION</th>
<th>NUC*FISSION</th>
<th>CHI</th>
<th>NUP</th>
<th>NDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.503615D-03</td>
<td>2.085197D-01</td>
<td>1.707075D-02</td>
<td>2.324194D-01</td>
<td>1.572191D-03</td>
<td>4.447421D-03</td>
<td>9.673173D-01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>6.456643D-03</td>
<td>3.597723D-01</td>
<td>7.595680D-03</td>
<td>3.464879D-01</td>
<td>1.499999D-02</td>
<td>0.0</td>
<td>3.266479D-02</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1.828476D-02</td>
<td>4.069950D-01</td>
<td>1.828476D-02</td>
<td>4.382179D-01</td>
<td>1.449999D-02</td>
<td>0.0</td>
<td>1.947201D-05</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GROUP</th>
<th>POWER CONVERSION FACTOR</th>
<th>DIRECTIONAL DIFFUSION COEFF. MULTIPLIER</th>
<th>DIRECTIONAL DIFFUSION COEFF. ADDITIVE TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DIMENSION 1</td>
<td>DIMENSION 2</td>
<td>DIMENSION 3</td>
</tr>
<tr>
<td>1</td>
<td>5.0310000D-14</td>
<td>1.000000D+00</td>
<td>1.000000D+00</td>
</tr>
<tr>
<td>2</td>
<td>3.639996D-63</td>
<td>1.000000D+00</td>
<td>1.000000D+00</td>
</tr>
<tr>
<td>3</td>
<td>3.639996D-63</td>
<td>1.000000D+00</td>
<td>1.000000D+00</td>
</tr>
</tbody>
</table>

**TOTAL SCATTERING CROSS SECTION**

<table>
<thead>
<tr>
<th>GROUP</th>
<th>PROD GROUP, CROSS SECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1, 2.076730D-01</td>
</tr>
<tr>
<td>2</td>
<td>1, 1.368120D-02 2, 1.56292D-01</td>
</tr>
<tr>
<td>3</td>
<td>1, 7.015717D-09 2, 1.051819D-03 3, 3.891484D-01</td>
</tr>
</tbody>
</table>
### ICRI = 0

<table>
<thead>
<tr>
<th>GROUP</th>
<th>ABSORPTION</th>
<th>TOTAL</th>
<th>REMOVAL</th>
<th>TRANSPORT</th>
<th>NP</th>
<th>ND</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.028794D-04</td>
<td>1.741392D-01</td>
<td>5.736323D-03</td>
<td>1.056762D-01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>1.077997D-03</td>
<td>3.577280D-01</td>
<td>2.075549D-03</td>
<td>3.793023D-01</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>7.891622D-03</td>
<td>6.468946D-01</td>
<td>7.891622D-03</td>
<td>6.468946D-01</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GROUP</th>
<th>POWER CONVERSION FACTOR</th>
<th>DIRECTIONAL DIFFUSION COEFF. MULTIPLIER</th>
<th>DIRECTIONAL DIFFUSION COEFF. ADDITIVE TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>DIMENSION 1</td>
<td>DIMENSION 2</td>
</tr>
<tr>
<td>1</td>
<td>0.0</td>
<td>1.000000D+00</td>
<td>1.000000D+00</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>1.000000D+00</td>
<td>1.000000D+00</td>
</tr>
<tr>
<td>3</td>
<td>0.0</td>
<td>1.000000D+00</td>
<td>1.000000D+00</td>
</tr>
</tbody>
</table>

**TOTAL SCATTERING CROSS SECTION**

<table>
<thead>
<tr>
<th>INTO GROUP/ FROM GROUP</th>
<th>CROSS SECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1, 1.703031D-01</td>
</tr>
<tr>
<td>2</td>
<td>1, 5.433439D-03</td>
</tr>
<tr>
<td>3</td>
<td>1, 6.916808D-09</td>
</tr>
</tbody>
</table>
### EDIT OF (NONFISSIONABLE) COMPOSITION NO. 6

**ICHI = 0**

<table>
<thead>
<tr>
<th>GROUP</th>
<th>ABSORPTION</th>
<th>TOTAL</th>
<th>REMOVAL</th>
<th>TRANSPORT</th>
<th>NUP</th>
<th>NDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.219654D-03</td>
<td>1.893785D-01</td>
<td>3.074153D-02</td>
<td>2.305605D-01</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>2.706030D-02</td>
<td>3.225606D-01</td>
<td>2.875027D-02</td>
<td>4.146590D-01</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1.772749D-01</td>
<td>5.525084D-01</td>
<td>1.772749D-01</td>
<td>6.175915D-01</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GROUP</th>
<th>POWER CONVERSION FACTOR</th>
<th>DIRECTIONAL DIFFUSION COEFF. MULTIPLIER</th>
<th>DIRECTIONAL DIFFUSION COEFF. ADDITIVE TERM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DIMENSION 1</td>
<td>DIMENSION 2</td>
<td>DIMENSION 3</td>
</tr>
<tr>
<td>1</td>
<td>0.0</td>
<td>1.00000D+00</td>
<td>1.00000D+00</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>1.00000D+00</td>
<td>1.00000D+00</td>
</tr>
<tr>
<td>3</td>
<td>0.0</td>
<td>1.00000D+00</td>
<td>1.00000D+00</td>
</tr>
</tbody>
</table>

**TOTAL SCATTERING CROSS SECTION**

<table>
<thead>
<tr>
<th>INTO GROUP/FROM GROUP, CROSS SECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1, 1.7477316D-01</td>
</tr>
<tr>
<td>2, 2.552287D-02, 3.09996D-01</td>
</tr>
<tr>
<td>3, 1.297796D-04, 1.68996D-03</td>
</tr>
</tbody>
</table>
EDIT OF NONFISSIONABLE COMPOSITION NO. 7

ICHI = 0

<table>
<thead>
<tr>
<th>GROUP</th>
<th>ABSORPTION</th>
<th>TOTAL</th>
<th>REMOVAL</th>
<th>TRANSPORT</th>
<th>NUP</th>
<th>NDN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.335810D-05</td>
<td>5.592931D-02</td>
<td>1.01986D-03</td>
<td>6.34839D-02</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>5.305912D-04</td>
<td>6.298489D-02</td>
<td>8.686827D-04</td>
<td>1.026733D-01</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1.309709D-04</td>
<td>6.623252D-02</td>
<td>1.345709D-04</td>
<td>7.422477D-02</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GROUP</th>
<th>POWER CONVERSION FACTOR</th>
<th>DIRECTIONAL DIFFUSION COEFF. MULTIPLIER</th>
<th>DIRECTIONAL DIFFUSION COEFF. ADDITIVE TERN</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>DIMENSION 1</td>
<td>DIMENSION 2</td>
</tr>
<tr>
<td>1</td>
<td>0.0</td>
<td>1.00000D+00</td>
<td>1.00000D+00</td>
</tr>
<tr>
<td>2</td>
<td>0.0</td>
<td>1.00000D+00</td>
<td>1.00000D+00</td>
</tr>
<tr>
<td>3</td>
<td>0.0</td>
<td>1.00000D+00</td>
<td>1.00000D+00</td>
</tr>
</tbody>
</table>

TOTAL SCATTERING CROSS SECTION

<table>
<thead>
<tr>
<th>INTO GROUP</th>
<th>FROM GROUP, CROSS SECTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1, 5.587675D-02</td>
</tr>
<tr>
<td>2</td>
<td>1, 4.078628D-03 2, 1.286808D-02</td>
</tr>
<tr>
<td>3</td>
<td>2, 4.176236D-04 3, 6.619230D-02</td>
</tr>
</tbody>
</table>
SIZE OF CONTAINER ALLOCATED FOR HMGNC - 5000
SIZE OF CONTAINER ACTUALLY USED BY HMGNC - 445

ELAPSED CPU TIME = 0.22 SEC.
ELAPSED PP TIME = 0.00 SEC.

* * * END OF HMGNC * * *
SYN3D, OVERLAY INTEGRATION
THREE GROUP, FUNDAMENTAL MODE CALCULATION

GENERAL PROBLEM DATA
NUMBER OF GROUPS = 3 (NGROUP)
BPOINTER CONTAINER SIZE = 5000
NUMBER OF AXIAL MESH INTERVALS = 3
THERE IS NO GROUP COLLAPSING
THERE IS NO INPUT FUNCTION SCALING
THERE WERE NO INPUT INTEGRALS

BOUNDARY CONDITIONS
LOWER X BOUNDARY CONDITION = ZERO CURRENT
UPPER X BOUNDARY CONDITION = ZERO CURRENT
LOWER Y BOUNDARY CONDITION = ZERO CURRENT
UPPER Y BOUNDARY CONDITION = ZERO CURRENT
LOWER Z BOUNDARY CONDITION = ZERO CURRENT
UPPER Z BOUNDARY CONDITION = ZERO CURRENT
**SYN3D, OVERTLAY INTEG**

**THREE GROUP, FUNDAMENTAL MODE CALCULATION**

<table>
<thead>
<tr>
<th>FUNCTION FILE</th>
<th>FILE NAME</th>
<th>GROUPS USED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>999</td>
<td>0</td>
</tr>
</tbody>
</table>

**AXIAL POSITION**

<table>
<thead>
<tr>
<th>GEOMETRY NUMBER</th>
<th>FILE NUMBER</th>
<th>GEOMETRY FILE NAME</th>
<th>EXPANSION FUNCTION NUMBERS</th>
<th>WEIGHTING FUNCTION NUMBERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0 TO 10.0</td>
<td>7</td>
<td>11</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**AXIAL MESH INTERVALS**

3.333D+00 3.333D+70 3.333D+00

**DATA MANAGEMENT PARAMETERS FOR INTEGRAL CALCULATION**

- **SPACE AVAILABLE DURING FLUX AND CROSS SECTION REWRITE = 7998**
- **REQFLX FILE, WORDS/RECORD = 226**
- **ROWS OF FLUXES/RECORD = 14**
- **REQST FILE, WORDS/RECORD = 15**
- **COMPOSITIONS/RECORD FOR FISSION-REMOVAL RECORDS = 1**
- **SPACE AVAILABLE DURING INTEGRATION = 2120**
- **VOLINT AND DIFINT FILES, WORDS/RECORD = 2000**

**NUMBER OF CONCURRENT RECORDS DURING REWRITE = 34**
**COLUMNS OF FLUXES/RECORD = 14**
**NUMBER OF CONCURRENT RECORDS DURING REWRITE = 513**
**GROUPS/RECORD FOR DIFFUSION COEFFICIENT RECORDS = 3**
**NUMBER OF CONCURRENT RECORDS DURING INTEGRATION = 1**
### VOLINT FILE

<table>
<thead>
<tr>
<th>ENTRY</th>
<th>1=REM, 2=FIS</th>
<th>RECORD</th>
<th>POINTER</th>
<th>PLANAR</th>
<th>WEIGHTING</th>
<th>EXPANSION</th>
<th>NO. 3F</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>1</td>
<td>16</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>1</td>
<td>19</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

### DIFINT FILE

**MESH RATIO = (LOWER DELTA Z)/(UPPER DELTA Z)**

**LOWER PLANE = 0 FOR LOWER BOUNDARY INTEGRAL**

**UPPER PLANE = 0 FOR UPPER BOUNDARY INTEGRAL**

<table>
<thead>
<tr>
<th>ENTRY</th>
<th>RECORD</th>
<th>POINTER</th>
<th>LOWER PLANAR</th>
<th>UPPER PLANAR</th>
<th>MESH</th>
<th>WEIGHTING</th>
<th>EXPANSION</th>
<th>NO. OF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1.00</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1.00</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>0</td>
<td>1.00</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

**TIME SPENT IN OVERLAY INTEG, CP = 0.4, PP = 0.0**

- SUBROUTINE IMPRO1, CP = 0.0, PP = 0.0
- SUBROUTINE REDTOC, CP = 0.0, PP = 0.0
- SUBROUTINE REWTIT, CP = 0.1, PP = 0.0
- SUBROUTINE INT1, CP = 0.0, PP = 0.0
- SUBROUTINE INT2, CP = 0.0, PP = 0.0
- SUBROUTINE INT3, CP = 0.0, PP = 0.0
- SUBROUTINE INT4, CP = 0.0, PP = 0.0
- SUBROUTINE RITTOC, CP = 0.0, PP = 0.0
- SUBROUTINE INTRIT, CP = 0.0, PP = 0.0
SYN3D, OVERLAY SOLVE
THREE GROUP, FUNDAMENTAL MODE CALCULATION

INTTOC FILE LABEL = 1/20/0443.6, VERSION NUMBER = 2

DATA MANAGEMENT PARAMETERS FOR SOLUTION
INTGLS FILE, WORDS/RECORD = 45
NUMBER OF CONCURRENT RECORDS DURING GAUSS ELIMINATION = 1
NUMBER OF RECORDS = 1

FILE, WORDS/RECORD = 18
NUMBER OF RECORDS = 1

DIRECT EIGENVALUE CALCULATION
EIGENVALUE ESTIMATE = 0.0
MAXIMUM NUMBER OF ITERATIONS = 10
CONVERGENCE CRITERION = 0.10000E-04

ITERATION EIGENVALUE
1  0.129062+01
2  0.130172+01
3  0.130172+01
4  0.130172+01

THE CALCULATION HAS CONVERGED ON AN EIGENVALUE
NORMALIZATION, PLIF * SIGMA-P * POWER/PISSON = 0.10000E+01 WATTS.
SYN3D, OVERLAY SOLVE  THREE GROUP, FUNDAMENTAL MODE CALCULATION

COMBINING COEFFICIENTS FOR DIRECT CALCULATION

FUNCTION NO. =  1  1  1
GROUP =  1  2  3

AXIAL AXIAL
POSITION GEOM.

