Dr-864

LA-5610-MS
Informal Report

UC-32
Reporting Date: April 1974
Issued: May 1974

XSMOD: A Code for the Modification of the Los Alamos Master Data File

by

Martha S. Hoyt
Frank McGirt

los alamos
scientific laboratory
of the University of California
LOS ALAMOS, NEW MEXICO 87544

UNITED STATES
ATOMIC ENERGY COMMISSION
CONTRACT W-7405-ENG. 36

In the interest of prompt distribution, this LAMS report was not edited by the Technical Information staff.

Printed in the United States of America. Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, Virginia 22151
Price: Printed Copy $5.45 Microfiche $1.45

XSMOD: A CODE FOR THE MODIFICATION OF THE LOS ALAMOS MASTER DATA FILE

by

Martha S. Hoyt and Frank McGirt

ABSTRACT

XSMOD is a CDC 6600 FORTRAN code which allows modifications to group and/or cross-section sets in the Los Alamos Master Data File. The code uses the DATATRAN data manipulation language and is modular in structure. Data sets may be added to or deleted from the library; they may be retrieved from the library and printed or may be written to a tape or disk file in a format acceptable to processing codes. A description of the code, flow diagrams, a code listing and instructions for using the code are included.

I. INTRODUCTION

Over the past several years Roger Lazarus at the Los Alamos Scientific Laboratory (LASL) collected cross-section data from the Evaluated Nuclear Cross-Section Library of the Lawrence Livermore Laboratory (LLL) and from the United Kingdom Atomic Energy Authority Nuclear Data Library (UK) and developed a library of his own on the MANIAC. To make multigroup processing possible he also built up a library of group sets. The combined library of group sets and cross-section data is the Los Alamos Master Data File (LAMDF).

Lazarus and Frank McGirt decided to transfer the cross-section effort to one of the large computers in the CCF. The multigroup processing code EVXS was rewritten for the CDC 6600 by Margaret Asprey.[1] Lazarus' MANIAC code LIB66[2] prepared a library tape for use by EVXS on the 6600. McGirt decided to use the DATATRAN[3] system to create the 6600 library. DATATRAN is a user-oriented data manipulation language developed at the Knolls Atomic Power Laboratory. It is a tool for saving, retrieving and modifying data in a simple manner. The DATATRAN system available at LASL is described in Ref. 4.

The DATATRAN libraries currently available on the 6600 have come from two sources: 1) The original LAMDF was converted to the 6600 DATATRAN version by processing the library tape prepared by LIB66 with Martha Hoyt's XSDTLIB[5] code; 2) A DATATRAN library of the latest LLL data was created on the 6600 by XSMOD.

XSMOD is a 6600 code which provides a means of modifying a DATATRAN library. It can add, delete or print group sets and/or cross-section sets with angular distributions; print out the table of contents of the DATATRAN directory; prepare a library tape of the entire library or selected data sets for input to EVXS; prepare linkage to EVXS if the user wishes to make use of the modular execution feature provided by the DATATRAN system. The capability of changing part of a data set has not been implemented at this time.

II. THE MAIN CODE XSMOD

XSMOD recalls the directory to the DATATRAN library, calls the module RDJOBS to read the job requests and set up the necessary retrievals, acts as a control program to call upon individual modules or subroutines to perform the tasks requested, defines and saves the directory if it was modified, and sets up linkage with EVXS if requested. Flow diagrams for the main code and all modules are in Appendix A. The FORTRAN code listing is in Appendix B.

The modules called and their functions are these:

RDJOBS Read in job requests, set up add, tape and retrieve lists, call RECLIB to scan library tape and transfer requested blocks to ECS.

ADDG Add and/or list a group set.

ADDXS Add a cross-section set.

LISTXS Print a cross-section set.

TAPE Prepare a file TAPE1 in the same format as that produced by LIB66.

Communication with modules is accomplished by means of DATATRAN N.lists, which are defined in Ref. 4.

XSMOD also calls the following subroutines:

DELETE Delete a group or cross-section set from the DATATRAN library directory.

CONTENT Print contents of DATATRAN directory.

CHANGE Print message and return to main code. This is a dummy subroutine which can be replaced by a module if the CHANGE keyword is implemented.

Communication with the subroutines is standard, i.e., by arguments and common.

The subroutine DELETE modifies the directory by replacing with zeroes all directory information associated with the group identification number (GID) or cross-section identification number (SID) being deleted. The directory structure is described in detail in the section MODULE RDJOBS. DELETE modifies only the directory since removal of the data set is not necessary. To remove the data set from the library tape a separate DTPURGE job must be done. That job is described in Ref. 4, and the information needed as input for it is printed out by the subroutine DELETE. When all the DELETE's requested on a single card have been done, the directory is closed up and the item count is modified. The current directories are defined in N.lists to be available to all modules, and the directory save flag is set so that a new directory will be saved at the end of the job.

The subroutine CONTENT prints out the list of GID's and SID's in the directory at the time the CONTENT card is processed. It also prints the directory recall number and the save flag which indicates whether or not the recall number is valid.

III. MODULE RDJOBS

The option cards, which communicate to the main code the jobs which the user wants to do, are read by the module RDJOBS. The implemented keywords and the jobs they do are as follows:

ADD SID Add specified cross-section sets to the DATATRAN library and print.

ADD GID Add specified group sets to the DATATRAN library and print.

DELETE SID Delete specified cross-section sets from the DATATRAN library directory.

DELETE GID Delete specified group sets from the DATATRAN library directory.

LIST SID Print specified cross-section sets from the DATATRAN library.

LIST GID Print specified group sets from the DATATRAN library.

TAPE ALL Write the entire DATATRAN library to TAPE1.

TAPE SID Write specified cross-section sets to TAPE1.

TAPE GID Write specified group sets to TAPE1.

CONTENTS Print table of contents of the DATATRAN library.

EVXS Link with EVXS code.

LAST Last option card.

These cards are read by format (A10, 2X, 10I6). The keyword must start in column 1. Up to 10 GID's or SID's may be specified on a single card. The first ID specified must end in column 18. Remaining ID's must occupy the rightmost portion of successive 6-column fields.

RDJOBS defines as FORTRAN variables the DATATRAN library directories for the GID's and SID's in the library. The entire directory D.DIRECT consists of seven records which contain the following information:

Record 1	N.DIRGID.1/	GID's in the library
2	N.DIRGID.2/	DATATRAN library recall numbers
3	N.DIRGID.3/	Numbers of words for GID sets
4	N.DIRSID.1/	SID's in the library
5	N.DIRSID.2/	DATATRAN library recall numbers
6	N.DIRSID.3/	Numbers of words for SID sets

7 N.IID Recall number of the directory

The GID's and SID's are simple integers in the directory. They appear in the order in which they were added to the library. The recall numbers in Records 2 and 5 are ordered so that they correspond to the GID's and SID's in Records 1 and 4, e.g., the nth word of Record 2 is the recall number of the nth GID in Record 1. Records 3 and 6 contain the numbers of words in the blocks saved for given GID's and SID's. These also are ordered to correspond to Records 1 and 4. For retrieval of the information associated with a given GID or SID, the recall number and number of words in the block are essential. When a job requiring retrieval of a GID or SID is requested, the directory is searched for the given GID or SID and the corresponding recall number and number of words are picked up. This information is passed to the module RECLIB[6] which scans the DATATRAN library tape once and puts into ECS all information which will be recalled in the job. The actual recall of the information into core is done by constructing a particular recall card image in various modules at the time the information is needed.

RDJOBS calls the subroutine READOPT to read in the option cards. An entire card, i.e., keyword, specified GID's or SID's and blank fields, is read into the array KDIMAG and printed. The keyword is compared with all allowed keywords until a match is found. By means of a computed GO TO, control passes to a part of the code which makes lists, sets flags, and sets indices for reading the next card. The keywords and the processing which takes place at this point are given below:

ADD SID	Add SID's from this card to the list IADDX and increase the count of items in the list.
ADD GID	Add GID's from this card to the list IADDG and increase the count of items in the list.
DELETE SID	Increase indices to read the next card.
DELETE GID	Increase indices to read the next card.
LIST SID	Call subroutine TEMPLST to add SID's from this card to a temporary list for retrieval.
LIST GID	Call subroutine TEMPLST to add GID's from this card to a temporary list for retrieval.
TAPE ALL	Set flag IALL = 1.
TAPE SID	Add SID's from this card to the list ITAPX and increase the count of items in the list.
TAPE GID	Add GID's from this card to the list ITAPG and increase the count of items in the list.
CONTENTS	Increase indices to read the next card.
LAST	Indicates all option cards have been read. Process lists.

Encountering the LAST card ends reading of option cards. If a TAPE ALL has been requested, the list IADDG is put in increasing numerical order, the list ITAPG is filled from the GID directory and put in increasing numerical order. The list IADDG is merged into ITAPG with the GID's from IADDG flagged minus. The flagged GID's are not retrieved from the library in the module TAPE but are read from a diskfile. The lists ITAPX and IADDX are processed in the same way for the SID's. If a TAPE ALL was not requested, the tape lists ITAPG and ITAPX are put in increasing numerical order and compared with the corresponding add lists IADDG and IADDX. GID's and SID's which appear on both lists are flagged minus in the tape lists. The tape lists contain all GID's and SID's to be written to TAPE1; the add lists contain all GID's and SID's to be added to the library during this job. If tape and/or add lists are empty, their processing is skipped. READOPT returns to RDJOBS IX and IG, the number of SID's and number of GID's to be written to TAPE1.

RDJOBS now prepares the input lists for module RECLIB. Three lists are required:

N.REEL = *MHX1*	Identifier for the library tape
N.RECLID	List of recall numbers and associated numbers of words in pairs of all information on the library tape to which access is allowed, i.e., all GID's and SID's in the directory
N.USE	Recall numbers of all GID's and SID's to be retrieved in this job

To prepare the list N.USE, RDJOBS calls subroutine USELST or entry USEALL if the entire library is to be retrieved. USELST removes any duplications from the GID and SID lists made in TEMPLST. It searches the directory for the GID's and SID's to be recalled and puts the corresponding recall number into an array to be returned to RDJOBS. It keeps count of the GID's and SID's to be retrieved and finally puts the recall numbers in increasing order. N.USE can then be constructed from the recall number arrays and counts returned. The module RECLIB transfers the information to be retrieved in this job from the library tape to ECS from where it is retrieved as needed.

RDJOBS then defines the DATATRAN list N.RDJOUT to return the following information to the main code:

N.RDJOUT.1/ Entire option card array XKDIMAG
N.RDJOUT.2/ N(1) = number of SID in N.USE, N(2) = number of GID in N.USE, IALL
N.RDJOUT.3/ GID tape list ITAPG
N.RDJOUT.4/ SID tape list ITAPX

IV. MODULE ADDXS

ADDXS adds a cross section to the DATATRAN library from cards or tape. The input must be in the UK or LLL format. The format used by LLL is basically the same as UK. The differences are these: 1) For reactions in which more than one neutron come out, UK gives separate angular distribution and secondary energy data for each neutron. LLL gives only one angular distribution and one secondary energy distribution to be applied to all neutrons. The LAMDF stores only one angular distribution for each reaction. All of the secondary energy data given is stored. 2) Cross-section data from UK is given for the same energy mesh for all reactions. Data from LLL can be on a different mesh for each reaction. Original LLL tapes must be pre-processed by Frank McGirt's LRLUK[7] code which makes a common energy mesh for all reactions. The output of LRLUK is the input to ADDXS.

For each nuclide to be added to the library, the following two cards are required in the input stream:

Card 1: SID No., SOURCE, CTIND, NIN
Card 2: 80 column descriptive message

The format for Card 1 is (I6, 2A6, I6); the format for Card 2 is (8A10). The items or Card 1 are defined as follows:

SID No. The identifier number of this nuclide in the library
SOURCE "LLL" or "UK"
CTIND Indicator for card or tape input. "CARDS" or "TAPE"
NIN Nuclide identification number to be searched for on input tape or to be checked on input cards.

```
Example:
                        1    1    2
col.     123456         2    8    4
Card 1   ^^^521^^^LLL^^TAPE^^^521
Card 2   LLL 521 S32 FROM TAPE 59463 4/16/73 by MSH
```

The information from Card 1 is used for checking and for setting flags in the code. If the SID to be added is already in the library, a message is printed, and the code goes on to the next job. If SOURCE is UK, the first file on the tape is a directory to the tape. It is read and printed. The source flag is set for future use. The tape is searched for NIN. If NIN is not found, a message is printed, and the code goes on to the next job.

The format for the input data is described in a widely circulated draft by K. Parker[8]. The data for each nuclide is divided into blocks. The first block, Block 0, is a table of contents for the remaining blocks. It tells what kind of information (cross-section, angular distribution, secondary energy or fission) is given for each reaction, what block number contains that information and the number of cards in each block. The identifier numbers for the different kinds of data are:

1000 + reaction number for cross-section data
2000 + reaction number for angular distributions
3000 + reaction number for secondary energies
4000 + reaction number for energy-nu pairs.

The definitions of the reaction numbers for UK data are given in the document mentioned above. Definitions used by LLL vary slightly from the UK and are given in Ref. 9.

The kind of data contained in a given block is determined from the table in Block 0. Control then passes to the appropriate subroutine, READXS, READAD, READSEC, or READNU, to read in the card images, make appropriate checks on the data, and define DATATRAN arrays to save the data in the library. When all the blocks have been read, the DATATRAN cross-section control record is defined. This contains the following information:

ISID SID number in LAMDF
ND($\leq$25) Number of angular distributions
ITIDS(N),N=1,ND ID's of angular distributions in order in which they were encountered on input tape

SDES (8 words)	Description from Card 2
IR(≤50)	Number of reactions
ITIDR(I),I=1,IR	ID of the angular distribution to be used with the I'th reaction (-7 if no angular distribution)
NES(≤4000)	Number of energies in the cross-section mesh
ES(I),I=1,NES	Energies
IDR(I),I=1,IR	ID's of reactions
ME(I),I=1,IR	In I'th word, lower and upper indices to the energy block for the I'th reaction
IRS(I),I=1,IR	Number of secondary energy items for I'th reaction
QR(I),I=1,IR	Q values
IZ	Z-number
MIXT	Unused indicator for mixture or element
K2	Total number of words in secondary energy blocks for all reactions
LLL	Source (0 for UK; 1 for LLL)

The DATATRAN array for this nuclide is defined and saved on the library tape. It contains the following records:

N.CNTRL	Cross-section control record described above
N.XS(I)	One cross-section record for each reaction I
N.ESJ(J)	One secondary energy record for each reaction J that has secondary energies. Energy-nu data is included in this block.
N.ANGD(K)	One angular distribution record for each angular distribution (K) given.

When the DATATRAN save is made, the card needed to recall that array is punched out as a receipt. It contains the array name, the number of words in the block on the library tape, and the recall ID number. The physical card is not needed for a later retrieve job since XSMØD is able to construct the card image from information in the directory. However, the information on that card must be used to update the directory now. The punch file is rewound and the name, number of words and recall ID number are read from it and stored in the directory. The count of items in the directory is increased, and the directory save flag is set equal to the ID number. The new directory is defined in a DATATRAN N.array so that it can be returned to the main code, where the latest directory is saved at the end of the job. The directory is not saved now because in a single job the directory may be updated several times. The directory save flag is also returned. When the latest directory is saved, its recall ID equals the value of the directory save flag plus one.

If the nuclide being added is requested on a TAPE1 file in this same job for later processing by EVXS or for input to other libraries, it is written to the diskfile TAPE6 now. The TAPE1 file must be written after all other options have been processed because its format is rigid. All requested groups must appear in ascending order in the first file. The SID files must also be in ascending numerical order. Writing to the temporary file TAPE6 is more efficient than retrieving since for retrieval it would be necessary for RECLIB to make a pass through the entire modified library.

The data for the added nuclide is listed by calling the module LISTXS.

The subroutine PRCT is called every time a card is read. It checks the sequencing information (nuclide ID number, block number, and card number) which appears in columns 73-80 of every card image. If a sequence error is found, a message is printed, and a flag set so that control goes to the error entry at 1000.

Whenever any error is detected, the code transfers to 1000 where IERR is set to the number of cards already read plus one. The code then skips the remaining cards for this nuclide and looks for the next job.

All of the SID's requested on a single ADD SID card are added before leaving the module ADDXS. For the standard return at 2000, the data tape, TAPE2, is rewound and the directory save flag is defined as an N.name to be returned to the main code.

A. Subroutine READXS

READXS reads cross-section data (identifier number 1000 + reaction number) and defines it in DATATRAN lists to be saved in the library. The data is checked for several kinds of discrepancies, and if an error is suspected, the nuclide is not added to the library.

The card image format is described in detail in the draft by K. Parker, which was referenced earlier. The items which are checked by the code are:

1) The sequence information on each card read.
2) The reaction type number. This is compared to what is expected from the table of contents.
3) The interpolation indicator. If interpolation is not log-log, a warning message is printed, and reading continues.
4) The number of cross sections. This must not be greater than the code dimension for the energy array.
5) Temperature. The code is not designed to treat temperature dependence.

One requirement of the UK format and of the present code is that the cross sections for all reactions of a given nuclide must be on the same energy mesh. For any single reaction the cross sections may start or end at any energy contained in the mesh. Thus only one energy array is set up and stored. An index array ME must be formed to contain the lower and upper indices to the energy mesh. For each reaction, then, there is a word in ME which contains the lower index in the leftmost 36 bits and the upper index in the rightmost portion of the word.

Elastic scattering data is required for all nuclides entered in the LAMDF and occurs first on the input tape. Since its energy mesh is complete, the energy mesh for the nuclide is set up when the data for elastic scattering (1002) is read. The lowest energy is set equal to 1.0×10^{-20}. Then the energies and corresponding cross sections are read in. An energy of 1000 is stored at the high end. The energy array now contains two more energies than the data cards contained. The cross section corresponding to the energy 1.0×10^{-20} is set equal to the first cross section read in, and the cross section corresponding to energy 1000 is set to the last cross section read in. The word for the ME array is constructed, and the DATATRAN list of cross sections for this reaction is defined.

For other reactions the energies must be checked against the master mesh. The first energy read is compared with all energies in the master mesh until a match is found. A cross section of 1.0×10^{-20} is assigned to correspond to the energy preceding the matched energy, and the energy index is saved to be used in constructing the index word in the ME array later. If the energy read in matches the second energy in the master mesh (The lowest energy in the master mesh is the dummy 1.0×10^{-20}.), the cross section assigned to the lowest energy is the same as the cross section for the second energy. All succeeding energies read in are compared with those in the master mesh. If they do not agree, a message is printed and the job is dropped. When all cards in the block have been read, the cross section at the upper end is set to 1.0×10^{-20} unless the energy corresponding to the last cross section read is the second highest (Highest was set to 1000.). In that case the cross section corresponding to energy = 1000 is set to that for the highest energy read. The word for the ME array is constructed, and the DATATRAN list of cross sections for this reaction is defined. Control returns to ADDXS.

B. Subroutine READAD

The angular distribution for any given reaction is assumed to be isotropic unless a special angular distribution is given. For both isotropic and special angular distributions the center of mass system is used to describe the data for elastic and discrete inelastic reactions, and the laboratory system is used to describe data for all other reactions. A special angular distribution is always given for elastic scattering.

READAD reads angular distribution data (identifier number 2000 + reaction number) and defines it in DATATRAN lists to be saved in the library. All cards read are sequence checked. When the first card is read, the reaction type is compared to that expected from Block 0, and the code checks to be sure the first angular distribution given is for elastic scattering. The system flag is checked to determine whether it follows the convention laboratory system for all reactions except elastic and discrete inelastic. If it does not, a warning message is printed. If the input is from LLL, the flag is corrected, because we have been assured by R. J. Howerton that the data follows the convention, but the flag is sometimes in error.

The identification number (TID) for the given angular distribution is formed. If the reaction is elastic, the TID is the same as the SID. For any other reaction the TID is formed by multiplying the SID by 100 and adding the two-digit reaction number.

The code checks the flag IDATA to choose the appropriate read: Mu-T pairs or Legendre coefficient representation. Most of the angular distributions currently in the library are in the form of

Mu-T pairs. A combination of the two representations is also possible but occurs so rarely that it was not considered worthwhile to code for that case.

Mu-T Pairs. Angular distributions are given for several different energies. The energies are stored in a separate array E, and the pairs are stored in the array TK. For each energy in the E array there is a word in the index array MD which contains the lower and upper indices to the words of the TK block which correspond to the given energy. The lower index is contained in the leftmost 36 bits, and the upper index in the rightmost 24. The pairs in the TK block are cosine of scattering angle (Mu) and differential scattering probability (T). If the lowest energy given is greater than 2.5×10^{-8}, the lowest energy in the array E is set to 1.0×10^{-11} and an isotropic distribution for that energy is entered into the TK block. After all the pairs have been read in, the integral over the cosine of the scattering angle is checked by trapezoidal rule to make sure the data has been normalized to 1. If it has not been, the normalization is performed now. The DATATRAN list for the angular distribution for this reaction is then defined as follows:

ITIDS (ND)	TID number for this reaction
SDES (8 words)	Description of cross section for heading
NED($\leq$400)	Number of energies for which angular distributions are given
E(J),J=1,NED	Energies in Mev
ISYS	Flag: 1= center of mass; 2 = lab system
A	Atomic weight
LEGN	Number of Legendre coefficients
KT ($\leq$4000)	Number of words in TK block
TK(J),J=1,KT	Mu-T pairs or $NF,F_1,F_2,...F_{NF}$
MD(J),J=1,NED	In J'th word, lower and upper indices to the part of the TK block which corresponds to the J'th energy
NINC	Number of sub-divisions to use in integrating. Set=40.

Control returns to ADDXS.

The UK format allows multiple angular distributions for each outcoming neutron. The LAMDF accepts data for the first neutron only and skips the rest. It is assumed that the data for the first neutron applies to all neutrons. Multiple angular distributions for one neutron, though seldom encountered, can be read and combined in the subroutine MULTI to make a single distribution which is then saved in the library.

Legendre Polynomial Representation. Legendre representation differs from Mu-T pairs only in the contents of the TK block. The E and MD arrays are formed in the same way for both representations. For the Legendre representation the TK block contains for each energy the number of coefficients NF followed by the coefficients.

On the input tape for each L up to LMAX, energies and coefficients are given in pairs. The data is rearranged in the library so that for each energy the coefficients for all L's are tabulated. To do this all the energy-coefficient pairs are read into a temporary storage block. All of the different energies are then picked out and stored in the E block. Then for each energy the coefficients are found and stored in the TK block. A count of the coefficients, NF, is kept. If the coefficients for the highest L's are zero, they are removed from the TK block and NF is corrected.

C. Subroutine MULTI

This subroutine reads several angular distributions for each neutron and combines them into one distribution which is stored in the library. The first distribution is read into the residual arrays XMU and T. The T values are multiplied by the probability for this distribution. The next distribution is read into the temporary arrays XMU1 and T1. The T1 values are multiplied by their probability. As each card is read into the temporary arrays, its Mu's are compared with those already in the MXU array. If a Mu exists in the XMU array but not in the XMU1 array, it is inserted into the XMU1 array and a corresponding T1 value is inserted by linear interpolation. If a Mu exists in the XMU1 array but not in the MXU array, it is inserted into the XMU array and a corresponding T value is inserted by linear interpolation. When all the cards have been read in for this distribution, the values in the T and T1 arrays are added and stored in T. Then the next distribution is read into XMU1 and T1 and combined as above. When all the distributions have been read in and combined, the final distribution exists in XMU and T. It is transferred to the TK block, and KT is set to point at the current last word of the TK block. Control returns to READAD.

D. Subroutine READSEC

READSEC reads secondary energy data (identifier number 3000 + reaction number) and defines it in DATATRAN lists to be saved in the library. The usual data block and card sequence checks are made.

Secondary energy data is given so that the distributions may be calculated according to specific laws. The laws are nine in number, 1, 2, 3, 4, 5, 6, 7, 8, 10. Laws 1-7 and 10 are described in Ref. 8 for UK libraries. Ref. 9 describes laws 3, 8 and 10 as used in the LLL libraries. Briefly the laws are as follows:

1. Neutrons coming out with a known energy.
2. Energy of emitted neutrons dependent on discrete energy loss and reduction factor.
3. Continuous normalized spectrum independent of initial energy.
4. 5. 6. Normalized probability function; $(E^1/E^q) = f(E, E^1/E^q)$, where q = 0 gives law 4, q = 1/2 gives law 5, and q = 1 gives law 6.
7. Four constants to define a normalized probability function to represent the fission spectrum.
8. Secondary neutron production cross sections as a function of E, E' with all secondaries covered.
10. Simple evaporation spectrum specified by a single parameter.

For each energy range one or more laws may apply and a weight is given with each law. The sum of these weights must equal the number of outcoming neutrons. In UK format each neutron may be represented by one or more laws. For a given energy the sum of all of the weights for all of the laws equals the number of outcoming neutrons. LLL does not tabulate data for individual neutrons, but gives one set of data with weights summing to 1 to apply to all neutrons.

The conventions of the LAMDF are such that special handling is necessary in entering LLL data. Law 3 data is accepted as is since LLL uses it to describe fission reactions and thus the number of neutrons out is a special case. Law 8 is reformatted to look like Law 3 to the processing codes. Since the one set of data represents all secondaries, the probabilities are divided by the number of neutrons out. The weight is multiplied by that same number since now the probabilities can be thought of as applying to only one neutron. For Law 10 the weight is multiplied by the number of neutrons out but the data is left as is since it applies to only one neutron.

The data is read into the ESJ block. When reading for a law begins, J points to the first unused word is ESJ. All numbers in the block are floating point; all positive numbers are reaction identifiers or data and all negative numbers are markers. The significance of the markers is as follows:

- 4.0 denotes end of data for a given law
- 5.0 denotes end of data for a given neutron
- 2.0 denotes end of data for a given energy
- 1.0 denotes end of data for a given reaction.

The end of neutron marker, - 5.0, is not used in the straightforward way implied above. Since LLL data does not provide for individual neutrons, a - 5.0 is always followed by a - 2.0 when it is entered into the LAMDF. The processing code EVXS expects this combination. Therefore something special must be done to make UK data compatible with the above restriction. The - 5.0 marker is not used to denote end of neutron but is paired with the -2.0 for end of energy. Data for individual neutrons is separated only by the end of law marker, - 4.0. The sum of the weights equals the number of outcoming neutrons as required. In some cases the same law may be represented more than once since originally it was meant to describe different neutrons, but this causes no problem in EVXS. Thus the same storage configuration in the ESJ block can arise from having several laws describe one neutron or from having two or more neutrons. The sum of the weights tells which case it is.

The ESJ block contains the following information:

NRT	Reaction number in floating point, e.g., 16.0
E_i	Lower bound of the energy range throughout which the following laws apply. The ranges are in order of increasing energy.
W_k	Weight associated with following law number.
Law_k	Law number
Data	Required data for the law given above
.	
.	
.	
-4.0	End of law marker

W_{k+1}

Law_{K+1}

Data

.

.

.

-4.0 End of law

-5.0 End of neutron

-2.0 End of energy

E_{i+1} Next energy

W_k

Law_k

Data

.

.

.

-4.0

-5.0

-2.0

-1.0 End of reaction

For fission some additional information may appear at the end of this block. It is described in SUBROUTINE READNU. The total number of words in the ESJ block, including the markers, is equal to J and is saved in IRS. If no extra fission data follows, the ESJ block for this reaction is defined in a DATATRAN list, and control returns to ADDXS.

Secondary energy distributions are supposedly normalized. For laws 1 and 2 the code compares the sum of the probabilities to 1. For laws 3, 4, 5 and 6 the code uses trapezoidal rule to evaluate the integral over the probabilities and compares the result to 1. If the sum or integral differs significantly from 1, a warning message is printed. In the interests of preserving original data, no normalization is performed in the LAMDF. If the user expects normalized data for his code, he must do the normalization himself.

E. Subroutine READNU

READNU reads energy-nu pairs for the fission reactions 4018 and 4019 into the ESJ block. A 4000 block never appears alone but follows some secondary energy (3000) block. Thus an ESJ block will already have been started. READNU gets the ESJ pointer by adding 1 to IRS(IR) which contains the number of items already in the ESJ block. The first quantity stored in ESJ by READNU is the floating point reaction number 4018.0 or 4019.0. The energy-nu pairs are then read in and the marker -1.0 is stored at the end. The corrected number of items in the ESJ block is stored in IRS(IR); the DATATRAN list is defined, and control returns to ADDXS.

V. MODULE LISTXS

LISTXS lists a cross-section set from the library. When a cross-section set is added to the library, it is listed automatically. As part of the ADD process, the data exists as DATATRAN lists in ECS. The code simply defines the FORTRAN variables from these lists and proceeds to set up the print. A cross-section set already in the library may be listed by requesting it on a LIST SID card. It is retrieved from the DATATRAN library and printed.

The print format is essentially the same as that developed by M. Asprey for the EVLIST[10] code. It gives the following summary information for each reaction:

Reaction index
Reaction identifier
Angular distribution identifier
Lower and upper indices to the energy block
Q value
Reaction title
Length of secondary energy block

The first seven reactions are set up for printing since eight (one for energy) is the maximum number of columns that fit on a page. Secondary energy data and markers in the ESJ block are printed for any of the first seven reactions for which an ESJ block exists. This is followed by the energy mesh and cross sections for the first seven reactions. Secondary energy data and cross sections are then printed for the remaining reactions. This is followed by data for each special angular distribution given. The following information is printed:

Energy index
Energy
Lower and upper indices to the TK block
TK block (Mu-T pairs or Legendre coefficients).

The cross-section print is set up in the SS block which holds up to 2000 cross sections for up to seven reactions. The block may be thought of as seven strips, each of which is 2000 words long. The number of strips and the maximum length needed are cleared. The cross sections for a reaction are brought into the array SIG which is large enough to hold all the data. They are then transferred into the first unused strip of SS starting at the I'th word of the strip, where I is the index to the energy to which the first cross section corresponds. The strip is then filled up to the 2000th word. The count of strips used is kept. If data exists beyond the 2000th word, the strip just filled is written to ECS via DATATRAN statements and the code continues to transfer data from SIG into the same strip starting at the first word of the strip. When all data have been transferred from SIG, the second filling of the strip is written out to ECS and the first filling is restored. When all seven strips have been filled or when all reactions have been set up in SS, subroutine PRSIG is called to print SS. If a given nuclide has more than seven reactions, the SS block is cleared and set up to print the remaining ones.

The angular distributions are defined as FORTRAN variables from the DATATRAN lists and printed one at a time. If this listing is part of an ADD job, control returns to module ADDXS. If the listing was requested on a LIST SID card, the code checks for another request. When all requests have been filled, control returns to the main code.

At the present time there are three libraries from which listings can be made: UK, LLL, and RBL, which is the library which existed on the MANIAC. The libraries do not agree on their definitions of all reaction numbers. Table I shows the reaction numbers and the definitions for the three libraries. The TYPE, which is the number of outcoming neutrons, is given as it exists in the code. Each nuclide in the DATATRAN library is flagged to indicate which library it belongs to: UK, flag=0; LLL, flag=1; RBL, flag=2. The standard reaction titles in the code are for UK. Before a listing is made, the flag is checked, and the appropriate titles are changed if the nuclide is from the LLL or RBL library. The library being used is identified as part of the heading.

Subroutine PRESJ is called to print the secondary energy block. After the heading is printed, the code counts data items, which are positive, and prints them. Then it counts and prints the markers, which are negative. The sequence of count and print continues until all words in the block have been printed. Control returns to LISTXS.

TABLE I

REACTION DEFINITIONS

Reaction	Type	UK	LLL[a]	RBL[a]
1	1	Total		
2	1	Elastic		
3	1	Nonelastic	---	
4	1	Total N,N'		
5	1	N,N'1		
6	1	N,N'2		
7	1	N,N'3		
8	1	N,N'4		
9	1	N,N'5		
10	1	N,N'6		
11	1	N,N'7		
12	1	N,N'8		
13	1	N,N'9		
14	1	N,N'10		
15	1	N,N'C		
16	2	N,2N		
17	3	N,3N		
18	0	N,F		
19	0	N,F		
20	1	N,N'F		---
21	2	N,2NF		---
22	1	N,N'A		
23	1	N,N'3A		
24	2	N,2NA		
25	3	N,3NA		
26	2	N,2NISO		
27	0	Absorption (fission + n,γ)		N,N'P
28	1	N,N'P		N,N'Gma
29	1	N,N'2A		N,N'D
30	2	N,2N2A		N,N'He3
31	1	---	N,N'11	---
32	1	---	N,N'12	---
33	1	---	N,N'13	---
34	1	---	N,N'14	---
35	1	---	N,N'15	---
36	1	---	N,N'16	---
37	1	---	N,N'17	---
38	1	---	N,N'18	---
39	1	---	N,N'19	---
40	1	---	N,N'20	---

TABLE I (cont)

Reaction	Type	UK	LLL[a]	RBL[a]
41	1	---	more levels	---
.	1	---	if needed	---
.				
.				
79	1	---		---
80	1	---	---	---
81	1	---	N,PN	---
82	1	---	N,N'D	---
83	1	---	N,N'T	---
84	1	---	N,N'He3	---
85	1[b]	---	N,4N	---
86	0	---	---	---
.	0	---	---	---
.				
.				
100	0	---	---	---
101	0	N,Parab	---	
102	0	N,Gma		
103	0	N,P		
104	0	N,D		
105	0	N,T		
106	0	N,He3		
107	0	N,A		
108	0	N,2A		
109	0	N,PA		
110	0	Destruction	---	---
111	0	---	---	---
.	0	---	---	---
.				
.				
150	0	---	---	---

[a]No entry under LLL or RBL means definition agrees with UK. --- means definition unassigned.

[b]Code changes TYPE to 4 when adding LLL nuclides having reaction 85.

Subroutine PRSIG prints the SS block. A print line contains the energy index, energy, and cross sections for up to seven reactions. If all cross sections for a given energy are zero, that line is omitted. If the number of energies for a given nuclide is greater than 2000, the data for the first 2000 are printed, and the SS block is set up for the remaining energies. This is done by clearing the amount of the SS block that is needed and filling it by strips from the DATATRAN lists that were written on ECS in LISTXS. When the set-up is complete, SS is printed. Control returns to LISTXS.

Subroutine RETREV is similar to the subroutine of the same name included in module TAPE. The difference is that the RETREV in module LISTXS is concerned only with SID's.

VI. MODULE ADDG

The module ADDG does two jobs: it adds a group set to the library from cards; it retrieves a group set already in the library and lists it.

The following cards are required to add a group set to the library:

Card 1	IGID	(I12)	Group identification number
Card 2	GDES	(8A10)	Descriptive message
Card 3	IGG,IRV	(2I12)	Number of groups; reverse numbering flag
Card 4 . . . Card n	EMX(I),I=1,IGG+1	(6F12)	Group energy boundaries in order of increasing energy
Card n + 1	KGRP,T(K),K=1,5	(I12,5F12)	Group number (lowest energy group first); energy-unnormalized flux pairs for this group
Card n + 2 . . . Card n + m	T(K),K=6...	(6F12)	
Card n + m + 1	Repeat for each group cards like Card n + 1 to Card n + m		
. . .			
Blank card at end			

All cards except Card 2 have 6 fields of 12 columns each, and all numbers are entered in the rightmost portion of the field. Columns 73-80 are not checked for sequencing information.

The code compares the group identification number (IGID) with that requested on the ADD GID card. If they are not the same, the cards for this IGID are skipped and the code goes on to the next job. The data cards for groups to be added must be in the same order as the GID's on the ADD GID card(s). Next the code checks whether the IGID to be added is already in the library. If so, a message is printed, the data cards are skipped, and the next job is tried.

The energy boundaries are read into the array EMX. They must be in order of increasing energy. The individual groups may be numbered such that the lowest number (1) denotes the lowest energy group or the highest number (IGG) denotes the lowest energy group. If the latter numbering scheme is desired, the reversing flag IRV on input Card 3 must be nonzero.

The main loop reads and processes one group at a time. The first card for each group contains the group number (checked for consistency with the reversing flag) followed by energy-flux pairs. Additional pairs are on following cards. The pairs are read into a temporary storage T. The energy of the first pair is compared with the energy of the lower boundary of this group, EMX(I). If they are not the same, an error message is printed, and the next job is begun. As each card is read, it is checked by the subroutine LAST to determine whether it is the last card, to make sure the energies are in increasing order, and to make sure the energies do not exceed the upper boundary for this group. If an error is detected, the job is dropped, and the next one is begun. When all cards for one group have been read in, LAST returns KT, the number of items in the T array, and processing begins.

For the special case of a flat flux the only card for the group contains the group number in the first field and -1. in the second field. The code then sets up the T array as follows:

T(1)=EMX(I) lower energy boundary for this group

T(2)=1.0

T(3)=EMX(I+1) upper energy boundary for this group

T(4)=1.0

The number of items in the T array, KT, is set to 4 and processing for this group begins.

The flux is normalized and stored in the PH array. The energies are stored in the EK array. The average energy and average velocity for this group are calculated, and the index word for this group, MG(I), is constructed. The array MG contains one word for each group within the group set. Each of these words contains in the leftmost 36 bits and the rightmost 24 bits the lower and upper indices respectively which point to the parts of the PH and EK arrays which belong to this group. The upper energy boundary for a group is the same as the lower energy boundary for the next group. The boundary energy appears in both groups because the flux can be discontinuous across the boundary. The DATATRAN array for the input unnormalized fluxes for this group is defined now, and reading for the next group begins.

When the energy-flux pairs for all groups have been read in and processed, the DATATRAN list for this GID is defined and saved:

IGID	Group identification number
GDES	Description from Card 2
IGG	Number of groups
MG(I),I=1,IGG	In I'th word lower and upper indices to PH and EK blocks for I'th group
GE(I),I=1,IGG	Average energy for each group
KG	Number of energy-flux pairs for the group set
PH(I),I=1,KG	Normalized fluxes
EK(I),I=1,KG	Energies
VB(I),I=1,IGG	Average velocity for each group
IRV	Reversing flag. If=0, lowest energy group is 1; if≠0, lowest energy group is IGG
N,T(I)	One DATATRAN record of unnormalized input fluxes for each group

The directory is updated and returned to the main job along with the directory save flag in the same way as in module ADDXS. If the group set being added is requested on a TAPE1 file, it is written to TAPE7 now. Again this procedure is the same as that in module ADDXS.

Next the data is listed. The following quantities are printed for each group:

group number as indicated by reversing flag

average energy

average velocity

energy-flux pair number

energy

normalized flux

unnormalized input flux

flux per Mev

indices to the PH and EK arrays

The print format is similar to that used by M. Asprey in her EVLIST code.

If there is a request to list a GID already in the library, it is retrieved by the subroutine RETREV, the FORTRAN variables are defined from the DATATRAN lists, and the above quantities are printed. Libraries which originated on the MANIAC do not have available the average velocity and the unnormalized input flux. Zeroes are printed for these unavailable quantities. These libraries also do not pass along the reversing flag which indicates the input group numbering scheme. The chosen numbering is highest group number for lowest energy group since this convention is used in many codes.

Control returns to the main code when all GID's on a LIST GID card have been listed or when all GID's on an ADD GID card have been added and listed.

VII. MODULE TAPE

The module TAPE prepares a file, TAPE1 on disk or magnetic tape to be used as input to EVXS or to be processed by TD-6 for their cross-section library. All GID's and SID's in the DATATRAN library can be written to the file by using the TAPE ALL card. The file includes all data added to the library during this run. Selected GID's and SID's may be written by using the TAPE GID and TAPE SID cards. GID's appear on TAPE1 in ascending numerical order. They are all included as individual records of one file--the first file on the tape. The SID's also are written in ascending numerical order. Each SID is in a separate file. A double end of file denotes the end of information on TAPE1.

A tape identifier message is required in the input stream. It is read by format 8A10 into an array of 100 words. Thus it is possible to have a message 12 cards long. For shorter messages a blank card is used to stop the read. The message is written at the beginning of TAPE1 and is in the same file as the GID's. If no GID's are requested on TAPE1, the first file contains only the identifier message.

The array ITAPG is a list in ascending absolute value numerical order of the GID's to be written to TAPE1. If the GID was added during this run, it resides on the disk file TAPE7 and is read from there. A negative flag in the ITAPG list indicates this fact. Any GID which is not flagged is retrieved from the DATATRAN library. Each GID requested is brought into core and written out to TAPE1. An end of file is written after all GID's in the list have been written.

The SID's requested on TAPE1 are in the list ITAPX. This list also is in ascending absolute value numerical order with minus flags on the SID's that were added during this run. The added SID's are on the disk file TAPE6. For a given SID the control record is brought into core and written out to TAPE1. Then the cross-section records are brought in and written out one at a time. Finally the angular distribution records are brought in and written out one at a time. When these are finished, an end of file is written on TAPE1, and the next SID is processed. When all of the SID's in the list have been written, another end of file is written on TAPE1. Control returns to the main code.

The subroutine RETREV constructs the recall card image and initiates the retrieval of a particular KID (GID or SID requested). It searches the appropriate directory for the KID and picks up the recall identification number associated with it. It chooses the correct format to encode the DATATRAN name required for the recall card image. When control reaches the recall card image, the retrieve is made. This means that the DATATRAN lists for the requested KID are in ECS waiting to be defined as FORTRAN variables in small core. Control returns to TAPE.

A detailed description of the TAPE1 file is in Ref. 1. Briefly the file looks like this.

File 1 <u>Record 1</u>:

TAD (100)	Tape description
NS	No. of cross sections

File 1 <u>Record 2</u>:

IGID	Group set ID number
GDES (8)	Group description
IGG	Number of energies
MG(I),I=1,IGG	In I'th word, lower and upper indices to PH and EK values associated with the I'th group
GE(I),I=1,IGG	Weighted energy for each group
KG	Number of energy flux pairs
PH(K),K=1,KG	Normalized flux values
EK(K),K=1,KG	Energy values in Mev

File 1 Record 3: (and following)

For each group there is a record like Record 2.

.

.

.

End of file

File 2 Record 1:

ISID	Cross-section set ID number
ND	Number of angular distributions
ITIDS(N),N=1,ND	ID's of angular distributions in order on tape
SDES (8)	Cross-section description
IR	Number of reactions
ITIDR(I),I=1,IR	ID's of angular distribution to be used with I'th reaction (-7 if no angular distribution)
NES	Number of energies
ES(J),J=1,NES	Energies
IDR(I),I=1,IR	Reaction ID's
ME(I),I=1,IR	In I'th word lower and upper indices to the energy block for the I'th reaction
IRS(I),I=1,IR	Number of secondary energy items for I'th reaction
QR(I),I=1,IR	Q values
IZ	Z number
MIXT	Unused indicator for mixture or element
K2	Total number of words in secondary energy blocks for all reactions
LLL	Source (0 for UK; 1 for LLL; 2 for RBL)

File 2 Record 2:

SIG(J)	Cross sections for one reaction
ESJ(I)	Secondary energy data for this reaction if given

File 2 Record 3: (and following)

For each reaction there is a record like Record 2. There is a total of IR records, the last of which is numbered IR+1.

.

.

.

File 2 Record IR+2:

ITID	Angular distribution ID
SDES (8)	Cross-section description
NED(≤ 400)	Number of energies for which distributions are given
E(I),I=1,NED	Energies
ISYS	System flag (1 for data in center of mass; 2 for data in laboratory)
A	Atomic weight
LEGN	Number of Legendre coefficients
KT(≤ 4000)	Number of words in TK block
TK(K),K=1,KT	Mu-T pairs or Legendre coefficients
MD(I),I=1,NED	In I'th word, lower and upper indices to the part of the TK block which corresponds to the I'th energy
NINC	Number of sub-divisions to use in integrating

File 2 Record IR+3: (and following)

For each angular distribution there is a record like Record IR+2. There is a total of ND records, the last of which is numbered IR+ND+1

.

.

.

File 2 Record IR+ND+1:

Like Record IR+2

End of File

File 3 (and following)

Like File 2. One file for each SID

.

.

.

End of File (End of tape)

VIII. USING THE CODE

A. Over-all Deck Setup

The deck for running XSMOD is as follows:

Control Card Deck	
. 8-9	
Update Input I	Required only if adding to the
7-8-9	cross-section library. Puts card image from OLDPL=DATA onto COMPILE=TAPE2.
Update Input II	XSMOD code
7-8-9	
Option Deck	Tell XSMOD what jobs to do.
7-8-9	
Data Deck	Required by certain options. See Section C, Data Setup.
EOJ	

B. Control Card Setup

The following control card deck is used to run a typical job:

Control card	Comment
$JOB(NAME=MHØYT,CL=U,AC=XXX,UA=XXXXXXXX,TL=2M,	
$1SC=*156000B,	
$2LC=2000000B,PL=*,MX=66)	
RFL(6000)	
ASSIGN MT,NEWT(PLB,LXXXXL00,SHB)	Contains OLDPL of XSMOD as 1st file, then DATATRAN overlays.
ASSIGN MT,DTLIB(PLB,LXXXXL00,SHB)	DATATRAN library tape, TAPE30
ASSIGN MT,NEWLIB(NLB,LXXXXL00,SHB)	New DATATRAN library tape. Assign only if library is being changed.
ASSIGN MT,TAPE1(NLB,LXXXXL00,SHB)	TAPE1 for EVXS or TD-6 library. Assign only if file is needed on a tape.
ASSIGN MT,DATA(PLB,LXXXXL00,SHB)	OLDPL containing card images of cross-section data to be added to the library. Use only with ADD SID job.
ASSIGN AB,UPDATE2.	
CØPYBF(UPDATE2,UPD)	
RETURN(UPDATE2)	
CØPY(DTLIB,TAPE30)	
RELTAPE(DTLIB)	
REWIND(TAPE30)	
CØPYBF(NEWT,ØLDPL)	OLDPL for XSMOD
CØPYBF(NEWT,NØVAF)	
CØPYBF(NEWT,CLASP)	
CØPYBF(NEWT,DTPATH)	
CØPYBF(NEWT,RDTFILE)	DATATRAN system needed to run a DT job.
CØPYBF(NEWT,SCAN)	
CØPYBF(NEWT,RECALL)	
CØPYBF(NEWT,SØRT)	
CØPYBF(NEWT,INITIAL)	
CØPYBF(NEWT,DTCATLG)	
RELTAPE(NEWT)	

Control card	Comment
ONSWCH1.	
REWIND(RDTFILE)	
REWIND(LGO)	Required for DATATRAN run.
COPYBF(RDTFILE,LGO)	
RETURN(RDTFILE)	
BKSP(LGO)	
REWIND(UPD)	Required only if tape DATA has been assigned.
RFL(60000)	
UPD(D,P=DATA,C=TAPE2,L=A124)	
RELTAPE(DATA)	
REWIND(UPD)	
RFL(60000)	
UPD(P,L=A124)	Code XSMOD
RETURN(ØLDPL)	
RETURN(UPD)	
COPYSBF(COMPILE)	
RFL(156000)	
SCAN(COMPILE)	Pre-compiler: makes DATATRAN statements acceptable to FORTRAN compiler.
RETURN(SCAN)	
RETURN(COMPILE)	
RUN(S,I=DTAPE3)	
RETURN(DTAPE3)	
LØAD(LGO)	
NØCØ.	
ONSWCH4.	
NOVAF.	Starts execution.
REWIND(TAPE30)	
COPY(TAPE30,NEWLIB)	Use only if tape NEWLIB assigned.
EXIT.	
REWIND(TAPE30)	
COPY(TAPE30,NEWLIB	
7-8-9	

C. Data Setup

Certain options require additional information from the input stream. Options and their requirements are shown below. The required data cards are described in detail in the modules which read them:

ADD SID

Card 1: SID No., Source, Ctind, Nin

Card 2: 80 column descriptive message

.

.

.

7-8-9

Cards 1 and 2 are required for each SID punched on the ADD SID card. The last Card 2 must be followed by a 7-8-9.

ADD GID

Card 1: IGID

Card 2: GDES

Card 3: IGG,IRV

Card 4 to Card n: Group energy boundaries

Card(n+1) to Card(n+m): Group number and energy-flux pairs

Blank card

.

.

.

7-8-9

Cards 1 through Blank are required for each GID punched on the ADD GID card. The last set must be followed by a 7-8-9.

TAPE ALL, TAPE SID, TAPE GID

Card 1 to Card n: Tape identifier message. n must be ≤ 12.

Blank card if n < 12

EOJ

No data cards are required for the following options:

DELETE SID, DELETE GID

LIST SID, LIST GID

CONTENTS

EVXS

LAST

In general options are processed in the order in which they appear on the option cards, and the required data must appear in the same order. The only exceptions are the tape options, TAPE ALL, TAPE SID, and TAPE GID. TAPE is always processed last and requires only one tape identifier message since only one tape is written. The tape identifier cards must be last in the data deck regardless of where the TAPE card appears in the option deck.

Example

DELETE SID	551	554		
ADD GID	16			
TAPE GID	35	16		
ADD SID	501	520	536	Option deck
TAPE SID	520			
LIST GID	35			
CONTENTS				
LAST				
7-8-9				

Data for ADD GID 16 (several cards)

7-8-9

Data for ADD SID 501 (2 cards)

Data for ADD SID 520 (2 cards) Data deck

Data for ADD SID 536 (2 cards)

7-8-9

Data for TAPE GID, TAPE SID

EOJ

D. Assigning the Library

The library which the DATATRAN system works on is called TAPE30. It can be read or written on at any time. To make sure that the library on the magnetic tape is not ruined if the job fails during execution, the library on tape, DTLIB, should be copied to a diskfile, TAPE30, and released. If TAPE30 is modified during the job, TAPE30 must be copied to a blank magnetic tape, NEWLIB, after job execution is finished. To gain access to the library on TAPE30, the code recalls the directory to that library. Since a new directory and recall card are made every time the library is modified, the directory recall card and appropriate comment cards are entered into the code by means of an UPDATE *IDENT in every run. The directory recall number is printed at the end of every job. The directory recall card looks like this:

```
Col.                    4        5       7
 7                      4        8       1
 RECALL D.DIRECT  ERASE IN-1-DAY  1533WRD  RMHX1  60
```

The recall number ends in column 71.

If the user wishes to create a new library, he must assign as his DTLIB a tape which contains a null directory. The null directory is recalled in the same way as any other directory. Its recall number is 3. The code checks for an empty directory and re-routes its flow accordingly.

E. Use of IMAGE

The first block on every DATATRAN library tape is the module IMAGE, and XSMOD recalls it at the beginning of every job. It is useful for printing out DATATRAN lists and is called by the following sequence of statements:

N.LIST=name(s) of list(s)

KALL IMAGE

IMAGE must be called only from a main program in a module or the main code. It may not be called within a subroutine.

F. Procedure for LLL Tapes

The following procedure has been used successfully to enter data from LLL tapes into the library with

XSMOD:

(1) Determine the data structure on the LLL tape.

Assign the LLL tape PUS. Buffer in three records, get the length, print in octal. This can be done with Hoyt's DMPLLL.[11]

(2) Prepare an UPDATE OLDPL of the raw data.

After determining the data structure from an examination of the output from (1), buffer in a record at a time and write out card images using 8A10 to a file TAPE2. Hoyt's LLLTAPE,[12] possibly with some variation because of differences in record structure, will do this. Use TAPE2 as input to UPDATE.

(3) Prepare input for XSMOD.

Run McGirt's LRLUK[7] with input from the UPDATE OLDPL made in (2). LRLUK puts all reactions for one nuclide on a common energy mesh, throws out the second representation of the same angular distribution (LLL gives both Mu-T pair and Legendre form), and ignores data for General Classification Numbers above 12. The output is card images on TAPE3. For convenience TAPE3 is used as input to make an UPDATE OLDPL. It is then possible to pick off only the nuclides the user wants to enter into the library without reading the entire tape.

(4) Run XSMOD to enter nuclides into the library.

Use UPDATE to put nuclides to be entered into the library onto COMPILE=TAPE2.

G. Linking with EVXS

It is possible to link directly with EVXS after running XSMOD. The user may wish to do this if he needs to run with EVXS a group or cross-section set not previously in the library. The procedure is as follows:

(1) Request XSMOD to make the file TAPE1 containing all GID's and SID's needed by EVXS. TAPE1 may be a diskfile. Request EVXS processing by using the keyword EVXS in the option deck.

(2) Alter EVXS:

Insert before program card: OVERLAY(EVXS,0,0). Change program card to eliminate file name MADSK. Use file name TAPE1.

(3) Compile EVXS with the above changes, and make the absolute overlay file by using

RUN(S)
LOAD(LGO)
NOGO.

(4) Run XSMOD. It will call EVXS when it is finished. Data in the input stream required by EVXS should be preceded by a 7-8-9 card to separate it from data required by XSMOD. EVXS finishes normally. There is no return to XSMOD.

REFERENCES

1. M. W. Asprey, R. B. Lazarus, and R. E. Seamon, "EVXS: A Code to Generate Multigroup Cross Sections from the Los Alamos Master Data File," Los Alamos Scientific Laboratory report LA-4855 (in preparation).

2. R. B. Lazarus, Los Alamos Scientific Laboratory, LIB66, undocumented code for MANIAC computer, 1968.

3. H. J. Kopp, J. D. Morris, and W. E. Schilling, "DATATRAN Modular Programming System for Digital Computers," KAPL-M-6997, Knolls Atomic Power Laboratory, Schenectady, New York (July 1968).

4. Frank McGirt, "LAMPS, Los Alamos Modular Programming System User's Manual," Los Alamos Scientific Laboratory report LA-4371 (November 1970)

5. Martha Hoyt, Los Alamos Scientific Laboratory, XSDTLIB, undocumented code for CDC 6600 computer, 1970.

6. W. E. Schilling, "DATATRAN Note #6," Knolls Atomic Power Laboratory, Schenectady, New York, (February 24, 1969). Schilling's module RECALL was re-named RECLIB at LASL to avoid a conflict in our version of the DATATRAN system.

7. Frank McGirt, Los Alamos Scientific Laboratory, LRLUK, undocumented code for CDC 6600 computer, 1970.

8. K. Parker, "The Format and Conventions of the U.K.A.E.A. Nuclear Data Library as of June 1965," Atomic Weapons Research Establishment draft, Aldermaston, England (June 1965).

9. R. J. Howerton, et al., "An Integrated System for Production of Neutronics and Photonics Calculational Constants," volume 4, Lawrence Radiation Laboratory report UCRL-50400 Vol. 4 (April 15, 1971).

10. M. W. Asprey, Los Alamos Scientific Laboratory, EVLIST, undocumented code for CDC 6600 computer, 1969.

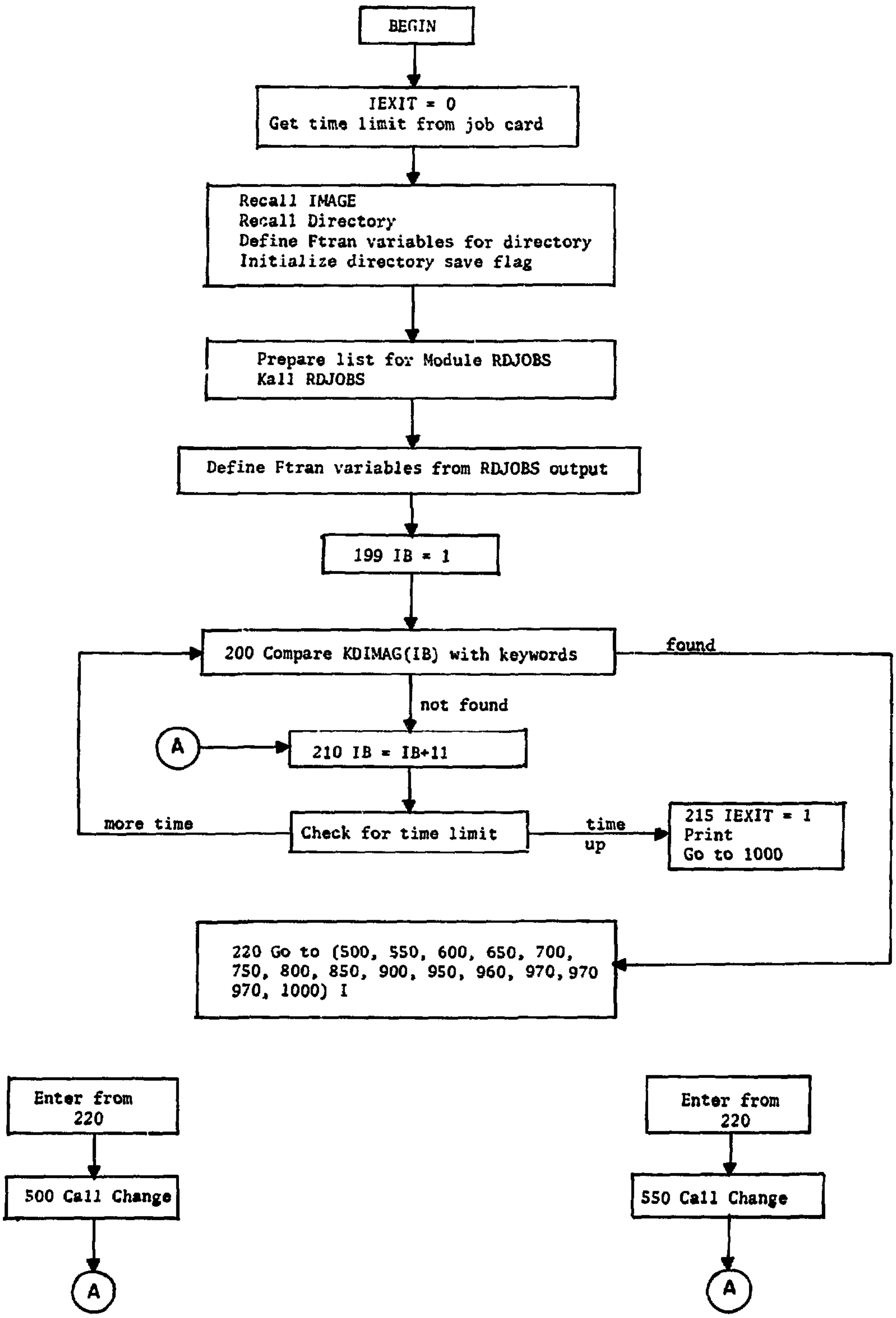
11. Martha Hoyt, Los Alamos Scientific Laboratory, DMPLLL, undocumented code for CDC 6600 computer, February 1972.

