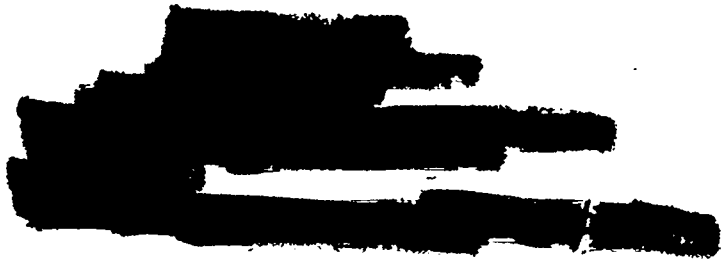
LA-7396-M

Manual

2

MCNP—A General Monte Carlo Code for Neutron and Photon Transport



University of California

LOS ALAMOS SCIENTIFIC LABORATORY
Post Office Box 1663 Los Alamos, New Mexico 87545

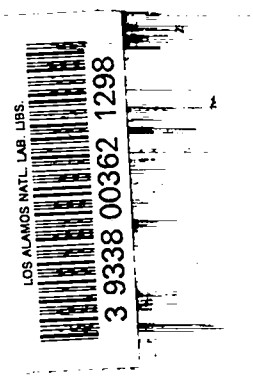
UNITED STATES
DEPARTMENT OF ENERGY
CONTRACT W-7405-ENG. 36

LA-7396-M
Manual

Special Distribution
Issued: July 1978

MCNP—A General Monte Carlo Code for Neutron and Photon Transport

LASL Group TD-6



FOREWORD

This manual is written as a practical guide for the use of our general purpose Monte Carlo code MCNP. The intent is that the first chapter describe the mathematics and physics found in MCNP. However, this discussion is not meant to be exhaustive--details of the particular techniques and of the Monte Carlo method itself will have to be found elsewhere. The second chapter shows the user how to prepare input for the code. The third chapter explains the output. The appendixes show how to use MCNP on the particular computer system at the Los Alamos Scientific Laboratory and also give details about some of the code internals that those who wish to modify the code may find useful.

Neither the code nor the manual is static. The code is changed from time to time as the need arises, and the manual is changed to reflect the latest version of the code.

MCNP and this manual are the product of a combined effort of the people in Group TD-6 at the Los Alamos Scientific Laboratory.

CONTENTS

MCNP - A General Monte Carlo Code
for Neutron and Photon Transport

LASL Group TD-6

ABSTRACT

The general purpose Monte Carlo code MCNP can be used for neutron, photon, or coupled neutron-photon transport. The code treats an arbitrary three-dimensional configuration of materials in geometric cells bounded by first- and second-degree surfaces and some special fourth-degree surfaces (elliptical tori).

Pointwise cross-section data are used. For neutrons, all reactions given in a particular cross-section evaluation (such as ENDF/B-IV) are accounted for. For photons, the code takes account of incoherent and coherent scattering, the possibility of fluorescent emission following photoelectric absorption, and absorption in pair production with local emission of annihilation radiation.

Standard optional variance reduction schemes include geometry splitting and Russian roulette, the exponential transformation, energy splitting, forced collisions in designated cells, flux estimates at point detectors, track-length estimators, and source biasing.

The standard output of MCNP includes two-way current as a function of energy, time, and angle with the normal, across any subset of bounding surfaces in the problem. Fluxes across any set of bounding surfaces are available as a function of time and energy. Similarly, the flux at designated points and the average flux in a cell (track length per unit volume) are standard tallies. Reactions such as fissions or absorptions may be obtained in a subset of geometric cells. The heating tallies give the energy deposition per starting particle. In addition, particles may be flagged when they cross specified surfaces or enter designated cells, and the contributions of these flagged particles to certain of the tallies are listed separately. All quantities printed out have their relative errors listed also.

NOTES:

CHAPTER 1. MCNP MATHEMATICS AND PHYSICS

This Chapter has not been completed; hopefully it will be available soon. In the meantime please refer to LA-4751 for physics of neutrons, mathematics, and some of the variance reduction schemes. For photons LA-5157-MS discusses the simple physics of MCG and the detailed physics of MCP.

Carter and Cashwell's book *Particle-Transport Simulation with the Monte Carlo Method* (ERDA Critical Review Series, TID-266607, 1975) is a good general reference and was written around what is in MCNP.

One useful feature of MCNP is correlated sampling, but this is not mentioned in the following chapter since it requires no special input cards. MCNP provides for correlated sampling to estimate the change in a quantity resulting from a small perturbation in the system. This technique enables the evaluation of small quantities that would otherwise be masked by the statistical errors of uncorrelated calculations. MCNP correlates a pair of runs by providing each new history in the unperturbed and perturbed problems with the same initial pseudorandom number (as well as the same sequence of subsequent numbers). Small changes can thus be effectively estimated since many of the histories in the two runs will be identical.

A final item to be mentioned here is the concept of weight cutoff. None of the above references discusses it, and it is a very user-abused quantity. In a large system without time or energy cutoffs, a particle could essentially rattle around forever doing no more than wasting computer time. Weight cutoff is a practical way to kill particles of little importance. Unfortunately weight cutoff is one on the most difficult of all Monte Carlo subjects to discuss. That is why you will not find much (if anything) written about it. It is very problem dependent and its setting is an art. If anything very specific is said at all, it will undoubtedly get you into trouble before the day is over. Hopefully the short, general discussion that follows will be some improvement to throwing salt over your left shoulder followed by the two step. The description of the MCNP cutoff cards is on page 35.

Two values which govern the game of Russian roulette played as particles lose weight in the normal transport process are inserted into the code. The second and smaller value is the so-called "weight cutoff" and defines the lower bound of weights tolerated by MCNP; the larger value is

the new weight the particle receives if it survives Russian roulette. The default values in MCNP are −1.0 and −0.5 (the negative signs make the values relative to the source weight). In order to explain how these values are used, suppose we assume a problem with all cell importances identical, say they are 1.0 in value. Call the upper and lower weight values U and L, respectively (for neutrons MCNP calls them WCN1 and WCN2). Then if a particle's weight W falls below L, with probability W/U the weight is jacked up to U, and with probability (1 − W/U) it is killed. The expected surviving weight is W. In the code a random number ξ is compared with W/U; if ξ < W/U, the particle survives with weight U, otherwise it is killed.

In a problem with varying cell importances, the values U and L are modified by MCNP in a given cell by multiplying these values by the ratio of the importance of the source cell to the importance of the given cell (the ratio of these importances multiplied by the source weight is the weight particles would have in the given cell just due to the splitting process).

Some care should be exercised by the user in selecting U and L. If the particles in a problem have a starting weight of 1.0 or if the average starting weight is 1.0 (and many problems fit into one of these categories), then U is usually chosen to be 1.0. The choice of L is generally more difficult and should take into consideration as many of the physical aspects of the problem as possible. For example, if capture is negligible in a particular problem and no biasing schemes (other than splitting which is accounted for, as described above) are used which cause particles to have unequal weights in a given cell, then the choice of L is easy − it may be chosen to be any value less than U, say L = 0.5. However, usually the choice is not so simple. If a cell can contain particles with a wide variation in weights, then the user has to consider that the low weighted particles cannot contribute as heavily to any tally as those with larger weights, and therefore some of them should be eliminated (to save machine time) in favor of fewer particles with larger weights. A very important consideration here is that the error is generally reduced in a calculation if the particles contributing to the tally have weights which are as nearly equal as possible. So, besides the fact that machine time may be wasted in transporting very low-weighted particles, there are statistical reasons for keeping weights from varying too widely. Naturally one does not want to play Russian roulette so frequently that the sample size deteriorates seriously.

Consideration of the other cutoffs, such as time and energy may help

the user to decide upon an appropriate weight cutoff. Certainly a few short runs of a problem may be helpful, even necessary, to arrive at the weight cutoff. It is particularly important to determine that the cutoff is not prejudicing the scores for the various tallies.

As mentioned, a good way to set the cutoffs is simply by trial and error - run a few short jobs with different weight cutoffs and see what the effect is. In particular, look at the MCNP summary pages (see page 44) for the number of tracks lost to weight cutoff. If a large number is lost to weight cutoff then the second weight cutoff L is probably too high or the cell importances are not appropriate to the problem. On the other hand, the second cutoff is probably too low if it has been reduced and the tallies are about the same but the running time increases.

Caution should be exercised when calculating system multiplication (see page 44) when using weight cutoffs. If the weight balance from Russian roulette in the weight cutoff game is not very small (and also from Russian roulette in surface importance sampling), the system multiplication may be in error especially if you are interested in multiplication to several decimal places. It probably will be more accurate in this case to enter the weight cutoff parameters as zero to have analog capture (see page 35) and avoid the weight cutoff game.

The units used in MCNP are as follows:

1. Lengths in centimeters
2. Times in shakes (10^{-8} sec)
3. Energies in MeV
4. Temperatures in MeV
5. Atomic densities in units of 10^{24} atoms/cm^3
6. Weight densities in gm/cm^3

CHAPTER 2. DESCRIPTION OF MCNP INPUT

The input to MCNP consists of several files, but only the one dealing with problem description, INP, is addressed in this Chapter. A second user-provided file is optional depending on the user's needs and contains a FORTRAN subroutine SOURCE which describes a non-standard source of neutrons or photons for the particular problem. Its requirements are described in Appendix A.

Throughout this chapter maximum dimensions will be given for various MCNP input. If the user wishes to increase any of these maximum values, he can do so by altering the code and re-compiling. Instructions for doing this are described in Appendix A.

I. INP FILE

The INP file can have two forms: (1) initiate-run and (2) continue-run.

A. Initiate-Run

This form is used to set up a Monte Carlo problem (describe geometry, materials, tallies, etc.) and run if desired. The initiate-run file has the following form:

ID Card
Cell Cards
.
.
.
Blank Line Delimiter
Surface Cards
.
.
.
Blank Line Delimiter
Data Cards
.
.
.
Blank Line Terminator

B. Continue-Run

Continue-run is used to continue running histories in a problem that was terminated earlier - for example, to run the job two hours on one night and then to run it an additional hour some later night.

In general, two files, in addition to the C. option on the MCNP execution line (see page 50), are needed for this procedure: (1) the continue-run file and (2) the run file RUNTPE (this is the default name). The run file is produced by MCNP in the initiate-run sequence and contains the geometry, cross sections, problem parameters, and all other necessary information to restart the job. The continue-run file must have the word CONTINUE starting in column one of the first line. The file has the following form:

CONTINUE
Data Cards
.
.
.
Blank Line Terminator

The data cards allowed in the continue-run file are a subset of the data cards available for an initiate-run file. The allowed continue-run data cards are CUTN, CUTP, CTME, PRDMP, NPS, LOST, and DBCN (see pages 34 through 38). If none of these items is to be changed, then the continue-run INP file is not required - only the run file and the C. option.

If the initiate-run producing the run file was stopped because of particle cutoff (NPS card, page 37), then NPS must be increased for a continue-run. The NPS card refers to the total number of histories to be run, including preceding continue-runs and the initial initiate-run. In a continue-run, entering a negative number on the NPS card will produce a print output file at the time of the last dump. No more histories will be run.

C. Card Format

All input lines (we will refer to "cards" in this manual) of INP are limited to columns 1-72 and consist of card images. Columns 73-80 may be

used for comments. The ID card is limited to one line and can contain any information the user desires. It usually contains information describing the particular problem. Note that a blank card is used as a delimiter and as a terminator.

Cell cards, surface cards, and data cards all conform to the same format. Columns 1–5 are reserved for the name (or number) associated with the card. The name (or number) field can appear anywhere in columns 1–5. Blanks in these columns indicate a continuation of the data from the last named card. Columns 6–72 are for free-field format data entry associated with the card name. With some exceptions on cell cards, separation between data entries is by one or more blank columns. In general, data entries may be of any type (fixed point, floating point, octal, or exponential) inasmuch as MCNP makes the appropriate conversion. An exception is any input to the PLOT overlay; see page 109.

Two features have been incorporated into the code to facilitate card preparation:

1. nR which means repeat the last entry before this statement n times, and

2. kI which means insert k linear interpolates between the entries immediately preceding and following this statement.

These features apply to both integer and floating point quantities and may be used wherever applicable.

Comment cards may be used anywhere in the INP file after the ID card and before the last blank terminator card. These cards must have the letter C in the first column of the card followed by four blanks, and then columns 6–80 are available to the user for any comments.

A duplication of any cards in the INP file is not permitted.

II. CELL CARDS

The problem number of the cell is in columns 1–5. Columns 6–72 will contain, in the following order:

(a) the cell material number. This material is described

by the material card (see page 29) with the same material number. If the cell is a void, a zero should be entered for the material number.

(b) the cell material density. A positive entry is interpreted as the atomic density in units of 10^{24} atoms/cm^3. A negative entry is interpreted as weight density in units of gm/cm^3. Nothing is entered for a void cell; the list in (c) is started immediately.

(c) a complete list consisting of the signed problem number of a surface bounding the cell followed by the problem numbers of those cells on the other side of the surface which would be entered by a neutron leaving the given cell; a second surface, if it exists, followed by the cells on the other side into which a particle may escape, etc., running through all bounding surfaces of the cell.

The problem numbers referred to above may be chosen by the user in any order he wishes. However, MCNP assigns the cells monotonically increasing integer values (beginning with one) as the cell cards are read in. These integers are referred to as program cell numbers. The user needs to be aware of the distinction between problem and program cell numbers since subsequent data cards will refer to one or the other.

As an example of problem versus program numbers, you may have numbered your first four cells in the input as 1, 2, 16, and 4. These are the problem numbers. However, MCNP internally reassigns sequential numbers to these cells as 1, 2, 3, and 4. These are the program numbers.

The problem numbers of the surfaces bounding a cell are signed quantities, the sign being determined by the sense (see LA-4751) any point within the cell has with respect to the surface. If the sense is positive, the sign may be omitted. In the list consisting of a problem surface number followed by the problem numbers of the cells on the other side, each entry except the last must be followed immediately by a comma. The absence of the comma (i.e., at least one blank) indicates that another bounding surface follows with its attendant cells on the other side. Ambiguity and reflecting surfaces (see LA-4751) are entered with no cells on the other side. In this case, omit the comma following the surface number. One

should never use a surface as an ambiguity surface which has also been designated as a reflecting surface on the surface card.

The maximum number of cells allowed is AMAX=175. The maximum number of surface numbers entered on all cell cards is 7*AMAX. The maximum number of cell numbers entered with all surfaces on the cell cards is 12*AMAX.

III. SURFACE CARDS

The problem number of the surface appears in columns 1–5. MCNP provides that any surface appearing in the problem may be a reflecting surface except a surface used as an ambiquity surface. To designate a reflecting surface, the space on the surface card immediately preceding the surface number should contain an asterisk. A neutron hitting such a surface is specularly reflected and the calculation continues. A point or ring detector should not be used with a reflecting surface. Columns 6–72 contain, in the following order,

(a) an alphabetic mnemonic indicating the surface type, and

(b) the required card entries for the specific surface in proper order.

A. Surfaces Defined by Equations

The surface types for surfaces defined by coefficients of equations, their mnemonics, and the order of the card entries are given in Table 2.1. The order of the surfaces listed in Table 2.1 is the same order in which the surfaces are used internally in MCNP.

The planes, spheres, and cylinders in Table 2.1 are straightforward and will be given no further explanation. Two options are available for cones: one or two sheets. The +1 or the −1 entry causes the one sheet cone treatment to be used. The sign of the entry is the sign of t for the sheet desired. If the two sheet cone is desired, this entry should be omitted. The one sheet cone is useful since it does not require an ambiguity surface to distinguish between the two sheets.

The use of the SQ and GQ surfaces is determined by the orientation of the major axis. One should never use a more complicated specification

(i.e., a GQ instead of an SQ) unless it is required since it will unnecessarily slow down the calculations.

A torus is a fourth degree equation given by

$$(x^2+z^2+py^2-2pyy+B)^2 = A(x^2+z^2),$$

where $p = c^2/b^2$
$B = z^2-c^2+py^2$
$A = 4z^2.$

However, the user only needs to specify the equation of an ellipse (or circle) which is to be revolved about the y-axis. A torus for which c>z is called a degenerate torus since the surface actually closes in on itself. For such a torus, the user must specify a plus or minus 1. A plus one entry implies that the user wishes to use the "outside" surface; a minus one, the "inside" surface. If a torus is degenerate and a ±1 is not entered, it will be assumed that the user wishes to use the outside (+1) surface.

As with cell cards, MCNP assigns the surfaces monotonically increasing integer values as the surface cards are read. These integers are referred to as program surface numbers in contrast to the problem surface numbers designated on the cell and surface cards. The maximum number of surfaces allowed is JMAX=175. The maximum number of surface coefficients allowed on all surface cards is 5*JMAX.