<table>
<thead>
<tr>
<th>Position</th>
<th>Function</th>
<th>Axial 1</th>
<th>Axial 2</th>
<th>Axial 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.67</td>
<td>1</td>
<td>4.191E+07</td>
<td>5.868E+07</td>
<td>1.921E+06</td>
</tr>
<tr>
<td>5.00</td>
<td>1</td>
<td>4.191E+07</td>
<td>5.868E+07</td>
<td>1.921E+06</td>
</tr>
<tr>
<td>8.33</td>
<td>1</td>
<td>4.191E+07</td>
<td>5.868E+07</td>
<td>1.921E+06</td>
</tr>
</tbody>
</table>

TIME SPENT IN OVERLAY SOLVE, CP = 0.2, PP = 0.0
SUBROUTINE REDTOC, CP = 0.0, PP = 0.0
SUBROUTINE WINTGL, CP = 0.0, PP = 0.0
SUBROUTINE CALPOW, CP = 0.0, PP = 0.0
SUBROUTINE SPROB, CP = 0.1, PP = 0.0
SUBROUTINE OUTPRO, CP = 0.0, PP = 0.0
APPENDIX H. Sample Problem Output. Problem 2 - Complete Output

SYN3D, OVERLAY CARDS

2D BUCKLED PLANE (THE UNRODDED CORE PLANE FROM THE 3D MODEL)
HALF CORE AXIAL SYMMETRY

1/20/, 05.07., PAGE 1

LISTING OF SYN3D INPUT

<table>
<thead>
<tr>
<th>CARDS-FOR-CARD-TYPE DATA</th>
<th>DATA</th>
</tr>
</thead>
<tbody>
<tr>
<td>2D BUCKLED PLANE (THE UNRODDED CORE PLANE FROM THE 3D MODEL)</td>
<td>2 1 1 1 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
</tbody>
</table>
SYN3D, OVERLAY INTEGR
2D BUCKLED PLANE (THE UNRODDED CORE PLANE FROM THE 3D MODEL)
HALF CORE AXIAL SYMMETRY

GENERAL PROBLEM DATA
NUMBER OF GROUPS = 3 (NGROUP)
BPOINT CONTAINER SIZE = 6000
NUMBER OF AXIAL MESH INTERVALS = 30
THERE IS NO GROUP COLLAPSING
THERE IS NO BUBBLING
THERE IS NO INPUT FUNCTION SCALING
THERE WERE NO INPUT INTEGRALS

BOUNDARY CONDITIONS
LOWER X BOUNDARY CONDITION = ZERO CURRENT
UPPER X BOUNDARY CONDITION = ZERO FLUX
LOWER Y BOUNDARY CONDITION = ZERO CURRENT
UPPER Y BOUNDARY CONDITION = ZERO FLUX
LOWER Z BOUNDARY CONDITION = ZERO CURRENT
UPPER Z BOUNDARY CONDITION = ZERO FLUX
SYN3D. OVERLAY INTEGRATION
2D BUCKLED PLANE (THE UNGROUNDED CORE PLANE FROM THE 3D MODEL) 1/20/, 05.07., PAGE 3
HALF CORE AXIAL SYMMETRY

<table>
<thead>
<tr>
<th>FUNCTION FILE NUMBER</th>
<th>FILE NAME</th>
<th>GROUPS USED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RTPLIFX, UNMOD, CORE, 1</td>
<td>ALL 3</td>
</tr>
<tr>
<td>2</td>
<td>999</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AXIAL POSITION (CM)</th>
<th>GEOMETRY FILE NAME</th>
<th>EXPANSION</th>
<th>WEIGHTING</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0 TO 60.0</td>
<td>GECDST, CORE, UNMOD, 1</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AXIAL MESH INTERVALS</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.000D-00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00</td>
</tr>
<tr>
<td>2.000D-00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00 2.000D+00</td>
</tr>
</tbody>
</table>

DATA MANAGEMENT PARAMETERS FOR INTEGRAL CALCULATION
SPACE AVAILABLE DURING "LUX AND CROSS SECTION" REWRITE = 9858
REQFLUX FILE, WORDS/RECORD = 814
ROWS OF PLUXES/RECORD = 14
SPACE AVAILABLE DURING INTEGRATION = 2446
VOLINT AND DIFINT FILES, WORDS/RECORD = 2000

NUMBER OF CONCURRENT RECORDS DURING "LUX AND CROSS SECTION" REWRITE = 11
COLUMNS OF PLUXES/RECORD = 14
NUMBER OF CONCURRENT RECORDS DURING INTEGRATION = 127
COMPOSITIONS/RECORD FOR FISSION-REMOVAL RECORDS = 5
GROUPS/RECORD FOR DIFFUSION COEFFICIENT RECORDS = 3
NUMBER OF CONCURRENT RECORDS DURING INTEGRATION = 1
### FINAL INTEGRAL TOCS

#### VOLINT FILE

<table>
<thead>
<tr>
<th>ENTRY</th>
<th>1=REM, 2=FIS</th>
<th>RECORD</th>
<th>F'INTER</th>
<th>PLANAR</th>
<th>WEIGHTING</th>
<th>EXPANSION</th>
<th>NO. OF RECORDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3</td>
<td>1</td>
<td>16</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>4</td>
<td>1</td>
<td>19</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>1</td>
<td>22</td>
<td>1</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>1</td>
<td>25</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>2</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

#### DIFINT FILE

Mesh Ratio = (Lower Delta Z)/(Upper Delta Z)
Lower Plane = 0 for Lower Boundary Integral
Upper Plane = 0 for Upper Boundary Integral

<table>
<thead>
<tr>
<th>ENTRY</th>
<th>NO.</th>
<th>IN RECORD</th>
<th>GEOMETRY NO.</th>
<th>GEOMETRY NO.</th>
<th>MESH RATIO</th>
<th>WEIGHTING</th>
<th>EXPANSION</th>
<th>NO. OF RECORDS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1.00</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>4</td>
<td>1</td>
<td>1</td>
<td>1.00</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>7</td>
<td>1</td>
<td>1</td>
<td>1.00</td>
<td>2</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Time Spent in VOLINT INTEG, CP = 0.5, PP = 0.0

- SUBROUTINE IMPRO1, CP = 0.1, PP = 0.0
- SUBROUTINE REDTOC, CP = 0.0, PP = 0.0
- SUBROUTINE BEVTIT, CP = 0.1, PP = 0.0
- SUBROUTINE INT1, CP = 0.0, PP = 0.0
- SUBROUTINE INT2, CP = 0.0, PP = 0.0
- SUBROUTINE INT3, CP = 0.0, PP = 0.0
- SUBROUTINE INT4, CP = 0.0, PP = 0.0
- SUBROUTINE INTTIT, CP = 0.0, PP = 0.0
- SUBROUTINE INTTIT, CP = 0.0, PP = 0.0
SYN3D, OVERLAY SOLVE
2D BUCKLED PLANE (THE UNRODDED CORE PLANE FROM THE 3D MODEL)
HALF CORE AXIAL SYMMETRY

INTTOC FILE LABEL = 1/20/ 0507.8, VERSION NUMBER = 2

DATA MANAGEMENT PARAMETERS FOR SOLUTION
INTG3S FILE, WORDS/RECORD = 45 NUMBER OF CONCURRENT RECORDS DURING GAUSS ELIMINATION = 1
A FILE, WORDS/RECORD = 1 NUMBER OF RECORDS = 1
B FILE, WORDS/RECORD = 261 NUMBER OF RECORDS = 1

DIRECT EIGENVALUE CALCULATION
EIGENVALUE ESTIMATE = 0.10000E+01, MAXIMUM NUMBER OF ITERATIONS = 20, CONVERGENCE CRITERION = 0.10000E-04

ITERATION EIGENVALUE
1 0.99500E+00
2 0.99600E+00
3 0.99600E+00
4 0.99600E+00

THE CALCULATION HAS CONVERGED ON AN EIGENVALUE

NORMALIZATION, FLUX * SIGMA-F * POWER/FISSION = 0.10000E+02 WATTS.
### COMBINING COEFFICIENTS FOR DIRECT CALCULATION

<table>
<thead>
<tr>
<th>FUNCTION NO. =</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>2</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP = 1</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AXIAL POSITION</th>
<th>AXIAL GEOM.</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.00</td>
<td>1</td>
<td>2.015E+01 2.015E+01 2.015E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>3.00</td>
<td>1</td>
<td>2.010E+01 2.010E+01 2.010E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>5.00</td>
<td>1</td>
<td>1.998E+01 1.998E+01 1.998E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>7.00</td>
<td>1</td>
<td>1.982E+01 1.982E+01 1.982E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>9.00</td>
<td>1</td>
<td>1.960E+01 1.960E+01 1.960E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>11.00</td>
<td>1</td>
<td>1.933E+01 1.933E+01 1.933E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>13.00</td>
<td>1</td>
<td>1.900E+01 1.900E+01 1.900E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>15.00</td>
<td>1</td>
<td>1.862E+01 1.862E+01 1.862E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>17.00</td>
<td>1</td>
<td>1.819E+01 1.820E+01 1.820E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>19.00</td>
<td>1</td>
<td>1.772E+01 1.772E+01 1.772E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>21.00</td>
<td>1</td>
<td>1.719E+01 1.719E+01 1.719E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>23.00</td>
<td>1</td>
<td>1.661E+01 1.661E+01 1.661E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>25.00</td>
<td>1</td>
<td>1.599E+01 1.599E+01 1.599E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>27.00</td>
<td>1</td>
<td>1.533E+01 1.533E+01 1.533E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>29.00</td>
<td>1</td>
<td>1.462E+01 1.462E+01 1.462E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>31.00</td>
<td>1</td>
<td>1.388E+01 1.388E+01 1.388E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>33.00</td>
<td>1</td>
<td>1.309E+01 1.309E+01 1.309E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>35.00</td>
<td>1</td>
<td>1.227E+01 1.227E+01 1.227E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>37.00</td>
<td>1</td>
<td>1.142E+01 1.142E+01 1.142E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>39.00</td>
<td>1</td>
<td>1.053E+01 1.053E+01 1.053E+01 0.0 0.0 0.0</td>
</tr>
<tr>
<td>41.00</td>
<td>1</td>
<td>9.619E+00 9.619E+00 9.619E+00 0.0 0.0 0.0</td>
</tr>
<tr>
<td>43.00</td>
<td>1</td>
<td>8.679E+00 8.679E+00 8.679E+00 0.0 0.0 0.0</td>
</tr>
<tr>
<td>45.00</td>
<td>1</td>
<td>7.714E+00 7.715E+00 7.715E+00 0.0 0.0 0.0</td>
</tr>
<tr>
<td>47.00</td>
<td>1</td>
<td>6.729E+00 6.729E+00 6.729E+00 0.0 0.0 0.0</td>
</tr>
<tr>
<td>49.00</td>
<td>1</td>
<td>5.725E+00 5.726E+00 5.726E+00 0.0 0.0 0.0</td>
</tr>
<tr>
<td>51.00</td>
<td>1</td>
<td>4.706E+00 4.706E+00 4.706E+00 0.0 0.0 0.0</td>
</tr>
<tr>
<td>53.00</td>
<td>1</td>
<td>3.674E+00 3.674E+00 3.674E+00 0.0 0.0 0.0</td>
</tr>
<tr>
<td>55.00</td>
<td>1</td>
<td>2.631E+00 2.631E+00 2.631E+00 0.0 0.0 0.0</td>
</tr>
<tr>
<td>57.00</td>
<td>1</td>
<td>1.582E+00 1.582E+00 1.582E+00 0.0 0.0 0.0</td>
</tr>
<tr>
<td>59.00</td>
<td>1</td>
<td>5.277E-01 5.277E-01 5.277E-01 0.0 0.0 0.0</td>
</tr>
</tbody>
</table>

**TIME SPENT IN OVERLAY SOLVE**

| SUBROUTINE REDCO, CP = 0.0, PP = 0.0 |
|-------------------------------|--------------|
| SUBROUTINE WINTGL, CP = 0.0, PP = 0.0 |
| SUBROUTINE CALPOW, CP = 0.0, PP = 0.0 |
| SUBROUTINE SPROE, CP = 0.2, PP = 0.0 |
| SUBROUTINE OUTFRO, CP = 0.0, PP = 0.0 |
APPENDIX H. Sample Problem Output. Problem 3 - Partial Output

SYN3D, OVERLAY CARDS
3 GROUP, 3-DIMENSIONAL MODEL, FLUX WEIGHTED SYNTHESIS
BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
3D GEOMETRY BUILT UP FROM 2D GEODST FILES

LISTING OF SYN3D INPUT

CARDS-PER-CARD-TYPE DATA
3 1 1 6 0 4 0 0 0 0 0 0 0 0 0 1 1 0 1 0 1

<table>
<thead>
<tr>
<th>CARD TYPE</th>
<th>3 GROUP, 3-DIMENSIONAL MODEL, FLUX WEIGHTED SYNTHESIS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE</td>
</tr>
<tr>
<td>1</td>
<td>3D GEOMETRY BUILT UP FROM 2D GEODST FILES</td>
</tr>
<tr>
<td>2</td>
<td>1 1 20 0 0 0 0 0 1 1 0 0</td>
</tr>
<tr>
<td>3</td>
<td>0.16000E+01 0.10000E+02</td>
</tr>
<tr>
<td>4</td>
<td>0.20000E+03 0.0 0.0 0.0</td>
</tr>
<tr>
<td>5</td>
<td>1 0.0 0.20000E+02 0.0 0.0</td>
</tr>
<tr>
<td>5</td>
<td>2 0.20000E+02 0.60000E+02 0.0 0.0</td>
</tr>
<tr>
<td>5</td>
<td>3 0.60000E+02 0.10000E+03 0.0 0.0</td>
</tr>
<tr>
<td>5</td>
<td>4 0.10000E+03 0.18000E+03 0.0 0.0</td>
</tr>
<tr>
<td>5</td>
<td>5 0.14000E+03 0.18000E+03 0.0 0.0</td>
</tr>
<tr>
<td>5</td>
<td>6 0.18000E+03 0.20000E+03 0.0 0.0</td>
</tr>
<tr>
<td>7</td>
<td>RTFLUX 1 0.0 0.20000E+03 0.0</td>
</tr>
<tr>
<td>7</td>
<td>RTFLUX 2 0.0 0.20000E+03 0.0</td>
</tr>
<tr>
<td>7</td>
<td>RTFLUX 3 0.0 0.80000E+02 0.0</td>
</tr>
<tr>
<td>7</td>
<td>RTFLUX 4 0.12000E+03 0.20000E+03 0.0</td>
</tr>
<tr>
<td>16</td>
<td>0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>17</td>
<td>1 2 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>18</td>
<td>1 10 2 10 3 10 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>21</td>
<td>10 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
</tbody>
</table>
SYJ3D, OVERLAY INTG 3 GROUP, 3-DIMENSIONAL MODEL, FLUX WEIGHTED SYNTHESIS
BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
3D GEOMETRY BUILT UP FROM 2D GEOST FILES

GENERAL PROBLEM DATA
NUMBER OF GROUPS = 3 (NGROUP)
BPOINTER CONTAINER SIZE = 10000
NUMBER OF AXIAL MESH INTERVALS = 20
THERE IS NO GROUP COLLAPSING
THERE IS NO INPUT BUFFERING
THERE IS NO INPUT FUNCTION SCALING
THERE WERE NO INPUT INTEGRALS

BOUNDARY CONDITIONS
LOWER X BOUNDARY CONDITION = ZERO CURRENT
UPPER X BOUNDARY CONDITION = ZERO FLUX
LOWER Y BOUNDARY CONDITION = ZERO CURRENT
UPPER Y BOUNDARY CONDITION = ZERO FLUX
LOWER Z BOUNDARY CONDITION = ZERO FLUX
UPPER Z BOUNDARY CONDITION = ZERO FLUX
SYN3D, OVERLAY INTEG

3 GROUP, 3-DIMENSIONAL MODEL, FLUX WEIGHTED SYNTHESIS

BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE

3D GEOMETRY BUILT UP FROM 2D GEODST FILES

---

<table>
<thead>
<tr>
<th>FUNCTION NUMBER</th>
<th>FILE NAME</th>
<th>GROUPS USED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RTFLUI, ONROD, CORE</td>
<td>ALL 3</td>
</tr>
<tr>
<td>2</td>
<td>RTFLUI, PID, CORE</td>
<td>ALL 3</td>
</tr>
<tr>
<td>3</td>
<td>RTFLUI, UNROD, BLANKT</td>
<td>ALL 3</td>
</tr>
<tr>
<td>4</td>
<td>RTFLUI, PID, BLANKT</td>
<td>ALL 3</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>AXIAL POSITION (CM)</th>
<th>PLANAR GEOMETRY NUMBER</th>
<th>EXPANSION FUNCTION NUMBERS</th>
<th>WEIGHTING FUNCTION NUMBERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0 TO 20.0</td>
<td>1</td>
<td>1 2 3</td>
<td>1 2 3</td>
</tr>
<tr>
<td>20.0 TO 60.0</td>
<td>2</td>
<td>1 2 3</td>
<td>1 2 3</td>
</tr>
<tr>
<td>60.0 TO 100.0</td>
<td>3</td>
<td>1 2 3</td>
<td>1 2 3</td>
</tr>
<tr>
<td>100.0 TO 120.0</td>
<td>4</td>
<td>1 2 3</td>
<td>1 2 4</td>
</tr>
<tr>
<td>120.0 TO 140.0</td>
<td>4</td>
<td>1 2 4</td>
<td>1 2 4</td>
</tr>
<tr>
<td>140.0 TO 200.0</td>
<td>6</td>
<td>1 2 4</td>
<td>1 2 4</td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>AXIAL MESH INTERVALS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01</td>
</tr>
<tr>
<td>1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01</td>
</tr>
<tr>
<td>1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01 1.000D+01</td>
</tr>
</tbody>
</table>