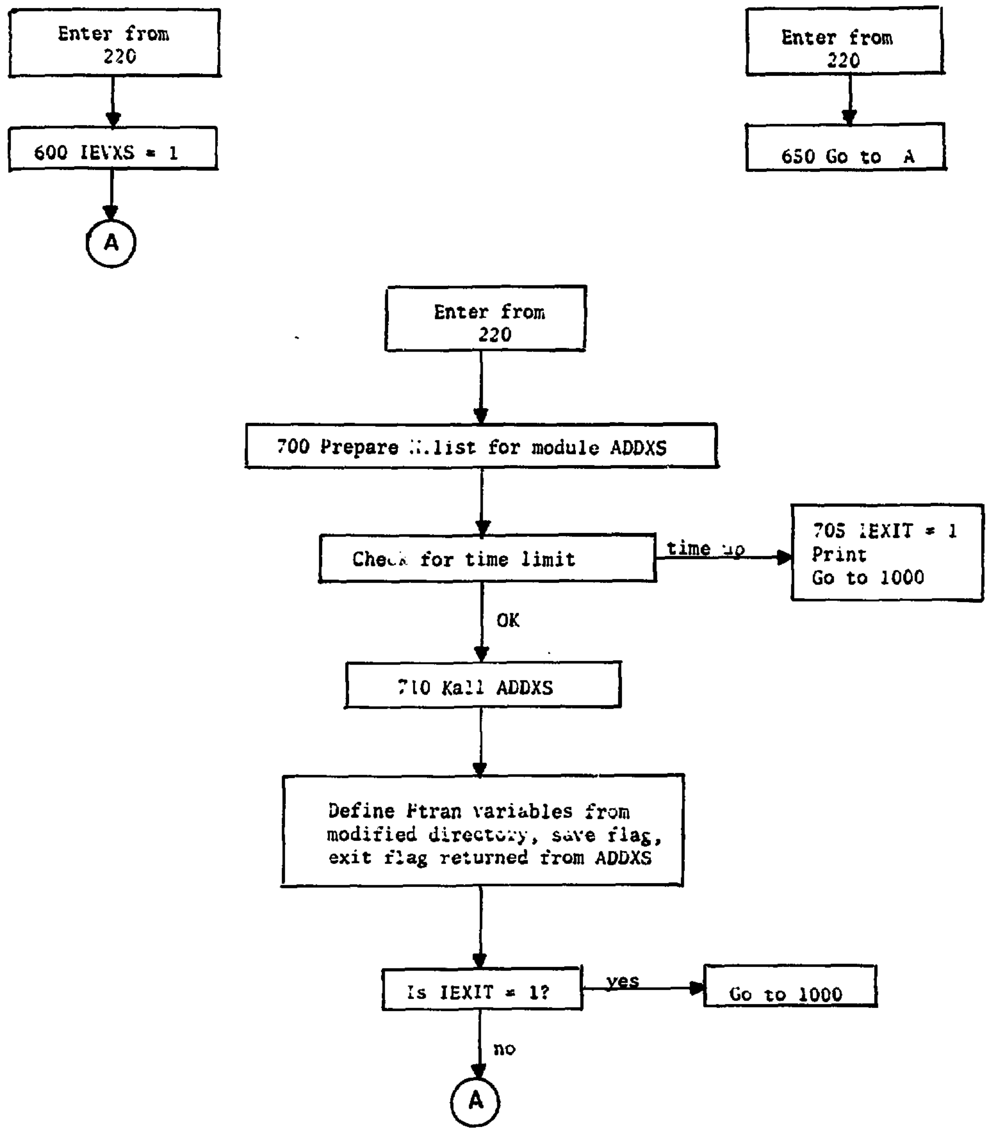
12. Martha Hoyt, Los Alamos Scientific Laboratory, LLLTAPE, undocumented code for CDC 6600 computer, February 1972.

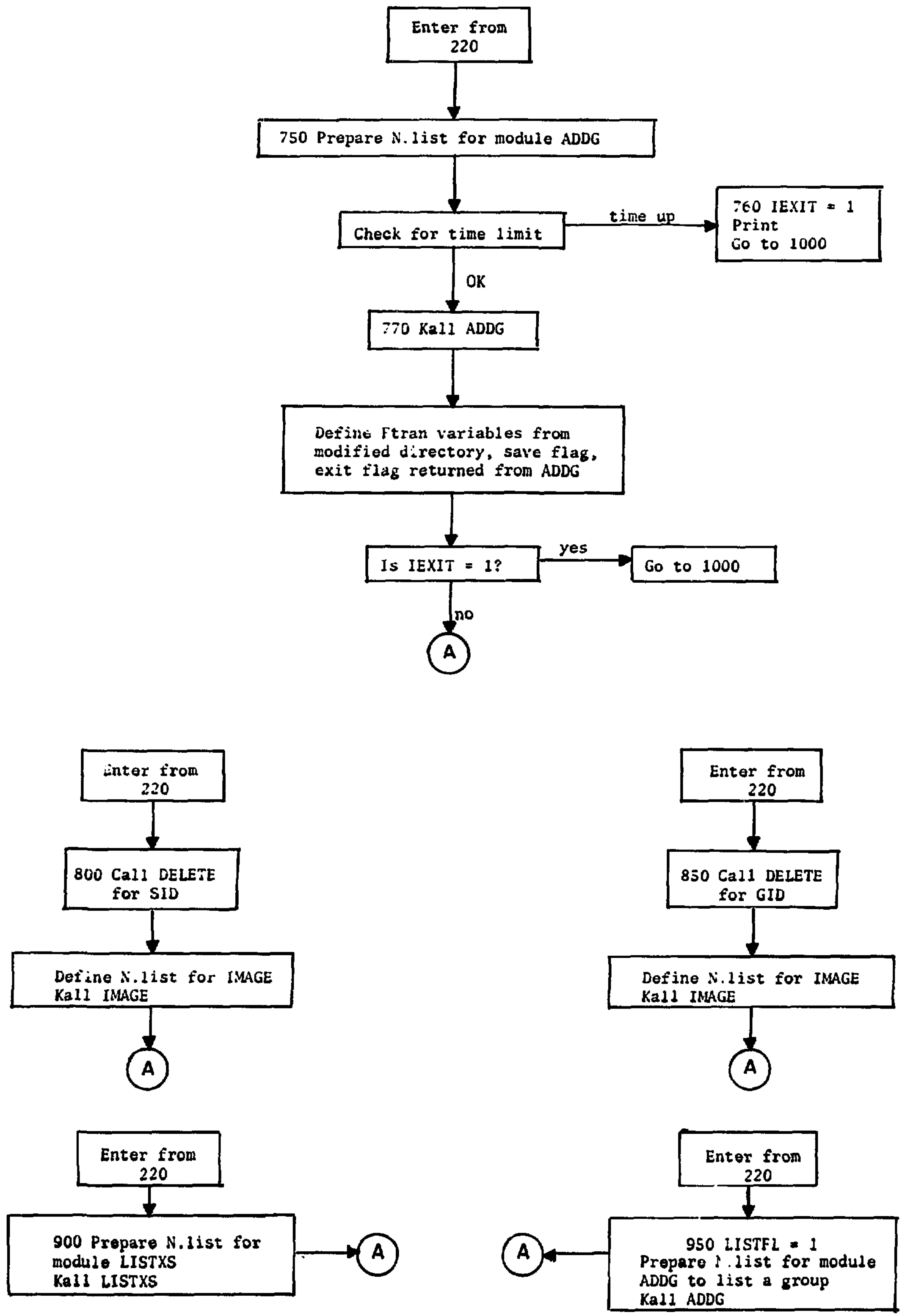
APPENDIX A

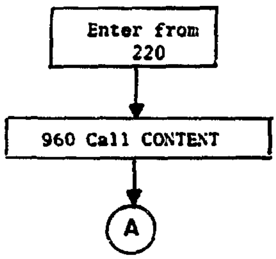
FLOW DIAGRAMS

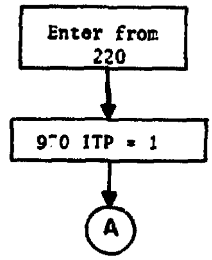
PROGRAM XSMOD

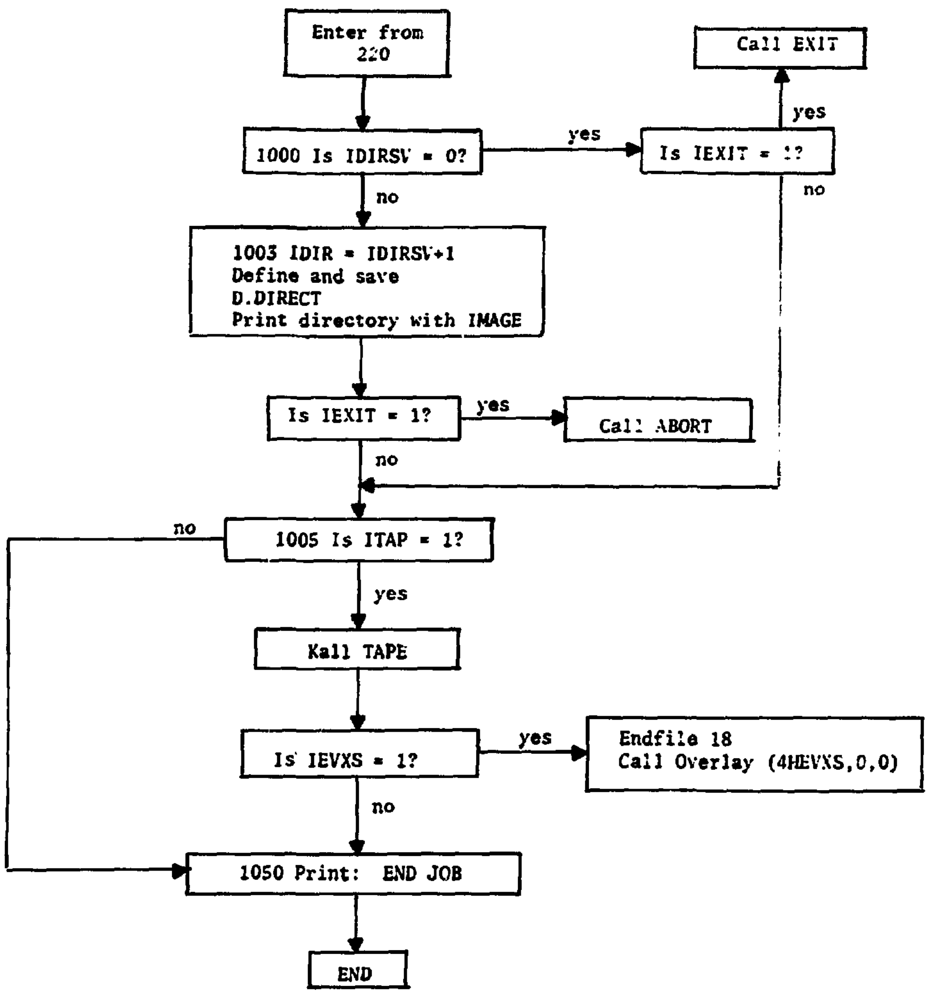



Enter from 220
600 IEVXS = 1
A
Enter from 220
650 Go to A
Enter from 220
700 Prepare list for module ADDXS
Check for time limit
time up
705 IEXIT = 1
Print
Go to 1000
OK
710 Kall ADDXS
Define Ftran variables from modified directory, save flag, exit flag returned from ADDXS
Is IEXIT = 1?
yes
Go to 1000
no
A

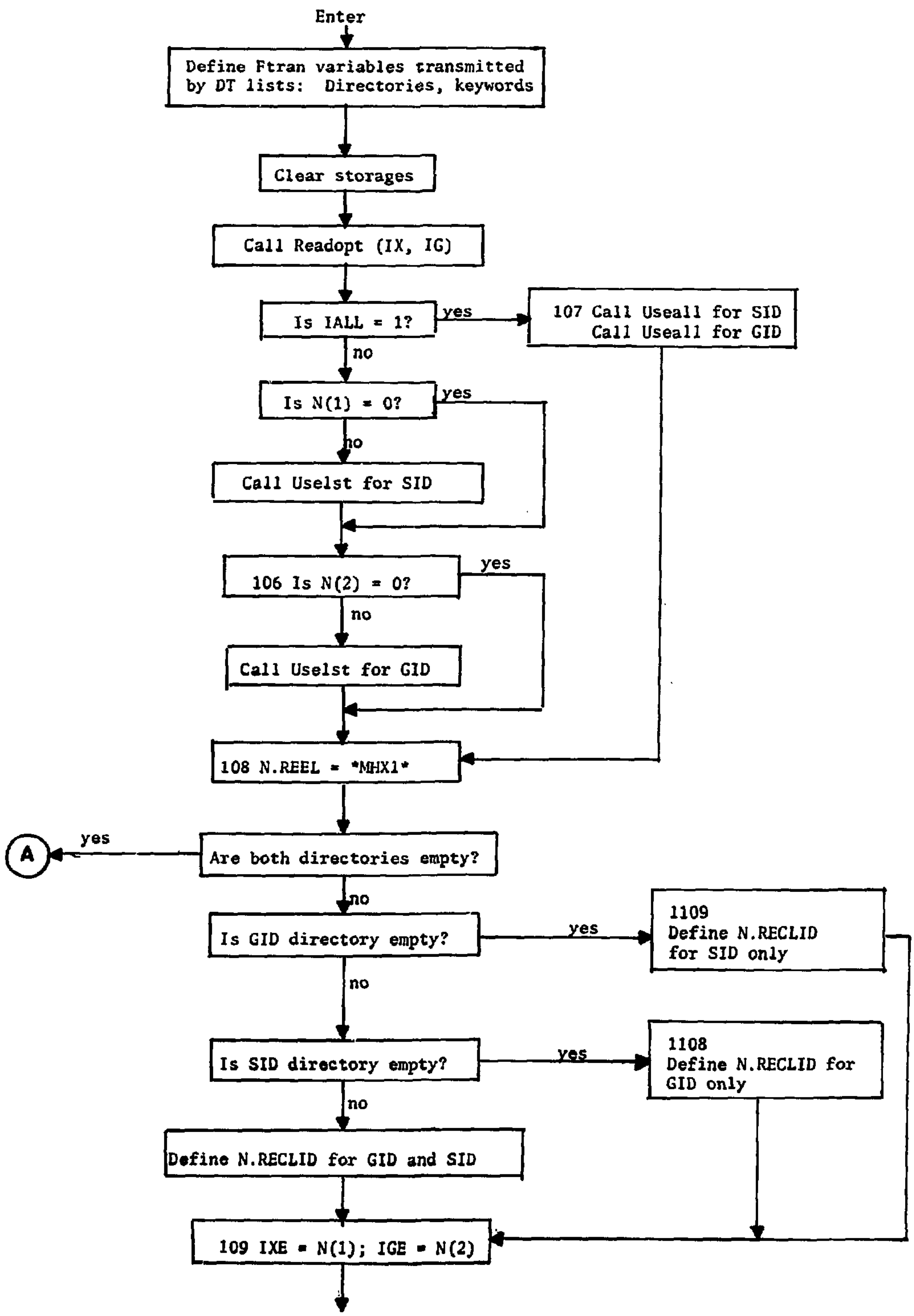

Enter from 220
750 Prepare N.list for module ADDG
Check for time limit
time up
760 IEXIT = 1
Print
Go to 1000
OK
770 Kall ADDG
Define Ftran variables from modified directory, save flag, exit flag returned from ADDG
Is IEXIT = 1?
yes
Go to 1000
no
A
Enter from 220
800 Call DELETE for SID
Define N.list for IMAGE
Kall IMAGE
A
Enter from 220
850 Call DELETE for GID
Define N.list for IMAGE
Kall IMAGE
A
Enter from 220
900 Prepare N.list for module LISTXS
Kall LISTXS
A
Enter from 220
950 LISTFL = 1
Prepare N.list for module ADDG to list a group
Kall ADDG
A

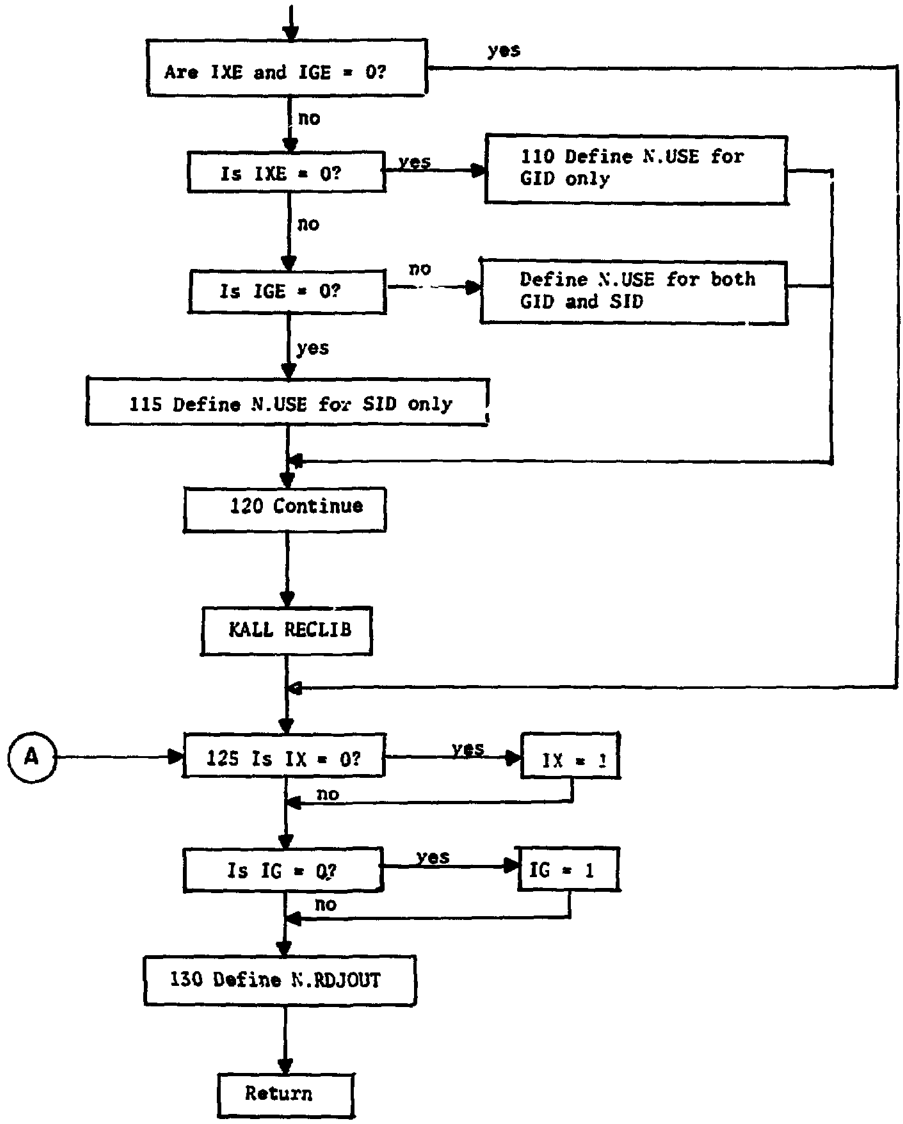

Enter from 220
960 Call CONTENT
A


Enter from 220
970 ITP = 1
A

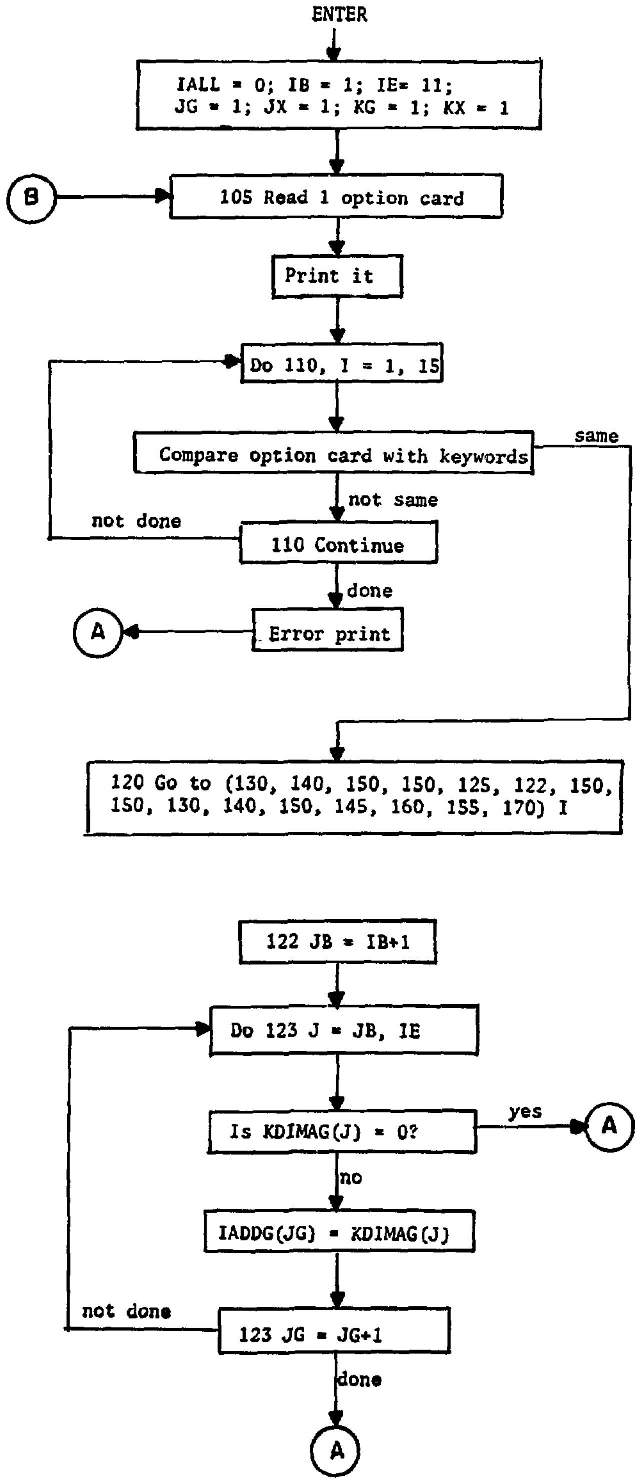

Enter from 220
1000 Is IDIRSV = 0?
yes
Is IEXIT = 1?
yes
Call EXIT
no
no
1003 IDIR = IDIRSV+1
Define and save
D.DIRECT
Print directory with IMAGE
Is IEXIT = 1?
yes
Call ABORT
no
no
1005 Is ITAP = 1?
yes
Kall TAPE
Is IEVXS = 1?
yes
Endfile 18
Call Overlay (4HEVXS,0,0)
no
1050 Print: END JOB
END

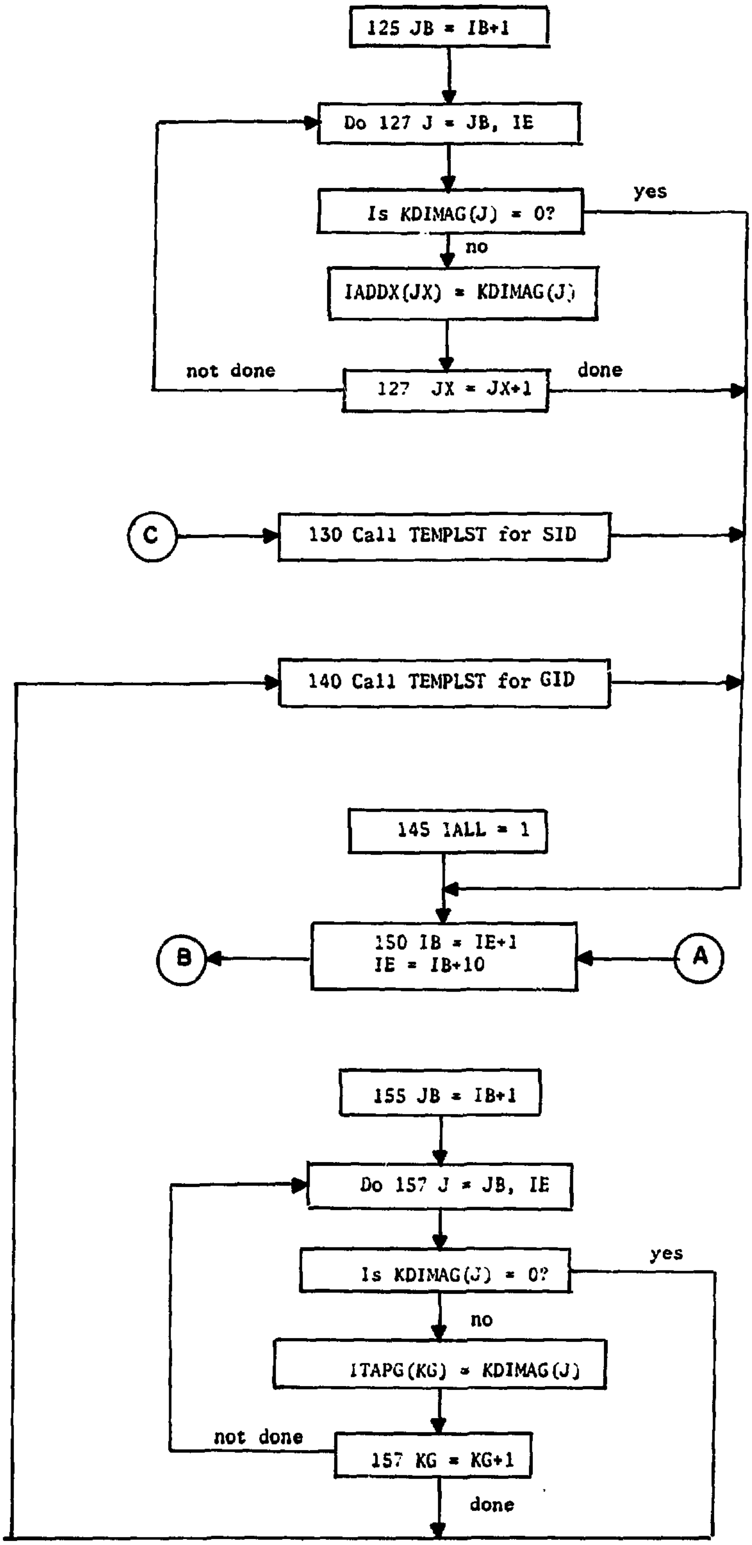
MODULE RDJOBS

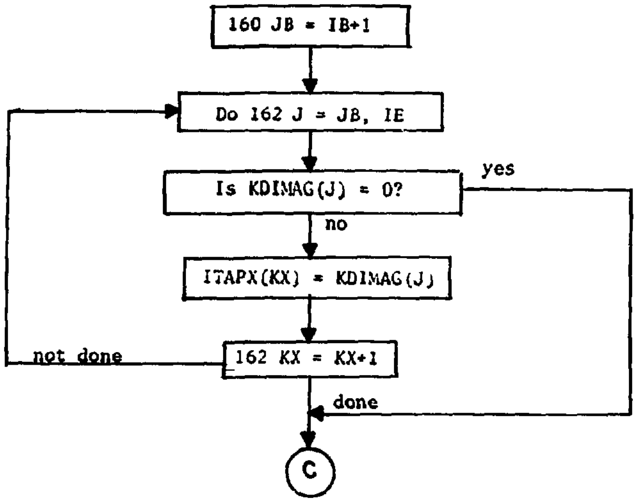


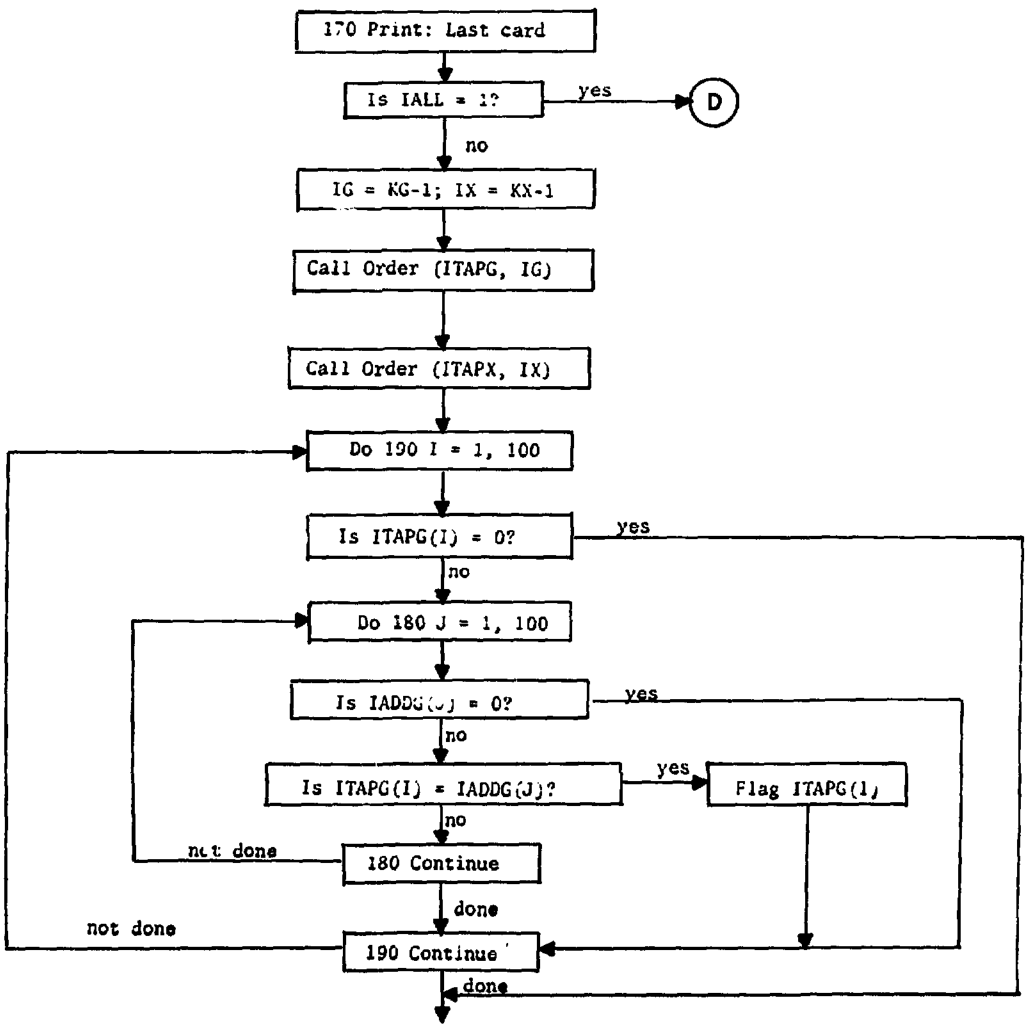

Are IXE and IGE = 0?
yes
no
Is IXE = 0?
yes
110 Define N.USE for GID only
no
Is IGE = 0?
no
Define N.USE for both GID and SID
yes
115 Define N.USE for SID only
120 Continue
KALL RECLIB
A
125 Is IX = 0?
yes
IX = 1
no
Is IG = 0?
yes
IG = 1
no
130 Define N.RDJOUT
Return

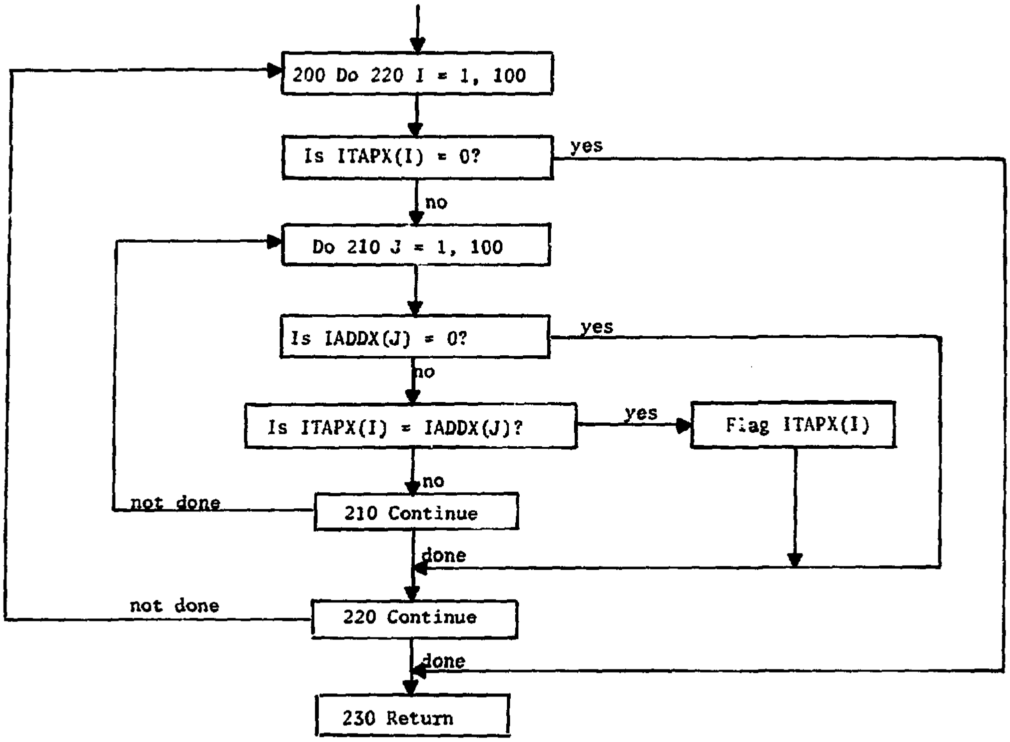
SUBROUTINE READOPT

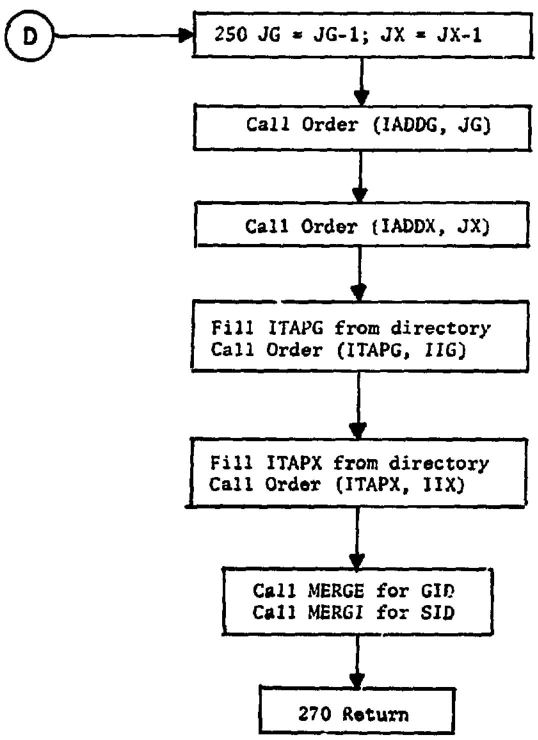



125 JB = IB+1
Do 127 J = JB, IE
Is KDIMAG(J) = 0?
yes
no
IADDX(JX) = KDIMAG(J)
not done
127 JX = JX+1
done
C
130 Call TEMPLST for SID
140 Call TEMPLST for GID
145 IALL = 1
B
150 IB = IE+1
IE = IB+10
A
155 JB = IB+1
Do 157 J = JB, IE
Is KDIMAG(J) = 0?
yes
no
ITAPG(KG) = KDIMAG(J)
not done
157 KG = KG+1
done

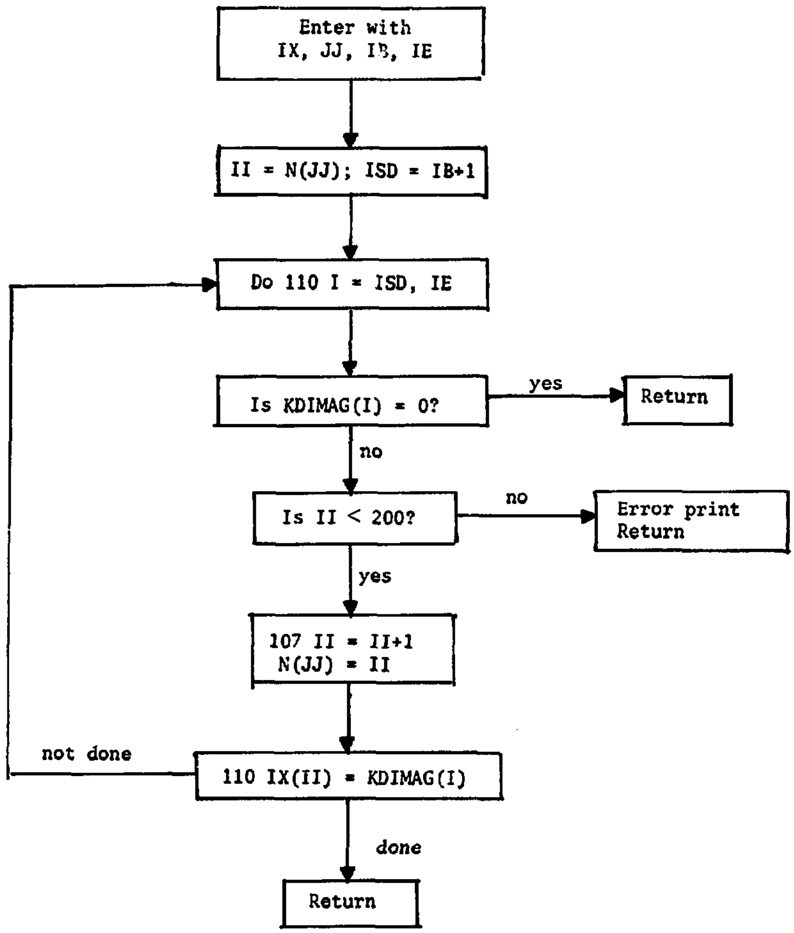

160 JB = IB+1
Do 162 J = JB, IE
Is KDIMAG(J) = 0?
yes
no
ITAPX(KX) = KDIMAG(J)
not done
162 KX = KX+1
done
C


170 Print: Last card
Is IALL = 1?
yes
D
no
IG = KG-1; IX = KX-1
Call Order (ITAPG, IG)
Call Order (ITAPX, IX)
Do 190 I = 1, 100
Is ITAPG(I) = 0?
yes
no
Do 180 J = 1, 100
Is IADDG(J) = 0?
yes
no
Is ITAPG(I) = IADDG(J)?
yes
Flag ITAPG(I)
no
not done
180 Continue
done
not done
190 Continue
done

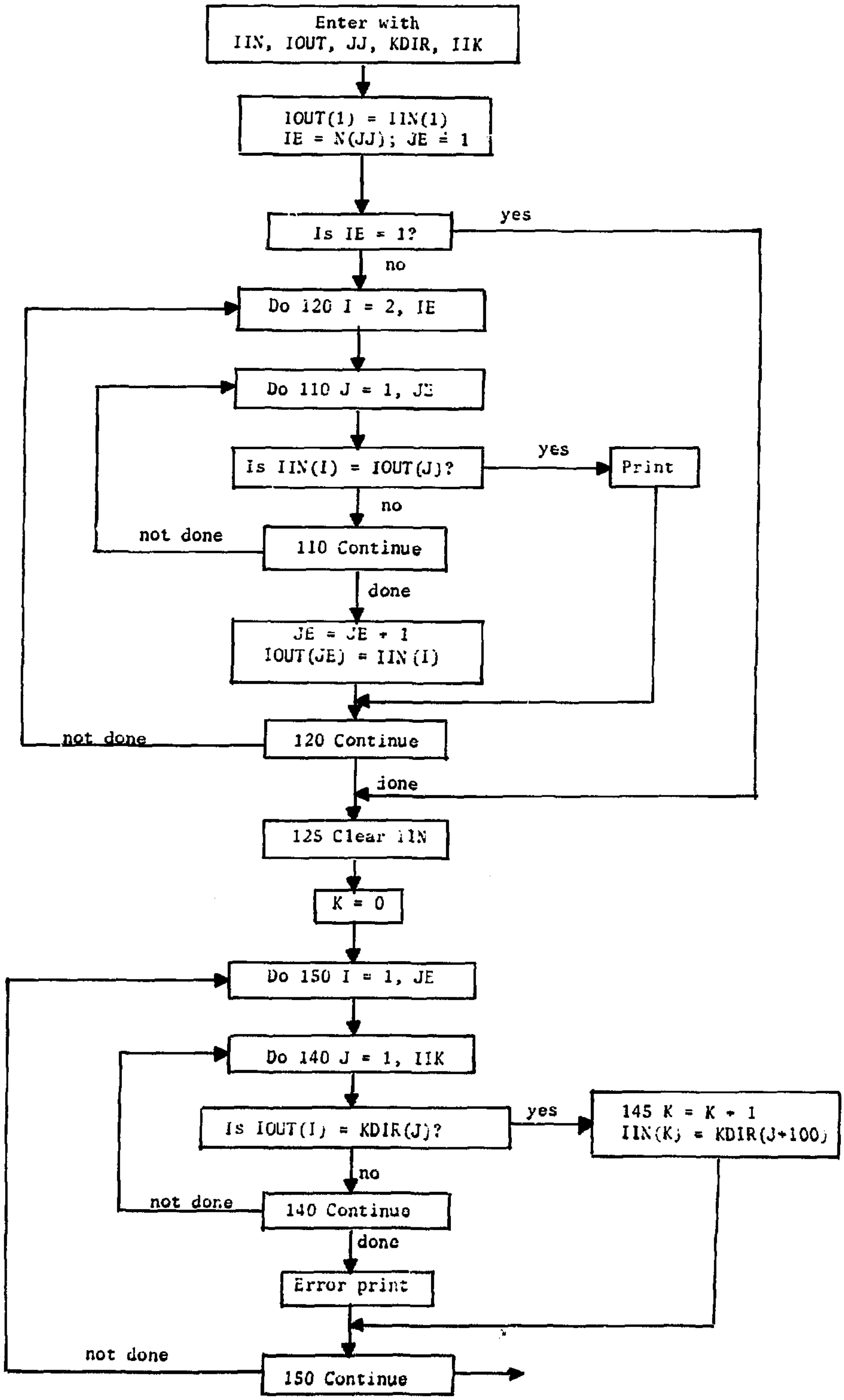

200 Do 220 I = 1, 100
Is ITAPX(I) = 0?
yes
no
Do 210 J = 1, 100
Is IADDX(J) = 0?
yes
no
Is ITAPX(I) = IADDX(J)?
yes
Flag ITAPX(I)
no
not done
210 Continue
done
not done
220 Continue
done
230 Return

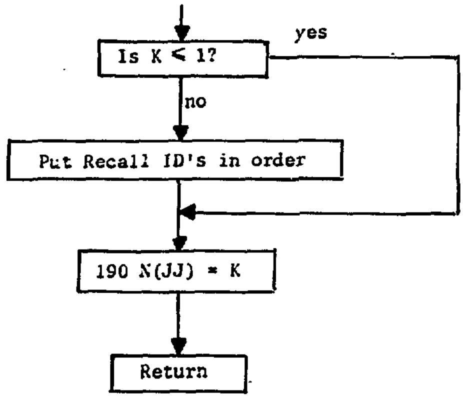

D
250 JG = JG-1; JX = JX-1
Call Order (IADDG, JG)
Call Order (IADDX, JX)
Fill ITAPG from directory
Call Order (ITAPG, IIG)
Fill ITAPX from directory
Call Order (ITAPX, IIX)
Call MERGE for GID
Call MERGI for SID
270 Return

SUBROUTINE TEMPLST

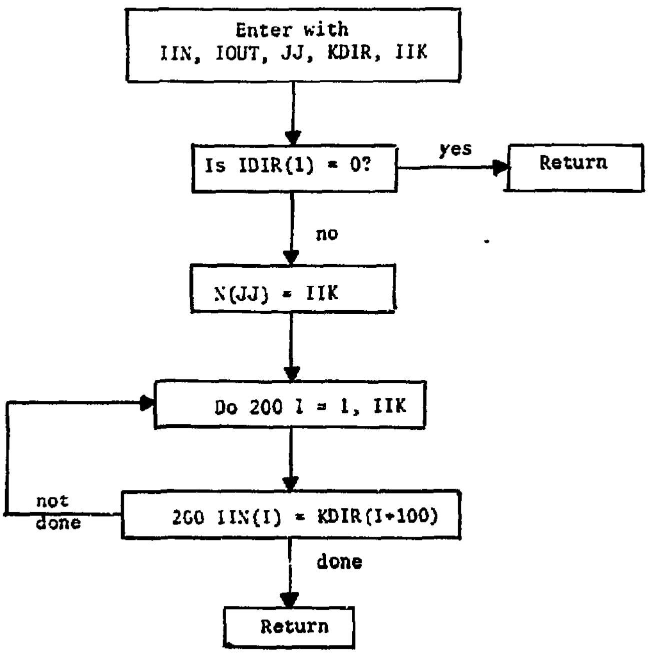


SUBROUTINE USELST

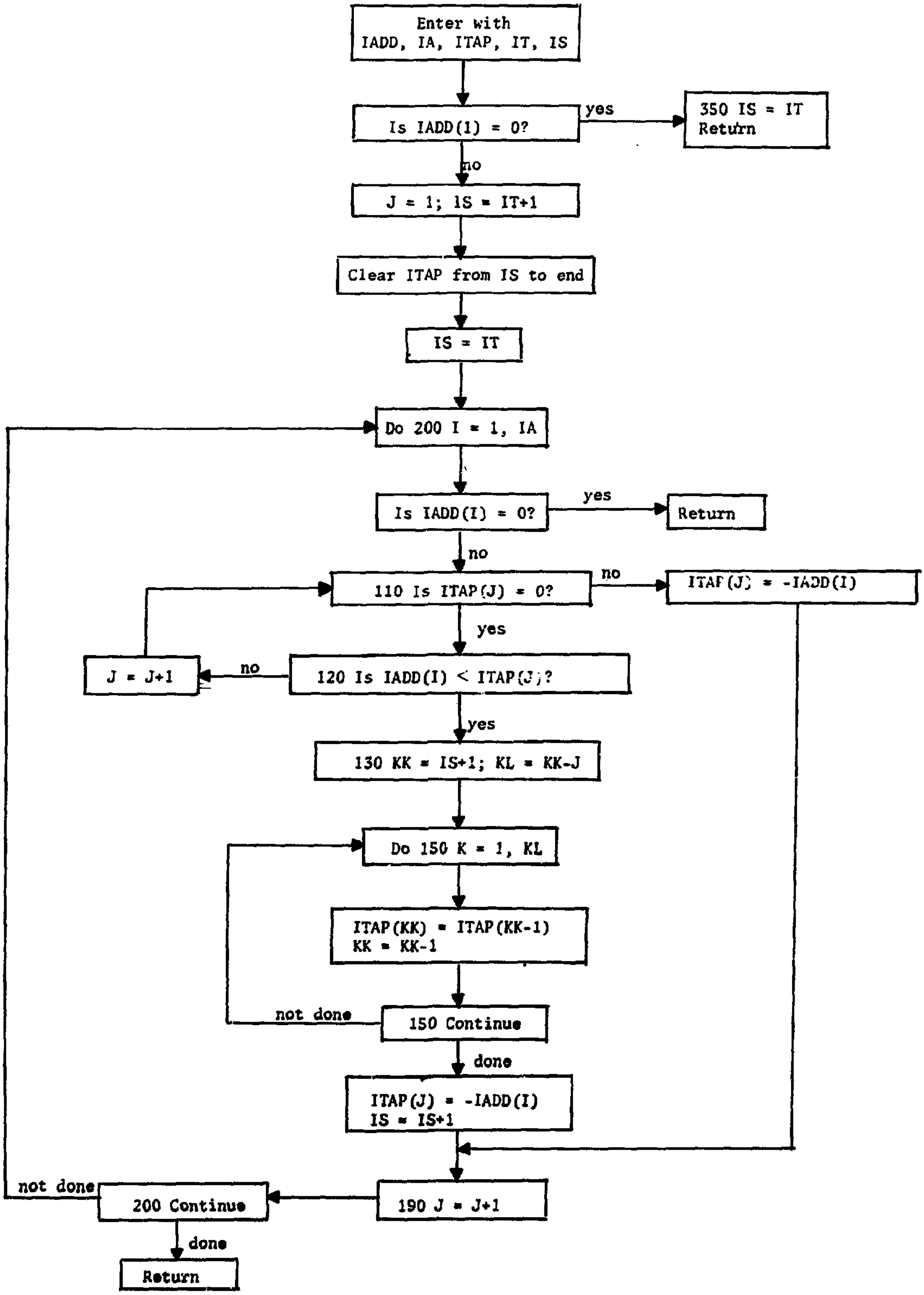




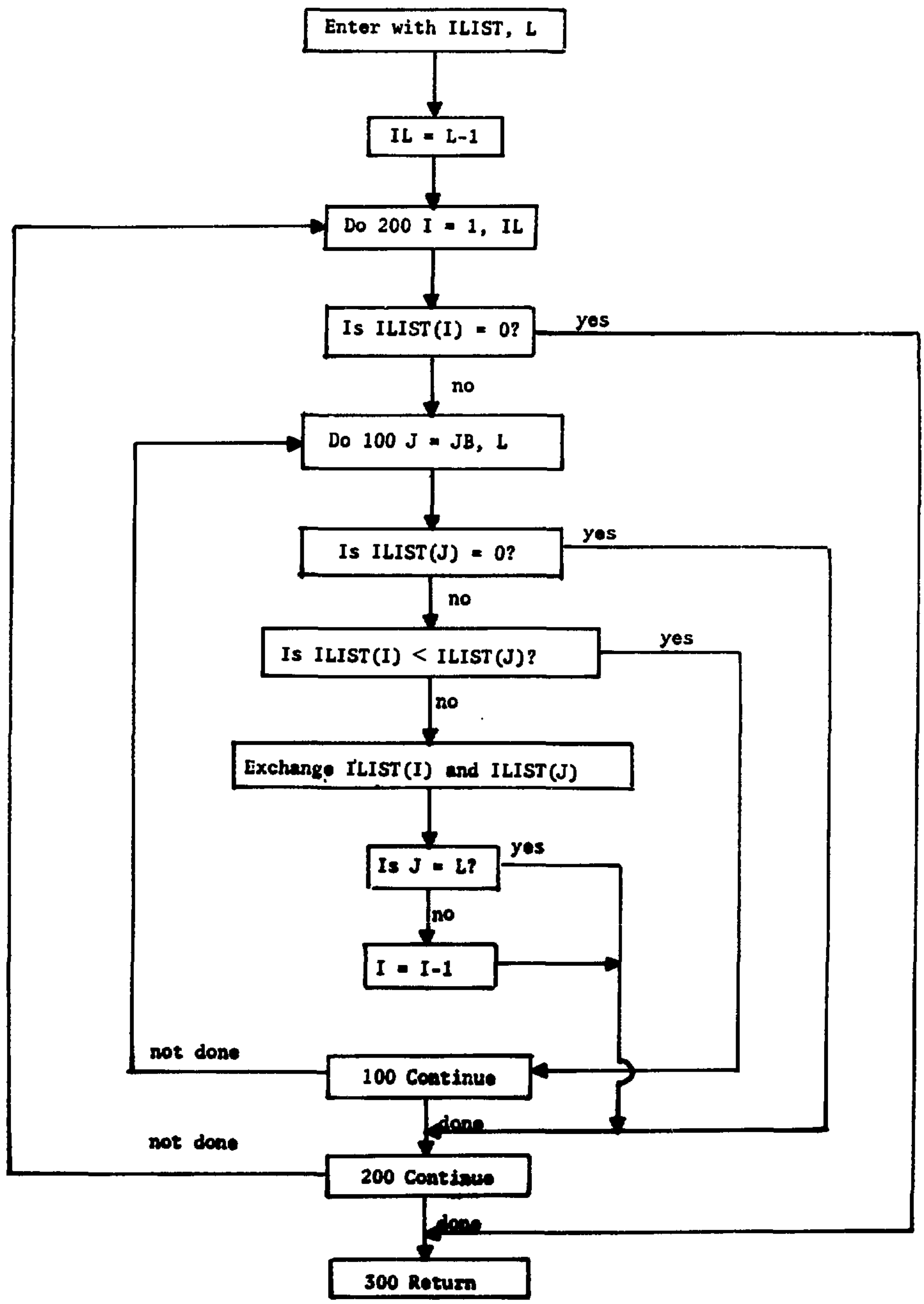
ENTRY USEALL



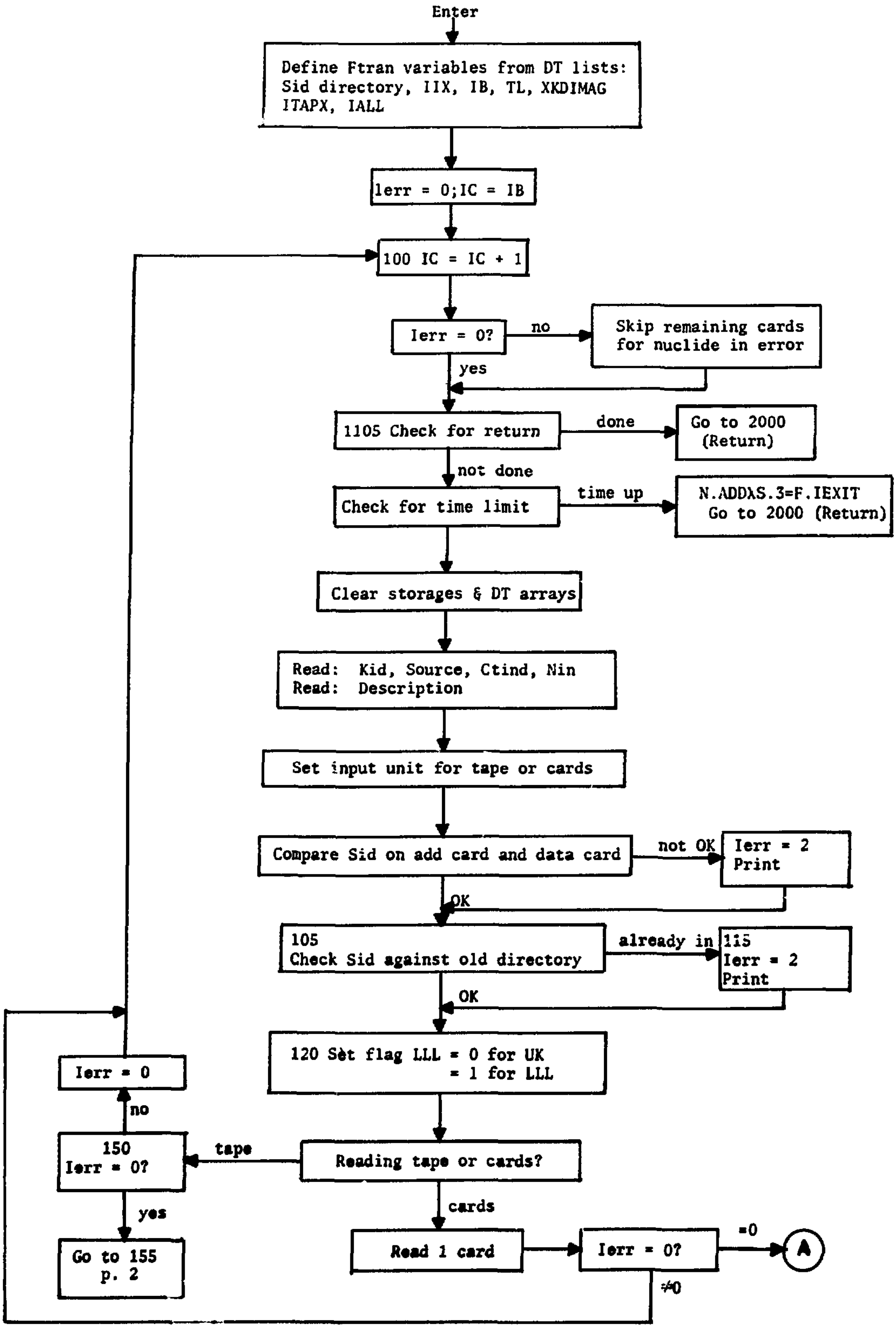
SUBROUTINE MERGE

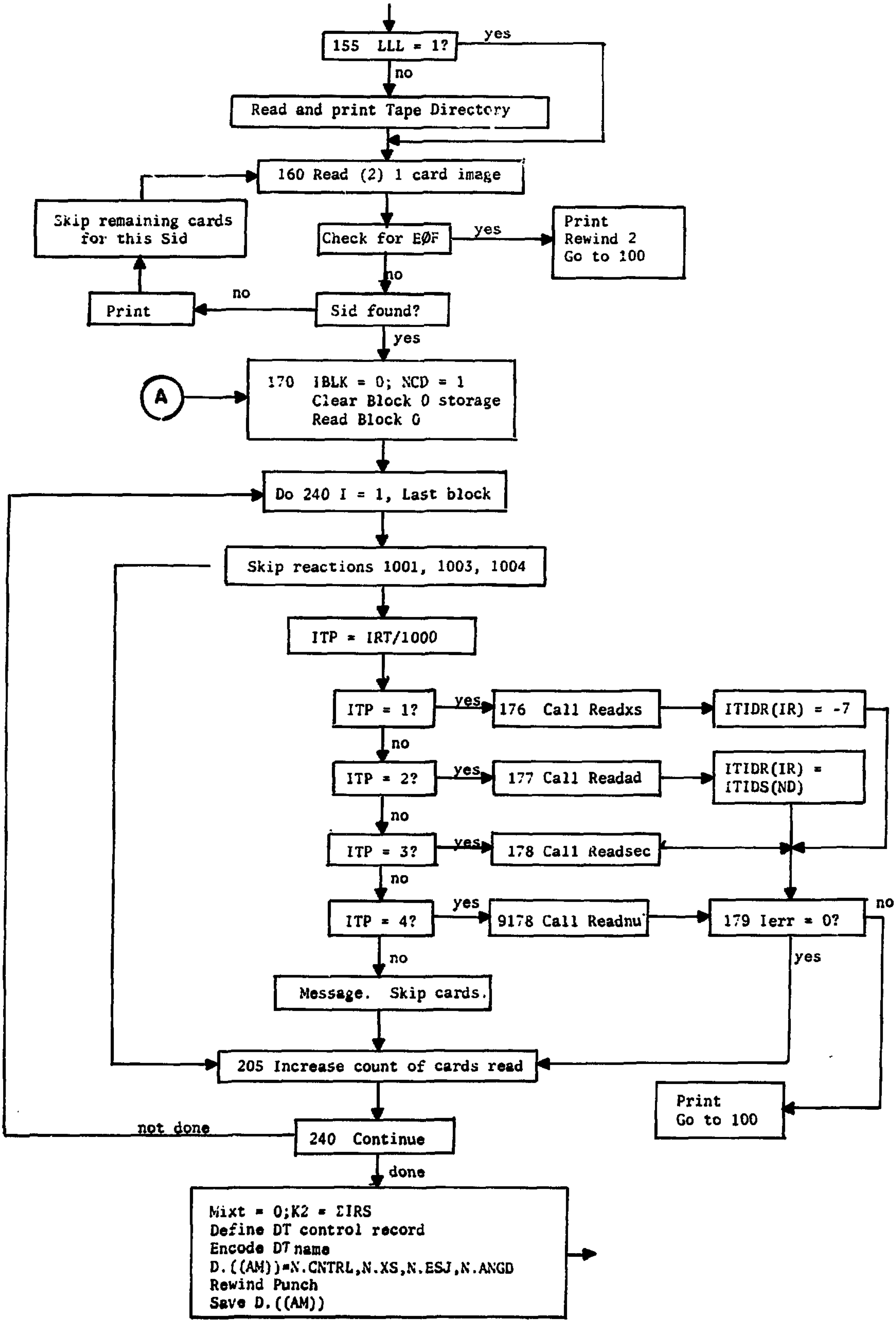


SUBROUTINE ORDER

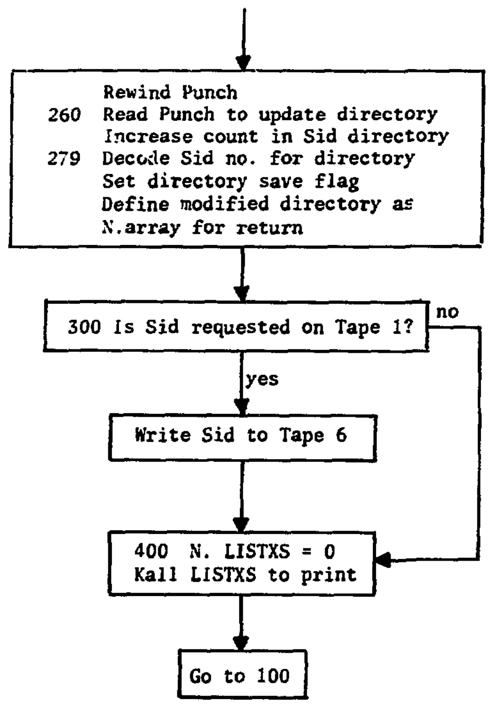


MODULE ADDXS

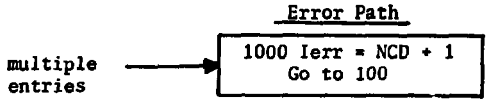




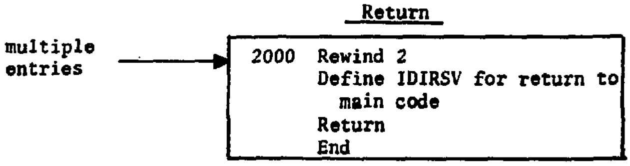
155 LLL = 1?
yes
no
Read and print Tape Directory
160 Read (2) 1 card image
Skip remaining cards for this Sid
Check for EØF
yes
Print
Rewind 2
Go to 100
no
Print
no
Sid found?
yes
A
170 IBLK = 0; NCD = 1
Clear Block 0 storage
Read Block 0
Do 240 I = 1, Last block
Skip reactions 1001, 1003, 1004
ITP = IRT/1000
ITP = 1?
yes
176 Call Readxs
ITIDR(IR) = -7
no
ITP = 2?
yes
177 Call Readad
ITIDR(IR) = ITIDS(ND)
no
ITP = 3?
yes
178 Call Readsec
no
ITP = 4?
yes
9178 Call Readnu
179 Ierr = 0?
no
no
yes
Message. Skip cards.
205 Increase count of cards read
Print
Go to 100
not done
240 Continue
done
Mixt = 0;K2 = ΣIRS
Define DT control record
Encode DT name
D.((AM))=N.CNTRL,N.XS,N.ESJ,N.ANGD
Rewind Punch
Save D.((AM))