Table 2.1 MCNP Surface Cards

Mnemonic	Type	Description	Equation	Card Entries
P	Plane	General	$Ax + By + Cz - D = 0$	A, B, C, D
PX		Normal to X-axis	$x - D = 0$	D
PY		Normal to Y-axis	$y - D = 0$	D
PZ		Normal to Z-axis	$z - D = 0$	D
SO	Sphere	Centered at Origin	$x^2+y^2+z^2-R^2=0$	R
S		General	$(x-\bar{x})^2 + (y-\bar{y})^2 + (z-\bar{z})^2-R^2=0$	$\bar{x}$, $\bar{y}$, $\bar{z}$, R
SX		Centered on X-axis	$(x-\bar{x})^2 + y^2 + z^2 - R^2 = 0$	$\bar{x}$, R
SY		Centered on Y-axis	$x^2 + (y-\bar{y})^2 + z^2 - R^2 = 0$	$\bar{y}$, R
SZ		Centered on Z-axis	$x^2 + y^2 + (z-\bar{z})^2 - R^2 = 0$	$\bar{z}$, R
C/X	Cylinder	Parallel to X-axis	$(y-\bar{y})^2 + (z-\bar{z})^2 - R^2 = 0$	$\bar{y}$, $\bar{z}$, R
C/Y		Parallel to Y-axis	$(x-\bar{x})^2 + (z-\bar{z})^2 - R^2 = 0$	$\bar{x}$, $\bar{z}$, R
C/Z		Parallel to Z-axis	$(x-\bar{x})^2 + (y-\bar{y})^2 - R^2 = 0$	$\bar{x}$, $\bar{y}$, R
CX		On X-axis	$y^2 + z^2 - R^2 = 0$	R
CY		On Y-axis	$x^2 + z^2 - R^2 = 0$	R
CZ		On Z-axis	$x^2 + y^2 - R^2 = 0$	R
K/X	Cone	Parallel to X-axis	$\sqrt{(y-\bar{y})^2 + (z-\bar{z})^2} - t(x-\bar{x}) = 0$	$\bar{x},\bar{y},\bar{z},t^2,\pm1$
K/Y		Parallel to Y-axis	$\sqrt{(x-\bar{x})^2 + (z-\bar{z})^2} - t(y-\bar{y}) = 0$	$\bar{x},\bar{y},\bar{z},t^2,\pm1$
K/Z		Parallel to Z-axis	$\sqrt{(x-\bar{x})^2 + (y-\bar{y})^2} - t(z-\bar{z}) = 0$	$\bar{x},\bar{y},\bar{z},t^2,\pm1$
KX		On X-axis	$\sqrt{y^2 + z^2} - t(x-\bar{x}) = 0$	$\bar{x}$, t^2, ±1
KY		On Y-axis	$\sqrt{x^2 + z^2} - t(y-\bar{y}) = 0$	$\bar{y}$, t^2, ±1
KZ		On Z-axis	$\sqrt{x^2 + y^2} - t(z-\bar{z}) = 0$	$\bar{z}$, t^2, ±1
			for 1 sheet cone	±1 used only
SQ	Ellipsoid Hyperboloid Paraboloid	Major axis parallel to X, Y, or Z-axis	$A(x-\bar{x})^2 + B(y-\bar{y})^2 + C(z-\bar{z})^2$ $+ 2D(x-\bar{x}) + 2E(y-\bar{y})$ $+ 2F(z-\bar{z}) + G = 0$	A, B, C, D, E, F, G, $\bar{x}$, $\bar{y}$, $\bar{z}$
GQ	Cylinder Cone Ellipsoid Hyperboloid Paraboloid	Major axis is not parallel to X, Y, or Z-axis	$Ax^2 + By^2 + Cz^2 + Dxy + Eyz$ $+ Fzx + Gx + Hy + Jz + K = 0$	A, B, C, D, E, F, G, H, J, K
QD	Torus	Elliptical (or Circular) with Y-axis as major axis	Equation of ellipse to be revolved about Y-axis $(y-\bar{y})^2/b^2 + (z-\bar{z})^2/c^2 = 1$	$\bar{y}$,$\bar{z}$,b,c, $\pm$1 $\pm$1 used only when $c > \bar{z}$,

B. Surfaces Defined by Points

Surface cards X, Y, and Z may be used to describe surfaces by coordinates rather than by equation coefficients as in the previous section. The surfaces described by these cards must be symmetric about the x, y, or z axes, respectively, and they must be unique, real, and continuous.

The entries on these three cards are one to three coordinate pairs, each pair defining a geometrical point on the surface. On the Y card, for example, the entries may be

$$Y \quad y_1 \; r_1 \quad y_2 \; r_2$$

where $r_i = SQRT(x_i^2 + z_i^2)$.

If one coordinate pair is used, then a plane (PX, PY, or PZ) is defined.

If two coordinate pairs are used, then a linear surface (PX, PY, PZ, CX, CY, CZ, KX, KY, or KZ) is defined.

If three coordinate pairs are used, then a quadratic surface (PX, PY, PZ, SO, SX, SY, SZ, CX, CY, CZ, KX, KY, KZ, or SQ) is defined.

The conditions that a surface be unique, real, continuous, and axisymmetric must all be met, or MCNP will reject the coordinate pairs and print out a warning. Some examples may be helpful:

Example 1: X 7 5 3 2 4 3

describes a surface symmetric about the x-axis which passes through the three (x,r) points (7,5), (3,2), and (4,3). This surface is equivalent to, and in MCNP is converted to,

SQ .083333333 1 1 0 0 0 68.52083 -26.5 0 0

Example 2: Y 1 2 1 3 3 4

describes two parallel planes at Y=1 and Y=3 and is rejected because the condition of a single, continuous surface is not met.

Example 3: Z 1 3 2 4 3 7

is rejected because the coordinates are on two different branches of the SQ hyperbola

$$x^2 + y^2 - 13x^2 + 32z - 28 = 0.$$

However, the surface

```
Z       2 4 3 7 4 10.3923
```

which has the same surface equation as above is accepted because all coordinates lie on a single surface, namely the right branch of the hyperbola.

IV. DATA CARDS

All input cards to MCNP other than cell and surface cards are entered after the second blank card delimiter. These cards fall into the following categories:

(A) Mode
(B) Cell specification
(C) Source specification
(D) Tally specification
(E) Material specification
(F) Energy and thermal treatment specification
(G) Problem cutoffs
(H) Running parameters
(I) User data arrays.

These card categories are described below. Only cards of categories G and H (problem cutoffs and running parameters) are allowed in a continue-run input file.

An important concept to consider while using these cards (and also the source cards) is that of particle weight. As far as Monte Carlo is concerned, one particle is equivalent to two particles if the two particles have the same properties (position, energy, etc.) but each has a weight (or importance) of half the original particle. In other words a particle of weight w may be replaced by any number k identical particles of weight

$w_1,...,w_k$ where $w_1+...+w_k = w$. This is the basis for cell importances and surfaces where the games of splitting and Russian roulette are played in order to keep particle population built up to provide a good sample.

A. Mode (MODE) Card

The MCNP code can be run in three different modes:

Mode 0 – Neutron transport only
1 – Neutron and photon transport
2 – Photon transport only

The MODE card consists of the mnemonic MODE in columns 1–5 and either a 0, 1, or 2 in columns 6–72 depending on which mode is being used. If the MODE card is omitted, a mode of zero is assumed.

Gamma production cross sections do not exist for all nuclides, and if they are not available for a Mode 1 problem, XACT will print out warning messages. If an isotope has gamma production cross sections, the letters *GPXS* are found with the isotope in the Appendix F cross-section list.

B. Cell Specification Cards

The following cards contain information specified by program cell number:

Mnemonic (Columns 1–5)	Card Type
IN	Importances, Neutron
IP	Importances, Photon
VOL	Cell volumes
PWT	Photon weights
EXTYN	Exponential Transform, Neutron
EXTYP	Exponential Transform, Photon
FCN	Forced collision, Neutron
FCP	Forced collision, Photon
PDETN	Detector Contribution, Neutron
PDETP	Detector Contribution, Photon

Since there are no ordering restrictions on cell cards, the n^{th} entry

on a cell specification card must be the value assigned to program cell number n. The n^{th} entry does not necessarily correspond to problem cell number n as entered in columns 1–5 of the cell cards. The number of entries on a cell specification card must always equal the number of cells in the problem (thus the number of entries is also limited to AMAX=175).

1. Importance (IN and IP) Cards

The importance of a cell serves two purposes:

(a) it is used to terminate the particle's history if the importance is zero, and
(b) the importances are used for geometry splitting and Russian roulette as described in LA-4751.

Because the user may wish to split photons differently than neutrons in a neutron–photon problem, two sets of importances can be used (IN for neutrons, IP for photons). If an IP card is not present, the values on the IN card will be used for photon transport.

Importances and splitting surfaces are commonly used together in problems where normally the number of particles drops to a small size. To insure reliable sampling, it is desired to keep the number of particles throughout a geometry about the same as the number of source particles. A rule of thumb that will help keep the sample population about constant is to have a splitting surface about every mean free path and to then split two for one. See *Tracks Entering* on page 44.

2. Cell Volume (VOL) Card

The volume card is used only if a Tally 4, 6, 7, 14, or 16 is being used. The volumes of each cell are entered in units of cm^3. If a volume card is not used or if a volume is not entered for a cell, a volume of 1 cm^3 is used.

The volumes of cells can be calculated within the MCNP code by using the proper execution line option (see Appendix A), but the cells of interest must be symmetric about the y–axis. The MCNP code will then calculate the volumes which are set to 1. In this way, the user can specify some of the volumes himself and enter 1 for those he wishes MCNP to calculate. The volume calculator prints out the masses and surface

intersections as well as the volumes for each cell. Irregular volumes may also be calculated by using the ray tracing technique of Appendix E.

3. Photon Weight (PWT) Card

The PWT card is used only for combined neutron–photon problems. For each cell a minimum photon weight, $W_m(IA)$ can be specified. In order to turn off photon production in a cell a value of −1.0E6 should be entered. If any other negative number is entered, the minimum weight used is given by

$$|W_m(IA)|*W_s$$

where W_s is the weight of the source neutron that eventually led to this photon production.

By setting all the $W_m(IA)$ to some negative constant, the number of photons being created will be roughly proportional to the biased collision rate of neutrons. It is recommended for most applications that a constant of such magnitude be used to produce from one to four photons per source neutron. If the PWT card is omitted all $W_m(IA) = -1$. This default should be adequate for most problems.

4. Exponential Transform (EXTYN and EXTYP) Cards

The exponential transformation may be useful for problems where the material thickness is so great (many mean free paths) that the probability of a particle getting through the material is very small.

The entries on these cards are the q(IA) used for the exponential transformation in the Y direction as described in LA–4751. If these cards (EXTYN for neutrons, EXTYP for photons) are omitted, the q(IA) are set to zero which is the case for non–biased transport. Care must be used in choosing the value(s) of q(IA). A good rule of thumb is to choose q(IA) so that the effective material thickness is on the order of one mean free path.

If an EXTYN card is used in a neutron–photon problem but an EXTYP is not, only the neutrons will be biased.

5. Forced Collision (FCN and FCP) Cards

For cells that are almost transparent (a fraction of a mean free

path thick) to a particle but where it is desired to have the particles interact in these cells, forcing collisions may be useful.

The number of forced collisions desired for each cell may be entered on this card (FCN for neutrons, FCP for photons). A value of zero turns off forced collisions for a cell and is used as the default in case an FC(N or P) card is not used.

If an FCN card is used in a neutron-photon problem but an FCP card is not, only neutrons will have forced collisions.

The weight-cutoff game is bypassed (in subroutines COLLIDN and COLLIDP) for forced-collision cells.

6. Point and Ring Detector Contribution (PDETN and PDETP) Cards

The entries on these cards are the probabilities, P_I, of contributing to the detectors for neutron (PDETN) and photon (PDETP) transport.

At each collision in cell I, the detector tallies are made with probability P_I ($0 \le P_I \le 1$). The tally is then increased by the factor $1/P_I$ to obtain unbiased results for all cells except those where $P_I=0$. This enables the user to increase the running speed by setting $P_I<1$ for cells many mean free paths from the detectors. It also enables the selective suppressing of contributions from cells by setting the P_I's to zero.

If these cards are omitted, all $P_I=1$. If a PDETN card is present in a neutron-photon problem but a PDETP card is not, all $P_I=1$ for photon transport.

C. Source Specification

The following cards specify information for the neutron or photon source:

Mnemonic (Columns 1-5)	Card Type
SRCn (n=1,2,3,blank)	Source Type
SBIAS	Source Energy Bias
SPROB	Source Energy Probabilities
SERG	Source Energies

The specification of a source particle consists of the following information:

(1) geometry location
(2) angular description
(3) energy
(4) time
(5) particle weight

To further complicate matters, probability distributions can exist for any of the above variables. Because of the infinite number of possible sources, the MCNP code allows the user to input his own subroutine SOURCE by using an SRC card or he can pick from three standard sources by using an SRCn card. The energy distribution for the standard sources is specified by using the SBIAS, SPROB, and SERG cards as described below.

1. Source Energy (SERG) and Energy Probability (SPROB) Cards

The SERG and SPROB cards together give the distribution function of the energy spectrum for the sources. The first entry on the SERG card should be the minimum particle energy from the source, followed by the energy entries in order of increasing magnitude through the maximum allowable energy. The entries on the SPROB card are, in one case, the cumulative probabilities that a source particle has an energy less than or equal to the corresponding entry on the SERG card. The first entry on the SPROB card must always be 0 and the last entry must be 1 for a cumulative distribution.

However, if the source probability distribution is derived from data giving the number of particles started in each energy group, this data can be entered directly onto the SPROB card. The first entry is again 0, followed by the input for each energy group up through the highest energy group. The code will process these entries to form the corresponding probability distribution. The code distinguishes between the two modes of entry on the SPROB card by examining the last entry. If this is 1, it assumes that a cumulative probability distribution was read in; otherwise, it processes the data to form the distribution. Therefore, the user should be certain that a 1 is not used for the last entry if the distribution is not cumulative.

The maximum number of entries on each of the SERG and SPROB cards is 51, and the number of entries on each card must be the same.

Two examples of these cards may be instructive. For a monoenergetic source energy of 14 MeV, the two cards would be

```
SERG     14    14
SPROB     0     1
```

For a linear energy distribution from 0 to 14 MeV, the cards become:

```
SERG      0    14
SPROB     0     1
```

2. Source Energy Bias (SBIAS) Card

The entries on the SBIAS card are used to bias the energy distribution of the source. These entries are called track fractions. A track fraction is the fraction of particle histories (regardless of the weights attached to these histories) or "tracks" started in a given energy interval. For example, it is possible to start more tracks at high energies in a problem and correct the distribution by lowering the weights assigned to these tracks. By altering the weights, this technique conserves the total weight of neutrons started in an energy interval.

The first entry on the SBIAS card must be a 0. This is followed by an entry proportional to the number of tracks to be started in the lowest energy group, the entries continuing in the same way, one for each energy group, through the highest group defined for the source. Like the SPROB card this card can also contain a cumulative probability distribution with the last entry being 1.

The code normalizes the entries on the SBIAS and SPROB cards and then divides the fraction of actual source particles in an energy bin by the fraction of tracks started (the "fictitious source") in that bin to obtain the weight assigned to particles in that bin. These weights are then stored in the SBIAS data block. The cumulative probability distribution for the fictitious source is then calculated and stored in the SPROB data block.

Like the SERG and SPROB cards, the maximum number of entries is 51, and the number must be the same as on the SERG and SPROB cards.

3. Source Type (SRCn) Card

The type of source used is determined by the n on the SRCn card as

follows:

Mnemonic (Columns 1-5)	Source Type
SRC	Subroutine SOURCE is supplied
SRC1	Biased Point Source
SRC2	Outward Cosine Distribution on Biased Spherical Surface
SRC3	Inward Biased Cosine Distribution on Spherical Surface

The time distribution for all standard sources assumes that all particles are emitted at time TME=0. The energy distributions are input through the SBIAS, SPROB, and SERG cards as described. The geometric locations and angular descriptions are input on the SRCn cards.

Source Type	Entries and Description
SRC1	x y z I W p ν

This card specifies an isotropic point source in problem cell I located at the point (x,y,z) with an average particle weight W. The direction cosine with the y-axis, v, is sampled uniformly within the cone $\nu<v<1$ with probability p and within $-1<v<\nu$ with the complementary probability (1-p). The weights assigned are $W(1-\nu)/(2p)$ and $W(1+\nu)/\{2(1-p)\}$, respectively. The sampling of the direction cosines with the x and z axis (u and w) is not biased. The default values set by MCNP are x=y=z=0, I=1, W=1, p=.5, ν=0.

SRC2 J W P ν

This card specifies an outward cosine distribution on the spherical problem surface J directed into problem cell I with an average particle weight W. The surface J has to

be either type SY or SO and there must be only one cell inside the surface. The point on the sphere is obtained by sampling (u,v,w) exactly as described for the SRC1 card using ν and p. The weight is set as for the SRC1 card and the particle is advanced along the (u,v,w) vector from the center of the sphere to the surface. The actual (u,v,w) direction is then selected by sampling the cosine of the angle with respect to the outward normal from a cosine distribution {cos(φ)=sqrt(r) with r a random number} and determining (u,v,w) by sampling a random azimuthal angle between 0 and 2π on the cone defined by cos(φ). The default values are J=1, W=1, p=.5, and ν=0.

SRC3 J M P ν

This card specifies an inward cosine distribution on the spherical problem surface J with weight multiplier M. The particles are biased inward from the spherical surface. For a vacuum this source provides a uniform isotropic flux inside the spherical surface. The default values are J=1, M=1, p=.5, and ν=0.

A word of warning considering biasing. One should never bias a source completely. For example, if p=1 on a SRC1 card, all particles would be started within the cone ν<v<1. Such a source is no longer an isotropic point source. If biasing is not desired for any of the above sources, the last two entries should be deleted.

4. Subroutines SOURCE and SRCDX

If an SRC card is used, it signals the MCNP code that the user wishes to describe his source by supplying a FORTRAN subroutine SOURCE. See Appendix A, page 56, for requirements of this subroutine.

D. Tally Specification

The tally cards are used to specify what type of information the user wants to gain from the Monte Carlo calculation, i.e., current across a

surface, flux at a point, heating in a region, etc. This information is requested by using the following cards:

Mnemonic (Columns 1-5)	Card Type
Fn (n=1,...,7,11,...,16)	Tally Type
Fna (n=5,15;a=x,y, or z)	Ring Detector Tally
En (n=1,...,7,11,...,16)	Tally Energies
Tn (n=1,...,7,11,...,16)	Tally Times
Cn (n=1,11)	Cosines - Tallies 1 and 11
IFLUX	Cell Numbers - Tallies 3 and 13
VECT	Reference Vector - Tallies 3 and 13
CYL	Z-axis - Tallies 3 and 13
CPHI	Cos φ - Tallies 3 and 13
AREA	Areas - Tallies 3 and 13
CFGn (n=4,6,7,14,16)	Cell Flagging
SFGn (n=4,6,7,14,16)	Surface Flagging

Much of the information entered on these cards is used to describe tally "bins." A tally bin is the smallest unit of a tally and is for a specific increment: (T_1<Time≤T_2), (E_1<Energy≤E_2), etc. The last part of this section (page 28) addresses the problem of determining how many tally bins one is using.

The results of all tallies are normalized to be per source particle. If reflecting planes are used, the user may have to further normalize the tallies himself (can be done by setting the weight of the source particles or by using the RESn card).

1. Tally (Fn) Cards

There are 8 neutron tallies (n=1,...,7 and 5a) and 7 photon tallies (n=11,...,16 and 15a) available in the MCNP code. All are normalized to be per source particle.

Mnemonic (Columns 1-5)	Tally Description
F1 and F11	Current across any designated subset of the bounding surfaces in the problem. (units=particles)

F2 and F12	Flux integrated over any designated subset of the bounding surfaces in the problem. To arrive at an average flux on the area, this tally should be divided by the surface area. (units=particles)
F3 and F13	Heating in a material at a surface as a function of solid angle bins on the surface. (units=MeV/gm).
F4 and F14	Track length per unit volume, or average flux, for any specified subset of cells. (units=$1/cm^2$)
F5 and F15	Flux at a designated set of points in space. (units=$1/cm^2$)
F5a and F15a	Average flux on a designated set of spatial rings symmetric about the a = x, y, or z axis. (units=$1/cm^2$)
F6 and F16	Track length estimate of charged particle heating including fission heating (F6), or photon heating (F16) for any specified subset of cells. (units=MeV/gm)
F7	Track length estimate of energy deposition due to fissions in any specified subset of cells. This is a subset of Tally F6. (units=MeV/gm).