---

DATA MANAGEMENT PARAMETERS FOR INTEGRAL CALCULATION

SPACE AVAILABLE DURING INTEGRATION = 2784
VOLUME FILE, WORDS/RECORD = 2100
VOLUME AND DIFINT FILES, WORDS/RECORD = 2100
NUMBER OF CONCURRENT RECORDS DURING INTEGRATION = 1

SPACE AVAILABLE DURING REWRITE = 2793
NUMBER OF CONCURRENT RECORDS DURING REWRITE = 150
NUMBER OF CONCURRENT RECORDS DURING REWRITE = 5
COLUMNS OF FLUXES/RECORD = 14
GROUPS/RECORD FOR DIFUSION COEFFICIENT RECORDS = 3
NUMBER OF CONCURRENT RECORDS DURING REWRITE = 105
NUMBER OF CONCURRENT RECORDS DURING REWRITE = 7
NUMBER OF CONCURRENT RECORDS DURING REWRITE = 16132
NUMBER OF CONCURRENT RECORDS DURING REWRITE = 6
NUMBER OF CONCURRENT RECORDS DURING REWRITE = 1

---

FILE NAME | NUMBER
-----------|--------
UNROD, CORE |
ROD, CORE |
UNROD, BLANKT |
ROD, BLANKT |
UNROD, CORE |
ROD, CORE |
UNROD, PLAN |
ROD, PLAN |
UNROD, MORE |
ROD, MORE |
UNROD, CORE |
ROD, CORE |
UNROD, PLAN |
ROD, PLAN |
UNROD, CORE |
ROD, CORE |
UNROD, PLAN |
ROD, PLAN |
SYN3D, OVERLAY SOLVE

3 GROUP, 3-DIMENSIONAL MODEL, FLUX WEIGHTED SYNTHESIS
BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
3D GEOMETRY BUILT UP FROM 2D GEOBST FILES

INTFOC FILE LABEL = 1/20/ 0513.0, VERSION NUMBER = 2

DATA MANAGEMENT PARAMETERS FOR SOLUTION INTGSS FILE, WORDS/RECORD = 2655
NUMBER OF RECORDS DURING GAUSS ELIMINATION = 1
H FILE, WORDS/RECORD = 1350
NUMBER OF RECORDS = 1

DIRECT EIGENVALUE CALCULATION
EIGENVALUE ESTIMATE = 0.10000E+01, MAXIMUM NUMBER OF ITERATIONS = 20, CONVERGENCE CRITERION = 0.10000E-04

ITERATION EIGENVALUE
1 0.96700E+00
2 0.97495E+00
3 0.97505E+00
4 0.97506E+00
5 0.97506E+00

THE CALCULATION HAS CONVERGED ON AN EIGENVALUE
NORMALIZATION, FLUX * SIGMA-F * POWER/FISSION = 0.10000E+02 WATTS.
**Combining Coefficients for Direct Calculation**

<table>
<thead>
<tr>
<th>AXIAL POSITION</th>
<th>AXIAL COEFFICIENTS</th>
<th>GROUP 1</th>
<th>GROUP 2</th>
<th>GROUP 3</th>
<th>GROUP 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>15.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>25.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>35.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>45.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>55.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>65.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>75.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>85.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>95.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>105.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>115.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>125.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>135.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>145.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>155.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>165.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>175.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>185.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
<tr>
<td>195.00</td>
<td></td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
<td>1.000</td>
</tr>
</tbody>
</table>
**SYN3D, OVERLAY SOLVE**

3 GROUP, 3-DIMENSIONAL MODEL, FLUX WEIGHTED SYNTHESIS
BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
3D GEOMETRY BUILT UP FROM 2D GEODST FILES

COMBINING COEFFICIENTS FOR DIRECT CALCULATION

<table>
<thead>
<tr>
<th>FUNCTION NO.</th>
<th>4</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AXIAL POSITION</th>
<th>AXIAL GEOM.</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>3</th>
<th>3</th>
<th>3</th>
<th>3</th>
<th>4</th>
<th>4</th>
<th>0.0</th>
<th>0.0</th>
<th>0.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.00</td>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.428E-04</td>
<td>1.425E-04</td>
<td></td>
</tr>
<tr>
<td>15.00</td>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.122E-03</td>
<td>1.122E-03</td>
<td></td>
</tr>
<tr>
<td>25.00</td>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>4.308E-03</td>
<td>5.536E-03</td>
<td></td>
</tr>
<tr>
<td>35.00</td>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>5.536E-03</td>
<td>8.542E-03</td>
<td></td>
</tr>
<tr>
<td>45.00</td>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>4.736E-03</td>
<td>5.697E-03</td>
<td></td>
</tr>
<tr>
<td>55.00</td>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>5.376E-03</td>
<td>3.735E-03</td>
<td></td>
</tr>
<tr>
<td>65.00</td>
<td>3</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>2.035E-03</td>
<td>2.896E-03</td>
<td></td>
</tr>
<tr>
<td>75.00</td>
<td>3</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>6.639E-04</td>
<td>9.832E-04</td>
<td></td>
</tr>
</tbody>
</table>

**TIME SPENT IN OVERLAY SOLVE**

<table>
<thead>
<tr>
<th>CP</th>
<th>1.2</th>
<th>PP</th>
<th>0.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUBROUTINE REDTOC, CP</td>
<td>0.0</td>
<td>PP</td>
<td>0.0</td>
</tr>
<tr>
<td>SUBROUTINE WENGL, CP</td>
<td>0.1</td>
<td>PP</td>
<td>0.0</td>
</tr>
<tr>
<td>SUBROUTINE CALPOW, CP</td>
<td>0.0</td>
<td>PP</td>
<td>0.0</td>
</tr>
<tr>
<td>SUBROUTINE SPROE, CP</td>
<td>0.0</td>
<td>PP</td>
<td>0.0</td>
</tr>
<tr>
<td>SUBROUTINE OUTPRO, CP</td>
<td>0.1</td>
<td>PP</td>
<td>0.0</td>
</tr>
</tbody>
</table>
### PLANAR FLUX DISTRIBUTION

**Z = 95.00, GROUP = 1**

<table>
<thead>
<tr>
<th>POW</th>
<th>COLUMN 1</th>
<th>COLUMN 2</th>
<th>COLUMN 3</th>
<th>COLUMN 4</th>
<th>COLUMN 5</th>
<th>COLUMN 6</th>
<th>COLUMN 7</th>
<th>COLUMN 8</th>
<th>COLUMN 9</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.9046D+08</td>
<td>2.0897D+08</td>
<td>2.0565D+08</td>
<td>1.9274D+08</td>
<td>1.7381D+08</td>
<td>1.5046D+08</td>
<td>9.8743D+07</td>
<td>4.8302D+07</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2.0897D+08</td>
<td>2.1277D+08</td>
<td>2.1206D+08</td>
<td>2.0468D+08</td>
<td>1.9112D+08</td>
<td>1.7346D+08</td>
<td>1.5419D+08</td>
<td>1.1214D+08</td>
<td>4.8659D+07</td>
</tr>
<tr>
<td>3</td>
<td>2.1233D+08</td>
<td>2.1206D+08</td>
<td>2.0892D+08</td>
<td>1.9906D+08</td>
<td>1.8543D+08</td>
<td>1.6811D+08</td>
<td>1.5272D+08</td>
<td>1.1512D+08</td>
<td>4.8747D+07</td>
</tr>
<tr>
<td>4</td>
<td>2.0565D+08</td>
<td>2.0488D+08</td>
<td>1.9906D+08</td>
<td>1.8904D+08</td>
<td>1.7473D+08</td>
<td>1.5914D+08</td>
<td>1.3953D+08</td>
<td>1.2942D+08</td>
<td>4.6609D+07</td>
</tr>
<tr>
<td>5</td>
<td>1.9234D+08</td>
<td>1.9112D+08</td>
<td>1.8543D+08</td>
<td>1.7473D+08</td>
<td>1.5914D+08</td>
<td>1.3953D+08</td>
<td>1.2942D+08</td>
<td>1.2942D+08</td>
<td>4.2206D+07</td>
</tr>
<tr>
<td>6</td>
<td>1.7381D+08</td>
<td>1.7346D+08</td>
<td>1.6811D+08</td>
<td>1.5843D+08</td>
<td>1.5843D+08</td>
<td>1.5814D+08</td>
<td>1.5814D+08</td>
<td>1.5814D+08</td>
<td>4.2206D+07</td>
</tr>
<tr>
<td>7</td>
<td>1.5046D+08</td>
<td>1.5419D+08</td>
<td>1.4432D+08</td>
<td>1.2964D+08</td>
<td>1.2964D+08</td>
<td>1.2964D+08</td>
<td>1.2964D+08</td>
<td>1.2964D+08</td>
<td>4.2206D+07</td>
</tr>
<tr>
<td>8</td>
<td>9.8743D+07</td>
<td>1.1214D+08</td>
<td>1.1512D+08</td>
<td>1.1067D+08</td>
<td>9.3527D+07</td>
<td>8.4747D+07</td>
<td>6.5384D+07</td>
<td>4.0483D+07</td>
<td>1.9421D+07</td>
</tr>
<tr>
<td>10</td>
<td>1.8800D+07</td>
<td>1.8798D+07</td>
<td>1.8798D+07</td>
<td>1.8798D+07</td>
<td>1.8798D+07</td>
<td>1.8798D+07</td>
<td>1.8798D+07</td>
<td>1.8798D+07</td>
<td>1.8798D+07</td>
</tr>
<tr>
<td>11</td>
<td>8.4087D+06</td>
<td>8.3831D+06</td>
<td>8.3831D+06</td>
<td>8.2339D+06</td>
<td>8.2339D+06</td>
<td>8.2339D+06</td>
<td>8.2339D+06</td>
<td>8.2339D+06</td>
<td>8.2339D+06</td>
</tr>
<tr>
<td>12</td>
<td>2.4764D+06</td>
<td>2.4656D+06</td>
<td>2.4656D+06</td>
<td>2.4115D+06</td>
<td>2.4115D+06</td>
<td>2.4115D+06</td>
<td>2.4115D+06</td>
<td>2.4115D+06</td>
<td>2.4115D+06</td>
</tr>
</tbody>
</table>

**ROW COLUMN COLUMN COLUMN COLUMN COLUMN COLUMN COLUMN COLUMN COLUMN**

<table>
<thead>
<tr>
<th>10</th>
<th>11</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.8800D+07</td>
<td>8.4087D+06</td>
</tr>
<tr>
<td>2</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>3</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>4</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>5</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>6</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>7</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>8</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>9</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>10</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>11</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>12</td>
<td>1.8798D+07</td>
<td>8.3831D+06</td>
</tr>
<tr>
<td>ROW</td>
<td>COLUMN 1</td>
<td>COLUMN 2</td>
</tr>
<tr>
<td>-----</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td>1</td>
<td>0.0</td>
<td>3.4749E-05</td>
</tr>
<tr>
<td>4</td>
<td>3.3536E-05</td>
<td>3.3312E-05</td>
</tr>
<tr>
<td>5</td>
<td>3.1213E-05</td>
<td>3.1004E-05</td>
</tr>
<tr>
<td>6</td>
<td>2.7897E-05</td>
<td>2.7796E-05</td>
</tr>
<tr>
<td>7</td>
<td>3.0186E-05</td>
<td>3.0652E-05</td>
</tr>
<tr>
<td>8</td>
<td>0.0</td>
<td>2.3299E-05</td>
</tr>
<tr>
<td>9</td>
<td>2.4301E-06</td>
<td>2.4300E-06</td>
</tr>
<tr>
<td>11</td>
<td>0.0</td>
<td>0.0</td>
</tr>
<tr>
<td>12</td>
<td>0.0</td>
<td>0.0</td>
</tr>
</tbody>
</table>

Planar power density (Watts/CC), Z = 95.00
SY3D, OVERLAY EDITS  3 GROUP, 3-DIMENSIONAL MODEL, FLUX WEIGHTED SYNTHESIS
BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
3D GEOMETRY BUILT UP FROM 2D GEOBST FILES

<table>
<thead>
<tr>
<th>ZONE</th>
<th>GROUP</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>1.5162E+08</td>
<td>1.1302E+09</td>
<td>5.7973E+06</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>9.3248E+07</td>
<td>1.0261E+08</td>
<td>3.0418E+06</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>1.4131E+07</td>
<td>3.4796E+07</td>
<td>4.0616E+07</td>
</tr>
<tr>
<td>4</td>
<td></td>
<td>1.6114E+07</td>
<td>5.349E+06</td>
<td>5.349E+06</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>1.8132E+06</td>
<td>3.7031E+06</td>
<td>3.7031E+06</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>2.4008E+07</td>
<td>5.9773E+07</td>
<td>2.9702E+06</td>
</tr>
<tr>
<td>7</td>
<td></td>
<td>1.9452E+07</td>
<td>5.9773E+07</td>
<td>2.9702E+06</td>
</tr>
</tbody>
</table>
CONTRIBUTIONS TO AXIAL REAL FLUX FROM EACH FUNCTION

GROUP 2

MAX = 1.175E+12
MIN = -6.523E+10

MIN = 5.000E+00
MAX = 1.950E+02

0 = SDM
1,2...ETC = INDIVIDUAL MODES
APPENDIX H. Sample Problem Output. Problem 4 - Partial Output

SYN3D, OVERLAY CARDS
3 GROUP, 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJACENT WEIGHTING
BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
GROUP 3 OF BLANKET FUNCTIONS NOT USED AT ALL
INPUT 3D GEODST FILE

LISTING OF SYN3D INPUT

CARDS-FER-CARD-TYPE DATA
4 1 1 0 0 2 4 4 0 0 0 0 2 0 0 1 1 1 1

CARD TYPE
1 3 GROUP, 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJACENT WEIGHTING
1 BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
1 GROUP 3 OF BLANKET FUNCTIONS NOT USED AT ALL
1 INPUT 3D GEODST FILE
2 15 2 20 0 0 0 0 3 3 0 1
3 0.10000E+01 0.0 0.10000E+02 0.0
4 0.21300E+01 0.0 0.21300E+01 0.0
5 0.21300E+01 0.0 0.21300E+01 0.0
6 0.10000E+03 0.0 0.10000E+03 0.0
7 0.20000E+03 0.0 0.20000E+03 0.0
8 0.20000E+03 0.0 0.20000E+03 0.0
9 0.20000E+03 0.0 0.20000E+03 0.0
10 0.20000E+03 0.0 0.20000E+03 0.0

THE DATA FOR THE TYPE 4 AND 5 CARDS IS BEING EXTRACTED FROM THE FILE GEODST, 3D, 3GP, MODEL, 1
4 20 0.20000E+03 0.0 0.0
5 2 0.0 0.20000E+02
6 3 0.20000E+02 0.60000E+02
7 4 0.60000E+02 0.10000E+03
8 5 0.10000E+03 0.14000E+03
9 6 0.14000E+03 0.18000E+03
10 7 0.18000E+03 0.20000E+03
GENERAL PROBLEM DATA
NUMBER OF GROUPS = 3 (NGROUP)
BPOINTER CONTAINER SIZE = 15000
NUMBER OF AXIAL MESH INTERVALS = 20
THERE IS NO GROUP COLLAPSING
THERE IS NO BUFFERING
THERE IS NO INPUT FUNCTION SCALING
THERE WERE NO INPUT INTEGRALS

BOUNDARY CONDITIONS
LOWER X BOUNDARY CONDITION = ZERO CURRENT
UPPER X BOUNDARY CONDITION = ZERO FLUX
LOWER Y BOUNDARY CONDITION = ZERO CURRENT
UPPER Y BOUNDARY CONDITION = ZERO FLUX
LOWER Z BOUNDARY CONDITION = LOGARITHMIC
  C(1), BY GROUP
    2.130E+00 2.130E+00 2.130E+00
  C(2), BY GROUP
    1.000E+00 1.000E+00 1.000E+00
UPPER Z BOUNDARY CONDITION = LOGARITHMIC
  C(1), BY GROUP
    2.130E+00 2.130E+00 2.130E+00
  C(2), BY GROUP
    1.000E+00 1.000E+00 1.000E+00
### SYN3D, OVERLAY INTEGRATION

**3 GROUP, 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJOINT WEIGHTING**

Blanket functions (3 and 4) not used everywhere.