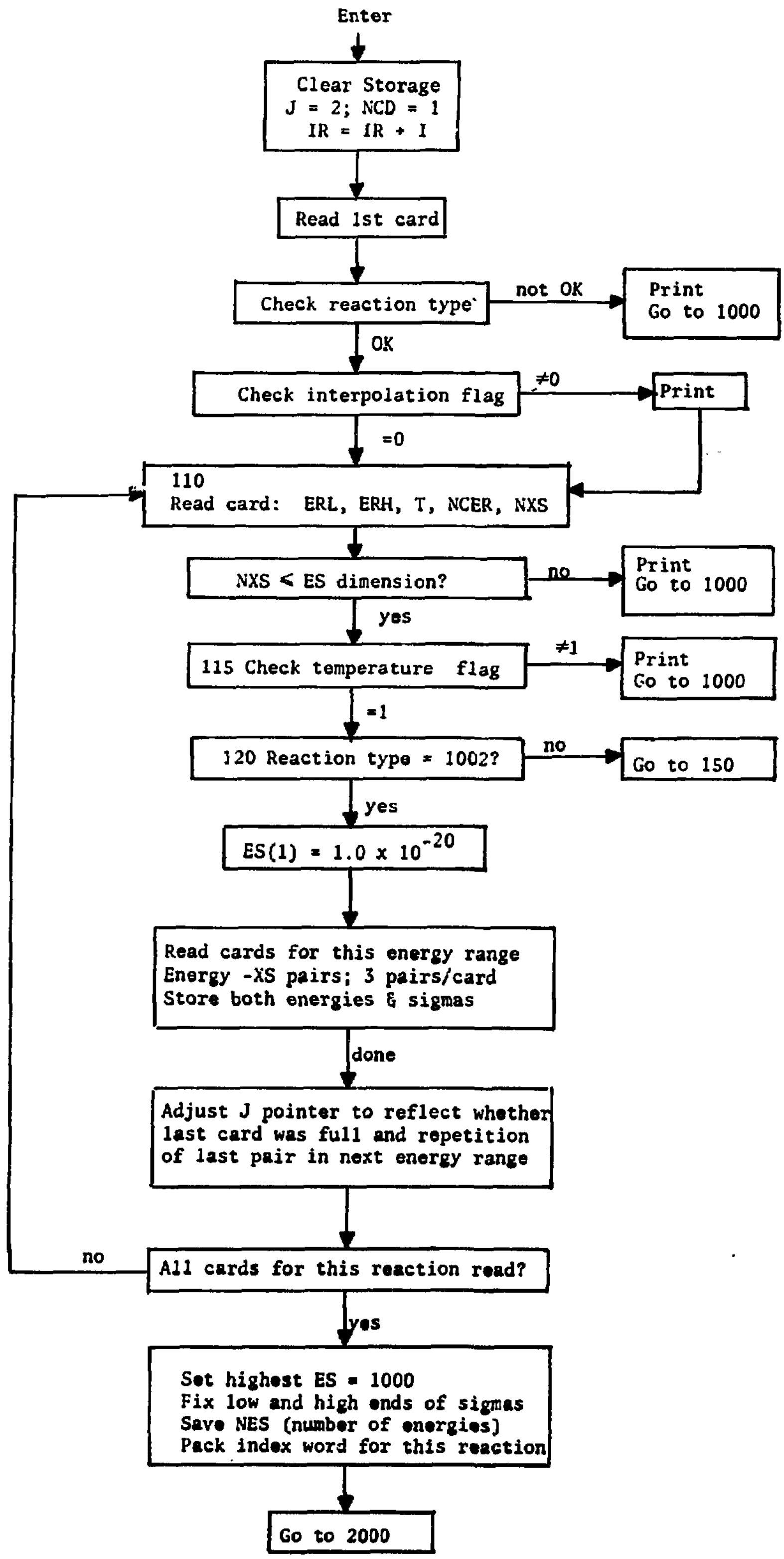
Error Path

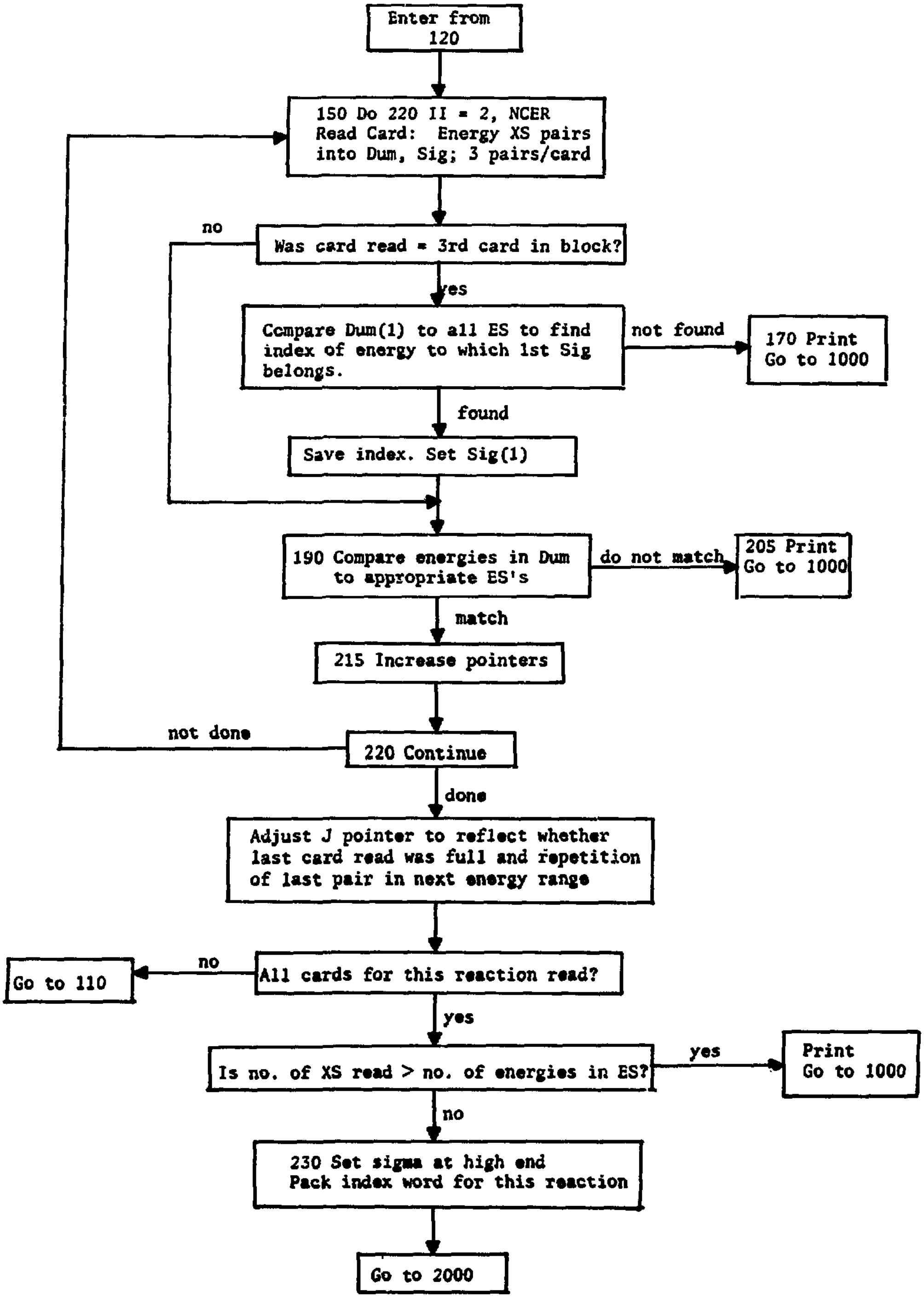


Return

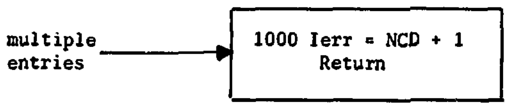


SUBROUTINE READXS

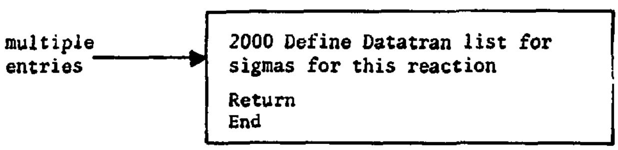



Enter from 120
150 Do 220 II = 2, NCER
Read Card: Energy XS pairs into Dum, Sig; 3 pairs/card
Was card read = 3rd card in block?
no
yes
Compare Dum(1) to all ES to find index of energy to which 1st Sig belongs.
not found
170 Print
Go to 1000
found
Save index. Set Sig(1)
190 Compare energies in Dum to appropriate ES's
do not match
205 Print
Go to 1000
match
215 Increase pointers
not done
220 Continue
done
Adjust J pointer to reflect whether last card read was full and repetition of last pair in next energy range
All cards for this reaction read?
no
Go to 110
yes
Is no. of XS read > no. of energies in ES?
yes
Print
Go to 1000
no
230 Set sigma at high end
Pack index word for this reaction
Go to 2000

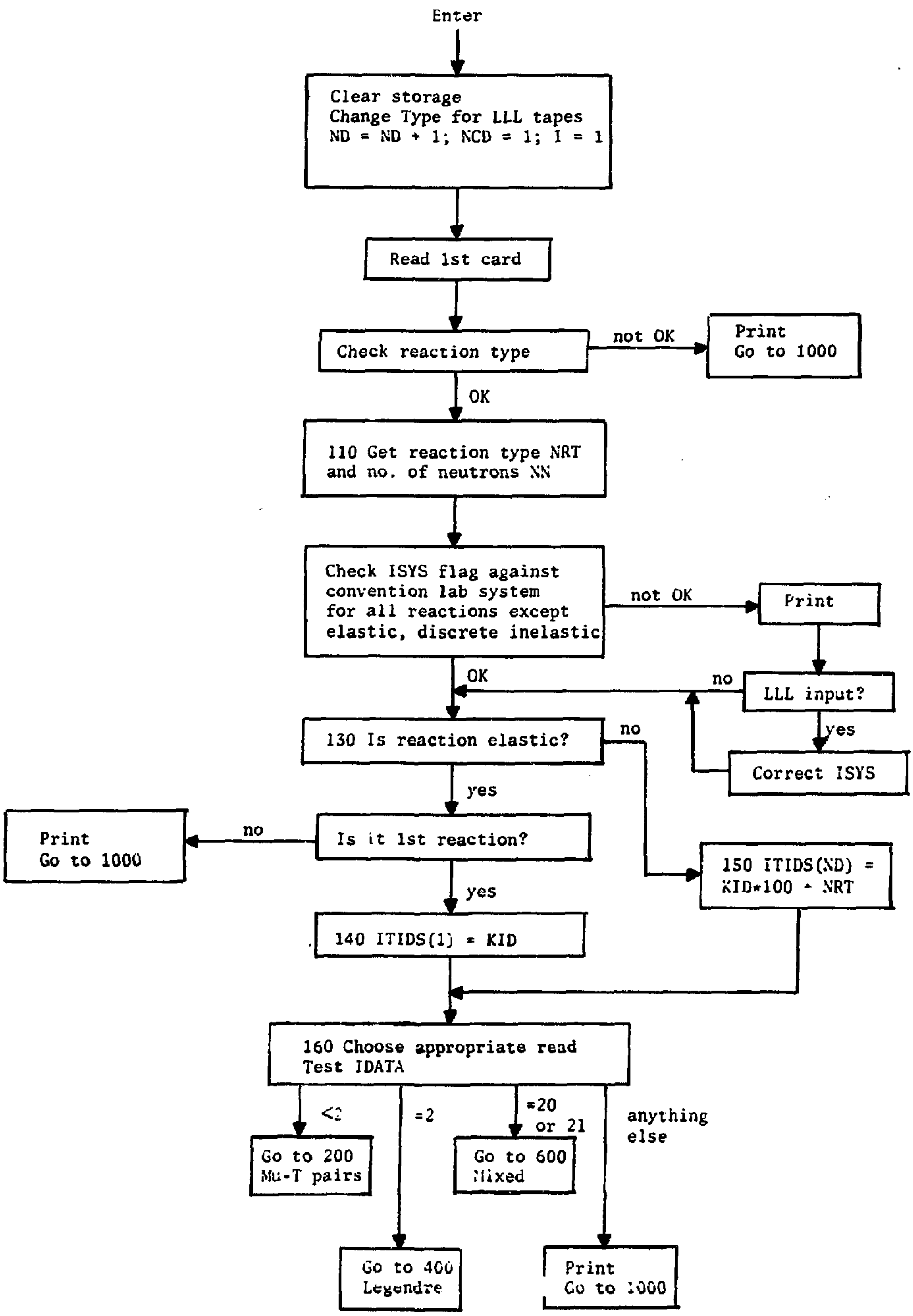
Error Path



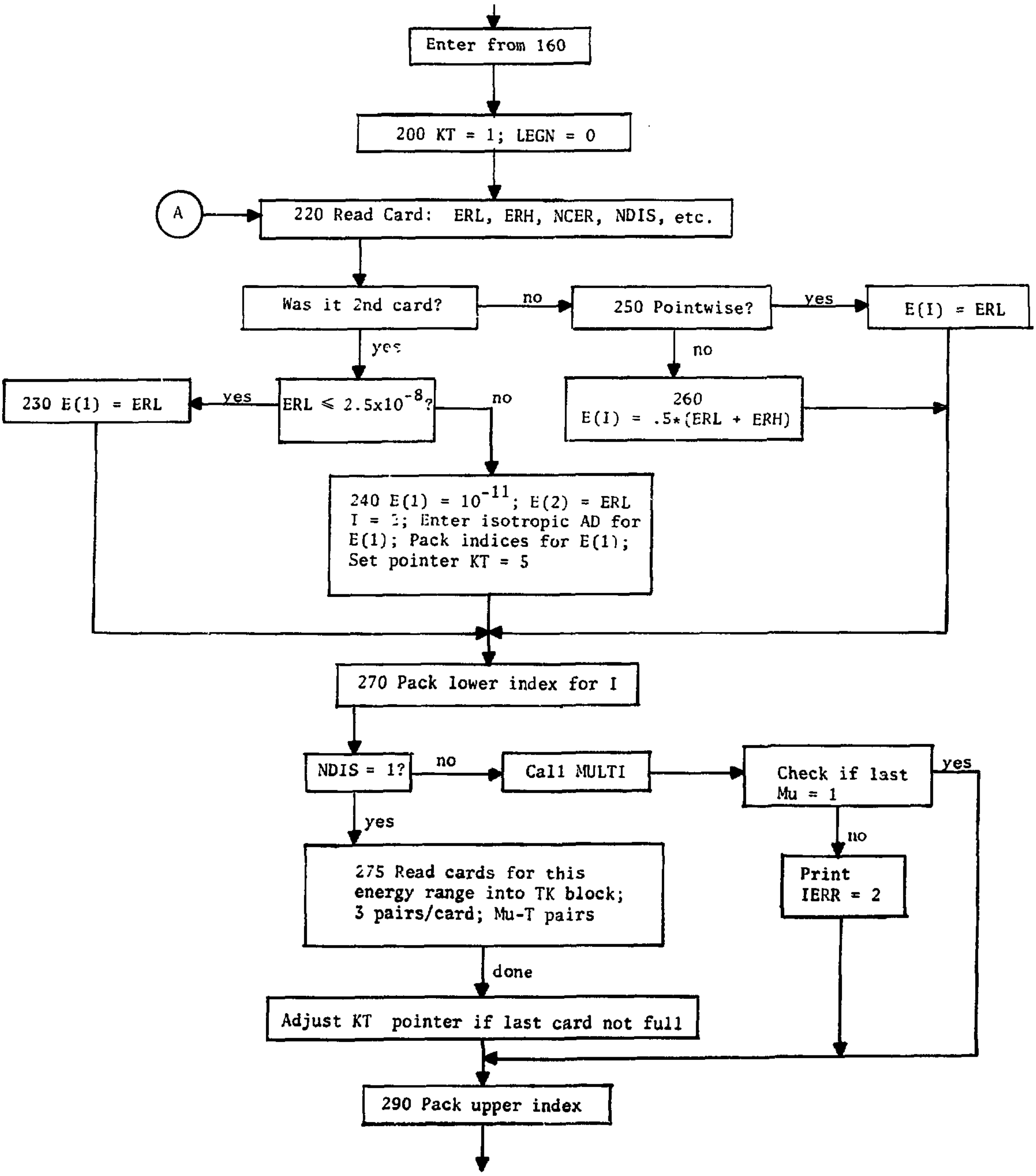
Normal Return

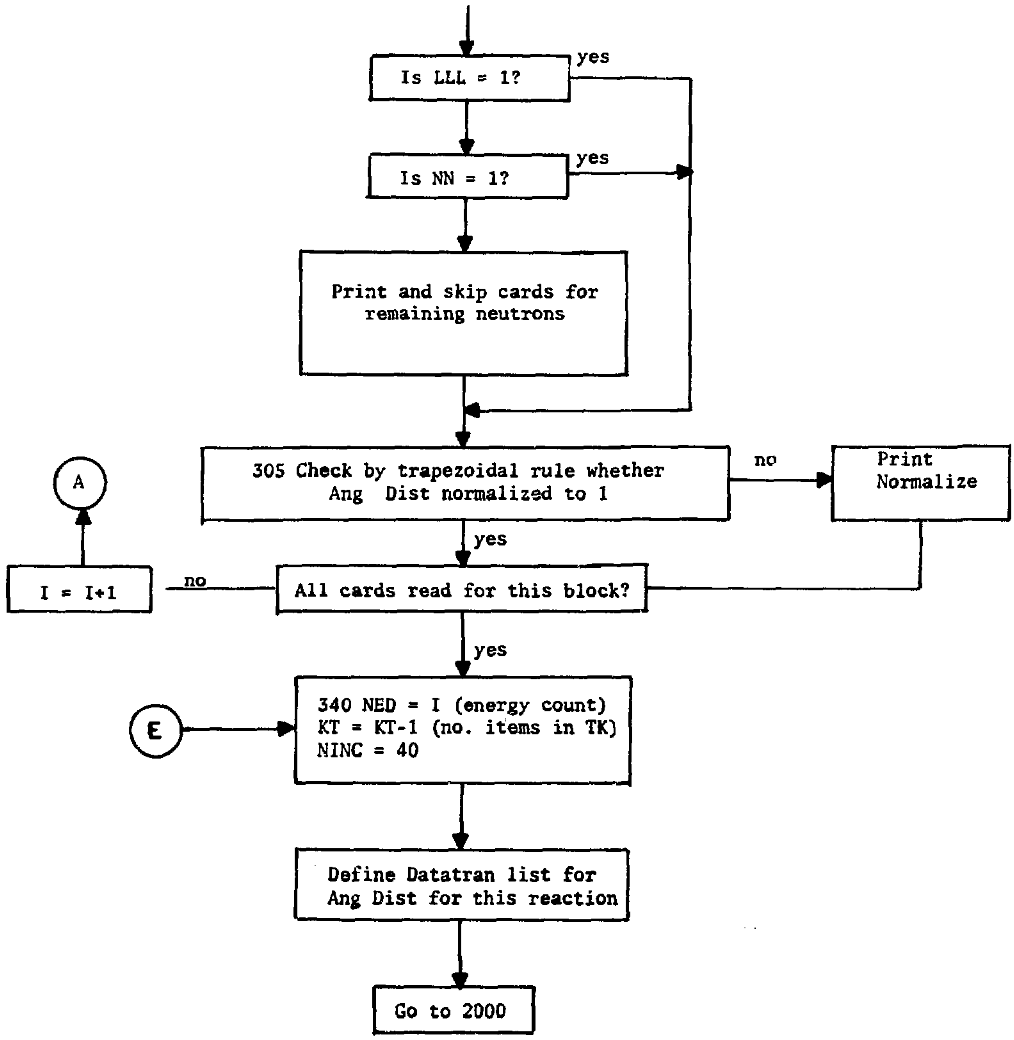


SUBROUTINE READAD



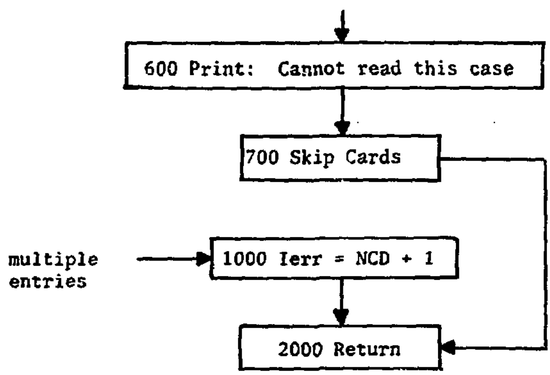
Mu-T Pairs


Enter from 160
200 KT = 1; LEGN = 0
A
220 Read Card: ERL, ERH, NCER, NDIS, etc.
Was it 2nd card?
no
250 Pointwise?
yes
E(I) = ERL
yes
no
230 E(1) = ERL
yes
ERL ≤ 2.5x10^{-8}?
no
260
E(I) = .5*(ERL + ERH)
240 E(1) = 10^{-11}; E(2) = ERL
I = 2; Enter isotropic AD for
E(1); Pack indices for E(1);
Set pointer KT = 5
270 Pack lower index for I
NDIS = 1?
no
Call MULTI
Check if last
Mu = 1
yes
no
yes
275 Read cards for this
energy range into TK block;
3 pairs/card; Mu-T pairs
Print
IERR = 2
done
Adjust KT pointer if last card not full
290 Pack upper index

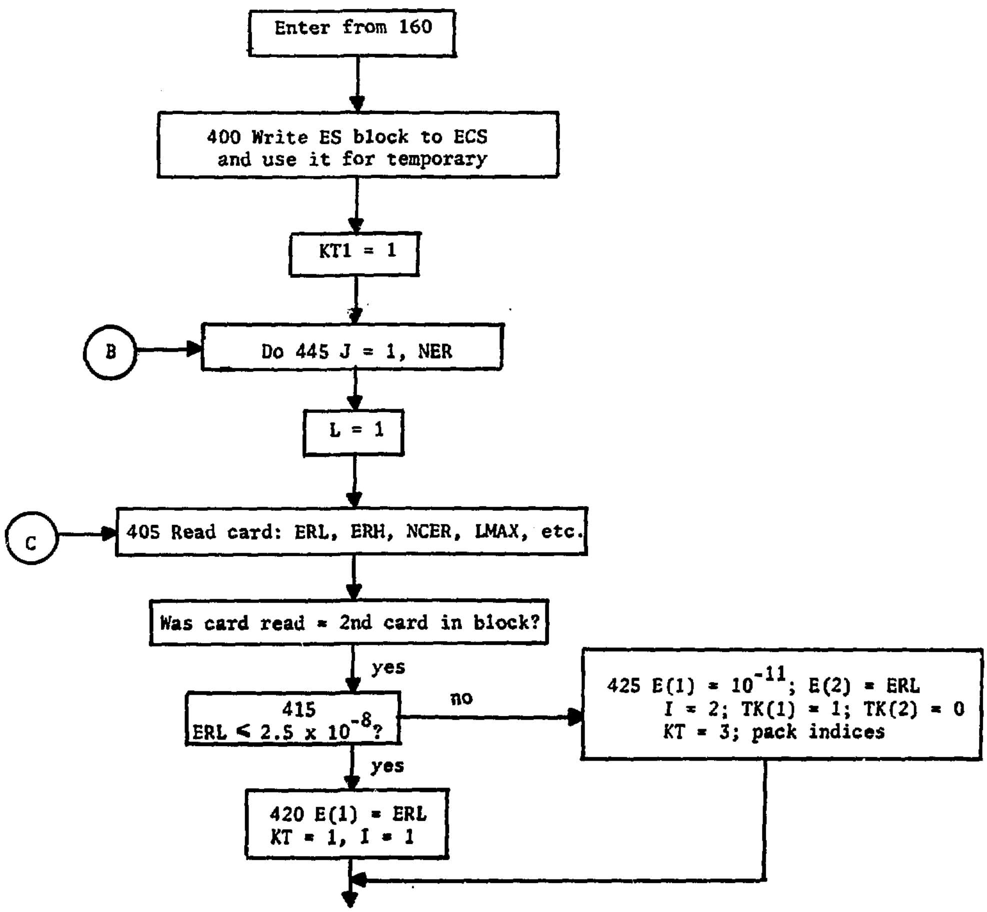


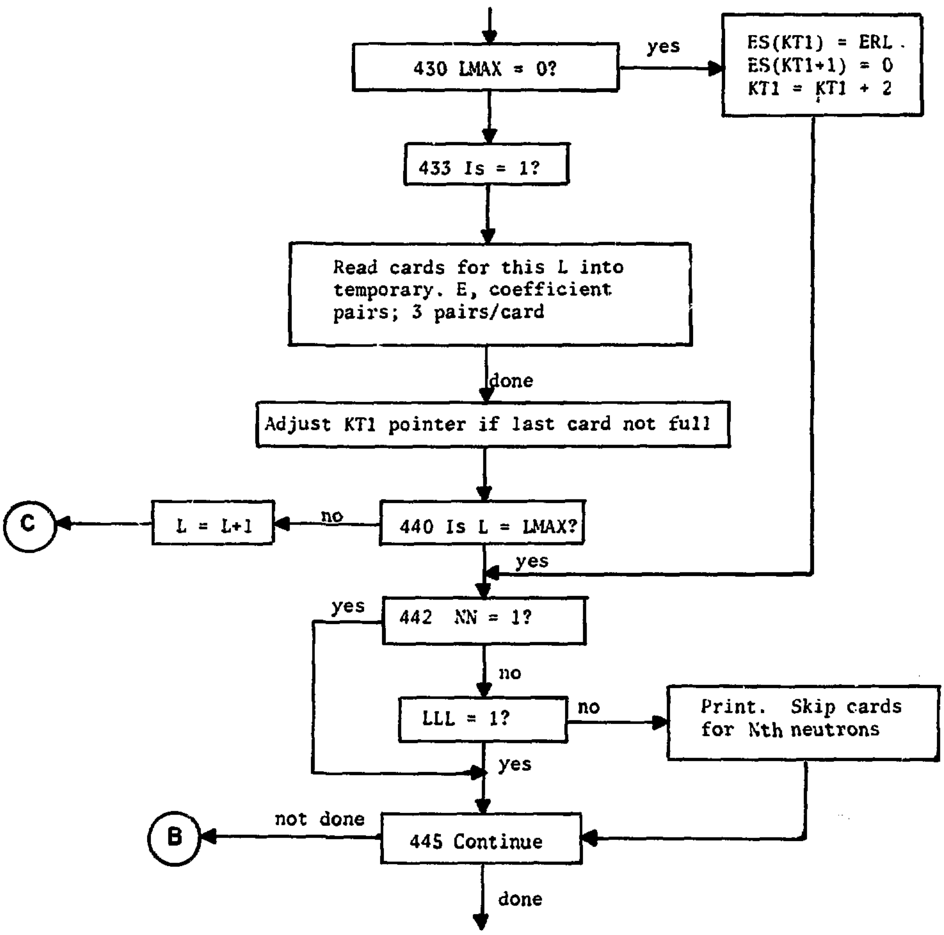
Is LLL = 1?
yes
Is NN = 1?
yes
Print and skip cards for remaining neutrons
305 Check by trapezoidal rule whether Ang Dist normalized to 1
no
Print Normalize
A
yes
I = I+1
no
All cards read for this block?
yes
E
340 NED = I (energy count)
KT = KT-1 (no. items in TK)
NINC = 40
Define Datatran list for Ang Dist for this reaction
Go to 2000

Mixed Legendre and Mu-T

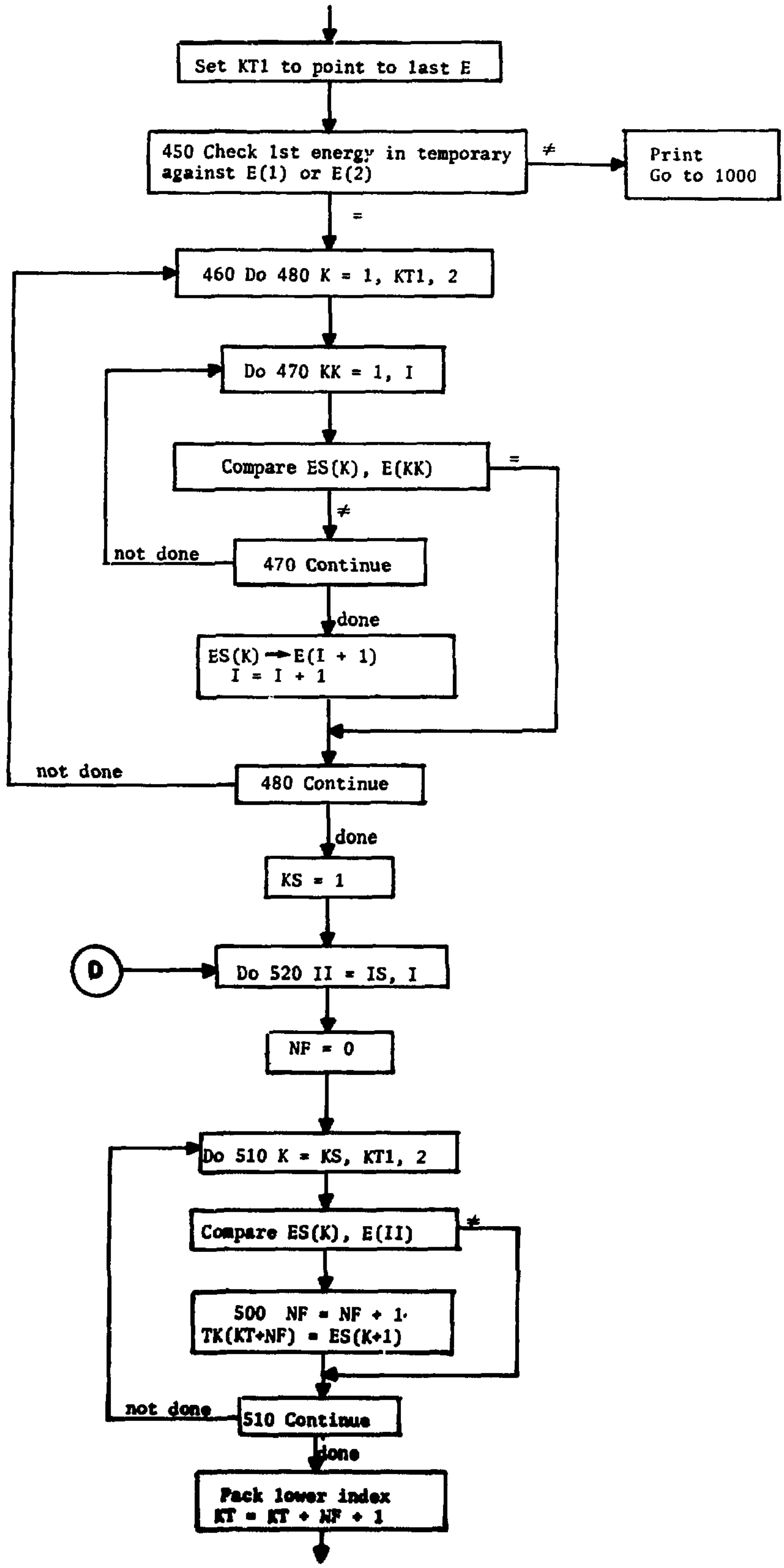


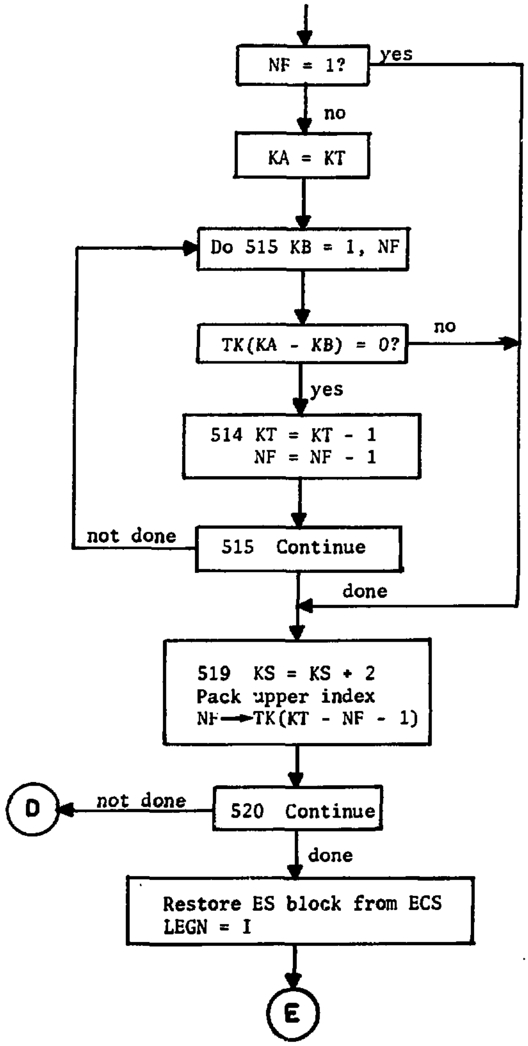
Legendre



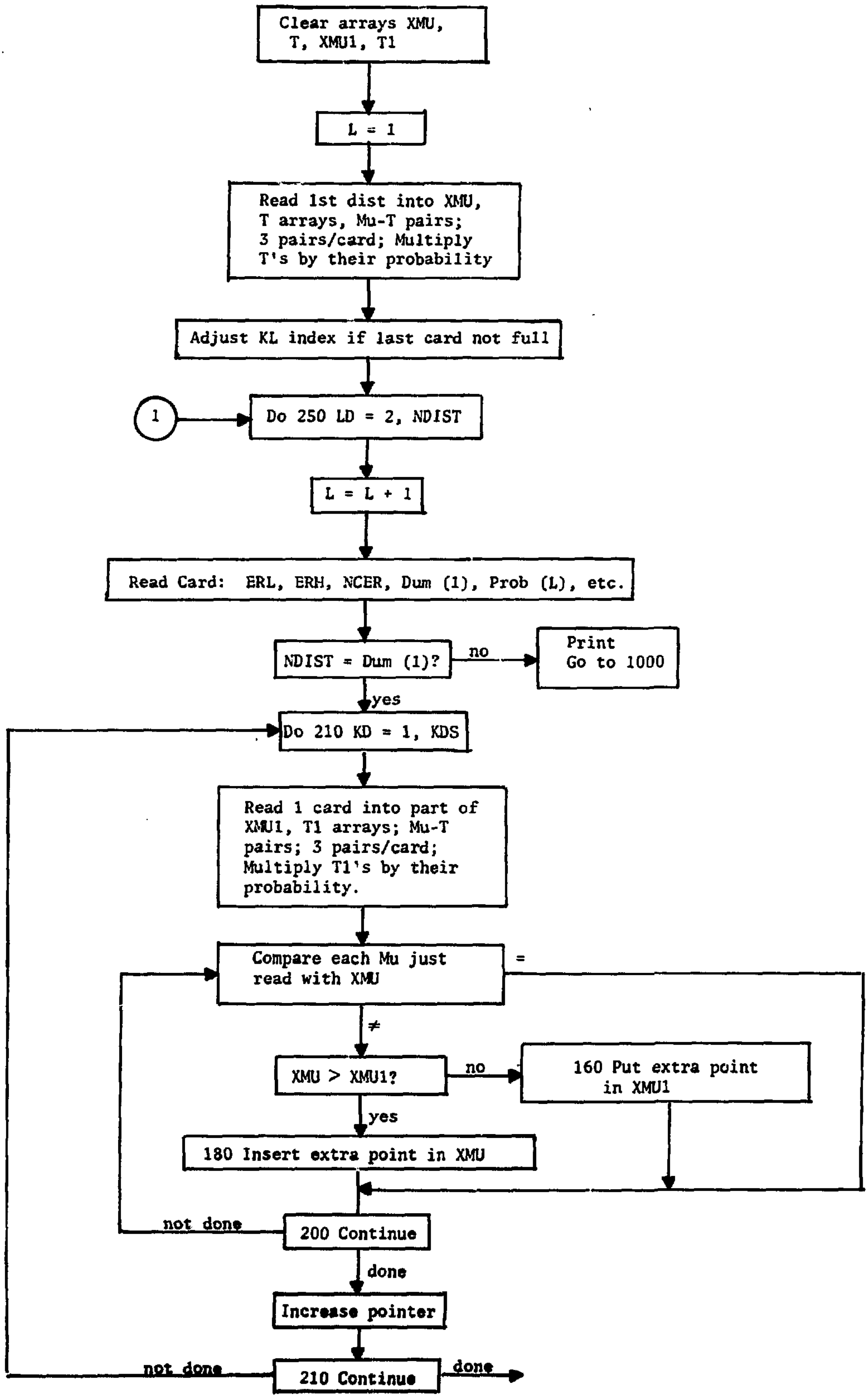


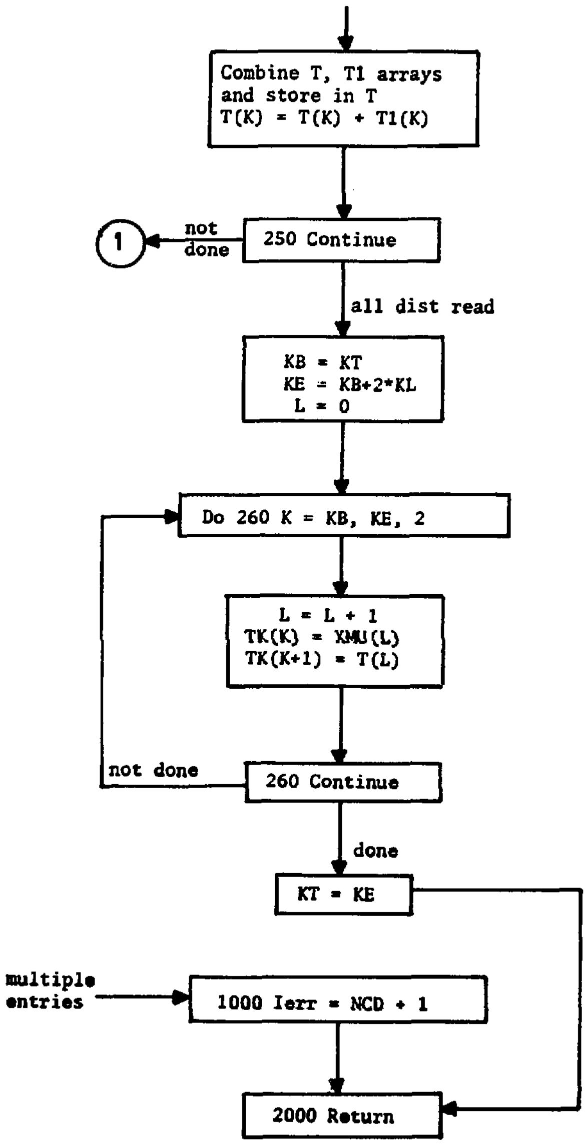
430 LMAX = 0?
yes
ES(KT1) = ERL .
ES(KT1+1) = 0
KT1 = KT1 + 2
433 Is = 1?
Read cards for this L into temporary. E, coefficient pairs; 3 pairs/card
done
Adjust KT1 pointer if last card not full
C
L = L+1
no
440 Is L = LMAX?
yes
yes
442 NN = 1?
no
LLL = 1?
no
Print. Skip cards for Nth neutrons
yes
B
not done
445 Continue
done


Set KT1 to point to last E
450 Check 1st energy in temporary against E(1) or E(2)
≠
Print
Go to 1000
=
460 Do 480 K = 1, KT1, 2
Do 470 KK = 1, I
Compare ES(K), E(KK)
=
≠
not done
470 Continue
done
ES(K) → E(I + 1)
I = I + 1
not done
480 Continue
done
KS = 1
D
Do 520 II = IS, I
NF = 0
Do 510 K = KS, KT1, 2
Compare ES(K), E(II)
≠
500 NF = NF + 1
TK(KT+NF) = ES(K+1)
not done
510 Continue
done
Pack lower index
KT = KT + NF + 1

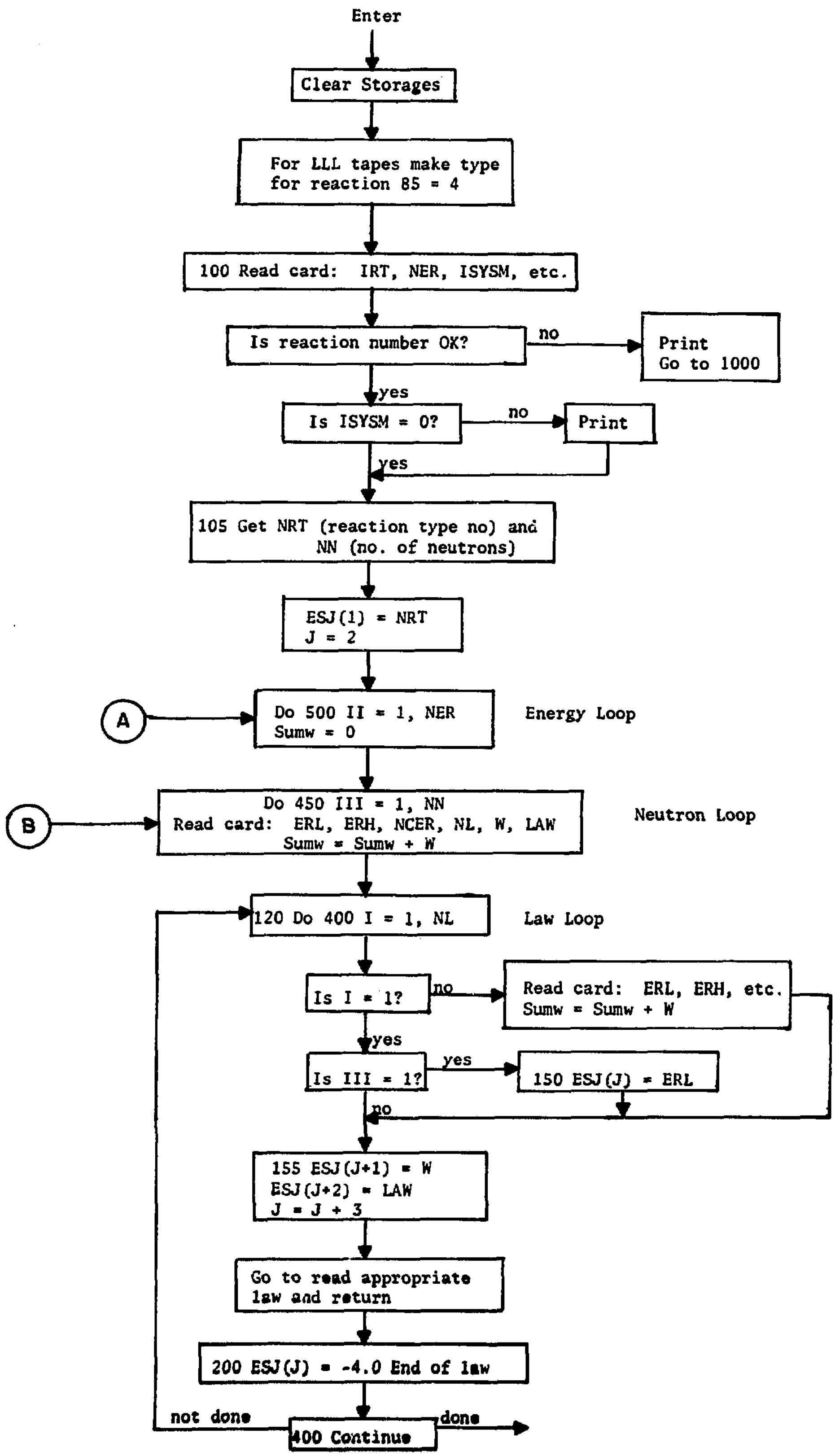

NF = 1?
yes
no
KA = KT
Do 515 KB = 1, NF
TK(KA - KB) = 0?
no
yes
514 KT = KT - 1
NF = NF - 1
not done
515 Continue
done
519 KS = KS + 2
Pack upper index
NF→TK(KT - NF - 1)
D
not done
520 Continue
done
Restore ES block from ECS
LEGN = I
E

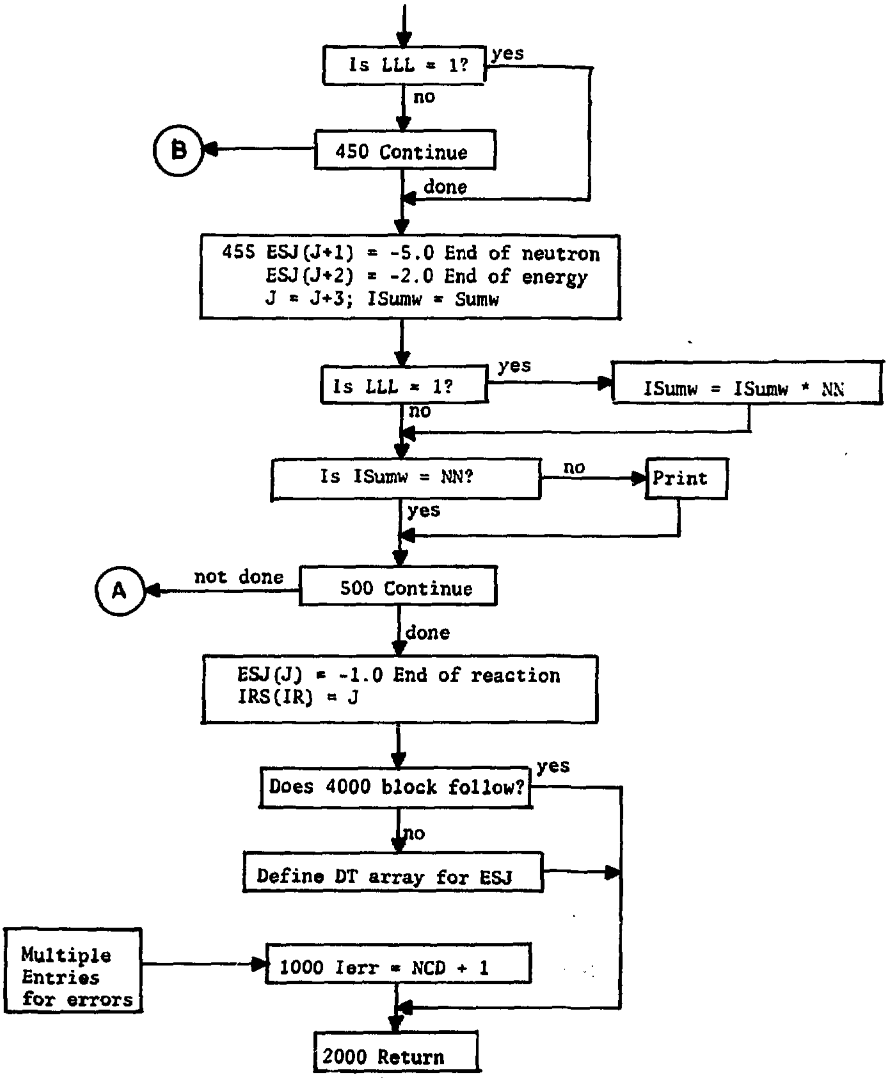
SUBROUTINE MULTI




Combine T, T1 arrays
and store in T
T(K) = T(K) + T1(K)
1
not
done
250 Continue
all dist read
KB = KT
KE = KB+2*KL
L = 0
Do 260 K = KB, KE, 2
L = L + 1
TK(K) = XMU(L)
TK(K+1) = T(L)
not done
260 Continue
done
KT = KE
multiple
entries
1000 Ierr = NCD + 1
2000 Return

SUBROUTINE READSEC

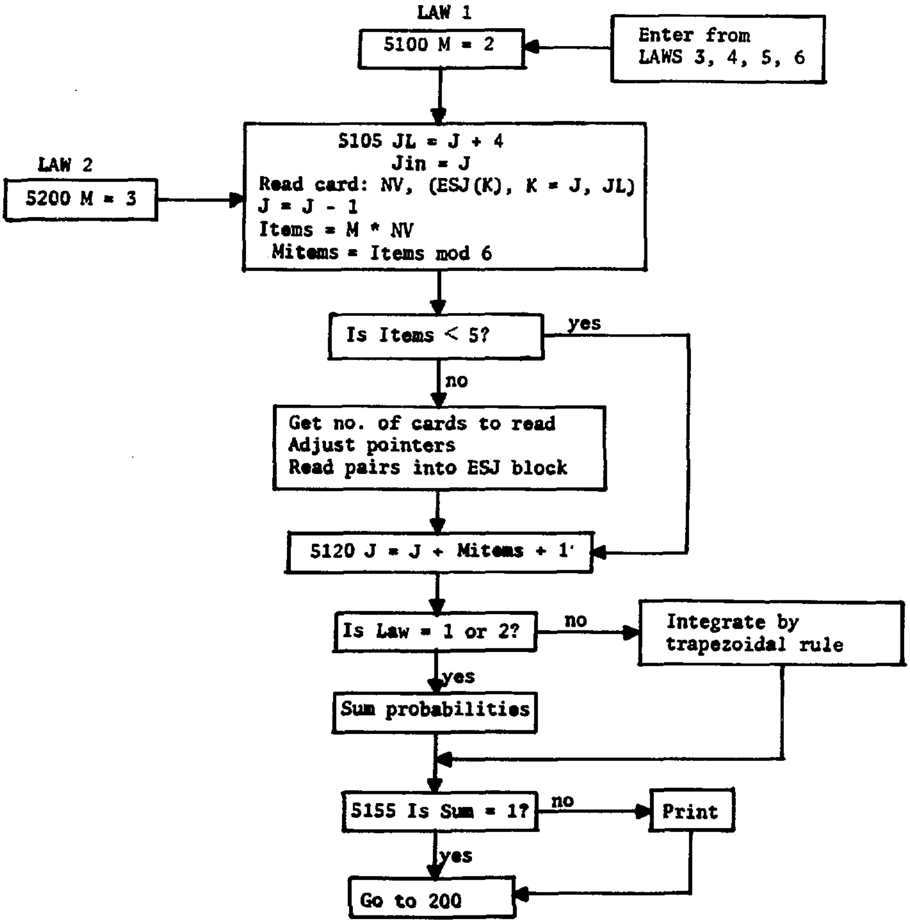




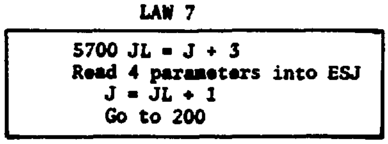
Is LLL = 1?
yes
no
450 Continue
B
done
455 ESJ(J+1) = -5.0 End of neutron
ESJ(J+2) = -2.0 End of energy
J = J+3; ISumw = Sumw
Is LLL = 1?
yes
ISumw = ISumw * NN
no
Is ISumw = NN?
no
Print
yes
A
not done
500 Continue
done
ESJ(J) = -1.0 End of reaction
IRS(IR) = J
Does 4000 block follow?
yes
no
Define DT array for ESJ
Multiple Entries for errors
1000 Ierr = NCD + 1
2000 Return

LAWS 1-6

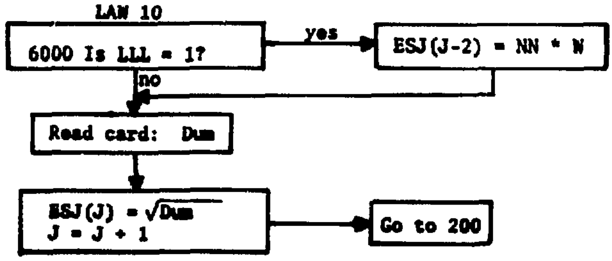
Enter all laws with J pointing to first unused word in ESJ



