176
3-6-74

D-1620

UCRL-51493

ETRANMS — A ONE-DIMENSIONAL MONTE CARLO ELECTRON/PHOTON TRANSPORT CODE FOR MULTIMATERIAL TARGETS

Frederick R. Kovar

November 30, 1973

Prepared for U.S. Atomic Energy Commission under contract No. W-7405-Eng-48

LAWRENCE LIVERMORE LABORATORY

University of California/Livermore

MASTER

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 $___*; Microfiche $0.95

*Pages	NTIS Selling Price
1-50	$4.00
51-150	$5.45
151-325	$7.60
326-500	$10.60
501-1000	$13.60

TID-4500, UC-32
Mathematics and Computers

LAWRENCE LIVERMORE LABORATORY
University of California Livermore, California 94550

UCRL-51493

ETRANMS — A ONE-DIMENSIONAL MONTE CARLO ELECTRON/PHOTON TRANSPORT CODE FOR MULTIMATERIAL TARGETS

Frederick R. Kovar

MS. date: November 30, 1973

Contents

ETRANMS – A ONE-DIMENSIONAL MONTE CARLO ELECTRON/PHOTON TRANSPORT CODE FOR MULTIMATERIAL TARGETS

Abstract

ETRANMS is an LLL-modified version of the one-dimensional electron/photon transport code ETRAN 15 developed at the National Bureau of Standards. The major modifications include the use of LLL photon cross sections and the application to multislab, multimaterial targets. The code uses Monte Carlo sampling techniques to calculate electron and photon transport and energy and charge deposition within target material subjected to electron or photon radiation. It has been programmed to be a very rapid running, user-oriented code for use on LLL's CDC 7600 computers.

Introduction

ETRANMS is a Fortran code, using Monte Carlo sampling techniques to calculate electron/photon transport and energy and charge deposition in multimaterial one-dimensional slab systems. It is presently in operation at Lawrence Livermore Laboratory on the CDC 7600 computers.

The basis of ETRANMS is the code ETRAN 15, originally developed by Berger and Seltzer[1] at the National Bureau of Standards. The ETRAN 15 code is capable of following electrons and photons through plane-parallel, single-material targets of finite thickness, and tabulating charge deposition and energy deposition within zones and electron and photon transport across boundaries.

Here at LLL, we have modified the ETRAN 15 code by including the LLL cross sections of Plechaty and Terrall[2] for photon interactions and added the capability of following the electron photon interactions in multimaterial targets. Several other features were added to the code when it was programmed for our 7600 computers, so as to make it a very rapid running, user-oriented code. However, because of this programming, there is little core storage space remaining in ETRANMS; if other functions are desired, they must be done on an auxiliary code.

Because of our specific needs to calculate fields and voltages, we have developed an auxiliary code, ETRANED, to take the output of ETRANMS and calculate the electric field and potential in each zone. ETRANED also includes the capability of rezoning the problem,

transferring charge from zone to zone (e.g., current conduction), giving CRT output, and evaluating the statistical accuracy of the Monte Carlo calculations. This auxiliary code may be used, modified, or replaced according to the needs of the individual user.

This document touches briefly on the physics within ETRANMS and on the mechanics of running the various routines necessary to solve the electron/photon transport problem. The changes incorporated within L.L.L.'s version of ETRANMS are thoroughly discussed. The reader is referred to the references[1] on ETRAN 15 for a more complete analysis of the physics in the code and the complete glossary of items in the original code. Finally, a section on code verification is included.

In the Appendix is an example of how a typical electron/photon transport problem is to be set up and solved.

Physics In ETRANMS

The program begins by following either single photon or single electron histories, depending on the type of source. The source may be monoenergetic or spectral in nature (1 GeV maximum for electrons, 100 MeV for photons; 1 keV minimum energy), and it may be a plane source, a point source, or a cosine-law source. The code allows for arbitrary angular incidence on a target.

The target may consist of one material, or several, up to a total of five different materials. The target may be divided into as many as 20 different regions, where a region is specified by a particular material. The regions may be further subdivided into zones of uniform thickness and density. The maximum number of zones for any problem is 50. The ETRANMS code evaluates charge deposition and energy deposition within zones and computes charge transport across the boundaries separating regions. Running times of a few minutes on a CDC 7600 computer are required for reasonable statistical accuracy. The history of an input quantum is followed until it leaves a target surface or its energy drops below a specified cutoff energy ($\geq$1 keV). In turn, generations of the resultant photon/electron cascades are followed in like manner. Available output information includes photon/electron boundary or surface emission, differential in energy and direction, and charge deposition, energy deposition, and internal electron flux as a function of distance along the target normal.

Physical processes for photon and electron interactions included in ETRANMS are elastic and inelastic scattering, pair production, and photoelectric and density effects. Secondary entities followed are Compton electrons, photoelectrons, knock-on electrons, electron-positron pairs, annihilation photons, characteristic x rays, Auger electrons, and bremsstrahlung. Analytical solutions or partial solutions and approximations for the individual physical processes are used when available, supplemented by numerical methods when necessary. Probabilities for the competing physical processes were put into

tabular arrays from which the random sampling is done. Significant tape and computer core storage is saved, however, by including those variables of simple functional dependence as part of the code logic.

Physics information for the Monte Carlo choices is almost exclusively in the form of tabular arrays in which interpolations are made. Prior to running Monte Carlo histories, these arrays are formed by a code called DATAPAC which draws its information partially from analytic expressions incorporated within itself and partially from prepared tabular information stored in the public file, CUTE. Only arrays for the materials and energy range relevant to the particular problem being run are assembled. In what follows, the physics discussion of this two-step process of array assembly and Monte Carlo history is lumped together.

ELECTRON HISTORIES

Analytical formulations for electron transport are fraught with approximations and limitations. However, to rely on numerical integration of the transport equation (which involves a large number of variables), or to use random sampling techniques for the enormous number of collisions that an electron undergoes in slowing down, would be a formidable task. Thus, a Monte Carlo model is used wherein groups of electron interactions are followed, lumped together, and treated as a single step in a random walk. Energy loss and angular deflection for these composite steps are sampled from theoretical multiple scattering distributions (see Berger[3]).

Berger has described several schemes to trace each step in the random walk. The first, which he calls Class I, the simplest, relies entirely on grouping of collisions and involves a predetermined set of path lengths. A variation of Class I (called Class I') is based on a predetermined set of energy losses. The continuous slowing down approximation is used throughout Class I. Class II is based on a mixed procedure in which collisions with small energy losses and deflections are subject to grouping, but occasional "catastrophic" collisions are allowed and treated by standard sampling procedures using single-scattering cross sections.

Options are available in the code to use either the continuous slowing down approximation, where energy loss by collision is computed with the stopping power formula, or to use Class II procedures.