Group 3 of blanket functions not used at all.

**INPUT 3D GEOST FILE**

<table>
<thead>
<tr>
<th>FUNCTION FILE NUMBER</th>
<th>FILE NAME</th>
<th>GROUPS USED</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>RTFLUX, UNROD, CORE,</td>
<td>1 ALL 3</td>
</tr>
<tr>
<td>2</td>
<td>RTFLUX, ROD, CORE,</td>
<td>2 ALL 3</td>
</tr>
<tr>
<td>3</td>
<td>RTFLUX, UNROD, BLANKT,</td>
<td>3 1</td>
</tr>
<tr>
<td>4</td>
<td>RTFLUX, ROD, BLANKT,</td>
<td>4 1</td>
</tr>
<tr>
<td>5</td>
<td>ATFLUX, UNROD, CORE,</td>
<td>1 ALL 3</td>
</tr>
<tr>
<td>6</td>
<td>ATFLUX, ROD, CORE,</td>
<td>2 ALL 3</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>AXIAL POSITION (CM)</th>
<th>PLANAR GEOMETRY NUMBER</th>
<th>GEOMETRY FILE NUMBER</th>
<th>GEOMETRY FILE NAME</th>
<th>EXPANSION FUNCTION NUMBERS</th>
<th>WEIGHTING FUNCTION NUMBERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0 TO 20.0</td>
<td>1</td>
<td>12</td>
<td>GEOGST, 1/20/, 0521.2,</td>
<td>1 2 3</td>
<td>3 5 6</td>
</tr>
<tr>
<td>20.0 TO 60.0</td>
<td>2</td>
<td>13</td>
<td>GEOGST, 1/20/, 0521.2,</td>
<td>1 2 3</td>
<td>3 5 6</td>
</tr>
<tr>
<td>60.0 TO 100.0</td>
<td>3</td>
<td>14</td>
<td>GEOGST, 1/20/, 0521.2,</td>
<td>1 2 3</td>
<td>3 5 6</td>
</tr>
<tr>
<td>100.0 TO 150.0</td>
<td>4</td>
<td>15</td>
<td>GEOGST, 1/20/, 0521.2,</td>
<td>1 2 4</td>
<td>5 6 4</td>
</tr>
<tr>
<td>150.0 TO 180.0</td>
<td>5</td>
<td>16</td>
<td>GEOGST, 1/20/, 0521.2,</td>
<td>1 2 4</td>
<td>5 6 4</td>
</tr>
<tr>
<td>180.0 TO 200.0</td>
<td>6</td>
<td>17</td>
<td>GEOGST, 1/20/, 0521.2,</td>
<td>1 2 4</td>
<td>5 6 4</td>
</tr>
</tbody>
</table>

**AXIAL MESH INTERVALS**

<table>
<thead>
<tr>
<th>(CM)</th>
<th>GEOMETRY NUMBER</th>
<th>FILE NUMBER</th>
<th>GEOMETRY FILE NAME</th>
<th>EXPANSION</th>
<th>WEIGHTING</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.000D+01</td>
<td>1.000D+01</td>
<td>1.000D+01</td>
<td>1.000D+01</td>
<td>1.000D+01</td>
<td>1.000D+01</td>
</tr>
<tr>
<td>1.000D+01</td>
<td>1.000D+01</td>
<td>1.000D+01</td>
<td>1.000D+01</td>
<td>1.000D+01</td>
<td>1.000D+01</td>
</tr>
</tbody>
</table>

**DATA MANAGEMENT PARAMETERS FOR INTEGRAL CALCULATION**

Space available during flux and cross section rewrite = 26970

REQLFLX file, words/record = 4734

Columns of fluxes/record = 14

INITIAL ESTIMATES OF STORAGE REQUIREMENTS FOR FLUX, GEOMETRY AND/OR CROSS SECTION REWRITE WAS WRONG. SYN3D WILL START THE REWRITE AGAIN WITH BETTER ESTIMATES.

**DATA MANAGEMENT PARAMETERS FOR INTEGRAL CALCULATION**

Space available during flux and cross section rewrite = 25846

REQLFLX file, words/record = 4734

Columns of fluxes/record = 14

Reql FLUX file, words/record = 105

Columns of fluxes/record for pission-removal records = 7

Space available during integration = 5541

Volint and diffint files, words/record = 2000

NUMBER OF CONCURRENT RECORDS DURING INTEGRATION = 2

NUMBER OF CONCURRENT RECORDS DURING REWRITE = 5

NUMBER OF CONCURRENT RECORDS DURING REWRITE = 5

NUMBER OF CONCURRENT RECORDS DURING REWRITE = 240

GROUPS/RECORD FOR DIFFUSION COEFFICIENT RECORDS = 3
SYN3D, OVERLAY SOLVE  3 GROUP, 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJOINT WEIGHTING

BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
GROUP 3 OF BLANKET FUNCTIONS NOT USED AT ALL
INPUT 3D GEODST FILE

INTTOC FILE LABEL = 1/20/ 0521.7, VERSION NUMBER = 2

DATA MANAGEMENT PARAMETERS FOR SOLUTION
INTGLS FILE, WORDS/RECORD = 2214  NUMBER OF CONCURRENT RECORDS DURING GAUSS ELIMINATION = 1
INT FILE, WORDS/RECORD = 1216  NUMBER OF RECORDS = 1

DIRECT EIGENVALUE CALCULATION
EIGENVALUE ESTIMATE = 0.10000E+01, MAXIMUM NUMBER OF ITERATIONS = 20, CONVERGENCE CRITERION = 0.10000E-04

ITERATION  EIGENVALUE
   1  0.97289E+00
   2  0.97510E+00
   3  0.97520E+00
   4  0.97520E+00
   5  0.97520E+00

THE CALCULATION HAS CONVERGED ON AN EIGENVALUE

NORMALIZATION, FLUX * SIGMA-F * POWER/FISSION = 0.10000E+02 WATTS.
**SYN3D, OVERLAY SOLVE**  
3 GROUP, 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJOINT WEIGHTING

**GROUP 1 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJOINT WEIGHTING**

**OVERLAY SOLVE**

**GROUP 3 OF BLANKET FUNCTIONS NOT USED EVERYWHERE**

**GROUP 3 OF BLANKET FUNCTIONS NOT USED AT ALL**

**INPUT 3D GEODST FILE**

**COMBINING COEFFICIENTS FOR DIRECT CALCULATION**

<table>
<thead>
<tr>
<th>FUNCTION NO.</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>3</th>
<th>3</th>
<th>4</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP 1 2 3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

**AXIAL AXIAL POSITION GFOM.**

| 5.00 | -2.205E-03 | 3.212E-01 | 1.944E+00 | -4.265E-02 | -3.450E-02 | -7.119E-01 | 6.798E-04 | 6.021E-04 | 0.0 | 0.0 |
| 15.00 | -3.573E-02 | -1.853E-01 | 3.242E+00 | -1.966E-02 | 6.313E-02 | 1.706E-00 | 1.402E-03 | 1.958E-03 | 0.0 | 0.0 |
| 35.00 | -8.764E-02 | -1.261E-01 | 4.739E+00 | -1.785E-02 | 4.536E-02 | -1.025E-01 | 2.806E-03 | 3.036E-03 | 0.0 | 0.0 |
| 45.00 | -6.116E-02 | 1.676E+00 | 7.937E+00 | 2.480E-01 | 7.690E-02 | 1.201E+00 | 8.230E-03 | 4.059E-03 | 0.0 | 0.0 |
| 55.00 | 1.097E+00 | 4.715E+00 | 9.134E+00 | 5.667E-01 | -6.315E-02 | 2.323E+00 | 1.006E-04 | 2.297E-03 | 0.0 | 0.0 |
| 65.00 | 6.707E+00 | 8.895E+00 | 9.969E+00 | 2.086E-01 | -7.160E-01 | -1.133E+00 | 6.387E-04 | -1.974E-04 | 0.0 | 0.0 |
| 75.00 | 8.051E+00 | 9.826E+00 | 1.045E+01 | 6.109E-01 | -1.116E-01 | -6.752E-01 | 4.906E-05 | -2.784E-04 | 0.0 | 0.0 |
| 85.00 | 9.306E+00 | 1.023E+01 | 1.072E+01 | 1.599E+00 | 9.198E-01 | 2.259E-01 | 1.246E-03 | -1.005E-04 | 0.0 | 0.0 |
| 95.00 | 7.632E+00 | 9.666E+00 | 1.014E+01 | 3.828E+00 | 2.266E+00 | 1.449E+00 | 3.626E-03 | -2.628E-04 | 0.0 | 0.0 |
| 105.00 | 2.756E+00 | 3.285E+00 | 8.460E+00 | 1.004E+01 | 9.016E+00 | 1.248E+00 | 0.0 | 1.829E-03 | 1.101E-03 | 0.0 |
| 115.00 | 7.082E-01 | 9.588E+00 | 1.094E-01 | 1.156E+01 | 1.086E+01 | 1.226E+01 | 0.0 | 0.0 | 3.830E-04 | 7.380E-04 |
| 125.00 | 1.316E-01 | 1.289E-01 | 3.661E-02 | 1.028E+01 | 1.058E+01 | 1.085E+01 | 0.0 | 0.0 | -2.793E-05 | 2.825E-05 |
| 135.00 | -8.476E-02 | -2.275E-01 | -2.615E-01 | 7.538E+00 | 8.768E+00 | 9.089E+00 | 0.0 | 0.0 | 9.650E-04 | 1.371E-04 |
| 145.00 | -5.773E-02 | -9.156E-01 | -1.117E+00 | 2.106E+00 | 6.125E+00 | 1.251E+01 | 0.0 | 0.0 | 9.611E-03 | 2.302E-03 |
| 155.00 | -1.486E-02 | -4.825E-01 | -1.038E+00 | 3.104E+00 | 2.416E+00 | 9.916E+00 | 0.0 | 0.0 | 8.501E-03 | 4.562E-03 |
| 165.00 | -4.595E-03 | -1.757E-01 | -7.635E-01 | -1.002E+00 | 5.644E-01 | 6.900E-00 | 0.0 | 0.0 | 5.498E-03 | 4.436E-03 |
| 175.00 | -3.712E-03 | -5.592E-02 | -5.467E-01 | -2.615E-01 | 1.347E-01 | -7.338E-02 | 4.673E+00 | 0.0 | 0.0 | 3.181E-03 | 3.410E-03 |
| 185.00 | -9.076E-03 | -6.586E-02 | -5.061E-01 | -8.511E-02 | -1.071E-01 | 3.541E+00 | 1.756E-03 | 2.193E-03 | 0.0 | 0.0 |
| 195.00 | -5.959E-03 | -2.290E-02 | -2.005E-01 | -3.724E-02 | -6.697E-02 | 1.387E+00 | 0.0 | 0.0 | 7.301E-04 | 8.869E-04 |
SYN3D, OVERLAY SOLVE 3 GROUP, 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJOINT WEIGHTING

BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE
GROUP 3 OF BLANKET FUNCTIONS NOT USED AT ALL
INPUT 3D GEODES FILE

COMBINING COEFFICIENTS FOR DIRECT CALCULATION

<table>
<thead>
<tr>
<th>AXIAL</th>
<th>AXIAL</th>
</tr>
</thead>
<tbody>
<tr>
<td>POSITION GEOM.</td>
<td>FUNCTION NO.</td>
</tr>
<tr>
<td>5.00</td>
<td>1</td>
</tr>
<tr>
<td>15.00</td>
<td>1</td>
</tr>
<tr>
<td>25.00</td>
<td>2</td>
</tr>
<tr>
<td>35.00</td>
<td>2</td>
</tr>
<tr>
<td>45.00</td>
<td>2</td>
</tr>
<tr>
<td>55.00</td>
<td>3</td>
</tr>
<tr>
<td>65.00</td>
<td>3</td>
</tr>
<tr>
<td>75.00</td>
<td>3</td>
</tr>
<tr>
<td>85.00</td>
<td>3</td>
</tr>
<tr>
<td>95.00</td>
<td>3</td>
</tr>
<tr>
<td>105.00</td>
<td>4</td>
</tr>
<tr>
<td>115.00</td>
<td>4</td>
</tr>
<tr>
<td>125.00</td>
<td>4</td>
</tr>
<tr>
<td>135.00</td>
<td>4</td>
</tr>
<tr>
<td>145.00</td>
<td>5</td>
</tr>
<tr>
<td>155.00</td>
<td>5</td>
</tr>
<tr>
<td>165.00</td>
<td>5</td>
</tr>
<tr>
<td>175.00</td>
<td>5</td>
</tr>
<tr>
<td>185.00</td>
<td>5</td>
</tr>
<tr>
<td>195.00</td>
<td>6</td>
</tr>
</tbody>
</table>
SYN3D, OVERLAY SOLVE  3 GROUP, 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJOINT WEIGHTING

BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE

GROUP 3 OF BLANKET FUNCTIONS NOT USED AT ALL

INPUT 3D GEODEST FILE

DATA MANAGEMENT PARAMETERS FOR SOLUTION

INTGELS FILE, WORDS/RECORD = 2216

NUMBER OF RECORDS = 1

FILE, WORDS/RECORD = 1216

NUMBER OF RECORDS = 1

ADJOINT EIGENVALUE CALCULATION

EIGENVALUE ESTIMATE = 0.97620E+00, MAXIMUM NUMBER OF ITERATIONS = 20, CONVERGENCE CRITERION = 0.10000E-04

ITERATION EIGENVALUE

1   0.97541E+00
2   0.97520E+00
3   0.97520E+00
4   0.97520E+00

THE CALCULATION HAS CONVERGED ON AN EIGENVALUE

NORMALIZATION, ADJOINT * SIGMA-F * POWER/FISSION = 0.10000E+02 WATTS.
## SYM3D, OVERLAY SOLVE

3 GROUP, 3-DIMENSIONAL MODEL, MIXED FLUX AND ADJOINT WEIGHTING

**BLANKET FUNCTIONS (3 AND 4) NOT USED EVERYWHERE**

**GROUP 3 OF BLANKET FUNCTIONS NOT USED AT ALL**

**INPUT 3D GROUP FILE**

**COMBINING COEFFICIENTS FOR ADJOINT CALCULATION**

<table>
<thead>
<tr>
<th>FUNCTION NO.</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>2</th>
<th>2</th>
<th>3</th>
<th>1</th>
<th>a</th>
<th>a</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP 4</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