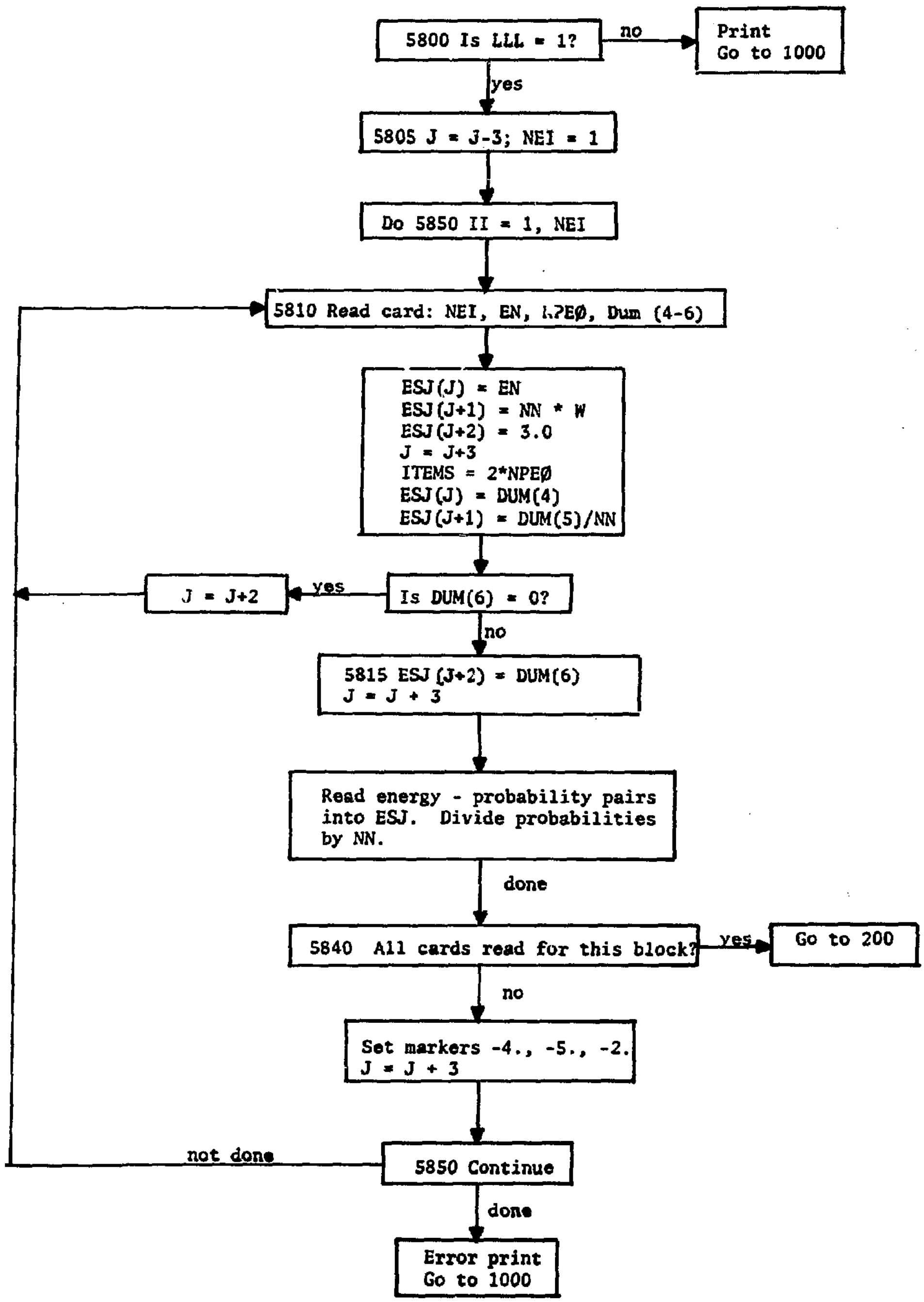
LAW 7



LAW 10

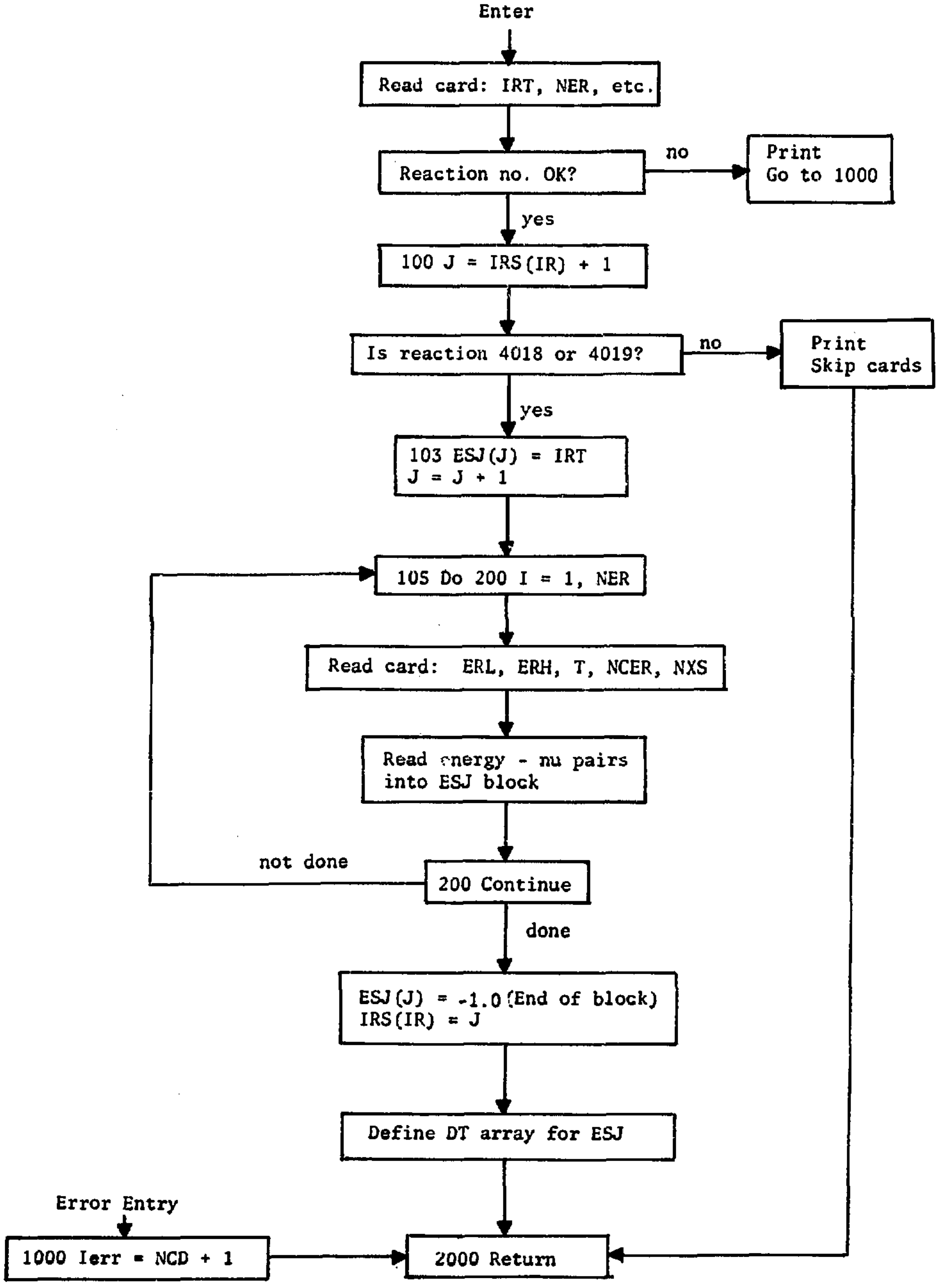


LAW 8

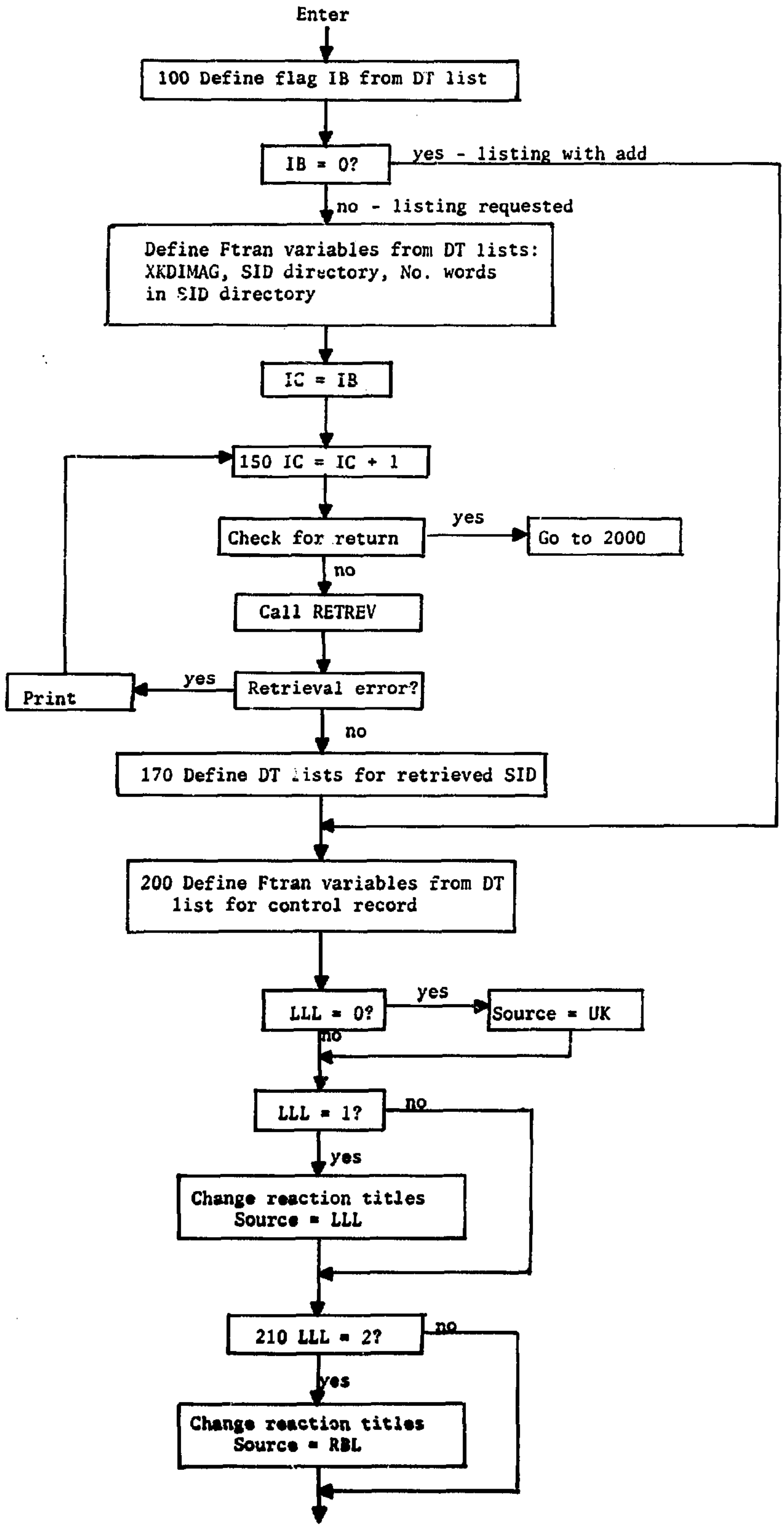


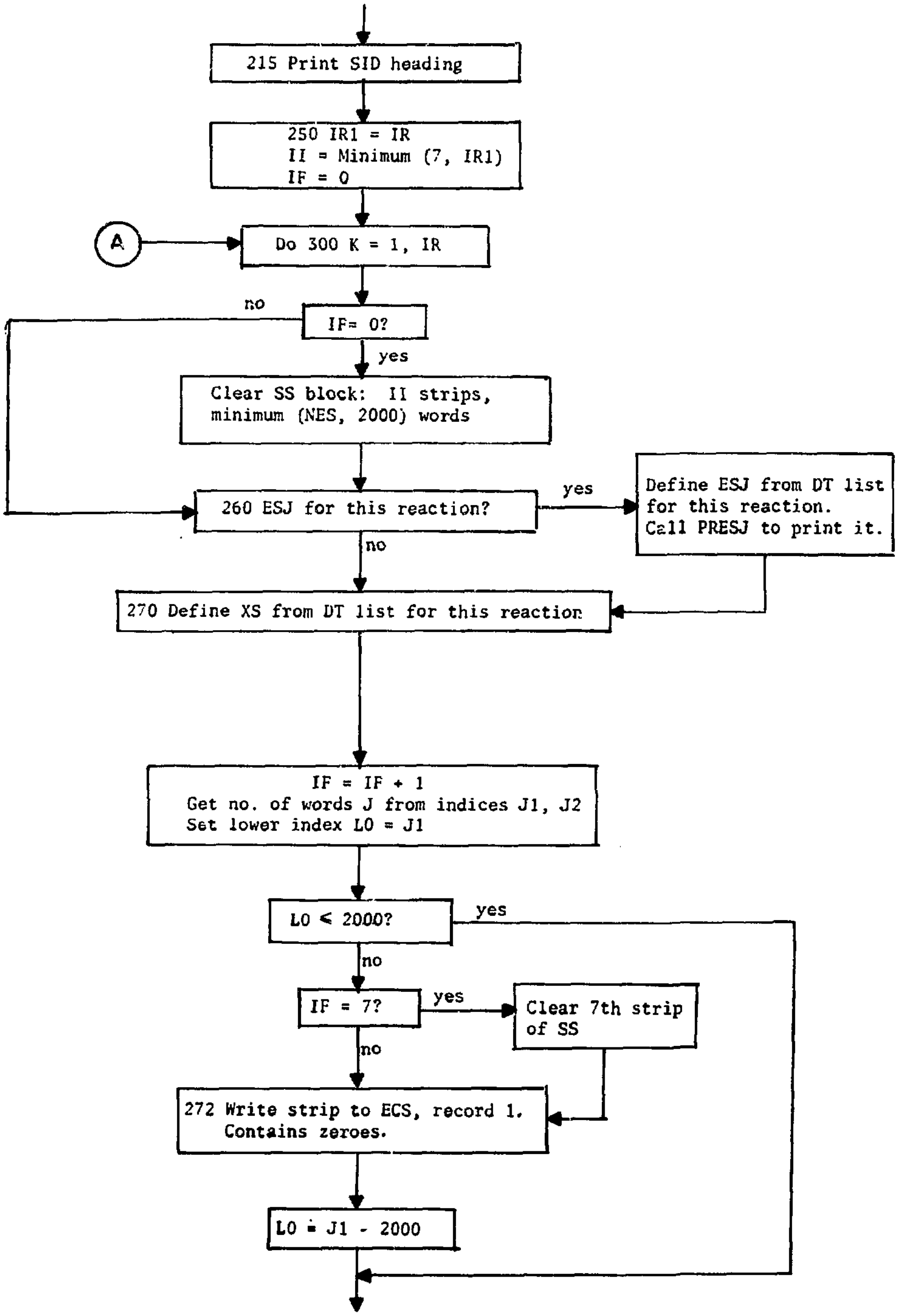
5800 Is LLL = 1?
no
Print
Go to 1000
yes
5805 J = J-3; NEI = 1
Do 5850 II = 1, NEI
5810 Read card: NEI, EN, NPEØ, Dum (4-6)
ESJ(J) = EN
ESJ(J+1) = NN * W
ESJ(J+2) = 3.0
J = J+3
ITEMS = 2*NPEØ
ESJ(J) = DUM(4)
ESJ(J+1) = DUM(5)/NN
J = J+2
yes
Is DUM(6) = 0?
no
5815 ESJ(J+2) = DUM(6)
J = J + 3
Read energy - probability pairs into ESJ. Divide probabilities by NN.
done
5840 All cards read for this block?
yes
Go to 200
no
Set markers -4., -5., -2.
J = J + 3
not done
5850 Continue
done
Error print
Go to 1000

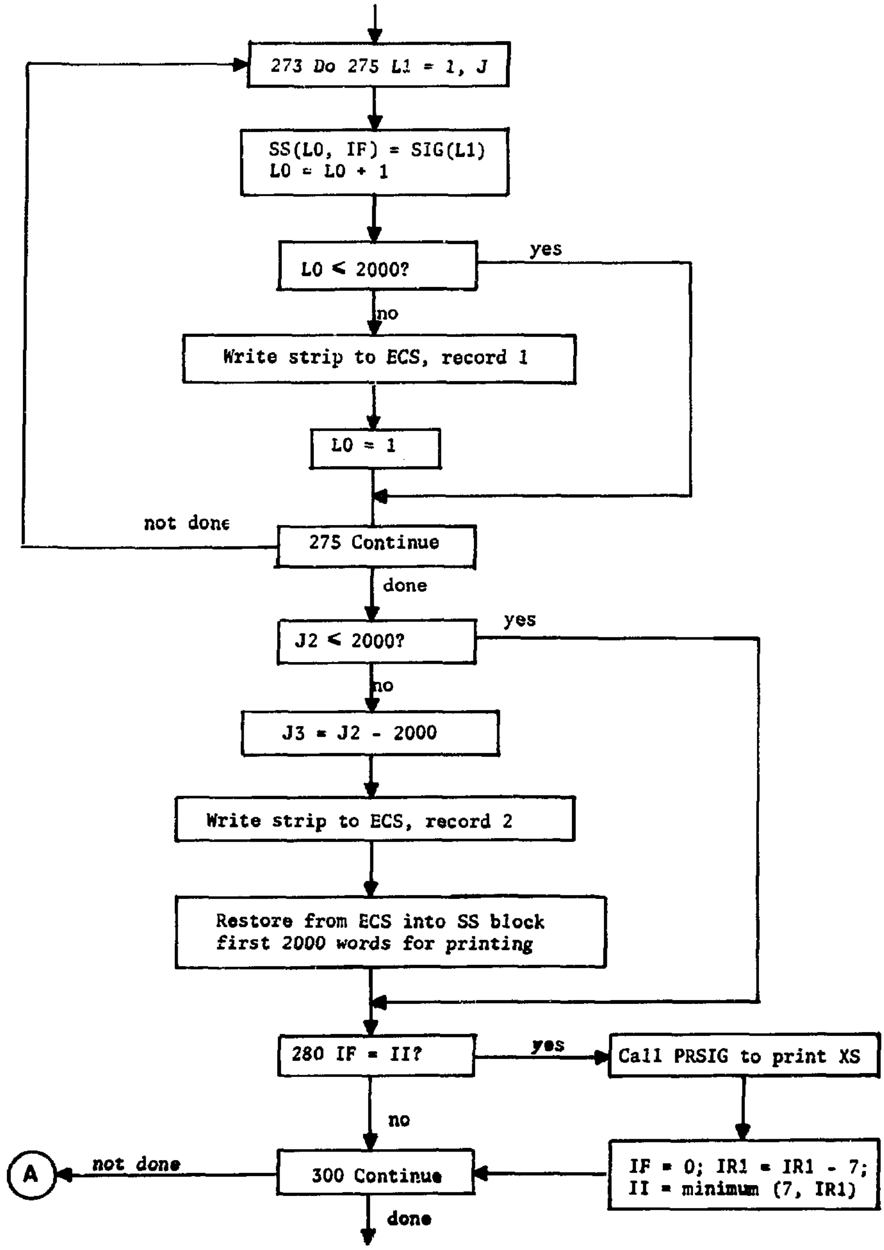
SUBROUTINE READNU

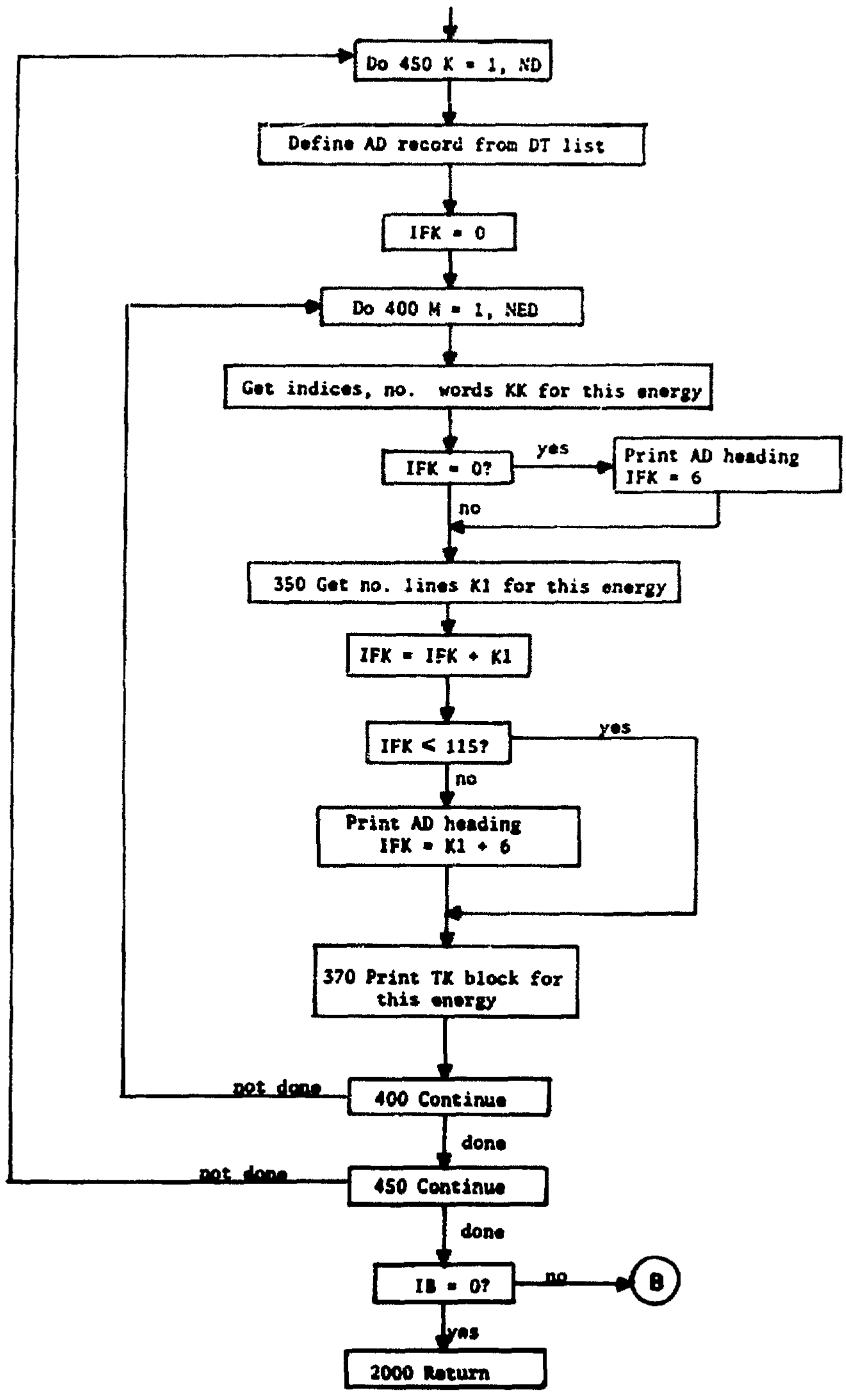


MODULE LISTXS

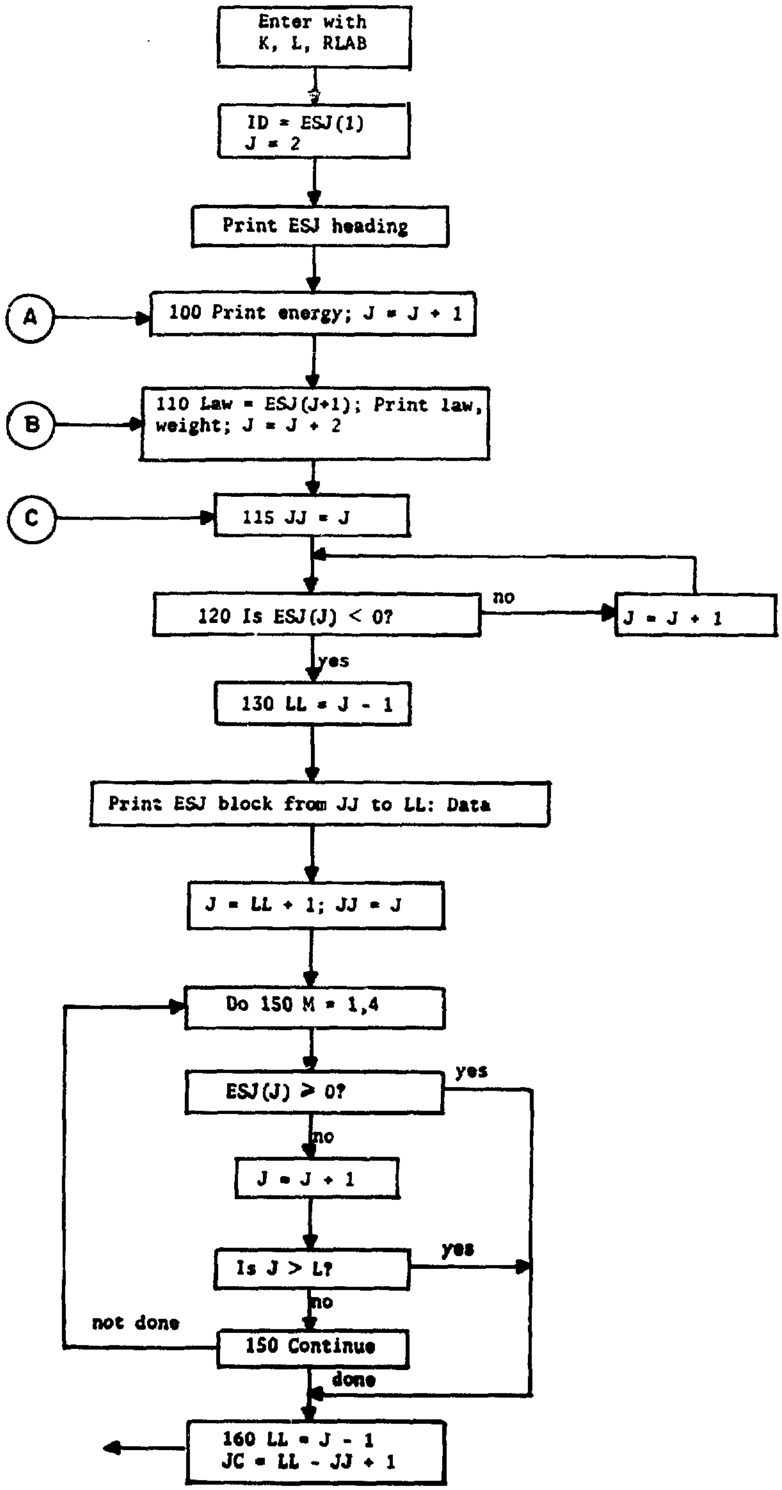


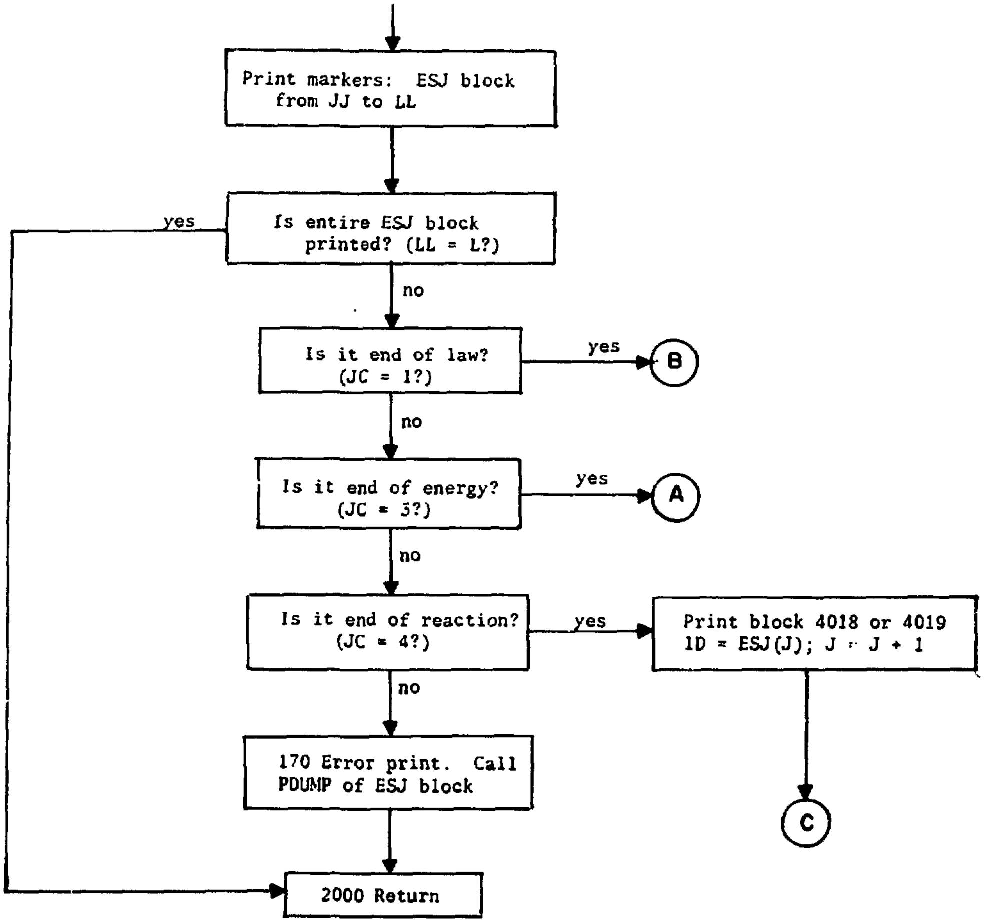

215 Print SID heading
250 IR1 = IR
II = Minimum (7, IR1)
IF = 0
A
Do 300 K = 1, IR
IF= 0?
no
yes
Clear SS block: II strips, minimum (NES, 2000) words
260 ESJ for this reaction?
yes
Define ESJ from DT list for this reaction. Call PRESJ to print it.
no
270 Define XS from DT list for this reaction
IF = IF + 1
Get no. of words J from indices J1, J2
Set lower index L0 = J1
L0 < 2000?
yes
no
IF = 7?
yes
Clear 7th strip of SS
no
272 Write strip to ECS, record 1.
Contains zeroes.
L0 = J1 - 2000


273 Do 275 L1 = 1, J
SS(L0, IF) = SIG(L1)
L0 = L0 + 1
L0 < 2000?
yes
no
Write strip to ECS, record 1
L0 = 1
not done
275 Continue
done
J2 < 2000?
yes
no
J3 = J2 - 2000
Write strip to ECS, record 2
Restore from ECS into SS block
first 2000 words for printing
280 IF = II?
yes
Call PRSIG to print XS
no
IF = 0; IR1 = IR1 - 7;
II = minimum (7, IR1)
A
not done
300 Continue
done


Do 450 K = 1, ND
Define AD record from DT list
IFK = 0
Do 400 M = 1, NED
Get indices, no. words KK for this energy
IFK = 0?
yes
Print AD heading
IFK = 6
no
350 Get no. lines K1 for this energy
IFK = IFK + K1
IFK ≤ 115?
yes
no
Print AD heading
IFK = K1 + 6
370 Print TK block for this energy
not done
400 Continue
done
not done
450 Continue
done
IB = 0?
no
B
yes
2000 Return

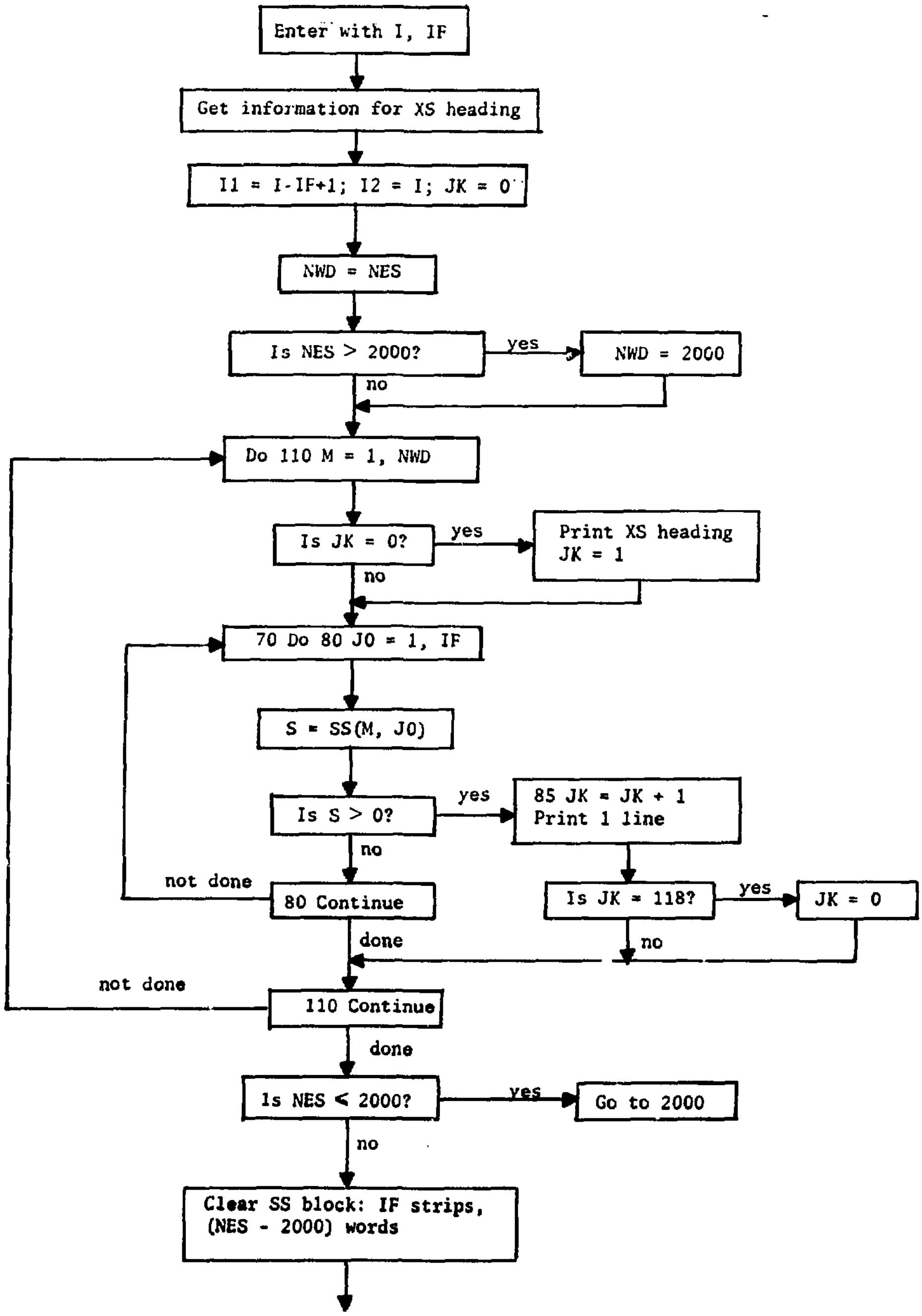
SUBROUTINE PRESJ

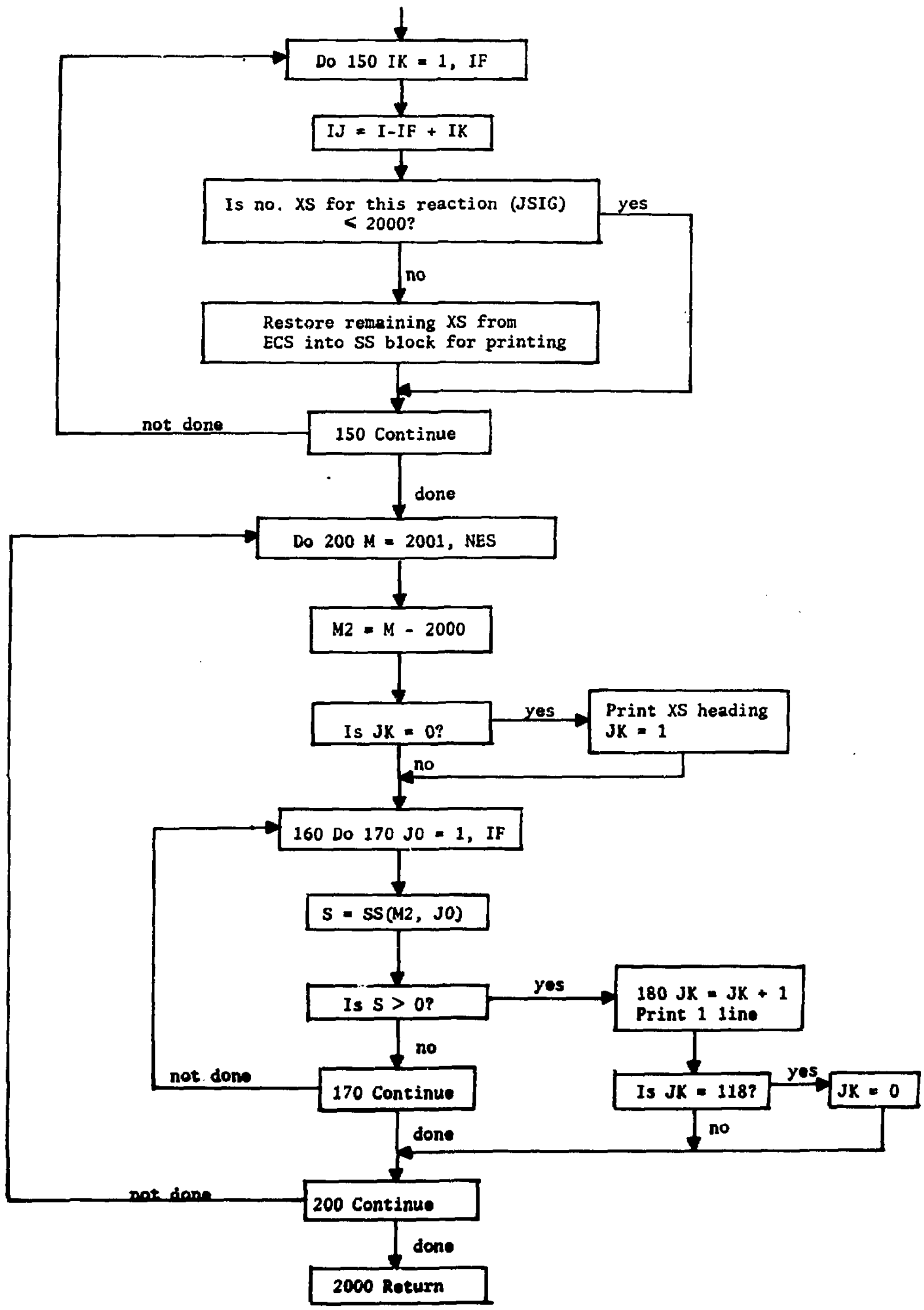




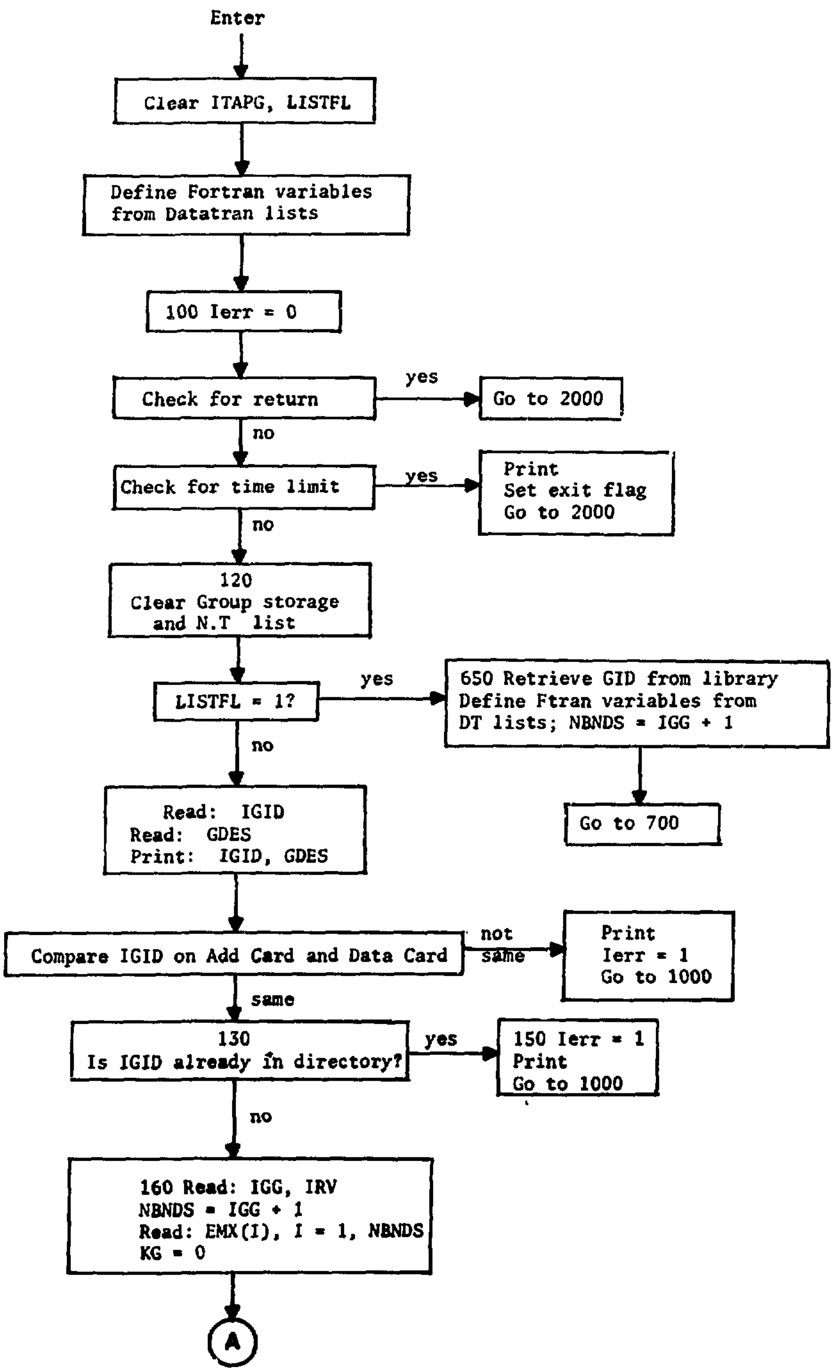
Print markers: ESJ block from JJ to LL
Is entire ESJ block printed? (LL = L?)
yes
no
Is it end of law? (JC = 1?)
yes
B
no
Is it end of energy? (JC = 3?)
yes
A
no
Is it end of reaction? (JC = 4?)
yes
Print block 4018 or 4019 ID = ESJ(J); J = J + 1
C
no
170 Error print. Call PDUMP of ESJ block
2000 Return

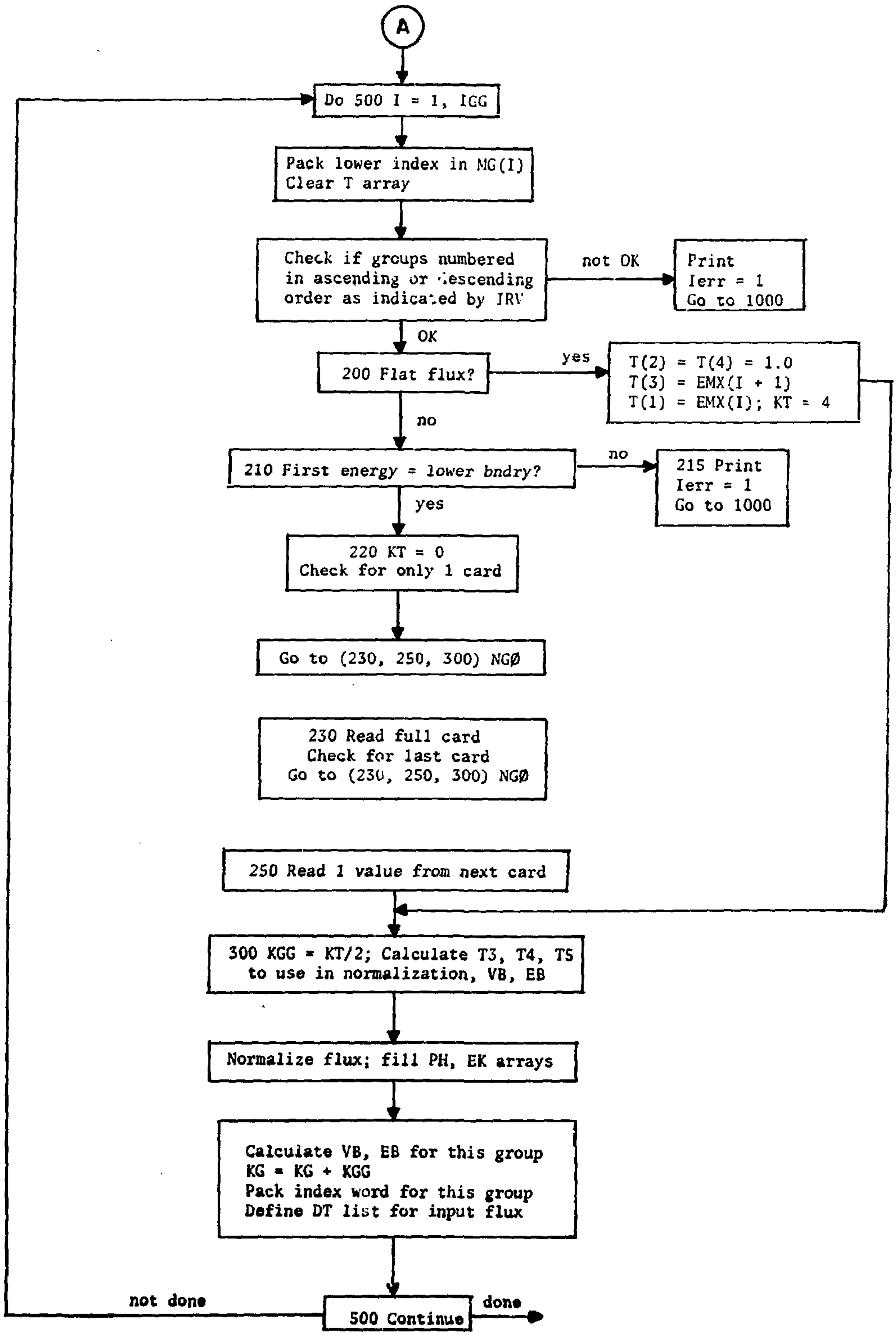
SUBROUTINE PRSIG



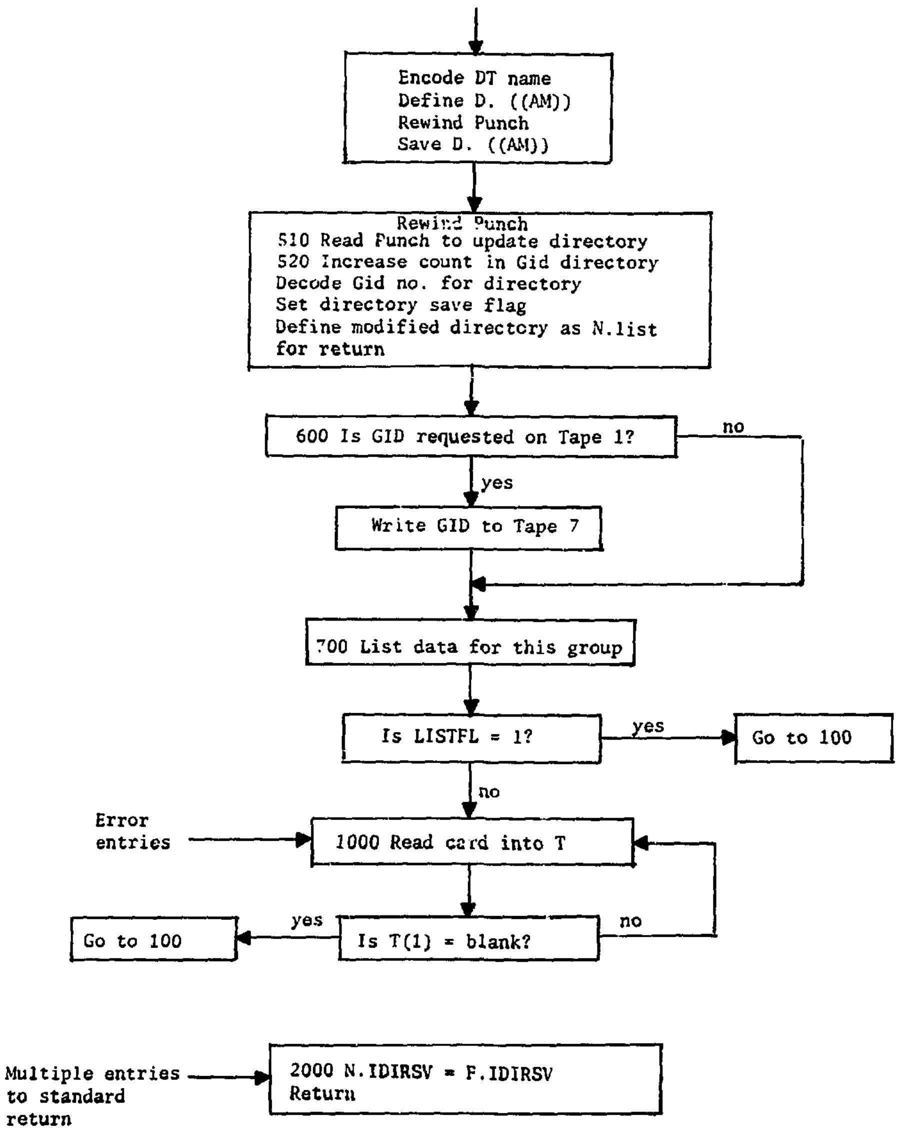

Do 150 IK = 1, IF
IJ = I-IF + IK
Is no. XS for this reaction (JSIG) < 2000?
yes
no
Restore remaining XS from ECS into SS block for printing
not done
150 Continue
done
Do 200 M = 2001, NES
M2 = M - 2000
Is JK = 0?
yes
Print XS heading JK = 1
no
160 Do 170 J0 = 1, IF
S = SS(M2, J0)
Is S > 0?
yes
180 JK = JK + 1 Print 1 line
no
not done
170 Continue
Is JK = 118?
yes
JK = 0
done
no
not done
200 Continue
done
2000 Return

MODULE ADDG

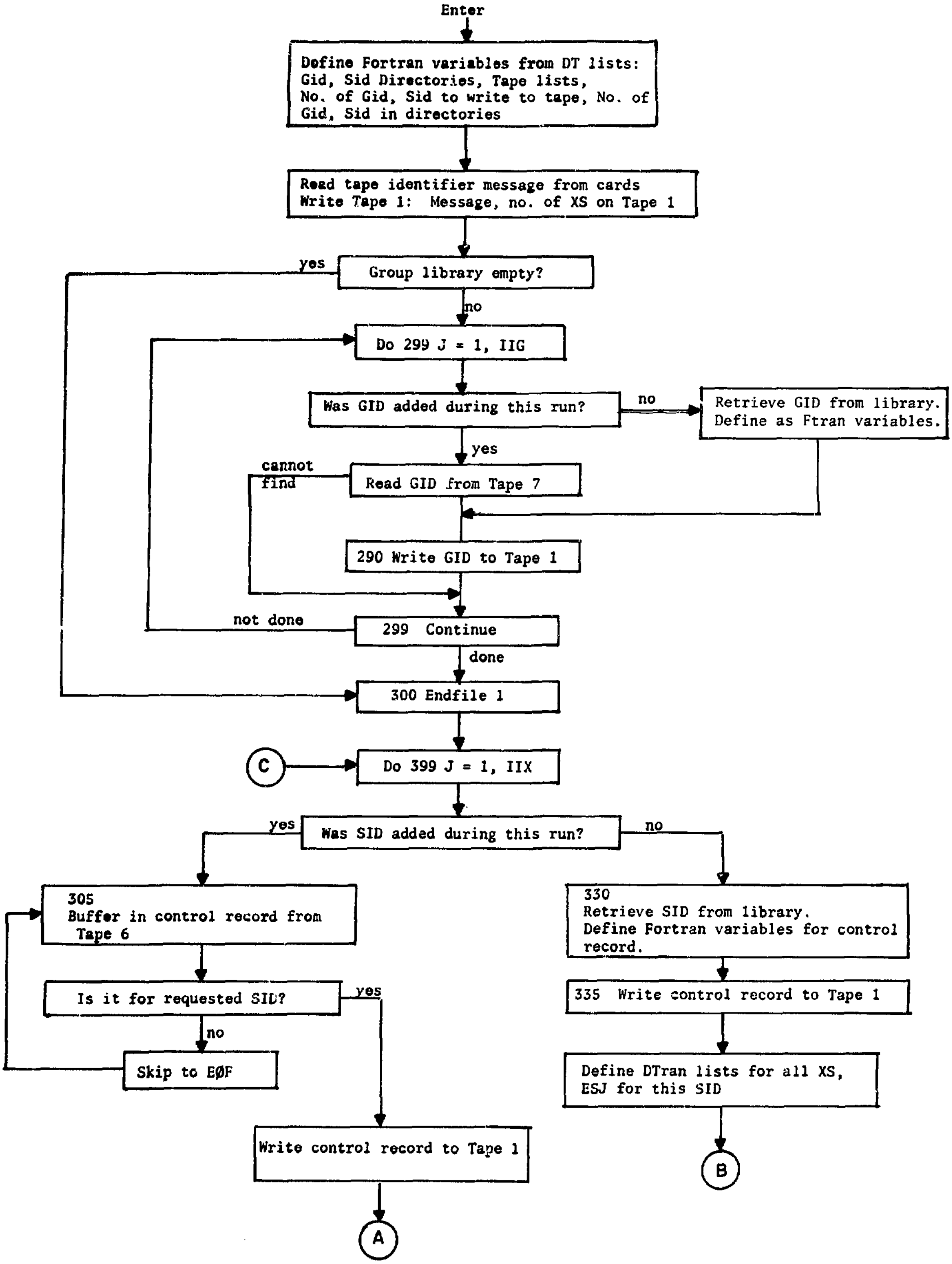

Enter
Clear ITAPG, LISTFL
Define Fortran variables from Datatran lists
100 Ierr = 0
Check for return
yes
Go to 2000
no
Check for time limit
yes
Print
Set exit flag
Go to 2000
no
120
Clear Group storage and N.T list
LISTFL = 1?
yes
650 Retrieve GID from library
Define Ftran variables from DT lists; NBNDS = IGG + 1
Go to 700
no
Read: IGID
Read: GDES
Print: IGID, GDES
Compare IGID on Add Card and Data Card
not same
Print
Ierr = 1
Go to 1000
same
130
Is IGID already in directory?
yes
150 Ierr = 1
Print
Go to 1000
no
160 Read: IGG, IRV
NBNDS = IGG + 1
Read: EMX(I), I = 1, NBNDS
KG = 0
A



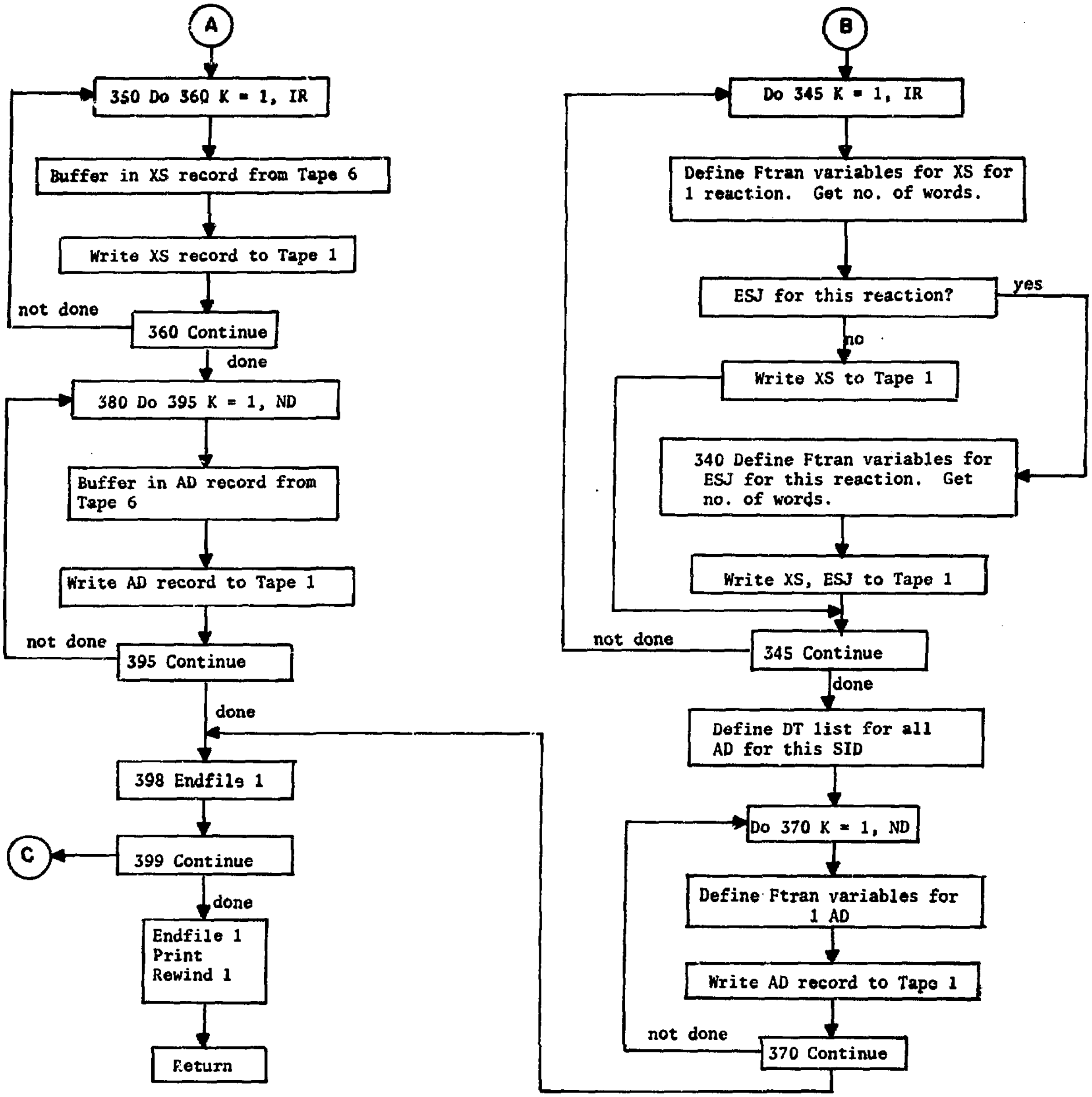
A
Do 500 I = 1, IGG
Pack lower index in MG(I)
Clear T array
Check if groups numbered in ascending or descending order as indicated by IRV
not OK
Print
Ierr = 1
Go to 1000
OK
200 Flat flux?
yes
T(2) = T(4) = 1.0
T(3) = EMX(I + 1)
T(1) = EMX(I); KT = 4
no
210 First energy = lower bndry?
no
215 Print
Ierr = 1
Go to 1000
yes
220 KT = 0
Check for only 1 card
Go to (230, 250, 300) NGØ
230 Read full card
Check for last card
Go to (230, 250, 300) NGØ
250 Read 1 value from next card
300 KGG = KT/2; Calculate T3, T4, T5 to use in normalization, VB, EB
Normalize flux; fill PH, EK arrays
Calculate VB, EB for this group
KG = KG + KGG
Pack index word for this group
Define DT list for input flux
not done
500 Continue
done