Note the units of flux for the F2/12 and detector tallies. In all cases these tallies are a function of time and energy. The F2/12 tally, however, is integrated over surface area. If the flux is uniform over a surface, then the results of the F2/12 tally divided by the surface area (in cm^2) should be equal to the results of a detector on that surface. The current in the F1/11 tally is a vector quantity and is equal to the F2/12

tally if the F1/11 score has been divided by the cosine of the angle that the trajectory makes with the surface crossed. In the case of a point isotropic source, for example, at the center of a void (i.e., no collisions) sphere, the F1/11 and F2/12 tallies at the surface of the sphere should be equal.

All of the above tallies require En and Tn cards (see description below). Tallies 1 and 11 require a Cn card and tallies 3 and 13 require CYL, CPHI, VECT, IFLUX, and AREA cards. The cell tallies 4, 6, 7, 14, and 16 may need a VOL card (see Page 14).

The entries on the F1, F2, F3, F11, F12, and F13 cards specify the problem surface numbers for which the tallies are performed. If both F3 and F13 cards are used the surface entries on each must be identical. The entries on the F4, F6, F7, F14, and F16 cards specify the problem cell numbers for which the tallies are performed.

The F5 and F15 cards contain sets of ordered quadruples (x,y,z,R_o), one quadruple for each detector point where (x,y,z) designates the location of the point in space, and R_o is the radius of a fictitious sphere with center at (x,y,z) (see description of point detector in LA-4751). The F5a and F15a cards (where "a" represents the axis of symmetry - either x, y, or z) contain sets of ordered triples (a_o,r,R_o), one triple for each detector ring where (a_o,r) designates a detector ring in space. The a_o entry is the point of interception of the plane of the ring with the axis of symmetry and r is the ring radius. R_o has the same meaning as with the F5 and F15 tallies. Detectors should not be used with reflecting surfaces. The use of the PDETN and PDETP cards (see page 16) or the DRECT card (see page 27) with tallies 5, 5a, 15, and 15a is optional.

When collisions can occur arbitrarily close to the detector, care should be used in the choice of R_o. The sphere defined by R_o should be large enough to enclose a reasonable number of collisions; however, approximations needed for the detector scheme become worse for large R_o. Generally, R_o is chosen to be some fraction of a mean free patch, perhaps a fraction on the order of 1/8 to 1/2. For a detector in a region with few collisions, such as air, R_o may be set to zero.

The total number of detectors is restricted to 10. However, their use should be minimized since they can significantly add to computer time. Consider using the track length estimator (F4,14 tallies) in place of detectors.

The ring detector tallies are modified point detector tallies. They allow enhanced computing efficiency for problems symmetric about a major axis. For example, if your geometry and spatial source distribution

are symmetric (or at least the asymmetry is insignificant) about a major axis and you desire a point detector flux estimate at radius r from the symmetry axis, the ring detector will yield a smaller relative error than the point detector for a given number of histories. Ring detectors may also be used for non-axisymmetric problems where one is interested in the average flux around a ring about a major axis.

There are some cases where use of the ring detector will not be advantageous. For example, if r=0 (detector on axis of symmetry) or if r is very large compared to the dimensions of the scattering media (such that the detector sees essentially a point source in a vacuum), the ring detector is less efficient than a point detector.

Tallies 1, 2, 4, 5, 5a, 11, 12, 14, 15, and 15a are normally weight tallies; however, if the F card is flagged with an asterisk (i.e., *F1), energy will be tallied. The units will then be MeV, MeV, MeV/cm^3, MeV/cm^2, MeV/cm^2, MeV, MeV, MeV/cm^3, MeV/cm^2, and MeV/cm^2. respectively. The asterisks flagging can also be used on tallies F6, F7, and F16 to change the units from MeV/gm to jerks/gm.

The sum of entries on all Fn cards is limited to JFM=150. The F3 and F13 cards are limited to 5 entries.

2. Tally Energy (En) Card

The entries in MeV on this card are the upper bounds of the energy bins for tally n. The entries must be entered in the order of increasing magnitude. If a particle has an energy greater than the last entry, it will not be tallied.

3. Tally Time (Tn) Card

The entries in shakes on this card are the upper bounds of the time bins and like the En card must be entered in order of increasing magnitude. If a particle has a time greater than the last entry on the Tn card, it will not be tallied. The last time bin entry should always be less than or equal to the time cutoff (see page 35).

If the time bins are entered greater than the time cutoff, the first bin limit over the cutoff will be lowered to the cutoff. All other bins will remain the same.

4. Cosine (Cn) Card (Tallies 1 and 11 only)

The entries on this card are the angular limits of the cosine bins used for tallies 1 and 11. The angular limits are defined with respect to the normal to the surface at the particle point of entry. The normal to the surface is always in the direction of a cell that has positive sense with respect to that surface.

The card entries are the upper bounds of the cosine bins where the order of entry starts with the angle at 180° to the normal and continues around to the normal (cos=1). Thus, to tally currents within the angular limits 180° to 150°, 150° to 120°, 120° to 90°, 90° to 60°, 60° to 30°, and 30° to 0° with respect to the normal, the entries on the Cn card would be -.866, -.5, 0., .5, .866, 1.0. The last entry must always be 1. A lower bound of -1 is set in the code and is not entered on the card.

As an example of the relation between a surface normal and sense for the C1 card, consider a source at the origin of a coordinate system and a plane intersecting the +y axis. An entry of 0 and 1 on the C1 card will tally all source particles transmitted through the plane in the 0 to 1 angular bin (0° to 90°) and all particles reflected back across the plane in the -1 to 0 angular bin (90° to 180°). A plane intersecting the -y axis will result in a tally of all source particles transmitted through the second plane in the -1 to 0 bin (90° to 180°) and all particles reflected back across the plane in the 0 to 1 bin (0° to 90°).

5. Cell Numbers (IFLUX) Card (Tallies 3 and 13 Only)

The number of entries on this card should equal the number of entries on the F3 or F13 card. The entries are the problem cell numbers used to designate materials for the heating tallies of the corresponding surface on the F3 or F13 card. The material of a cell is used with the density of the cell. Two or more surfaces on the F3 or F13 card may have identical values with the corresponding cell numbers on the IFLUX card not identical. This enables one to calculate heating at the surface for cell materials on either side of the surface. The entries on this card are used for both neutron and photon tallies. The limit of 5 entries that applies to the F3 and F13 cards also applies to this card.

6. Reference Vector (VECT) Card (Tallies 3 and 13 Only)

Nine entries are required on the VECT card:

v_1 v_2 v_3 v_4 v_5 v_6 v_7 v_8 v_9

The entries (v_1, v_2, v_3) locate the "origin" of the "z-axis" used to describe surface segments. The entries on the CYL card are "z distances" with respect to this origin. The entries (v_4, v_5, v_6) specify the direction cosines of the z-axis. The entries (v_7, v_8, v_9) specify a direction orthogonal to (v_4, v_5, v_6); i.e., $v_4v_7 + v_5v_8 + v_6v_9 = 0$. The azimuthal angle φ in a cylindrical coordinate system is referenced with respect to the (v_7, v_8, v_9) vector. The entries on this card are used for both tallies 3 and 13.

7. Z-axis (CYL) Card (Tallies 3 and 13 only)

The entries on this card are the "z-axis" boundaries and are entered in order of increasing magnitude (include lower bound as first entry). The product of the number of entries on CYL and CPHI cards should be no more than 80. The entries on this card are used for both tallies 3 and 13.

8. Cos(φ) (CPHI) Card (Tallies 3 and 13 only)

The entries on this card are the upper boundaries of the cos(φ) bins and are entered in order of increasing magnitude, similar to the Cn card entries.

The product of the number of entries on the CYL and CPHI cards should be no more than 80. The entries on this card are used for both tallies 3 and 13.

9. Areas (AREA) Card (Tallies 3 and 13 only)

The entries on the AREA card are the areas of the surface segments used for the bins of tallies 3 and 13. The number of entries, k, is given by k=g*m*n where g, m, and n are the number of entries on the CYL, CPHI, and F3 (or F13) cards respectively. Actually, g is one less than the number of entries since an additional lower bound is included on the CYL card.

The total heating over a surface segment on the N^{th} surface entry on the F3 or F13 card (located between the L^{th} and $(L+1)^{th}$ entries on the CYL card and between the M^{th} and $(M+1)^{th}$ entries on the CPHI card) is divided by the K^{th} entry on the AREA card where

$$K = (N-1)*g*m+(L-1)*m+M$$

to obtain heating in units of MeV/g.

The default value for these entries is 1.

10. Cell Flagging (CFGn) Card (Tallies 4, 6, 7, 14, and 16 only)

The entries on the CFGn card are problem cell numbers whose contributions are to be "flagged" for the n^{th} tally (n=4,6,7,14, and 16).

Example:

```
F4      6      10      13
CFG4    3
```

The flag is turned on when a neutron enters cell 3. The print of Tally 4 is doubled. The first print is the total track length tally in cells 6, 10, and 13. The second print is the tally in these cells for only those neutrons that have passed through cell 3 at some time before passing through cell 6, 10, or 13.

For a Mode 1 problem, if you specify both F4 and F14 and/or both F6 and F16 tallies and want flagging, using either CFG4 or CFG14 or using either CFG6 or CFG16 will flag the cells for both neutrons and photons. You cannot flag one set of cells for neutrons and another set for photons. Finally, in Mode 1 the flagged neutron tallies are those caused by neutrons passing through the flagged cell, but the flagged photon tallies can be caused by either a photon passing through a flagged cell or a neutron passing through a flagged cell and then leading to a photon which is tallied.

11. Surface Flagging (SFGn) Card (Tallies 4, 6, 7, 14, and 16 only)

The entries on this card are like those on the CFGn card except that they are problem surface numbers. Thus a second Tally print is given for only those particles that have crossed some surface specified on the SFGn card.

12. Direct Contribution from Point Detector (DRECT) Card

This card has no data entry associated with it but is used as an

option flag instead. If this card is present, the printed output for Tallies 5, 5a, 15, and 15a is doubled. The doubled output contains: (1) the contributions from both the source and collisions (this is the usual output) and (2) the contribution from source particles only (the uncollided flux).

13. Response Function (RESn) Card

This card can be used with any tally (specified by n) to calculate a response function rather than the usual current, flux, etc. There should be one entry for each energy entry on the corresponding En card. When a tally is being recorded within a certain energy bin, the entry on the RESn card corresponding to that bin is multiplied by the regular contribution. Thus one can tally any reaction rate, dose rate, etc., he desires provided he knows the specific response function.

14. Tally Storage Limitations

Below is given the number of words of storage used by each Tally:

$$W_{1,11} = (N+1)*P*(M+1)*D$$

$$W_{2,12} = (N+1)*(M+1)*D$$

$$W_{3,13} = (N+1)*P*A*(M+1)*D$$

$$W_{4,6,7,14,16} = (N+1)*(M+1)*D*d$$

$$W_{5,15} = D*(N+1)*(M+1)$$

$$W_{5a,15a} = D*(N+1)*(M+1)$$

where W_n = no. of words used by Tally n
N = no. of entries on E card
M = no. of entries on T card
P = no. of cosine bins specified on Cn card or CPHI card
D = no. of entries on F card (No. of detectors for n = 5)
A = no. of entries on CYL card
d = 1 if no flagged contribution, = 2 if CFGn , SFGn, or DRECT cards used.

The total number of words used by all tallies is given by

$$W_{tot} = \sum_{n=1}^{16} W_n * d_n \qquad N \neq 8,9,10$$

where $d_n = 1$ if tally n is used
$d_n = 0$ if not.

The number of words available for tally bins depends on the number of words of cross sections required and other data in the MCNP 57650_{10} word FL array. How the FL block is apportioned is printed in the MCNP output at the end of the XACT overlay output.

E. Materials Specification

1. Material (Mn) Card

The following card is used to specify materials:

Mnemonic (Columns 1-5)	Card Type
Mn (n=1,...,no. of mat'l.)	Material Card

The n on a material card is the cell material number (see page 6) to which this material description corresponds.

The entries on the material card should consist of the identifying number of a constituent nuclide followed by the atomic fraction (or weight fraction if entered as a negative number) of that element, the number of a second constituent element followed by its atomic fraction, etc., running through all the elements needed to define the material. These two entries are discussed in further detail below.

Nuclide Identifer Number, ZAID. This number is used to identify a nuclide to any degree desired by the user. The form of the number is

ZZZAAA.nn

where ZZZ is the atomic number of the nuclide,
AAA is the atomic mass number of the nuclide, and

nn is the neutron cross-section set identifier.

If a photon-only problem is being run the AAA can be set to 000 and the nn omitted. If AAA=000, the ZAID represents an element; not an isotope. Thus, ZAID=74182.01 represents the isotope 74-W-182, and ZAID=74000.01 represents the element tungsten. Photon cross sections are specified exactly like the neutron cross sections, but for photons ZZZAAA.nn is equivalent to ZZZAAA. There is no distinction between isotope and element for photons.

There are currently six different cross-section files used by MCNP:

(1) Neutron Recommended (RMCCS),
(2) Neutron Alternate (AMCCS),
(3) Neutron Extraneous (XMCCS),
(4) Neutron Ubangi (UMCCS),
(5) Neutron Discrete Reaction (DRMCCS), and
(6) Photon (MCPLIB).

The contents of these files are periodically changed and are listed in Appendix F. If nn is not specified the RMCCS file will be searched for any nuclide with the same ZZZAAA. If it is not found on this file the AMCCS file is searched and so forth until all appropriate files are searched. If the user wishes to use a specific nn he must look at the contents of these files (Appendix F) and decide which cross-section set he wishes to use. Users should be aware that when using the ZZZAAA. specification in runs which use different versions of the cross-section files, their cross-section sets may vary. It is more consistent to always use the ZZZAAA.nn specification.

The amount of space available for cross-section storage is at least $262{,}140_{10}$ words. This should be considered your upper limit for planning purposes; however, depending upon how many tallies you have and a few other things, you may get a few thousand more words from the FL array. The storage required for each isotope is listed in Table F.1 on page 124.

Nuclide Fraction. The nuclide fractions may be normalized to 1 or left unnormalized. For instance if the material is H_2O the fractions can be entered as (.667 and .333) or as (2 and 1) for (H and O) respectively. If the fractions are entered with negative signs they are assumed to be weight fractions; otherwise, atomic fractions. Weight fractions and atom fractions cannot be mixed on the same Mn card.

The total number of "nuclide-fraction" entries allowed is MEMAX=120. The total number of different nuclides allowed is EMAX=40.

2. Discrete Reaction Cross Section (DRXS) Card

Any nuclide listed on the optional DRXS card is given a discrete treatment (pseudo-multigroup) instead of the regular fully continuous-energy cross-section treatment if the necessary discrete data are available (check the list in Appendix F). Nuclides are listed on the DRXS card by ZAID number. Any number of entries is allowed, but if there are no entries on the DRXS card, discrete cross sections will be used for every nuclide if available.

This 240-group discrete treatment applies to neutron reaction cross sections only and is therefore not called a multigroup treatment. Secondary angular and energy distribution data are still given by the regular continuous-energy treatment.

Use of the discrete treatment reduces average computer storage requirements per isotope by a factor of two and for some isotopes by a factor of seven. Accuracy is degraded only when resonance self-shielding is important.

When a threshold reaction is encountered and the particular energy is less than that of the threshold but still in the threshold group the reaction is abandoned. To conserve neutrons the reaction selection process is repeated. Experience has shown that this rarely occurs, but how many times it does happen is indicated in the problem output. If the reaction is rejected 100 times the neutron is relegated to a lost particle and a debug message "inelastic reaction not found properly" is printed in the output file.

3. Total Fission $\bar{\nu}$ (TOTNU) Card

Unless a TOTNU card is used, prompt $\bar{\nu}$ is used for all fissionable nuclides. If a TOTNU card is present then total $\bar{\nu}$ will be used for those fissionable nuclides for which total $\bar{\nu}$ values are available. The symbol Ψ in the nuclide list of Appendix F indicated which fissionable nuclides have only prompt $\bar{\nu}$ data available. There are no entries on the TOTNU card. The MCNP neutron cross-section summary print from XACT will indicate whether prompt or total $\bar{\nu}$ was used.

F. Energy And Thermal Treatment Specification

The following cards are involved with energy aspects of MCNP:

Mnemonic (Columns 1–5)	Card Type
ERGN	Energy Card, Neutron
ERGP	Energy Card, Photon
ESPLT	Energy Splitting Card
TEMPn	Thermal Temperature Card
THTME	Thermal Times Card
TI	Thermal Isotopes Card

All energy entries on these cards are in units of MeV.

1. Neutron Energy (ERGN) Card

The neutron energy card has two entries in units of MeV:

$$E_{th} \qquad E_{max}$$

All neutrons having an energy less than E_{th} are given the thermal treatment by using the free gas model (see LA-4751) for designated thermal isotopes (see page 34) and by making all elastic collisions with other isotopes isotropic (in the lab system) with no energy loss.

This thermal cut-in is usually chosen to be a factor of 10 greater than the energy a neutron would have if it were at the temperature entered on the TEMPn cards. For example, if the temperature of a cell is given as $kT=2.5\times10^{-8}$ MeV then a neutron at that temperature would have an energy of 3/2 $kT=3.75\times10^{-8}$ MeV, and an E_{th} of 3.75×10^{-7} should be used. If E_{th} is negative, the thermal cut-in used is given by

$$|E_{th}|*Temp$$

where Temp is the thermal temperature entered on the TEMPn card for the appropriate time and cell. It is recommended that an entry of −15 be used for most problems where the thermal treat- ment is desired.

If the thermal treatment is not desired, E_{th} should be set to zero which is the default value. If $E_{th}\neq0$ then the user must supply THTME and TEMPn cards.

All neutron cross sections pertaining to energies greater than E_{max} are eliminated. This entry should be at least as large as the energy of any neutron in the problem. If a neutron is transported at an energy greater than E_{max}, the cross sections at E_{max} will be used. The purpose of this entry is to trim off unnecessary cross sections to save storage. The default (and maximum) value used for E_{max} is 100 MeV.

2. Photon Energy (ERGP) Card

This card has a single entry, EMCPF. All photons having an energy greater than EMCPF will be given the simple physics treatment of MCG as described in LA-5157-MS. All photons with an energy less than EMCPF will be treated with the more detailed physics of MCP. If this card is omitted, the detailed treatment will be used at all energies below 100 MeV.