1. Input Distribution

As input we have the electron's initial energy and direction at the surface of the target. If the input is in the form of an energy spectrum, the spectrum is first integrated and divided into a number of equally probable energy bins, the number being the number of Monte Carlo histories desired. The different histories are taken in energy sequence, starting with the most energetic. If the input has an angular distribution, the angle of incidence is taken from a random choice of equally probable angles. If, instead of electrons, photons are incident on the target, they are handled in like manner. Creation of photons will arise in the discussion of an electron history and will be treated at that time.

2. Energy Step

The initial step in the calculation is to set up an energy grid, running from the maximum energy considered down to the desired cutoff energy. Input options exist for either equally spaced energy bins or a grid based on the rule $E_{n+1} = K E_n$, where K is a constant slightly less than one. The latter approach allows for a finer grid at the lower energies and is probably preferable since it insures that the most probable angular deflection per energy step is nearly constant. The grid then defines the size of the step on which the Monte Carlo electron histories are sampled.

The energy loss in a step is sampled from a distribution that is a convolution of a Landau distribution[4] with a Gaussian. The Gaussian represents the Blunck-Leisegang correction,[5] which takes into account, in an approximate manner, the effect of resonance scattering from atomic electrons.

The path length step corresponding to the energy step ΔE_n is calculated from

$$\Delta S_n = S_{n+1} - S_n = \int_{E_{n+1}}^{E_n} \frac{dE}{|dE/dS|},$$

where the denominator includes the mean energy loss in the target in the energy interval $\Delta E_n \equiv E_n - E_{n+1}$.

3. Electron Energy Loss Mechanisms

The average energy loss per unit path length follows the formulation of Rohrlich and Carlson.[6]

Electron-Electron Collisions

The energy transferred in an electron-electron collision depends on the distance of closest approach, the impact parameter. To obtain the average loss, an integration is required over the impact parameter. This corresponds to an energy transfer range from a minimum which is the binding energy of the target electron to a maximum which is one-half the incident electron energy. A mean excitation energy (I) is used in the ETRAN codes (NAS-NRC Publication 1133, Report No. 6). The collision cross section used in this integration is that of Møller,[7] using Born's time-dependent perturbation theory for the scattering of relativistic electrons by free electrons, and includes spin and exchange terms.

Knock-on Electrons

The production of knock-on electrons is sampled in each short step with the use of a probability distribution derived from the Møller cross section for collisions between free electrons. If a knock-on electron is produced, its direction is determined from conservation of energy and momentum in the electron-electron collision, at a random position within the short step. Only electrons with energies above some chosen cutoff energy are sampled, and they follow procedures similar to those for primary electrons.

Density Effect

In a real material the electrons are sufficiently packed so that polarization of near electrons reduces the field of the transient electron at greater distances. This effect, which causes a reduction in the rate at which the electron loses energy, is a function of the electron density, the electron polarization states available, and the velocity of the incident electron.

ETRANMS uses the work of Sternheimer[8] to make this correction. He does a Fourier analysis of both the electric field of the passing particle and the polarization of the medium. Coupling between the two increases with energy and falls with increasing atomic number. Figure 1 (taken from Birkhoff[9]) indicates the magnitude of the effect for a few sample materials.

Information from the work of Sternheimer is available for only 44 materials and elements. We have included interpolations for the balance of the elements. These are presently in our version of ETRANMS. A density effect calculation for all elements has been done by H. M. Colbert of Sandia Laboratories, Livermore. For those comparisons that have been made, the differences between Colbert's values and our interpolated values are insignificant.

Straggling

After traversing a distance ΔS in the target, monoenergetic electrons will have an energy distribution about the average energy $E - \Delta E$ because of the statistical nature of the collision-loss process. Collisions in which there is a large energy transfer are too infrequent to affect the distribution. If, in traversing a distance ΔS, electrons experienced the same number of collisions with a constant energy loss per collision, the resultant energy distribution would be nearly Gaussian, which turns out to be a fair approximation to the actual distribution. However, ETRANMS does its sampling from a more exact distribution from the work of Landau[4] and the later work of Blunck and Leisegang.[5] The latter reference includes as input the Møller cross section and the fact that the target electrons are bound.

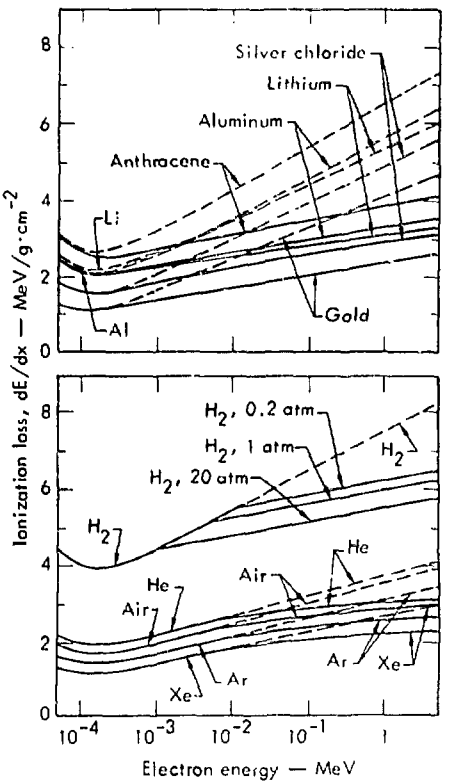

Fig. 1. Stopping power of various materials for high energy electrons. Dotted lines give stopping power if reduction due to density effect is neglected.

Landau's universal function has been evaluated and tabulated on DATATAPE, from which the energy broadening is calculated.

4. Radiation Loss

The passage of an electron through the Coulomb field of the nucleus results in the emission of radiation. The rate of this energy loss is tabulated on DATATAPE, from a separate code calculation of the work of Bethe and Heitler.[10] Born's

collision theory is used and includes screening of the nuclear field by bound electrons.

Continuous Bremsstrahlung

The production of bremsstrahlung photons is sampled in each short step with the use of a probability distribution derived from the bremsstrahlung cross sections (Berger,[3] Koch and Motz[11]). The energy of the secondary bremsstrahlung photons is subtracted from the energy of the electrons producing them. Thus, photon emission contributes to the energy loss straggling of the electrons. The photons are started out at a random position in the short step in a direction relative to that of the primary electron and specified by the bremsstrahlung emission angle.

For problems where thick-target bremsstrahlung is of prime interest, there is an option available (BNUM) to increase the rate of occurrence of bremsstrahlung events by a specified factor. This allows a sufficiently large number of events to be sampled for statistical significance. After such sampling, the events are renormalized to give unbiased photon production and energy-loss straggling.

Characteristic X Rays

Production of secondary characteristic x rays in each step is sampled with the use of the K-ionization cross sections of Arthurs and Moiseiwitsch[12] and Kolbenstvedt.[13] X-ray critical absorption and emission energies and intensities are taken from Hornyak.[14]

The treatment of characteristic x rays is analogous to that of the continuous bremsstrahlung except that their energies are not subtracted from the energies of the primary electrons. This subtraction has already been made implicitly in the evaluation of the electron collision loss.