Encode DT name
Define D. ((AM))
Rewind Punch
Save D. ((AM))
Rewind Punch
510 Read Punch to update directory
520 Increase count in Gid directory
Decode Gid no. for directory
Set directory save flag
Define modified directory as N.list for return
600 Is GID requested on Tape 1?
no
yes
Write GID to Tape 7
700 List data for this group
Is LISTFL = 1?
yes
Go to 100
no
Error entries
1000 Read card into T
yes
Is T(1) = blank?
no
Go to 100
Multiple entries to standard return
2000 N.IDIRSV = F.IDIRSV
Return

MØDULE TAPE



Enter
Define Fortran variables from DT lists: Gid, Sid Directories, Tape lists, No. of Gid, Sid to write to tape, No. of Gid, Sid in directories
Read tape identifier message from cards Write Tape 1: Message, no. of XS on Tape 1
Group library empty?
yes
no
Do 299 J = 1, IIG
Was GID added during this run?
no
Retrieve GID from library. Define as Ftran variables.
yes
Read GID from Tape 7
cannot find
290 Write GID to Tape 1
299 Continue
not done
done
300 Endfile 1
C
Do 399 J = 1, IIX
Was SID added during this run?
yes
no
305 Buffer in control record from Tape 6
Is it for requested SID?
yes
no
Skip to EØF
Write control record to Tape 1
A
330 Retrieve SID from library. Define Fortran variables for control record.
335 Write control record to Tape 1
Define DTran lists for all XS, ESJ for this SID
B



A
350 Do 360 K = 1, IR
Buffer in XS record from Tape 6
Write XS record to Tape 1
not done
360 Continue
done
380 Do 395 K = 1, ND
Buffer in AD record from Tape 6
Write AD record to Tape 1
not done
395 Continue
done
398 Endfile 1
C
399 Continue
done
Endfile 1
Print
Rewind 1
Return
B
Do 345 K = 1, IR
Define Ftran variables for XS for 1 reaction. Get no. of words.
ESJ for this reaction?
yes
no
Write XS to Tape 1
340 Define Ftran variables for ESJ for this reaction. Get no. of words.
Write XS, ESJ to Tape 1
not done
345 Continue
done
Define DT list for all AD for this SID
Do 370 K = 1, ND
Define Ftran variables for 1 AD
Write AD record to Tape 1
not done
370 Continue