### AXIAL POSITION GROUP

<table>
<thead>
<tr>
<th>Position</th>
<th>Function</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>1</th>
<th>a</th>
<th>a</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.00</td>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.23E-03</td>
<td>0.94E-05</td>
<td>0.0</td>
</tr>
<tr>
<td>15.00</td>
<td>1</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>2.66E-03</td>
<td>1.84E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>25.00</td>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>4.76E-03</td>
<td>3.01E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>35.00</td>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>7.08E-03</td>
<td>4.49E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>45.00</td>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>9.19E-03</td>
<td>5.95E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>55.00</td>
<td>2</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>8.46E-03</td>
<td>5.93E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>65.00</td>
<td>3</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.32E-03</td>
<td>1.30E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>75.00</td>
<td>3</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>6.39E-03</td>
<td>1.29E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>85.00</td>
<td>3</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>6.80E-04</td>
<td>1.35E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>95.00</td>
<td>3</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>6.47E-04</td>
<td>2.13E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>105.00</td>
<td>4</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>3.82E-04</td>
<td>2.84E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>115.00</td>
<td>4</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.44E-04</td>
<td>2.02E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>125.00</td>
<td>4</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>5.13E-04</td>
<td>1.04E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>135.00</td>
<td>4</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.68E-03</td>
<td>1.51E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>145.00</td>
<td>5</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>9.56E-03</td>
<td>6.97E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>155.00</td>
<td>5</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.02E-02</td>
<td>6.55E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>165.00</td>
<td>5</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>7.81E-03</td>
<td>4.60E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>175.00</td>
<td>5</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>5.16E-03</td>
<td>2.91E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>185.00</td>
<td>6</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>3.07E-03</td>
<td>1.70E-04</td>
<td>0.0</td>
</tr>
<tr>
<td>195.00</td>
<td>6</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>0.0</td>
<td>1.33E-03</td>
<td>6.48E-05</td>
<td>0.0</td>
</tr>
</tbody>
</table>
### Combining Coefficients for Adjoint Calculation

<table>
<thead>
<tr>
<th>Function No.</th>
<th>Group</th>
<th>1</th>
<th>2</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1</td>
<td>-5.279E-01</td>
<td>1.119E-01</td>
<td>1.965E-01</td>
</tr>
<tr>
<td>25.00</td>
<td>2</td>
<td>-2.637E+00</td>
<td>3.317E-01</td>
<td>4.614E-01</td>
</tr>
<tr>
<td>35.00</td>
<td>2</td>
<td>-3.698E+00</td>
<td>7.020E-01</td>
<td>9.814E-01</td>
</tr>
<tr>
<td>55.00</td>
<td>2</td>
<td>-4.166E+00</td>
<td>1.737E+00</td>
<td>2.065E+00</td>
</tr>
<tr>
<td>65.00</td>
<td>3</td>
<td>6.685E+00</td>
<td>7.086E+00</td>
<td>7.094E+00</td>
</tr>
<tr>
<td>75.00</td>
<td>3</td>
<td>9.617E+00</td>
<td>9.453E+00</td>
<td>9.478E+00</td>
</tr>
<tr>
<td>85.00</td>
<td>3</td>
<td>1.099E+00</td>
<td>1.071E+00</td>
<td>1.075E+00</td>
</tr>
<tr>
<td>95.00</td>
<td>3</td>
<td>1.042E+01</td>
<td>1.022E+01</td>
<td>1.054E+01</td>
</tr>
<tr>
<td>105.00</td>
<td>4</td>
<td>3.677E+00</td>
<td>2.319E+00</td>
<td>6.326E+00</td>
</tr>
<tr>
<td>115.00</td>
<td>4</td>
<td>1.106E+00</td>
<td>4.493E-01</td>
<td>4.551E-02</td>
</tr>
<tr>
<td>125.00</td>
<td>4</td>
<td>3.041E+01</td>
<td>6.759E+00</td>
<td>8.073E+00</td>
</tr>
<tr>
<td>135.00</td>
<td>4</td>
<td>8.162E+02</td>
<td>6.551E+02</td>
<td>1.732E+04</td>
</tr>
<tr>
<td>145.00</td>
<td>5</td>
<td>7.049E+01</td>
<td>2.500E+01</td>
<td>2.652E+02</td>
</tr>
<tr>
<td>155.00</td>
<td>5</td>
<td>8.832E+01</td>
<td>1.865E+01</td>
<td>8.380E+03</td>
</tr>
<tr>
<td>165.00</td>
<td>5</td>
<td>7.091E+01</td>
<td>1.167E+01</td>
<td>2.621E+03</td>
</tr>
<tr>
<td>175.00</td>
<td>5</td>
<td>4.781E+01</td>
<td>7.472E+01</td>
<td>9.318E+02</td>
</tr>
<tr>
<td>185.00</td>
<td>6</td>
<td>2.749E+01</td>
<td>5.540E+01</td>
<td>1.028E+02</td>
</tr>
<tr>
<td>195.00</td>
<td>6</td>
<td>1.123E+01</td>
<td>2.209E+01</td>
<td>2.430E+00</td>
</tr>
</tbody>
</table>

**TIME SPENT IN OVERLAY SOLVE:** CP = 2.1, PP = 0.0

- SUBROUTINE REDTOC, CP = 0.0, PP = 0.0
- SUBROUTINE VINTG, CP = 0.0, PP = 0.0
- SUBROUTINE CALPNN, CP = 0.0, PP = 0.0
- SUBROUTINE SPIND, CP = 1.2, PP = 0.0
- SUBROUTINE OUTPO, CP = 0.1, PP = 0.0
APPENDIX I

CCCC Version III Standard Interface Files

The CCCC Standard Interface Files used in SYN3D are defined by the specifications in Ref. 24 with the following two additions:

(1) In the File Specification record of GEODST. The 23rd entry (one of the "Reserved" in Ref. 24) is used to specify the orientation of the (1,1) mesh triangle for triangular mesh geometries (IGOM=9 or 17) bounded, in the x-y plane, by rectangles (NTRIAG=2). In SYN3D that entry is designated "NTRIPT" and has the following definition.

\[ \text{NTRIPT} = \begin{cases} 
1 & \text{Triangle (1,1) points away from the first dimensions axis. i.e., no internal mesh line intersects the origin.} \\
2 & \text{Triangle (1,1) points towards the first dimension axis. i.e., an internal mesh line intersects the origin.} 
\end{cases} \]

(2) In the Mesh Interval Boundary record of GEODST. For triangular mesh geometries (IGOM=9 or 17) the length (L) of a side of a triangle is calculated from

\[ L = 2 \frac{\text{XMESH}(2) - \text{XMESH}(1)}{\text{IFINTS}(1)} \]

The other entries in XMESH and YMESH are never used.

The rest of this appendix consists of listings of the following CCCC Standard Interface Files from Reference 24:

- GEODST
- ISOTXS
- NDXSRF
- ZNATDN
- RTFLUX
- ATFLUX
- PWDINT
- RZFLUX
APPENDIX I. CCCC Version III Standard Interface Files. GEOBST

REVISED 07/27/75

GEOBST = III

GEOBSTY DESCRIPTION

FILE IDENTIFICATION

MNAME, (MUSE(I), I=1,2), IVERS
1 = MULT = NUMBER OF WORDS

MNAME = MOLLEWTH FILE NAME = GEOBST = (AB)
MUSE(I) = MOLLEWTH USER IDENTIFICATION (AB)
IVERS = FILE VERSION NUMBER
MULT = DOUBLE PRECISION PARAMETER
1 = AB WORD IS SINGLE WORD
2 = AB WORD IS DOUBLE PRECISION WORD

FILE SPECIFICATIONS (10 RECORD)

IGOM, NZONE, NREG, MCL, MCINTJ, MCINTK, MINTJ, MINTK, MINT, IM1, IM2, JM1, JM2, KML, KMB, AED, AEC, ARE, ATR, MB, MBB, MAB, POINT
(NGP(I), I=1,5)

2 = NUMBER OF WORDS

IGOM = GEOMETRY 0 = POINT (FUNDAMENTAL MODE)
1 = SLAB
2 = CYLINDER
3 = SPHERE
4 = X-Y
5 = X-Z
6 = Y-Z
7 = THETA-N
8 = TRIAGONAL (6 MESH POINTS IN EACH HEXAGONAL ELEMENT)
10 = HFAAGONAL (1 MESH POINT IN EACH HEXAGONAL ELEMENT)
11 = R-THETA
12 = R-THETA-Z
13 = R-THETA-ALPHA
14 = X-Y-Z
15 = THETA-N-Z
16 = R-THETA-Z
17 = TRIAGONAL-Z (MESH POINTS AS IN 9, ABOVE)
18 = HFAAGONAL-Z (MESH POINTS AS IN 10, ABOVE)
### APPENDIX I: CCC Version III Standard Interface Files. GROUND (Contd.)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NZONE</td>
<td>Number of zones (each homogeneous in neutronics)</td>
</tr>
<tr>
<td>NREG</td>
<td>Number of regions</td>
</tr>
<tr>
<td>NZCL</td>
<td>Number of zone classifications (for I purposes)</td>
</tr>
<tr>
<td>NCINTJ</td>
<td>Number of first dimension coarse mesh intervals</td>
</tr>
<tr>
<td>NINTJ</td>
<td>Number of second dimension coarse mesh intervals</td>
</tr>
<tr>
<td>NINT K</td>
<td>Number of third dimension coarse mesh intervals</td>
</tr>
<tr>
<td>NINTI</td>
<td>Number of first dimension fine mesh intervals</td>
</tr>
<tr>
<td>NINTJ</td>
<td>Number of second dimension fine mesh intervals</td>
</tr>
<tr>
<td>NINTK</td>
<td>Number of third dimension fine mesh intervals</td>
</tr>
<tr>
<td>IBOUND</td>
<td>First boundary on first dimension</td>
</tr>
<tr>
<td>IBOUND2</td>
<td>Last boundary on first dimension</td>
</tr>
<tr>
<td>JBOUND1</td>
<td>First boundary on second dimension</td>
</tr>
<tr>
<td>JBOUND2</td>
<td>Last boundary on second dimension</td>
</tr>
<tr>
<td>KBOUND1</td>
<td>First boundary on third dimension</td>
</tr>
<tr>
<td>KBOUND2</td>
<td>Last boundary on third dimension</td>
</tr>
<tr>
<td>NGBS</td>
<td>Number of coupling specifications</td>
</tr>
<tr>
<td>NCSCS</td>
<td>Number of constants for external boundaries</td>
</tr>
</tbody>
</table>

- **NZONE**: Each zone is homogeneous in neutronics.
- **NREG**: Each zone has a unique region.
- **NZCL**: Each zone contains specific classification.
- **NCINTJ**: Coarse mesh intervals for first dimension.
- **NINTJ**: Coarse mesh intervals for second dimension.
- **NINT K**: Coarse mesh intervals for third dimension.
- **NINTI**: Fine mesh intervals for first dimension.
- **NINTJ**: Fine mesh intervals for second dimension.
- **NINTK**: Fine mesh intervals for third dimension.
- **IBOUND**: First boundary condition for first dimension.
- **JBOUND1**: First boundary condition for second dimension.
- **KBOUND1**: First boundary condition for third dimension.
- **NGBS**: Specifies coupling for each zone.
- **NCSCS**: Specifies constants for each external boundary.

**Note**: Boundary conditions are case-sensitive.
APPENDIX 1. CECX Version III Standard Interface Files. GHOST (Contd.)

**NGBCS**
Number of Constants for Internal Boundaries
1. Single values used everywhere
2. **LT.1** - Values are given by energy group
3. 1st non-black condition indicated by zero entry - last value applies to additional groups

**NZNO**
Number of Zones which are black absorbers

**NTROG**
Triangular Geometry Option
0. Rhombus with coordinates at 120 degrees
1. Origin is at the center of a hexagonal assembly, boundaries pass through corners of hexagonal assemblies,
2. Same as option 0 except coordinates at 60 degrees
3. First boundary perpendicular to hexagonal flat.
4. Equilateral (60 degree) triangle, two boundaries originating at center of hexagonal assembly pass through corners of hexagonal assemblies.
5. Triangle (30-60 degree), first boundary perpendicular to flat.

**NRASS**
Region Assignments
0. To Coarse Mesh
1. To Fine Mesh
**NGOP(1)**
Reserved

**ONE DIMENSIONAL COARSE MESH INTERVAL BOUNDARIES AND FINE MESH INTERVALS (20 RECORD)**

Present if IGCx,1,d, and IGCx,M,E,j

**NCOND**
Number of Cond.

**KMBH(I)**
Coarse Mesh Boundaries, First Dimension

**IFINTS(I)**
Number of Fine Mesh Intervals per Coarse Mesh

**ICND(I)**
NCINT(I)+1, Number of First Dimension Coarse Mesh Boundaries

Units are cm for linear dimensions and radians for angular dimensions
APPENDIX I. CCCC Version III Standard Interface Files. GEOBST (Contd.)

---

**TWO DIMENSIONAL COARSE MESH INTERVAL BOUNDARIES AND FINE MESH INTERVALS (3D RECORD)**

**PRESENT IF IGOM, GE, O AND IGOM, IE, II**

- X_MESH(I), J_MESH(I) - Coarse mesh boundaries, second dimension
- J_FINTS(I) - Number of fine mesh intervals per coarse mesh interval, second dimension
- NCBNDJ - NCINTJ+1, number of second dimension coarse mesh boundaries

---

**THREE DIMENSIONAL COARSE MESH INTERVAL BOUNDARIES AND FINE MESH INTERVALS (3D RECORD)**

**PRESENT IF IGOM, GE, 12**

- X_MESH(I), K_MESH(I) - Coarse mesh boundaries, third dimension
- K_FINTS(I) - Number of fine mesh intervals per coarse mesh interval, third dimension
- NCBNDK - NCINTK+1, number of third dimension coarse mesh boundaries

---

**GEOMETRY DATA (5D RECORD)**

**PRESENT IF IGOM, GT, U OR NGB, GT, U**

- VOL(N), NBG, NBG = Region volumes (CC)
- BSEC(N) = Buckling (Q=2) values (CH=2)
- BNOC(N) = Boundary constants (DEL PHI/PHI = C/D)
- BNOCI(N) = Internal black boundary constants
- NZMBB(N) = Zone numbers with black absorber conditions
- NZC(N) = Zone classifications
APPENDIX I. CCC Version III Standard Interface Files. GEODST (Contd.)