An option is available to follow an artificially enlarged sample of characteristic x-ray photons (XNUM).

5. Angular Deflections

At this point the transported electron has a position and energy, but its new direction still must be obtained. ETRAN incorporates the multiple angular scattering work of Goudsmit and Saunderson.[15] The intensity of scattering per unit solid angle is given by an infinite Legendre series.

The Legendre series converges rather slowly, requiring a large number of expansion coefficients for computational accuracy. Recursion relations derived by Spencer[16] allow such a calculation to be practical. Calculated multiple angular scattering is in error only to the extent of the series cutoff and the error in the single-scattering cross section. The resultant angular distributions are valid even for single scattering.

We now have an energy, position, and direction for the electron after completing one step. Its energy loss is randomly deposited along its path and tabulated. If its energy after completing the step falls below the cutoff energy, its remaining energy and charge are deposited at that point. Otherwise, we continue for the next step.

6. Deposition of Energy

The target is divided into several regions which are subdivided further into thin sublayers called zones. The energy

deposited in each zone is recorded for each sampled track. The energy allowed to be deposited is that dissipated by electrons in inelastic collisions resulting in the production of slow secondary electrons with energies below the chosen cutoff value. The energy given to secondary electrons with energies above the cutoff is not scored immediately. Secondary electron histories are followed further so that their energy may be deposited in appropriate zones.

Photon losses (bremsstrahlung) are not scored immediately. Rather, these photons are allowed to penetrate further. Energy is deposited in zones through losses from the electrons set in motion by the photons.

7. Deposition of Charge

Deposition of charge is handled similarly to energy deposition. When the energy of an electron falls below that necessary to exit the zone boundary, a unit electron is recorded as deposited in that zone. When a secondary electron is created (knock-on, or photo-Compton) with sufficient energy to exit the zone, an electron is removed from that zone and followed until its energy falls below the cutoff and is deposited in a different zone (or until it totally exits the target). Electron-positron pairs are excluded from this scheme because, on the average, their production does not lead to a net transfer of charge.

PHOTON HISTORIES

Photons may either be created by processes such as bremsstrahlung or K x-ray emission, as mentioned in the section above, or they may be input directly as source particles. Photon transport in a target is done in the following way. The total interaction cross section, which is the sum of the photoelectric, Compton, and pair-production integrated cross sections, is found in a table look-up. This table is part of the input data from DATAPAC. For a description of how these tables are formed, the reader is referred to Grodstein.[17] In combination with this cross section information, a random number is chosen to determine the path length to the point at which the photon interacts. If this point is outside the target, then the photon's energy and direction are tabulated. If the point is inside the target, a random choice is made of the different interaction possibilities with weights in proportion to each interaction's contribution to the total cross section. If the choice happens to be Compton collision, the energy of the scattered photon is chosen from a wavelength distribution derived from the Klein-Nishina law. From the initial photon energy and direction and the scattered photon energy, the directions of the scattered photon and the scattered electron are easily calculated.

The photoelectric process yields photoelectrons, x rays, and Auger electrons. The angular distribution of photoelectrons is pretabulated, based on Sauter[18] and Fischer[19] cross sections. X-ray emission is discussed above in the section on electron histories. Finally, the energy distributions for electron-positron pairs are given by Bethe and Ashkin.[20]

Once the input photon's energy has fallen below the 1-keV cutoff, the code

goes back and follows the histories of any electrons created by the above processes, using the procedures outlined in the section on electron histories.

LIMITATIONS

A limitation of multiple scattering theory as presently constructed is that electron energy loss and angular deflection are treated independently. This limitation is carried over into the random sampling of ETRANMS. Consequently, the correlation between large electron energy loss and angular deflection (knock-on electrons, bremsstrahlung) is lost. However, for the case of the forward electron flux, electrons undergoing many small energy losses and angular deflections will dominate electrons which have been incorrectly sampled as forward-scattered with a large energy loss.

Multiple scattering theory is further restricted to collisions in which the target electron's initial velocity is much smaller than the velocity of the incident electron. As the incident electron's energy approaches that of the K-binding energy, this constraint is violated. However, this becomes evident only for the mid- and high-Z elements, and even then this error is relatively small, because only a few of the total number of target electrons are tightly bound. With the cross section input to ETRANMS as presently constituted, a conservative low-energy limit is provided by the K-shell binding energy of the atom involved.

Finally, the theoretical treatment in ETRANMS assumes random scattering centers, and will therefore be in error to the extent that diffraction and polarization effects are important.

Code Processing

ORGANIZATION

The organization of the ETRANMS program falls into four categories:

1. XSEC1—A cross section library stored in the public file, CUTE. This library contains the cross section data from Berger's DATATAPE plus interpolated elements and useful compounds.

2. DATAPAC—This is a program that accepts certain specific data from the cross section library, processes it, and expands it to form tabular arrays suitable for rapid look-up for the code ETRANMS.

3. ETRANMS—This is a Monte Carlo program that generates, through random sampling, large sets of electron and photon histories on the basis of the information in the cross section file created by DATAPAC.

4. ETRANED (for ETRAN EDIT)—This program organizes and analyzes the output of the Monte Carlo program, and through the use of other computational methods extends the usefulness of the information.

As presently configured at LLL, ETRANMS is coded in standard Fortran and is compiled and loaded via the PUTT compiler. A list of PUTT options may be obtained by typing PUTT HELP / T V at a Teletype station.

In order to execute entirely from small core memory (SCM), ETRANMS has an

overlay structure. There are seven primary (Ø-level) segments and six secondary (1-level) segments. The primary segments cannot execute a RETURN statement; they can only exit by calling another primary segment (or by ending the run). Each primary segment must connect itself to ETRANMS files (input, output, cross section, etc.) every time it is called. The secondary segments do execute RETURNs.

The source file for ETRANMS is stored in seven parts, or links, a different link for each primary segment. These source links (LINKØS, LINK1S, ... , LINK6S) are currently stored in the library file CUTE. ETRMAKE, a small code written for PROXY, will extract the source links from CUTE, merge them, compile and load the code, and store all of the resulting segments in a library file called ETRANMS. Figure 2 shows a listing of ETRMAKE. The execute line for ETRMAKE is:

PROXY ETRMAKE / T V

When ETRANMS is being compiled, several system routines are taken from the public file CUTE. These routines are *IOFILES, *IOFILEZ, *CHANE, *KWIZ, and *OVERLAY.

```
EXE SCRAWL SMERG CUTE $
LINKS$                                                          } Merge
LINKØS LINK1S LINK2S LINK3S LINK4S LINK5S LINK6S$               }
↑↑[PK>00]
$ALL\
PUTT (CUTE)LINKS,OL,B,115000,157500$                            } Compile and load
$ALL\
DESTROY ETRANMS
$ALL\
COPY COPY ETRANMS L 260000
$ALL\
LIB ETRANMS NEWFILE$
A LINKØ$
A LINK1$
A LINK2$
A LINK3$
A LINK4$
A LINK5$
A LINK6$
A ETRØ100$                                                      } Store in library file
A ETRØ101$                                                      }
A ETRØ102$                                                      }
A ETRØ103$                                                      }
A ETRØ104$                                                      }
A ETRØ105$                                                      }
RO$$                                                            }
S$                                                              }
E                                                               }
$AL.\                                                           }
*                                                               }
END
*
```

Fig. 2. Listing of ETRMAKE.