APPENDIX B

CODE LISTING

LASL Identification: LP-0287

```
          JOB MARTY                                                              XSMOD     2
          PROGRAM XSMOD(INPUT,OUTPUT,PUNCH,TAPE1,TAPE2,TAPE18=OUTPUT,TAPE10=     XSMOD     3
     1        PUNCH)                                                             XSMOD     4
C         THE DIRECTORY RECALL CARD IS FOR TAPE30=LB204 RECALL=67                LB204     1
C                                                                                XSMOD     6
C                                                                                XSMOD     7
C         *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *                      XSMOD     8
C                                                                                XSMOD     9
C                               PROGRAM XSMOD                                    XSMOD    10
C                                                                                XSMOD    11
C         THIS PROGRAM WAS WRITTEN FOR THE CDC6600 BY MARTHA S. HOYT AT THE      XSMOD    12
C         LOS ALAMOS SCIENTIFIC LABORATORY, LOS ALAMOS, NEW MEXICO               XSMOD    13
C                                                                                XSMOD    14
C                               AUGUST 1973                                      XSMOD    15
C                                                                                XSMOD    16
C         *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *  *                      XSMOD    17
C                                                                                XSMOD    18
C                                                                                XSMOD    19
C         TAPE30   DATATRAN TAPE                                                 XSMOD    20
C         TAPE1    FILE TO BE USED FOR EVXS.                                     XSMOD    21
C         TAPE2    UK OR LRLUK INPUT TAPE FOR ADDXS.                             XSMOD    22
C         KEYWORDS                                                               XSMOD    23
C         ADD      ADD A DATASET,MAKE DTRAN TAPE,LIST.                           XSMOD    24
C         DELETE   DELETE A DATASET,MAKE DTRAN TAPE.                             XSMOD    25
C         EVXS     EVXS PROCESSING REQUESTED.                                    XSMOD    26
C         LIST     LIST A DATASET.                                               XSMOD    27
C         CONTENTS PRINT GID,SID CURRENTLY IN DATATRAN DIRECTORY                 XSMOD    28
C         TAPE ALL WRITE ALL GID,SID TO TAPE1                                    XSMOD    29
C         TAPE     WRITE A DATASET TO TAPE1                                      XSMOD    30
C         LAST     LAST INPUT CARD.                                              XSMOD    31
C                                                                                XSMOD    32
          COMMON   /CDID/IA7,IA8,IA9,IBLNO(100),IDB(100),ICDS(100),DUM(9),LLL    CDID      2
          COMMON/DIR/  KGID(300),KSID(300),IDIR,IIG,IIX,IDIRSV                   DIR       2
          DIMENSION  IDNOG(100),NWDSG(100),IDNOS(100),NWDSS(100)                 DIR       3
          EQUIVALENCE  (KGID(101),IDNOG(1)),(KGID(201),NWDSG(1)),                DIR       4
     1                 (KSID(101),IDNOS(1)),(KSID(201),NWDSS(1))                 DIR       5
          COMMON   /OPTS/ KDIMAG(1100),N(2),IALL,IB                              OPTS      2
          DIMENSION  XKDIMAG(1100)                                               OPTS      3
          EQUIVALENCE  (KDIMAG,XKDIMAG)                                          OPTS      4
          DIMENSION  M(15),XM(15)                                                XSMOD    36
          EQUIVALENCE  (M,XM)                                                    XSMOD    37
C                                                                                XSMOD    38
          DATA  (M(I),I=1,15) /10HCHANGE SID, 10HCHANGE GID, 10HEVXS      ,      XSMOD    39
     1     10HPUNCH GID , 10HADD SID   , 10HADD GID   , 10HDELETE SID,           XSMOD    40
     2     10HDELETE GID, 10HLIST SID  , 10HLIST GID  , 10HCONTENTS  ,           XSMOD    41
     3     10HTAPE ALL  , 10HTAPE SID  , 10HTAPE GID  , 10HLAST      /           XSMOD    42
C                                                                                XSMOD    43
          IEXIT=0                                                                XSMOD    44
          CALL CPAREA(KDIMAG)                                                    XSMOD    45
          ITL=KDIMAG(19) .AND. 77777700000000B                                   XSMOD    46
          TL=ITL/100000000B                                                      XSMOD    47
          IC=LOCF(IB)-LOCF(IBLNO(1))+1                                           XSMOD    48
          DO 100 I=1,IC                                                          XSMOD    49
  100     IBLNO(I)=0                                                             XSMOD    50
          ITP=0                                                                  XSMOD    51
          PRINT 8                                                                XSMOD    52
    8     FORMAT(1H1,43X,*+ + + +   CROSS SECTION LIBRARY MODIFICATION CODE  + + +  XSMOD  53
     1    + *//)                                                                 XSMOD    54
          RECALL N. IMAGES            ERASE IN-1-DAY 21973WRD RMHX1     2        XSMOD    55
```

```
C         RECALLS DIRECTORY TO DATATRAN TAPE                                      XSMOD    56
C         THIS DIRECTORY RECALL CARD MUST BE CHANGED IF A DIFFERENT TAPE30        XSMOD    57
C         IS USED.                                                                XSMOD    58
          RECALL D. DIRECT              ERASE IN-1-DAY  1533WRD RMHX1    67       LB204     2
C         DEFINE FORTRAN VARIABLES FROM RECALLED DIRECTORY.                       XSMOD    60
          F. KGID=N. DIRGID. 1/                                                   XSMOD    61
          F. IDNOG=N. DIRGID. 2/                                                  XSMOD    62
          F. NWDSG=N. DIRGID. 3/                                                  XSMOD    63
          F. KSID=N. DIRSID. 1/                                                   XSMOD    64
          F. IDNOS=N. DIRSID. 2/                                                  XSMOD    65
          F. NWDSS=N. DIRSID. 3/                                                  XSMOD    66
          F. IIG=WORDS OF N. DIRGID. 1/                                           XSMOD    67
          F. IIX=WORDS OF N. DIRSID. 1/                                           XSMOD    68
          F. IDIR=N. IID                                                          XSMOD    69
C         INITIALIZE N. IDIRSV                                                    XSMOD    70
          N. IDIRSV=0                                                             XSMOD    71
C                                                                                 XSMOD    72
C         READ IN OPTIONS AND THEIR IDS.                                          XSMOD    73
C                                                                                 XSMOD    74
C         PREPARE LISTS FOR MODULE RDJOBS                                         XSMOD    75
C                                                                                 XSMOD    76
          N. RDJOBS=F. XM                                                         XSMOD    77
          KALL RDJOBS                                                             XSMOD    78
C                                                                                 XSMOD    79
C         2ND PASS.  PROCESS OPTIONS.                                             XSMOD    80
C                                                                                 XSMOD    81
C         DEFINE FORTRAN VARIABLES FROM RDJOBS OUTPUT LIST                        XSMOD    82
          F. XKDIMAG=N. RDJOUT. 1/                                                XSMOD    83
          F. N=N. RDJOUT. 2/                                                      XSMOD    84
          F. IALL=N. RDJOUT. 2/. 3                                                XSMOD    85
  199     IB=1                                                                    XSMOD    86
  200     DO 205 I=1,15                                                           XSMOD    87
          IF(KDIMAG(IB) . EQ.  M(I))GO TO 220                                     XSMOD    88
  205     CONTINUE                                                                XSMOD    89
  210     IB=IB+ 11                                                               XSMOD    90
          CALL SECOND(CP)                                                         XSMOD    91
          IF(TL- CP- 5. 0)215,215,200                                             XSMOD    92
  215     IEXIT=1                                                                 XSMOD    93
          PRINT 10, TL,CP                                                         XSMOD    94
   10     FORMAT(* LESS THAN 5 SEC LEFT.  EXIT FLAG SET.  TL=*F10. 3,5X*CP=*      XSMOD    95
    1        F10. 3)                                                              XSMOD    96
          GO TO 1000                                                              XSMOD    97
  220     GO TO(500,550,600,650,700,750,800,850,900,950,960,970,970,970,          XSMOD    98
    1        1000)I                                                               XSMOD    99
  500     CALL CHANGE                                                             XSMOD   100
          GO TO 210                                                               XSMOD   101
  550     CALL CHANGE                                                             XSMOD   102
          GO TO 210                                                               XSMOD   103
  600     IEVXS=1                                                                 XSMOD   104
          GO TO 210                                                               XSMOD   105
  650     GO TO 210                                                               XSMOD   106
          GO TO 210                                                               XSMOD   107
C                                                                                 XSMOD   108
C         PREPARE LIST FOR MODULE ADDXS                                           XSMOD   109
  700     N. ADDXS=F. IB,F. TL,F. IEXIT                                           XSMOD   110
          CALL SECOND(CP)                                                         XSMOD   111
          IF(TL- CP- 15. 0)705,705,710                                            XSMOD   112
  705     IEXIT=1                                                                 XSMOD   113
          PRINT 12, TL,CP                                                         XSMOD   114
   12     FORMAT(* LESS THAN 15 SEC LEFT AND NEXT JOB IS ADDXS. EXIT FLAG SET     XSMOD   115
    1      TL=*F10. 3,5X*CP=*F10. 3)                                              XSMOD   116
          GO TO 1000                                                              XSMOD   117
  710     KALL ADDXS                                                              XSMOD   118
C         MODIFIED DIRECTORY RETURNED FROM ADDXS                                  XSMOD   119
          F. KSID=N. DIRSID. 1/                                                   XSMOD   120
          F. IDNOS=N. DIRSID. 2/                                                  XSMOD   121
          F. NWDSS=N. DIRSID. 3/                                                  XSMOD   122
          F. IIX=WORDS OF N. DIRSID. 1/                                           XSMOD   123
          F. IDIRSV=N. IDIRSV                                                     XSMOD   124
          F. IEXIT=N. ADDXS. 3                                                    XSMOD   125
          IF(IEXIT . EQ.  1)GO TO 1000                                            XSMOD   126
```

```
         GO TO 210                                                        XSMOD
C                                                                         XSMOD
C        PREPARE LIST FOR MODULE ADDG                                     XSMOD
  750    N. ADDG=F. IB,F. TL,F. IEXIT,0                                   XSMOD
         CALL SECOND(CP)                                                  XSMOD
         IF(TL- CP- 10. 0)760,760,770                                     XSMOD
  760    IEXIT=1                                                          XSMOD
         PRINT 14, TL,CP                                                  XSMOD
   14    FORMAT(* LESS THAN 10 SEC LEFT AND NEXT JOB IS ADDG.  EXIT FLAG SET  XSMOD
    1    . TL=*F10. 3,5X*CP=*F10. 3)                                      XSMOD
         GO TO 1000                                                       XSMOD
  770    KALL ADDG                                                        XSMOD
C        MODIFIED DIRECTORY RETURNED FROM ADDG                            XSMOD
         F. KGID=N. DIRGID. 1/                                            XSMOD
         F. IDNOG=N. DIRGID. 2/                                           XSMOD
         F. NWDSG=N. DIRGID. 3/                                           XSMOD
         F. IIG=WORDS OF N. DIRGID. 1/                                    XSMOD
         F. IDIRSV=N. IDIRSV                                              XSMOD
         F. IEXIT=N. ADDG. 3                                              XSMOD
         IF(IEXIT . EQ. 1)GO TO 1000                                      XSMOD
         GO TO 210                                                        XSMOD
  800    CALL DELETE(KSID,IIX,1)                                          XSMOD
         GO TO 210                                                        XSMOD
  850    CALL DELETE(KGID,IIG,2)                                          XSMOD
         GO TO 210                                                        XSMOD
C        PREPARE LIST FOR MODULE LISTXS                                   XSMOD
  900    N. LISTXS=F. IB                                                  XSMOD
         KALL LISTXS                                                      XSMOD
         GO TO 210                                                        XSMOD
C        PREPARE LIST FOR MODULE ADDG TO LIST A GROUP                     XSMOD
  950    LISTFL=1                                                         XSMOD
         N. ADDG=F. IB,F. TL,F. IEXIT,F. LISTFL                           XSMOD
         KALL ADDG                                                        XSMOD
         GO TO 210                                                        XSMOD
  960    CALL CONTENT                                                     XSMOD
         GO TO 210                                                        XSMOD
C        SET FLAG FOR MAKING TAPE1 FILE. ENTER TAPE MODULE LAST.          XSMOD
  970    ITP=1                                                            XSMOD
         GO TO 210                                                        XSMOD
C        DEFINE AND SAVE DIRECTORY IF MODIFIED                            XSMOD
 1000    IF(IDIRSV . NE. 0)GO TO 1003                                     XSMOD
         IF(IEXIT . EQ. 1)CALL EXIT                                       XSMOD
         GO TO 1005                                                       XSMOD
 1003    IDIR=IDIRSV + 1                                                  XSMOD
         N. IID=F. IDIR                                                   XSMOD
         D. DIRECT=N. DIRGID/N. DIRSID/N. IID                             XSMOD
         PRINT 16                                                         XSMOD
   16    FORMAT(1H1)                                                      XSMOD
         SAVE D. DIRECT                                                   XSMOD
         N. LIST=DIRECT                                                   XSMOD
         KALL IMAGE                                                       XSMOD
         IF(IEXIT . EQ. 1)CALL ABORT                                      XSMOD
 1005    IF(ITP . NE. 1)GO TO 1050                                        XSMOD
         KALL TAPE                                                        XSMOD
         IF(IEVXS . NE. 1)GO TO 1050                                      XSMOD
    2    FORMAT(//* ALL OPTIONS PROCESSED. READY TO BEGIN LINKING. DIRECTORY  XSMOD
    1    RECALL NO. =*I4)                                                 XSMOD
         PRINT 2, IDIR                                                    XSMOD
         ENDFILE 18                                                       XSMOD
         CALL OVERLAY(4HEVXS,0,0)                                         XSMOD
         PRINT 4                                                          XSMOD
    4    FORMAT(//* RETURNED FROM EVXS. END JOB. *)                       XSMOD
         CALL EXIT                                                        XSMOD
 1050    PRINT 6, IDIR                                                    XSMOD
    6    FORMAT(//* END JOB. NO EVXS PROCESSING REQUESTED. DIRECTORY RECALL N  XSMOD
    1    O. =*I4)                                                         XSMOD
         END                                                              XSMOD
         SUBROUTINE CHANGE                                                XSMOD
         PRINT 2                                                          XSMOD
    2    FORMAT(/* ENTERED SUBROUTINE CHANGE. *)                          XSMOD
         RETURN                                                           XSMOD
```

```
      END                                                               XSMOD    198
      SUBROUTINE CONTENT                                                XSMOD    199
C     PRINT CURRENT CONTENTS OF DIRECTORY                               XSMOD    200
C                                                                       XSMOD    201
      COMMON/DIR/  KGID(300),KSID(300),IDIR,IIG,IIX,IDIRSV              DIR        2
      DIMENSION  IDNOG(100),NWDSG(100),IDNOS(100),NWDSS(100)            DIR        3
      EQUIVALENCE  (KGID(101),IDNOG(1)),(KGID(201),NWDSG(1)),           DIR        4
     1             (KSID(101),IDNOS(1)),(KSID(201),NWDSS(1))            DIR        5
C                                                                       XSMOD    203
      PRINT 2                                                           XSMOD    204
    2 FORMAT(/*1CONTENTS OF DATATRAN TAPE DIRECTORY*/)                  XSMOD    205
      PRINT 4, (KGID(I),I=1,IIG)                                        XSMOD    206
    4 FORMAT(*  GID*/(12I10))                                           XSMOD    207
      PRINT 6, (KSID(I),I=1,IIX)                                        XSMOD    208
    6 FORMAT(//*  SID*/(12I10))                                         XSMOD    209
      PRINT 8, IDIR,IDIRSV                                              XSMOD    210
    8 FORMAT(//*  RECALL ID NO. OF DIRECTORY *I3*  DIRECTORY SAVE FLAG= XSMOD    211
     1*I4//* IF SAVE FLAG=0 DIRECTORY ID OK. OTHERWISE DIRECTORY IS BEIN XSMOD    212
     2G UPDATED AND ID NO. NOT AVAILABLE UNTIL END OF JOB.*)            XSMOD    213
      RETURN                                                            XSMOD    214
      END                                                               XSMOD    215
      SUBROUTINE DELETE(KDIR,IIK,JJ)                                    XSMOD    216
C     DELETES FROM DIRECTORY SPECIFIED KIDS AND THEIR ASSOCIATED RECALL XSMOD    217
C     NOS. AND NOS. OF WORDS. CLOSES UP DIRECTORY.                      XSMOD    218
C     PRINTS OUT INFORMATION NEEDED FOR A DTPURGE JOB.                  XSMOD    219
C     IB        INDEX TO KDIMAG                                         XSMOD    220
C     JJ        =1,DELETING SID.  =2,DELETING GID                       XSMOD    221
C     KDIR       DIRECTORY FROM WHICH DELETE IS BEING MADE.             XSMOD    222
C     IIK       NO. OF ITEMS IN DIRECTORY.                              XSMOD    223
C                                                                       XSMOD    224
      COMMON  /DIR/  KGID(300),KSID(300),IDIR,IIG,IIX,IDIRSV            XSMOD    225
      COMMON  /OPTS/ KDIMAG(1100),N(2),IALL,IB                          OPTS       2
      DIMENSION  XKDIMAG(1100)                                          OPTS       3
      EQUIVALENCE  (KDIMAG,XKDIMAG)                                     OPTS       4
C                                                                       XSMOD    227
      DIMENSION  KDIR(300)                                              XSMOD    228
      DIMENSION  HOL(2)                                                 XSMOD    229
      DATA HOL(1),HOL(2)  /3HSID, 3HGID/                                XSMOD    230
C                                                                       XSMOD    231
C     SET FLAG TO SAVE NEW DIRECTORY=HIGHEST ID NO. IN LIBRARY          XSMOD    232
      IDIRSV=MAX0(KGID(IIG+100),KSID(IIX+100),IDIR)                     XSMOD    233
      IC=IB                                                             XSMOD    234
  100 IC=IC+1                                                           XSMOD    235
      IF(KDIMAG(IC). EQ. 0)GO TO 150                                    XSMOD    236
      IF((KDIMAG(IC) . AND. 77000000000000000000B) . NE. 0)GO TO 150    XSMOD    237
      KID=KDIMAG(IC)                                                    XSMOD    238
      DO 110 I=1,IIK                                                    XSMOD    239
      IF(KID . NE. KDIR(I))GO TO 110                                    XSMOD    240
      PRINT 2, HOL(JJ),KDIR(I),KDIR(I+100),KDIR(I+200)                  XSMOD    241
    2 FORMAT(/* DELETE FROM LIBRARY *A3,I4* WITH RECALL ID=*I4* NWDS=*I6 XSMOD    242
     1)                                                                 XSMOD    243
      KDIR(I)=0                                                         XSMOD    244
      KDIR(I+100)=0                                                     XSMOD    245
      KDIR(I+200)=0                                                     XSMOD    246
      GO TO 100                                                         XSMOD    247
  110 CONTINUE                                                          XSMOD    248
      PRINT 4, HOL(JJ),KID                                              XSMOD    249
    4 FORMAT(/* CANNOT FIND *A3,I4* IN DIRECTORY. DELETE WAS REQUESTED*) XSMOD    250
      GO TO 100                                                         XSMOD    251
C                                                                       XSMOD    252
C     PRINT RECALL ID NO. OF OLD DIRECTORY FOR DTPURGE JOB.             XSMOD    253
  150 PRINT 6, IDIR                                                     XSMOD    254
    6 FORMAT(* RECALL ID NO. OF OLD DIRECTORY IS*I4*. USE FOR DTPURGE JOB XSMOD    255
     1 TO TAKE OLD DIRECTORY OFF LIBRARY TAPE.*)                        XSMOD    256
C     CLOSE UP DIRECTORY                                                XSMOD    257
      DO 170 I=1,IIK                                                    XSMOD    258
      IF(KDIR(I) . NE. 0)GO TO 170                                      XSMOD    259
      IIK=IIK-1                                                         XSMOD    260
      IF(I . GT. IIK)GO TO 170                                          XSMOD    261
      DO 160 J=I,IIK                                                    XSMOD    262
      KDIR(J)=KDIR(J+1)                                                 XSMOD    263
```

```
      KDIR(J+ 100)=KDIR(J+ 101)                                         XSMOD     264
      KDIR(J+ 200)=KDIR(J+ 201)                                         XSMOD     265
  160 CONTINUE                                                          XSMOD     266
C     CHECK SAME LOCATION FOR ANOTHER 0.                                XSMOD     267
      I=I- 1                                                            XSMOD     268
  170 CONTINUE                                                          XSMOD     269
C                                                                       XSMOD     270
C     DEFINE CURRENT GID/SID DIRECTORIES                                XSMOD     271
      IF(IIK . EQ. 0)IIK=1                                              XSMOD     272
      IF(JJ . EQ. 1)IIX=IIK                                             XSMOD     273
      IF(JJ . EQ. 2)IIG=IIK                                             XSMOD     274
      IIG2=IIG+ 100                                                     XSMOD     275
      IIG3=IIG+ 200                                                     XSMOD     276
      N. DIRGID=(F. KGID(I),I=1,IIG)/(F. KGID(I),I=101,IIG2)/           XSMOD     277
     1    (F. KGID(I),I=201,IIG3)                                       XSMOD     278
      IIX2=IIX+ 100                                                     XSMOD     279
      IIX3=IIX+ 200                                                     XSMOD     280
      N. DIRSID=(F. KSID(I),I=1,IIX)/(F. KSID(I),I=101,IIX2)/           XSMOD     281
     1    (F. KSID(I),I=201,IIX3)                                       XSMOD     282
      RETURN                                                            XSMOD     283
      END                                                               XSMOD     284
      MODULE ADDG                                                       XSMOD     285
      PROGRAM ADDG(INPUT,OUTPUT,PUNCH,TAPE10=PUNCH,TAPE7)               XSMOD     286
C                                                                       XSMOD     287
C     ADDS GID TO LIBRARY FROM CARDS. USES UNNORMALIZED ENERGY- FLUX PAIRS XSMOD  288
C     6 12 COLUMN RIGHT JUSTIFIED NUMBERS/CARD. SOME INTEGERS. SOME F FMT. XSMOD  289
C                                                                       XSMOD     290
C     KT        NO. OF ITEMS IN T ARRAY FOR 1 GROUP                     XSMOD     291
C     KGG       NO. OF ENERGY- FLUX PAIRS FOR 1 GROUP                   XSMOD     292
C     KG        TOTAL ENERGY- FLUX PAIRS FOR ALL GROUPS                 XSMOD     293
C     IGG       NO. OF GROUPS                                           XSMOD     294
C     IRV       =0 INPUT GROUPS NUMBERED IN ASCENDING ORDER.            XSMOD     295
C               0 INPUT GROUPS NUMBERED IN DESCENDING ORDER.            XSMOD     296
C                                                                       XSMOD     297
      COMMON/DIR/  KGID(300),KSID(300),IDIR,IIG,IIX,IDIRSV              DIR         2
      DIMENSION  IDNOG(100),NWDSG(100),IDNOS(100),NWDSS(100)            DIR         3
      EQUIVALENCE  (KGID(101),IDNOG(1)),(KGID(201),NWDSG(1)),           DIR         4
     1             (KSID(101),IDNOS(1)),(KSID(201),NWDSS(1))            DIR         5
      COMMON  /OPTS/ KDIMAG(1100),N(2),IALL,IB                          OPTS        2
      DIMENSION  XKDIMAG(1100)                                          OPTS        3
      EQUIVALENCE  (KDIMAG,XKDIMAG)                                     OPTS        4
      COMMON  /GRP/ IGID(1),GDES(8),IGG(1),MG(80),GE(80),KG(1),PH(500), GRP         2
     1 EK(500),VB(80),EMX(80)                                           GRP         3
      COMMON T(1000)                                                    XSMOD     301
      DIMENSION  MASK(2),MBLNK(2),FMT(3),F(3),ITAPG(100)                XSMOD     302
      DATA  I24 /100000000B/                                            XSMOD     303
      DATA  (F(I),I=1,3) /10H(*GID*I3) ,10H(*GID*I2) ,10H(*GID*I1) /    XSMOD     304
      DATA  (FMT(I),I=1,3) /10H(5XI1)     ,10H(5XI2)     ,10H(5XI3)    /  XSMOD   305
      DATA  MASK(1),MASK(2) /77770000B, 770000B/                        XSMOD     306
      DATA  MBLNK(1),MBLNK(2) /55550000B, 550000B/                      XSMOD     307
C                                                                       XSMOD     308
C     INITIALIZE STORAGES                                               XSMOD     309
      DO 90 I=1,100                                                     XSMOD     310
   90 ITAPG(I)=0                                                        XSMOD     311
      LISTFL=0                                                          XSMOD     312
C                                                                       XSMOD     313
C     DEFINE FORTRAN VARIABLES FROM DT LISTS                            XSMOD     314
      F. KGID=N. DIRGID. 1/                                             XSMOD     315
      F. IDNOG=N. DIRGID. 2/                                            XSMOD     316
      F. NWDSG=N. DIRGID. 3/                                            XSMOD     317
      F. IIG=WORDS OF N. DIRGID. 1/                                     XSMOD     318
      F. XKDIMAG=N. RDJOUT. 1/                                          XSMOD     319
      F. ITAPG=N. RDJOUT. 3/                                            XSMOD     320
      F. IALL=N. RDJOUT. 2/. 3                                          XSMOD     321
      F. IB=N. ADDG. 1                                                  XSMOD     322
      F. TL=N. ADDG. 2                                                  XSMOD     323
      F. LISTFL=N. ADDG. 4                                              XSMOD     324
      F. IDIRSV=N. IDIRSV                                               XSMOD     325
C                                                                       XSMOD     326
      SAVE REEL MHX1                                                    XSMOD     327
      IC=IB                                                             XSMOD     328
```

```
  100     IERR=0                                                              XSMOD
          IC=IC+1                                                             XSMOD
 C        CHECK FOR RETURN                                                    XSMOD
          IF(KDIMAG(IC) . EQ. 0)GO TO 2000                                    XSMOD
          IF((KDIMAG(IC) . AND. 77000000000000000000B) . NE. 0)GO TO 2000     XSMOD
 C        CHECK FOR TIME LIMIT                                                XSMOD
          CALL SECOND(CP)                                                     XSMOD
          IF(TL- CP- 10. 0)110,110,120                                        XSMOD
  110     IEXIT=1                                                             XSMOD
          PRINT 2, TL,CP                                                      XSMOD
    2     FORMAT(* ADDG. LESS THAN 10 SEC LEFT. ANOTHER ADD REQUESTED.  EXIT FL XSMOD
     1    AG SET.  TL=*F10. 3,5X*CP=*F10. 3)                                  XSMOD
          N. ADDG. 3=F. IEXIT                                                 XSMOD
          GO TO 2000                                                          XSMOD
 C        CLEAR STORAGES AND N LIST                                           XSMOD
  120     I=LOCF(EMX(80))- LOCF(IGID)+ 1                                      XSMOD
          DO 125 J=1,I                                                        XSMOD
  125     IGID(J)=0                                                           XSMOD
          DO 126 J=1,1000                                                     XSMOD
  126     T(J)=0                                                              XSMOD
          N. T=NULL                                                           XSMOD
 C                                                                            XSMOD
 C        CHECK FOR LISTING A GID ALREADY IN THE LIBRARY.                     XSMOD
          IF(LISTFL . EQ. 1)GO TO 650                                         XSMOD
 C                                                                            XSMOD
          READ 5, IGID                                                        XSMOD
          READ 3, GDES                                                        XSMOD
    3     FORMAT(8A10)                                                        XSMOD
          PRINT 6, IGID,GDES                                                  XSMOD
    6     FORMAT(*1WILL READ GID=*I4,5X,8A10)                                 XSMOD
 C        COMPARE GID ON ADD CARD AND DATA CARD                               XSMOD
          IF(KDIMAG(IC) . EQ. IGID)GO TO 130                                  XSMOD
          IERR=1                                                              XSMOD
          PRINT 4, KDIMAG(IC),IGID                                            XSMOD
    4     FORMAT(//* INPUT ERROR FOR GID.  ADD CARD= *I4* DATA CARD= *I4)     XSMOD
          GO TO 1000                                                          XSMOD
 C        CHECK GID AGAINST DIRECTORY.                                        XSMOD
  130     DO 140 I=1,IIG                                                      XSMOD
          IF(KGID(I) . EQ. IGID)GO TO 150                                     XSMOD
  140     CONTINUE                                                            XSMOD
          GO TO 160                                                           XSMOD
  150     IERR=1                                                              XSMOD
          PRINT 8, IGID                                                       XSMOD
    8     FORMAT(//* GID*I3* IS ALREADY IN THE LIBRARY. *)                    XSMOD
          GO TO 1000                                                          XSMOD
 C                                                                            XSMOD
 C        READ DATA                                                           XSMOD
  160     READ 5, IGG,  IRV                                                   XSMOD
    5     FORMAT(2I12)                                                        XSMOD
          NBNDS=IGG+1                                                         XSMOD
          READ 11, (EMX(I),I=1,NBNDS)                                         XSMOD
          KG=0                                                                XSMOD
 C                                                                            XSMOD
 C        MAIN LOOP. READ AND PROCESS 1 GROUP AT A TIME.                      XSMOD
          DO 500 I=1,IGG                                                      XSMOD
 C        SET LOWER INDEX IN ARRAY. CLEAR T STORAGE.                          XSMOD
          MG(I)=(KG+1)*124                                                    XSMOD
          DO 170 J=1,1000                                                     XSMOD
  170     T(J)=0                                                              XSMOD
          READ 9, KGRP,(T(K),K=1,5)                                           XSMOD
    9     FORMAT(I12,5F12)                                                    XSMOD
 C        CHECK IF GROUPS IN NORMAL OR REVERSE ORDER AS INDICATED BY IRV      XSMOD
          IF(IRV . EQ. 0)GO TO 190                                            XSMOD
          IF(NBNDS- KGRP . EQ. 1)GO TO 200                                    XSMOD
  180     PRINT 10, IRV,KGRP,I                                                XSMOD
   10     FORMAT(//* INPUT ERROR. REVERSE FLAG=*I2* GROUP NO. =*I3* I=*I3)    XSMOD
          IERR=1                                                              XSMOD
          GO TO 1000                                                          XSMOD
  190     IF(KGRP . NE. 1)GO TO 180                                           XSMOD
 C        CHECK FOR FLAT FLUX                                                 XSMOD
  200     IT1=T(1)                                                            XSMOD
```

```
      IF(IT1 .NE. -1)GO TO 210                                          XSMOD
C     FLAT FLUX                                                         XSMOD
      T(1)=EMX(I)                                                       XSMOD
      T(2)=T(4)=1.0                                                     XSMOD
      T(3)=EMX(I+1)                                                     XSMOD
      KT=4                                                              XSMOD
      GO TO 300                                                         XSMOD
C     CHECK LOWEST ENERGY VALUE AGAINST BOUNDARY                        XSMOD
  210 IF(ABS(T(1)-EMX(I))-.001*EMX(I))220,220,215                       XSMOD
  215 PRINT 12, T(1),EMX(I),I,IGID                                      XSMOD
   12 FORMAT(//* FIRST ENERGY VALUE NOT EQUAL TO LOWER BNDRY. E=*E13.6*  XSMOD
     1 BNDRY=*E13.6* I=*I3* GID=*I4)                                    XSMOD
      IERR=1                                                            XSMOD
      GO TO 1000                                                        XSMOD
C     CHECK FOR ONLY ONE CARD                                           XSMOD
  220 KT=0                                                              XSMOD
      CALL LAST(1,5,KT,NGO,IERR,EMX(I+1))                               XSMOD
      IF(IERR .NE. 0)GO TO 1000                                         XSMOD
      GO TO(230,250,300)NGO                                             XSMOD
C                                                                       XSMOD
C     READ NEXT CARD                                                    XSMOD
  230 KL=KT+5                                                           XSMOD
      READ 11, (T(K),K=KT,KL)                                           XSMOD
   11 FORMAT(6F12)                                                      XSMOD
      KS=KT+1                                                           XSMOD
C     CHECK FOR LAST CARD                                               XSMOD
      CALL LAST(KS,KL,KT,NGO,IERR,EMX(I+1))                             XSMOD
      IF(IERR .NE. 0)GO TO 1000                                         XSMOD
      GO TO(230,250,300)NGO                                             XSMOD
C                                                                       XSMOD
C     READ ONE VALUE FROM NEXT CARD                                     XSMOD
  250 READ 11,T(KT)                                                     XSMOD
C                                                                       XSMOD
C     PROCESS                                                           XSMOD
  300 T3=T4=T5=0                                                        XSMOD
C     GET NO. OF PAIRS FOR THIS GROUP                                   XSMOD
      KGG=KT/2                                                          XSMOD
      DO 310 J=3,KT,2                                                   XSMOD
      T3=T3 + AINTLN(T(J-2),T(J),T(J-1),T(J+1))                         XSMOD
      T2S=SQRT(T(J-2))                                                  XSMOD
      TS=SQRT(T(J))                                                     XSMOD
      F1=T(J-1)/T2S                                                     XSMOD
      F2=T(J+1)/TS                                                      XSMOD
      T4=T4 + AINTLN(T(J-2),T(J),F1,F2)                                 XSMOD
      F3=T(J-1)*T2S                                                     XSMOD
      F4=T(J+1)*TS                                                      XSMOD
      T5=T5 + AINTLN(T(J-2),T(J),F3,F4)                                 XSMOD
  310 CONTINUE                                                          XSMOD
C                                                                       XSMOD
C     NORMALIZE FLUX AND FILL PH AND EK ARRAYS                          XSMOD
      JB=KG+1                                                           XSMOD
      JE=KG+KGG                                                         XSMOD
      JJ=0                                                              XSMOD
      DO 320 J=JB,JE                                                    XSMOD
      JJ=JJ+2                                                           XSMOD
      EK(J)=T(JJ-1)                                                     XSMOD
      PH(J)=T(JJ)/T3                                                    XSMOD
  320 CONTINUE                                                          XSMOD
C                                                                       XSMOD
C     CALCULATE AVERAGE ENERGY AND VELOCITY FOR THIS GROUP              XSMOD
      VB(I)=13.83*T3/T4                                                 XSMOD
      GE(I)=T5/T4                                                       XSMOD
C     UPDATE TOTAL PAIRS FOR ALL GROUPS AND PACK INDEX                  XSMOD
      KG=KG+KGG                                                         XSMOD
      MG(I)=MG(I)+KG                                                    XSMOD
C                                                                       XSMOD
C     DEFINE DTRAN ARRAY FOR INPUT FLUX FOR 1 GROUP                     XSMOD
      N. T(I)=(F. T(J),J=2,KT,2)/                                       XSMOD
  500 CONTINUE                                                          XSMOD
C     DEFINE AND SAVE DTRAN ARRAYS,UPDATE DIRECTORY                     XSMOD
      IF(IGID .LT. 1000)GO TO 505                                       XSMOD
```

```
          PRINT 14,IGID                                                       XSMOD
   14     FORMAT(//* ADDG.  GID=*I6* OUT OF RANGE. CANNOT SAVE DT ARRAYS. *)   XSMOD
          IERR=1                                                              XSMOD
          GO TO 1000                                                          XSMOD
C                                                                             XSMOD
C         CHOOSE FORMAT TO ENCODE DTRAN NAME                                  XSMOD
  505     I=1                                                                 XSMOD
          IF(IGID . LT.  100)I=2                                              XSMOD
          IF(IGID . LT.  10)I=3                                               XSMOD
          ENCODE(7,F(I),AM)IGID                                               XSMOD
          D. ((AM))=F. IGID/F. GDES/F. IGG/(F. MG(I),I=1,IGG)/(F. GE(I),I=1,IGG)/  XSMOD
     1    F. KG/(F. PH(I),I=1,KG)/(F. EK(I),I=1,KG)/(F. VB(I),I=1,IGG)/F. IRV/N. T  XSMOD
          REWIND 10                                                           XSMOD
          SAVE D. ((AM))                                                      XSMOD
          PRINT 16, IGID,GDES                                                 XSMOD
   16     FORMAT(//* DEFINED AND SAVED GID=*I3,2X,8A10)                       XSMOD
          REWIND 10                                                           XSMOD
          NEOF=0                                                              XSMOD
C         IF DIRECTORY EMPTY,RESET NO. OF ITEMS TO ZERO                       XSMOD
          IF(KGID(1) . EQ.  0)IIG=0                                           XSMOD
  510     READ(10,13) NAME,NWDSG(IIG+1),IDNOG(IIG+1)                          XSMOD
   13     FORMAT(13X,A9,30X,I5,11X,I3)                                        XSMOD
          IF(EOF,10)550,520                                                   XSMOD
C         INCREASE ITEM COUNT IN GID DIRECTORY                                XSMOD
  520     IIG=IIG+1                                                           XSMOD
          I=IIG                                                               XSMOD
C         CHOOSE FORMAT TO DECODE DT NAME INTO DIRECTORY                      XSMOD
          DO 530 J=1,2                                                        XSMOD
          IM=NAME . AND.  MASK(J)                                             XSMOD
          IF(IM . EQ.  MBLNK(J))GO TO 540                                     XSMOD
  530     CONTINUE                                                            XSMOD
          J=3                                                                 XSMOD
  540     DECODE(9,FMT(J),NAME)KGID(I)                                        XSMOD
C         RETURN MODIFIED DIRECTORY TO DTJOB                                  XSMOD
          IDIRSV=IDNOG(IIG)                                                   XSMOD
          N. DIRGID=(F. KGID(I),I=1,IIG)/(F. IDNOG(I),I=1,IIG)/               XSMOD
     1     (F. NWDSG(I),I=1,IIG)                                              XSMOD
          GO TO 600                                                           XSMOD
C                                                                             XSMOD
  550     PRINT 18, IGID                                                      XSMOD
   18     FORMAT(//* ADDG.  EOF ON PUNCH FILE.  GID=*I4)                      XSMOD
          NEOF=NEOF+1                                                         XSMOD
          IF(NEOF . LT.  2)GO TO 510                                          XSMOD
          PRINT 20, (KGID(I),I=1,300)                                         XSMOD
   20     FORMAT(//* ADDG.  CANNOT READ PUNCH FILE TO UPDATE DIRECTORY. */     XSMOD
     1     (10I12))                                                           XSMOD
          GO TO 100                                                           XSMOD
C                                                                             XSMOD
C         IF GID REQUESTED ON TAPE1,WRITE IT TO TAPE7 NOW                     XSMOD
  600     IF(IALL . EQ.  1)GO TO 620                                          XSMOD
          DO 610 I=1,100                                                      XSMOD
          IF(ITAPG(I) . EQ.  0)GO TO 700                                      XSMOD
          IF(ITAPG(I) . EQ.  -IGID)GO TO 620                                  XSMOD
  610     CONTINUE                                                            XSMOD
          GO TO 700                                                           XSMOD
C         WRITE GID TO TAPE7                                                  XSMOD
  620     BACKSPACE 7                                                         XSMOD
          WRITE(7) IGID,GDES,IGG,(MG(I),I=1,IGG),(GE(I),I=1,IGG),KG,          XSMOD
     1     (PH(I),EK(I),I=1,KG)                                               XSMOD
          ENDFILE 7                                                           XSMOD
          PRINT 22, IGID                                                      XSMOD
   22     FORMAT(/* GID*I4* WRITTEN TO TAPE7 IN MOD ADDG. *)                  XSMOD
          GO TO 700                                                           XSMOD
C                                                                             XSMOD
C         IF LISTING GID ALREADY IN LIBRARY,MUST RETRIEVE IT HERE.            XSMOD
  650     KID=KDIMAG(IC)                                                      XSMOD
          CALL RETREV(KID,KGID,IIG,2,IERR,AM)                                 XSMOD
          IF(IERR . EQ.  0)GO TO 660                                          XSMOD
          PRINT 36,KID                                                        XSMOD
   36     FORMAT(* ADDG.  LIBRARY RETRIEVAL ERROR FOR GID *I4* NO LISTING*)   XSMOD
          GO TO 100                                                           XSMOD
```

```
C        DEFINE FTRAN VARIABLES FOR RETRIEVED GID
  660    F. IGID=D. ((AM)). 1/
         F. GDES=D. ((AM)). 2/
         F. IGG=D. ((AM)). 3/
         F. MG=D. ((AM)). 4/
         F. GE=D. ((AM)). 5/
         F. KG=D. ((AM)). 6/
         F. PH=D. ((AM)). 7/
         F. EK=D. ((AM)). 8/
         F. VB=D. ((AM)). 9/
         F. IRV=D. ((AM)). 10/
         N. T=D. ((AM)). N. T
C
         NBNDS=IGG+ 1
C
C        LIST DATA HERE
  700    PRINT 30,IGID,GDES
   30    FORMAT(*1    GID= *I6,6X,8A10//
     1   3X*G*2X*EBAR*10X*VBAR*10X*KG*2X*ENERGY*8X*NORMED FLUX*3X*INPUT FLU
     2   X*4X*FLUX/MEV*10X*INDICES*/)
         K=1
         X=PH(1)
         DO 950 I=1,IGG
         F. T=N. T(I)
         JT=1
         J1=MG(I)/I24
         J2=MG(I)- J1*I24
         II=I
         IF(IRV . NE. 0)II=NBNDS- I
         FL=PH(K)/X
         PRINT 32, II,GE(I),VB(I),K,EK(K),PH(K),T(1),FL,J1,J2
   32    FORMAT(I4,1P2E14. 6,I4,1P4E14. 6,2I8)
         J1=J1+ 1
         DO 925 J=J1,J2
         JT=JT+ 1
         FL=PH(J)/X
         PRINT 34, J,EK(J),PH(J),T(JT),FL
   34    FORMAT(32X,I4,1P4E14. 6)
  925    CONTINUE
         K=J2+ 1
         X=X*PH(K)/PH(J2)
  950    CONTINUE
C
C        IF LISTING ONLY,CHECK FOR NEXT GID
         IF(LISTFL . EQ. 1)GO TO 100
C        IF NO ERROR, SKIP BLANK CARD AT END OF GROUP.
C        ERROR. SKIP REMAINING DATA CARDS FOR THIS GROUP
 1000    READ 15, (T(K),K=1,8)
   15    FORMAT(8A10)
         IF(T(1) . NE. 10H          )GO TO 1000
         GO TO 100
 2000    N. IDIRSV=F. IDIRSV
         RETURN
         END
         FUNCTION AINTLN(E1,E2,F1,F2)
C
         X1=E1*F1
         X2=E2*F2
         IF(TEST(X1,X2) . NE. 0)GO TO 200
         AINTLN=X1*ALOG(E2/E1)
         RETURN
C
  200    AINTLN=(X2- X1)*ALOG(E2/E1)/ALOG(X2/X1)
         RETURN
         END
         FUNCTION TEST(X,Y)
C
         IF(ABS(X- Y))100,200,100
  100    Z=(X- Y)/(ABS(X)+ ABS(Y))
         IF(ABS(Z)- 1. 0E- 06)200,200,110
  110    TEST=Z
```

```
      RETURN                                                            XSMOD
C                                                                       XSMOD
  200 TEST=0                                                            XSMOD
      RETURN                                                            XSMOD
      END                                                               XSMOD
      SUBROUTINE LAST(JS,JE,KT,NGO,IERR,EMX)                            XSMOD
C     CHECKS TO SEE IF CARD JUST READ IS LAST ONE FOR THIS GROUP AND    XSMOD
C     CHECKS FOR INCREASING ENERGY WITHIN GROUP.                        XSMOD
C     JS        INDEX TO T ARRAY WHERE CHECK WILL START                 XSMOD
C     JE        INDEX TO T ARRAY WHERE CHECK WILL END                   XSMOD
C     KT        INDEX TO T ARRAY WHICH RETURNS POINTER TO NEXT PLACE IN T XSMOD
C               ARRAY IF ANOTHER CARD IS TO BE READ. IF LAST CARD HAS   XSMOD
C               BEEN READ, KT RETURNS NO. OF ITEMS IN T ARRAY.          XSMOD
C     NGO       FLAG TO BE RETURNED FOR COMPUTED GO TO                  XSMOD
C               =1 READ ANOTHER FULL CARD                               XSMOD
C               =2 READ ONLY 1 VALUE FROM NEXT CARD                     XSMOD
C               =3 READING DONE. START PROCESSING.                      XSMOD
C     IERR      ERROR FLAG. IF=0,EVERYTHING OK.                         XSMOD
C     EMX       UPPER ENERGY BOUNDARY.                                  XSMOD
C                                                                       XSMOD
      COMMON  T(1000)                                                   XSMOD
C                                                                       XSMOD
      DO 160 J=JS,JE,2                                                  XSMOD
      IF(T(J)-EMX)130,110,100                                           XSMOD
  100 PRINT 2, T(J),EMX,J,KT                                            XSMOD
    2 FORMAT(//* SUBR LAST. INTERMEDIATE ENERGY GREATER THAN EMAX.  T(J)=* XSMOD
     1E15. 8* EMX=*E15. 8* J=*I4* KT=*I4)                               XSMOD
      GO TO 1000                                                        XSMOD
  110 IF(J .EQ. JE)GO TO 120                                            XSMOD
C     READY TO PROCESS                                                  XSMOD
      KT=J+1                                                            XSMOD
      NGO=3                                                             XSMOD
      GO TO 2000                                                        XSMOD
C     READ ONE MORE VALUE                                               XSMOD
  120 KT=J+1                                                            XSMOD
      NGO=2                                                             XSMOD
      GO TO 2000                                                        XSMOD
C                                                                       XSMOD
  130 IF(J .EQ. 1)GO TO 160                                             XSMOD
C     CHECK FOR INCREASING ENERGY                                       XSMOD
      IF(T(J)-T(J-2))140,140,160                                        XSMOD
  140 PRINT 4, J,KT,T(J),T(J-2),EMX                                     XSMOD
    4 FORMAT(/* SUBR LAST. ENERGIES NOT INCREASING.  J=*I4* KT=*I4/     XSMOD
     1* T(J)=*E15. 8* T(J-2)=*E15. 8* EMX=*E15. 8)                      XSMOD
      GO TO 1000                                                        XSMOD
  160 CONTINUE                                                          XSMOD
      KT=KT+6                                                           XSMOD
      NGO=1                                                             XSMOD
      GO TO 2000                                                        XSMOD
 1000 IERR=1                                                            XSMOD
 2000 RETURN                                                            XSMOD
      END                                                               XSMOD
      SUBROUTINE RETREV(KID,KGID,IIK,DUM,IERR,AM)                       XSMOD
C     RETRIEVES SELECTED GID FROM DTRAN TAPE                            XSMOD
C     KID       GID NO. IN LIBRARY                                      XSMOD
C     KGID      DTRAN DIRECTORY TO GID                                  XSMOD
C     IIK       NO. OF GID IN DIRECTORY                                 XSMOD
C     DUM       DUMMY                                                   XSMOD
C     IERR      ERROR FLAG. 0 IF OK.                                    XSMOD
C     AM        DTRAN NAME FOR THIS GID                                 XSMOD
C                                                                       XSMOD
      DIMENSION  KGID(300),M(3)                                         XSMOD
      DATA  (M(I),I=1,3) /10H(*GID*I3) ,10H(*GID*I2) ,10H(*GID*I1) /    XSMOD
C                                                                       XSMOD
      IERR=0                                                            XSMOD
      DO 110 I=1,IIK                                                    XSMOD
      IF(KID .EQ. KGID(I))GO TO 120                                     XSMOD
  110 CONTINUE                                                          XSMOD
      PRINT 2, KID                                                      XSMOD
    2 FORMAT(* ADDG,RETREV.  CANNOT FIND GID *I3* ON DTRAN TAPE. *)     XSMOD
      IERR=1                                                            XSMOD
```

```
          RETURN                                                        XSMOD    684
C                                                                       XSMOD    685
C         GET RECALL NO. ASSOCIATED WITH THIS GID                       XSMOD    686
  120     K=KCID(I+100)                                                 XSMOD    687
C         SELECT FORMAT FOR ENCODING DTRAN NAME                         XSMOD    688
          IF(KID .LT. 1000)I=1                                          XSMOD    689
          IF(KID .LT. 100)I=2                                           XSMOD    690
          IF(KID .LT. 10)I=3                                            XSMOD    691
          ENCODE(7,M(I),AM)KID                                          XSMOD    692
C         DO RETRIEVE                                                   XSMOD    693
          D.((AM))=D.DT00010 *MHX1* F.K                                 XSMOD    694
          RETURN                                                        XSMOD    695
          END                                                           XSMOD    696
          END MODULE                                                    XSMOD    697
          MODULE ADDXS                                                  XSMOD    698
          PROGRAM ADDXS(INPUT,OUTPUT,PUNCH,TAPE2,TAPE6,TAPE3=INPUT,     XSMOD    699
     1    TAPE10=PUNCH)                                                 XSMOD    700
C                                                                       XSMOD    701
C         ADDS SID TO LIBRARY FROM CARDS OR TAPE IN UK OR LLL FORMAT.   XSMOD    702
C         CARDS OR TAPE, LLL OR UK ARE SPECIFIED ON 1ST DATA CARD READ. XSMOD    703
C         SENSE 4 ON TO PRINT UKTAPE DIRECTORY.                         XSMOD    704
C         TAPE2   UK OR LLL TAPE. ORIGINAL LLL TAPES MUST BE PRE-PROCESSED XSMOD 705
C                 BY MCGIRT S LRLUK CODE.                               XSMOD    706
C         N       INPUT UNIT. 3=CARDS. 2=TAPE                           XSMOD    707
C         SOURCE  LLL OR UK DATA                                        XSMOD    708
C         IB      INDEX TO KDIMAG                                       XSMOD    709
C         CTIND   CARDS OR TAPE INDICATOR                               XSMOD    710
C         NIN     NUCLIDE ID NO. TO SEARCH FOR ON TAPE OR CHECK ON CARDS XSMOD   711
C         KID     SID NO. IN LIBRARY                                    XSMOD    712
C         ISID    NUCLIDE ID NO. READ FROM TAPE OR CARDS                XSMOD    713
C                                                                       XSMOD    714
          COMMON /CNTRL/ ISID(1),KID,ND,ITIDS(25),SDES(8),IR,ITIDR(50),NES, CNTRL  2
     1    ES(4000),IDR(50),ME(50),IRS(50),QR(50),IZ,MIXT,K2             CNTRL      3
          COMMON /X/ X(4850)                                            X          2
          DIMENSION ITID(1),TDES(8),E(400),TK(4000),MD(400),PROB(10),   ANGD       2
     1    NMU(10)                                                       ANGD       3
          EQUIVALENCE (ITID(1), X(1)), (TDES(1),X(2)), (NED,X(10)),     ANGD       4
     1    (E(1),X(11)), (ISYS,X(411)), (A,X(412)), (LEGN,X(413)),       ANGD       5
     2    (KT,X(414)), (TK(1),X(415)), (MD(1),X(4415)), (NINC,X(4815)), ANGD       6
     3    (PROB(1),X(4816)), (NMU(1),X(4826))                           ANGD       7
C                                                                       ANGD       8
          DIMENSION SIG(4000)                                           XS         2
          EQUIVALENCE (SIG(1),X(1))                                     XS         3
C                                                                       XS         4
          DIMENSION ESJ(3000)                                           XSMOD    719
          EQUIVALENCE (ESJ(1),ES(1))                                    XSMOD    720
          COMMON /CDID/IA7,IA8,IA9,IBLNO(100),IDB(100),ICDS(100),DUM(9),LLL CDID   2
          COMMON/DIR/ KGID(300),KSID(300),IDIR,IIG,IIX,IDIRSV           DIR        2
          DIMENSION IDNOG(100),NWDSG(100),IDNOS(100),NWDSS(100)         DIR        3
          EQUIVALENCE (KGID(101),IDNOG(1)),(KGID(201),NWDSG(1)),        DIR        4
     1    (KSID(101),IDNOS(1)),(KSID(201),NWDSS(1))                     DIR        5
          COMMON /OPTS/ KDIMAG(1100),N(2),IALL,IB                       OPTS       2
          DIMENSION XKDIMAG(1100)                                       OPTS       3
          EQUIVALENCE (KDIMAG,XKDIMAG)                                  OPTS       4
C                                                                       XSMOD    724
          DIMENSION HOL(4)                                              XSMOD    725
          DIMENSION MASK(3),MBLNK(3),FMT(4),F(3)                        XSMOD    726
          DIMENSION NUKID(200),NCARDS(200),P(200),DES(200)              XSMOD    727
          DIMENSION ITAPX(100)                                          XSMOD    728
C                                                                       XSMOD    729
          DATA HOL(1),HOL(2),HOL(3) /1HX,5H TAPE,6H CARDS/              XSMOD    730
          DATA SOURCE /6H UK /                                          XSMOD    731
          DATA (FMT(I),I=1,4) /10H(5XI1)    ,10H(5XI2)    ,10H(5XI3)    XSMOD    732
     1    10H(5XI4)    /                                                XSMOD    733
          DATA (MASK(I),I=1,3) /00000000000077777700,00000000000000777700 XSMOD  734
     1    , 00000000000000007700/                                       XSMOD    735
          DATA (MBLNK(I),I=1,3) /00000000000055555500,                  XSMOD    736
     1    00000000000000555500, 00000000000000005500/                   XSMOD    737
          DATA I24 /100000000B/                                         XSMOD    738
          DATA (F(I),I=1,3) /10H(*SID*I4) ,10H(*SID*I3) ,10H(*SID*I2) / XSMOD    739
C                                                                       XSMOD    740
```

```
      DO 90 I=1,100
   90 ITAPX(I)=0
C
C     DEFINE FORTRAN VARIABLES TRANSMITTED BY DT LISTS.
      F. KSID=N. DIRSID. 1/
      F. IDNOS=N. DIRSID. 2/
      F. NWDSS=N. DIRSID. 3/
      F. IIX=WORDS OF N. DIRSID. 1/
      F. IB=N. ADDXS. 1
      F. TL=N. ADDXS. 2
      F. XKDIMAG=N. RDJOUT. 1/
      F. ITAPX=N. RDJOUT. 4/
      F. IALL=N. RDJOUT. 2/. 3
      F. IDIRSV=N. IDIRSV
C
      SAVE REEL MHX1
      IERR=0
      IC=IB
  100 IC=IC+ 1
      IF(IERR . EQ. 0)GO TO 1105
C     IF ERROR DETECTED BEFORE ALL CARDS READ,MUST SKIP REMAINING CARDS
C     FOR NUCLIDE IN ERROR.
      KTOTCDS=KTOTCDS+ IERR
      DO 1100 I=KTOTCDS,NTOTCDS
 1100 READ(N,11)
C     CHECK FOR RETURN TO MAIN CODE.
 1105 IF(KDIMAG(IC) . EQ. 0)GO TO 2000
      IF((KDIMAG(IC) . AND. 77000000000000000000B) . NE. 0)GO TO 2000
C     CHECK FOR TIME LIMIT
      CALL SECOND(CP)
      IF(TL- CP- 15. 0)1110,1110,1115
 1110 IEXIT=1
      PRINT 28, TL,CP
   28 FORMAT(* ADDXS. LESS THAN 15 SEC LEFT. ANOTHER ADD REQUESTED. EXIT
     1 FLAG SET. TL=*F10. 3,5X*CP=*F10. 3)
      N. ADDXS. 3=F. IEXIT
      GO TO 2000
 1115 CONTINUE
C     CLEAR STORAGLS AND DT ARRAYS
      I=LOCF(K2)- LOCF(ISID)+ 1
      DO 101 J=1,I
  101 ISID(J)=0
      I=LOCF(DUM(9))- LOCF(IBLNO(1))+ 1
      DO 102 J=1,I
  102 IBLNO(J)=0
      IERR=0
      KTOTCDS=0
      N. CNTRL=NULL
      N. ANGD=NULL
      N. XS=NULL
      N. ESJ=NULL
C     READ INPUT STREAM
      READ 1,KID,SOURCE,CTIND,NIN
    1 FORMAT(I6,2A6,I6)
      READ 15, SDES
   15 FORMAT(8A10)
      N=2
      IF(CTIND . EQ. 6H CARDS)N=3
      PRINT 2, KID,HOL(N),SDES
    2 FORMAT(*1SID*I5* WILL BE READ FROM *A5/1X,8A10)
C     COMPARE SID ON ADD CARD AND DATA CARD.
      IF(KDIMAG(IC) . EQ. KID)GO TO 105
      IERR=2
      PRINT 4, KDIMAG(IC),KID
    4 FORMAT(*DINPUT ERROR FOR SID. ADD CARD= *I4* DATA CARD= *I4)
C
C     CHECK SID AGAINST OLD DIRECTORY.
  105 DO 110 I=1,IIX
      IF(KSID(I) . EQ. KID)GO TO 115
  110 CONTINUE
      GO TO 120
```

```
  115     IERR=2                                                          XSMOD  812
          PRINT 6, KID                                                    XSMOD  813
    6     FORMAT(//* SID*I4* IS ALREADY IN THE LIBRARY. *)                XSMOD  814
C                                                                         XSMOD  815
C         CHECK FOR LLL OR UK                                             XSMOD  816
  120     LLL=0                                                           XSMOD  817
          IF(SOURCE . EQ. 6H    LLL)LLL=1                                 XSMOD  818
C                                                                         XSMOD  819
C         CHECK FOR READING TAPE OR CARDS                                 XSMOD  820
          IF(N . EQ. 2)GO TO 150                                          XSMOD  821
C         SKIP CARDS IF ERROR FLAG SET                                    XSMOD  822
          READ(3,7) ISID, NTOTCDS,ICDS0,IZ,A,LBLNO,IA7,IA8,IA9            XSMOD  823
          IF(IERR . EQ. 0)GO TO 170                                       XSMOD  824
          GO TO 100                                                       XSMOD  825
C                                                                         XSMOD  826
C         READ DATA                                                       XSMOD  827
  150     IF(IERR . EQ. 0)GO TO 155                                       XSMOD  828
          IERR=0                                                          XSMOD  829
          GO TO 100                                                       XSMOD  830
  155     IF(LLL . EQ. 1)GO TO 160                                        XSMOD  831
          IF(JSWFL . EQ. 2)GO TO 160                                      XSMOD  832
C         READ TAPE DIRECTORY                                             XSMOD  833
    5     FORMAT(I11,1X,I11)                                              XSMOD  834
          READ(2,5) NFILES                                                XSMOD  835
          READ(2,13) (P(I),NUKID(I),NCARDS(I),DES(I),I=1,NFILES)          XSMOD  836
   13     FORMAT(A1,I10,1X,I11,2X,A10)                                    XSMOD  837
          CALL SSWTCH(4,JSW)                                              XSMOD  838
          IF(JSW . EQ. 2)GO TO 160                                        XSMOD  839
          IF(JSWFL . EQ. 2)GO TO 160                                      XSMOD  840
          JSWFL=2                                                         XSMOD  841
          PRINT 14, NFILES,(P(I),NUKID(I),NCARDS(I),DES(I),I=1,NFILES)    XSMOD  842
   14     FORMAT(//* DIRECTORY TO UKTAPE. NO. OF FILES= *I5//             XSMOD  843
    1     * NUCLIDE ID NO. ,NO. OF CARDS AND DESCRIPTION FOLLOW. *        XSMOD  844
    2     //(3(1X,A1,1X,I5,I6,2X,A10,15X)))                               XSMOD  845
C         SEARCH FOR REQUESTED SID                                        XSMOD  846
  160     READ(N,7) ISID,NTOTCDS,ICDS0,IZ,A,LBLNO,IA7,IA8,IA9             XSMOD  847
    7     FORMAT(1X,I10,1X,3(I11,1X),F11. 6,1X,I11,1X,I3,I2,I3)           XSMOD  848
          IF(EOF,2)161,162                                                XSMOD  849
  161     PRINT 16, NIN,KID                                               XSMOD  850
   16     FORMAT(*D CANNOT FIND NIN=*I4* ON TAPE.  SID=*I4)               XSMOD  851
          REWIND 2                                                        XSMOD  852
          GO TO 100                                                       XSMOD  853
  162     CONTINUE                                                        XSMOD  854
          IF(ISID . EQ. NIN)GO TO 170                                     XSMOD  855
          PRINT 18, ISID                                                  XSMOD  856
   18     FORMAT(* NIN=*I4* ENCOUNTERED. *)                               XSMOD  857
          DO 165 I=2,NTOTCDS                                              XSMOD  858
          READ(2,11)                                                      XSMOD  859
   11     FORMAT(1X)                                                      XSMOD  860
  165     CONTINUE                                                        XSMOD  861
          GO TO 160                                                       XSMOD  862
C         FOUND SID.  CLEAR AND READ BLOCK 0                              XSMOD  863
  170     IBLK=0                                                          XSMOD  864
          NCD=1                                                           XSMOD  865
          ICHK=PRCT(1,IBLK,ISID)                                          XSMOD  866
          IF(ICHK . NE. 0)GO TO 1000                                      XSMOD  867
          DO 172 I=1,300                                                  XSMOD  868
  172     IBLNO(I)=0                                                      XSMOD  869
          PRINT 26,ISID,SOURCE,HOL(N),KID                                 XSMOD  870
   26     FORMAT(/* READING NIN=*I5* FROM*A6,A6* TO GO INTO LIBRARY AS SID*  XSMOD  871
    1         I5)                                                         XSMOD  872
          IE=ICDS0-1                                                      XSMOD  873
          J=-1                                                            XSMOD  874
          DO 174 I=1,IE                                                   XSMOD  875
          J=J+2                                                           XSMOD  876
          READ(N,9) IBLNO(J),IDB(J),ICDS(J),IBLNO(J+1),IDB(J+1),          XSMOD  877
    1          ICDS(J+1),IA7,IA8,IA9                                      XSMOD  878
    9     FORMAT(6(I11,1X),I3,I2,I3)                                      XSMOD  879
          NCD=I+1                                                         XSMOD  880
          ICHK=PRCT(I+1,0,ISID)                                           XSMOD  881
          IF(ICHK . NE. 0)GO TO 1000                                      XSMOD  882
  174     CONTINUE                                                        XSMOD  883
```

```
      KTOTCDS=ICDS0                                                        XSMOD
C     READ CROSS SECTIONS,ANG DISTRS AND SECONDARY ENERGIES.               XSMOD
      DO 240 I=1,LBLNO                                                     XSMOD
      IRT=IDB(I)                                                           XSMOD
C     DO NOT SAVE REACTIONS 1001,1003,1004                                 XSMOD
      IF(IRT .NE. 1001 .AND. IRT .NE. 1003 .AND. IRT .NE. 1004)GOTO 1174   XSMOD
      IL=ICDS(I)                                                           XSMOD
      DO 2174 J=1,IL                                                       XSMOD
 2174 READ(N,11)                                                           XSMOD
      GO TO 205                                                            XSMOD
 1174 ITP=IRT/1000                                                         XSMOD
      IF(ITP   .EQ.  1)GO TO 176                                           XSMOD
      IF(ITP   .EQ.  2)GO TO 177                                           XSMOD
      IF(ITP   .EQ.  3)GO TO 178                                           XSMOD
      IF(ITP   .EQ.  4)GO TO 9178                                          XSMOD
  175 PRINT 20, IRT,I,LBLNO,IA7                                            XSMOD
   20 FORMAT(//* ADDXS. CANNOT READ IN THIS CLASSIFICATION. WILL SKIP. */  XSMOD
     1 * REACTION TYPE=*I5* BLOCK NO=*I3* LAST BLOCK=*I3* NIN=*I5)         XSMOD
      JE=ICDS(I)                                                           XSMOD
      DO 1176 J=1,JE                                                       XSMOD
 1176 READ(N,11)                                                           XSMOD
      GO TO 205                                                            XSMOD
  176 CALL READXS(KID,I,N,IERR)                                            XSMOD
      ITIDR(IR)=-7                                                         XSMOD
      GO TO 179                                                            XSMOD
  177 CALL READAD(KID,I,N,IERR)                                            XSMOD
      ITIDR(IR)=ITIDS(ND)                                                  XSMOD
      GO TO 179                                                            XSMOD
  178 CALL READSEC(KID,I,N,IERR)                                           XSMOD
      GO TO 179                                                            XSMOD
 9178 CALL READNU(KID,I,N,IERR)                                            XSMOD
  179 IF(IERR .EQ. 0)GO TO 205                                             XSMOD
      PRINT 22, KID,HOL(N)                                                 XSMOD
   22 FORMAT(//* COULD NOT ADD SID*I4* TO LIBRARY FROM *A7)                XSMOD
      GO TO 100                                                            XSMOD
C     INCREASE COUNT OF CARDS READ                                         XSMOD
  205 KTOTCDS=KTOTCDS+ICDS(I)                                              XSMOD
  240 CONTINUE                                                             XSMOD
C                                                                          XSMOD
C     ALL DATA FOR ONE NUCLIDE READ                                        XSMOD
      MIXT=0                                                               XSMOD
      K2=0                                                                 XSMOD
      DO 207 J=1,IR                                                        XSMOD
  207 K2=K2+IRS(J)                                                         XSMOD
C                                                                          XSMOD
C     DEFINE XS CONTROL FILE.                                              XSMOD
      N. CNTRL=F. KID/F. ND/(F. ITIDS(J),J=1,ND)/F. SDES/F. IR/(F. ITIDR(J), XSMOD
     1    J=1,IR)/F. NES/(F. ES(J),J=1,NES)/(F. IDR(J),J=1,IR)/(F. ME(J),  XSMOD
     2    J=1,IR)/(F. IRS(J),J=1,IR)/(F. QR(J),J=1,IR)/F. IZ/F. MIXT/F. K2/ XSMOD
     3    F. LLL/                                                          XSMOD
   24 FORMAT(* ADDXS. CANNOT SAVE DT ARRAYS. KID OUT OF RANGE. *I6)        XSMOD
      IF(KID .LT. 10000. AND. KID .GE. 10)GO TO 210                        XSMOD
      PRINT 24,KID                                                         XSMOD
      GO TO 100                                                            XSMOD
C     CHOOSE FORMAT FOR ENCODING DTRAN NAME                                XSMOD
  210 IF(KID .LT. 10000)K=1                                                XSMOD
      IF(KID .LT. 1000)K=2                                                 XSMOD
      IF(KID .LT. 100)K=3                                                  XSMOD
      ENCODE(7,F(K),AM)KID                                                 XSMOD
      D. ((AM))=N. CNTRL,N. XS,N. ESJ,N. ANGD                              XSMOD
      REWIND 10                                                            XSMOD
      SAVE D. ((AM))                                                       XSMOD
C                                                                          XSMOD
C     ALL DATA FOR ONE NUCLIDE SAVED ON DT TAPE.                           XSMOD
C     UPDATE DIRECTORY                                                     XSMOD
      REWIND 10                                                            XSMOD
      NEOF=0                                                               XSMOD
C     IF SID DIRECTORY EMPTY,RESET NO. OF ITEMS TO ZERO                    XSMOD
      IF(XSID(1) .EQ. 0)IIX=0                                              XSMOD
  260 READ(10,3) NAME,NWDSS(IIX+1),IDNOS(IIX+1)                            XSMOD
    3 FORMAT(13X,A9,30X,I5,11X,I3)                                         XSMOD
```

```
      IF(EOF,10)280,270
  270 CONTINUE                                                          XSMOD
C     INCREASE ITEM COUNT IN SID DIRECTORY                              XSMOD
      IIX=IIX+1                                                         XSMOD
      I=IIX                                                             XSMOD
C     CHOOSE FORMAT TO DECODE DT NAME INTO DIRECTORY                    XSMOD
      DO 275 J=1,3                                                      XSMOD
      IM=NAME .AND. MASK(J)                                             XSMOD
      IF(IM .EQ. MBLNK(J))GO TO 277                                     XSMOD
  275 CONTINUE                                                          XSMOD
      J=4                                                               XSMOD
  277 IFMT=J                                                            XSMOD
  279 DECODE(9,FMT(IFMT),NAME)KSID(I)                                   XSMOD
C     RETURN MODIFIED DIRECTORY TO DTJOB                                XSMOD
      IDIRSV=IDNOS(IIX)                                                 XSMOD
C     SET DIRECTORY SAVE FLAG=HIGHEST RECALL ID                         XSMOD
      N.DIRSID=(F.KSID(I),I=1,IIX)/(F.IDNOS(I),I=1,IIX)/                XSMOD
     1    (F.NWDSS(I),I=1,IIX)                                          XSMOD
      GO TO 300                                                         XSMOD
  280 PRINT 10                                                          XSMOD
   10 FORMAT(* ADDXS. EOF ON PUNCH FILE. *)                             XSMOD
      NEOF=NEOF+1                                                       XSMOD
      IF(NEOF .LT. 2)GO TO 260                                          XSMOD
      PRINT 12, (KSID(I),I=1,300)                                       XSMOD
   12 FORMAT(* ADDXS. CANNOT READ PUNCH FILE TO UPDATE DIRECTORY. */    XSMOD
     1    (10I12))                                                      XSMOD
      IERR=0                                                            XSMOD
      GO TO 100                                                         XSMOD
C                                                                       XSMOD
C     IF SID REQUESTED ON TAPE1, WRITE IT TO TAPE6 NOW                  XSMOD
  300 IF(IALL .EQ. 1)GO TO 324                                          XSMOD
C     SEARCH TAPE LIST                                                  XSMOD
      DO 305 I=1,100                                                    XSMOD
      IF(ITAPX(I) .EQ. 0)GO TO 400                                      XSMOD
      IF(ITAPX(I) .EQ. -KID)GO TO 324                                   XSMOD
  305 CONTINUE                                                          XSMOD
      GO TO 400                                                         XSMOD
C     WRITE SID TO TAPE6                                                XSMOD
  324 BACKSPACE 6                                                       XSMOD
  325 WRITE(6)  KID,ND,(ITIDS(I),I=1,ND),SDES,IR,(ITIDR(I),I=1,IR),NES, XSMOD
     1    (ES(I),I=1,NES),(IDR(I),I=1,IR),(ME(I),I=1,IR),(IRS(I),I=1,IR) XSMOD
     2    ,(QR(I),I=1,IR),IZ,MIXT,K2,LLL                                XSMOD
C     ONLY ONE REACTION AT A TIME IN CORE                               XSMOD
      DO 340 K=1,IR                                                     XSMOD
      F.SIG=N.XS(K)                                                     XSMOD
      F.IWD1=WORDS OF N.XS(K)                                           XSMOD