3. Energy Splitting (ESPLT) Card

The entries on this card consist of pairs of energy splitting parameters, N_{spl} and E_{spl}, with a maximum of five pairs allowed. The E_{spl} is the energy at which neutrons are to be split, and N_{spl} is the number of particles into which a neutron will be split. For example

ESPLT 2 .1 2 .01

specifies a 2 for 1 split when the neutron energy falls below .1 MeV and another 2 for 1 split when the energy falls below .01 MeV.

Energy splitting is available for neutron transport only.

4. Thermal Temperatures (TEMPn) Cards

These cards are necessary for neutron transport only if the user wishes to use the thermal treatment as described in LA-4751 (also see energy card description, page 32) and the problem includes thermal isotopes. The thermal isotopes are hydrogen, deuterium, and isotopes specified on a thermal isotope card, TI. Thermal temperatures are entered as a function of time with a maximum of 7 time entries allowed. These times ($t_1, t_2, \ldots t_n$, $n<8$) are entered on a thermal time (THTME) card. The thermal temperatures at time t_1 are listed, cell by cell, on the TEMP1 card; the corresponding cell thermal temperatures at time t_2 are listed on the TEMP2 card, etc. A linear interpolation is used to determine the cell

thermal temperatures at times between two entries. Time values occurring before t_1, or after t_n, use the thermal temperatures at the nearest time entry. Because thermal temperature entries are required only for those cells whose material composition includes one or more thermal isotopes, all other cell entries can be set to zero.

If the user does not wish to use the thermal treatment for special isotopes (see TI card) in a given cell the TEMPn entry for that cell should be entered negative. The absolute value of this entry will then be used for hydrogen and deuterium only.

We use kT to denote the thermal temperature of a cell and use units of MeV. The following formulas can be used to provide the values of kT for temperatures in degrees kelvin, Celsius, Rankine, and Fahrenheit.

$$\begin{aligned} kT(MeV) &= 8.617\times10^{-11}T \text{ where T is in degrees k} \\ &= 8.617\times10^{-11}(T+273.15) \text{ where T is in degrees C} \\ &= 4.787\times10^{-11}T \text{ where T is in degrees R} \\ &= 4.787\times10^{-11}(T+459.67) \text{ where T is in degrees F} \end{aligned}$$

5. Thermal Times (THTME) Card

The entries on this card are the times in shakes (10^{-8} sec) at which thermal temperatures are specified on the TEMPn cards. A maximum of 7 time entries is allowed in order of increasing magnitude. For each entry on this card, a TEMPn card is required. The default for this card is a single time equal to 0.

6. Thermal Isotopes (TI) Card

The entries on this card are the nuclide identification numbers (see Mn card description) of those nuclides (up to Z=8) for which a thermal free gas treatment is desired. Numbers to the right of the decimal point are ignored so that either ZZZAAA.nn or ZZZAAA. can be entered. Note that hydrogen and deuterium will always be treated with the free gas treatment regardless of whether a TI card is used or omitted. The user should only input important isotopes since the thermal treatment is time consuming.

C. Problem Cutoffs

Particles are eliminated from the Monte Carlo transport process by the

energy, time, and weight cutoffs. The following cards can be used in an initiate-run or a continue-run input file in order to specify these cutoffs.

Mnemonic (Columns 1-5)	Card Type
CUTN	Neutron Cutoffs
CUTP	Photon Cutoffs

1. Neutron Cutoffs (CUTN) Card

Four entries exist on this card:

$$T_{con} \qquad E_{con} \qquad WCN1 \qquad WCN2$$

The time cutoff, T_{con}, is in units of shakes (10^{-8} seconds). The transport of a neutron is immediately stopped and the neutron killed if its time becomes greater than T_{con}. The default value of T_{con} is 1.0E123 shakes (1.0E115 seconds).

The energy cutoff, E_{con}, is in units of MeV. Any neutron having an energy lower than E_{con} is killed. The default value for E_{con} is 0.

If a neutron's weight (WT) falls below WCN2 times the ratio R of the source cell importance to the collision cell importance, then with probability (WT/WCN1)*R, the neutron survives and is assigned WT=WCN1*R. If negative values are entered for the weight cutoffs, the values

$$|WCN1|*W_s \text{ and } |WCN2|*W_s$$

will be used for WCN1 and WCN2, respectively, where W_s is the weight assigned to the source neutron. These negative entries are recommended for most problems. The default values for WCN1 and WCN2 are -1 and WCN1/2, respectively. See page 1 for a discussion of weight cutoffs.

If WCN1 is set to zero, capture is then treated explicitly in the analog fashion versus implicitly by reducing the neutron's weight according to the capture probability.

2. Photon Cutoffs (CUTP) Card

As with the CUTN card there are four entries on this card:

T_{cop} E_{cop} WCP1 WCP2

The time and energy cutoffs, T_{cop} and E_{cop}, are identical to those for neutron transport, T_{con} and E_{con}, except for their defaults. The default for E_{cop} is .001 MeV. The default for T_{cop} is T_{con}. If T_{con} is not specified, the default is 1.0E123 shakes.

The weight cutoffs are also the same except that (1) they are only used for energies above the simple physics (MCG) cutin EMCPF (see ERGP card description, page 33) and (2) there is no provision for explicit analog capture in the simple physics treatment. For energies below EMCPF, the cutoffs have no effect since the detailed physics (MCP) considers nothing but analog capture.

In a Mode 1 problem, the photon weight cutoffs are the same as the neutron weight cutoffs (either the default values or from the CUTN card) unless overridden on the CUTP card. Again, the photon weight cutoffs have no effect at energies below EMCPF.

H. Running Parameters

The following cards are used to control MCNP run cycles and can be used in either an initiate-run or a continue-run:

Mnemonic (Columns 1-5)	Card Type
CTME	Computer Time Cutoff
PRDMP	Print and Dump Cycle
NPS	Particle Cutoff
LOST	Lost Particle Cutoff
DBCN	Debug Information

1. Computer Time Cutoff (CTME)

The single entry on this card is the maximum amount of computer time (in minutes) to be spent in the Monte Carlo calculation. If the value entered here is greater than the amount of time that is available for the job, or if this card is omitted, the computer time cutoff used is that remaining for the job. For a continue-run this time limit is the time relative to the start of the continue-run.

2. Print and Dump Cycle (PRDMP)

The two entries on this card

NDP NDM

are the cycle limits for printing out tallies and dumping information on a run file for a continue-run in addition to printing and dumping at the end of the calculation. After every NDP particle the tallies are printed to the output file. After every NDM particles the necessary information is dumped to the run file. A negative entry on this card changes the cycle limit from particles to minutes of computer time. The default for NDP is to print only after the calculation has successfully ended. The default for NDM is to dump every fifteen minutes plus at the end of the problem.

It is recommended to print fairly frequently (every few thousand histories or so, depending upon the total number of histories expected to be run) in order to watch the variance of the variance. This will give you an indication of how reliable your results are - this is especially important when using detectors. For example, a result after 10000 histories may have an *indicated* error of less than 10%, but the variance after 20000 histories may be up to 20% - indicating the earlier result with the 10% variance is not reliable.

3. Particle Cutoff (NPS) Card

The single entry N on this card is used to terminate the Monte Carlo calculation (provided it has not already terminated due to computer time cutoff). Thus after N particles have been transported, the calculation stops. If this card is omitted, the calculation continues until the computer time cutoff is reached. In the case of a continue-run, N is the total number of particles including runs prior to the continue-run, or a negative entry means to print an output file for the last dump.

4. Lost Particle (LOST) Card

There are two entries on this card:

LOST(1) LOST(2)

The first sets the number of histories that may be lost before the job terminates. The second entry determines the number of debug prints obtained as a result of lost particles. The default for both is 10.

This card should be used cautiously: the user should know why the particles are being lost, and the number lost should be statistically insignificant out of the total sample.

5. Debug Information (DBCN)

The two entries on this card

KRNT NMBD

are used primarily for debugging problems. KRNT is the random number used for starting the transport of a particle history, and the default is zero. For example if a user has had difficulties with particle 124 and wishes to restart a calculation with particle 124 as the first particle, he should enter the random number that started particle 124 as KRNT. KRNT is entered in free field format, although the sixteen digit octal number followed by the letter B is the most common form. NMDB is used to print out information about every NMDBth particle. The information consists of: (1) the particle history number, (2) the total number of collisions, (3) the number of random numbers generated, and (4) the current random number at the beginning of the NMDBth history. This information is printed at the first of the particle history.

I. User Data Arrays

Two arrays, IDUMMY and RDUMMY, are in MCNP Common and are available for the user. Each array is dimensioned 50, and they may be filled by cards in the input file INP. IDUMMY is an integer array and RDUMMY is a real array.

1. Integer Array (IDUM) Card

The entries (up to 50) on this card fill the IDUMMY array with integer numbers. If real numbers are entered on this card, they will be truncated and converted to integers.

2. Real Array (RDUM) Card

The entries (up to 50) on this card fill the RDUMMY array with real numbers.

V. SUMMARY OF MCNP INPUT FILE

This section summarizes three frequently-used categories for determining MCNP input.

A. Input Cards

Below is given a table (Table 2.2) summarizing the various input cards and when they are required. An *R* specifies that the card is required by the specified Mode. An *O* indicates that the card is optional depending on the user's needs and depending upon the other cards that are included. An *X* indicates that this card should not be used with a specific Mode. The column labeled *Requires* lists those cards that must be included if the specified card is used. The column labeled *Required by* indicates that the card is required if any of the cards listed in this column are used.

Table 2.2 Summary of MCNP Input Cards

Card	Mode 0	Mode 1	Mode 2	Requires	Required by
ID Card	R	R	R		
Cell Cards	R	R	R		
Surface Cards	R	R	R		
Cell Specification					
IN	R	R	X		
IP	X	O	R		
TEMPn	O	O	X		ERGN
VOL	O	O	O		Fn (n=4,6,7,14,16)
PWT	X	O	X		
EXTYN	O	O	X		
EXTYP	X	O	O		
FCN	O	O	X		
FCP	X	O	O		
PDETN	O	O	X		
PDETP	X	O	O		
Source Specification					
SRCn	R	R	R		
n=1,2,3				SPROB,SERG	
n=blank				Subr. SOURCE	
SBIAS	O	O	O		
SPROB	O	O	O		SRCn
SERG	O	O	O		SRCn
Tally Specification					
Fn	R	R	R		
n=1,11				En,Tn,Cn	
n=2,5,12,15				En,Tn	
n=3,13				En,Tn,IFLUX,VECT	
n=4,6,7,14,16				CYL,CPHI,AREA En,Tn,VOL	
Enf	R	R	R		Fn
Tn	R	R	R		Fn
Cn	O	O	O		F1,F11
IFLUX	O	O	O		F3,F13
VECT	O	O	O		F3,F13
CYL	O	O	O		F3,F13
CPHI	O	O	O		F3,F13
AREA	O	O	O		F3,F13
CFLGn	O	O	O		
SFLGn	O	O	O		
RESn	O	O	O		
DRECT	O	O	O		
Materials					
Mn	R	R	R		Cell Cards
Energy					
ERGN	R	R	X		
ERGP	X	O	O		
ESPLT	O	O	X		
THTME	O	O	X		ERGN (E_{th}≠0)
TI	O	O	X		
Mode					
MODE	O	R	R		
Problem Cutoffs					
CUTN	O	O	X		
CUTP	X	O	O		
Running Parameters					
CTME	O	O	O		
PRDMP	O	O	O		
NPS	O	O	O		
DBCN	O	O	O		

B. Storage Limitations

Table 2.3 summarizes some of the more important limitations on the MCNP input.

Table 2.3
Storage Limitations

No. of Cross-section Words	262140
No. of Cell Cards	175
No. of Surface Cards	175
No. of Surface Entries on all Cell Cards	1225
No. of Cell Entries on all Cell Cards	2100
No. of Surface Coefficients on all Surface Cards	875
No. of Times for Thermal Energies	7
No. of Energy Groups on SERG,SPROB, and SBIAS Cards	50
No. of Entries on SRC card	51
No. of Detectors	10
Product of the no. of Entries on CYL and CPHI Cards	80
No. of "nuclide-fraction" Entries on All Mn Cards	120
No. of Different Nuclides	40
No. of Energy Splitting Surfaces	5
No. of Entries on all Fn Cards (n=5,15)	150
No. of Entries on F3,F13, and IFLUX Card	5
No. of Entries on IDUM Card	50
No. of Entries on RDUM Card	50
No. of "bin limits" specified on all En, Tn, Cn, CPHI, and CYL Cards	150

C. *Defaults*

Table 2.4 summarizes the default information for each input card.

Table 2.4
Input Defaults

Card	Defaults
AREA	1 cm^2
Cn	none
CFGn	none
CPHI	none
CTME	Execution line time limit used
CUTN	T_{con}=1.0E123 shakes E_{con}=0 WCN1=-1 WCN2=WCN1/2
CUTP	T_{cop}=T_{con} if specified and 1.0E123 if not E_{cop}=0.001 MeV WCP1=WCN1 WCP2=WCN2 (Weight cutoffs used only above ERGP)
CYL	none
DBCN	KRNT=0 NMBD=NPS
DRECT	none
DRXS	fully continuous energy
En	none
ERGN	E_{th}=0, E_{max}=100 MeV
ERGP	100 MeV
ESPLT	no energy splitting
EXTN	0
EXTP	0
FCN	0
FCP	0
Fn	none
IDUM	0 49r
IFLUX	none

IN	0
IP	IN card (0 if no IN card)
LOST	10
MODE	0
NPS	use CTME for limit
PDETN	1
PDETP	1
PRDMP	NDP = end of calculation
	NDM = -15 and end of calculation
PWT	-1
RDUM	0 49r
RESn	1
SBIAS	none
SERG	none
SFGn	none
SPROB	none
SRC1	x=y=z=0, I=W=1, p=.5, ν=0
SRC2	J=W=1, p=.5, ν=0
SRC3	J=M=1, p=.5, ν=0
TEMPn	0
THTME	0
TI	hydrogen and deuterium only
Tn	none
TOTNU	prompt fission $\bar{\nu}$
VECT	none
VOL	1 cm^3

CHAPTER 3

Like Chapter 1, this Chapter is in its infancy. However, with Version 1B of MCNP some experimental output is being introduced that deserves some explanation. In addition to the regular balance sheet that has always appeared in the output immediately after the conclusion of the MCRUN overlay, some new information is now being printed. Hopefully this new information will give the user better insight into his problem and how various Monte Carlo games he may have played worked. This new output is experimental – it is for neutrons only and we do not know how much of it is useful or irrelevant. By experience we hope to improve this type of information. Any feedback from users regarding what is useful, not useful, or what other information would be useful will be appreciated. Note that lost particles may lead to inconsistent results in both the old and new summary information.

The first page of the new summary information is somewhat redundant of the old balance sheet. The reason for this is that someday the old balance sheet may be deleted in favor of the new information sheets because the format of the old sheet is not very conducive to change. Much of the information on this first new page is self-explanatory. The *Weight Cutoff* information gives you an idea of how well the Russian roulette game was played in a cell with weight cutoffs (see page 1 and page 35) and maybe how well or poorly the weight cutoffs were set. The *Import. Sampling* information tells you the number of tracks created from surface splitting and the number of tracks lost to Russian roulette on surfaces (see page 14). It also tells you what the total weight gain and loss is from Russian roulette on surfaces. If there is a weight imbalance here, either cell importances were poorly set or there was not enough particle traffic to adequately play the roulette game. The term *All Capture* refers to both analog capture and capture by weight reduction. *System Multiplication* is defined as one plus fission weight plus (n,xn) weight divided by the source weight. *Weight Lost Per Neutron* on the old balance sheet is equal to *System Multiplication* plus the net weight gain from Russian roulette in the weight cutoff game plus the net weight gain from Russian roulette in the surface importance sampling game. On the old balance sheet, the *Weight Lost Per Neutron* is simply the sum of the losses to energy cutoff, time cutoff, capture, and escape. See page 3 regarding a caution when using weight cutoffs and multiplication is important to you.

The second new summary page gives an idea of the particle traffic

throughout the geometry. *Tracks Entering* is very explicit – only tracks crossing a surface to enter a cell plus any source particles are counted. Furthermore, if a track leaves a cell and later re–enters that cell, it is counted again. Therefore these numbers dc not represent population in a cell and so do not necessarily tell you how well your splitting is doing in keeping the population constant (see page 14). *Weight Balance* is here defined as (remember it's experimental) the weight of particles leaving a cell minus the weight of particles entering the cell normalized per starting neutron. Source particles are excluded from this balance. Items adding positively to this balance are fission and (n,xn). Items making a negative contribution are capture, energy cutoff, and time cutoff. Russian roulette for weight cutoff may contribute either way depending on how well it is sampled. Splitting and Russian roulette on surfaces have no effect on the balance. An idea of the contributions of these components for the total problem (but not on a per cell basis) can be found on the first new summary page. The total weight balance on this page is equal to the weight lost per neutron on the old balance sheet minus the weight loss to escape minus the weight gain from fission and (n,xn) reactions minus the new weight gain from the two Russian roulette games.

The third page of new information will give you an indication of how important certain isotopes are in your materials. If an isotope is relatively unimportant, you may consider changing cross–section evaluations for that isotope to an evaluation with fewer words and then use a better evaluation (with perhaps more words) for the more important isotopes. The information on this page can perhaps give you an indication of just how important trace elements are in your geometry.

APPENDIX A

HOW TO USE MCNP ON LTSS

This appendix is divided into four sections:

(1) Running MCNP
(2) Updating MCNP
(3) Subroutine SOURCE
(4) DBCTRL

Users should take the time to become familiar with these four sections.

Several example ORDER files are illustrated in this Appendix, but in no way are we saying these are the ones that you should use. They are merely possible ways of doing things, and you should rewrite them to do your particular job for your particular environment. However, you will probably find much (if not all in some cases) in the files useful to you. Any items you will most likely need to change are written in italics.

All MCNP Version B files (except cross sections) are in a LIX file called MCNP1B on photostore. This file includes backups to the public files and to other files on disk. Included is also a file called INFO that tells what all the other files are. Some of the files in MCNP1B are:

INFO	information file
MCNPB	executable binary file (MCNP)
TGMCNPB	UPDATE correction file going from 1A to 1B
MCNLODB	LOD instructions (MCNPLOD)
MCNPMAP	load map from LOD
COMPASS	Compass source library, LIX file containing FRN, EX, FRNS, NRAND, CONT, LOCF, COMBLIB, and EXLCM
COMBLIB	binary library of COMPASS
TEKTLIB	graphics library
ORDB	LIX file of ORDER routines
MCNPEST	LIX file of sample input and patches
MCNTEKB	binary library file including TEKTLIB and COMPLIB (MCNPLIB)
MCNPOPL	UPDATE OLDPL file

The equivalent public files are indicated in parentheses in the above list. MCNPU, MCNPOPL, MCNPMAP, ORDB, and MCNPEST are also on Hydra disk.