CD NZNH(N) ZONE NUMBER ASSIGNED TO EACH REGION

CR REGION ASSIGNMENTS TO COARSE MESH INTERVALS
(6D RECORD)
CR
CC PRESENT IF IGOM.GT.0 AND NRASS.EQ.0
CC
CL ((NH(I,J),I=1,NCINTI),J=1,NCINTJ))--NOTE STRUCTURE BELOW-----
CC
CN NCINTI*NCINTJ=NUMBER OF NODES
CC
CS DO 1 K=1,NCINTK
CS 1 READ(N) *LIST AS ABOVE*
CD
CM MR(I,J) REGION NUMBERS ASSIGNED TO COARSE MESH INTERVALS

CR REGION ASSIGNMENTS TO FINE MESH INTERVALS
(7D RECORD)
CR
CC PRESENT IF IGOM.GT.0 AND NRASS.EQ.1
CC
CL ((MR(I,J),I=1,NINTI),J=1,NINTJ))--NOTE STRUCTURE BELOW-----
CC
CN NINTI*NINTJ=NUMBER OF NODES
CS DO 1 K=1,NINTK
CS 1 READ(N) *LIST AS ABOVE*
CD
CM MR(I,J) REGION NUMBERS ASSIGNED TO FINE MESH INTERVALS

CLOF
APPENDIX I. CCCC Version III Standard Interface Files, ISOTXS

---

FILE STRUCTURE

- RECORD TYPE
- PRESENT IF
- FILE IDENTIFICATION
  AUTHOR, USER NAME, ISOTYPE, TITLE, SOURCE
- FILE CONTROL
  FILE IDENTIFICATION, FILE NAME, FILE DESCRIPTION
- FILE DATA
  GROUP NEUTRON CROSS SECTIONS
- FILE IDENTIFICATION
  FILE NAME, FILE DESCRIPTION
- FILE CONTROL
  FILE IDENTIFICATION, FILE NAME, FILE DESCRIPTION
- FILE DATA
  GROUP NEUTRON CROSS SECTIONS
- FILE IDENTIFICATION
  FILE NAME, FILE DESCRIPTION
- FILE CONTROL
  FILE IDENTIFICATION, FILE NAME, FILE DESCRIPTION
- FILE DATA
  GROUP NEUTRON CROSS SECTIONS

---

FILE IDENTIFICATION

- MNAME, NUMEB (1), IS1, IS2
- 103 MULTNAMNBER OF WORDS
- FORMAT: (1M50 ISOTYPES, 1A8, 1M14, 1M14, 1M14)
- MNAME: HOLLERIHM FILE NAME
  ISOTYPES: (A6)
- MUXB (1): HOLLERITH USER IDENTIFICATION (A4)
- IERS: FILE VERSION NUMBER
- NTL: DOUBLE PRECISION PARAMETER
- NTL = 1: A6 WORD IS SINGLE WORD
- NTL = 2: A6 WORD IS DOUBLE PRECISION WORD

---

FILE STRUCTURE

- RECORD TYPE
- PRESENT IF
- FILE IDENTIFICATION
  AUTHOR, USER NAME, ISOTYPE, TITLE, SOURCE
- FILE CONTROL
  FILE IDENTIFICATION, FILE NAME, FILE DESCRIPTION
- FILE DATA
  GROUP NEUTRON CROSS SECTIONS
- FILE IDENTIFICATION
  FILE NAME, FILE DESCRIPTION
- FILE CONTROL
  FILE IDENTIFICATION, FILE NAME, FILE DESCRIPTION
- FILE DATA
  GROUP NEUTRON CROSS SECTIONS
- FILE IDENTIFICATION
  FILE NAME, FILE DESCRIPTION
- FILE CONTROL
  FILE IDENTIFICATION, FILE NAME, FILE DESCRIPTION
- FILE DATA
  GROUP NEUTRON CROSS SECTIONS

---

FILE IDENTIFICATION

- MNAME, NUMEB (1), IS1, IS2
- 103 MULTNAMNBER OF WORDS
- FORMAT: (1M50 ISOTYPES, 1A8, 1M14, 1M14, 1M14)
- MNAME: HOLLERIHM FILE NAME
  ISOTYPES: (A6)
- MUXB (1): HOLLERITH USER IDENTIFICATION (A4)
- IERS: FILE VERSION NUMBER
- NTL: DOUBLE PRECISION PARAMETER
- NTL = 1: A6 WORD IS SINGLE WORD
- NTL = 2: A6 WORD IS DOUBLE PRECISION WORD
APPENDIX 1. CCCC Version III Standard Interface Files. ISOTXS (Contd.)

FILE CONTROL (1D RECORD)

<table>
<thead>
<tr>
<th>C</th>
<th>CR</th>
<th>FILE CONTROL (1D RECORD)</th>
</tr>
</thead>
</table>
| C | CL | C
| C | CL | FILE CONTROL (1D RECORD)
| C | =  | CR
| C | =  | CL
| C | =  | nginx, niso, maxup, maxon, maxord, ichist, nscmax, nsblk
| C | =  | C
| C | =  | A=NUMBER OF WORDS
| C | =  | CB
| C | =  | FORMAT (4M1D, A16)
| C | =  | CD
| C | =  | ngroup, number of energy groups in set
| C | =  | CD
| C | =  | NISO, number of isotopes in set
| C | =  | CD
| C | =  | maxup, maximum number of backscatter groups
| C | =  | CD
| C | =  | MAXON, maximum number of downscatter groups
| C | =  | CD
| C | =  | MAXORD, maximum scattering order (maximum value of
| C | =  | CD
| C | =  | LEGENDRE EXPANSION INDEX USED IN FILE)
| C | =  | CD
| C | =  | ICHIST, set fission spectrum flag
| C | =  | CD
| C | =  | ICHIST, EQ, 0, no fission spectrum
| C | =  | CD
| C | =  | ICHIST, EQ, 1, set vector
| C | =  | CD
| C | =  | ICHIST, GT, 1, set matrix
| C | =  | CD
| C | =  | NSCMAX, maximum number of blocks of scattering data
| C | =  | CD
| C | =  | NSBLK, blocking control for scattering matrices, the
| C | =  | CD
| C | =  | scattering data are blocked into NSBLK
| C | =  | CD
| C | =  | RECORDS PER SCATTERING BLOCK.

FILL DATA (2D RECORD)

<table>
<thead>
<tr>
<th>C</th>
<th>CR</th>
<th>FILL DATA (2D RECORD)</th>
</tr>
</thead>
</table>
| C | CL | C
| C | CL | (MSETID(I), I=1, 12), (MISONM(I), I=1, NISO),
| C | CL | I(CHI(J), J=1, NGROUP), (VEL(J), J=1, NGROUP),
| C | CL | Z(EMAX(J), J=1, NGROUP), EMIN, (LOCA(I), I=1, NISO)
| C | CL | (NISO+12)*MULT+10*NISO
| C | CL | NGROUP*(2+ICHIST)*(2/(ICHIST+1))=NUMBER OF WORDS
| C | CL | CB
| C | CL | FORMAT (4M2V, 1M8, 1A6, 1Ho/ MSETID, MISONM
| C | CL | CB
| C | CL | 1M8, 1A6, 1H0, 9(1X, A6)/(10(1X, A6))
| C | CL | CB
| C | CL | FORMAT (IPE12, 5) CHI (PRESENT IF ICHIST, EQ, 1)
| C | CL | CB
| C | CL | FORMAT (IPE12, 5) VEL, EMAX, EMIN
| C | CL | CB
| C | CL | FORMAT (12I6)
| C | CL | HOLLERITH IDENTIFICATION OF SET (A6)
| C | CL | CB
| C | CL | HOLLERITH ISOTOPE LABEL FOR ISOTOPE I (A6)
| C | CL | CB
| C | CL | CHI(J), SET FISSION SPECTRUM (PRESENT IF ICHIST, EQ, 1)
| C | CL | CB
| C | CL | VEL(J), MEAN NEUTRON VELOCITY IN GROUP J (CM/SEC)
| C | CL | CB
| C | CL | EMAX(J), MAXIMUM ENERGY BOUND OF GROUP J (EV)
| C | CL | CB
| C | CL | EMIN, MINIMUM ENERGY BOUND OF SET (EV)
| C | CL | CB
| C | CL | LOCA(I), NUMBER OF RECORDS TO BE SKIPPED TO READ DATA FOR
| C | CL | CB
| C | CL | ISOTOPE I. LOCA(1)=0
APPENDIX I. CCCC Version III Standard Interface Files. 1.xTXS (Contd.)

---

**SET CHI DATA (3D RECORD)**

Present if \( ICHIST > 1 \)

\[
((CHI(K,J), K=1, ICHIST), J=1, NGROUP), (ISSPEC(I), I=1, NGROUP)
\]

NGROUP * ICHIST = NUMBER OF WORDS

**FORMAT**

\((4N 30, 1P5E12.5/(6E12.5)) CHI, ISSPEC\)

**FORMAT**

\((1216)\)

**CHI(K,J)**

Fraction of neutrons emitted in group J as a result of fission in any group using spectrum K.

**ISSPEC(I)**

ISSPEC(I) = 1 IMPLIES THAT SPECTRUM K IS USED TO CALCULATE EMISSION SPECTRUM FROM FISSION IN GROUP I.

---

**ISOPOPE CONTROL AND GROUP INDEPENDENT DATA (4D RECORD)**

**HABSID, MIDENT, WMAT, AMASS, EFIBS, ECAPS, TEMP, BIGSPAN, ADENS, ABR, ICHI,**

**IFIS, IALP, IMP, IN2, IND, INT, LOTT, LTRN, ISTRPO,**

**2(IIDCT(N), N=1, NSMAX), (LOND(N), N=1, NSMAX),**

**5((JAND(J,N), N=1, NGROUP), N=1, NSMAX),**

**4((IJJ(J,N), J=1, NGROUP), N=1, NSMAX),**

3 * MULT * NSMAX = (2 * NGROUP * 2) = NUMBER OF WORDS

**FORMAT**

\((4N 4D, 3(IX, 16), 1P5E12.9/(1216))\)

**HABSID**

HOLLERITH ABSOLUTE ISOTOPE LABEL - SAME FOR ALL VERSIONS OF THE SAME ISOPOPE IN SET (A6)

**MIDENT**

IDENTIFIER OF LIBRARY FROM WHICH BASIC DATA CAPE(E, C, ENDF/N) (A6)

**WMAT**

ISOTOPE IDENTIFICATION (E, C, ENDF/B WMAT NO.) (A6)

**AMASS**

GRAN ATOMIC WEIGHT

**EFIBS**

TOTAL THERMAL ENERGY YIELD/FISSION (W, SEC/FISSION)

**ECAPS**

TOTAL THERMAL ENERGY YIELD/CAPTURE (W, SEC/CAPT)

**TEMP**

ISOTOPE TEMPERATURE (DEGREES KELVIN)

**BIGSPAN**

AVERAGE EFFECTIVE POTENTIAL SCATTERING IN RESONANCE RANGE (BARN/ATOM)

**ADENS**

DENSITY OF ISOTOPE IN MIXTURE IN WHICH ISOPOPE CROSS SECTIONS WERE GENERATED (A/BARN, CH)

**XSR**

ISOTOPE CLASSIFICATION

0 = UNDEFINED

1 = FISSILE

2 = FERTILE

3 = OTHER ACTINIDE

4 = FISSION PRODUCT

5 = STRUCTURE

6 = COOLANT

7 = CONTROL

**ICH1**

ISOTOPE FISSION SPECTRUM FLAG
APPLIADX 1. CCC Version 11 Standard Interface Files. ISOTXS (Contd.)

**CD**
- ICHI, EQ, A, USE SET CHI
- ICMZ, EQ, I, ISOPORE CHI VECTOR
- ICMZ, EQ, C, ISOPORE CHI MATRIX

**CD IFIS**
- (N,F) CROSS SECTION FLAG
  - IFIS, NO FISSION DATA IN PRINCIPAL CROSS SECTION RECORD
  - IFIS, FISSION DATA PRESENT IN PRINCIPAL CROSS SECTION RECORD

**CD IALF**
- (N, AL) CROSS SECTION FLAG
  - SAME OPTIONS AS IFIS

**CD INP**
- (N,P) CROSS SECTION FLAG
  - SAME OPTIONS AS IFIS

**CD INZM**
- (N,2N) CROSS SECTION FLAG
  - SAME OPTIONS AS IFIS

**CD IHR**
- (N, O) CROSS SECTION FLAG
  - SAME OPTIONS AS IFIS

**CD IMT**
- (N, T) CROSS SECTION FLAG
  - SAME OPTIONS AS IFIS

**CD LTO1**
- NUMBER OF MOMENTS OF TOTAL CROSS SECTION PROVIDED IN PRINCIPAL CROSS SECTIONS RECORD

**CD LTRN**
- NUMBER OF MOMENTS OF TRANSPORT CROSS SECTION PROVIDED IN PRINCIPAL CROSS SECTIONS RECORD

**CD ISTRP**
- NUMBER OF COORDINATE DIRECTIONS FOR WHICH COORDINATE DEPENDENT TRANSPORT CROSS SECTIONS ARE GIVEN, IF ISTRP = 0, NO COORDINATE DEPENDENT TRANSPORT CROSS SECTIONS ARE GIVEN.

**CD IDSCT(N)**
- SCATTERING MATRIX TYPE IDENTIFICATION FOR SCATTERING BLOCK N, SIGNIFICANT ONLY IF LORD(N) = 1

**CD IDSCT(N) = 00 E**
- TOTAL SCATTERING (SUM OF ELASTIC, INELASTIC, AND N, 2N SCATTERING)

**CD IDSCT(N) = 01 E**
- ELASTIC SCATTERING

**CD IDSCT(N) = 02 E**
- INELASTIC SCATTERING

**CD IDSCT(N) = 03 E**
- (N, 2N) SCATTERING PER EMITTED NEUTRON,

**CD LORD(N)**
- NUMBER OF SCATTERING ORDERS IN BLOCK N, IF LORD(N) = 0, THIS BLOCK IS NOT PRESENT FOR THIS ISOTOPE, IF NN IS THE VALUE TAKEN FROM IDSCT(N), THEN THE MATRICES IN THIS BLOCK HAVE LEGENDRE EXPANSION INDICES OF NN, NN+1, NN+2, ..., NN+LORD(N)-1

**CD JBLA(N)**
- SCATTERING BANDWIDTH FOR GROUP J, SCATTERING BLOCK N

**CD IJJ(N)**
- POSITION OF IN-GROUP SCATTERING CROSS SECTION IN SCATTERING DATA FOR GROUP J, SCATTERING BLOCK N, COUNTED FROM THE FIRST WORD OF GROUP J DATA.
APPENDIX I. CCC Version III Standard Interface Files. ISOTXS (Contd.)

--- PRINCIPAL CROSS SECTIONS (5D RECORD) ---

```
((STRPL(J,L),J=1,NGROUP),L=1,LTRN),
1((STOTPL(J,L),J=1,NGROUP),L=1,LTOT),
(SNGAM(J),J=1,NGROUP),
2(SFIS(J),J=1,NGROUP),
(SNUTOT(J),J=1,NGROUP),
3(ISOPEC(I),I=1,ISOTPD),
(SN2N(J),J=1,NGROUP),
(SNT(J),J=1,NGROUP),
4(SNP(J),J=1,NGROUP),
5(SND(J),J=1,NGROUP),
6((STRPD(J,I),J=1,NGROUP),I=1,ISTRPD)
```

**PL WEIGHTED TRANSPORT CROSS SECTION**

**PL WEIGHTED TOTAL CROSS SECTION**

The first element of array STRPL is the current (P1) weighted transport cross section.