      IF(IRS(K) .NE. 0)GO TO 330                                        XSMOD
      WRITE(6)(SIG(I),I=1,IWD1)                                         XSMOD
      GO TO 340                                                         XSMOD
  330 F.ESJ=N.ESJ(K)                                                    XSMOD
      F.IWD2=WORDS OF N.ESJ(K)                                          XSMOD
      WRITE(6) (SIG(I),I=1,IWD1),(ESJ(I),I=1,IWD2)                      XSMOD
  340 CONTINUE                                                          XSMOD
C     ONLY ONE A-D AT A TIME IN CORE                                    XSMOD
      DO 350 K=1,ND                                                     XSMOD
      F.ITID=N.ANGD(K).1/                                               XSMOD
      F.TDES=N.ANGD(K).2/                                               XSMOD
      F.NED=N.ANGD(K).3/                                                XSMOD
      F.E=N.ANGD(K).4/                                                  XSMOD
      F.ISYS=N.ANGD(K).5/                                               XSMOD
      F.A=N.ANGD(K).6/                                                  XSMOD
      F.LEGN=N.ANGD(K).7/                                               XSMOD
      F.KT=N.ANGD(K).8/                                                 XSMOD
      F.TK=N.ANGD(K).9/                                                 XSMOD
      F.MD=N.ANGD(K).10/                                                XSMOD
      F.NINC=N.ANGD(K).11/                                              XSMOD
      WRITE(6) ITID,TDES,NED,(E(I),I=1,NED),ISYS,A,LEGN,KT,             XSMOD
     1    (TK(I),I=1,KT),(MD(I),I=1,NED),NINC                           XSMOD
  350 CONTINUE                                                          XSMOD
      ENDFILE 6                                                         XSMOD
      ENDFILE 6                                                         XSMOD
      PRINT 351,KID                                                     XSMOD
```

```
  351      FORMAT(* SID*I5* WRITTEN TO TAPE6 IN MOD ADDXS*)                 XSMOD  1027
C                                                                           XSMOD  1028
C          LIST DATA IN S-4 FORMAT HERE                                     XSMOD  1029
C          PREPARE LIST FOR MODULE  LISTXS                                  XSMOD  1030
  400      N. LISTXS=0                                                      XSMOD  1031
           KALL LISTXS                                                      XSMOD  1032
           GO TO 100                                                        XSMOD  1033
C          ERROR ENTRY. SET IERR TO SKIP REMAINING CARDS                    XSMOD  1034
 1000      IERR=NCD+1                                                       XSMOD  1035
           GO TO 100                                                        XSMOD  1036
C          STANDARD RETURN                                                  XSMOD  1037
 2000      REWIND 2                                                         XSMOD  1038
           N. IDIRSV=F. IDIRSV                                              XSMOD  1039
           RETURN                                                           XSMOD  1040
           END                                                              XSMOD  1041
           FUNCTION MOD(N,L)                                                XSMOD  1042
           IF(N .LT. L)GO TO 10                                             XSMOD  1043
    1      N=N-L                                                            XSMOD  1044
           IF(N .GE. L)GO TO 1                                              XSMOD  1045
   10      MOD=N                                                            XSMOD  1046
           RETURN                                                           XSMOD  1047
           END                                                              XSMOD  1048
           FUNCTION   PRCT(I,IB,NIN)                                        XSMOD  1049
           COMMON  /CDID/IA7,IA8,IA9,IBLNO(100),IDB(100),ICDS(100),DUM(9),LLL   CDID      2
           IERR=0                                                           XSMOD  1051
           IM=MOD(I,1000)                                                   XSMOD  1052
           IF(IA9 .NE. IM)IERR=1                                            XSMOD  1053
           IF(IA8 .NE. IB)IERR=2                                            XSMOD  1054
           IF(IA7 .NE. NIN)IERR=3                                           XSMOD  1055
           IF(IERR .EQ .0)GO TO 100                                         XSMOD  1056
           PRINT 2, IA7,IA8,IA9,NIN,IB,I                                    XSMOD  1057
    2      FORMAT(//* ID FIELD ERROR ON DATA CARD.  A7,8,9= *3I6* NIN,BLK,CDNO  XSMOD  1058
    1      = *3I6)                                                          XSMOD  1059
  100      PRCT=IERR                                                        XSMOD  1060
           RETURN                                                           XSMOD  1061
           END                                                              XSMOD  1062
           SUBROUTINE MULTI(KID,IBLK,N,IERR,NCD,NDIST)                      XSMOD  1063
C          READS MULTIPLE ANG DISTRS FOR FIRST NEUTRON AND COMBINES THEM INTO   XSMOD  1064
C           ONE A-D FOR LIBRARY.                                            XSMOD  1065
C          KID        SID NO.  IN LIBRARY                                   XSMOD  1066
C          IBLK       BLOCK NO.                                             XSMOD  1067
C          IERR       ERROR FLAG.  NO ERROR=0.                              XSMOD  1068
C          N          INPUT UNIT.  3=CARDS.  2=TAPE                         XSMOD  1069
C                                                                           XSMOD  1070
           COMMON  /X/ X(4850)                                              X         2
           DIMENSION  ITID(1),TDES(8),E(400),TK(4000),MD(400),PROB(10),     ANGD      2
    1          NMU(10)                                                      ANGD      3
           EQUIVALENCE  (ITID(1), X(1)), (TDES(1),X(2)), (NED,X(10)),       ANGD      4
    1          (E(1),X(11)), (ISYS,X(411)), (A,X(412)), (LEGN,X(413)),      ANGD      5
    2          (KT,X(414)), (TK(1),X(415)), (MD(1),X(4415)), (NINC,X(4815)),  ANGD    6
    3          (PROB(1),X(4816)), (NMU(1),X(4826))                          ANGD      7
C                                                                           ANGD      8
           COMMON  /CDID/IA7,IA8,IA9,IBLNO(100),IDB(100),ICDS(100),DUM(9),LLL   CDID      2
C                                                                           XSMOD  1074
           DIMENSION XMU(100),T(100),XMU1(100),T1(100)                      XSMOD  1075
C                                                                           XSMOD  1076
C          READ IN FIRST DISTRIBUTION INTO RESIDUAL                         XSMOD  1077
           DO 100 I=1,100                                                   XSMOD  1078
           XMU(I)=0                                                         XSMOD  1079
           T(I)=0                                                           XSMOD  1080
           XMU1(I)=0                                                        XSMOD  1081
           T1(I)=0                                                          XSMOD  1082
  100      CONTINUE                                                         XSMOD  1083
           L=1                                                              XSMOD  1084
           KDS=NMU/3                                                        XSMOD  1085
           NMUR=MOD(NMU,3)                                                  XSMOD  1086
           IF(NMUR .NE. 0)KDS=KDS+1                                         XSMOD  1087
           K1=1                                                             XSMOD  1088
           DO 130 KD=1,KDS                                                  XSMOD  1089
           KL=K1+2                                                          XSMOD  1090
           READ(N,1) (XMU(K),T(K),K=K1,KL),IA7,IA8,IA9                      XSMOD  1091
    1      FORMAT(6(F11.6,1X),I3,I2,I3)                                     XSMOD  1092
```

```
      NCD=NCD+ 1
      ICHK=PRCT(NCD,IBLK,ISID)
      IF(ICHK . NE. 0)GO TO 1000
      DO 120 K=K1,KL
  120 T(K)=PROB*T(K)
      K1=K1+ 3
  130 CONTINUE
      KLADJ=3- NMUR
      KL=KL- KLADJ
C
C     READ IN REMAINING DISTRBNS AND COMBINE WITH RESIDUAL ONE AT A TIME
      DO 250 LD=2,NDIST
      L=L+ 1
      READ(N,3) ERL,ERH,NCER,DUM(1),PROB(L),NMU(L),IA7,IA8,IA9
    3 FORMAT(2(F11. 6,1X),2(I11,1X),F11. 6,1X,I11,1X,I3,I2,I3)
      NCD=NCD+ 1
      ICHK=PRCT(NCD,IBLK,ISID)
      IF(ICHK . NE. 0)GO TO 1000
      IF(NDIST . EQ. DUM(1))GO TO 140
      PRINT 2, NDIST,DUM(1),IRT,IBLK,ISID
    2 FORMAT(* MULTI. DISCREPENCY IN NO. OF DISTRS FOR A GIVEN NEUTRON. N
     1 DIST=*I4* DUM=*I4* REACTION=*I5* BLOCK=*I5* NIN=*I5)
      GO TO 1000
  140 KDS=NMU(L)/3
      NMUR=MOD(NMU(L),3)
      IF(NMUR . NE. 0)KDS=KDS+ 1
      K1=M=1
      DO 210 KD=1,KDS
      KL1=K1+ 2
      READ(N,1) (XMU1(K),T1(K),K=K1,KL),IA7,IA8,IA9
      NCD=NCD+ 1
      ICHK=PRCT(NCD,IBLK,ISID)
      IF(ICHK . NE. 0)GO TO 1000
      DO 150 K=K1,KL1
  150 T1(K)=PROB(L)*T1(K)
C
C     FOR 1 CARD CHECK MU≠S AGAINST RESIDUAL
      DO 200 K=K1,KL1
      IF(ABS(XMU(M)- XMU1(K)). LE. 1. 0E- 05)GO TO 195
      IF(XMU(M) . GT. XMU1(K))GO TO 180
  160 JJ=KL1
C     PUT EXTRA POINT IN TEMP
      DO 170 KK=K,KL1
      T1(JJ+ 1)=T1(JJ)
      XMU1(JJ+ 1)=XMU1(JJ)
  170 JJ=J- 1
      XMU1(K)=XMU(M)
      T1(K)=T1(K- 1)+ (XMU(M)- XMU1(K- 1))/(XMU1(K)- XMU1(K- 1))
     1     *(T1(K)- T1(K- 1))
      K1=K1+ 1
      KL1=KL1+ 1
      GO TO 195
  180 JJ=KL
C     PUT EXTRA POINT IN RESIDUAL
      DO 190 KK=M,KL
      T(JJ+ 1)=T(JJ)
      XMU(JJ+ 1)=XMU(JJ)
  190 JJ=JJ- 1
      XMU(M)=XMU1(K)
      T(M)=T(M- 1)+ (XMU1(K)- XMU(M- 1))/(XMU(M+ 1)- XMU(M- 1))
     1     *(T(M+ 1)- T(M- 1))
      KL=KL+ 1
  195 M=M+ 1
  200 CONTINUE
      K1=K1+ 3
  210 CONTINUE
C     COMBINE TEMP AND RESIDUAL T ARRAYS.
      DO 220 K=1,KL
  220 T(K)=T(K)+ T1(K)
  250 CONTINUE
C     ALL DSTRBNS READ IN AND COMBINED.
      KB=KT
```

```
      KE=KB+2*KL                                                        XSMOD   1165
      L=0                                                               XSMOD   1166
C     PUT RESIDUAL INTO PERMANENT ANG DISTR ARRAY                       XSMOD   1167
      DO 260 K=KB,KE,2                                                  XSMOD   1168
      L=L+1                                                             XSMOD   1169
      TK(K)=XMU(L)                                                      XSMOD   1170
      TK(K+1)=T(L)                                                      XSMOD   1171
  260 CONTINUE                                                          XSMOD   1172
      KT=KE                                                             XSMOD   1173
      GO TO 2000                                                        XSMOD   1174
 1000 IERR=NCD+1                                                        XSMOD   1175
 2000 RETURN                                                            XSMOD   1176
      END                                                               XSMOD   1177
      SUBROUTINE READAD(KID,IBLK,N,IERR)                                XSMOD   1178
C     READS ANGULAR DISTRIBUTION DATA FROM UK FORMAT TAPE OR CARDS      XSMOD   1179
C     KID      SID NO. IN LIBRARY                                       XSMOD   1180
C     IBLK     BLOCK NO.                                                XSMOD   1181
C     IERR     ERROR FLAG. NO ERROR=0.                                  XSMOD   1182
C     N        INPUT UNIT. 3=CARDS. 2=TAPE                              XSMOD   1183
C     ISID     NUCLIDE ID NO. READ FROM TAPE OR CARDS                   XSMOD   1184
C                                                                       XSMOD   1185
      COMMON /X/ X(4850)                                                X          2
      COMMON /CNTRL/ ISID(1),KID,ND,ITIDS(25),SDES(8),IR,ITIDR(50),NES, CNTRL      2
     1     ES(4000),IDR(50),ME(50),IRS(50),QR(50),IZ,MIXT,K2            CNTRL      3
      DIMENSION ITID(1),TDES(8),E(400),TK(4000),MD(400),PROB(10),       ANGD       2
     1     NMU(10)                                                      ANGD       3
      EQUIVALENCE (ITID(1), X(1)), (TDES(1),X(2)), (NED,X(10)),         ANGD       4
     1     (E(1),X(11)), (ISYS,X(411)), (A,X(412)), (LEGN,X(413)),      ANGD       5
     2     (KT,X(414)), (TK(1),X(415)), (MD(1),X(4415)), (NINC,X(4815)), ANGD      6
     3     (PROB(1),X(4816)), (NMU(1),X(4826))                          ANGD       7
C                                                                       ANGD       8
      COMMON /CDID/IA7,IA8,IA9,IBLNO(100),IDB(100),ICDS(100),DUM(9),LLL CDID       2
C                                                                       XSMOD   1190
C                                                                       XSMOD   1191
      COMMON /TYPE/ ITYPER(150)                                         TYPE       2
      DATA ITYPER/15*1,2,3,0,0,1,2,1,1,2,3,2,0,1,1,2,55*1,65*0/         TYPE       3
C                                                                       TYPE       4
      DATA I24 /100000000B/                                             XSMOD   1193
C                                                                       XSMOD   1194
C     CLEAR STORAGES                                                    XSMOD   1195
      DO 80 I=1,4850                                                    XSMOD   1196
   80 X(I)=0                                                            XSMOD   1197
C                                                                       XSMOD   1198
C     ITYPER DATA STATEMENT GIVES UK-LLL TYPES. LLL85 IS N,4N           XSMOD   1199
      IF(LLL .EQ. 1)ITYPER(85)=4                                        XSMOD   1200
C                                                                       XSMOD   1201
  100 ND=ND+1                                                           XSMOD   1202
      NCD=I=1                                                           XSMOD   1203
      READ(N,1) IRT,NER,A,ISYS,IDATA,NEL,IA7,IA8,IA9                    XSMOD   1204
    1 FORMAT(2(I11,1X),F11.6,1X,3(I11,1X),I3,I2,I3)                     XSMOD   1205
      ICHK=PRCT(1,IBLK,ISID)                                            XSMOD   1206
      IF(ICHK .NE. 0)GO TO 1000                                         XSMOD   1207
      IF(IRT .EQ. IDB(IBLK))GO TO 110                                   XSMOD   1208
      PRINT 6, IDB(IBLK),IRT,IBLK,ISID                                  XSMOD   1209
    6 FORMAT(//* READAD. REACTION TYPE DOES NOT MATCH THAT IN BLOCK0. IDB=  XSMOD 1210
     1 *I4* IRT=*I4* BLOCK=*I4* NIN=*I5)                                XSMOD   1211
      GO TO 1000                                                        XSMOD   1212
C     GET TWO-DIGIT REACTION NUMBER AND NO. OF NEUTRONS                 XSMOD   1213
  110 NRT=MOD(IRT,1000)                                                 XSMOD   1214
      NN=ITYPER(NRT)                                                    XSMOD   1215
C     CHECK ISYS FLAG AGAINST CONVENTION. FOR LLL CHANGE IF INCORRECT   XSMOD   1216
      IF(NRT .EQ. 2)GO TO 115                                           XSMOD   1217
      IF(NRT .GE. 5 .AND. NRT .LE. 14)GO TO 115                         XSMOD   1218
      IF(LLL .EQ. 1 .AND. NRT .GE. 31 .AND. NRT .LE. 79)GO TO 115       XSMOD   1219
      IF(ISYS .EQ. 2)GO TO 130                                          XSMOD   1220
      PRINT 14, ISYS,NRT,IBLK,ISID,LLL                                  XSMOD   1221
   14 FORMAT(//* READAD. ISYS DISCREPANCY. ISYS=*I2* REACTION=*I3* BLOCK=*  XSMOD 1222
     1 I5* NIN=*I5* LLL=*I2)                                            XSMOD   1223
      IF(LLL .EQ. 1)ISYS=2                                              XSMOD   1224
      PRINT 22, ISYS                                                    XSMOD   1225
   22 FORMAT(* ISYS CHANGED TO *I2)                                     XSMOD   1226
```

```
      GO TO 130                                                       XSMOD
  115 IF(ISYS .EQ. 1)GO TO 130                                        XSMOD
      PRINT 14, ISYS,NRT,IBLK,ISID,LLL                                XSMOD
      IF(LLL .EQ. 1)ISYS=1                                            XSMOD
      PRINT 22, ISYS                                                  XSMOD
  130 IF(NRT .NE. 2)GO TO 150                                         XSMOD
      IF(ND .EQ. 1)GO TO 140                                          XSMOD
      PRINT 2,IRT,IBLK,ISID                                           XSMOD
    2 FORMAT(//* READAD. 1ST ANG D NOT FOR ELASTIC. REACTION=*I5,* BLOCK=*  XSMOD
    1   I5,* NIN=*I5)                                                 XSMOD
      GO TO 1000                                                      XSMOD
C     FORM TID                                                        XSMOD
  140 ITIDS(1)=ITID=KID                                               XSMOD
      GO TO 160                                                       XSMOD
  150 ITIDS(ND)=KID*100+ NRT                                          XSMOD
C                                                                     XSMOD
C     CHOOSE APPROPRIATE READ                                         XSMOD
  160 IF(IDATA .LT. 2)GO TO 200                                       XSMOD
      IF(IDATA .EQ. 2)GO TO 400                                       XSMOD
      IF(IDATA .EQ. 20 .OR. IDATA .EQ. 21)GO TO 600                   XSMOD
      PRINT 4, IDATA,IRT,IBLK,NIN                                     XSMOD
    4 FORMAT(//* READAD. ERROR IN AD TYPE SPEC. IDATA=*I5* REACTION=*I5,  XSMOD
    1   * BLOCK=*I5* NIN=*I5)                                         XSMOD
      GO TO 1000                                                      XSMOD
C                                                                     XSMOD
C     READ MU,T PAIRS                                                 XSMOD
  200 KT=1                                                            XSMOD
      LEGN=0                                                          XSMOD
  220 READ(N,3) ERL,ERH,NCER,NDIS,PROB(1),NMU(1),IA7,IA8,IA9          XSMOD
    3 FORMAT(2(F11.6,1X),2(I11,1X),F11.6,1X,I11,1X,I3,I2,I3)          XSMOD
      KCER=1                                                          XSMOD
      NCD=NCD+1                                                       XSMOD
      ICHK=PRCT(NCD,IBLK,ISID)                                        XSMOD
      IF(ICHK .NE. 0)GO TO 1000                                       XSMOD
      IF(NCD .NE. 2)GO TO 250                                         XSMOD
C     CHECK IF NEED TO ADD LOWEST ENERGY                              XSMOD
      IF(ERL-2.5E-08)230,230,240                                      XSMOD
  230 E(1)=ERL                                                        XSMOD
      GO TO 270                                                       XSMOD
  240 E(1)=1.0E-11                                                    XSMOD
      E(2)=ERL                                                        XSMOD
      I=2                                                             XSMOD
C     PACK LOWER INDEX. ENTER ISOTROPIC DISTR FOR LOWEST ENERGY       XSMOD
      MD(1)=KT*I24                                                    XSMOD
      TK(1)=-1.0                                                      XSMOD
      TK(2)=TK(4)=0.5                                                 XSMOD
      TK(3)=1.0                                                       XSMOD
      KT=4                                                            XSMOD
C     PACK UPPER INDEX AND INCREASE KT POINTER                        XSMOD
      MD(1)=MD(1)+ KT                                                 XSMOD
      KT=5                                                            XSMOD
      GO TO 270                                                       XSMOD
C                                                                     XSMOD
C     AVERAGE ENERGY IF RANGE GIVEN                                   XSMOD
  250 IF(ERH .NE. 0)GO TO 260                                         XSMOD
      E(I)=ERL                                                        XSMOD
      GO TO 270                                                       XSMOD
  260 E(I)=0.5*(ERL+ ERH)                                             XSMOD
C     PACK LOWER INDEX AND CHECK FOR MULTIPLE DISTRIBUTIONS           XSMOD
  270 MD(I)=KT*I24                                                    XSMOD
      IF(NDIS .EQ. 1)GO TO 275                                        XSMOD
      CALL MULTI(KID,IBLK,N,IERR,NCD,NDIS)                            XSMOD
      IF(TK(KT-2) .EQ. 1)GO TO 290                                    XSMOD
      PRINT 8, TK(KT-2),KT,IRT,IBLK,ISID,NDIS                         XSMOD
    8 FORMAT(* READAD. AFTER MULTI LAST MU NOT=1. MU=*E12.5* KT=*I5* REAC  XSMOD
    1 TION=*I5* BLOCK=*I4* NIN=*I5* NDIS=*I2)                         XSMOD
      IERR=2                                                          XSMOD
      GO TO 290                                                       XSMOD
C     READ MU,T PAIRS INTO TK BLOCK                                   XSMOD
  275 KDS=NMU/3                                                       XSMOD
      NMUR=MOD(NMU,3)                                                 XSMOD
```

```
        IF(NMUR . NE.  0)KDS=KDS+ 1
        DO  280  J=1,KDS
        KL=KT+ 5
        READ(N,5)  (TK(K),K=KT,KL),IA7,IA8,IA9
 5      FORMAT(6(F11. 6,1X),I3,I2,I3)
        KCER=KCER+ 1
        NCD=NCD+ 1
        ICHK=PRCT(NCD,IBLK,ISID)
        IF(ICHK  . NE.  0)GO  TO  1000
        KT=KT+ 6
 280    CONTINUE
        IF(NMUR  . EQ.  0)GO  TO  290
C       ADJUST  POINTER  IF  LAST  CARD  NOT  FULL  AND  PACK  UPPER  INDEX
        KTADJ=2*(3- NMUR)
        KT=KT- KTADJ
 290    MD(I)=MD(I)+ KT- 1
C
C       ON  LLL  TAPE  AD  GIVEN  FOR  1ST  NEUTRON  ONLY.  ON  UK  SKIP  AD  GIVEN  FOR
C       2ND  AND  3RD  NEUTRONS
C
        IF(LLL  . EQ.  1)GO  TO  305
        IF(NN  . EQ.  1)GO  TO  305
        PRINT  12,IRT,IBLK,ISID,NN
 12     FORMAT(*  READAD. SKIPPING  AD  FOR  NTH  NEUTRONS. REACTION=*I5*  BLOCK=*
 1      I4*  NIN=*I5*  TYPE=*I2)
        READ(N,3)  ERL,ERH,NCER,DUM(1),DUM(2),DUM(3),IA7,IA8,IA9
        KCER=KCER+ 1
        NCD=NCD+ 1
        ICHK=PRCT(NCD,IBLK,ISID)
        IF(ICHK  . NE.  0)GO  TO  1000
        KCER=KCER+ 1
        DO  295  J=KCER,NCER
        NCD=NCD+ 1
 295    READ(N,7)
 7      FORMAT(1X)
C
C       NORMALIZE  ANG  DISTR
C
 305    JB=MD(I)/124
        JE=MD(I)- JB*124- 3
        S=0
        DO  310  J1=JB,JE,2
        S=0. 5*(TK(J1 + 1)+ TK(J1 + 3))*(TK(J1 + 2)- TK(J1))+ S
 310    CONTINUE
        IF(ABS(S- 1. 0)  . LT.  . 02)GO  TO  320
        PRINT  10,  E(I),S,IRT,IBLK,ISID
 10     FORMAT(*  NORMALIZING  ANG  DISTR  FOR  E=*E12. 5*  NORM  FACTOR=*E12. 5*  R
 1      EACTION=*I5*  BLOCK=*I4*  NIN=*I5)
 320    JB=JB+ 1
        JE=JE+ 3
        DO  330  J1=JB,JE,2
        TK(J1)=TK(J1)/S
 330    CONTINUE
C       CHECK  FOR  END  OF  BLOCK
        IF(NCD  . EQ.  ICDS(IBLK))GO  TO  340
        I=I+ 1
        GO  TO  220
C
C       ENTER  HERE  FROM  BOTH  LEGENDRE  AND  MU,T  PAIRS  TO  WRAP  UP  ANG  DIST
 340    NED=I
        KT=KT- 1
        NINC=40
C
C       DEFINE  DT  ARRAY  FOR  ANGULAR  DISTRIBUTION
        N. ANGD(ND)=F. ITIDS(ND)/F. SDES/F. NED/(F. E(J),J=1,NED)/F. ISYS/F. A/
 1          F. LEGN/F. KT/(F. TK(J),J=1,KT)/(F. MD(J),J=1,NED)/F. NINC/
        GO  TO  2000
C
C       READ  LEGENDRE  REPRESENTATION  HERE.
C       READ  IN  ENTIRE  BLOCK  USING  ES  BLOCK  AS  TEMPORARY  STORAGE.
C       ON  INPUT  TAPE  ENERGIES  AND  CORRESPONDING  COEFFS  ARE  GIVEN  FOR  EACH
C       L.  REARRANGE  IN  LIBRARY  SO  THAT  COEFFS  CORRESPONDING  TO  L=1,L=2,
```

```
C          ETC. ARE GIVEN FOR EACH E.
C
  400      N. ES=F. ES
           KT1=1
           DO 445 J=1,NER
           L=1
  405      READ(N,9)ERL,ERH,NCER,LMAX,INT,NE,IA7,IA8,IA9
    9      FORMAT(2(F11. 6,1X),4(I11,1X),I3,I2,I3)
           KCER=1
           NCD=NCD+ 1
           ICHK=PRCT(NCD,IBLK,ISID)
           IF(ICHK . NE. 0)GO TO 1000
           IF(NCD . EQ. 2)GO TO 415
           GO TO 430
C          CHECK IF NEED TO ADD LOWEST ENERGY.
  415      IF(ERL- 2. 5E- 08)420,420,425
  420      E(1)=ERL
           KT=1
           I=1
           GO TO 430
  425      E(1)=1. 0E- 11
           E(2)=ERL
           I=2
           TK(1)=1
           TK(2)=0
           KT=3
           MD(1)=00000000000100000002B
  430      IF(LMAX . NE. 0)GO TO 433
C          IF LMAX=0 DSTRBN IS ISOTROPIC
           ES(KT1)=ERL
           ES(KT1+ 1)=0
           KT1=KT1+ 2
           GO TO 442
  433      IS=I
C          NE IS NO. OF ENERGIES FOR WHICH COEFFS ARE GIVEN FOR THIS L.
           KDS=NE/3
           NNER=MOD(NE,3)
           IF(NNER . NE. 0)KDS=KDS+ 1
           DO 435 K=1,KDS
           JL=KT1+ 5
C          READ IN ENERGY- COEFF PAIRS
           READ(N,5) (ES(J),J=KT1,JL),IA7,IA8,IA9
           KCER=KCER+ 1
           NCD=NCD+ 1
           ICHK=PRCT(NCD,IBLK,ISID)
           IF(ICHK . NE. 0)GO TO 1000
           KT1=KT1+ 6
  435      CONTINUE
           IF(NNER . EQ. 0)GO TO 440
C          ADJUST POINTER IF LAST CARD WAS NOT FULL
           KTADJ=2*(3- NNER)
           KT1=KT1- KTADJ
  440      IF(L . EQ. LMAX)GO TO 442
           L=L+ 1
C          GO BACK AND READ FOR NEXT L.
           GO TO 405
C          HAVE READ FOR ALL L.
  442      IF(NN . EQ. 1)GO TO 445
           IF(LLL . EQ. 1)GO TO 445
           PRINT 18, IRT,IBLK,NIN,NN
   18      FORMAT(* READAD. SKIP LEGENDRE AD FOR NTH NEUTRONS. REACTION=*I5* BL
    1      OCK=*I4* NIN=*I5* TYPE=*I2)
           READ(N,3) ERL,ERH,NCER,DUM(1),DUM(2),DUM(3),IA7,IA8,IA9
           KCER=KCER+ 1
           NCD=NCD+ 1
           ICHK=PRCT(NCD,IBLK,ISID)
           IF(ICHK . NE. 0)GO TO 1000
           KCER=KCER+ 1
           DO 443 JJ=KCER,NCER
           NCD=NCD+ 1
  443      READ(N,7)
  445      CONTINUE
```

```
C          SET KT1 TO POINT TO LAST E
           KT1=KT1- 2
C
C          FILL E BLOCK
C
  450      IF(ABS(ES(1)- E(I))- . 00001)460,460,455
  455      PRINT 16, ES(1),E(I),IRT,IBLK,NIN
   16      FORMAT(* READAD. POSSIBLE INDEX OR INPUT ERROR. ES1=*E12. 5* E=*E12. 5
     1     * I=*I5,* REACTION=*I5* BLOCK=*I5* NIN=*I5)
           GO TO 1000
C          CHECK EACH ENERGY IN TEMPORARY ES BLOCK AGAINST THOSE ALREADY
C          TABULATED IN E BLOCK. IF NOT IN E BLOCK, PUT IT IN AND INCREASE I.
  460      DO 480 K=1,KT1,2
           DO 470 KK=1,I
           IF(ABS(ES(K)- E(KK))- . 00001)480,480,470
  470      CONTINUE
           E(I+ 1)=ES(K)
           I=I+ 1
  480      CONTINUE
C
C          FILL TK BLOCK
C
           KS=1
C          FOR EACH ENERGY IN E,FIND AND STORE IN TK CORRESPONDING COEFFS AND
C          KEEP COUNT NF.
           DO 520 II=IS,I
           NF=0
           DO 510 K=KS,KT1,2
           IF(ABS(ES(K)- E(II))- . 00001)500,500,510
  500      NF=NF+ 1
           TK(KT+ NF)=ES(K+ 1)
  510      CONTINUE
C          HAVE FOUND ALL COEFFS FOR 1 E. PACK LOWER INDEX AND INCREASE KT.
           MD(II)=KT*I24
           KT=KT+ NF+ 1
           IF(NF . EQ. 1)GO TO 519
C          REMOVE ZERO COEFFS AT HIGHEST L
           KA=KT
           DO 515 KB=1,NF
           IF(TK(KA- KB))519,514,519
  514      KT=KT- 1
           NF=NF- 1
  515      CONTINUE
  519      KS=KS+ 2
C          PACK UPPER INDEX AND STORE NO. OF COEFFS
           MD(II)=MD(II)+ KT- 1
           TK(KT- NF- 1)=NF
  520      CONTINUE
C
C          RESTORE ES BLOCK
           F. ES=N. ES
           LEGN=I
           GO TO 340
  600      PRINT 601
  601      FORMAT(* CANNOT READ MIXED REPRESENTATION YET. *)
  700      NCDS=ICDS(IBLK)
           DO 800 I=2,NCDS
  800      READ(N,801)
  801      FORMAT(1X)
           GO TO 2000
C          ERROR RETURN. SET TO SKIP REMAINING CARDS.
 1000      IERR=NCD + 1
C          STANDARD RETURN
 2000      RETURN
           END
           SUBROUTINE READNU(KID,IBLK,N,IERR)
C          READS ENERGY- NU PAIRS FOR FISSION REACTIONS
C          KID        SID NO. IN LIBRARY
C          IBLK       BLOCK NO.
C          IERR       ERROR FLAG. NO ERROR=0.
C          N          INPUT UNIT. 3=CARDS. 2=TAPE
C          ISID       NUCLIDE ID NO. READ FROM TAPE OR CARDS
```

```
C                                                                            XSMOD   1514
      COMMON  /CDID/IA7,IA8,IA9,IBLNO(100),IDB(100),ICDS(100),DUM(9),LLL     CDID       2
      COMMON /CNTRL/ ISID(1),KID,ND,ITIDS(25),SDES(8),IR,ITIDR(50),NES,      CNTRL      2
     1    ES(4000),IDR(50),ME(50),IRS(50),QR(50),IZ,MIXT,K2                  CNTRL      3
      COMMON  /X/ X(4850)                                                    X          2
      DIMENSION  ESJ(3000)                                                   ESJ        2
      EQUIVALENCE  (ESJ(1),X(1))                                             ESJ        3
C                                                                            XSMOD   1519
      NCD=1                                                                  XSMOD   1520
      READ(N,1) IRT,NER,DUM(3),INT,D1,D2,IA7,IA8,IA9                         XSMOD   1521
    1 FORMAT(6(I11,1X),I3,I2,I3)                                             XSMOD   1522
      ICHK=PRCT(1,IBLK,ISID)                                                 XSMOD   1523
      IF(ICHK .NE. 0)GO TO 1000                                              XSMOD   1524
      IF(IRT .EQ. IDB(IBLK))GO TO 100                                        XSMOD   1525
      PRINT 2, IDB(IBLK),IRT,IBLK,ISID                                       XSMOD   1526
    2 FORMAT(//* READNU. REACTION TYPE DOES NOT MATCH THAT IN BLOCK0. IDB=   XSMOD   1527
     1 *I4* IRT= *I4* BLOCK= *I4* NIN=*I5)                                   XSMOD   1528
      GO TO 1000                                                             XSMOD   1529
C     J WILL POINT TO FIRST UNUSED WORD IN ESJ BLOCK.                        XSMOD   1530
  100 J=IRS(IR)+1                                                            XSMOD   1531
      IF(IRT .EQ. 4018 .OR. IRT .EQ. 4019)GO TO 103                          XSMOD   1532
      PRINT 4, IRT,IBLK,ISID                                                 XSMOD   1533
    4 FORMAT(//* READNU. WILL STORE IN ESJ BLOCK ONLY REACTIONS 4018 OR      XSMOD   1534
     1 4019. WILL SKIP REACTION=*I5* BLOCK=*I4* NIN=*I5)                     XSMOD   1535
      IL=ICDS(IBLK)                                                          XSMOD   1536
      DO 102 I=2,IL                                                          XSMOD   1537
  102 READ(N,7)                                                              XSMOD   1538
    7 FORMAT(1X)                                                             XSMOD   1539
      GO TO 2000                                                             XSMOD   1540
C     STORE REACTION IDENTIFIER IN ESJ.                                      XSMOD   1541
  103 ESJ(J)=IRT                                                             XSMOD   1542
      J=J+1                                                                  XSMOD   1543
  105 DO 200 I=1,NER                                                         XSMOD   1544
      READ(N,3) ERL,ERH,T,NCER,NXS,IA6,IA7,IA8,IA9                           XSMOD   1545
    3 FORMAT(3(F11.6,1X),3(I11,1X),I3,I2,I3)                                 XSMOD   1546
      NCD=NCD+1                                                              XSMOD   1547
      ICHK=PRCT(NCD,IBLK,ISID)                                               XSMOD   1548
      IF(ICHK .NE. 0)GO TO 1000                                              XSMOD   1549
      DO 150 II=2,NCER                                                       XSMOD   1550
      JL=J+5                                                                 XSMOD   1551
C     READ IN ENERGY-NU PAIRS                                                XSMOD   1552
      READ(N,5) (ESJ(J1),J1=J,JL),IA7,IA8,IA9                                XSMOD   1553
    5 FORMAT(6(F11.6,1X),I3,I2,I3)                                           XSMOD   1554
      NCD=NCD+1                                                              XSMOD   1555
      ICHK=PRCT(NCD,IBLK,ISID)                                               XSMOD   1556
      IF(ICHK .NE. 0)GO TO 1000                                              XSMOD   1557
      J=JL+1                                                                 XSMOD   1558
  150 CONTINUE                                                               XSMOD   1559
      ITEMS=2*NXS                                                            XSMOD   1560
      MITEMS=MOD(ITEMS,6)                                                    XSMOD   1561
      IF(MITEMS .EQ. 0)GO TO 200                                             XSMOD   1562
      J=J-6+MITEMS                                                           XSMOD   1563
  200 CONTINUE                                                               XSMOD   1564
C     STORE END OF BLOCK POINTER                                             XSMOD   1565
      ESJ(J)=-1.0                                                            XSMOD   1566
      IRS(IR)=J                                                              XSMOD   1567
C     DEFINE DT ARRAY FOR ESJ                                                XSMOD   1568
      N. ESJ(IR)=(F. ESJ(J1),J1=1,J)/                                        XSMOD   1569
      GO TO 2000                                                             XSMOD   1570
C                                                                            XSMOD   1571
 1000 IERR=NCD+1                                                             XSMOD   1572
 2000 RETURN                                                                 XSMOD   1573
      END                                                                    XSMOD   1574
      SUBROUTINE READSEC(KID,IBLK,N,IERR)                                    XSMOD   1575
C     READS SECONDARY ENERGIES FROM UK FORMAT TAPE OR CARDS                  XSMOD   1576
C                                                                            XSMOD   1577
C     KID     SID NO. IN LIBRARY                                             XSMOD   1578
C     IBLK    BLOCK NO.                                                      XSMOD   1579
C     IERR    ERROR FLAG. NO ERROR=0.                                        XSMOD   1580
C     N       INPUT UNIT. 3=CARDS. 2=TAPE                                    XSMOD   1581
C     ISID   NUCLIDE ID NO. READ FROM TAPE OR CARDS                          XSMOD   1582
C                                                                            XSMOD   1583
      COMMON  /CDID/IA7,IA8,IA9,IBLNO(100),IDB(100),ICDS(100),DUM(9),LLL     CDID       2
```

```
      COMMON /CNTRL/ ISID(1),KID,ND,ITIDS(25),SDES(8),IR,ITIDR(50),NES,   CNTRL      2
     1    ES(4000),IDR(50),ME(50),IRS(50),QR(50),IZ,MIXT,K2               CNTRL      3
      COMMON  /X/ X(4850)                                                 X          2
      DIMENSION  ESJ(3000)                                                ESJ        2
      EQUIVALENCE  (ESJ(1),X(1))                                          ESJ        3
      COMMON  /TYPE/  ITYPER(150)                                         TYPE       2
      DATA ITYPER/15*1,2,3,0,0,1,2,1,1,2,3,2,0,1,1,2,55*1,65*0/           TYPE       3
C                                                                         TYPE       4
C     CLEAR STORAGE                                                       XSMOD   1589
      DO 80 I=1,4850                                                      XSMOD   1590
   80 X(I)=0                                                              XSMOD   1591
      NCD=1                                                               XSMOD   1592
C     FOR LLL TAPES CHANGE TYPE FOR REACTION 85                           XSMOD   1593
      IF(LLL . EQ. 1)ITYPER(85)=4                                         XSMOD   1594
  100 READ(N,1) IRT,NER,ISYSM,DUM(4),DUM(5),DUM(6),IA7,IA8,IA9            XSMOD   1595
    1 FORMAT(6(I11,1X),I3,I2,I3)                                          XSMOD   1596
      ICHK=PRCT(NCD,IBLK,ISID)                                            XSMOD   1597
      IF(ICHK . NE. 0)GO TO 1000                                          XSMOD   1598
      IF(IRT . EQ. IDB(IBLK))GO TO 103                                    XSMOD   1599
      PRINT 10,IDB(IBLK),IRT,IBLK,ISID                                    XSMOD   1600
   10 FORMAT(//* READSEC. REACTION TYPE DOES NOT MATCH THAT IN BLOCK0. IDB XSMOD   1601
     1 =*I5* IRT=*I5* BLOCK=*I4* NIN=*I5)                                 XSMOD   1602
      GO TO 1000                                                          XSMOD   1603
  103 IF(ISYSM . EQ. 0)GO TO 105                                          XSMOD   1604
      PRINT 2, IRT,IBLK,ISID                                              XSMOD   1605
    2 FORMAT(/* SECONDARY ENERGY AND SCATTERING ANGLE GIVEN IN CENTER OF  XSMOD   1606
     1  MASS SYSTEM. REACTION=*I5* BLOCK=*I4* NIN=*I4)                    XSMOD   1607
C     GET REACTION NO.  AND NO. OF NEUTRONS                               XSMOD   1608
  105 NRT=MOD(IRT,1000)                                                   XSMOD   1609
      NN=MAX0(ITYPER(NRT),1)                                              XSMOD   1610
      ESJ(1)=NRT                                                          XSMOD   1611
      J=2                                                                 XSMOD   1612
C     LOOP FOR ENERGY RANGES                                              XSMOD   1613
      DO 500 II=1,NER                                                     XSMOD   1614
      SUMW=0. 0                                                           XSMOD   1615
C     LOOP FOR NEUTRONS                                                   XSMOD   1616
      DO 450 III=1,NN                                                     XSMOD   1617
      READ(N,3) ERL,ERH,NCER,NL,W,LAW,IA7,IA8,IA9                         XSMOD   1618
    3 FORMAT(2(F11. 6,1X),2(I11,1X),F11. 6,1X,I11,1X,I3,I2,I3)            XSMOD   1619
      NCD=NCD+ 1                                                          XSMOD   1620
      ICHK=PRCT(NCD,IBLK,ISID)                                            XSMOD   1621
      IF(ICHK . NE. 0)GO TO 1000                                          XSMOD   1622
      SUMW=SUMW+ W                                                        XSMOD   1623
C     LOOP FOR LAWS                                                       XSMOD   1624
  120 DO 400 I=1,NL                                                       XSMOD   1625
      IF(I . EQ. 1 . AND. III . EQ. 1)GO TO 150                           XSMOD   1626
      IF(I . EQ. 1)GO TO 155                                              XSMOD   1627
      READ(N,3) ERL,ERH,NCER,NL,W,LAW,IA7,IA8,IA9                         XSMOD   1628
      NCD=NCD+ 1                                                          XSMOD   1629
      ICHK=PRCT(NCD,IBLK,ISID)                                            XSMOD   1630
      IF(ICHK . NE. 0)GO TO 1000                                          XSMOD   1631
      SUMW=SUMW+ W                                                        XSMOD   1632
      GO TO 155                                                           XSMOD   1633
  150 ESJ(J)=ERL                                                          XSMOD   1634
  155 ESJ(J+ 1)=W                                                         XSMOD   1635
      ESJ(J+ 2)=LAW                                                       XSMOD   1636
      J=J+ 3                                                              XSMOD   1637
C     GO TO READ INDICATED LAW.  RETURN TO 200.                           XSMOD   1638
      GO TO(5100,5200,5300,5400,5500,5600,5700,5800,5900,6000)LAW         XSMOD   1639
C     END OF LAW                                                          XSMOD   1640
  200 ESJ(J)=- 4. 0                                                       XSMOD   1641
  400 CONTINUE                                                            XSMOD   1642
      IF(LLL . EQ. 1)GO TO 455                                            XSMOD   1643
  450 CONTINUE                                                            XSMOD   1644
C     END OF NEUTRON                                                      XSMOD   1645
  455 ESJ(J+ 1)=- 5. 0                                                    XSMOD   1646
C     END OF ENERGY                                                       XSMOD   1647
      ESJ(J+ 2)=- 2. 0                                                    XSMOD   1648
      J=J+ 3                                                              XSMOD   1649
      ISUMW=SUMW                                                          XSMOD   1650
      IF(LLL . EQ. 1)ISUMW=ISUMW*NN                                       XSMOD   1651
```

```
          IF(ISUMW .EQ. NN)GO TO 500
          PRINT 4, IRT,IBLK,ISID,NN,ISUMW
    4     FORMAT(/* READSEC. SUM OF WEIGHTS NOT EQUAL NO. NEUTRONS OUT. REACT
     1    ION=*I5* BLOCK=*I4* NIN=*I5* NN=*I2* ISUMW=*I2)
  500     CONTINUE
C
C         END OF REACTION
          ESJ(J)=-1.0
C         STORE NO. OF ITEMS IN ESJ BLOCK.
          IRS(IR)=J
C
C         DEFINE DT ARRAY FOR ESJ IF NO 4000 BLOCK FOLLOWING.
          INEXT=IDB(IBLK+1)
          IF(INEXT .EQ. 4018 .OR. INEXT .EQ. 4019)GO TO 2000
          N.ESJ(IR)=(F.ESJ(J1),J1=1,J)/
          GO TO 2000
 1000     IERR=NCD+1
 2000     RETURN
C
C         LAW 1
 5100     M=2
 5105     JL=J+4
          JIN=J
          READ(N,9) NV,(ESJ(K),K=J,JL),IA7,IA8,IA9
    9     FORMAT(I11,1X,5(F11.6,1X),I3,I2,I3)
          NCD=NCD+1
          ICHK=PRCT(NCD,IBLK,ISID)
          IF(ICHK .NE. 0)GO TO 1000
          J=J-1
          ITEMS=M*NV
          MITEMS=MOD(ITEMS,6)
          IF(ITEMS .LT. 5)GO TO 5120
          KDS=ITEMS/6
          DO 5110 K=1,KDS
          J=J+6
          JL=J+5
          READ(N,11) (ESJ(J1),J1=J,JL),IA7,IA8,IA9
   11     FORMAT(6(F11.6,1X),I3,I2,I3)
          NCD=NCD+1
          ICHK=PRCT(NCD,IBLK,ISID)
          IF(ICHK .NE. 0)GO TO 1000
 5110     CONTINUE
 5120     J=J+MITEMS+1
C         FOR LAWS 1,2 ONLY CHECK FOR SUM OF PROBABILITIES=1.
          IF(LAW .NE. 1 .AND. LAW .NE. 2)GO TO 5150
          JB=JIN+M-1
          SUM=0.0
          DO 5130 K=JB,J,M
 5130     SUM=SUM + ESJ(K)
          GO TO 5155
C         FOR LAWS 3,4,5,6 CHECK BY TRAPEZOIDAL RULE WHETHER INTEGRAL=1.
 5150     SUM=0.0
          SUM=0.0
          JE=J-3
          DO 5151 K=JIN,JE,2
 5151     SUM=SUM + 0.5*(ESJ(K+3)+ESJ(K+1))*(ESJ(K+2)-ESJ(K))
 5155     IF(ABS(1.0-SUM)-.001)200,200,5160
 5160     PRINT 14, SUM,LAW,ERL,IRT,IBLK,ISID
   14     FORMAT(/* READSEC. SUM=*E12.5* FOR LAW*I2* ENERGY=*E12.5* REACTION
     1    =*I5* BLOCK=*I4* NIN=*I4//)
          GO TO 200
C
C         LAW 2
 5200     M=3
          GO TO 5105
C         LAW 3
 5300     IF(NN .EQ. 1)GO TO 5305
          PRINT 12, IRT,NN
   12     FORMAT(//* USING TRUE LAW3 FOR REACTION *I4* WITH *I1* NEUTRONS OU
     1    T. WRITE-UPS NOT CLEAR ON WHETHER TO MULTIPLY WEIGHT BY NN. *//)
 5305     GO TO 5100
C         LAW 4
```

```
 5400   GO TO 5100                                                      XSMOD  1724
C       LAW 5                                                           XSMOD  1725
 5500   GO TO 5100                                                      XSMOD  1726
C       LAW 6                                                           XSMOD  1727
 5600   GO TO 5100                                                      XSMOD  1728
C       LAW 7                                                           XSMOD  1729
 5700   JL=J+ 3                                                         XSMOD  1730
        READ(N,13) (ESJ(J1),J1=J,JL),IA7,IA8,IA9                        XSMOD  1731
   13   FORMAT(4(F11. 6,1X),24X,I3,I2,I3)                               XSMOD  1732
        NCD=NCD+ 1                                                      XSMOD  1733
        ICHK=PRCT(NCD,IBLK,ISID)                                        XSMOD  1734
        IF(ICHK . NE. 0)GO TO 1000                                      XSMOD  1735
        J=JL+ 1                                                         XSMOD  1736
        GO TO 200                                                       XSMOD  1737
C                                                                       XSMOD  1738
C       LAW 8                                                           XSMOD  1739
 5800   IF(LLL . EQ. 1)GO TO 5805                                       XSMOD  1740
        PRINT 8, IBLK,ISID                                              XSMOD  1741
    8   FORMAT(//* NO LAW 8 FOR UKDATA. BLOCK=*I4* NIN=*I5)             XSMOD  1742
        GO TO 1000                                                      XSMOD  1743
C                                                                       XSMOD  1744
C       CHANGE LAW NUMBER TO 3 AS IN S4.                                XSMOD  1745
C       BREAK UP LAW 8 FORMAT INTO SEVERAL LAW 3S.                      XSMOD  1746
C                                                                       XSMOD  1747
 5805   J=J- 3                                                          XSMOD  1748
C       SET NEI=1 TO GET STARTED. SET TO ITS REAL VALUE WHEN CARD IS READ. XSMOD  1749
        NEI=1                                                           XSMOD  1750
        DO 5850 II=1,NEI                                                XSMOD  1751
 5810   READ(N,17) NEI,EN,NPEO,DUM(4),DUM(5),DUM(6),IA7,IA8,IA9         XSMOD  1752
   17   FORMAT(I11,1X,F11. 6,1X,I11,1X,3(F11. 6,1X),I3,I2,I3)           XSMOD  1753
        NCD=NCD+ 1                                                      XSMOD  1754
        ICHK=PRCT(NCD,IBLK,ISID)                                        XSMOD  1755
        IF(ICHK . NE. 0)GO TO 1000                                      XSMOD  1756
        ESJ(J)=EN                                                       XSMOD  1757
        ESJ(J+ 1)=NN*W                                                  XSMOD  1758
        ESJ(J+ 2)=3. 0                                                  XSMOD  1759
        J=J+ 3                                                          XSMOD  1760
        ITEMS=2*NPEO                                                    XSMOD  1761
        ESJ(J)=DUM(4)                                                   XSMOD  1762
        ESJ(J+ 1)=DUM(5)/NN                                             XSMOD  1763
        IF(DUM(6) . NE. 0)GO TO 5815                                    XSMOD  1764
        J=J+ 2                                                          XSMOD  1765
        GO TO 5810                                                      XSMOD  1766
 5815   ESJ(J+ 2)=DUM(6)                                                XSMOD  1767
        J=J+ 3                                                          XSMOD  1768
        IT3=ITEMS+ 3                                                    XSMOD  1769
        KDS=IT3/6                                                       XSMOD  1770
        DO 5830 KK=1,KDS                                                XSMOD  1771
        JL=J+ 5                                                         XSMOD  1772
        READ(N,11) (ESJ(J1),J1=J,JL),IA7,IA8,IA9                        XSMOD  1773
        NCD=NCD+ 1                                                      XSMOD  1774
        ICHK=PRCT(NCD,IBLK,ISID)                                        XSMOD  1775
        IF(ICHK . NE. 0)GO TO 1000                                      XSMOD  1776
        IF(NN . EQ. 1)GO TO 5825                                        XSMOD  1777
        DO 5820 J1=J,JL,2                                               XSMOD  1778
C       MUST DIVIDE PROBS BY NN SINCE DATA IS COMPOSITE FOR ALL NEUTRONS. XSMOD  1779
        ESJ(J1)=ESJ(J1)/NN                                              XSMOD  1780
 5820   CONTINUE                                                        XSMOD  1781
 5825   J=JL+ 1                                                         XSMOD  1782
 5830   CONTINUE                                                        XSMOD  1783
C       ADJUST J POINTER                                                XSMOD  1784
        MIT3=MOD(IT3,6)                                                 XSMOD  1785
        IF(IT3 . EQ. 0)GO TO 5840                                       XSMOD  1786
        J=J- 6+ MIT3                                                    XSMOD  1787
 5840   IF(NCD . EQ. ICDS(IBLK))GO TO 200                               XSMOD  1788
C       IF ALL CARDS IN BLOCK NOT READ,PUT IN MARKERS AND READ NEXT ENERGY XSMOD  1789
        ESJ(J)=- 4. 0                                                   XSMOD  1790
        ESJ(J+ 1)=- 5. 0                                                XSMOD  1791
        ESJ(J+ 2)=- 2. 0                                                XSMOD  1792
        J=J+ 3                                                          XSMOD  1793
 5850   CONTINUE                                                        XSMOD  1794
```

```
      PRINT 16, NCD,ICDS(IBLK),IBLK,ISID                                 XSMOD    1795
   16 FORMAT(//* READSEC. ERROR IN LAW 8 DATA. NCD=*I5* ICDS=*I5* BLOCK=  XSMOD    1796
    1 *I4* NIN=*I5)                                                      XSMOD    1797
      GO TO 1000                                                         XSMOD    1798
C                                                                        XSMOD    1799
C     LAW 9                                                              XSMOD    1800
 5900 PRINT 6, IBLK,ISID                                                 XSMOD    1801
    6 FORMAT(//* NO LAW 9. BLOCK=*I4* NIN=*I5)                           XSMOD    1802
      GO TO 1000                                                         XSMOD    1803
C                                                                        XSMOD    1804
C     LAW 10                                                             XSMOD    1805
C     FOR LLL DATA MULTIPLY WEIGHT BY NO. OF NEUTRONS OUT                XSMOD    1806
 6000 IF(LLL .EQ. 1)ESJ(J-2)=NN*W                                        XSMOD    1807
      READ(N,15) DUM,IA7,IA8,IA9                                         XSMOD    1808
   15 FORMAT(12X,F11.6,49X,I3,I2,I3)                                     XSMOD    1809
      NCD=NCD+1                                                          XSMOD    1810
      ICHK=PRCT(NCD,IBLK,ISID)                                           XSMOD    1811
      IF(ICHK .NE. 0)GO TO 1000                                          XSMOD    1812
      ESJ(J)=SQRT(DUM)                                                   XSMOD    1813
      J=J+1                                                              XSMOD    1814
      GO TO 200                                                          XSMOD    1815
      END                                                                XSMOD    1816
      SUBROUTINE READ S(KID,IBLK,N,IERR)                                 XSMOD    1817
C     READS XS DATA FROM UK FORMAT TAPE OR CARDS                         XSMOD    1818
C                                                                        XSMOD    1819
C     KID       SID NO. IN LIBRARY                                       XSMOD    1820
C     IBLK      BLOCK NO.                                                XSMOD    1821
C     IERR      ERROR FLAG. NO ERROR=0.                                  XSMOD    1822
C     N         INPUT UNIT. 3=CARDS. 2=TAPE                              XSMOD    1823
C     ISID      NUCLIDE ID NO. READ FROM TAPE OR CARDS                   XSMOD    1824
C                                                                        XSMOD    1825
      COMMON /CDID/IA7,IA8,IA9,IBLNO(100),IDB(100),ICDS(100),DUM(9),LLL  CDID        2
      COMMON /CNTRL/ ISID(1),KID,ND,ITIDS(25),SDES(8),IR,ITIDR(50),NES,  CNTRL       2
    1     ES(4000),IDR(50),ME(50),IRS(50),QR(50),IZ,MIXT,K2              CNTRL       3
      COMMON /X/ X(4850)                                                 X           2
      DIMENSION SIG(4000)                                                XS          2
      EQUIVALENCE (SIG(1),X(1))                                          XS          3
C                                                                        XS          4
C                                                                        XSMOD    1830
      DATA I24 /100000000B/                                              XSMOD    1831
C                                                                        XSMOD    1832
C     INITIALIZE                                                         XSMOD    1833
      DO 90 I=1,4850                                                     XSMOD    1834
   90 X(I)=0                                                             XSMOD    1835
      J=2                                                                XSMOD    1836
      NCD =1                                                             XSMOD    1837
      IR=IR+1                                                            XSMOD    1838
C     BEGIN READING BLOCK                                                XSMOD    1839
      READ(N,1) IRT,NER,QR(IR),INT,D1,D2,IA7,IA8,IA9                     XSMOD    1840
    1 FORMAT(2(I11,1X),F11.6,1X,3(I11,1X),I3,I2,I3)                      XSMOD    1841
      ICHK=PRCT(1,IBLK,ISID)                                             XSMOD    1842
      IF(ICHK .NE. 0)GO TO 1000                                          XSMOD    1843
      IF(IRT .EQ. IDB(IBLK))GO TO 105                                    XSMOD    1844
      PRINT 2, IDB(IBLK),IRT,IBLK,ISID                                   XSMOD    1845
    2 FORMAT(//* READXS. REACTION TYPE DOES NOT MATCH THAT IN BLOCK0. IDB= XSMOD  1846
    1 *I4* IRT= *I4* BLOCK=*I4* NIN=*I5)                                 XSMOD    1847
      GO TO 1000                                                         XSMOD    1848
  105 IDR(IR)=IRT                                                        XSMOD    1849
C     CHECK INDICATOR FOR INTERPOLATION SCHEME. PRINT WARNING ONLY ONCE.  XSMOD   1850
      IF(INT .EQ. 0)GO TO 110                                            XSMOD    1851
      IF(IRT .NE. 1002)GO TO 110                                         XSMOD    1852
      PRINT 12, INT,IRT,IBLK,ISID                                        XSMOD    1853
   12 FORMAT(//* READXS. INTERPOLATION BETWEEN TABULATED POINTS IS NOT L  XSMOD   1854
    1 OG LOG. INDICATOR=*I2* REACTION=*I5* BLOCK=*I5* NIN=*I5/           XSMOD    1855
    2 * INDICATOR=1 MEANS SIGMA LINEAR IN E. INDICATOR=2 MEANS SIGMA LIN  XSMOD   1856
    3 EAR IN LETHARGY*/                                                  XSMOD    1857
    4 * THIS MESSAGE IS ONLY A WARNING. READ CONTINUES BUT NOTHING IS DON XSMOD   1858
    5 E ABOUT DIFFERENCE IN INTERPOLATION METHOD. *)                     XSMOD    1859
  110 READ(N,3)ERL,ERH,T,NCER,NXS,IA6,IA7,IA8,IA9                        XSMOD    1860
    3 FORMAT(3(F11.6,1X),3(I11,1X),I3,I2,I3)                             XSMOD    1861
      NCD =NCD +1                                                        XSMOD    1862
```

```
      ICHK=PRCT(NCD,IBLK,ISID)                                          XSMOD
      IF(ICHK .NE. 0)GO TO 1000                                         XSMOD
      IF(NXS .LE. 4000)GO TO 115                                        XSMOD
      PRINT 14, NXS,IBLK,ISID                                           XSMOD
   14 FORMAT(//* READXS. NO. OF XS GREATER THAN ES DIMENSION. NXS=*I5* BL XSMOD
     1OCK=*I4* NIN=*I5)                                                 XSMOD
      GO TO 1000                                                        XSMOD
  115 IF(IA6 .EQ. 1)GO TO 120                                           XSMOD
      PRINT 4, IA6,IRT,IBLK,ISID                                        XSMOD
    4 FORMAT(//* READXS. IA6=*I5* CANNOT HANDLE TEMPERATURE DEPENDENCE. XSMOD
     1REACTION=*I5* BLOCK=*I4* NIN=*I5)                                 XSMOD
      GO TO 1000                                                        XSMOD
C     SET UP ENERGY MESH FROM ELASTIC BLOCK.                            XSMOD
  120 IF(IRT .GT. 1002)GO TO 150                                        XSMOD
C     SET LOWEST ENERGY                                                 XSMOD
      ES(1)=1.0E-20                                                     XSMOD
      DO 130 II=2,NCER                                                  XSMOD
      NCD=NCD+1                                                         XSMOD
      READ(N,5) ES(J),SIG(J),ES(J+1),SIG(J+1),ES(J+2),SIG(J+2),IA7,IA8, XSMOD
     1     IA9                                                          XSMOD
    5 FORMAT(6(F11.6,1X),I3,I2,I3)                                      XSMOD
      ICHK=PRCT(NCD,IBLK,ISID)                                          XSMOD
      IF(ICHK .NE. 0)GO TO 1000                                         XSMOD
      J=J+3                                                             XSMOD
  130 CONTINUE                                                          XSMOD
C     ADJUST J POINTER IF LAST CARD READ WAS NOT FULL AND BECAUSE NEXT  XSMOD
C     ENERGY RANGE REPEATS HIGHEST E FROM RANGE JUST FINISHED           XSMOD
      L=MOD(NXS,3)                                                      XSMOD
      JADJ=3-L+1                                                        XSMOD
      IF(L .EQ. 0)JADJ=1                                                XSMOD
      J=J-JADJ                                                          XSMOD
C     CHECK FOR ALL CARDS READ FOR THIS BLOCK                           XSMOD
      IF(NCD .NE. ICDS(1))GO TO 110                                     XSMOD
C     SET HIGHEST E VALUE AND SIGMAS AT BOTH ENDS                       XSMOD
      J=J+1                                                             XSMOD
      ES(J)=1.0E03                                                      XSMOD
      SIG(1)=SIG(2)                                                     XSMOD
      SIG(J)=SIG(J-1)                                                   XSMOD
      NES=J                                                             XSMOD
C     PACK LOWER AND UPPER INDICES FOR THIS REACTION                    XSMOD
      IS=1                                                              XSMOD
      ME(IR)=IS*124+NES                                                 XSMOD
      GO TO 2000                                                        XSMOD
C     FOR REACTIONS OTHER THAN ELASTIC                                  XSMOD
  150 DO 220 II=2,NCER                                                  XSMOD
      NCD=NCD+1                                                         XSMOD
      READ(N,5) DUM(1),SIG(J),DUM(2),SIG(J+1),DUM(3),SIG(J+2),IA7,IA8,  XSMOD
     1     IA9                                                          XSMOD
      ICHK=PRCT(NCD,IBLK,ISID)                                          XSMOD
      IF(ICHK .NE. 0)GO TO 1000                                         XSMOD
C     COMPARE WITH MASTER ENERGY MESH TO FIND INDEX OF STARTING ENERGY  XSMOD
      IF(IA9 .NE. 3)GO TO 190                                           XSMOD
      DO 160 I=2,NES                                                    XSMOD
      IF(DUM(1)-ES(I))170,180,160                                       XSMOD
  160 CONTINUE                                                          XSMOD
  170 PRINT 6, DUM(1),ES(I),I,IRT,IBLK,ISID                             XSMOD
    6 FORMAT(//* READXS. FIRST ENERGY NOT IN MESH. ENERGY=*E11.6* ES(I)=* XSMOD
     1     E11.6* I=*I5* REACTION=*I5* BLOCK=*I5* NIN=*I5)              XSMOD
      GO TO 1000                                                        XSMOD
C     ASSIGN A SIGMA TO ENERGY PRECEDING STARTING ENERGY                XSMOD
  180 IS=ISX=I-1                                                        XSMOD
      SIG(1)=SIG(2)                                                     XSMOD
      IF(IS .NE. 1)SIG(1)=1.0E-20                                       XSMOD
C     CHECK ENERGIES READ AGAINST MASTER MESH                           XSMOD
  190 DO 210 I=1,3                                                      XSMOD
      IF(DUM(I)-ES(ISX+I))200,210,200                                   XSMOD
  200 IF(DUM(I))205,215,205                                             XSMOD
  205 PRINT 8, IRT,IBLK,ISID,IS,ISX,(ES(ISX+I),I=1,3),(DUM(I),I=1,3)    XSMOD
    8 FORMAT(//* READXS. ENERGIES DONT MATCH MESH. REACTION=*I5* BLOCK=* XSMOD
     1     I5* NIN=*I5* IS=*I5* ISX=*I5/* MESH ENERGIES=*3(E15.6)* ENERGI XSMOD
     2ES=*3(E15.6))                                                     XSMOD
```

```
      GO TO 1000                                                        XSMOD    1934
  210 CONTINUE                                                          XSMOD    1935
C     ADJUST POINTERS TO READ NEXT CARD                                 XSMOD    1936
  215 J=J+ 3                                                            XSMOD    1937
      ISX=ISX+ 3                                                        XSMOD    1938
  220 CONTINUE                                                          XSMOD    1939
C     HAVE READ ALL CARDS IN THIS ENERGY RANGE. ADJUST POINTERS.        XSMOD    1940
      L=MOD(NXS,3)                                                      XSMOD    1941
      JADJ=3- L+ 1                                                      XSMOD    1942
      IF(L . EQ. 0)JADJ=1                                               XSMOD    1943
      J=J- JADJ                                                         XSMOD    1944
      ISX=ISX- JADJ                                                     XSMOD    1945
      IF(NCD . NE. ICDS(IBLK))GO TO 110                                 XSMOD    1946
C     HAVE READ ALL CARDS IN BLOCK. ASSIGN SIGMA AT HIGH END.           XSMOD    1947
      J=J+ 1                                                            XSMOD    1948
      IF(J . LE. NES)GO TO 230                                          XSMOD    1949
      PRINT 10, IRT,IBLK,ISID,NES,J                                     XSMOD    1950
   10 FORMAT(//* READXS. HAVE READ MORE THAN NES XS. REACTION=*15* BLOCK XSMOD    1951
     1 =*15* NIN=*15* NES=*15* J=*15)                                   XSMOD    1952
      GO TO 1000                                                        XSMOD    1953
  230 SIG(J)=SIG(J- 1)                                                  XSMOD    1954
      IF((ISX+ 2) . LT. NES)SIG(J)=1. 0E- 20                            XSMOD    1955
C     PACK LOWER AND UPPER INDICES FOR THIS REACTION.                   XSMOD    1956
      ME(IR)=IS*124+ IS+ J- 1                                           XSMOD    1957
      GO TO 2000                                                        XSMOD    1958
 1000 IERR=NCD+1                                                        XSMOD    1959
      RETURN                                                            XSMOD    1960
C                                                                       XSMOD    1961
C     DEFINE DT ARRAY FOR SIGMAS                                        XSMOD    1962
 2000 J1=ME(IR)/124                                                     XSMOD    1963
      J2=ME(IR)- J1*124                                                 XSMOD    1964
      K=J2- J1+ 1                                                       XSMOD    1965
      N. XS(IR)=(F. SIG(J),J=1,K)/                                      XSMOD    1966
      RETURN                                                            XSMOD    1967
      END                                                               XSMOD    1968
      END MODULE                                                        XSMOD    1969
      MODULE LISTXS                                                     XSMOD    1970
C     RETRIEVES AND LISTS SID DATA FROM DTRAN LIBRARY. IF SID ADDED IN  XSMOD    1971
C     THIS RUN,LISTS DATA FROM DTRAN N LISTS IN ECS.                    XSMOD    1972
C     KID       SID NO. IN LIBRARY                                      XSMOD    1973
C     IB        INDEX TO KDIMAG. IF=0,ENTERED FROM ADDXS.               XSMOD    1974
C                                                                       XSMOD    1975
      COMMON  /X/ X(4850)                                               X           2
      COMMON /CNTRL/ ISID(1),KID,ND,ITIDS(25),SDES(8),IR,ITIDR(50),NES, CNTRL       2
     1     ES(4000),IDR(50),ME(50),IRS(50),QR(50),IZ,MIXT,K2            CNTRL       3
      DIMENSION  ITID(1),TDES(8),E(400),TK(4000),MD(400),PROB(10),      ANGD        2
     1     NMU(10)                                                      ANGD        3
      EQUIVALENCE  (ITID(1), X(1)), (TDES(1),X(2)), (NED,X(10)),        ANGD        4
     1     (E(1),X(11)), (ISYS,X(411)), (A,X(412)), (LEGN,X(413)),      ANGD        5
     2     (KT,X(414)), (TK(1),X(415)), (MD(1),X(4415)), (NINC,X(4815)), ANGD       6
     3     (PROB(1),X(4816)), (NMU(1),X(4826))                          ANGD        7
C                                                                       ANGD        8
      DIMENSION  SIG(4000)                                              XS          2
      EQUIVALENCE  (SIG(1),X(1))                                        XS          3
C                                                                       XS          4
      COMMON/DIR/  KGID(300),KSID(300),IDIR,IIG,IIX,IDIRSV              DIR         2
      DIMENSION  IDNOG(100),NWDSG(100),IDNOS(100),NWDSS(100)            DIR         3
      EQUIVALENCE  (KGID(101),IDNOG(1)),(KGID(201),NWDSG(1)),           DIR         4
     1                (KSID(101),IDNOS(1)),(KSID(201),NWDSS(1))         DIR         5
      COMMON  /OPTS/ KDIMAG(1100),N(2),IALL,IB                          OPTS        2
      DIMENSION  XKDIMAG(1100)                                          OPTS        3
      EQUIVALENCE  (KDIMAG,XKDIMAG)                                     OPTS        4
      COMMON  /MSR/ MSR(150)                                            MSR         2
      DATA MSR/                                                         MSR         3
     1 5HTOT) , 5HN)  ,6HNONEL), 5HN*)  , 5HN*1) , 5HN*2) , 5HN*3) ,    MSR         4
     2 5HN*4) , 5HN*5) , 5HN*6) , 5HN*7) , 5HN*8) , 5HN*9) , 5HN*10),   MSR         5
     3 5HN*C) , 5H2N)  , 5H3N)  ,2*4HF)  ,5HN*F) ,5H2NF) ,4HN*A),5HN*3A), MSR       6
     4 5H2NA) , 5H3NA) ,6H2NISO),6HGMA)+ F, 5HN*P) , 5HN*2A), 5H2N2A),  MSR         7
     A 70*3HX) ,                                                        MSR         8
     5 6HPARAB),5HGMA) , 5HP)  , 5HD)  , 5HT)  , 5HHE3) , 5HA)  ,       MSR         9
     6 5H2A)  , 5HPA)  ,6HDESTR),40*3HX) /                              MSR        10
```

```
      COMMON  SS(2000,7)
      DIMENSION  DUM(2000)
      EQUIVALENCE  (DUM(1),SS(1,7))
      DIMENSION  ESJ(3000)
      EQUIVALENCE  (ESJ(1),X(1))
      DATA  I24 /100000000B/
C
C     DEFINE FLAG FROM DT LIST.
 100  F. IB=N. LISTXS
      IF(IB . EQ. 0)GO TO 200
C     LISTING REQUESTED ON OPTION CARD. DEFINE FTRAN VARIABLES FROM DT.
      F. XKDIMAG=N. RDJOUT. 1/
      F. KSID=N. DIRSID. 1/
      F. IDNOS=N. DIRSID. 2/
      F. NWDSS=N. DIRSID. 3/
      F. IIX=WORDS OF N. DIRSID. 1/
C
      IC=IB
 150  IC=IC+1
C     CHECK FOR RETURN
      IF(KDIMAG(IC) . EQ. 0)GO TO 2000
      IF((KDIMAG(IC) . AND. 77000000000000000000B) . NE. 0)GO TO 2000
      KID=KDIMAG(IC)
C     RETRIEVE FROM DTRAN LIBRARY
      CALL RETREV(KID,KSID,IIX,1,IERR,AM)
      IF(IERR . EQ. 0) GO TO 170
      PRINT 2, KID
   2  FORMAT(* LISTXS. LIBRARY RETRIEVAL ERROR FOR SID *I5* NO LISTING*)
      GO TO 150
C     DEFINE DT LISTS FOR RETRIEVED SID
 170  N. CNTRL=D. ((AM)). N. CNTRL
      N. XS=D. ((AM)). N. XS
      N. ESJ=D. ((AM)). N. ESJ
      N. ANGD=D. ((AM)). N. ANGD
C     DEFINE FTRAN VARIABLES FROM DT LIST
 200  F. KID=N. CNTRL. 1/
      F. ND=N. CNTRL. 2/
      F. ITIDS=N. CNTRL. 3/
      F. SDES=N. CNTRL. 4/
      F. IR=N. CNTRL. 5/
      F. ITIDR=N. CNTRL. 6/
      F. NES=N. CNTRL. 7/
      F. ES=N. CNTRL. 8/
      F. IDR=N. CNTRL. 9/
      F. ME=N. CNTRL. 10/
      F. IRS=N. CNTRL. 11/
      F. QR=N. CNTRL. 12/
      F. IZ=N. CNTRL. 13/
      F. MIXT=N. CNTRL. 14/
      F. K2=N. CNTRL. 15/
      F. LLL=N. CNTRL. 16/
C
      IF(LLL . EQ. 0)SOURCE=3HUK
C     CHANGE REACTION TITLES FOR LLL. TYPE NOT USED.
      IF(LLL . NE. 1)GO TO 210
      MSR(31)=5HN*11)
      MSR(32)=5HN*12)
      MSR(33)=5HN*13)
      MSR(34)=5HN*14)
      MSR(35)=5HN*15)
      MSR(36)=5HN*16)
      MSR(37)=5HN*17)
      MSR(38)=5HN*18)
      MSR(39)=5HN*19)
      MSR(40)=5HN*20)
      MSR(81)=5HPN*)
      MSR(82)=5HN*D)
      MSR(83)=5HN*T)
      MSR(84)=6HN*HE3)
      MSR(85)=5H4N)
      MSR(101)=5HX)
      MSR(110)=5HX)
```

```
      SOURCE=3HLLL
C
C     CHANGE REACTION TITLES FOR RBL. TYPE NOT USED.
  210 IF(LLL .NE. 2)GO TO 215
      MSR(27)=5HN*P)
      MSR(28)=6HN*GMA)
      MSR(29)=5HN*D)
      MSR(30)=6HN*HE3)
      MSR(31)=5HN*T)
      MSR(109)=5HX)
      MSR(110)=5HX)
      SOURCE=3HRBL
C     PRINT HEADING
  215 PRINT 4, KID,SDES,ND,IR,NES,SOURCE
    4 FORMAT(22H1   * * * CROSS SECTION,I6,6X,8A10//25X,*NO. OF ANG.  DIST.
     1*I6,6X,*NO. OF REACT. *I6,6X,*NO. OF ENERG. *I6,6X,*SOURCE= *A3)
      PRINT 6, IZ,K2,(ITIDS(I),I=1,ND)
    6 FORMAT(32H0   * * ANG. IDS * * TIDS(ND) * * ,20X,*Z=*I6,4X,*K2=*I6//
     1(12I8))
      PRINT 8
    8 FORMAT(28H0 * * DATA FOR EACH REACTION//* REACT.     ID     ANG. DIST.
     1 INDICES(ES)       ENERGY REL(Q)          REACTION       LENGTH(ESJ)*/)
C     PRINT SUMMARY FOR EACH REACTION. UNPACK INDICES.
      DO 220 I=1,IR
      J1=ME(I)/I24
      J2=ME(I)-J1*I24
      ID=IDR(I)
      ID=MOD(ID,1000)
      PRINT 10, I,ID,ITIDR(I),J1,J2,QR(I),MSR(ID),IRS(I)
   10 FORMAT(2I6,I8,3X,2I7,1PE19. 8,5X,*(N,*A6,4X,I7)
  220 CONTINUE
  250 IR1=IR
C     CAN PRINT CROSS SECTIONS FOR ONLY 7 REACTIONS ACROSS PAGE
      II=MIN0(7,IR1)
      IF=0
      DO 300 K=1,IR
      IF(IF .NE. 0) GO TO 260
C     CLEAR SS BLOCK II*NES WORDS
      NWD=NES
      IF(NES .GT. 2000)NWD=2000
      DO 255 JJ=1,II
      DO 254 LL=1,NWD
      SS(LL,JJ)=0. 0
  254 CONTINUE
  255 CONTINUE
C
C     IF ESJ FOR THIS REACTION, DEFINE FROM DT LIST AND PRINT
  260 L=IRS(K)
      IF(L .EQ. 0)GO TO 270
      F. ESJ=N. ESJ(K)
      ID=IDR(K)
      ID=MOD(ID,1000)
      MSG=MSR(ID)
      CALL PRESJ(K,L,MSG)
C
C     DEFINE SIGMAS FROM DT LIST
  270 F. SIG=N. XS(K)
C     FILL PART OF SS BLOCK WITH SIGMAS FOR THIS REACTION
C     SS BLOCK IS 7 STRIPS OF 2000 WDS. IF IS COUNT OF STRIPS FILLED.
      IF=IF+1
C     GET NO. OF WORDS FROM INDICES
      J1=ME(K)/I24
      J2=ME(K)-J1*I24
      J=J2-J1+1
C     SET LOWER INDEX
      LO=J1
C     IF LOWER INDEX GT 2000,MUST SAVE ZEROES
      IF(LO .LE. 2000)GO TO 273
C     IF NEED TO USE PART OF SS THAT IS DUM,MUST CLEAR IT FIRST
      IF(IF .NE. 7)GO TO 272
      DO 271 M=1,2000
      SS(M,7)=0. 0
```

```
  271   CONTINUE                                                          XSMOD
  272   N. SS(K). 1/=(F. SS(M,IF),M=1,2000)/                              XSMOD
        LO=J1- 2000                                                       XSMOD
  273   DO 275 L1=1,J                                                     XSMOD
        SS(LO,IF)=SIG(L1)                                                 XSMOD
        LO=LO+ 1                                                          XSMOD
        IF(LO . LE. 2000)GO TO 275                                        XSMOD
        N. SS(K). 1/=(F. SS(M,IF),M=1,2000)/                              XSMOD
        LO=1                                                              XSMOD
  275   CONTINUE                                                          XSMOD
        IF(J2. LE. 2000)GO TO 280                                         XSMOD
        J3=J2- 2000                                                       XSMOD
        N. SS(K). 2/=(F. SS(M,IF),M=1,J3)/                                XSMOD
C       RESTORE FIRST 2000 QUANTITIES FOR PRINTING                        XSMOD
C       MUST RESTORE FROM DTRAN TO DUM,THEN TRANSFER TO CORRECT LOCATION  XSMOD
C       IN SS BLOCK.  DUM IS 7TH STRIP OF SS.                             XSMOD
        F. DUM=N. SS(K). 1/                                               XSMOD
        DO 277 M=1,2000                                                   XSMOD
        SS(M,IF)=DUM(M)                                                   XSMOD
  277   CONTINUE                                                          XSMOD
C                                                                         XSMOD
  280   IF(IF . NE. II)GO TO 300                                          XSMOD
C       READY TO PRINT SS BLOCK                                           XSMOD
        CALL PRSIG(K,IF)                                                  XSMOD
C       SET UP SS BLOCK FOR NEXT SET OF 7 XS TO PRINT                     XSMOD
        IF=0                                                              XSMOD
        IR1=IR1- 7                                                        XSMOD
        II=MIN0(7,IR1)                                                    XSMOD
  300   CONTINUE                                                          XSMOD
C                                                                         XSMOD
C       ANGULAR DISTRIBUTIONS                                             XSMOD
        DO 450 K=1,ND                                                     XSMOD
C       DEFINE FTRAN VARIABLES FROM DT LIST                               XSMOD
        F. ITID=N. ANGD(K). 1/                                            XSMOD
        F. TDES=N. ANGD(K). 2/                                            XSMOD
        F. NED=N. ANGD(K). 3/                                             XSMOD
        F. E=N. ANGD(K). 4/                                               XSMOD
        F. ISYS=N. ANGD(K). 5/                                            XSMOD
        F. A=N. ANGD(K). 6/                                               XSMOD
        F. LEGN=N. ANGD(K). 7/                                            XSMOD
        F. KT=N. ANGD(K). 8/                                              XSMOD
        F. TK=N. ANGD(K). 9/                                              XSMOD
        F. MD=N. ANGD(K). 10/                                             XSMOD
        F. NINC=N. ANGD(K). 11/                                           XSMOD
C       IFK IS LINE COUNT                                                 XSMOD
        IFK=0                                                             XSMOD
        DO 400 M=1,NED                                                    XSMOD
C       GET INDICES, NO. OF WORDS FOR THIS ENERGY                         XSMOD
        JJ1=MD(M)/I24                                                     XSMOD
        JJ2=MD(M)- JJ1*I24                                                XSMOD
        KK=JJ2- JJ1+ 1                                                    XSMOD
        IF(IFK . NE. 0)GO TO 350                                          XSMOD
C       PRINT HEADING                                                     XSMOD
        PRINT 12, ITID,TDES,NED,ISYS,LEGN,KT,NINC,A                       XSMOD
   12   FORMAT(1H1,4X*TID=*I6,6X,8A10/15X*NED=*I6,6X*SYS=*I6,6X*LEGN=*    XSMOD
    1   I6,6X,*KT=*I6,6X,*NINC=*I6,6X,*A=*1PE20. 10/)                     XSMOD
        PRINT 14                                                          XSMOD
   14   FORMAT(//5X*I*6X*E(I)*9X*INDICES(TK)*11X*TK BLOCK*)               XSMOD
        IFK=6                                                             XSMOD
C                                                                         XSMOD
C       GET NO. OF LINES FOR THIS ENERGY                                  XSMOD
  350   K1=KK/6 + 2                                                       XSMOD
        IFK=IFK+ K1                                                       XSMOD
        IF(IFK . LE. 115)GO TO 370                                        XSMOD
C       PRINT HEADING ON EVERY OTHER PAGE                                 XSMOD
        PRINT 12, ITID,TDES,NED,ISYS,LEGN,KT,NINC,A                       XSMOD
        PRINT 14                                                          XSMOD
        IFK=K1+ 6                                                         XSMOD
C       PRINT TK BLOCK FOR THIS ENERGY                                    XSMOD
  370   PRINT 16, M,E(M),JJ1,JJ2,(TK(I),I=JJ1,JJ2)                        XSMOD
   16   FORMAT(/I6,1PE15. 6,2I6,4X,1P6E15. 6/(37X,1P6E15. 6))             XSMOD
```

```
  400     CONTINUE                                                          XSMOD   2198
  450     CONTINUE                                                          XSMOD   2199
C         LISTING FINISHED FOR ONE ELEMENT                                  XSMOD   2200
          IF(IB .NE. 0)GO TO 150                                            XSMOD   2201
 2000     RETURN                                                            XSMOD   2202
          END                                                               XSMOD   2203
          SUBROUTINE PRESJ(K,L,RLAB)                                        XSMOD   2204
C         PRINTS ESJ BLOCK                                                  XSMOD   2205
C         K          REACTION INDEX                                         XSMOD   2206
C         L          NO. OF WORDS IN ESJ BLOCK                              XSMOD   2207
C         RLAB       TITLE FOR THIS REACTION                                XSMOD   2208
C                                                                           XSMOD   2209