Also located in CUTE is the cross section file XSEC1 as well as DATAPAC, ETRANMS, and ETRANED. When DATAPAC is executed, the cross section information is extracted from CUTE. Since XSEC1 is the first file in CUTE it is accessed by the name CUTE rather than bringing it out of the public file CUTE. The file XSEC1 can be modified using RGXS. RGXS is a source file and is stored in CUTE.

Each of ETRANMS's primary segments performs a specific function. ETRANMS spends a minimal amount of time getting overlays from large core (LCM). The following is a brief description of the tasks the primary segments perform:

LINKØ—Calls IOFILEZ to set up all of the ETRANMS files. LINKØ is never called again.

LINK1—Stores default values into variables, reads the input deck, and calls DATAPAC (LINK6) if necessary.

LINK2—Sets up elements from the cross section file.

LINK3—Sets up calculations for the problem and prints out several tables. LINK1, LINK2, and LINK3 are called once for every internal ETRANMS "run" (IRNMAX).

LINK4—This is the heart of ETRANMS. LINK4 does all the physics calculations. For every internal ETRANMS "run" it is called ISTATS number of times.

LINK5—Prints the results of the calculations and calls LINK4 if another ISTATS calculation is required or calls LINK1 for another "run" calculation.

LINK6—This is the coding for DATAPAC.

RUNNING A PROBLEM

The actual running of a complete transport problem may be done either directly by the user from a Teletype station or through batch (production) running. Examples of both these techniques are discussed in the Appendix. This section describes the various codes and files necessary to ETRANMS, some of the NAMELIST variables for ETRANMS and ETRANED, and how to execute the programs.

1. Execution

Any of the three routines (DATAPAC, ETRANMS, or ETRANED) may be run directly out of CUTE, or may first be transferred into the user's private files (either from CUTE or photostore) and then run directly. The routine ETRANMS, for example, may be run in either of the following ways:

1) Directly out of CUTE: execute line is

```
X CUTE ETRANMS (FILE, INPUT) BOX NO ID ; T V
```

After completion of the problem the routine ETRANMS will be stored in the user's private files, and subsequent problems may be run without "X CUTE."

2) Transfer to user's file:

```
EXE CUTE ETRANMS DR. / T V
```

After "ALL DONE," then execute:

```
ETRANMS (FILE, INPUT) BOX NO ID ; T V
```

For the examples in this section it will be assumed that the routines already exist in the user's file, so that only the execute line in item 2 above will be shown.

2. INPUT/OUTPUT Files

The execution lines of the three routines specify input/output (I/O) files between the two parentheses, e.g.,

```
ETRANED (INPUT,OUTPUT) BOX NO ID   T V
```

This may also be written as

```
ETRANED (INPUT,OUTF=10000,P) BOX NO ID   T V
```

where OUTF is the output file name, 10000 is the length of file in octal, and .P is one of the options available. .P will cause the output file to be sent to the on-line printer.

For any of the routines, omitting the output file name causes the file to go to the high speed printer (HSP). Naming the file with no options causes the output file to remain on disk in the user's file, and it must be sent out by the user if he desires printed output. If no BOX NO is written on the execute line, output will be sent to box CØ4 as the default.

Other options available are:

.R for close read only,
.D for close and destroy,
.PUB for just open,
.H for HSP (send to HSP when done),
.P send to on-line printer 1 when done,
.L send to on-line printer 2 when done,
.TP for tape,
.X for "just skip over,"
.$ for destroy first then create.

Unless otherwise specified the program will try to open any existing file; if unsuccessful it will then try to create the file at the length specified or 50K if no length is specified. Omission of all options is available.

3. Routines

DATAPAC

The first step in running an electron/photon transport problem is to create the abbreviated cross-section library best suited for any specific calculations. This involves choosing the materials and the maximum and minimum energies to be used in the calculation. The energy spread through which the electron histories are followed is limited to about 64 energy bins, so that the fineness of the energy grid depends on the increment between the highest and lowest energy. For optimum accuracy, as small an energy spread as is compatible with the problem should be chosen.

To create the DATAPAC file for any material and energy range, the execute line is:

```
DATAPAC (XSECFIL,  n X 60K,  (INPF,OUTF,CUTE,PUB)
                                   BOX NO ID   T V
```

where

XSECFIL is the name the user gives to the cross section file being created.

n is the number of separate elements or compounds to be called out of the cross section library for inclusion in the DATAPAC-created file. Each element requires 60K octal spaces. If for example n = 5, the actual entry in the underlined portion of the execute line would be 360K in octal (i.e., 5 X 60K in octal).

INPF is the input file made up by the user and described in the appendix.

OUTF is the output file name (may be omitted; output then goes to HSP).

CUTE is the name of the large cross section library.

.PUB indicates that CUTE is a public file.

Upon completion of this program, the file XSECFIL will be placed in the user's private files. From there it may be immediately used as part of the input for an ETRANMS problem, placed in photostore, or placed on magnetic tape.

An example of a typical input deck for DATAPAC is shown in the Appendix.

The cross sections and parameters of some compounds that have rather universal use have been included in the library file equivalently as elements. These are such materials as water, air, bone, Lucite, etc. These compounds have IZIP > 100 (see NBS Circular 9836 (Ref. 1)), whereas the pure elements have IZIP = their atomic number. Compounds made up by the user specify IZIP = -1.

The sequencing of materials in DATAPAC is unimportant. However, the NSET sequencing in ETRANMS is important and must be done in ascending order. ETRANMS uses the set number from the DATAPAC file to specify the elements and their cross sections.

ETRANMS

The execute line for ETRANMS is:

```
ETRANMS (XSECFIL, INPETR) BOX NO ID   T V
```

The DATAPAC-created cross section file (e.g., XSECFIL) and the ETRANMS input deck (e.g., INPETR) must be present in the user's private file prior to executing ETRANMS.

The ETRANMS problem may be run for either electrons or photons as source particles, either monoenergetic or spectral in energy input. The problem may be geometrically zoned for up to 50 zones with up to five different materials, and up to 20 regions (a region is a grouping of zones of the same material and thickness).

In the program, ETRANMS tabulates:

1. Number and energy of electrons generated by knock-ons, pair production, photoelectric, Compton, and Auger effects.
2. Number of electrons and photons reflected and transmitted at slab boundaries, differential in energy and direction.
3. Energy deposition in each zone.
4. Absorbed energy as a function of energy in thick targets.
5. Production and emergence from the target of continuous bremsstrahlung and characteristic x rays.
6. Electron deposition, D field, E field, and potential in each zone.