The first element of array STOTPL is the flux (P0) weighted total cross section.

```
SNGAM(J)  (NGAMMA)
SFIS(J)   (NF)   (PRESENT IF IFIS.GT.0)
SNUTOT(J) TOTAL NEUTRON YIELD/FISSION (PRESENT IF IFIS.GT.0)
CHISO(J)  ISOTOPIC CHI (PRESENT IF ICMI.GT.1)
SNALF(J)  (N,ALPHA)
SNP(J)    (NP)   (PRESENT IF INP.GT.0)
SN2N(J)   (N,2N) (LOSS) (PRESENT IF IN2N.GT.0)
SN(T)     (NT)   (PRESENT IF IND.GT.0)
STRPD(J,I) COORDINATE DIRECTION I TRANSPORT CROSS SECTION
```

--- ISOTOPIC CHI DATA (6D RECORD) ---

```
((CHISO(K,J),K=1,ICMI),J=1,NGROUP),(ISOPEC(I),I=1,NGROUP)
```

**PRESENT IF ICMI.GT.1**

```
CHISO(K,J) FRACTION OF NEUTRONS EMITTED IN GROUP J AS A RESULT OF FISSION IN ANY GROUP USING SPECTRUM K
ISOPEC(I) ISOPEC(I).K IMPLIES THAT SPECTRUM K IS USED TO CALCULATE EMISSION SPECTRUM FROM FISSION IN GROUP I
```

--- FORMAT(4H 50,1P5E,25/(6E12,5)) LENGTH OF LIST AS ABOVE ---

--- FORMAT(4H 6D,1P5E12.5/(6E12.5)) CHISO ISOPEC ---

---"
APPENDIX I. CCCC Version III Standard Interface Files. ISOTXS (Contd.)

CC SCATTPING SUB-BLOCK (7D RECORD)
CC
CC PRESENT IF LORD(N),GT,0
CC
CC ((SCAT(K,L),K=1,KMAX),L=1,LORDN)
CC
CC KMAX=SUM OVER J OF JBAND(J) WITHIN THE J-GROUP RANGE OF THIS
CC RANGE CONTAINED WITHIN THIS SUB-BLOCK IS
CC
CC JL=(M-1)*((NGROUP-1)/NSBLOK+1)+1 TO JU=M*(((NGROUP-1)/NSBLOK+1)
CC
CC LORDN=LORD(N)
CC
CC KMAX=LORDN=NUMBER OF WORDS
CC
CC FORMAT(4H 7U 1P5L12,5/(bE12.5))
CC
CC SCAT(K,L) SCATTERING MATRIX OF SCATTERING ORDER L, FOR
CC REACTION TYPE IDENTIFIED BY IDSCT(N) FOR THIS
CC BLOCK. JBAND(J) VALUES FOR SCATTERING INTO
CC GROUP J ARE STORED AT LOCATIONS K=SUM FROM 1
CC TO (J-1) OF JBAND(1) PLUS 1 TO K=1+JBAND(J),
CC THE SUM IS ZERO WHEN J=1, J=TO=J SCATTER IS
CC THE IJJ(J)-TH ENTRY IN THE RANGE JBAND(J),
CC VALUES ARE STORED IN THE ORDER (J*JUP),
CC
CC (J*JUP=1), (J*JUP=2), ..., (J*JUP=J),
CC WHERE JUP=IJJ(J)-1 AND JDN=JBAND(J)=IJJ(J)
CC
CEOF
APPENDIX I. CCCC Version III Standard Interface Files. NDXSRF

REVISED 07/22/75

NDXSRF * III

NUCLEIDE DENSITY, DATA, CROSS SECTION REFERENCING

FILE IDENTIFICATION

HNAME,(HUSE(I),I=1,2),IVERS

1+3*MULT=NUMBER OF WORDS

HNAME HOLLERITH FILE NAME = NDXSRF = (A6)
HUSE(I) HOLLERITH USER IDENTIFICATION (A6)
IVERs FILE VERSION NUMBER
MULT DOUBLE PRECISION PARAMETER
1=A6 WORD IS SINGLE WORD
2=A6 WORD IS DOUBLE PRECISION WORD

SPECIFICATIONS (1D RECORD)

NON,NSN,NNS,NAN,NZONE,NSZ

6 = NUMBER OF WORDS

NON NUMBER OF NUCLIDES IN CROSS SECTION DATA
NSN NUMBER OF NUCLIDE SETS IDENTIFIED
NNS MAXIMUM NUMBER OF NUCLIDES IN ANY SET
NAN NUMBER OF DIFFERENT NUCLIDES IN DATA
NZONE NUMBER OF ZONES
NSZ NUMBER OF SUBZONES (SUBASSEMBLIES)

NUCLEIDE REFERENCING DATA (2D RECORD)

(HNAME(N),N=1,NON),(HNAME(N),N=1,NON),(MPF(N),N=1,NON),
(AT(J),J=1,NAN),(NL(N),N=1,NON),(NDXS(L),L=1,NSN),
((NOSE(I),I=1,NNS),L=1,NSN),((NONE(N),N=1,NON),L=1,NSN)

NAN+2*NON*(1+MULT)+NNS*(4+NNS+NON)=NUMBER OF WORDS

HNAME(N) UNIQUE REFERENCE NUCLEIDE NAME, IN LIBRARY ORDER =
(A6) ALPHANUMERIC
HNAME(N) ABSOLUTE NUCLEIDE NAME, IN LIBRARY ORDER =
(A6) ALPHANUMERIC
APPENDIX I. CCCC Version III Standard Interface Files. NDXSRF (Contd.)

CD WPF(N) RESERVED
CD ATMT(J) ATOMIC WEIGHT
CD NCLN(N) NUCLIDE CLASSIFICATION
   1 = FISSION
   2 = FERTILE
   3 = OTHER ACTINIDE
   4 = FISSION PRODUCT
   5 = STRUCTURAL
   6 = COOLANT
   7 = CONTROL MOD
   GREATER THAN 7, UNDEFINED
CD NDXS(K,L) REFERENCE DATA FOR SET L
   K = 1, NUMBER OF NUCLIDES IN SET
   K = 2, RESERVED
   K = 3, RESERVED
   K = 4, RESERVED
CD NDS(I,L) ORDER NUMBER OF NUCLIDE IN CROSS SECTION DATA
   (IN NAME LIST) OF NUCLIDE ORDERED I IN SET L
CD NDR(N,L) ORDER NUMBER OF NUCLIDE IN SET L GIVEN ORDER
   NUMBER N IN CROSS SECTION DATA

------ NUCLIDE CONCENTRATION ASSIGNMENT DATA (3D RECORD) -----
CD (VOLZ(N), N=1, NZONE), (VFPA(N), N=1, NZONE), (VLSA(M), M=1, NSZ),
CD (NSPA(N), N=1, NZONE), (NSSA(M), M=1, NSZ), (NZSZ(M), M=1, NSZ)
CD 3*(NZONE+NSZ) NUMBER OF WORDS
CD VOLZ(N) VOLUMES OF ZONES, CC
CD VFPA(N) VOLUME FRACTIONS FOR PRIMARY ZONE ASSIGNMENTS
CD VLSA(M) VOLUMES OF SUBZONES
CD NSPA(N) NUCLIDE SET REFERENCE, PRIMARY ZONE ASSIGNMENT
   (MAY BE ZERO ONLY IF THERE ARE SUBZONES)
CD NSSA(M) NUCLIDE SET REFERENCE ASSIGNMENT TO SUBZONES
CD NZSZ(M) ZONE CONTAINING SUBZONE

NOTE THAT TO CALCULATE MACROSCOPIC CROSS SECTIONS FOR A ZONE,
IT IS NECESSARY TO CONSIDER THE CONCENTRATION OF EACH NUCLIDE
IN THE PRIMARY SET ASSIGNMENT (UNLESS A ZERO IN NSPA INDICATES
THERE ARE NONE) TIMES THE VOLUME FRACTION, AND THE CONCENTRATION
OF EACH NUCLIDE IN EACH SUBZONE ASSIGNED TO THE ZONE TIMES THE
RATIO OF THE SUBZONE VOLUME TO THE ZONE VOLUME.
APPENDIX I. CCCC Version III Standard Interface Files. ZNATDN

FILE IDENTIFICATION

%NAME, (HUSE(I), I=1, 2), IVERS

1+3=MULT=NUMBER OF WORDS

%NAME = MULLERITH FILE NAME = ZNATDN = (A6)
%HUSE(I) = MULLERITH USER IDENTIFICATION = (A6)
%IVER = FILE VERSION NUMBER
%MULT = DOUBLE PRECISION PARAMETER
1 = A6 WORD IN SINGLE WORD
2 = A6 WORD IN DOUBLE PRECISION WORD

SPECIFICATIONS (1D RECORD)

TIME,NCY, NTZSZ, NNS, NBLKAD

5=NUMBER OF WORDS

TIME = REFERENCE REAL TIME, DAYS
NCY = REFERENCE CYCLE NUMBER
NTZSZ = NUMBER OF ZONES PLUS NUMBER OF SUBZONES
NNS = MAXIMUM NUMBER OF NUCLIDES IN ANY SET
NBLKAD = NUMBER OF BLOCKS OF ATOMIC DENSITY DATA

ZONE ATOMIC DENSITIES (OF NUCLIDES) (2D RECORD)

((ADEN(N,J), N=1, NNS), J=1L, JU) ---SEE STRUCTURE WFD---

NNS=((NTZSZ-1)/NBLKAD+1)=NUMBER OF WORDS

DO 1 M=1,NBLKAD
1 READ(N) *LIST AS ABOVE*

WITH M AS THE BLOCK INDEX, JL=1(N-1)*(NTZSZ-1)/NBLKAD+1)
AND JU=M*((NTZSZ-1)/NBLKAD+1)
APPENDIX I. CCCC Version III Standard Interface Files. ZNATION (Contd.)

C
CD ADEN(M,J) ATOMIC DENSITY OF NUCLIDE ORDERED M IN THE
CD ASSOCIATED SFT GIVEN IN ORDER FOR EACH ZONE
CD FOLLOWED IN ORDER FOR EACH SUBZONE
C

CEOF
APPENDIX I. CCCC Version III Standard Interface Files. RTFLUX

REVISED 01/23/75

RTFLUX-III
REGULAR TOTAL FLUXES

ORDER OF GROUPS IS ACCORDING TO DECREASING ENERGIES. NOTE THAT DOUBLE PRECISION FLUXES ARE GIVEN WHEN MULT = 2.

FILE IDENTIFICATION

NAME, (MUSF(I), I=1,2), INVRS
1+3*MULT=NUMBER OF WORDS
NAME HOLLERITH FILE NAME = RTFLUX = (A6)
MUSE(I) HOLLERITH USER IDENTIFICATION (A6)
INVRS FILE VERSION NUMBER
MULT DOUBLE PRECISION PARAMETER
1 = A6 WORD IS SINGLE WORD
2 = A6 WORD IS DOUBLE PRECISION WORD

SPECIFICATIONS (1D RECORD)

NUIM, NGROUP, NINTI, NINTJ, NINTK, ITER, EFK, POWER
A NUMBER OF WORDS

NDIM NUMBER OF DIMENSIONS
NGROUP NUMBER OF ENERGY GROUPS
NINTI NUMBER OF FIRST DIMENSION FINE MESH INTERVALS
NINTJ NUMBER OF SECOND DIMENSION FINE MESH INTERVALS
NINTK NUMBER OF THIRD DIMENSION FINE MESH INTERVALS
NINTK, EQ.1 IF NDIM, LE, 2
ITER OUTER ITERATION NUMBER AT WHICH FLUX WAS WRITTEN
EFK EFFECTIVE MULTIPLICATION FACTOR
POWER POWER IN WATTS TO WHICH FLUX IS NORMALIZED

ONE DIMENSIONAL REGULAR TOTAL FLUX (2D RECORD)

PRESENT IF NOIM, EQ.1

((FREG(I,J), I=1, NINTI), J=1, NGROUP)
APPENDIX I. CCCC Version III Standard Interface Files. RTFLUX (Contd.)

C* NINTI*NGROUP*MULT=NUMBER OF WORDS
C
CD FNREG(I,J) ONE DIMENSIONAL REGULAR TOTAL FLUX BY INTERVAL AND GROUP.
CD
C-----------------------------------------------

C* MULTl-DIMENSIONAL REGULAR TOTAL FLUX (3D RECORD)
C
CC PRESENT IF NDIM GE 2
CC
CL ((FNREG(I,J),I=1,NINTI),J=1,NINTJ) ---NOTE STRUCTURE BELOW---
CM NINTI*NINTJ*MULT=NUMBER OF WORDS
CC
DO 1 L=1,NGROUP
DO 1 K=1,NINTK
1 READ(N) LIST AS ABOVE *
CD FNREG(I,J) MULTl-DIMENSIONAL REGULAR TOTAL FLUX
CD BY INTERVAL AND GROUP.
C
C-----------------------------------------------
CEND
APPENDIX I. CCCC Version III Standard Interface Files. ATFLUX

---

FILE IDENTIFICATION

- **HNAME**, (**MUSE(I), I=1,2**), **IVERS**
- **1+3**=MULT=NUMBER OF WORDS
- **HNAME** - MOLLERITH FILE NAME - ATFLUX = (A6)
- **MUSE(I)** - MOLLERITH USER IDENTIFICATION (A6)
- **IVERS** - FILE VERSION NUMBER
- **MULT** - DOUBLE PRECISION PARAMETER
  - 1 - A6 WORD IS SINGLE WORD
  - 2 - A6 WORD IS DOUBLE PRECISION WORD

SPECIFICATIONS (10 RECORD)

- **NOSM**, **NGROUP**, **NINTI**, **NINTJ**, **NINTK**, **ITEM**, **EFFK**, **ADUM**
- **NOSM** - NUMBER OF DIMENSIONS
- **NGROUP** - NUMBER OF ENERGY GROUPS
- **NINTI** - NUMBER OF FIRST DIMENSION FINE MESH INTERVALS
- **NINTJ** - NUMBER OF SECOND DIMENSION FINE MESH INTERVALS
- **NINTK** - NUMBER OF THIRD DIMENSION FINE MESH INTERVALS
- **ITEM** - OUTER ITERATION NUMBER AT WHICH FLUX WAS WRITTEN
- **EFFK** - EFFECTIVE MULTIPLICATION FACTOR
- **ADUM** - RESERVED

ONE DIMENSIONAL ADJOINT TOTAL FLUX (2D RECORD)

- PRESENT IF NOSM.EQ.1
- (**PADJ(I,J), I=1, NINTI), J=1,NGROUP**)

---

ORDER OF GROUPS IS ACCORDING TO INCREASING ENERGY. NOTE THAT DOUBLE PRECISION FLUXES ARE GIVEN WHEN MULT.EQ.2.
APPENDIX I. CCCC Version III Standard Interface Files. ATFLUX (Contd.)