          COMMON /X/ X(4850)                                                X          2
          DIMENSION ESJ(3000)                                               XSMOD   2211
          EQUIVALENCE (ESJ(1),X(1))                                         XSMOD   2212
C                                                                           XSMOD   2213
          ID=ESJ(1)                                                         XSMOD   2214
          J=2                                                               XSMOD   2215
C         PRINT HEADING                                                     XSMOD   2216
          PRINT 2, RLAB,K,ID,L                                              XSMOD   2217
    2     FORMAT(///                                                        XSMOD   2218
     1     25X,* - - - LAWS FOR REACTION   (N,*A6,2X*(ESJ BLOCK) - - - */    XSMOD   2219
     2     25X,* - - - R =*I4,6X,*ID =*I6,6X*LENGTH =*I6* - - - *//)        XSMOD   2220
  100     PRINT 4, ESJ(J)                                                   XSMOD   2221
    4     FORMAT(*0 E = *1PE15. 5)                                          XSMOD   2222
          J=J+1                                                             XSMOD   2223
  110     LAW=ESJ(J+1)                                                      XSMOD   2224
          PRINT 6, LAW,ESJ(J)                                               XSMOD   2225
    6     FORMAT(15X*L =*I3,4X*W =*1PE12. 5)                                XSMOD   2226
          J=J+2                                                             XSMOD   2227
C         J NOW POINTS TO FIRST DATA ITEM                                   XSMOD   2228
  115     JJ=J                                                              XSMOD   2229
  120     IF(ESJ(J) .LT. 0)GO TO 130                                        XSMOD   2230
          J=J+1                                                             XSMOD   2231
          GO TO 120                                                         XSMOD   2232
C         PRINT DATA                                                        XSMOD   2233
  130     LL=J-1                                                            XSMOD   2234
          PRINT 8, (ESJ(IJ),IJ=JJ,LL)                                       XSMOD   2235
    8     FORMAT(15X,1P6E18. 5)                                             XSMOD   2236
          J=LL+1                                                            XSMOD   2237
          JJ=J                                                              XSMOD   2238
          DO 150 M=1,4                                                      XSMOD   2239
          IF(ESJ(J) .GE. 0)GO TO 160                                        XSMOD   2240
          J=J+1                                                             XSMOD   2241
          IF(J .GT. L)GO TO 160                                             XSMOD   2242
  150     CONTINUE                                                          XSMOD   2243
  160     LL=J-1                                                            XSMOD   2244
          JC=LL-JJ+1                                                        XSMOD   2245
C         PRINT MARKERS                                                     XSMOD   2246
          PRINT 10, (ESJ(IJ),IJ=JJ,LL)                                      XSMOD   2247
   10     FORMAT(101X,4F6. 0)                                               XSMOD   2248
          IF(LL .EQ. L)GO TO 2000                                           XSMOD   2249
C         END OF LAW                                                        XSMOD   2250
          IF(JC .EQ. 1)GO TO 110                                            XSMOD   2251
C         END OF ENERGY                                                     XSMOD   2252
          IF(JC .EQ. 3)GO TO 100                                            XSMOD   2253
C         END OF REACTION                                                   XSMOD   2254
          IF(JC .NE. 4)GO TO 170                                            XSMOD   2255
C         PRINT BLOCK 4018 OR 4019                                          XSMOD   2256
          ID=ESJ(J)                                                         XSMOD   2257
          PRINT 12,ID                                                       XSMOD   2258
   12     FORMAT(12X,*ID = *I10)                                            XSMOD   2259
          J=J+1                                                             XSMOD   2260
          GO TO 115                                                         XSMOD   2261
C         ERROR                                                             XSMOD   2262
  170     PRINT 14,JC                                                       XSMOD   2263
   14     FORMAT(* PRESJ. ERROR. MARKER COUNT=*I3,*ESJ DUMP FOLLOWS*///)    XSMOD   2264
          CALL PDUMP(ESJ(1),ESJ(L),1)                                       XSMOD   2265
 2000     RETURN                                                            XSMOD   2266
          END                                                               XSMOD   2267
          SUBROUTINE PRSIG(I,IF)                                            XSMOD   2268
```

```
C           PRINTS SIGMAS                                                XSMOD  2269
C           I          REACTION INDEX                                    XSMOD  2270
C           IF         NO. OF REACTIONS FOR WHICH SIGMAS WILL BE PRINTED XSMOD  2271
C           JK         LINE COUNT                                        XSMOD  2272
            COMMON /CNTRL/ ISID(1),KID,ND,ITIDS(25),SDES(8),IR,ITIDR(50),NES, CNTRL  2
     1         ES(4000),IDR(50),ME(50),IRS(50),QR(50),IZ,MIXT,K2         CNTRL     3
            COMMON  /TYPE/  ITYPER(150)                                  XSMOD  2274
            COMMON  /MSR/  MSR(150)                                      XSMOD  2275
            COMMON  SS(2000,7)                                           XSMOD  2276
            DIMENSION  DUM(2000)                                         XSMOD  2277
            EQUIVALENCE  (DUM(1),SS(1,7))                                XSMOD  2278
            DIMENSION  KHEAD(10)                                         XSMOD  2279
            DATA  HOL /4H (N,/                                           XSMOD  2280
C                                                                        XSMOD  2281
C           GET INFORMATION FOR HEADING AND PRINT IT                     XSMOD  2282
            DO 100 IK=1,IF                                               XSMOD  2283
            IJ=I- IF+ IK                                                 XSMOD  2284
            ID=IDR(IJ)                                                   XSMOD  2285
            ID=MOD(ID,1000)                                              XSMOD  2286
            KHEAD(IK)=MSR(ID)                                            XSMOD  2287
  100       CONTINUE                                                     XSMOD  2288
            I1=I- IF+ 1                                                  XSMOD  2289
            I2=I                                                         XSMOD  2290
            JK=0                                                         XSMOD  2291
C                                                                        XSMOD  2292
            NWD=NES                                                      XSMOD  2293
            IF(NES . GT.  2000)NWD=2000                                  XSMOD  2294
            DO 110 M=1,NWD                                               XSMOD  2295
            IF(JK . NE.  0)GO TO 70                                      XSMOD  2296
            PRINT 2, I1,I2,(HOL,KHEAD(IK),IK=1,IF)                       XSMOD  2297
    2       FORMAT(1H1,25X,* - - - SIG FOR ALL REACTIONS - - - R = *I2*,*I2 XSMOD  2298
     1      * - - - *//4X*I*6X*ES(I)*6X,7(A4,A6,4X))                     XSMOD  2299
            JK=1                                                         XSMOD  2300
C           IF ALL SIGMAS FOR AN ENERGY ARE ZERO,DO NOT PRINT            XSMOD  2301
   70       DO 80 JO=1,IF                                                XSMOD  2302
            S=SS(M,JO)                                                   XSMOD  2303
            IF(S . GT.  0)GO TO 85                                       XSMOD  2304
   80       CONTINUE                                                     XSMOD  2305
            GO TO 110                                                    XSMOD  2306
   85       JK=JK+ 1                                                     XSMOD  2307
            PRINT 4,  M,ES(M),(SS(M,J),J=1,IF)                           XSMOD  2308
    4       FORMAT(I5,1P8E14. 5)                                         XSMOD  2309
            IF(JK . EQ.  118)JK=0                                        XSMOD  2310
  110       CONTINUE                                                     XSMOD  2311
C                                                                        XSMOD  2312
C           IF MORE THAN 2000 ENERGIES MUST GET REMAINING DATA FROM ECS  XSMOD  2313
            IF(NES . LE.  2000)GO TO 2000                                XSMOD  2314
C           CLEAR SS BLOCK IF*(NES- 2000) WORDS                          XSMOD  2315
            JL=NES- 2000                                                 XSMOD  2316
            DO 130 J=1,IF                                                XSMOD  2317
            DO 120 L=1,JL                                                XSMOD  2318
            SS(L,J)=0. 0                                                 XSMOD  2319
  120       CONTINUE                                                     XSMOD  2320
  130       CONTINUE                                                     XSMOD  2321
C           FILL SS BLOCK                                                XSMOD  2322
            DO 150 IK=1,IF                                               XSMOD  2323
            IJ=I- IF+ IK                                                 XSMOD  2324
C           FIND OUT IF THIS REACTION HAS MORE THAN 2000 SIGMAS          XSMOD  2325
            JSIG=ME(IJ) . AND.  77777777B                                XSMOD  2326
            IF(JSIG . LE.  2000)GO TO 150                                XSMOD  2327
C           FOR THIS REACTION GET SIGMAS FROM ECS                        XSMOD  2328
C           MUST RESTORE FROM DTRAN TO DUM,THEN TRANSFER TO CORRECT LOCATION XSMOD  2329
C           IN SS BLOCK.  DUM IS 7TH STRIP OF SS.  CLEAR IT WHEN IF=7    XSMOD  2330
            J3=JSIG- 2000                                                XSMOD  2331
            IF(IF . NE.  7)GO TO 140                                     XSMOD  2332
            DO 135 M=1,2000                                              XSMOD  2333
            DUM(M)=0. 0                                                  XSMOD  2334
  135       CONTINUE                                                     XSMOD  2335
  140       F. DUM=N. SS(IJ). 2/                                         XSMOD  2336
            DO 145 M=1,J3                                                XSMOD  2337
            SS(M,IK)=DUM(M)                                              XSMOD  2338
```

```
  145      CONTINUE                                                        XSMOD   2339
  150      CONTINUE                                                        XSMOD   2340
C                                                                          XSMOD   2341
C          SS BLOCK FILLED. READY TO PRINT                                 XSMOD   2342
           DO 200 M=2001,NES                                               XSMOD   2343
           M2=M-2000                                                       XSMOD   2344
           IF(JK .NE. 0)GO TO 160                                          XSMOD   2345
           PRINT 2, I1,I2,(HOL,KHEAD(IK),IK=1,IF)                          XSMOD   2346
           JK=1                                                            XSMOD   2347
  160      DO 170 JO=1,IF                                                  XSMOD   2348
           S=SS(M2,JO)                                                     XSMOD   2349
           IF(S .GT. 0)GO TO 180                                           XSMOD   2350
  170      CONTINUE                                                        XSMOD   2351
           GO TO 200                                                       XSMOD   2352
  180      JK=JK+1                                                         XSMOD   2353
           PRINT 4, M,ES(M),(SS(M2,J),J=1,IF)                              XSMOD   2354
           IF(JK .EQ. 118)JK=0                                             XSMOD   2355
  200      CONTINUE                                                        XSMOD   2356
 2000      RETURN                                                          XSMOD   2357
           END                                                             XSMOD   2358
           SUBROUTINE RETREV(KID,KSID,IIK,DUM,IERR,AM)                     XSMOD   2359
C          RETRIEVES SELECTED SID FROM DTRAN TAPE                          XSMOD   2360
C          KID     SID NO. IN LIBRARY                                      XSMOD   2361
C          KSID    DTRAN DIRECTORY TO SID                                  XSMOD   2362
C          IIK     NO. OF SID IN DIRECTORY                                 XSMOD   2363
C          DUM      DUMMY                                                  XSMOD   2364
C          IERR    ERROR FLAG. 0 IF OK.                                    XSMOD   2365
C          AM      DTRAN NAME FOR THIS SID                                 XSMOD   2366
C                                                                          XSMOD   2367
           DIMENSION  KSID(300),M(3)                                       XSMOD   2368
           DATA  (M(I),I=1,3) /10H(*SID*I4) ,10H(*SID*I3) ,10H(*SID*I2) /  XSMOD   2369
C                                                                          XSMOD   2370
           IERR=0                                                          XSMOD   2371
C          SEARCH DIRECTORY FOR THIS SID                                   XSMOD   2372
C          SEARCH FOR SID IN DIRECTORY                                     XSMOD   2373
           DO 110 I=1,IIK                                                  XSMOD   2374
           IF(KID .EQ. KSID(I))GO TO 120                                   XSMOD   2375
  110      CONTINUE                                                        XSMOD   2376
           PRINT 2, KID                                                    XSMOD   2377
    2      FORMAT(* LISTXS,RETREV.  CANNOT FIND SID *I4* ON DTRAN TAPE. *) XSMOD   2378
           IERR=1                                                          XSMOD   2379
           RETURN                                                          XSMOD   2380
C                                                                          XSMOD   2381
C          GET RECALL NO. ASSOCIATED WITH THIS SID                         XSMOD   2382
  120      K=KSID(I+100)                                                   XSMOD   2383
C          SELECT FORMAT FOR ENCODING DTRAN NAME                           XSMOD   2384
           IF(KID .LT. 10000)I=1                                           XSMOD   2385
           IF(KID .LT. 1000)I=2                                            XSMOD   2386
           IF(KID .LT. 100)I=3                                             XSMOD   2387
           ENCODE(7,M(I),AM)KID                                            XSMOD   2388
C          DO RETRIEVE                                                     XSMOD   2389
           D. ((AM))=D. DT00010 *MHX1* F. K                                XSMOD   2390
           RETURN                                                          XSMOD   2391
           END                                                             XSMOD   2392
           END MODULE                                                      XSMOD   2393
           MODULE RDJOBS                                                   XSMOD   2394
           PROGRAM RDJOBS(INPUT,OUTPUT)                                    XSMOD   2395
C          READS OPTION CARDS. MAKES LISTS OF ALL GID,SID TO BE RECALLED FROM  XSMOD   2396
C          LIBRARY AND ALL GID,SID TO BE WRITTEN TO TAPE. SETS UP DT LISTS XSMOD   2397
C          AND CALLS MODULE RECLIB.                                        XSMOD   2398
           COMMON/DIR/  KGID(300),KSID(300),IDIR,IIG,IIX,IDIRSV            DIR        2
           DIMENSION  IDNOG(100),NWDSG(100),IDNOS(100),NWDSS(100)          DIR        3
           EQUIVALENCE  (KGID(101),IDNOG(1)),(KGID(201),NWDSG(1)),         DIR        4
    1                   (KSID(101),IDNOS(1)),(KSID(201),NWDSS(1))          DIR        5
           COMMON  /OPTS/ KDIMAG(1100),N(2),IALL,IB                        OPTS       2
           DIMENSION  XKDIMAG(1100)                                        OPTS       3
           EQUIVALENCE  (KDIMAG,XKDIMAG)                                   OPTS       4
           COMMON /TEMP/ ITEMPX(200),ITEMPG(200),IADDX(100),IADDG(100),    TEMP       2
    1         ITAPX(100),ITAPG(100),M(15)                                  TEMP       3
           DIMENSION  XM(15)                                               TEMP       4
           EQUIVALENCE  (M,XM)                                             TEMP       5
```

```
      DIMENSION   INSID(150),INGID(150)
C
C     DEFINE FORTRAN VARIABLES TRANSMITTED BY DT LIST.
C
      F. KGID=N. DIRGID. 1/
      F. IDNOG=N. DIRGID. 2/
      F. NWDSG=N. DIRGID. 3/
      F. KSID=N. DIRSID. 1/
      F. IDNOS=N. DIRSID. 2/
      F. NWDSS=N. DIRSID. 3/
      F. IIG=WORDS OF N. DIRGID. 1/
      F. IIX=WORDS OF N. DIRSID. 1/
      F. XM=N. RDJOBS
C
C     CLEAR STORAGES
      IC=LOCF(ITAPG(100))- LOCF(ITEMPX(1))+ 1
      DO 100 I=1,IC
  100 ITEMPX(I)=0
      IC=LOCF(N(2))- LOCF(KDIMAG(1))+ 1
      DO 101 I=1,IC
  101 KDIMAG(I)=0
      DO 105 I=1,150
      INGID(I)=0
  105 INSID(I)=0
C     READ IN OPTIONS AND THEIR IDS.
C     RETURNS IX=NO. SID,IG=NO. GID TO WRITE TO TAPE1
      CALL READOPT(IX,IG)
      IF(IALL . EQ. 1)GO TO 107
C     CHECK FOR NO SIDS TO BE RECALLED.
      IF(N(1) . EQ. 0)GO TO 106
C     MAKE LIST OF SIDS TO BE RECALLED.
      CALL USELST(ITEMPX,INSID,1,KSID,IIX)
C     CHECK FOR NO GIDS TO BE RECALLED.
  106 IF(N(2) . EQ. 0)GO TO 108
C     MAKE LIST OF GIDS TO BE RECALLED.
      CALL USELST(ITEMPG,INGID,2,KGID,IIG)
      GO TO 108
C     ENTIRE LIBRARY TO BE RECALLED.
  107 CALL USEALL(ITEMPX,INSID,1,KSID,IIX)
      CALL USEALL(ITEMPG,INGID,2,KGID,IIG)
C
C     PREPARE LISTS FOR MODULE RECALL
  108 N. REEL=*MHX1*
      IF(KGID(1) . EQ. 0 . AND. KSID(1) . EQ. 0)GO TO 125
      IF(KGID(1) . EQ. 0)GO TO 1109
      IF(KSID(1) . EQ. 0)GO TO 1108
      N. RECLID=(F. IDNOG(I),F. NWDSG(I),I=1,IIG),(F. IDNOS(I),F. NWDSS(I),I=
     1 1,IIX)
      GO TO 109
 1108 N. RECLID=(F. IDNOG(I),F. NWDSG(I),I=1,IIG)
      GO TO 109
 1109 N. RECLID=(F. IDNOS(I),F. NWDSS(I),I=1,IIX)
  109 IXE=N(1)
      IGE=N(2)
      IF(IXE . EQ. 0 . AND. IGE . EQ. 0)GO TO 125
      IF(IXE . EQ. 0)GO TO 110
      IF(IGE . EQ. 0)GO TO 115
      N. USE=(F. ITEMPG(I),I=1,IGE),(F. ITEMPX(I),I=1,IXE)
      GO TO 120
  110 N. USE=(F. ITEMPG(I),I=1,IGE)
      GO TO 120
  115 N. USE=(F. ITEMPX(I),I=1,IXE)
  120 CONTINUE
      KALL RECLIB
C     PREPARE DT LIST FOR RETURN TO MAIN CODE
  125 IF(IX . EQ. 0)IX=1
      IF(IG . EQ. 0)IG=1
  130 N. RDJOUT=F. XKDIMAG/F. N,F. IALL/(F. ITAPG(I),I=1,IG)/
     1     (F. ITAPX(I),I=1,IX)/
      RETURN
      END
```

```
      SUBROUTINE READOPT(IX,IG)                                          XSMOD     2473
C     READS OPTIONS AND THEIR IDS,SETS UP COMPLETE LISTS OF SID,GID.     XSMOD     2474
C     MAKES ADD LISTS AND TAPE LISTS.  FLAGS ADDS IN TAPE LIST           XSMOD     2475
C     RETURNS IX,IG=NO. OF SID,NO. OF GID TO WRITE TO TAPE1              XSMOD     2476
C                                                                        XSMOD     2477
      COMMON/DIR/  KGID(300),KSID(300),IDIR,IIG,IIX,IDIRSV               DIR          2
      DIMENSION  IDNOG(100),NWDSG(100),IDNOS(100),NWDSS(100)             DIR          3
      EQUIVALENCE  (KGID(101),IDNOG(1)),(KGID(201),NWDSG(1)),            DIR          4
     1             (KSID(101),IDNOS(1)),(KSID(201),NWDSS(1))             DIR          5
      COMMON  /OPTS/ KDIMAG(1100),N(2),IALL,IB                           OPTS         2
      DIMENSION  XKDIMAG(1100)                                           OPTS         3
      EQUIVALENCE  (KDIMAG,XKDIMAG)                                      OPTS         4
      COMMON /TEMP/ ITEMPX(200),ITEMPG(200),IADDX(100),IADDG(100),       TEMP         2
     1    ITAPX(100),ITAPG(100),M(15)                                    TEMP         3
      DIMENSION  XM(15)                                                  TEMP         4
      EQUIVALENCE  (M,XM)                                                TEMP         5
C                                                                        XSMOD     2481
C     INITIALIZE INDICES,FLAGS,COUNTERS. READ OPTION CARDS INTO KDIMAG.  XSMOD     2482
      IALL=0                                                             XSMOD     2483
      IB=1                                                               XSMOD     2484
      IE=11                                                              XSMOD     2485
      JG=1                                                               XSMOD     2486
      JX=1                                                               XSMOD     2487
      KG=1                                                               XSMOD     2488
      KX=1                                                               XSMOD     2489
  105 READ 1,(KDIMAG(I),I=IB,IE)                                         XSMOD     2490
    1 FORMAT(A10,2X,10I6)                                                XSMOD     2491
      PRINT 2,(KDIMAG(I),I=IB,IE)                                        XSMOD     2492
    2 FORMAT(1X,A10,2X,10I6)                                             XSMOD     2493
C     DETERMINE WHICH OPTION REQUESTED AND GO TO APPROPRIATE SETUP.      XSMOD     2494
      DO 110 I=1,15                                                      XSMOD     2495
      IF(KDIMAG(IB) . EQ.  M(I))GO TO 120                                XSMOD     2496
  110 CONTINUE                                                           XSMOD     2497
      PRINT 4, (KDIMAG(I),I=IB,IE)                                       XSMOD     2498
    4 FORMAT(//* INPUT ERROR.  *A10,2X,10I6,//)                          XSMOD     2499
      GO TO 150                                                          XSMOD     2500
  120 GO TO(130,140,150,150,125,122,150,150,130,140,150,145,160,155,     XSMOD     2501
     1   170)I                                                           XSMOD     2502
C     MAKE ADD LISTS                                                     XSMOD     2503
  122 JB=IB+1                                                            XSMOD     2504
      DO 123 J=JB,IE                                                     XSMOD     2505
      IF(KDIMAG(J) . EQ.  0)GO TO 150                                    XSMOD     2506
      IADDG(JG)=KDIMAG(J)                                                XSMOD     2507
  123 JG=JG+1                                                            XSMOD     2508
      GO TO 150                                                          XSMOD     2509
  125 JB=IB+1                                                            XSMOD     2510
      DO 127 J=JB,IE                                                     XSMOD     2511
      IF(KDIMAG(J) . EQ.  0)GO TO 150                                    XSMOD     2512
      IADDX(JX)=KDIMAG(J)                                                XSMOD     2513
  127 JX=JX+1                                                            XSMOD     2514
      GO TO 150                                                          XSMOD     2515
C     MAKE RECALL LISTS                                                  XSMOD     2516
  130 CALL TEMPLST(ITEMPX,1,IB,IE)                                       XSMOD     2517
      GO TO 150                                                          XSMOD     2518
  140 CALL TEMPLST(ITEMPG,2,IB,IE)                                       XSMOD     2519
      GO TO 150                                                          XSMOD     2520
  145 IALL=1                                                             XSMOD     2521
C     GET NEXT OPTION CARD                                               XSMOD     2522
  150 IB=IE+1                                                            XSMOD     2523
      IE=IB+10                                                           XSMOD     2524
      GO TO 105                                                          XSMOD     2525
C     MAKE TAPE LISTS                                                    XSMOD     2526
  155 JB=IB+1                                                            XSMOD     2527
      DO 157 J=JB,IE                                                     XSMOD     2528
      IF (KDIMAG(J) . EQ.  0)GO TO 140                                   XSMOD     2529
      ITAPG(KG)=KDIMAG(J)                                                XSMOD     2530
  157 KG=KG+1                                                            XSMOD     2531
      GO TO 140                                                          XSMOD     2532
  160 JB=IB+1                                                            XSMOD     2533
      DO 162 J=JB,IE                                                     XSMOD     2534
      IF (KDIMAG(J) . EQ.  0)GO TO 130                                   XSMOD     2535
```

```
          ITAPX(KX)=KDIMAG(J)                                          XSMOD   2536
  162     KX=KX+1                                                      XSMOD   2537
          GO TO 130                                                    XSMOD   2538
  170     PRINT 6                                                      XSMOD   2539
    6     FORMAT(//* LAST CARD ENCOUNTERED. READY TO MAKE RECALL LIST. *)  XSMOD   2540
C                                                                      XSMOD   2541
          IF(IALL . EQ. 1)GO TO 250                                    XSMOD   2542
C         PUT TAPE LISTS IN ASCENDING NUMERICAL ORDER.                 XSMOD   2543
          IG=KG-1                                                      XSMOD   2544
          IX=KX-1                                                      XSMOD   2545
          CALL ORDER(ITAPG,IG)                                         XSMOD   2546
          CALL ORDER(ITAPX,IX)                                         XSMOD   2547
C         FLAG ADDS IN TAPE REQUEST LIST                               XSMOD   2548
          DO 190 I=1,100                                               XSMOD   2549
          IF(ITAPG(I) . EQ. 0)GO TO 200                                XSMOD   2550
          DO 180 J=1,100                                               XSMOD   2551
          IF(IADDG(J) . EQ. 0)GO TO 190                                XSMOD   2552
          IF(ITAPG(I) . NE. IADDG(J))GO TO 180                         XSMOD   2553
          ITAPG(I)=-ITAPG(I)                                           XSMOD   2554
          GO TO 190                                                    XSMOD   2555
  180     CONTINUE                                                     XSMOD   2556
  190     CONTINUE                                                     XSMOD   2557
  200     DO 220 I=1,100                                               XSMOD   2558
          IF(ITAPX(I) . EQ. 0)GO TO 230                                XSMOD   2559
          DO 210 J=1,100                                               XSMOD   2560
          IF(IADDX(J) . EQ. 0)GO TO 220                                XSMOD   2561
          IF(ITAPX(I) . NE. IADDX(J))GO TO 210                         XSMOD   2562
          ITAPX(I)=-ITAPX(I)                                           XSMOD   2563
          GO TO 220                                                    XSMOD   2564
  210     CONTINUE                                                     XSMOD   2565
  220     CONTINUE                                                     XSMOD   2566
  230     RETURN                                                       XSMOD   2567
C         FOR TAPEALL MERGE ADD AND TAPE LISTS,FLAG ADDS               XSMOD   2568
C         PUT ADD AND TAPE LISTS IN ASCENDING NUMERICAL ORDER.         XSMOD   2569
  250     JG=JG-1                                                      XSMOD   2570
          JX=JX-1                                                      XSMOD   2571
          CALL ORDER(IADDG,JG)                                         XSMOD   2572
          CALL ORDER(IADDX,JX)                                         XSMOD   2573
          DO 255 I=1,IIG                                               XSMOD   2574
  255     ITAPG(I)=KGID(I)                                             XSMOD   2575
          CALL ORDER(ITAPG,IIG)                                        XSMOD   2576
          DO 260 I=1,IIX                                               XSMOD   2577
  260     ITAPX(I)=KSID(I)                                             XSMOD   2578
          CALL ORDER(ITAPX,IIX)                                        XSMOD   2579
          CALL MERGE(IADDG,JG,ITAPG,IIG,IG)                            XSMOD   2580
          CALL MERGE(IADDX,JX,ITAPX,IIX,IX)                            XSMOD   2581
  270     RETURN                                                       XSMOD   2582
          END                                                          XSMOD   2583
          SUBROUTINE ORDER(ILIST,L)                                    XSMOD   2584
C         PUTS LIST OF INTEGERS IN ASCENDING ORDER                     XSMOD   2585
C         ILIST    LIST TO BE ORDERED                                  XSMOD   2586
C         L        NO OF ITEMS IN LIST                                 XSMOD   2587
C                                                                      XSMOD   2588
          DIMENSION  ILIST(100)                                        XSMOD   2589
          IL=L-1                                                       XSMOD   2590
          DO 200 I=1,IL                                                XSMOD   2591
          IF(ILIST(I) . EQ. 0)GO TO 300                                XSMOD   2592
          JB=I+1                                                       XSMOD   2593
          DO 100 J=JB,L                                                XSMOD   2594
          IF(ILIST(J) . EQ. 0)GO TO 200                                XSMOD   2595
          IF(ILIST(I) . LT. ILIST(J))GO TO 100                         XSMOD   2596
          IDUM=ILIST(I)                                                XSMOD   2597
          ILIST(I)=ILIST(J)                                            XSMOD   2598
          ILIST(J)=IDUM                                                XSMOD   2599
          IF(J . EQ. L)GO TO 200                                       XSMOD   2600
          I=I-1                                                        XSMOD   2601
          GO TO 200                                                    XSMOD   2602
  100     CONTINUE                                                     XSMOD   2603
  200     CONTINUE                                                     XSMOD   2604
  300     RETURN                                                       XSMOD   2605
          END                                                          XSMOD   2606
```

```
      SUBROUTINE MERGE(IADD,IA,ITAP,IT,IS)                                XSMOD  2607
C     MERGES ADD AND TAPE LISTS WITH IDS IN ASCENDING ORDER. ADDS ARE     XSMOD  2608
C     FLAGGED WITH MINUS                                                  XSMOD  2609
C                                                                         XSMOD  2610
C     IA        NO OF ID IN ADD LIST                                      XSMOD  2611
C     IT        NO OF ID IN DIRECTORY AND IN TAPE LIST                    XSMOD  2612
C     IS        TOTAL NO OF ID TO BE WRITTEN TO TAPE AFTER MERGE          XSMOD  2613
C                                                                         XSMOD  2614
      DIMENSION  IADD(100),ITAP(100)                                      XSMOD  2615
C                                                                         XSMOD  2616
C     CHECK FOR NO ADDITIONS                                              XSMOD  2617
      IF(IADD(1) . EQ.  0)GO TO 350                                       XSMOD  2618
      J=1                                                                 XSMOD  2619
      IS=IT+ 1                                                            XSMOD  2620
C     CLEAR UNUSED PART OF ITAP                                           XSMOD  2621
      DO 100 I=IS,100                                                     XSMOD  2622
  100 ITAP(I)=0                                                           XSMOD  2623
      IS=IT                                                               XSMOD  2624
      DO 200 I=1,IA                                                       XSMOD  2625
C     CHECK FOR ADD LIST FINISHED                                         XSMOD  2626
      IF(IADD(I) . EQ.  0)RETURN                                          XSMOD  2627
C     CHECK TO INSERT IN TAPE LIST OR PUT AT END                          XSMOD  2628
  110 IF(ITAP(J) . NE.  0)GO TO 120                                       XSMOD  2629
      ITAP(J)=- IADD(I)                                                   XSMOD  2630
      GO TO 190                                                           XSMOD  2631
  120 IF (IADD(I) .LT. ITAP(J))GO TO 130                                  XSMOD  2632
      J=J+1                                                               XSMOD  2633
      GO TO 110                                                           XSMOD  2634
C     WILL INSERT.MOVE REMAINING ITEMS                                    XSMOD  2635
  130 KK=IS+1                                                             XSMOD  2636
      KL=KK-J                                                             XSMOD  2637
      DO 150 K=1,KL                                                       XSMOD  2638
      ITAP(KK)=ITAP(KK- 1)                                                XSMOD  2639
      KK=KK- 1                                                            XSMOD  2640
  150 CONTINUE                                                            XSMOD  2641
      ITAP(J)=- IADD(I)                                                   XSMOD  2642
      IS=IS+ 1                                                            XSMOD  2643
  190 J=J+ 1                                                              XSMOD  2644
  200 CONTINUE                                                            XSMOD  2645
      RETURN                                                              XSMOD  2646
  350 IS=IT                                                               XSMOD  2647
      RETURN                                                              XSMOD  2648
      END                                                                 XSMOD  2649
      SUBROUTINE TEMPLST(IX,JJ,IB,IE)                                     XSMOD  2650
C     MAKES SID/GID LISTS FROM INPUT CARDS. INCLUDES DUPLICATIONS.        XSMOD  2651
      COMMON  /OPTS/ KDIMAG(1100),N(2),IALL,IB                            OPTS      2
      DIMENSION  XKDIMAG(1100)                                            OPTS      3
      EQUIVALENCE  (KDIMAG,XKDIMAG)                                       OPTS      4
      DIMENSION  IX(200)                                                  XSMOD  2653
C                                                                         XSMOD  2654
      II=N(JJ)                                                            XSMOD  2655
      ISD=IB+ 1                                                           XSMOD  2656
      DO 110 I=ISD,IE                                                     XSMOD  2657
      IF(KDIMAG(I) . EQ.  0)RETURN                                        XSMOD  2658
      IF(II . LT. ` 200)GO TO 107                                         XSMOD  2659
      PRINT 2, JJ,(IX(I),I=1,200)                                         XSMOD  2660
    2 FORMAT(//* SUBR TEMPLST. LIST HAS 200 ELEMENTS. JJ= *I2/(10I10))    XSMOD  2661
      RETURN                                                              XSMOD  2662
C                                                                         XSMOD  2663
  107 II=II+ 1                                                            XSMOD  2664
      N(JJ)=II                                                            XSMOD  2665
  110 IX(II)=KDIMAG(I)                                                    XSMOD  2666
      RETURN                                                              XSMOD  2667
      END                                                                 XSMOD  2668
      SUBROUTINE USELST(IIN,IOUT,JJ,KDIR,IIK)                             XSMOD  2669
C     MAKES COMPLETE SID/GID LIST WITH NO DUPLICATIONS                    XSMOD  2670
C     SEARCHES DIRECTORY FOR SID/GID NAME AND STORES CORRESPONDING        XSMOD  2671
C     RECALL ID NO. FOR N. USE LIST,PUTS THIS LIST IN ORDER.              XSMOD  2672
C     ENTERS WITH IIN CONTAINING SID/GID LIST WITH POSSIBLE DUPLICATIONS  XSMOD  2673
C     RETURN WITH IIN CONTAINING RECALL IDS IN ORDER.                     XSMOD  2674
```

```
C                                                                          XSMOD  2675
          COMMON  /OPTS/ KDIMAG(1100),N(2),IALL,IB                         OPTS      2
          DIMENSION  XKDIMAG(1100)                                         OPTS      3
          EQUIVALENCE  (KDIMAG,XKDIMAG)                                    OPTS      4
          DIMENSION  IIN(200),IOUT(150),KDIR(300),HOL(2)                   XSMOD  2677
          DATA  HOL(1), HOL(2)  /3HSID, 3HGID/                             XSMOD  2678
C                                                                          XSMOD  2679
C         CHECK LIST FOR DUPLICATIONS                                      XSMOD  2680
          IOUT(1)=IIN(1)                                                   XSMOD  2681
          IE=N(JJ)                                                         XSMOD  2682
          JE=1                                                             XSMOD  2683
          IF(IE .EQ. 1)GO TO 125                                           XSMOD  2684
          DO 120 I=2,IE                                                    XSMOD  2685
          DO 110 J=1,JE                                                    XSMOD  2686
          IF(IIN(I) .EQ. IOUT(J))GO TO 115                                 XSMOD  2687
  110     CONTINUE                                                         XSMOD  2688
          JE=JE+1                                                          XSMOD  2689
          IOUT(JE)=IIN(I)                                                  XSMOD  2690
          GO TO 120                                                        XSMOD  2691
  115     PRINT 10, HOL(JJ),IIN(I)                                         XSMOD  2692
   10     FORMAT(/* DUPLICATION IN *A3* LIST. *I4)                         XSMOD  2693
  120     CONTINUE                                                         XSMOD  2694
  125     DO 130 I=1,200                                                   XSMOD  2695
  130     IIN(I)=0                                                         XSMOD  2696
C         SEARCH SID/GID LIST AND GET RECALL ID NO.                        XSMOD  2697
          K=0                                                              XSMOD  2698
          DO 150 I=1,JE                                                    XSMOD  2699
          DO 140 J=1,IIK                                                   XSMOD  2700
          IF(IOUT(I) .EQ. KDIR(J))GO TO 145                                XSMOD  2701
  140     CONTINUE                                                         XSMOD  2702
          PRINT 4, HOL(JJ),IOUT(I)                                         XSMOD  2703
    4     FORMAT(/* SUBR USELST. CANNOT FIND *A3,I5* IN DTRAN TAPE DIRECTORY.  XSMOD  2704
     1    *)                                                               XSMOD  2705
          GO TO 150                                                        XSMOD  2706
  145     K=K+1                                                            XSMOD  2707
          IIN(K)=KDIR(J+100)                                               XSMOD  2708
  150     CONTINUE                                                         XSMOD  2709
          IF(K .LE. 1)GO TO 190                                            XSMOD  2710
C         PUT RECALL IDS IN ORDER                                          XSMOD  2711
          IE=K-1                                                           XSMOD  2712
          DO 180 I=1,IE                                                    XSMOD  2713
          JB=I+1                                                           XSMOD  2714
          DO 170 J=JB,K                                                    XSMOD  2715
          IF(IIN(I) .LT. IIN(J))GO TO 170                                  XSMOD  2716
          DUM=IIN(I)                                                       XSMOD  2717
          IIN(I)=IIN(J)                                                    XSMOD  2718
          IIN(J)=DUM                                                       XSMOD  2719
          IF(J .EQ. K)GO TO 180                                            XSMOD  2720
          I=I-1                                                            XSMOD  2721
          GO TO 180                                                        XSMOD  2722
  170     CONTINUE                                                         XSMOD  2723
  180     CONTINUE                                                         XSMOD  2724
  190     N(JJ)=K                                                          XSMOD  2725
          RETURN                                                           XSMOD  2726
C                                                                          XSMOD  2727
          ENTRY USEALL                                                     XSMOD  2728
C         MAKES LIST OF ALL RECALL IDS IN DIRECTORY FOR SID OR GID         XSMOD  2729
C                                                                          XSMOD  2730
          IF(KDIR(1) .EQ. 0) RETURN                                        XSMOD  2731
          N(JJ)=IIK                                                        XSMOD  2732
          DO 200 I=1,IIK                                                   XSMOD  2733
  200     IIN(I)=KDIR(I+100)                                               XSMOD  2734
          RETURN                                                           XSMOD  2735
          END                                                              XSMOD  2736
          END MODULE                                                       XSMOD  2737
          MODULE TAPE                                                      XSMOD  2738
          PROGRAM TAPE(INPUT,OUTPUT,TAPE1,TAPE6,TAPE7)                     XSMOD  2739
C         WRITES A FILE TAPE1 TO BE USED BY EVXS. TAPE OPTION IS PROCESSED XSMOD  2740
C         LAST. GID/SID ALREADY IN LIBRARY ARE RETRIEVED. THOSE ADDED DURING  XSMOD  2741
C         THIS RUN ARE READ FROM TAPE7/6.                                  XSMOD  2742
C                                                                          XSMOD  2743
```

```
C          FLAGS                                                          XSMOD    2744
C          IALL      0  SELECTED GID/SID                                  XSMOD    2745
C                    1  ALL GID/SID                                       XSMOD    2746
C                                                                         XSMOD    2747
           COMMON  /X/ X(4850)                                            X           2
           DIMENSION  ITID(1),TDES(8),E(400),TK(4000),MD(400),PROB(10),   ANGD        2
     1        NMU(10)                                                     ANGD        3
           EQUIVALENCE  (ITID(1), X(1)), (TDES(1),X(2)), (NED,X(10)),     ANGD        4
     1        (E(1),X(11)), (ISYS,X(411)), (A,X(412)), (LEGN,X(413)),     ANGD        5
     2        (KT,X(414)), (TK(1),X(415)), (MD(1),X(4415)), (NINC,X(4815)), ANGD      6
     3        (PROB(1),X(4816)), (NMU(1),X(4826))                         ANGD        7
C                                                                         ANGD        8
           COMMON/DIR/  KGID(300),KSID(300),IDIR,IIG,IIX,IDIRSV           DIR         2
           DIMENSION  IDNOG(100),NWDSG(100),IDNOS(100),NWDSS(100)         DIR         3
           EQUIVALENCE  (KGID(101),IDNOG(1)),(KGID(201),NWDSG(1)),        DIR         4
     1                 (KSID(101),IDNOS(1)),(KSID(201),NWDSS(1))          DIR         5
           COMMON  /GRP/ IGID(1),GDES(8),IGG(1),MG(80),GE(80),KG(1),PH(500), GRP      2
     1      EK(500),VB(80),EMX(80)                                        GRP         3
           COMMON  /OPTS/ KDIMAG(1100),N(2),IALL,IB                       OPTS        2
           DIMENSION  XKDIMAG(1100)                                       OPTS        3
           EQUIVALENCE  (KDIMAG,XKDIMAG)                                  OPTS        4
           COMMON /CNTRL/ ISID(1),KID,ND,ITIDS(25),SDES(8),IR,ITIDR(50),NES, CNTRL    2
     1        ES(4000),IDR(50),ME(50),IRS(50),QR(50),IZ,MIXT,K2           CNTRL       3
           DIMENSION  SIG(4000)                                           XS          2
           EQUIVALENCE  (SIG(1),X(1))                                     XS          3
C                                                                         XS          4
           DIMENSION  ESJ(3000)                                           XSMOD    2755
           EQUIVALENCE  (ESJ(1),ES(1))                                    XSMOD    2756
           DIMENSION  ITAPX(100),ITAPG(100)                               XSMOD    2757
           DIMENSION  TAD(100)                                            XSMOD    2758
           DIMENSION  IX(40)                                              XSMOD    2759
           EQUIVALENCE  (X(1),IX(1))                                      XSMOD    2760
C                                                                         XSMOD    2761
C          DEFINE FORTRAN VARIABLES TRANSMITTED BY DT LISTS               XSMOD    2762
           F. KGID=N. DIRGID. 1/                                          XSMOD    2763
           F. IDNOG=N. DIRGID. 2/                                         XSMOD    2764
           F. NWDSG=N. DIRGID. 3/                                         XSMOD    2765
           F. KSID=N. DIRSID. 1/                                          XSMOD    2766
           F. IDNOS=N. DIRSID. 2/                                         XSMOD    2767
           F. NWDSS=N. DIRSID. 3/                                         XSMOD    2768
           F. ITAPG=N. RDJOUT. 3/                                         XSMOD    2769
           F. ITAPX=N. RDJOUT. 4/                                         XSMOD    2770
           F. IIG=WORDS OF N. RDJOUT. 3/                                  XSMOD    2771
           F. IIX=WORDS OF N. RDJOUT. 4/                                  XSMOD    2772
           F. NGID=WORDS OF N. DIRGID. 1/                                 XSMOD    2773
           F. NSID=WORDS OF N. DIRSID. 1/                                 XSMOD    2774
           REWIND 1                                                       XSMOD    2775
           REWIND 6                                                       XSMOD    2776
           REWIND 7                                                       XSMOD    2777
C                                                                         XSMOD    2778
C          READ IN TAPE IDENTIFIER MESSAGE FROM INPUT CARDS.              XSMOD    2779
           DO 100 I=1,100                                                 XSMOD    2780
  100      TAD(I)=0                                                       XSMOD    2781
           DO 105 JB=1,100,8                                              XSMOD    2782
           JE=JB+7                                                        XSMOD    2783
           READ 1, (TAD(J),J=JB,JE)                                       XSMOD    2784
C          BLANK CARD STOPS READ                                          XSMOD    2785
           IF(TAD(JB) . EQ.  10H          )GO TO 110                      XSMOD    2786
  105      CONTINUE                                                       XSMOD    2787
    1      FORMAT(8A10)                                                   XSMOD    2788
  110      PRINT 2, (TAD(I),I=1,JB)                                       XSMOD    2789
    2      FORMAT(1H1*TAPE OPTION IS BEING PROCESSED. TAPE DESCRIPTION FOLLOWS XSMOD   2790
     1     . *///(5X13A10))                                               XSMOD    2791
C          WRITE TAPE DESCRIPTION AND NO. OF CROSS-SECTIONS ON TAPE1      XSMOD    2792
  150      WRITE(1) TAD,IIX                                               XSMOD    2793
           PRINT 6                                                        XSMOD    2794
    6      FORMAT(* TAPE DESCRIPTION HAS BEEN WRITTEN. *)                 XSMOD    2795
C                                                                         XSMOD    2796
C          WRITE GROUP LIBRARY                                            XSMOD    2797
C          IF GROUP LIBRARY EMPTY, SKIP TO WRITE EOF.  FIRST FILE CONTAINS XSMOD   2798
C          TAPE DESCRIPTION ONLY.                                         XSMOD    2799
           IF(KGID(1) . EQ.  0)GO TO 300                                  XSMOD    2800
C          IIG IS NO. OF GIDS TO WRITE TO TAPE1                           XSMOD    2801
```

```
      DO 299 J=1,IIG
      IF(ITAPG(J) .GT. 0)GO TO 230
C     GID WILL BE READ FROM TAPE7
      NEOF=0
  205 READ(7) IGID,GDES,IGG,(MG(I),I=1,IGG),(GE(I),I=1,IGG),KG,
     1      (PH(I),EK(I),I=1,KG)
      IF(EOF,7)210,220
  210 REWIND 7
      NEOF=NEOF+1
      IF(NEOF .LT. 2)GO TO 205
      PRINT 16, ITAPG(J)
   16 FORMAT(* TAPE. EOF ON TAPE7. CANNOT READ GID=*I6)
      GO TO 299
C
  220 IF(IGID .EQ. -ITAPG(J))GO TO 290
      GO TO 205
C     GID WILL BE RETRIEVED FROM LIBRARY.
  230 KID=ITAPG(J)
      CALL RETREV(KID,KGID,NGID,2,IERR,AM)
      IF(IERR .EQ. 0)GO TO 250
      PRINT 4,KID
    4 FORMAT(* GROUP LIBRARY RETRIEVAL ERROR FOR GID=*I5)
      GO TO 299
  250 F. IGID=D. ((AM)). 1/
      F. GDES=D. ((AM)). 2/
      F. IGG=D. ((AM)). 3/
      F. MG=D. ((AM)). 4/
      F. GE=D. ((AM)). 5/
      F. KG=D. ((AM)). 6/
      F. PH=D. ((AM)). 7/
      F. EK=D. ((AM)). 8/
  290 WRITE(1) IGID,GDES,IGG,(MG(I),I=1,IGG),(GE(I),I=1,IGG),KG,
     1      (PH(I),EK(I),I=1,KG)
      PRINT 18, IGID
   18 FORMAT(* GID*I5* WRITTEN TO TAPE1*)
  299 CONTINUE
  300 ENDFILE 1
C
C     WRITE CROSS SECTION AND ANGULAR DISTRIBUTION LIBRARIES.
C     IIX IS NO. OF SIDS TO WRITE TO TAPE1
      DO 399 J=1,IIX
      IF(ITAPX(J) .GT. 0)GO TO 330
C     SID WILL BE READ FROM TAPE6
      N6=0
  303 NEOF=0
  305 BUFFER IN(6,1)(X(1),X(4850))
  306 IF(UNIT,6)306,320,310
  310 NEOF=NEOF+1
      IF(NEOF .EQ. 2)GO TO 312
      GO TO 305
  312 IF(N6 .LE. 2)GO TO 315
      PRINT 8, ITAPX(J)
    8 FORMAT(/* TAPE. CANNOT FIND SID= *I5* ON TAPE6. *)
      GO TO 399
  315 N6=N6+1
      REWIND 6
      GO TO 303
  320 NEOF=0
      ND=IX(2)
      IR=IX(ND+11)
      L=LENGTH(6)
      KID=IX(1)
      IF(-KID .NE. ITAPX(J))GO TO 322
      WRITE(1) (X(I),I=1,L)
      GO TO 338
  322 BUFFER IN(6,1)(X(1),X(4850))
  323 IF(UNIT,6)323,322,325
  325 NEOF=1
      GO TO 305
C     RETRIEVE SID FROM LIBRARY
  330 KID=ITAPX(J)
```

```
      CALL RETREV(KID,KSID,NSID,1,IERR,AM)                          XSMOD
      IF(IERR . EQ. 0)GO TO 333                                     XSMOD
      PRINT 20, KID                                                 XSMOD
   20 FORMAT(* XS LIBRARY RETRIEVAL ERROR FOR SID *I5)              XSMOD
      GO TO 399                                                     XSMOD
C                                                                   XSMOD
C     DEFINE FTRAN VARIABLES FROM DT LISTS                          XSMOD
  333 N. CNTRL=D. ((AM)). N. CNTRL                                  XSMOD
      F. KID=N. CNTRL. 1/                                           XSMOD
      F. ND=N. CNTRL. 2/                                            XSMOD
      F. ITIDS=N. CNTRL. 3/                                         XSMOD
      F. SDES=N. CNTRL. 4/                                          XSMOD
      F. IR=N. CNTRL. 5/                                            XSMOD
      F. ITIDR=N. CNTRL. 6/                                         XSMOD
      F. NES=N. CNTRL. 7/                                           XSMOD
      F. ES=N. CNTRL. 8/                                            XSMOD
      F. IDR=N. CNTRL. 9/                                           XSMOD
      F. ME=N. CNTRL. 10/                                           XSMOD
      F. IRS=N. CNTRL. 11/                                          XSMOD
      F. QR=N. CNTRL. 12/                                           XSMOD
      F. IZ=N. CNTRL. 13/                                           XSMOD
      F. MIXT=N. CNTRL. 14/                                         XSMOD
      F. K2=N. CNTRL. 15/                                           XSMOD
      F. LLL=N. CNTRL. 16/                                          XSMOD
C     WRITE CONTROL RECORD TO TAPE1                                 XSMOD
  335 WRITE(1) KID,ND,(ITIDS(I),I=1,ND),SDES,IR,(ITIDR(I),I=1,IR),NES, XSMOD
     1    (ES(I),I=1,NES),(IDR(I),I=1,IR),(ME(I),I=1,IR),(IRS(I),I=1,IR)  XSMOD
     2    ,(QR(I),I=1,IR),IZ,MIXT,K2,LLL                            XSMOD
C     CROSS SECTION RECORDS                                         XSMOD
  338 IF(ITAPX(J) . LT. 0)GO TO 350                                 XSMOD
      N. XS=D. ((AM)). N. XS                                        XSMOD
      N. ESJ=D. ((AM)). N. ESJ                                      XSMOD
      DO 345 K=1,IR                                                 XSMOD
      F. SIG=N. XS(K)                                               XSMOD
      F. IWD1=WORDS OF N. XS(K)                                     XSMOD
      IF(IRS(K) . NE. 0)GO TO 340                                   XSMOD
      WRITE(1) (SIG(I),I=1,IWD1)                                    XSMOD
      GO TO 345                                                     XSMOD
  340 F. ESJ=N. ESJ(K)                                              XSMOD
      F. IWD2=WORDS OF N. ESJ(K)                                    XSMOD
      WRITE(1) (SIG(I),I=1,IWD1),(ESJ(I),I=1,IWD2)                  XSMOD
  345 CONTINUE                                                      XSMOD
      GO TO 365                                                     XSMOD
C     CROSS- SECTION RECORDS FROM TAPE6                             XSMOD
  350 DO 360 K=1,IR                                                 XSMOD
      BUFFER IN(6,1)(X(1),X(4850))                                  XSMOD
  352 IF(UNIT,6)352,356,354                                         XSMOD
  354 PRINT 22, KID,K,IR                                            XSMOD
   22 FORMAT(* EOF IN XS RECORDS FOR SID= *I5* RECORD= *I3* OF *I3*. XSMOD
     1 SID ON TAPE1 PROBABLY NO GOOD. *)                            XSMOD
      GO TO 360                                                     XSMOD
  356 L=LENGTH(6)                                                   XSMOD
      WRITE(1) (X(I),I=1,L)                                         XSMOD
  360 CONTINUE                                                      XSMOD
C     ANGULAR DISTRIBUTION RECORDS                                  XSMOD
  365 IF(ITAPX(J) . LT. 0)GO TO 380                                 XSMOD
      N. ANGD=D. ((AM)). N. ANGD                                    XSMOD
      DO 370 K=1,ND                                                 XSMOD
      F. ITID=N. ANGD(K). 1/                                        XSMOD
      F. TDES=N. ANGD(K). 2/                                        XSMOD
      F. NED=N. ANGD(K). 3/                                         XSMOD
      F. E=N. ANGD(K). 4/                                           XSMOD
      F. ISYS=N. ANGD(K). 5/                                        XSMOD
      F. A=N. ANGD(K). 6/                                           XSMOD
      F. LEGN=N. ANGD(K). 7/                                        XSMOD
      F. KT=N. ANGD(K). 8/                                          XSMOD
      F. TK=N. ANGD(K). 9/                                          XSMOD
      F. MD=N. ANGD(K). 10/                                         XSMOD
      F. NINC=N. ANGD(K). 11/                                       XSMOD
      WRITE(1) ITID,TDES,NED,(E(I),I=1,NED),ISYS,A,LEGN,KT,(TK(I),I=1,KT  XSMOD
     1    ),(MD(I),I=1,NED),NINC                                    XSMOD
```

```
  370     CONTINUE                                                          XSMOD    2944
          GO TO 398                                                         XSMOD    2945
 C        ANG DISTR RECORDS FROM TAPE6                                      XSMOD    2946
  380     DO 395 K=1,ND                                                     XSMOD    2947
          BUFFER IN(6,1) (X(1),X(4850))                                     XSMOD    2948
  382     IF(UNIT,6)382,386,384                                             XSMOD    2949
  384     PRINT 24, KID,K,ND                                                XSMOD    2950
   24     FORMAT(* EOF IN AD RECORDS FOR SID= *I5* RECORD= *I3* OF *I3*.   XSMOD    2951
     1    SID ON TAPE1 PROBABLY NO GOOD. *)                                 XSMOD    2952
          GO TO 395                                                         XSMOD    2953
  386     L=LENGTH(6)                                                       XSMOD    2954
          WRITE(1) (X(I),I=1,L)                                             XSMOD    2955
  395     CONTINUE                                                          XSMOD    2956
  398     ENDFILE 1                                                         XSMOD    2957
          PRINT 14, KID                                                     XSMOD    2958
   14     FORMAT(* SID*I5* WRITTEN TO TAPE1*)                               XSMOD    2959
  399     CONTINUE                                                          XSMOD    2960
          ENDFILE 1                                                         XSMOD    2961
          PRINT 12                                                          XSMOD    2962
   12     FORMAT(* TAPE1 FOR EVXS HAS BEEN WRITTEN. DOUBLE EOF AT TAPE END. *)  XSMOD    2963
          REWIND 1                                                          XSMOD    2964
          RETURN                                                            XSMOD    2965
          END                                                               XSMOD    2966
          SUBROUTINE RETREV(KID,KDIR,IIK,JJ,IERR,AM)                        XSMOD    2967
 C        RETRIEVES SELECTED SID/GID FROM DTRAN TAPE                        XSMOD    2968
 C                                                                          XSMOD    2969
          DIMENSION  KDIR(300),HOL(2),M(6)                                  XSMOD    2970
          DATA  HOL(1),HOL(2) /3HSID,3HGID/                                 XSMOD    2971
          DATA  (M(I),I=1,6) /10H(*GID*I3) ,10H(*GID*I2) ,10H(*GID*I1) ,   XSMOD    2972
     1        10H(*SID*I4) ,10H(*SID*I3) ,10H(*SID*I2) /                    XSMOD    2973
 C                                                                          XSMOD    2974
          IERR=0                                                            XSMOD    2975
 C        SEARCH FOR GID/SID IN DIRECTORY                                   XSMOD    2976
          DO 110 I=1,IIK                                                    XSMOD    2977
          IF(KID . EQ.  KDIR(I))GO TO 120                                   XSMOD    2978
  110     CONTINUE                                                          XSMOD    2979
          PRINT 2, HOL(JJ),KID                                              XSMOD    2980
    2     FORMAT(* CANNOT FIND *A3,I4* ON DTRAN TAPE. *)                    XSMOD    2981
          IERR=1                                                            XSMOD    2982
          RETURN                                                            XSMOD    2983
 C        GET ASSOCIATED RECALL ID NO.                                      XSMOD    2984
  120     K=KDIR(I+100)                                                     XSMOD    2985
          IF(JJ . EQ.  2)GO TO 130                                          XSMOD    2986
 C        CHOOSE FORMAT FOR ENCODING DT NAME                                XSMOD    2987
          IF(KID . LT.  10000)I=4                                           XSMOD    2988
          IF(KID . LT.  1000)I=5                                            XSMOD    2989
          IF(KID . LT.  100)I=6                                             XSMOD    2990
          GO TO 140                                                         XSMOD    2991
  130     IF(KID . LT.  1000)I=1                                            XSMOD    2992
          IF(KID . LT.  100)I=2                                             XSMOD    2993
          IF(KID . LT.  10)I=3                                              XSMOD    2994
  140     ENCODE(7,M(I),AM)KID                                              XSMOD    2995
 C        RECONSTRUCTED RECALL CARD                                         XSMOD    2996
          D. ((AM))=D. DT00010 *MHX1* F. K                                  XSMOD    2997
          PRINT 4, AM                                                       XSMOD    2998
    4     FORMAT(1X,A10* HAS BEEN RETRIEVED. *)                             XSMOD    2999
          RETURN                                                            XSMOD    3000
          END                                                               XSMOD    3001
          END MODULE                                                        XSMOD    3002
          MODULE RECLIB                                                     XSMOD    3003
 C        MODULE RECLIB CAME FROM KAPL                                      XSMOD    3004
          DIMENSION IRP(200,3)                                              XSMOD    3005
          COMMON /TAPE40/ ISTATUS,IREEL,IRP                                 XSMOD    3006
          COMMON /RECALL/ RECALL(2,1000),USE(200)                           XSMOD    3007
          EQUIVALENCE (IREEL,REEL)                                          XSMOD    3008
          DATA IRP /600*0/                                                  XSMOD    3009
 C        N. REEL = PHYSICAL REEL NO. OF LIBRARY TAPE                       XSMOD    3010
 C        N. RECLID = (RECALL ID, NO. OF WORDS, I = 1, NO OF IDS ON TAPE)   XSMOD    3011
 C        N. USE = ID NUMBERS TO BE RETRIEVED FOR THIS JOB                  XSMOD    3012
          F. NX= WORDS OF N. RECLID                                         XSMOD    3013
    1     NO = NX/2                                                         XSMOD    3014
```

```
          ITEST = 2*NO - NX
          IF (ITEST . EQ. 0)  GO TO 2
          PRINT 102
  102     FORMAT(1H0,10(1H*),*POSSIBLE ERROR IN N. RECLID - JOB WILL CONTINUE
     1     BUT RESULT IS RISKY*,10(1H*),//)
  101     FORMAT(1H0,10(1H*),*N. RECLID CARD IN ERROR OR MISSING - JOB TERMIN
     1    ATED*,10(1H*),//)
    2     IF (NO . LE. 1000)  GO TO 3
          PRINT 102
    3     F. RECALL = N. RECLID
          F. REEL = N. REEL
          ISTATUS = 1
          F. NUSE = WORDS OF N. USE
          IF (NUSE . LE. 200)  GO TO 4
          PRINT 103
  103     FORMAT(1H0,10(1H*),*USER REQUESTING MORE THAN 200 RECALLS - JOB TE
     1    RMINATED*,10(1H*),//)
          STOP
    4     F. USE = N. USE
          DO 5 I = 1,NUSE
          DO 6 J = 1,NO
          IF (USE(I) . EQ. RECALL(1,J))  GO TO 7
    6     CONTINUE
          PRINT 104, USE(I)
  104     FORMAT(1H0,10(1H*),*USER REQUESTED RECALL NOT IN SUPPLIED RECALL L
     1    IST - SKIPPED - RECALL ID=*I5,5X,10(1H*),//)
          GO TO 5
    7     IRP(I,1) = IREEL
          IRP(I,2) = RECALL(1,J)
          IRP(I,3) = RECALL(2,J)
    5     CONTINUE
C         SAVE RFL + 3
          IADD = 3 + 1 - LOCF(USE(1))
          RETAIN3 = USE(IADD)
          CALL TAPE40(IERROR)
          IF (IERROR . NE. 1)  GO TO 8
          PRINT 105
  105     FORMAT(1H0,10(1H*),*ERROR IN TAPE40 - JOB TERMINATED*,10(1H*),//)
          STOP
    8     CONTINUE
          CALL OVERLAY(7HRECALL,0,0)
          CALL READ40
C         RESTORE RFL + 3
          USE(IADD) = RETAIN3
          RETURN
          END
          SUBROUTINE TAPE40(IERROR)
C         THIS SUBROUTINE PREPARES A FILE CALLED TAPE40 WHICH CONTAINS ALL
C         THE NECESSARY INFORMATION FOR RECALLING DATA WITH IN A JOB.
C         TAPE40 IS READ BY PROGRAM RECALL CALLED FROM MODULE RECALL.  ALL
C         INPUT TO TAPE40 IS VIA A COMMON BLOCK OF THE STRUCTURE
C
C         COMMON /TAPE40/ ISTATUS,IREEL,IRP(200,3)
C
C         WHERE ISTATUS IS SET AS FOLLOWS
C              ISTATUS = 0, NO SAVE OR RECALL
C              ISTATUS = 1, JUST RECALL
C              ISTATUS = 2, JUST SAVE
C              ISTATUS = 3, BOTH SAVE AND RECALL.
C         IREEL CONTAINS THE REEL NO. ON WHICH ALL SAVES ARE TO BE MADE.
C         IREEL CONTAINS THE REEL NO. ON WHICH ALL SAVES ARE TO BE MADE.
C         FOR EACH RECALL CARD ENCOUNTERED IRP IS FILLED OUT AS FOLLOWS
C              IRP(I,1) = TAPE NO.       )
C              IRP(I,2) = ID             ) ALL INTEGERS
C              IRP(I,3) = NO. OF WORDS   )
C         IF ANY ERROR OCCURES DURING THE EXECUTION OF TAPE40 A MESSAGE IS
C         ISSUED, IERROR IS SET = 1, AND A RETURN EXECUTED.  IERROR IS SET
C         = 0 AFTER SUCCESSFUL COMPLETION OF SUBROUTINE TAPE40.
          INTEGER ORDER,CONVERT,TAPE,STAPE,SHIFTL,SHIFTR
          DIMENSION IRP(200,3),ITAPE(10),BUF(83),IRPX(200,2),
     1     IBUF(64),ORDER(200)
```

```
      COMMON /TAPE40/ ISTATUS,IREEL,IRP
      EQUIVALENCE (IRP(1),IRPX(1)),(IRP(401),ORDER(1))
      COMMON /CHECKX/ IBUF
      DATA ITAPE/10*0/
      IERROR = 0
      NOTAPE = 0
   13 CALL SETPARM(3,6HTAPE40,BUF,83)
      REWIND 40
      IBUF(1) = -1
      IBUF(2) = 0
    1 DO 2 I = 1,200
      TAPE = IRP(I,1)
      IF (TAPE .EQ. 0)  GO TO 3
      CALL LSHIFT(IRP(I,2))
      NOB = IRP(I,3)/511
      CALL PACKR(NOB,IRP(I,2))
      DO 4 J = 1,NOTAPE
      IF (TAPE .EQ. ITAPE(J))  GO TO 2
    4 CONTINUE
      NOTAPE = NOTAPE + 1
      IF (NOTAPE .LE. 10)  GO TO 6
      PRINT 100
  100 FORMAT(//////////5X,84HTOO MANY TAPES SPECIFIED ON RECALL CARDS.
    1 TEN IS THE MAXIMUM NO. ALLOWED IN ONE JOB/////)
      IERROR = 1
      RETURN
    6 ITAPE(NOTAPE) = TAPE
    2 CONTINUE
    3 DO 5 I = 1,NOTAPE
      IF (ITAPE(I) .NE. IREEL)  GO TO 5
      STAPE = ITAPE(I)
      ITAPE(I) = 6H999999
      GO TO 8
    5 CONTINUE
      IF (ISTATUS .EQ. 1)  GO TO 8
C     SAVE AND RECALL BUT NO RECALLS OFF SAVE TAPE
      NOTAPE = NOTAPE + 1
      ITAPE(NOTAPE) = 6H999999
      STAPE = IREEL
      ISTATUS = -3
C     SHIFT RIGHT WITH 0 FILL SO ORDER WILL ALWAYS BE ON ABSOLUTE VALUE
    8 DO 14 I = 1,NOTAPE
   14 ITAPE(I) = SHIFTR(ITAPE(I),6)
      CALL ARRANGE(ITAPE,NOTAPE)
C     SHIFT BACK AND RESTORE BLANK
      DO 15 I = 1,NOTAPE
   15 ITAPE(I) = SHIFTL(ITAPE(I),6) + 55B
      IF(ITAPE(NOTAPE).EQ.6H999999) ITAPE(NOTAPE) = STAPE
      DO 7 I = 1,NOTAPE
    7 IBUF(2+I) = ITAPE(I)
      IBUF(2) = NOTAPE
      LIB = 2 + NOTAPE
      IF (ISTATUS .EQ. -3)  NOTAPE = NOTAPE-1
      DO 9 I = 1,NOTAPE
      K = 0
      TAPE = ITAPE(I)
      DO 10 J = 1,200
      IF (IRPX(J,1) .EQ. 0)  GO TO 11
      IF (IRPX(J,1) .NE. TAPE)  GO TO 10
      K = K + 1
      ORDER(K) = IRPX(J,2)
   10 CONTINUE
   11 CALL ARRANGE(ORDER,K)
      DO 12 J = 1,K
      IX = ORDER(J)
      CALL RSHIFT(IX)
      IX = CONVERT(IX)
      CALL SHIFTC(IX)
      CALL CHECK(LIB)
      IBUF(LIB) = ITAPE(I)
      CALL PACKR(IX,IBUF(LIB))
```

```
      CALL CHECK(LIB)
12    IBUF(LIB) = ORDER(J)
      CALL CHECK(LIB)
9     IBUF(LIB) = 0
      CALL CHECK(LIB)
      IBUF(LIB) = 0
50    WRITE (40) IBUF
      END FILE 40
      RETURN
      END
      SUBROUTINE CHECK(LIB)
      COMMON /CHECKX/ IBUF(64)
      LIB = LIB + 1
      IF (LIB . LE. 64) RETURN
      WRITE (40) IBUF
      LIB = 1
      RETURN
      END
C -
      SUBROUTINE ARRANGE(IX,KX)
      DIMENSION IX(200)
      K = KX
      IF (K . LE. 1) RETURN
      K = K - 1
      IF (K . EQ. 0) K = 1
2     J = 0
      DO 1 I = 1,K
      IF (IX(I) . LE. IX(I+ 1)) GO TO 1
      J = 1
      ITEMP = IX(I)
      IX(I) = IX(I+ 1)
      IX(I+ 1) = ITEMP
1     CONTINUE
      IF (J . EQ. 0) RETURN
      GO TO 2
      END
C -
      INTEGER FUNCTION CONVERT(INTX)
      INT = INTX
      K = 1
      JP = 10
1     IF (INT/JP . EQ. 0) GO TO 2
      K = K + 1
      JP = JP*10
      GO TO 1
2     CONVERT = 55555555555555555555B
      DO 3 I = 1,K
      JP = JP/10
      NDIG = INT/JP
      I6 = 6*I
      CALL DIGIT(CONVERT,NDIG,I6)
3     INT = INT - NDIG*JP
      RETURN
      END
      SUBROUTINE READ40
      INTEGER RIGHT,FIELDR
      DIMENSION BUF(82),IBUF(64),NAMESR(2)
      DIMENSION NTAPE(10),ISET(506)
      DATA NAMESR /7HSAVEAND,10HRECALLDATA/
C     NEED TO USE CLASP LAIS BECAUSE THIS ROUTINE - NOT IN - OPENS LAIS
      EQUIVALENCE(DTPATHL(494),LAIS)
C     ABOVE EQUIVALENCE MAY BE SUJECT TO CHANGE WHENEVER CLASP IS
C     RECOMPILED
C     494=756B=LOCF OF BLOCK DT10002- LOCF OF BLOCK DT00001+61B
C     FROM CLASP MAP----- LOCF(DT10002)=34335B
C     FROM CLASP MAP----- LOCF(DT00001)=33440B
      CALL SETPARM(3,6HTAPE40,BUF,82)
      REWIND 40
      READ (40) IBUF
      NOTAPE = IBUF(2)
      LAIS = 1
```

```
      CALL OPENFI(LAIS,2,NAMESR,0,1,1)
      DO 40 I = 1,NOTAPE
40    NTAPE(I) = IBUF(I+2)
      K = 1
      J = NOTAPE + 3
      DO 50 I = 1,NOTAPE
      ISET(K) = NTAPE(I)
      RIGHT = 1
      NRHS = 1
42    J = J + 1
      IF (J .LE. 64) GO TO 43
      READ (40) IBUF
      J = 1
43    NRHS = NRHS + RIGHT - 1
44    RIGHT = FIELDR(IBUF(J))
      K = K + 1
      IF (K .GT. 506) GO TO 60
      ISET(K) = IBUF(J)
      CALL PACKR(NRHS,ISET(K))
      J = J + 1
      IF (J .LE. 64) GO TO 45
      READ (40) IBUF
      J = 1
45    IF (IBUF(J) .NE. 0) GO TO 42
      J = J + 1
      K = K + 1
      IF (K .GT. 506) GO TO 60
      IF (IBUF(J) EQ. 0) GO TO 61
50    CONTINUE
      GO TO 61
60    PRINT 100
100   FORMAT(1H0,5X,48H*****TOO MANY DATA RECALL CARDS JOB ABORTED*****)
      CALL PP(3HABT,0,0)
61    ISET(K) = 0
      CALL STOSET(ISET,-LAIS,506)
      CALL SAVEFI(LAIS)
      CALL CLOSFI(LAIS)
62    CONTINUE
      LAIS = 1
      CALL FINDDI(LAIS,2,NAMESR,0,1,1)
      IF (LAIS .GT. 0) GO TO 63
      PRINT 101
101   FORMAT(1H0,5X,51H*****CANT FIND LAIS FILE JOB ABORTED*****)
      CALL PP(3HABT,0,0)
63    CONTINUE
      RETURN
      END
          IDENT   PACKAGE
          ENTRY   DIGIT
DIGIT     BSS     1
          SA3     B3
          SB3     X3
          SB4     60
          MX4     54
          SA1     B1
          SA2     B2
          SX2     X2+33B
          SB4     B4-B3
          LX2     B4,X2
          LX4     B4,X4
          BX1     X4*X1
          BX6     X1+X2
          SA6     B1
          ZR      B0,DIGIT
          ENTRY   PACKR
PACKR     BSS     1
          MX1     30
          SA2     B2
          BX2     X1*X2
          SA1     B1
          BX6     X1+X2
```

```
                SA6    B2                                            XSMOD     3299
                ZR     B0,PACKR                                      XSMOD     3300
                ENTRY  LSHIFT                                        XSMOD     3301
LSHIF     T     BSS    1                                             XSMOD     3302
                SA1    B1                                            XSMOD     3303
                LX1    30                                            XSMOD     3304
                BX6    X1                                            XSMOD     3305
                SA6    B1                                            XSMOD     3306
                ZR     B0,LSHIFT                                     XSMOD     3307
                ENTRY  RSHIFT                                        XSMOD     3308
RSHIF     T     BSS    1                                             XSMOD     3309
                SA1    B1                                            XSMOD     3310
                AX1    30                                            XSMOD     3311
                BX6    X1                                            XSMOD     3312
                SA6    B1                                            XSMOD     3313
                ZR     B0,RSHIFT                                     XSMOD     3314
                ENTRY  SHIFTC                                        XSMOD     3315
SHIFT     C     BSS    1                                             XSMOD     3316
                SA1    B1                                            XSMOD     3317
                MX2    6                                             XSMOD     3318
                SX3    55B                                           XSMOD     3319
                LX3    54                                            XSMOD     3320
START     X     LX1    6                                             XSMOD     3321
                BX4    X2*X1                                         XSMOD     3322
                BX4    X4-X3                                         XSMOD     3323
                NZ     X4,STARTX                                     XSMOD     3324
                MX2    30                                            XSMOD     3325
                BX6    -X2*X1                                        XSMOD     3326
                SA6    B1                                            XSMOD     3327
                ZR     B0,SHIFTC                                     XSMOD     3328
                ENTRY  FIELDR                                        XSMOD     3329
FIELD     R     BSS    1                                             XSMOD     3330
                MX1    30                                            XSMOD     3331
                SA2    B1                                            XSMOD     3332
                BX6    -X1*X2                                        XSMOD     3333
                ZR     B0,FIELDR                                     XSMOD     3334
                END                                                  XSMOD     3335
                IDENT  SHIFTL                                        XSMOD     3336
*               AUTHOR - R. J. CULLEN                                XSMOD     3337
*               COPIED MANUALLY FROM LISTING BY W. E. SCHILLING      XSMOD     3338
*                                                                    XSMOD     3339
                ENTRY  SHIFTL,SHIFTR                                 XSMOD     3340
                VFD    36/0HSHIFTL,24/2                              XSMOD     3341
SHIFT     L     VFD    60/0                                          XSMOD     3342
                SA1    B1                                            XSMOD     3343
                SA2    B2                                            XSMOD     3344
                BX6    X1                                            XSMOD     3345
                ZR     X2,SHIFTL                                     XSMOD     3346
                MX6    1                                             XSMOD     3347
                SB2    X2-1                                          XSMOD     3348
                AX6    B2,X6                                         XSMOD     3349
                BX1    -X6*X1                                        XSMOD     3350
                LX6    B2,X1                                         XSMOD     3351
                LX6    1                                             XSMOD     3352
                EQ     SHIFTL                                        XSMOD     3353
                VFD    36/0HSHIFTR,24/2                              XSMOD     3354
SHIFT     R     VFD    60/0                                          XSMOD     3355
                SA1    B1                                            XSMOD     3356
                SA2    B2                                            XSMOD     3357
                BX6    X1                                            XSMOD     3358
                ZR     X2,SHIFTR                                     XSMOD     3359
                MX6    1                                             XSMOD     3360
                SB2    X2                                            XSMOD     3361
                AX1    B2,X1                                         XSMOD     3362
                SB2    B2-1                                          XSMOD     3363
                AX6    B2,X6                                         XSMOD     3364
                BX6    -X6*X1                                        XSMOD     3365
                EQ     SHIFTR                                        XSMOD     3366
                END                                                  XSMOD     3367
          END MODULE                                                 XSMOD     3368
          END JOB                                                    XSMOD     3369
```

JJ:250(45)

☆ US GOVERNMENT PRINTING OFFICE: 1974 — 784-529/39