The output file created is called HOUTPUT, and will be automatically sent to HSP unless otherwise specified in the execute line.

The meaning of the NAMELIST variables in ETRANMS may be found in NBS Circular 9837.[1] Variables added or expanded upon by LLL are listed in Table 1.

There is an option within ETRANMS which causes part or all of an ETRANED deck to be created and punched. To exercise this option, place the variables IEDIT and BOXNO in the ETRANMS input deck. The punched cards will be delivered to the box number specified in BOXNO and a copy of the deck will remain in the user's files with the name EDITA. The IEDIT options are given in Table 2.

For IEDIT = 2 or 4 ETRANMS punches the following ETRANED variables or vectors:

IMAX	DIACON
CAL	DEN
NZ	EAV
TAG	Z
MORE	QIN

If ETRANMS is executed in a multiple run mode (IRNMAX > 1), all of the above variables will be punched for the first run and the Z and QIN arrays are punched for each run as well as the $INPUT and $ cards (for use in ETRANED). On the last run ETRANMS punches a card denoting MORE = 0. Any other ETRANED information must be specified in ETRANMS.

Table 1. NAMELIST variables added or expanded upon by LLL. For basic NAMELIST variables see NBS Circular 9837.[1]

TTY = 0 or 1	For TTY = 0, the problem begins running immediately. TTY = 1 allows altering of the input deck from the Teletype before the problem begins. When TTY = 1, whether the namelist was altered or not, the directive $ must be typed to start the program.
CCM	Fluence in the input beam (cal/cm^2). This is used to calculate fields and voltages for a given fluence.
IUNT	Describes type of units for B or BR, the target thickness. If IUNT = +, then -B- is region boundaries. If IUNT = -, then -B- is delta between region boundaries.
IUNT = 1	Enter -B- in CSDA units (continuous slowing down approximation).
IUNT = 2	Enter -B- in g/cm^2.
IUNT = 3	Enter -B- in mils (also enter DENSE in g/cm^3).
IUNT = 4	Enter -B- in cm (also enter DENSE in g/cm^3).

Table 2. IEDIT options.

IEDIT = 0	No edit; Q's and Z's not punched.
IEDIT = 1	Punches Q's and Z's.
IEDIT = 2	Expects EDIT cards after first $. A complete ETRANED deck will be created and punched by ETRANMS. The deck will also be on disk under the name EDITA after ETRANMS finishes.
IEDIT = 4	Same as 2 except DOSE (energy deposited in each zone) is also punched. If DOSE is punched it must be removed from the ETRANED deck before running ETRANED. DOSE is not a namelist variable in ETRANED.
BOXNO = 0H BOX CXX	BOXNO and the first two words of TAG form the box number (address) and ID for the punched deck.
ENDEDIT	Ends EDIT input.

In order to collect a set of similar problems to estimate statistical accuracy, the ETRANMS problem may be run over and over, provided a new starting random number is used. This may be done by either of the following procedures: (1) By inputting the variable IRNMAX = n into the ETRANMS input file, where "n" is the number of separate runs desired. This technique then requires that the end of file $ be followed by n - 1 $ETRAN $ cards. Several different problems may be stacked by this method, provided that any changes in the input variables are placed between the $ETRAN and the $ of the successive problems. (2) By writing into the ETRANMS input file a new random number under the variable INRAN. The code requires a "good" random number. The code automatically begins with one random number and will generate a new one for each successive run in sequence, as in (1) above. However, when similar problems are input separately for statistics, it is suggested that the random number be obtained from the last ETRANMS output file under the label of INRAN.

If the allotted time runs out before the ETRANMS problem is completed, the problem may be restarted again by executing

+ETRANMS / T V

The output from the ETRANMS code is well labeled and quite self-explanatory. For the most part this output data existed in the original ETRAN 15 code. For our particular purposes, we were interested in computing the electric field and potential at a zone with reference to some ground plane, normally taken as the front surface of the sample, where the E field and potential are set arbitrarily to zero. Thus, we have added data printout for fields, voltages, and electrons born in each zone.

The quantities desired are obtained from an integration of the Maxwell's equation $\nabla \cdot D = \rho$ which reduces to a one-dimensional Poisson's equation. D is the electric displacement vector.

The electrons deposited in a zone, D field, E field, and potential are related in the following manner:

$$D(n) = C \sum_{i=1}^{n} Q_i ,$$

$$E(n) = \frac{D(n)}{\epsilon(n)}$$

$$V_n = - \frac{E_n + E_{n-1}}{2} \Delta X_n + V_{n-1},$$

where

D(n) is the electric displacement vector at the back surface of the nth zone,

C is a scaling constant (negative since Q_i are electrons),

Q_i is the number of electrons in the ith zone as computed by ETRANMS,

$\epsilon(n)$ is the permittivity of the nth zone, $\frac{1}{\epsilon} = \emptyset$ for conductors,

E(n) or E_n is the electric field at the back surface of the nth zone,

V_n is the potential of the nth zone with respect to some referenced potential (e.g., $V_{n=1} = \emptyset$),

ΔX_n is the thickness of the nth zone.

ETRANED

The auxiliary code ETRANED is capable of taking output information from

ETRANMS and giving charge deposition, D field, E field, and potential in each zone. It also can transfer charge from zone to zone (current conduction), give CRT output for Q, D, E, and V, and estimate the mean and standard deviation for the run. At present there is a 100-zone limit in ETRANED, although this can be enlarged easily. Thus, there is latitude to add zones (e.g., gaps, voids) or couple output from several ETRANMS problems into one ETRANED problem.

In order to check whether the zoning is fine enough, there is also an option that merges paired adjacent zones and then recalculates the charge, D and E fields, and voltage for the new zoning. In general, if the latter quantities do not change after rezoning, the original zone size was probably adequate. This option may be exercised twice for any ETRANED problem.

The execute line for ETRANED is:
ETRANED (INPUT) BOX NO ID / T V

The file INPUT is the data deck input by the user. See the ETRANED Appendix for example. The ETRANED output file is named OUTEDIT. It will be automatically sent to HSP, or it may be sent to output in any of the usual ways. NAMELIST input variables for ETRANED are given in Table 3.

NOTE: In ETRANED, a false zone must be created at the front of the sample; thus, there are really n + 1 zones in the ETRANED problem while there were n in ETRANMS. The namelist variable, DIACON, QIN and DEN must have a value input for the first zone. DIACON and DEN may simply use the same values as they have for the first zone in ETRANMS. QIN must have some charge value input; e.g., the number of backscattered electrons may be used.