C

NINTI*NGROUP*MULT=NUMBER OF WORDS
C

FAOJ(J,J) ONE DIMENSIONAL ADJOINT TOTAL FLUX BY INTERVAL AND GROUP,
C

C-------------------------------
C
C
CR

MULTI-DIMENSIONAL ADJOINT TOTAL FLUX (3D RECORD)
C

PRESENT IF NDIM,GE,2
C

((FAOJ(I,J),I=1,NINTI),J=1,NINTJ) NOTE STRUCTURE BELOW
C

NINTI*NINTJ*MULT=NUMBER OF WORDS
C

DO L=1,NGROUP
DO K=1,NINTK
1 READ(N) *LIST AS ABOVE*
C

FAOJ(I,J) MULTI-DIMENSIONAL ADJOINT TOTAL FLUX
C

BY INTERVAL AND GROUP,
C

C-------------------------------
C

CEOF
APPENDIX I. CCCC Version III Standard Interface Files. PWDINT

REVISED 07/23/75

FILE IDENTIFICATION

MNAME, (MUSF(I), I=1,2), IVERS

1+3=MULT=NUMBER OF WORDS

MNAME = MOLLERNTH FILE NAME - PWDINT - (A6)
MUSF(I) = MOLLERNITH USER IDENTIFICATION (A6)
IVERS = FILE VERSION NUMBER
MULT = DOUBLE PRECISION PARAMETER

1= A6 WORD IS SINGLE WORD
2= A6 WORD IS DOUBLE PRECISION WORD

SPECIFICATIONS (10 RECORD)

TIME, POWER, VOL, TH, JM, KM, NCY

7=NUMBER OF WORDS

TIME = REFERENCE REAL TIME, DAYS
POWER = POWER LEVEL FOR ACTUAL NEUTRONICS PROBLEM,
VOL = VOLUME OVER WHICH POWER WAS DETERMINED,
TH = NUMBER OF FIRST DIMENSION FINE INTERVALS
JM = NUMBER OF SECOND DIMENSION FINE INTERVALS
KM = NUMBER OF THIRD DIMENSION FINE INTERVALS
NCY = REFERENCE COUNT (CYCLE NUMBER)

POWER DENSITY VALUES (20 RECORD)

(D=RI,J), I=1,1M, J=1,JM) NOTE STRUCTURE BELOW

IMJ=NUMBER OF WORDS

DU=I AM,AM

I READ(N) = LIST AS ABOVE

P=RI,J = POWER DENSITY BY INTERVAL, WATTS/CC
APPENDIX I. CCCC Version III Standard Interface Files. RZFLUX

REVISED 07/23/75

FILE IDENTIFICATION

HNAMF, (MUSE(I), I=1,2), IVFRS

1+3*MULT = NUMBER OF WORDS

HNAMF = HOLLERITH FILE NAME = RZFLUX = (A6)
MUSE(I) = HOLLFRITH USER IDENTIFICATION (A6)
IVFRS = FILE VERSION NUMBER
MULT = DOUBLE PRECISION PARAMETER

1 = A6 word is single word
2 = A6 word is double precision word

SPECIFICATIONS (10 RECORD)

TIME, POMEH, VOL, EFFK, EIVS, DNOS, TNL, TNA, TNGL, TNL, TNLAL, TNCRA,

2B = NUMBER OF WORDS

TIME = REFERENCE REAL TIME, DAYS
POMEH = POWER LEVEL FOR ACTUAL NEUTRONICS PROBLEM, WATTS
VOL = VOLUME OVER WHICH POWER WAS DETERMINED, CC
EFFK = MULTIPLICATION FACTOR
EIVS = EIGENVALUE OF SEARCH OF SEARCH PROBLEM
DNOS = DERIVATIVE OF SEARCH PROBLEM
TNL = TOTAL NEUTRON LOSSES
TNA = TOTAL NEUTRON ABSORPTIONS
TNGL = TOTAL NEUTRON SURFACE LEAKAGE
TNLAL = TOTAL NEUTRON BUCKLING LOSS
TNCRA = TOTAL NEUTRON BLACK ABSORBER LOSS
TNCRA = TOTAL NEUTRON CONTRUO ROD ABSORPTIONS
K(I), I=1,4 = RESIZED
ITPS = ITERATIVE PROCESS STATE
00, NO ITERATIONS DONE
01, CONVERGENCE SATISFIED
02, NOT CONVERGED, BUT CONVERGING
03, NOT CONVERGED, NOT CONVERGING
NZONE = NUMBER OF GEOMETRIC ZONES
NGROUP = NUMBER OF NEUTRON ENERGY GROUPS
NCY = REFERENCE COUNT (CYCLE NUMBER)
APPENDIX I. CCCC Version III Standard Interface Files. RZFLUX (Contd.)

C----------------------------- FLUX VALUES (2D RECORD) -----------------------------
C
C  ((ZGF(K,M), K=1,NGROUP), M=1,NZONE)
C
CM  NGROUP=NZONE=NUMBER OF WORDS
C
CD  ZGF(K,M)  REGULAR ZONE FLUX BY GROUP, AVERAGED OVER ZONE
CD  NEUTRONS/SEC*CM**2
C

CEO
APPENDIX 3. ARCSP016 - THE ARC SYSTEM CATALOGUED PROCEDURE FOR GENERATING A FIXED SOURCE FOR AN INHOMOGENEOUS CALCULATION OF A LOW-REACTIVITY-PLANF SYNTHESIS EXPANSION FUNCTION.

//ARCSP016 PROC COMPSYS1='XXSCHIN1', COMPSYS2='XXSCHIN2',
// CXSTSP='(NPW,PASS)' , CXSTVCL=SCR001, CXSBLK1=1028, CXSBLK2=6116,
// DMNPST=F,
// FLUX2D=NULLFILE, LUX1D=NULLFILE, FLUXVCL=,
// HAINDX=6136,
// XCYXS1=NULLFILE, XCYXS2=NULLFILE,
// MCROVL=,
// PATH=STD016,
// PRPLTH='SYS1.DUMMYLIF',
// QHBOLX=3064,
// SORPS1='EFSD1D', SORPS2='EFSD2D',
// X1111='(600,0)',
// UNITS=BATCHDSK

******************************************************************************
*C CATALOGUED PROCEDURE FOR STANDARD PATH 16 - SETS UP
*C DISTRIBUTED SOURCE FILE CONTAINING, FOR EACH NPIN POINT AND
*C GRPH, THE PRODUCT OF THE FLUX AND LOCAL DIFFUSION COEFFICIENT.
*C
******************************************************************************

SYMBOLIC PARAMETERS

PARAMETER DEFAULT VALUE USAGE P**NM001

** **
*C XCYXS1 XXSCHIN1 DSN OF XS.C.MIN FILE 1 11
*C XCYXS2 XXSCHIN2 DSN OF XS.C.MIN FILE 2 11
*C CXSTSP (NPW,PASS) DISPOSITION OF XS.C.MIN FILE 11
*C CXSTVCL SCR001 VOLUME FOR XS.C.MIN 11
*C CXSBLK1 1024 BLKSIZE FOR XS.C.MIN FILE 1 11
*C CXSBLK2 6116 BLKSIZE FOR XS.C.MIN FILE 2 11
*C FLUX1D NULLFILE DSN OF INPUT FR.D1 11
*C FLUX2D NULLFILE DSN OF INPUT FR.D2 14
*C FLUXVOL ------ VOLUME FOR INPUT FLUX FILE 13, 14
*C HALPBLK 6116 HALP TRACF BLKSIZE
*C XCR11 ------ VOLUME OF XS.ISO FILES 23
*C XCR12 NULLFILE DSN OF XS.ISO FILE 1 23
*C XCR22 NULLFILE DSN OF XS.ISO FILE 2 23
*C PATH STD016 PROGRP NAMF PXFC
*C PRPLTH SYS1.DUMMYLIF ADDITIONAL LIBRARY STEPLTP
*C ORNPOLK 3064 QUARTER TRACF BLKSIZE
*C SORPS1 'EFSD1D' 1D SOURCPF DSN 15
*C SORPS2 'EFSD2D' 2D SOURCPF DSN 16
*C X1111 '(600,0)' STEP 'THE LIMIT PXFC
*C UNITS BATCHDSK DEFAULT UNIT PARAMETERS

******************************************************************************
// GO EXEC PGM=SPATH,TIME=TIMELIB
// STEPLIB DD DSN=SRPLIB,DISP=SHP
// DD DSN=C116.B21006.MODLIB,DISP=SHP
// DD DSN=C116.ARC.MODLIB,DISP=SHP
// TO5P001 DD DDNAME=SYSIN
// TO5P001 DD SYSDUP=A
// SYSDUP DD SYSDUP=GHMDEST
// TO9F001 DD UNIT=SASCR,SPACE=(CYL,(1,1)),
// DCL=(RECPM=VBS,RECCL=X,BLKSIZE=HALFBLK)
// ARC SYSTEM BCD INPUT STREAM
// TO1F001 DD DSN=FECOMPXS1,UNIT=SCRITS,VOL=SPR=6CXSVOL,
// DISP=6CXSDEST,SPACE=(TRK,(1,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=6CXSBLK)
// FILE 1 OP DATA SET VS.C.MIN
// TO1F002 DD DSN=FECOMPXS2,UNIT=SCRITS,VOL=SPR=6CXSVOL,
// DISP=6CXSDEST,SPACE=(CYL,(1,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=6CXSBLK)
// FILE 2 OP COMPOSITION CROSS SECTION SET VS.C.MIN
// TO12F001 DD DSN=FECOMPXS2,UNIT=SCRITS,VOL=SPR=6CXSVOL,
// DISP=6CXSDEST,SPACE=(CYL,(1,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=6CXSBLK)
// GEOMETRY SPECIFICATIONS DATA SET
// TO13F001 DD DSN=FLUXID,UNIT=SCRITS,VOL=SPR=6CXSVOL,
// DISP=(OLD,KEEP)
// 1D REAL FLUX DATA SET
// TO14F001 DD DSN=FLUXID,UNIT=SCRITS,VOL=SPR=6CXSVOL,
// DISP=(OLD,KEEP)
// 2D REAL FLUX DATA SET
// TO15F001 DD DSN=SOR210,UNIT=SCRITS,
// DISP=(NEW,PASS),SPACE=(CYL,(1,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=HALFBLK)
// 1D DISTRIBUTED SOURCE DATA SET
// TO16F001 DD DSN=SOR210,UNIT=SCRITS,
// DISP=(NEW,PASS),SPACE=(CYL,(1,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=HALFBLK)
// 2D DISTRIBUTED SOURCE DATA SET
// TO17F001 DD DSN=ANIP,UNIT=SCRITS,DISP=(NEW,PASS),SPACE=(TRK,(5,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=HALFBLK)
// GEOMETRY UNIFRONS INPUT DATA SET
// TO11F001 DD DSN=FARC,UNIT=SCRITS,DISP=(NEW,PASS),SPACE=(TRK,(1,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=HALFBLK)
// BOUNDARY CONDITION SPECIFICATIONS
// TO1F001 DD DSN=FGRONOC,UNIT=SCRITS,SPACE=(TRK,(5,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=HALFBLK)
// PARAMETAL AND COMPOSITION SPECIFICATIONS
// TO20F001 DD DSN=FSCR001,UNIT=SCRITS,SPACE=(CYL,(1,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=HALFBLK)
// SCRATCH DATA SET FOR NODEL UNCO01
// TO21F001 DD DSN=FSCR001,UNIT=SCRITS,SPACE=(CYL,(1,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=HALFBLK)
// SCRATCH DATA SET FOR NODEL UNCO01
// TO22F001 DD DSN=FSCRMN,UNIT=SCRITS,SPACE=(CYL,(1,1)),
// DCB=(RECPM=VBS,RECCL=X,BLKSIZE=HALFBLK)
// DATA SET VS.F.NIN CONTAINS MATERIALS CROSS SECTIONS
APPNOTX J.

T21002 DD NSN=EMACS

DISP=SH. VOLS=EMACS

DEPENDING

AS IS PER THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR.

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-

THE DATA SET XS.C.AUX

CONTAINS A SHOW-PAK FOR

FILL 20 DATA FOR

R90 "SCAT-

CRATCH DATA FOR

90" "SCAT-
APPENDIX X. STP016 - THE ARC SYSTEM STANDARD PATH FOR GENERATING A FIXED SOURCE FOR AN INHOMOGENEOUS CALCULATION OF A LOW-REACTIVITY-PLANE SYNTHETIC EXPANSION FUNCTION.

PATH DRIVER FOR PREPARATION OF DATA SETS ES.D1D OR ES.D2D

MODULES LNKFD TO BY PATH DRIVER

HU001 GENERAL NEUTRONICS INPUT PROCESSOR
AJC010 PREPARES DATA SETS ES.D1D OR ES.D2D USING DISPERSION COEFFICIENTS FROM DATA SET XS.C.MIN AND REAL FLUXES FROM DATA SETS ES.D1 OR ES.D2

SUBPROGRAMS CALLED BY PATH DRIVER

SYSTEM INITIALIZES THE SYSTEM ROUTINES
PROJ ROUTINE TO PRINT ERROR MESSAGES

FILES REFERENCED BY PATH DRIVER

FILE NAME LOGICAL UNIT NUMBER FPCORD PGNTR
========= =============== =========
GEM TREC

DECLARE VARIABLES TO BE DOUBLE PRECISION USING THE IBM IMPLICIT REAL*CONVENTION

IMPLICIT REAL* (A-H,C-7)
REAL* HU002
DIMENSION DSNAME(22)

INITIALIZE VARIABLES IN DATA STATEMENT
DATA DSNAME / PHYS.C.MIN, HXGEOM, SFR.D1, SFR.D2, PHYS.D1,
1 HXGEOM, 5HA.MIN, 2HSC, 7HA.HOPOR, 6HSC001, 6HSC002,
2 PHYS.C.MIN, PHYS.TSC, PHYS.TSO2, PHYS.C.ANT, PHYS.C.ANY,
3 PHYS.C.ANT, PHYS.M.AUX, 6HSCRAINT, 6HCSCRAINT, XS.C.MIN,
4 HXGEOM / N1002/6HSC002/, AJC010/6HAC010/, STP016/6STP016/,
5 FATAL/5FATAL/, SUBNAV/6HSTP016/, GROM/4HGEOM/, HOMOG/4HUC001/,
6 XHOMOG/6HUR01/ 

INITIALIZE THE SYSTEM

CALL SYSTEM(DSNAME)

DE库MNTRY TO THE DATA SET GROM HAS BEEN PROVIDED

TO=0
APPENDIX K. STP016. CONTINUED.

CALL SNIFF (GEOM, TGPOM, IO)
IF (TGPOM .GT. 0) GO TO 100
C
C <POOL IN THE INPUT STREAM
C
CALL RENDS (STP016, V1)
IF (V1 .GT. 0) GO TO 110
10100 CONTINUE

C******************************************************************************
C FATAL ERROR -10100. NO DATA SET GEOM WAS PROVIDED AND NO INPUT
C DATA WAS SUPPLIED
C******************************************************************************

VEPR=-10100
CALL ERROR (SUENAME, VERR)
CALL ERROR (FATAL, VERR)
110 CONTINUE
C
C LINK TO MODULE NU1002 TO PREPARE DATA SET GEOM
C
CALL LINK (NU1002, NOR)
100 CONTINUE
C
VEPR=-C.C. MIN TO ONE DOES NOT EXIST.
C
CALL SNIPP (DSNAME(1), IXSIDE, IO)
IF (IXSIDE .GT. 0) GO TO 120
IXS=1
IXS=4
CALL LINK (XHONG)
CALL LINK (HONG, ICALL, IO, IO, IO, IXS)
120 CONTINUE
C
C LINK TO MODULE AJC010
C
CALL LINK (AJC010)
RETURN
END
ACKNOWLEDGMENTS

A number of people made significant contributions to the programming of SYN3D and to this report. R. P. Hosteny coded the HMG4C overlay; W. F. Miller was responsible for the planar leakage integral calculations in the overlay INTEG. Contributions to this report were made by R. P. Hosteny and H. Henriksen, II. Users whose feedback uncovered bugs and helped shape the code include C. L. Beck, E. K. Fujita, G. L. Grasseschi, K. R. Krack, M. J. Lineberry, and D. C. Wade. A. R. Hinds made several programming suggestions which resulted in significant improvements in running times.
REFERENCES