The six cross-section files are each on photostore and are called RMCCS1B, AMCCS1B, DRMCS1B, XMCCS1B, UMCCS1B, and MCPIB1B. The corresponding public files are RMCCS, AMCCS, DRMCCS, and MCPLIB. XMCCS and UMCCS are also on Hydra disk but not public files.

All MCNP1A files (such as MCNP, MCNPU, and ORD) are in a LIX file called MCNP1A on photostore. One file in this LIX file is called INFO; it tells what the other files are. The names of the Version 1A cross-section files are RMCCS1A, AMCCS1A, DRMCS1A, XMCCS1A, and MCPIB1A; these are each on photostore.

All the above files are unclassified and have universal read-only access. The Hydra files all have access code TD6CODE.

(1) *Running MCNP*

The MCNP execution line has the following form:

MCNP *Files Options* / t p

where *Files* and *Options* are described below.

Files: The MCNP code uses several files for input and output. The default names for these files are given in Table A.1.

Table A.1
MCNP Files

Default Name	I/O Unit No.	Description
INP	2	Problem Input
OUTP	4	Problem Output
BANKFL	6	Bank Overflow
RUNTPE	12	Restart Dump
FILM	14	PLOT Film File
SURFACE	15	Surface Coefficients from PLOT
SNAPF	22	SNAP Source Distribution
SPARED1	30	User Neutron Cross Sections
RMCCS	30	Neutron Cross Sections

AMCCS	30	Neutron Cross Sections
XMCCS	30	Neutron Cross Sections
UMCCS	30	Neutron Cross Sections
SPAREC1	30	User Neutron Cross Sections
MCPLIB	34	Photon Cross Sections
	59	Messages to Controllee
MCNP	99	Code Overlays
DUMN1	.	User Scratch File
DUMN2	.	User Scratch File

The user can change a default file name on the execution line by entering

Default Name=User File Name

For example, if one wishes to call his input file MCIN and his output file MCOUT, he should enter the following on the execution line:

```
MCNP INP=MCIN OUTP=MCOUT / t p
```

If there are no changes in default names, then nothing is entered for *Files*.

If the specified file name for the printed output or the run file already exists in your local files, MCNP creates files with different unique names by changing the last letter of the name to the next letter in the alphabet.

RMCCS, AMCCS, DRMCCS, and MCPLIB are public files. XMCCS and UMCCS are Hydra disk files (AC=TD6CODE) that the user must put in his local files himself. SPAREC1 and SPARED1 are available for a user's private cross-section libraries.

Options: The MCNP code consists of up to five separate operations each performed by a single overlay. These operations and their corresponding overlays are listed below along with a one character mnemonic for each operation:

Mnemonic	Overlay	Operation
I	IMCN	process problem input file
P	PLOT	plot geometry
V	VOLUME	calculate cell volumes
X	XACT	process cross sections
R	MCRUN	particle transport

Note: to calculate volumes with the V option, the volumes of interest must be symmetric about the y-axis.

The user can control the execution of these operations by entering the proper mnemonic on the execution line. If more than one operation is desired, the single characters (in any order) may be combined to form a string. For example, if one wishes to execute IMCN, VOLUME, XACT, and MCRUN, he would enter

```
MCNP IVXR / t p
```

Similarly if one merely wishes to debug an input by plotting, he would enter

```
MCNP IP / t p
```

The character *I* should always be entered in an option string and the character *R* should not be entered without the *X* option. A string cannot be used for a continue-run (see below).

Some common strings formed by the above character options may be replaced with equivalent "period" options. These period options plus others that add more flexibility to MCNP are listed in Table A.2.

Table A.2
Period Options

Option Mnemonic	Operations
ALL.	Equivalent to IPVXR
RUN.	Equivalent to IVXR
C. m	Continue a run starting with m^{th} dump. If m is omitted, last dump is used

DBUG. n	Write debug information of DBCN card every n particles
GROSS.	Do not reduce the LCM field length
NOTEK.	Assumes use of a TTY versus a graphics terminal; plot output is in a local film file
FATAL.	Transport particles even if fatal errors are found in initiation
D.	Destroy drop file unless there is an abnormal termination of MCNP Default is do not destroy drop file

This list of options requires that a period be placed after the mnemonic. The options DBUG., GROSS., FATAL., D., and NOTEK. can be used along with any of the other options. For example the line

```
MCNP INP=MCIN RUN. DBUG. 1000 / t p
```

results in file MCIN being used for input, overlays IVXR being run, and a debug statement being printed every 1000 particles.

If no option is specified, the default RUN. is used.

Interrupts: The MCNP code allows four interrupts:

(ctrl e) IS	give MCNP status
(ctrl e) IT *time*	changes time limit to *time* in minutes
(ctrl e) IQ	stops calculation of MCNP
(ctrl e) IP	pause; continue with return key

The status of MCNP consists of: (1) overlay being run, (2) computer time used, and (3) if in MCRUN the number of particles run.

The IQ message simply stops the calculation if MCNP is not in the MCRUN overlay. However, if MCRUN is being executed, this command causes the run to stop after the current particle history. The process allows MCNP to terminate "gracefully" thus producing a dump file and final print output.

To run MCNP without any Updates or without an user-provided source

subroutine, simply type in the MCNP execution line. When the job is finished, you will have in your local files the BCD output file OUTP - unless you have changed this default name on the execution line to something else. All the MCNP output is on this file - look at it with TRIX AC and/or send it to a printer and/or fiche.

A simple ORDER program, ORD1B, to get your input file, execute MCNP, and send your output to the printer (or microfiche if you substitute FR80 for PRINTER in the last line) can be found in Figure 1. ORD1B is in a LIX file ORDB on Hydra disk (AC=TD6CODE). The program assumes the MCNP input file TEST1 is in the LIX file MCNPEST which is a Hydra disk file (AC=TD6CODE). To use this program, simply type in (assuming you have ORD1B in your local files):

ORDER ORD1B / t p

Figure 1
ORD1B

```
*id    account#          ord1b                    yourname          box###
$
$    run mcnp - no updates or subroutine source (fig. 1)
$
*xeq xport
*xeqmes td6code get mcnpest \ end
*nxt
*xeq lix
*xeqmes mcnpest \ get test1 \ e
*nxt
*xeq mcnp
*xeqmes ixr inp=test1
*nxt
*xeq allout
*xeqmes printer outp fam. box ### ord1b/yourname
```

You will have to change the Box and ID in the last line to your own and also change the ORDER ID card (line 1) for your particular use.

An ORDER program, ORD2B, that does the same job as ORD1B but saves the job drop file, run file, and output on tape using the LTSS routine MCT is shown in Figure 2. ORD2B (again available in ORDB) also contains a BATCH

line as the first line of the file and may be submitted to BATCH for night-time production. See the BATCH writeup for details. The BATCH line does not have to be removed for daytime use of ORD2B, since it starts with a $ and is therefore assumed to be a comment line by ORDER. ORD2B makes use of the fact that the drop file name is always +MCNP, the run file is RUNTPE, and the output file is OUTP (unless local files already exist by those names).

Figure 2
ORD2B

```
$batch (     your stuff       ) 1414 ord2b/batch
*id    account#          ord2b                    yourname          box###
$
$    run mcnp - no updates or source subroutine and save drop file,
$               run file, and output file (fig. 2)
$
*xeq xport
*xeqmes td6code get mcnpest \ end
*nxt
*xeq lix
*xeqmes mcnpest \ get test1 \ e
*nxt
*time n
*xeq mcnp
*xeqmes ixr inp=test1
*nxt
*xeq mct
*xeqmes id. yourname group check files runtpe +mcnp outp
*xeqmes tapes * end
*nxt
*xeq allout
*xeqmes printer outp fam. box ### ord2b/yourname
```

Something like ORD2B should be used routinely for BATCH jobs so the drop file will be available for debugging the following day. These extra lines in ORD2B can also be used in the sample ORDER programs that follow. RUNTPE is saved for a later continue-run, plus this is a very safe prodecure to circumvent system malfunction or lost output. The job can be

retrieved for a continue-run or debugged if there are problems, and the tape can then be released when everything is satisfactory (like output regenerated).

The MCT routine is very reliable; however, it may return several error messages like ERROR: 43, but rarely do they cause any problems. A tape will be assigned with a name like *FM008 - the asterisk is just decoration so don't use it when you later use MCT to read the tape.

Another safe and recommended procedure for production jobs that is shown in ORD2B is to use a *TIME n card (n for CPU minutes) in your ORDER file just before the *XEQ MCNP line. If the ORDER job has several more minutes on it than the *TIME card does and MCNP is not stopped by NPS or CTME, the *TIME will stop MCNP and then several more minutes will still be available to wrap up the rest of the ORDER job (like saving everything on tape).

(2) *Updating MCNP*

When Updating MCNP, the user will require two more files in addition to an UPDATE patch, both available on Hydra disk (AC=TD6CODE):

MCNPU	the MCNP compile file containing all the UPDATE identifiers, and
MCNPOPL	the UPDATE OLDPL of MCNP

The LOD instructions are in the public file MCNPLOD, and a copy of them is in lines 5 through 11 in MCNPU. An MCNP load map, MCNPMAP, is on HD, AC=TD6CODE. A listing of MCNPU is conveniently obtained by using

```
ALLOUT PRINTER MCNPU SEQ. CCSP. PPF. 300 BOX --- ----
```

MCNPU is itself compilable, so TRIX AC changes can be made to it directly and then re-compiled and re-loaded if desired. However, before compiling MCNPU, the UPDATE identifiers on the first four TRIX AC lines must be trimmed off.

The procedure for Updating is to first prepare your UPDATE patch by using an MCNPU listing to get FORTRAN lines and corresponding identifiers. An example of a patch, UPD, is given in Figure 3.

Figure 3
UPD Patch

```
1 *ident r1
2 *i,o.148
3 c
4 c (fig. 3)
5 c
6       write(4,22)
7       write(59,22)
8    22 format(7htesting)
9 *compile chat,c
```

The last line (or appropriate variant) of this patch must always be included. CHAT and C are deck names and consist of, for CHAT,

```
$CHAT %ME C MCNPA$ MCNPL M77 L L
$DESTROY MCNPB
$LIBMAK C. MCNPA MCNPB 260000
$LOD ABC # MCNPLOD%
```

and for C, the MCNP Common blocks. MCNPLIB is a public binary library of the basic MCNP code plus TEKTLIB and COMPLIB. The 260,000 LCM size on the fourth line should be adequate for most UPDATE patches. However, for very large patches this may have to be increased (and also the corresponding number in Figure 6).

Note that writing to the I/O Unit 4 before identifier 0.148 should not be attempted since the unit before then has not been assigned.

An alphabetical list of all the MCNP subroutines and UPDATE decks can be found in Table A.3 on page 64. At the end of the MCNPU file are two tables – the first lists all the MCNP subroutines in alphabetical order followed by the line number where they appear in MCNPU, and the second table is a numerical ordering (by MCNPU line number) of the subroutines.

A file called Compile will be in your local files after the UPDATE. This file is the FORTRAN listing (including UPDATE identifiers) of the portion of the code you modified and contains your changes. To obtain a listing of the entire code (not just the modified decks) containing your patch, you need to re-compile the entire code. This may be done by changing the last line of Figure 3 to be

*COMPILE CHAT.V

which compiles the entire code since CHAT is the first deck of MCNP and V is the last.

If COMMON or a parameter is being changed then all decks must be re-compiled and written to the COMPILE file by using

*COMPILE CHAT.V

After your UPDATE patch has been prepared, the most convenient way to go through the sequence of UPDATE, re-compiling, and re-loading is to use something like the ORDER file ORD3B of Figure 4. This program, again available in the LIX file ORDB on TD6CODE, assumes your input file is called TEST1, the patch is called UPD, and that both are in the LIX file MCNPEST which is on Hydra disk (AC=TD6CODE). After the UPDATE, your new MCNP is run with the options IXR and the output is sent to the printer. Please modify ORD3B for your own needs, i.e., the Box numbers, file names, access codes, etc.

Figure 4
ORD3B

```
*id    account#          ord3b                    yourname          box###
$
$    update mcnp but use a standard source (fig. 4)
$
*xeq xport
*xeqmes td6code get mcnpopl \ end
*nxt
*xeq xport
*xeqmes td6code get mcnpest \ end
*nxt
*xeq lix
*xeqmes mcnpest \ get test1 upd \ e
*nxt
*xeq update
*xeqmes (i=upd,p=mcnpopl,8,d)
*nxt
*xeq trix
*xeqmes ac \ o \ compile \ run \ end
```

```
*nxt
*xeq mcnp
*xeqmes ixr inp=test1
*nxt
*xeq allout
*xeqmes printer outp fam. box ### ord3b/yourname
```

The local file MCNP generated by ORD3B in Figure 4 may (and should) be saved if you intend to run several jobs with the same patch to avoid going through the UPDATE procedure again. Modify ORD3B to save MCNP on Hydra and then modify ORD1B of Figure 1 to get that version of MCNP from Hydra for you. In ORD1B, for example, you might add the three lines:

```
*NXT
*XEQ XPORT
*XEQMES TD2XYZ GET MCNP
```

Of course in ORD3B you could have added some lines to switch the name of MCNP to something like MCNPDWW for your own personal, modified version of MCNP. Then the corresponding name change would be made in ORD1B.

A problem with it being so easy to make your own version of MCNP is that when we change our MCNP, you will need to re-UPDATE to incorporate our changes. Naturally we'll let you know by Newsletter when we make a change, and it will be your responsibility to make the change too. The only version of MCNP that TD-6 will support will be our current public version.

(3) *SUBROUTINE SOURCE*

When providing your own source subroutine for MCNP, don't forget to include an SRC card in your MCNP input file. Unless your subroutine defines some parameters in the SRC array (up to 51 may be used) to be input on the SRC card, you are not required to have any - just an SRC card with nothing else on it. This source has the form

```
SUBROUTINE SOURCE
MCNP Common
Specification of Source Parameters
RETURN
```

```
ENTRY SRCDX
Specification of SRCDX Parameters
END
```

The source parameters which must be specified within the subroutine are:

Variable	Description
X	x-coordinate of particle's position (in cm)
Y	y-coordinate of particle's position (in cm)
Z	z-coordinate of particle's position (in cm)
U	x-axis direction cosine of particle's direction
V	y-axis direction cosine of particle's direction
W	z-axis direction cosine of particle's direction
IA	The program name of the cell containing the source particle, or in the case of a surface source, the cell which will be entered.
JA	For a surface source, this must be the program name of the surface upon which the particle is starting. If the particle is not starting on a surface, JA must be set to zero.
TME	Particle's starting time in shakes (10^{-8} sec)
WT	Particle's weight (usually 1.0)
ERG	Particle's energy in MeV

Prior to calling subroutine SOURCE, the MCNP code calculates an isotropic (U,V,W). Therefore, the user need not specify the direction cosines (U,V,W) if he desires an isotropic distribution.

The SERG, SPROB, and SBIAS cards may also be used with the SRC card to

define the particle's energy ERG and weight WT. To do this, however, a call to the energy sampling subroutine, CALL ERGSAMP, must be in the subroutine SOURCE.

Two random number functions are available for use by subroutine SOURCE: FRN(KRN) and FRNS(KRN). FRN(KRN) produces a random number between 0 and 1; FRNS(KRN), between -1 and 1.

Up to 51 entries (such as energy, coordinates, or anything else) may be put on the SRC card. They will be stored in the SRC array which is in Common and can therefore be used anywhere in MCNP. Most frequently, however, the SRC array is used to input parameters to the user-provided subroutine SOURCE, and the order and meaning of entries on the SRC card depend upon the structure of SOURCE.

If a user has supplied a subroutine SOURCE and he is using a point or ring detector in his calculation, he generally needs to supply another subroutine - SRCDX (or more conveniently, an entry point in subroutine SOURCE). The structure of this subroutine is the same as SOURCE except that usually only a single parameter needs to be specified - PSC. PSC is the probability density function for emitting a particle directly at the detector. If the source is isotropic (PSC=.5), SRCDX does not need to be specified.

PSC must be determined for <u>each</u> detector used and defined in SRCDX each time the source contribution is determined for a particular detector. A table of PSC values for each detector could be set up in subroutine SOURCE from which SRCDX could pick the appropriate PSC for the particular detector for which the source contribution is being calculated.

If subroutine SOURCE specifies a source on a surface that can send particles into more than one cell and detectors are being used, SRCDX must also determine the program cell number IA for the source particle. An alternate method that eliminates the cell determination in SRCDX is not to use a surface source, but specify your source an epsilon distance inside one of the cells.

Any directional-dependent source quantities, such as energy or weight, must also be specified in SRCDX if detectors are used.

When using detectors and a subroutine SOURCE, it is recommended to use the DRECT card to see if the source contribution to the detectors is close to what you think it should be.

There are two ways to get your own source subroutine into MCNP. One way is to do it through UPDATE - put your source in a patch and use the methods of the previous section. The other way is to compile the subroutine as a separate entity and then load it into MCNP.

With the UPDATE method, subroutine SOURCE already exists in MCNP as a dummy subroutine. Simply delete this subroutine and add your own:

```
*d,ss.2,ss.6
SUBROUTINE SOURCE
  .
  .
(FORTRAN lines)
  .
  .
END
```

If you use SRCDX as an entry point in SOURCE, then the dummy SRCDX already in MCNP must be deleted (identifiers SS.7 through SS.12).

You do not have to provide Common blocks to the subroutine with this method. An example of the entire patch UPDSRC (which is also combined with the patch of Figure 3) is shown in Figure 5. The ORDER program to run this job is identical to ORD3B, only the name UPD must be changed to UPDSRC in the LIX and UPDATE lines.