At present, the estimator of variance in the ETRANED code makes use of several repetitive runs of the ETRANMS problem, treating each run as a separate experiment, and obtaining the rms deviation by standard techniques. The error estimates are made only on computed voltages. The voltages of interest to us were really those between a few arbitrary zones; e.g., we were interested in calculating the potential difference between two plates of a capacitorlike sample, or between several different layers in a dielectric and a ground plane. As input variables in ETRANED, we use VSTAT1 and VSTAT2, where as many as five different paired zones are specified. Thus, differential voltages between a specified zone in VSTAT1 (e.g., m) and a specified zone in VSTAT2 (e.g., n) are summed over the several different runs and averaged to get the mean differential voltage,

$$\overline{\Delta V}_{mn} = \frac{1}{k}\sum_{i=1}^{k}\left(V_m^i - V_n^i\right) = \frac{1}{k}\sum_{i=1}^{k}\Delta V_{mn}^i ,$$

where k is the number of separate runs. The root-mean-square (rms) deviation, σ, is computed from the variance

$$\sigma^2 = \frac{1}{k-1}\sum_{i=1}^{k}\left(\overline{\Delta V}_{mn} - \Delta V_{mn}^i\right)^2 .$$

The common mode voltage, $(V_m + V_n)/2$, is averaged in a similar fashion.

VSTAT1 and VSTAT2 specify the zones between which the potential differences

Table 3. NAMELIST input variables for ETRANED.

TAG = 3H	Title or comment card; comment follows 0H, same as in ETRANMS.
KASE	Number of edit run.
IMAX	Number of particles input.
EAV	Average energy of input particles (in MeV).
CAL	Fluence (cal/cm^2) in input beam (used for scaling from IMAX and EAV).
PLOTIT	If = 1, plot Q, D, E, V as a function of position in target; if = 0, no plot.
KEEPIT	If = 1, store plot of Q, D, E, or V for future use or TMDS viewing; if = 0, no store.
MORE	If = 1, more edit problems to follow; if = 0, no more to follow.
MPY	= 1, 2, or 4 is rezoning option: 1 = no rezoning; 2 = successive pairs of zones are merged; 4 = successive pairs of zones are merged again. Fields and voltages are recomputed after each rezoning.
TTY	= 0 or 1 is Teletype control, same as in ETRANMS.
NZ	Number of zones in problem.
DIACON	Relative dielectric constant for each zone.
KZ	Zones to be shorted together; format is KZ(n) = m; connects zone n to m (n > m).
WZ	Weighting factor for zones.
QIN	Electrons deposited in each zone (obtained from ETRANMS).
DEN	Density of material in each zone (g/cm^3).
Z	Zone boundaries in units of g/cm^2 (from ETRANMS).
VSTAT1	= a, b, c, d, e
VSTAT2	= v, w, x, y, z These are zones over which statistics on voltage are calculated. Statistics will be given for differential mode ($V_v - V_a$) or common mode ($V_v + V_a$)/2, etc., for paired zones, a-v, b-w, c-x, d-y, and e-z, for as many as five different combinations, averaged over all the runs in the EDIT problem. The zones need not all be different. The answer appears as the mean of the differential or common mode voltage with a sigma equal to the standard deviation of the samples.
WHEN	Specifies when to do the statistics on the voltages. If = 1, statistics are obtained before any rezoning; if = 2, after the first rezoning (MPY=2); if = 4, after second rezoning (MPY=4). WHEN defaults to 2.

are calculated. Thus, VSTAT1 = 1, VSTAT2 = 10 implies that differential voltage between zones 1 and 10 ($V_1 - V_{10}$) and common mode ($V_1 + V_{10}$)/2 are computed for each of the runs, summed, and averaged to get the mean and the rms deviation on the samples computed.

This averaging may be done before or after any rezoning (see variable WHEN).

The user may incorporate similar procedures for any variables for which he may require statistics.

Code Verification

Several procedures and experiments have been undertaken to prove the reliability and accuracy of the code as adapted by LLL. The first such verification was to check whether the inclusion of the multimaterial, multislab changes incorporated within ETRANMS would give consistent results with the original code. Early calculations were done wherein the output data (in the form of forward- and back-scattered electrons at boundary interfaces) from multislab, multimaterial targets was compared with similar forward- and back-scattered output data from single-material targets and superposed by hand calculations. Agreement was excellent.

Several experiments done here at LLL and others collected from the literature have lent themselves well to code verification of electron surface emission from various targets. Calculations have been made for electron or photon sources, monoenergetic or spectral in energy input. Target materials have been metals and dielectrics. The comparisons between calculation and experiment have been discussed in detail.[21] Further verification comparisons between codes and calculations have been made by Berger[22] and Colbert and Dolan.[23]

In general, the agreement between calculation and experiment for electron surface emission or scattering is good to ±30% for incident photons from 10 to 100 keV in energy, and for incident electrons from 10 keV to 20 MeV for the materials studied.

There remains one set of experiments where agreement between calculation and experiment has not been good. This is for the phenomenon of "charge transport," i.e., motion of charge within a target sample, which has been reported by Weingart et al.[24] We are presently working on the reconciliation of this discrepancy.

To underline the fast-running capabilities of the ETRANMS code, a test comparison was made between ETRANMS and SANDYL, a similar coupled electron/photon transport code applicable to one-, two-, or three-dimensional geometry and developed by Colbert[25] at Sandia Laboratories, Livermore. For a relatively simple comparison in one-dimensional geometry, ETRANMS's running time was one-third that of SANDYL.

Acknowledgments

I would like to acknowledge several people for their very significant contributions to the establishment of this program at LLL. The initial reprogramming of ETRAN 15 and inclusion of the multimaterial option was accomplished by A. J. Ciplikas.* Responsible for the initial analysis of the physics and for parts of the physics portion of this writeup was Robert H. Ballett. Linnea Cook has programmed some of the more recent changes in the code as well as contributing to the code section of this report. Dr. Hans Kruger has guided the program from its inception.

*Presently at Science Applications, Inc., La Jolla, Calif.

References

1. M. J. Berger and S. M. Seltzer, Electron and Photon Transport Programs, NBS Circulars 9836, 9837 (1968).
2. E. F. Plechaty and J. R. Terrall, Photon Cross Sections 1 keV to 100 MeV, Lawrence Livermore Laboratory Rept. UCRL-50400-VI (1968).
3. M. J. Berger, "Monte Carlo Calculation of the Penetration and Diffusion of Fast Charged Particles," in Methods in Computational Physics, edited by B. Alder, S. Fernbach, and M. Rotenberg (Academic Press, New York, 1963), vol. 1, p. 135.
4. L. Landau, J. Phys. USSR 8, 201 (1944).
5. O. Blunck and S. Leisegang, Z. Phys. 128, 500 (1950).
6. F. Rohrlich and B. C. Carlson, Phys. Rev. 93, 38 (1954).
7. C. Møller, Ann. Phys. (Leipzig) 14, 531 (1932). See especially p. 568.
8. R. M. Sternheimer, Phys. Rev. 88, 851 (1952); Phys. Rev. 103, 511 (1956); Phys. Rev. 145, 8 (1966).
9. R. D. Birkhoff, "The Passage of Fast Electrons Through Matter," in Handbuch der Physik (Springer-Verlag, Berlin, 1958), vol. 34, p. 53.
10. H. Bethe and W. Heitler, Proc. Roy. Soc. A146, 83 (1934).
11. H. W. Koch and J. W. Motz, Rev. Mod. Phys. 31, 920 (1959).
12. A. M. Arthurs and B. L. Moiseiwitsch, Proc. Roy. Soc. (London) A247, 550 (1958).
13. H. Kolbenstvedt, J. Appl. Phys. 38, 4785 (1967).
14. W. F. Hornyak, "The Interaction of Gamma Rays with Matter," in Nuclear Spectroscopy: Part A, edited by F. Ajzenberg-Selove (Academic Press, New York, 1960), p. 211.
15. S. Goudsmit and J. L. Saunderson, Phys. Rev. 57, 24 (1940).
16. L. V. Spencer, Phys. Rev. 98, 1507 (1955).
17. G. W. Grodstein, X-Ray Attenuation Coefficients from 10 keV to 100 MeV, NBS Circular 583 (1957).
18. F. Sauter, Ann. Phys. (Leipzig) 9, 217 (1931).
19. J. Fischer, Ann. Phys. (Leipzig) 8, 821 (1931).
20. H. A. Bethe and J. Ashkin, "Passage of Radiation Through Matter," in Experimental Nuclear Physics, edited by E. Segrè (Wiley, New York, 1953).
21. F. R. Kovar, Comparison of ETRAN Calculations with Experiments, Lawrence Livermore Laboratory Rept. UCRL-74517 (1973).
22. M. J. Berger, memorandum to Dr. Edmunson on comparisons of electron transport calculations with experimental results (dated Jan. 2, 1969).
23. H. M. Colbert and K. W. Dolan, A Coupled Photon-Electron Monte Carlo Transport Code with Comparisons to Experiment, Sandia Laboratories, Livermore, Rept. RS-8341/69 (1970).

24. R. D. Weingart, R. H. Barlett, R. S. Lee, W. Hofer, and G. A. Fulk, Measurements of X-Ray-Induced Photoconductivity and Charge Transfer in Dielectric Structures, Lawrence Livermore Laboratory Rept. UCRL-74535 Preprint (1973).

25. H. M. Colbert, SANDYL: A Computer Program for Calculating Combined Photon-Electron Transport in Complex Systems, Sandia Laboratories, Livermore, Rept. SCL-DR-720109 (1973).

Appendix:
Sample Problem

This is an example of a rather simple electron transport problem to be solved using the ETRANMS code. We wish to obtain the dose enhancement and charge transport in a capacitorlike sample of borosilicate glass with aluminum electrodes. The source is a bremsstrahlung spectrum of photons from a tantalum target. Electrons were accelerated into the tantalum by means of a Blumlein generator.

The first step in the solution of any problem is to set up the proper geometry. This is generally straightforward, but attention must be paid to the allotment of zones among the materials as well as to the zone thickness in order to optimize the information obtained from the problem. The geometry for this problem is shown in Fig. A-1.

The target itself is made up of three regions (LMAX = 3): aluminum 0.4 mil thick, borosilicate glass 2 mils thick, and another layer of aluminum 0.4 mil thick. The aluminum layers are broken up into two zones each, and since we were primarily interested in effects within the glass, it contained the remaining 46 zones. It is not necessary, in general, to zone the problem so that the number of zones is divisible by two (or four); however, it is necessary to do so if the rezoning option in ETRANED (MPY = 2 or 4) is to be exercised.

The first calculational step is to set up the abbreviated cross section file for rapid look-up by means of DATAPAC. We know that the highest energy in our input spectrum is 67 keV, so we call for an EMAX in DATAPAC slightly above this (71 keV).

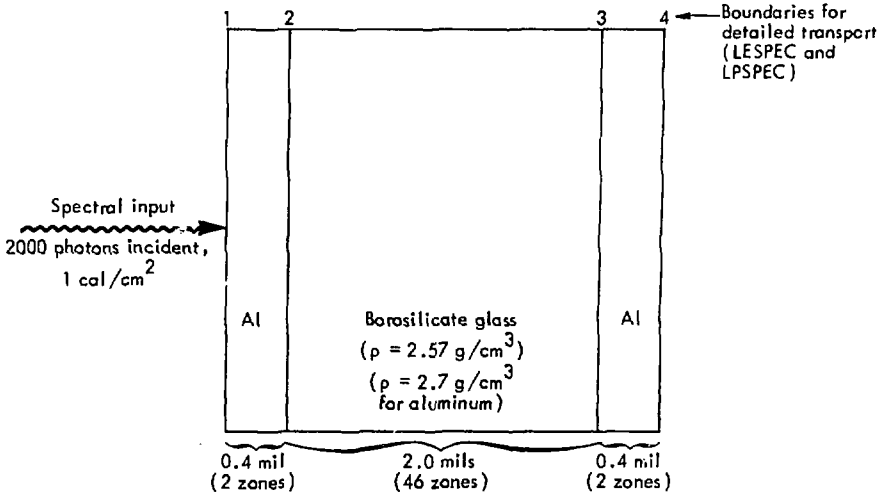

Fig. A-1. Geometry for sample problem.

```
$DTPAC TAG=ØH ALUMINUM
IZIP=13
EMAX=.071
EMIN=.ØØ1
NREC=5
$
$DTPAC TAG=ØH POLYETHYLENE
NSET=2
IZIP=112
$
$DTPAC TAG=ØH BERYLLIUM
NSET=3
IZIP=4
$
$DTPAC TAG=ØH COPPER
NSET=4
IZIP=29
$
$DTPAC TAG=ØH BOROSILICATE GLASS
NSET=5
IZIP=-1
JMAX=5
Z=5,8,14,19,30
W=8,26,10,1,1
$
```

Fig. A-2. Sample DATAPAC input deck.

The minimum energy is 1 keV. (Energies in ETRANMS problems are expressed in MeV.) We also know that we will be using this same source on targets of several other materials, so to save having to create several other DATAPAC files we have these elements in this one file which we call T9XBLUM.

The DATAPAC input deck is illustrated in Fig. A-2. Note the full usage of the IZIP variables, and that the only variables that are listed in successive sets are those that change. Those variables not listed are assumed to remain the same.

This input deck was given the name INPBLUM. Thus, to run DATAPAC, execute

X CUTE DATAPAC (T9XBLUM = 3ØØK, INPBLUM, CUTE.PUB) BOX NO ID / T V

After completion, the file T9XBLUM will be in the user's private file. Now to run ETRANMS, execute

X CUTE ETRANMS (T9XBLUM, INPETR) BOX NO ID / T V

The ETRANMS deck, INPETR, is shown in Fig. A-3.