Figure 5
UPDSRC Patch

```
 1 *ident r1
 2 *i,o.148
 3 c
 4 c    (fig. 5)
 5 c
 6         write(4,22)
 7         write(59,22)
 8     22 format(7htesting)
 9 *ident s1
10 *d,ss.2,ss.6
11 c
12         subroutine source
13         ia = 1
14         ja = 0
15         x = 0.
16         y = 0.
```

```
      z = 0.
      wt = 1.
      erg = src(1)
      tme = 0.
      u = 0.
      v = 0.
      w = 1.
      return
      end
*compile chat,c
```

The other method of providing your own source subroutine is to have a complete subroutine SOURCE (including the Common blocks) as a local file, compile it, and load it into MCNP. This is more efficient than using UPDATE if you are only adding a source subroutine. The source of Figure 5 becomes 99 lines long, and the first and last few lines are shown in Figure 6. This subroutine is the file SRC in the LIX file MCNPEST. Note the first three lines - they must be included for this scheme to work. The most convenient way to accomplish all the necessary steps is to use an ORDER file such as ORD4B shown in Figure 7. As before, ORD4B is available in the LIX file ORDB on Hydra disk under AC=TD6CODE.

Figure 6
Source

```
$chat %me c mcnpa$ mcnpl m77 1 1
$libmak mcnplib mcnpa mcnpb 260000
$lod abc # mcnplod%
   subroutine source
c
c       (fig. 6)
c
   integer amax,bmax,emax,pbmx,pmax,smax,tmax,tsurf
   parameter(amax=175,bmax=100,emax=40,jfm=150,jmax=175,memax=120,
  1 mxdt=10,mxfm=16,mxmt=250,mxsl=262140,n99c=18,n99e=30,
                .
                .
                .
                .
  4 (zrt,xsl(mp13)),(nrt,xsl(mp14)),(koj,xsl(mp15)),(joa,xsl(mp16)),
```

```
85    5 (lcj,xsl(mp17)),(nas,xsl(mp18)),(pthurm,fl(mp7a))
86
87          ia = 1
88          ja = 0
89          x = 0.
90          y = 0.
91          z = 0.
92          wt = 1.
93          erg = src(1)
94          tme = 0.
95          u = 0.
96          v = 0.
97          w = 1.
98          return
99          end
```

The Common blocks necessary for the subroutine can easily be obtained from the file SRC in the LIX file MCNPEST (HD, AC=TD6CODE) or from deck C of the MCNPU file. Open the MCNPU file with TRIX AC, use the TRIM command (just type in the word TRIM) if you want to get rid of all the identifiers, and copy lines 16 through 93 into your source file right after the fourth line - the one that says SUBROUTINE SOURCE.

Figure 7
ORD4B

```
*id    account#          ord4b                    yourname          box###
$
$    load in a source subroutine with no updates (fig. 7)
$
*xeq xport
*xeqmes td6code get mcnpest \ end
*nxt
*xeq lix
*xeqmes mcnpest \ get test2 src \ e
*nxt
*xeq destroy
*xeqmes mcnpb
*nxt
```

```
*xeq trix
*xeqmes ac \ o \ src \ run \ end
*nxt
*xeq mcnp
*xeqmes ixr inp=test2
*nxt
*xeq allout
*xeqmes printer outp fam. box ### ord4b/yourname
```

If you have an UPDATE patch plus a separate source subroutine file that is not part of the patch, ORD5B of Figure 8 will UPDATE with the patch, compile the subroutine, and do the necessary loading of both.

Regardless how you get your source subroutine into MCNP, if you plan many runs with this same source, you should save the new version of MCNP (or whatever you choose to call it) to run again without having to re-create it everytime.

Figure 8
ORD5B

```
*id    account#          ord5b                    yourname          box###
$
$    update plus load in a non-updated source (fig. 8)
$
*xeq xport
*xeqmes td6code get mcnpopl \ end
*nxt
*xeq xport
*xeqmes td6code get mcnpest \ end
*nxt
*xeq lix
*xeqmes mcnpest \ get test2 upd src \ e
*nxt
*xeq update
*xeqmes (i=upd,p=mcnpopl,8,d)
*nxt
*xeq trix
*xeqmes ac \ o \ compile \ run \ end
*nxt
```

```
*xeq switch
*xeqmes mcnpb mcnplib
*nxt
*xeq trix
*xeqmes ac \ o \ src \ run \ end
*nxt
*xeq mcnp
*xeqmes ixr inp=test2
*nxt
*xeq allout
*xeqmes printer outp fam. box ### ord5b/yourname
```

(4) *DBCTRL*

MCNP is set up to use John Norton's DBCTRL debug routine. To use DBCTRL, the user should get the latest DBCTRL manual from the CCF Program Library.

When running MCNP, note that the name of the drop file for your particular job is printed out in the form +MCNP (or +MCNPA, etc., if other drop files of the first name already exist). The symbol table is loaded at the end of the file MCNP and the possible overlays (or chains) to be entered are MCNP, IMCN, PLOT, VOLUME, XACT, or MCRUN. Note that the names of overlays are followed by semicolons, and names of subroutines are followed by colons.

A list of all MCNP subroutines with the overlay and UPDATE deck they are in is in Table A.3. Note that in this table several subroutines appear in more than one overlay. The end of the MCNPU file contains two additional lists of subroutines – one in alphabetical order and the other in order of occurrence in MCNP. The numbers in the latter two tables correspond to the line numbers of MCNPU.

APPENDIX A
DBCTRL

Table A.3

Overlay / Deck / Subroutine Structure

Overlay	Deck	Subroutine	Overlay	Deck	Subroutine
	CHAT		MCRUN	NX	ACECASE
	C			NX	ACECOLL
				NX	ACECOS
MCNP	O	CREXFIL		N	ACEERG
	O	ERRPRNT		NX	ACEFCN
	O	EXCHPKG		NX	ACEGAMA
	O	EXPIRE		NX	ACENU
	O	DBPNT		NX	ACETBL
	O	MCNPS		N	ACETOTN
	O	PRNTF		P	ACETOTP
	O	SECOND		M	ANALYS
	O	TPEFILE		G	ANGLE
	O	TTYINTR		M	BANKIT
				T	BIN
IMCN	I	CELSURF		N	CALNPSC
	I	CHEKCS		P	CALPPSC
	O	CREXFIL		G	CHKCELL
	O	ERRPRNT		N	COLLIDN
	I	IMCNS		P	COLLIDP
	I	ITALLY		O	CREXFIL
	I	MATERIA		M	DBUGG
	I	PRSPACE		T	DDDET
	I	PRSRF		SB	ERGSAMP
	I	PSURF		NX	FISP
	I	RDLINE		N	HSTORYN
	I	RDPROB		P	HSTORYP
	I	RECOS		NX	IFISP
	O	SECOND		G	ISOS
	I	SOURCEX		P	KLEIN
	I	TALLYPR		R	MCRUNS
	O	TPEFILE		G	NEWCELL
				P	PHOTONC
XACT	X	ACEFPUT		M	PRIN
	X	ACERIN		T	PRTET

	O	CREXFIL
	O	ERRPRNT
	X	EXPUNGE
	X	PHOCRS
	I	PRSRF
	O	SECOND
	X	SNAPPY
	O	TPEFILE
VOLUME	V	SQRZ
	V	VOLUM
PLOT	PL	ADVXX
	PL	CONVRT
	PL	DECUSO
	PL	DILATO
	PL	DRVXX
	PL	EMPTYXX
	PL	IMOVE
	PL	INVMAT
	PL	LEGERE
	PL	LINEAR
	PL	LJSTFY
	PL	NORMA
	PL	PLANUS
	PL	PLOTS
	PL	REGULA
	PL	RMOVE
	PL	SCRIBO
	PL	SENSUS
	PL	SURFTR
	PL	SWAP
	PL	TODAY
	PL	TSPXX
	PL	UNORM
	PL	WLAB
	PL	WLCVXX
	PL	ZBNDRY
	PL	ZSENSE

G	ROTAS
O	SECOND
SB	SOURC99
SS	SOURCE
SB	SOURCEA
SS	SRCDX
M	SUMMARY
N	SURFACN
P	SURFACP
T	TALLYC
N	TALLYDN
P	TALLYDP
T	TALLYP
T	TALLYS
T	TALSHFT
G	TORUS
O	TPEFILE
G	TRACK

APPENDIX A
DBCTRL

An MCNP execution may be statically examined with the DBCTRL facility by starting with the following teletype entry:

DBCTRL +MCNP MCNP / t p

where +MCNP is the drop file from an error-terminated or manually terminated MCNP run. The second file name is the name of the associated symbol table. For example, the contents of the variable IG in subroutine MATERIA in overlay IMCN may be examined by entering:

IMCN; MATERIA: IG

This entry sets the "current overlay and subroutine," and other variables in that subroutine may be examined by entering a blank delimited list of variable names. Several elements of an array may be printed out by following the array element with a comma and the number of elements desired [i.e., A(17),4 will print out A(17), A(18), A(19), and A(20)]. To examine variables in other overlays and subroutines, it is necessary to reset the current overlay and subroutine by entering the new overlay name followed by a semicolon and a new subroutine name followed by a colon (i.e., MCRUN; HSTORYN:).

For this example, the INP file is TEST2 in MCNPEST. An external source subroutine was used and written so that data are put on the SRC card and hence entered into the SRC array. The source is the same as in Figure 5 except for three lines: A=1.0, B=SRC(2), and X=A/B.

The job aborts in the MCRUN overlay (obvious from the output and terminal messages). The location of the error (73340) is found in the P register of the exchange package which is printed in the MCNP output file.

The user input in this example is in italics.

Figure 9
Static DBCTRL

```
dbctrl +mcnp mcnp
 dbctrl ver. 4  03/15/78
? mcrun;
? find. 73340
 073340 is 000013b locations after the origin of routine source
      in code block mcrun
```

```
? source:
? ia ja a b x y
 ia (0064564) = 1
 ja (0064565) = 1
 a (0073350) = 1.e+00
 b (0073351) = 0
 x (0064552) = 0
 y (0064553) = 0
? src(1),3
 src(1) (0054207) = 1.e+01, 0, 0
? wt erg
 wt (0064561) = 0
 erg (0064560) = 0
? end
 all done
```

The FIND. command locates the error somewhere near the beginning of the source subroutine. Several variables, including part of the SRC array, are interrogated. The job aborted when trying to calculate X since SRC(2) = 0 and hence B = 0. The problem is with the SRC card in the TEST2 input file; a second input value was not added to the card.

In the dynamic mode, an MCNP execution may be interrupted for interrogation at various preset breakpoints by use of DBCTRL. The teletype entry

```
DBCTRL MCNP MCNP / t p
```

will begin a dynamic DBCTRL execution. Breakpoints may be set initially or at any pause, and execution is resumed between breakpoints by the RUN. command. Execution of MCNP from a static DBCTRL examination may also be continued with the RUN. command. Breakpoints may be set and held at the beginning of particular subroutines, at statement labels, and at particular octal addresses by the commands BKPSUB., HBKP., and HBKPA. . Corresponding commands can be used to remove breakpoints. The TRACE. command enables a breakpoint to be set after passing a statement label several times (i.e., in the middle of a loop such as after every tenth history). All currently set breakpoints can be listed by the command LISTBKP. .

The MCNP execution line information is entered with the RUNM. command. It should be noted that variables appearing in a PARAMETER statement,

argument list of the current subroutine, and variables in Common that are not referenced in the current subroutine may not be available under DBCTRL.

Several other DBCTRL features may also be useful, including the FIND. and REPLACE. commands. The BYON. command will allow you to use the MCNP CTRL-E interrupts. Detailed examples of the use of DBCTRL can be found in LTSS-529, vols. 1 and 2. Note that by default DBCTRL sets a time limit for its controllee (MCNP) at one minute. If this is not sufficient, it can be changed with the TIME. command.

An example of using DBCTRL dynamically on the TEST1 problem of MCNPEST is found in Figure 10. This example illustrates setting breakpoints, looking at several variables, and manually changing variables (NPP). The user's input is shown in italics.

Figure 10
Dynamic DBCTRL

```
dbctrl mcnp mcnp / 3 1
 dbctrl ver.4  03/15/78
? copy.
 the controllee is now mcnp+
? bkpcball.
? mcrun;
? bkpsub. mcruns
 return address storage could not be located
      breakpoint may not be set at the proper location.
? hstoryn: trace. 30 3
? runm. ixr inp=test1
mcnp  version 1b      t 03/28/78  12:13:53  drop file is  +mcnp
 the controllee is now +mcnp
 breakpoint hit at the beginning of code block imcn
? npp nps wcn1
 npp (0061200) = 0
 nps (0061201) = 0
 wcn1 (0063131) = 0
? run.
 warning.  last time bin of tally  1 is less than cutoff.
 warning.  last time bin of tally  5 is less than cutoff.
imcn   is done
 breakpoint hit at the beginning of code block xact
? npp wcn1 wcn2
```

```
 npp (0061200) = 1000
 wcn1 (0063131) = -1.e+00
 wcn2 (0063132) = -5.e-01
? run.
dump no.  1 on file  runtpe    nps =     0    ctm =    0.
xact    is done
 breakpoint hit at the beginning of code block mcrun
? nfer nps trace(5),10
 nfer (0061176) = 0
 nps (0061201) = 0
 trace(5) (0054346) = 13, 41, 1063, 1089, 1102, 1115, 1887, 1901, 4847
          (0054357) = 4860
? run.
 breakpoint hit at the beginning of subroutine mcruns
      in code block mcrun              (address - 066776)
? nps oct. ijk
 nps (0061201) = 0
 ijk (0060662) = 0b
? dec. run.
lcm field length = 119739 = 0351673b      cp0 =  0.23
 breakpoint being traced at label 30       of subroutine hstoryn
     in code block mcrun      (address = 070036) has oddurred
          3 times. trace flag removed but breakpoint held.
? nps npp
 nps (0061201) = 3
 npp (0061200) = 1000
? rbkpall. mcruns: trace. 160 25 run.
 breakpoint being traced at label 160      of subroutine mcruns
      in code block mcrun       (address = 067320) has occurred
          25 times. trace flag removed but breakpoint held.
? nps npp
 nps (0061201) = 27
 npp (0061200) = 1000
? replace. npp 35
? mcrun; bkpsub. prin
? rbkp. 160 run.
     run terminated when         35 particle histories were done.

 breakpoint hit at the beginning of subroutine prin
      in code block mcrun        (address - 071536)
```

APPENDIX A
DBCTRL

```
? nps npp
 nps (0061201) = 35
 npp (0061200) = 35
? run.
dump no.  2 on file  runtpe     nps =     35    ctm =     0.62
mcrun is done
 +mcnp       has terminated
 warning - the dropfile is still considered to be +mcnp
           if this is not so, use the change. command.
 dynamic commands are herewith disabled for the current controllee
? end
 all done
```

APPENDIX B

Monte Carlo Neutron Cross-Section Format (ACE)

The MCNP neutron cross-section libraries are in ACE (A Compact ENDF) format. ACE format consists of a number of binary records as illustrated in Figure 1.

Nuclide Cross-Section Records

The first record of each MCNP neutron cross section library is the nuclide directory. Following the nuclide directory are all the neutron cross-section data for all the nuclides listed in the directory. Each nuclide has three consecutive records. The PREF record contains the TRACE array - a short (40 words) array of identifiers, pointers, flags and Hollerith information. Then after a pointer word comes the EST record which contains the reaction cross-section energy grid and the total cross-sections. After another pointer word comes the CAPE record which contains the remaining blocks of data.

The data in the PREF, EST and CAPE records for each nuclide may be thought of as a single array, IN(I), I=1,END where END is less than or equal to 100,000. The PREF record, which comprises the first NPREF=40 words of the IN array, contains pointers for the remaining data in the IN array. In this Appendix data are described in terms of a single IN array because many ACE format processing codes use a single IN array and since the cross-section data libraries are laid out like a series of IN arrays for each nuclide.

In MCNP the PREF, EST and CAPE records are loaded into three different arrays: the PREF record is loaded into the TRACE array; the EST record is loaded into the XSN array if there is sufficient space and into the XSL array if XSN is full; the CAPE record is loaded into the XSL array. All data in the EST and CAPE records which are not used, particularly data outside the energy range of interest, are deleted by routine EXPUNGE. Then the PREF record

pointers (which are now in the TRACE array) are changed to correctly locate data in the XSN and XSL arrays.

How to Locate ACE Format Data in the Cross-Section Libraries

To locate any ACE format data in the cross-section libraries, first use the nuclide directory (Table 1) to find the starting location of the IN array for the nuclide of interest. Then use the data in the PREF record (Table 2) to locate the specific data block of interest. These data blocks are described in Tables 4-16.

How to Locate ACE Format Data in MCNP

To locate ACE format data for the N^{th} nuclide in MCNP simply use the pointers in the TRACE array (Table 3). These pointers are of the form TRACE (ITR+I), I=1,NPREF where NPREF=40 and ITR=N*NPREF. The pointers locate different blocks of data in the XSN and XSL arrays. These blocks of data are described in Tables 4-16.

Units

ACE format requires that all energies be given in MeV; all cross sections are in barns ($10^{-24}cm^2$); heating numbers and Q-values are in MeV. Angles are described in terms of cosines.

Figure 1
Diagram of ACE Format Binary Records
for a Typical MCNP Neutron
Cross-Section Library

record number	starting address	record description	
			pointer word
1	1	Nuclide Directory See Table 1	
			pointer word
2	DRX(2,1) = 202	Nuclide #1 PREF RECORD (TRACE Array)	(All records seperated by a pointer word)
			pointer word
3	DRX(2,1)+NPREF+1	Nuclide #1 EST record	
			pointer word
4	DRX(2,1)+NPREF+NEST$_1$+2	Nuclide #1 CAPE record	
			pointer word
5	DRX(2,2)	Nuclide #2 PREF record (TRACE Array)	
			pointer word
6	DRX(2,2)+NPREF+1	Nuclide #2 EST record	
			pointer word
7	DRX(2,2)+NPREF+NEST$_2$+2	Nuclide #2 CAPE record	
			pointer word
.	.	.	.
.	.	.	.
.	.	.	.
			pointer word
3*N-1	DRX(2,N)	Nuclide #N PREF record (TRACE Array)	N < NDIR NDIR = 100

APPENDIX B

Neutrons

			pointer word
3*N	DRX(2,N)+NPREF+1	Nuclide #N	
		EST record	
			pointer word
3*N+1	DRX(2,N)+NPREF+$NEST_N$+2	Nuclide #N	
		CAPE record	

Note: a) NPREF = 40 in the present version of MCNP.

b) $NEST_I$ = the value of NEST for the I^{th} nuclide. NEST = 2*NES+1 where NES is the number of energies in the energy grid for the I^{th} nuclide. The value of NES for each nuclide is stored in the PREF record for each nuclide.

c) In the address of the EST record, DRX(2,i)+NPREF+1, the +1 is added to account for the pointer word between the PREF and EST records. Likewise, the +2 in the CAPE record address, DRX(2,i)+NPREF+$NEST_I$+2, is added to account for the pointer words after the PREF and EST records.