In this problem, note that we have asked for multiple runs. Thus, IRNMAX = 1Ø, and we have added a succession of nine $ETRAN $ commands. We have also set ISTATS = 2. This cuts the number of separate subtotals within any given run to two. For a single run these commands are left out. Normally ISTATS = 1Ø, and statistics on a few quantities are obtained for every 1Ø% of a run.

We have asked for a full ETRANED deck to be punched. This is done by the order IEDIT = 2.

```
$ETRAN TAG=ØH DOSE ENHANCE BOSIGL AL BLUM
BOXNO=ØH BOX C12
IEDIT=2
IRNMAX=10
ISTATS=2
NSETS=2
NSET=1,5
TPCUT=.00100001
IMAX=2000
LZMAX=50
LMAX=3
JBF=2,46,2
MAT=1,2,1
B=0.4,2.0,0.4
IUNT=-3
IFLUX=2
TSAVE=.004
CCM=1.
DENSE=2.7,2.57,2.7
DIACON=0.,6.7,0.
KSPEC=4
JSPEC=18
ESP=.015,.017,.019,.021,.023,.025,.027,.029,.031,.033,.035,
.040,.045,.050,.055,.060,.065,.070
SPECIN=.10,.19,.285,.36,.47,.54,.58,.61,.62,.615,.60,
.50,.39,.30,.22,.17,.12,.092
$
MORE=1
VSTAT1=1,1,1,1,1
VSTAT2=11,21,31,41,51
WHEN=1
KEEPIT=0
PLOTIT=1
ENDEDIT
$ETRAN $
$ETRAN $
$ETRAN $
$ETRAN $
$ETRAN $
$ETRAN $
$ETRAN $
$ETRAN $
$ETRAN $
```

Fig. A-3. Sample ETRANMS deck.

Upon completion of the ETRANMS run, a complete set of input data for ETRANED is available under the file name EDITA. The ETRANED source deck (EDITA) for this problem is shown (abbreviated) in Fig. A-4.

To execute the ETRANED problem, write:

ETRANED (EDITA) BOX NO ID / T V

Any other input file may be used for other ETRANED runs.

```
$INPUT
MORE=1
VSTAT1=1,1,1,1,1
VSTAT2=11,21,31,41,51
WHEN=1
KEEPIT=0
PLOTIT=1
IMAX=       2000
CAL=       1.00
NZ= 50
TAG=0H  DOSE ENHANCE BOSIGL AL BLUM
MORE=1
DIACON= 1.,  2*    .0, 46* 6.70,  2*    .0
DEN= 1.,  2* 2.700, 46* 2.570,  2* 2.700
EAV=        .0335532
Z =         .0E+00,       1.372E-03,       2.743E-03,       3.027E-03,
   3.311E-03,       3.595E-03,       3.878E-03,       4.162E-03,
   4.446E-03,       4.730E-03,       5.014E-03,       5.298E-03,
```

. . . etc. for Z's and Q's , ending with $

Fig. A-4. ETRANED source deck (abbreviated).

ABKCON CONTROLLER FOR PRODUCTION RUNS

For production or batch running, an example of an ORDER and an ABKCON* deck which will run both ETRANMS and ETRANED is shown in Fig. A-5. The deck in Fig. A-5 accomplishes the following:

1. Destroys any leftover files associated with ETRANMS.
2. Reads ETRANMS, ETRANED, and the cross section file (in this case, T9XBLUM) from a tape.
3. Initializes ETRANMS.
4. Initializes ETRANED.
5. If the job is sense-switched, +ETRANMS and its associated files are written to tape (CL122, e.g.), ETRANMS is sense-switched and ETRANED is executed.

Figure A-6 shows a deck where only ETRANMS is executed.

*For an explanation of ABKCON see LLL Computation Division Rept. CIE-43.

```
*ID   471BXX          ETRAN2                    KOVAR          BOXC1
      >>2
*B
*B    THIS IS AN -ABKCON- JOB
*B    RESPONDS TO SW1.
*B    ---- DO NOT HIT SW1. TWICE ----
*T    BD304
*T    CL122
*B    READS -BD304-
*B    WHEN SW1. IS HIT OR WHEN DONE -- WRITES ON CL122
*B
*     XEQ ABKCON
*     XEQMES ABKINP
*     DATA ABKINP
/BOX  BOX C12 KOVAR  ABKCON
/SW1  5 50
/CTL  ALTER BOX / 1 .1
/MSG  ALL INPETR INPETR
/MSG99END
/CTL  DESTROY ETRANMS HOUTPUT ETRDUMP T9XBLUM ETRANED / 2 .2
/MSG  NO
/MSG  END
/CTL  RDFILES BD304 ETRANMS ETRANED T9XBLUM END / 5 1
/MSG99END
/CTL  ETRANMS (T9XBLUM ) BOX C12 KOVAR  ..ETRAN..   / 600 60
/MSG99
/MSG-9CTLJMP---- 7
/CTL  ADT *CL122 +ETRANMS ETRANMS HOUTPUT EDITA END / 5 1
/MSG99END
/CTL  +ETRANMS
/MSG55SW1.
/CTL  ETRANED (EDITA ) BOX C12 KOVAR  ..EDIT..  / 10 1
/MSG99
/CTL  ALLOUT OUTEDIT / 10 1
/MSG  BOX C12 KOVAR  ..EDIT..
/MSG99
/FIL  INPETR

  .
  .          } ETRANMS input deck goes here
  .

/END
```

Fig. A-5. Sample ORDER and ABKCON deck.

```
*ID   471BVV            ETRAN2                    KOVAR          BOXC1
     >>2
*B
*B    THIS IS AN -ABKCON- JOB UNDER MONITOR
*B    RESPONDS TO SW1.
*B    ---- DO NOT HIT SW1. TWICE ----
*T    BD304
*T    CL122
*B    READS -BD304-
*B    WHEN SW1. IS HIT OR WHEN DONE -- WRITES ON CL122
*B
*TIME ,,1
*     XEQ ABKCON
*     XEQMES ABKINP
*     DATA ABKINP
/BOX  BOX C12 KOVAR  ..ETRAN..
/SW1  -11 -100
/CTL  ALTER BOX / 1 .1
/MSG  ALL INPETR INPETR
/MSG99END
/CTL  DESTROY ETRANMS HOUTPUT ETRDUMP T9XBLUM / 2 .2
/MSG  NO
/MSG  END
/CTL  RDFILES BD304 ETRANMS T9XBLUM END / 3 .3
/MSG99END
/CTL  ETRANMS (T9XBLUM  )BOX C12 KOVAR  ..ETRAN.. / 600 60
/MSG99
/CTL  +ETRANMS
/MSG55SW1.
/CTL  ADT CL122 ETRANMS +ETRANMS HOUTPUT END / 3 .3
/MSG99END
/FIL  INPETR
   .  )
   .  ) ETRANMS deck
   .  )
      )
/END
```

Fig. A-6. Deck in which only ETRANMS is executed.