Table 1

Nuclide Directory

Length of record = 2*NDIR = 200

Location	Parameter	Description
1	DRX(1,1)	ZAID number of first nuclide in library
2	DRX(2,1)	absolute location of first nuclide in library file (beginning of IN array)
3	DRX(1,2)	ZAID number of second nuclide
4	DRX(2,2)	location of second nuclide
.	.	.
.	.	.
.	.	.

2*N-1	DRX(1,N)	ZAID number of last (N^{th}) nuclide
2*N	DRX(2,N)	location of N^{th} nuclide
2*N+1	DRX(1,N+1)	0.0
2*N+2	DRX(2,N+1)	0.0
.	.	.
.	.	.
.	.	.
2*NDIR-2	DRX(2,NDIR-1)	0.0
2*NDIR-1	DRX(1,NDIR)	name of this library file version
2*NDIR	DRX(2,NDIR)	date when this library file was created

(The ZAID number is the nuclide identifier name. See Chapter 2 under Materials Specification for details.)

Table 2
PREF Record for a Typical Nuclide in an ACE Format Cross-Section Library

The PREF record is the first NPREF=40 words of the IN array. This record contains pointers for the rest of the data for this nuclide.

Location	Parameter	Description
IN(1)	ZAID	nuclide identifier number
2	AWR	atomic weight ratio
3	NTR	total number of reactions
4	NES	number of energies
5	NR	number of reactions having secondary neutrons (not elastic)
6	ESZ	loc. of energy table ESZ=NPREF+1=41
7	MTR	loc. of ENDF MT's
8	TYR	loc. of reaction types
9	LSIG	loc. of table of cross section locators
10	SIG	loc. of cross sections
11	LAND	loc. of table of cosine distribution locators
12	AND	loc. of cosine tables
13	LDLW	loc. of table of law locators

14	DLW	loc. of law data
15	GPD	loc. of gamma production data
16	LQR	loc. of table of reaction Q's
17	MGPT	multigroup and probability table flag
18	NU	loc. of fission ν data
19	FIS	loc. of total fission cross section
20	END	loc. of last word of this nuclide
21		
22		
23		
24	description of neutron cross sections (BCD Hollerith)	
25	description of neutron cross sections (BCD Hollerith)	
26	description of neutron cross sections (BCD Hollerith)	
27	description of neutron cross sections (BCD Hollerith)	
28	description of neutron cross sections (BCD Hollerith)	
29	description of neutron cross sections (BCD Hollerith)	
30		
31	ENDF MAT number (BCD Hollerith)	
32	description of gamma production data (BCD Hollerith)	
33	description of gamma production data (BCD Hollerith)	
34	description of gamma production data (BCD Hollerith)	
35	description of gamma production data (BCD Hollerith)	
36	description of gamma production data (BCD Hollerith)	
37	description of gamma production data (BCD Hollerith)	
38	description of gamma production data (BCD Hollerith)	
39	description of gamma production data (BCD Hollerith)	
40	date last processed (A10)	

Note: GPD=0 no gamma production data
NU=0 no fission
FIS=0 no total fission cross section; partial reaction cross sections used instead

Table 3
TRACE Array (PREF record) for a Typical Nuclide Stored in MCNP

This array has length NPREF=40 and contains locators for the rest of the data for this nuclide. Note: ITR=NPREF*N

for the N^{th} nuclide.

I	TRACE(ITR+I)	Description
1	ZAID	nuclide identifier number
2	AWR	atomic weight ratio
3	NTR	total number of reactions
4	NES	number of energies
5	NR	number of nonelastic reactions having secondary neutrons
6	-	-
7	MTR	loc. of ENDF MT's
8	TYR	loc. of reaction types
9	LSIG	loc. of table of cross section locators
10	SIG	loc. of cross sections
11	LAND	loc. of table of cosine distribution locators
12	AND	loc. of cosine tables
13	LDLW	loc. of table of law locatos
14	DLW	loc. of law data
15	GPD	loc. of gamma production data. 0 if none
16	LQR	loc. of reaction Q's
17	MGPT	multigroup and probability table flag
18	NU	loc. of fission ν data. 0 if no fission
19	FIS	loc. of total fission cross section. 0 if no fission
20	END	loc. of last word of this data set
21	SEST	loc. of energy table (EST record) relative to XSN(1). 0 if energy table in XSL
22	LEST	loc. of enegy table (EST record) relative to XSL(1). 0 if energy table in XSN
23	LCAPE	loc. of absorption cross section (start of CAPE record) relative to XSL(1)
24	LRXSL	loc. of XSL(1) in LCM-1
25	σ_t	total cross section at energy ERG without thermal effects (as taken from ACE format input data)
26	ERG	energy prior to collision, LAB
27	σ_t	total cross section at energy ERG with thermal effects included
28	IERG	i such that $ES_i \leq ERG < ES_{i+1}$ (ES_i = energy table energy. See ESZ block)
29	RINT	interpolation fraction: $(ERG-ES_i)/(ES_{i+1}-ES_i)$
30	EHD	variable in thermal procedure

31	σ_a	absorption cross section at energy ERG with thermal effects included
32	σ_{el}	elastic cross section at energy ERG with thermal effects included
33	FSIC	used only in K-code
34	IKRN	initial random number for a collision
35	INRN	another initial random number for a collision
36	LFL	thermal range indicator (0 for non thermal)
37	LTNN	thermal collision parameter
38	σ_{fpt}	fission cross section in probability table method
39	—	presently unused
40	—	presently unused

The above locations (LOC.) are all relative to XSL(1) unless otherwise noted.

Table 4
ESZ BLOCK
Total Length = 5*NES+1

Location	Parameter	Description
1. Location in IN Array (ESZ, NES defined in Table 2)		
IN(ESZ)	NES	number of energy grid points
IN(ESZ+1)	ES(I),I=1,NES	energy table
IN(ESZ+NES+1)	σ_t(I),I=1,NES	total cross section
In(ESZ+2*NES+1)	σ_a(I),I=1,NES	total absorption cross section
IN(ESZ+3*NES+1)	σ_{el}(I),I=1,NES	elastic cross section
IN(ESZ+4*NES+1)	H_{ave}(I),I=1,NES	average heating numbers
2. Location in MCNP (SEST, LEST, LCAPE, NES defined in Table 3)		
a. if SEST≠0, LEST=0		
XSN(SEST)	NES	number of energy grid points
XSN(SEST+1)	ES(I),I=1,NES	energy table
XSN(SEST+NES+1)	σ_t(I),I=1,NES	total cross section
b. if SEST=0, LEST≠0		
XSL(LEST)	NES	number of energy grid points
XSL(LEST+1)	ES(I),I=1,NES	energy table
XSL(LEST+NES+1)	σ_t(I),I=NES	total cross section

c. regardless of SEST, LEST values

XSL (LCAPE)	$\sigma_a(I), I=1,NES$	total absorption cross sectionn
XSL(LCAPE+NES)	$\sigma_{el}(I), I=1,NES$	elastic cross section
XSL(LCAPE+2*NES)	$H_{ave}(I), I=1,NES$	average heating numbers

Table 5

NU Block - Fission $\bar{\nu}$ data

XNU=IN(NU) for libraries, = XSL(NU) for MCNP

There are three possibilities for the NU Block:

1. XNU=0 no NU Block
2. XNU>0 either prompt $\bar{\nu}$ or total $\bar{\nu}$ is given. The NU array begins at location IN(KNU) or XSL(KNU) where KNU=NU
3. XNU<0 both prompt $\bar{\nu}$ and total $\bar{\nu}$ are given. The prompt $\bar{\nu}$ NU Array begins at IN(KNU) or XSL(KNU) where KNU=NU+1; the total $\bar{\nu}$ NU Array begins at IN(KNU) or XSL(KNU) where KNU=NU+ABS(XNU)+1.

The NU Array has two forms if it exists:

[Location = IN(IJK) for libraries, XSL(IJK) for MCNP]

a) Polynomial function form of NU Array:

IJK	Parameter	Description
KNU	LNU=1	polynomial function flag
KNU+1	NC	number of coeficients
KNU+2	C(I),I=1,NC	coefficients

$$\bar{\nu}(E) = \sum_{I=1}^{NC} C(I)*E**(I-1) \qquad E \text{ in MeV}$$

b) Tabular data form of NU array

IJK	Parameter	Description
KNU	LNU=2	tabular data flag
KNU+1	NR	number of interpolation regions
KNU+2	NBT(I)I=1,NR	ENDF interpolation parameters
KNU+2+2*NR	INT(I)<I=1,NR	If NR=0, NBT and INT are omitted and linear-linear

		interpolation is used.
KNU+2+2*NR	NE	number of energies
KNU+3+2*NR	E(I),I=1,NE	tabular energy points
KNU+3+2*NR+NE	$\bar{\nu}$(I),I=1,NE	corresponding values of $\bar{\nu}$

Note: For type 19 fission (FIS=0) the NU block follows the ESZ block. For type 18 fission (FIS>0) the NU block is buried in the DLW block. NU and FIS are pointers in the PREF record (Tables 2 and 3)

Table 6
MTR Block - ENDF MT Numbers
Length of block = NTR
Location of data: IN(IJK) in cross-section librries
XSL(IJK) in MCNP

IJK	Parameter	Description
MTR	MT_1	first ENDF reaction available
MTR+1	MT_2	second ENDF reaction available
.	.	.
.	.	.
.	.	.
MTR+NTR-1	MT_{NTR}	last ENDF reaction available

Note: 1. MT_1, MT_2, ... are standard ENDF MT numbers, i.e., MT=16=(n,2n); MT=17=(n,3n); etc.
2. MTR, NTR are defined in the PREF record, Tables 2 and 3

Table 7
LQR Block - Reaction Q-values
Length of block = NTR
Location of data: IN (IJK) in cross-section libraries
XSL(IJK) in MCNP

IJK	Parameter	Description
LQR	Q_1	Q-value of reaction MT_1
LQR+1	Q_2	Q-value of reaction MT_2
.	.	.

.	.	.
.	.	.
LQR+NTR-1	Q_{NTR}	Q-value of reaction MT_{NTR}

Note: LQR and NTR are defined in the PREF record, Tables 2 and 3; MT_i is defined in Table 6.

Table 8

TYR Block - Number of secondary neutrons released per collision for each reaction type

Length of block: NTR
Location of data: IN(IJK) for cross-section libraries
XSL(IJK) for MCNP

IJK	Parameter	Description
TYR	TY_1	neutron release for reaction MT_1
TYR+1	TY_2	neutron release for reaction MT_2
.	.	.
.	.	.
.	.	.
TYR+NTR-1	TY_{NTR}	neutron release for reaction MT_{NTR}

Note: the possible values of TY_i are ±1, ±2, ±3,..., 18, 19, 0. The sign indicates the system for scattering: negative = CM system; positive = LAB system.
Thus if TY_i = -3, then 3 neutrons are released in the CM system for reaction MT_i.
TY_i = 18 indicates type 18 fission. The number of secondary neutrons released is determined from the total fission cross section, the MCNP fission subroutine, and the fission $\bar{\nu}$ data.
TY_i = 19 indicates type 19 fission. The number of secondary neutrons released is determined from the individual fission reaction cross sections (ENDF reactions MT=19, 20, 21), individual secondary energy laws, and the fission $\bar{\nu}$ data.
TY_i=0 indicates absorption (ENDF reactions MT>100); no neutrons are released.

TYR and NTR are defined in the PREF record, Tables 2 and 3
MT_i is defined in the MTR block, Table 6

Table 9
LSIG Block - Reaction cross-section locators
for use in SIG Block (Table 10)

Length of block = NTR
Location of data: IN (IJK) for cross section libraries
XSL(IJK) for MCNP

IJK	Parameter	Description
LSIG	$LOCA_1$	location of reaction MT_1
LSIG+1	$LOCA_2$	location of reaction MT_2
.	.	.
.	.	.
.	.	.
LSIG+NTR-1	$LOCA_{NTR}$	location of reaction MT_{NTR}

Note: All locators are relative to SIG. See the SIG Block. (Table 10). $LOCA_1$ = 1 always.
LSIG and NTR are defined in the PREF record, Tables 2 and 3.
MT_i is defined in the MTR Block, Table 6

Table 10
Reaction Cross Sections

Location of data: IN(IJK) for cross-section libraries
XSL(IJK) for MCNP

IJK	Parameter	ENDF Reaction	XSEC energy	parameter description
SIG	IE_i	MT_i	—	energy grid index
SIG+1	NE_i	MT_i	—	number of consecutive entries
SIG+2	σ_i	MT_i	$ES(IE_i)$	reaction cross section
.	.	.	.	. . .
.	.	.	.	. . .
.	.	.	.	. . .

SIG+1+NE_1	σ_{NE1}	MT_1	ES(IE_1+NE_1-1)	reaction cross section
SIG+$LOCA_2$-1	IE_2	MT_2	—	energy grid index
SIG+$LOCA_2$	NE_2	MT_2	—	number of consecutive entries
SIG+$LOCA_2$+1	σ_1	MT_2	ES(IE_2)	reaction cross section
.	.	.	.	. . .
.	.	.	.	. . .
.	.	.	.	. . .
SIG+$LOCA_2$+NE_2	σ_{NE_2}	MT_2	ES(IE_2+NE_2-1)	reaction cross section
		...etc...		
SIG+$LOCA_{NTR}$-1	IE_{NTR}	MT_{NTR}	—	energy grid index
SOG+$LOCA_{NTR}$	NE_{NTR}	MT_{NTR}	—	number of consecutive entries
SIG+$LOCA_{NTR}$+1	σ_1	MT_{NTR}	ES(IE_{NTR})	reaction cross section
.	.	.	.	. . .
.	.	.	.	. . .
.	.	.	.	. . .
SIG+$LOCA_{NTR}$ +NE_{NTR}	$\sigma_{NE_{NTR}}$	MT_{NTR}	ES(IE_{NTR}-NE_{NTR}-1)	reaction cross section

Note: The values of $LOCA_1$ are given in the LSIG block (Table 9). The enery grid, ES(I), I=1,NES, is given in the ESZ block (Table 4). The reaction numbers, MT_1, are given in the MTR block (Table 6). The values of SIG, NES, and NTR are given in the PREF record (Tables 2 and 3).

Table 11
LAND Block: Cosine Distribution locators
for use in AND Block (Table 12)

Length of block: NR+1
Location of data: IN(IJK) for cross-section libraries
XSL(IJK) for MCNP

IJK	Parameter	Description
LAND	$LOCB_1$=1	location of cosine distribuion for elastic collision
LAND+1	$LOCB_2$	location of cosine distribution for reaction MT_1
.	.	.
.	.	.

LAND+NR	$LOCB_{NR+1}$	location of cosine distribution for reaction MT_{NR}

Note: all locators are relative to AND. If $LOCB_i$=0, there is no cosine distribution given for this reaction and isotropic scattering is assumed in either the LAB or CM system. Choice of LAB or CM system depends upon values for this reaction in the TYR block (Table 8). LAND, AND and NR are defined in the PREF record, Tables 2 and 3. MT_i is defined in the MTR block, Table 6.

Table 12
AND Block - Cosine Distribution Tables

Location of data: IN(IJK) for cross-section libraries
XSL(IJK) for MCNP

IJK	Description
AND+$LOCB_1$	AND Array for elastic collision
AND+$LOCB_2$	AND Array for reaction MT_1
.	
.	
.	
AND+$LOCB_{NR+1}$	AND Array for reaction MT_{NR}

Note:
1. $LOCB_i$ is defined in the LAND block, Table 11;
2. Whereas $LOCB_i$=1, the first word of the AND block, IN(AND) or XSL(AND), is not used;
3. If $LOCB_i$=0, then no AND Array is given and scattering is assumed to be isotropic in either the LAB or CM system. Choice of LAB or CM system depends on the TYR block (Table 8).

The i^{th} AND Array has the form

IJK	Parameter	Description
AND+$LOCB_i$	NE	number of energies tabulated in grid is numoer of energy bins +1

AND+LOCB$_1$+1	E(J),J=1,NE	energy grid for table
AND+LOCB$_1$+NE+1	LC(J),J=1,NE	location of tables associated with energy grid point E(J). Note that if LC(J)=0 then no table is given for energy E(J) and scattering is isotropic in the coordinate system indicated by the TYR Block.
AND+LC$_1$	P(1,K),K=1,33	32 equi-probable cosine bins for scattering at energy E_1. Note that $P_J(1)=-1$ and $P_J(33)=1$ always.
AND+LC$_2$	P(2,K),K=1,33	bins for scattering at energy E_2
.	.	.
.	.	.
AND+LC$_{NE}$	P(NE,K),K=1,33	bins for scattering at energy E_{NE}

Note: MT_1 is defined in the MTR block, Table 6.
AND and NR are defined in the PREF record, Tables 2 and 3.

Table 13
LDLW Block - Law Data locators for use in DLW Block (Table 14)

Length of block = NR
Location of data: IN(IJK) for cross-section libraries
XSL(IJK) for MCNP

IJK	Parameter	Description
LDLW	LOCC$_1$	law data locator for reaction MT_1
LDLW+1	LOCC$_2$	law data locator for reaction MT_2
.	.	.
.	.	.
.	.	.
LDLW+NR-1	LOCC$_{NR}$	law data locator for reaction MT_{NR}

Note: MT_1 is defined in the MTR block, Table 6. LDLW and NR are defined in the PREF record, Tables 2 and 3.

Table 14
DLW Block - Law Data for Each Reaction

Location of data: IN(IJK) for cross-section libraries
XSL(IJK) for MCNP

IJK	Parameter	Description
DLW+LOCC$_I$-1	LNW$_I$	location of next law. If LNW=0, then law LAW$_I$, is used regardless of other circumstances.
DLW+LOCC$_I$	LAW$_I$	name of this law
DLW+LOCC$_I$+1	IDAT$_I$	location of data for this law
DLW+LOCC$_I$+2	NR	number of interpolation regions to define law applicability regime
DLW+LOCC$_I$+3	NBT(I), I=NR	ENDF interpolation parameters.
DLW+LOCC$_I$+3+NR	INT(I), I=1, NR	If NR=0, NBT and INT are omitted and linear-linear interpolation is used.
DLW+LOCC$_I$+3+2*NR	NE	number of energies
DLW+LOCC$_I$+4+2*NR	E(I), I=NE	tabular energy points
DLW+LOCC$_I$+4 +2*NR+NE	P(I), I=1, NE	probability of law validity. If the particle energy, E, is E<E(1) then P(E)=P(1) If E>E(NE) then P(E)=P(NE). If more than 1 law is given, then law LAW$_I$ is used only if ξ<P(E) where ξ is a random number.
DLW+IDAT$_I$-1	LDAT(I), I=1, L	law data array for law LAW$_I$. The length, L, of the law data array, LDAT, is determined from parameters within LDAT. The various law data arrays, LDAT for each law, LAW$_I$, are given in Table 14
DLW+LNW$_1$-1	LNW$_2$	location of next law
DLW+LNW$_1$	LAW$_2$	name of this law
DLW+LNW$_1$+1	IDAT$_2$	location of data for this law
.	.	.
.	.	.
.	.	.

etc.

Note: The locator, $LOCC_i$ is defined in the LDLW Block, Table 13. DLW is defined in the PREF record, Tables 2 and 3.

Table 15
Description of Law Data of Various Laws

The location of the LDAT array is defined in the DLW Block, Table 14

a. LAW_i=1 ENDF Law 1 Tabular Energies Out

Location	Parameter	Description
LDAT(1)	NR	interpotation scheme
LDAT(2)	NBT(I),I=1,NR	between tables of E_{out}
LDAT(2+NR)	INT(I),I=1,NR	If NR=0 or if INT(I) ≠1= histogram, linear-linear interpolation is used
LDAT(2+2*NR)	NE	number of incident energies tabulated
LDAT(3+2*NR)	E_{in}(I),I=1,NE	list of incident energies for which E_{out} is tabulated
LDAT(3+2*NR+NE)	NET	number of outgoing energies in each E_{out} table
LDAT(4+2*NR+NE)	E_{out_1}(I),I=1,NET	E_{out} tables are NET endpoints of NET-1 equally likely energy intervals. Linear-linear interpolation is used between intervals.
	.	
	.	
	.	
	$E_{out_{NE}}$(I),I=1,NET	

b. LAW_i=3 ENDF Law 3 Level Scattering

E_{out} = LDAT(2)*(E-LDAT(1))

For CM system,

APPENDIX B
Neutrons

$$LDAT(1) = \left(\frac{A+1}{A}\right) |Q| \qquad LDAT(2) = \left(\frac{A}{A+1}\right)^2$$

For LAB system,

$$LDAT(1) = \frac{A(A+1)|Q|}{A^2 + 1} \qquad LDAT(2) = \frac{A^2+1}{(A+1)^2}$$

where E = incident particle energy
A = atomic mass number
Q = Q-value

c. LAW_i=5 ENDF Law 5 General Evaporation Spectrum
$E_{out} = X(\xi)*T(E)$ where $X(\xi)$ is a randomly sampled table of X's, and E is the incident energy

Location	Parameter	Description
LDAT(1)	NR	interpolation scheme between T's
LDAT(2)	NBT(I),I=1,NR	
LDAT(2+NR)	INT(I),I=1,NR	
LDAT(2+2*NR)	NE	number of incident energies tabulated
LDAT(3+2*NR)	E(I),I=1,NE	incident energy table
LDAT(3+2*NR+NE)	T(I),I=1,NE	tabulated function of incident energies
LDAT(3+2*NR+2*NE)	NET	number of X's tabulated
LDAT(4+2*NR+2*NE)	X(I),I=1,NET	tabulated probabilistic function

d. LAW_i=7 ENDF Law 7 Simple Maxwell Fission Spectrum
$f(E \rightarrow E_{out}) = SQRT(E_{out}) \exp(-E_{out}/T(E))$
with restriction $0 \leq E_{out} \leq E-U$

The format of the LDAT array is given below with the description of LAW 9

e. LAW_i=9 ENDF Law 9 Evaporation Spectrum

$$f(E \to E_{out}) = SQRT(E_{out}) \exp(-E_{out}/T(E))$$

with restriction $0 \le E_{out} \le E-U$

Location	Parameter	Description
LDAT(1)	NR	interpolation scheme between T's
LDAT(2)	NBT(I),I=1,NR	
LDAT(2+NR)	INT(I),I=1,NR	
LDAT(2+2*NR)	NE	number of incident energies tabulated
LDAT(3+2*NR)	E(I),I=1,NE	incident energy table
LDAT(3+2*NR+NE)	T(I),I=1,NE	tabulated T's
LDAT(3+2*NR+2*NE)	U	restriction energy

f. LAW_i=10 ENDF Law 10 Watt Spectrum

$$f(E \to E_{out}) = C_o \exp(-E_{out}/a) \sinh(b*E_{out})^{1/2}$$

This is sampled by the rejection scheme in LA-5061-MS (R11 pg. 45):

let $k = 1+ab/8$

$L = a(k+(k^2-1)^{1/2})$

$M = K+(k^2-1)^{1/2}-1$

let $X = -\ln \xi_1$ ξ_1, ξ_2 = random numbers [0,1]

$Y = -\ln \xi_2$

accept X if $(Y-M(X-1))^2 \le BLX$

then $E_{out} = L\ X$

LDAT(1)=b; LDAT(2)=M; LDAT(3)=L

g. LAW_i=18 Fission using MCNP fission subroutine. The LDAT Array is simply the NU Block (Table 5)

h. LAW_i=21 UK Law 1 Tabular Energies Out
(This law is the same as LAW 1 except a probability distribution is used rather than an equi-probable mesh.)

Location	Parameter	Description
LDAT(1)	NR	Interpolation parameters
LDAT(2)	NDT(I),I=1,NR	which are not used by MCNP
LDAT(2+NR)	INT(I),I=1,NR	
LDAT(2+2*NR)	NE	number of incident energies tabulated
LDAT(3+2*nr9	E_{in}(I),I=1,NE	list of incident energies for which E_{out} is tabulated
LDAT(3+2*NR+NE)	LOCD(I),I=1,NE	locators of E_{out} tables

Location IJK*	Parameter	Description
DLW+$LOCD_1$-1	NF_1	
DLW+$LOCD_1$	P_1(K),K=1,NF_1	
DLW+$LOCD_1$+NF_1	E_{out_1}(k),K=1,NF_1	if $E_{in_I} \le E < E_{in_{I+1}}$
DLW+$LOCD_2$-1	NF_2	and ξ is a random
DLW+$LOCD_2$	P_2(K),K=1,NF_2	number [0,1] then
DLW+$LOCD_2$+NF_2	E_{out_2}(K),K=1,NF_2	if $P_{IK} \le \xi < P_{I,K+1}$
.	.	
.	.	$E_{out} = E_{out_{I,K}}$
.	.	
DLW+$LOCD_{NE}$-1	NF_{NE}	
DLW+$LOCD_{NE}$	P_{NE}(K),K=1,NF_{NE}	
DLW+$LOCD_{NE}$+NF_2	$E_{out_{NE}}$(K),K=1,NF_{NE}	

*Location = IN(IJK) for cross-section libraries
= XSL(IJK) for MCNP

i. LAW_i = 22 UK Law 2 Tabular linear functions of incident energy out

Location	Parameter	Description
LDAT(1)	NR	Interpolation parameters
LDAT(2)	NBT(I),I=1,NR	which are not used by MCNP
LDAT(2+NR)	INT(I),I=1,NR	
LDAT(2+2*NR)	NE	number of incident energies tabulated

LDAT(3+2*NR)	$E_{in}(I), I=1, NE$	list of inicdent energies for E_{out} tables
LDAT(3+2*NR+NE)	LOCE(I), I=1, NE	locators of E_{out} tables

Location IJK*	Parameter	Description
DLW+$LOCE_1$-1	NF_1	
DLW+$LOCE_1$	$P_1(K), K=1, NF_1$	
DLW+$LOCE_1$+NF_1	$T_1(K), K=1, NF_1$	
DLW+$LOCE_1$+2*NF_1	$C_1(K), K=1, NF_1$	if $E_{in_I} \le E < E_{in_{I+1}}$
DLW+$LOCE_2$-1	NF_2	and ξ is a random
DLW+$LOCE_2$	$P_2(K), K=1, NF_2$	number [0,1] then
DLW+$LOCE_2$+NF_2	$T_2(K), K=1, NF_2$	if $P_{I,K} \le \xi < P_{I,K+1}$
DLW+$LOCE_2$+2*NF_2	$C_2(K), K=1, NF_2$	
.	.	$E_{out} = C_{I,K}*(E-T_{I,K})$
.	.	
.	.	
DLW+$LOCE_2$-1	NF_{NE}	
DLW+$LOCE_{NE}$	$P_{NE}(K), K=1, NF_{NE}$	
DLW+$LOCE_{NE}$+NF_{NE}	$T_{NE}(K\leftarrow, K=1, NF_{NE}$	
DLW+$LOCE_{NE}$+2*NF_{NE}	$C_{NE}(K), K=1, NF_{NE}$	

*Location = IN(IJK) for cross-section libraries
= XSL(IJK) for MCNP

j. LAW_i = 23 UK Law 5
$E_{out} = T_I*E^{1/2}$ where T_I is the same as E_{out} in LAW_i=1 except that there is no interpolation between tables.

k. LAW_i=24 UK Law 6
$E_{out} = T_I*E$ where T_I is the same as E_{out} in LAW_i=1 except that there is no interpolation between tables.

APPENDIX B
Neutrons

Table 16
Gamma Production Block
Length of block = NES+600

Location of data = IN(IJK) for cross-section libraries
= XSL(IJK) for MCNP

IJK	Parameter	Description
GPD	σ_γ(I),I=1,NES	Gamma production cross section
GPD+NES	EG(1,K),K=1,20	20 equally likely outgoing photon energies for incident neutron energy E ≤ EN(1)
GPD+NES+20	EG(2,K),K=1,20	20 equiprobable outgoing photon energies for incident neutron energy E=EN(2)
.	.	.
.	.	.
.	.	.
GPD+NES+580	EG(30,K),K=1,20	20 equiprobable outgoing photon energies for incident neutron energy E ≥ EN(30)

Note: the discrete incident neutron energy array is EN(J),J=1,30: 1.39E-10, 1.52E-7, 4.14E-7, 1.13E-6, 3.06E-6, 8,32E-6, 2.26E-5, 6.14E-5, 1.67E-4, 4.54E-4, 1.235E-3, 3.35E-3, 9.23E-3, 2.48E-2, 6.76E-2, .184, .303, .500, .823, 1.353, 1.738, 2.232, 2.865, 3.68, 6.07, 7.79, 10., 12., 13.5, 15.

GPD and NES are defined in the PREF record, Tables 2 and 3.

Photon Cross Sections

Merging of Cross Sections:

PXSEC is a short merging code that generated the public file of photon cross sections called MCPLIB that is used by MCNP. It is in the LIX file RGSPPR on AC=V6GRP, HD.

An earlier code of TD6 called MCP (see LA-5157-MS) had photon cross sections for Z=1 to Z=94 at energies up to 15 MeV. The current ENDF file of photon cross sections goes up to 100 MeV.

PXSEC does the following:

1) checks the ENDF file to verify that the partial cross sections above 15 MeV add to the total tabulated cross section.
2) Merges the cross sections of the MCP file, up to 15 MeV, with the ENDF file above 15 MeV.
3) Creates a file set directory similar to the neutron directory, and adds it to the merged file.
4) Calculates a heating number at each of the energy mesh points for each element.
5) Prints the cross sections in the format of MCP. The energies and cross sections are printed instead of the natural logs of these numbers, which is what is stored in MCPLIB.

Heating Numbers

The heating numbers are calculated from the average energy deposited from incoherent, photoelectric plus flourescence, and pair production processes. The pair production average energy deposited is deterministic.

The sampling for the average energy deposited from incoherent and flourescent scattering is done using the same code as MCNP, except that the random number selection is done by quadrature. That is, the interval (0,1) is divided into 1000 parts to yield 1000 values of the random number, equally spaced in the interval. The energy deposited is calculated at each of these points, and the average determined. For incoherent scattering, the random number is used to determined the outcoming energy E′ from the incoming energy E, in the Klein Nishina treatment.

The expected energy loss per photon collision has been calculated for MCG gamma cross sections. These cross sections have no coherent data, no incoherent form factors, and no flourescent treatment. A comparison has been made of MCG

versus MCP heating numbers at the MCP energy mesh points. This was for Z=94. In general, the agreement was good except at E=.15 MeV, where the emergent flourescent photon had an energy comparable to the incoming energy.

Definition of Files Used in PXSEC

Files:

BETA34--MCPLIB-input from LCMCPLB HP Ver. 2 (AC=V6GRP). These are the old MCP cross sections from CROS - run through CRSCNV.

BETA2--RGSPXS-ENDF file of photon cross sections. RGSPXS came from the photostore file DLC7EHD (ENDF-102, October, 1975). The file DLC7EHD is in CROS format. It was run through CRSCNV, CROSPL, and UPDATE with options 8,d in order to remove the identifiers. It was saved on photostore as RGSPXS Ver. 1 (V6GRP).

BETA4--print output file. This file is also saved on photostore as RGSPHXS Ver. 3 (V6GRP) so that microfiche listings of the cross sections can easily be made.

BETA35--NMCPLIB-binary cross section output of the merged files. This file is used on photostore as RGSMCP, and is also used as the public file MCPLIB.

Format of Library MCPLIB

Directory:

This is identical in format to the neutron cross section directory. The 200 word directory lists the Z-number of each element, followed by its beginning decimal address. For example, the first four words of the directory are 1., 202, 2., 589. Element 1, hydrogen, has its cross sections starting at location 202. Its length is 589-202, or 387 words.

Pointer Words:

Each element has a nine word pointer array P(I), where

P(1) = the Z-number of the element.
P(2) = the number of energies.
P(3) = the location of the incoherent data.
P(4) = the location of the coherent data.

P(5) = the location of the flourescent data.
P(6) = the length of the flourescent data.
P(7) = the location of the heating numbers.
P(8),P(9) not used at present.

Cross Sections:

Following the nine word pointer block are the partial cross sections and data:

A) - the natural log of the energies.
B) - the natural log of the incoherent cross sections.
C) - the natural log of the coherent cross sections.
D) - the natural log of the photoelectric cross sections.
E) - the natural log of the pair production cross sections.
F) - the incoherent form factors.
G) - the integrated coherent form factors.
H) - the coherent form factors.
I) - the flourescent data (if any).
J) - the heating numbers.

The lengths of blocks A) through E) and J) are given by P(2).
The length of block F) is 21.
The lengths of blocks G) and H) are 55 each.
The length of Block I) is P(6).

Notice that the total cross section is not stored in the file.

Hydrogen has 41 energies and no flourescent data.
Its total length is 9+6*41+21+2*55+1(eor)=387 words.

Comparison of MCPLIB and ENDF-102 data for Z=26.

Total Cross Section:

There is good agreement in the total cross section below 15 MeV. This is not surprising since the data base (Howerton) is the same. There are a few more points in the ENDF file. This file lists only the total and photoelectric cross section at a resonance energy. Consequently, the coherent and incoherent cross sections at this energy would have to be interpolated and added to the file in order to put it into the MCPLIB format.

APPENDIX B
Photons

Incoherent Form Factors:

The form factors of MCPLIB are a subset of the ENDF. It would require only a minor change in the code MCNP to use the ENDF data.

Coherent Form Factors:

MCNP uses the coherent form factors, (actually their squares), directly in the point detector calculation. The code uses a normed integral of these form factors for the scattering. See LA-5157-MS, page 8, for the definition of this integral, $A(Z,V^2)$. The code that originally calculated these factors is missing. A small LTSS code has been written which reads the MCPLIB file for the regular coherent form factors and re-calculates SCOHFF by a trapezoidal rule. This code has as input in a data statement a set of V(I), I=1,55, provided by Cashwell and Everett (Cromer/Hanson coherent scattering V's). This small code could easily be modified to read the ENDF file for the coherent form factors, and make the same calculation.

Some comments on the use of ENDF-102 data entirely:

This file has no data for the seven elements Z=84, 85, 87, 88, 89, 91, and 93. Consequently, the file MCPLIB has data for these elements only to 15 MeV, which is the old MCP data. These data are essentially the Storm/Isreal data from the nuclear data tables 7, 565-681 (1970).

The ENDF file appears to have no flourescent data. The report LA-5240-MS gives the data and the flourescent treatment that is in MCNP.

If ENDF heating numbers were used, they would need to be calculated at each of the energy mesh points of MCPLIB. The quadrature method used in PXSEC to calculate heating is consistent with the code for the scattering treatment in MCNP. However, the code could change. In particular, the latest method of sampling the Klein-Nishina probability distribution, (LA-4663), is not in the MCNP code yet.

APPENDIX C

This Appendix is for the serious updater making extensive changes to the code. The UPDATE examples in Appendix A are for the casual updater and should be adequate for most of his needs, like adding a source subroutine. Beyond that there is not much else we can do because of the large variety of ways the code is altered by users. The serious updater should certainly be familiar with the uses and options is UPDATE, LOD, CHAT, and LIBMAK so he can satisfy his own needs regardless of how we have set things up.

An abbreviated INDEX listing of MCNP will probably be useful to you. Appendix B contains very helpful information if you require and/or alter cross-section related data. In particular, a description of the TRACE array can be found on page 76. Anyone changing LCM should read the following section in this Appendix.

1. DYNAMIC FIELD LENGTH ADJUSTMENT

The less memory a job uses, the better it competes with other jobs in a timesharing computer system. MCNP adjusts its LCM field length from time to time during the early part of a run in order to use only as much memory as it has to. The data arrays are laid out to facilitate dynamic field length adjustment, with the arrays used later placed higher in memory than arrays used earlier.

The major sections of memory used by MCNP are in this order, with their approximate lengths in octal:

Main (level 1) overlay	61000
Level 2 overlays	47000
Misc. small LCM Common Blocks	2000
LCM1 Common Block	165000
LCM2 Common Block	1000000

LCM1 contains the BCD I/O buffers, the FL array, and two small arrays used by the SNAP source. The FL array contains tallies, photon cross sections, some neutron cross sections, the particle bank, and the big cosine table for the SNAP source. Any space not needed for other things in the FL array in a particular problem is used for more neutron cross sections. LCM2 contains only the array XSL, which is used mainly for neutron cross sections but also for some bulky temporary storage for some of the overlays that set up a problem.

The drop file for MCNP is created large enough for the code at its maximum

possible size and is never changed. The LCM field length is set by LOD to the beginning of Blank Common in the level 1 overlay. Immediately after creating the drop file, MCNP increases its LCM field length to the end of the I/O buffers in LCM1, which is sufficient for the needs of the main overaly and of the level 2 overlay IMCN. If the PLOT and VOLUME overlays run, they each set the field length large enough for a temporary array in LCM2. If the XACT overlay runs, it sets the field length to the end of LCM1, or if the SNAP source is used, to the end of a part of LCM2 which is used temporarily for a buffer for the SNAP source file. XACT then increases the field length as it reads the neutron cross sections and needs more room. The field length is not increased for every nuclide but rather in steps of one quarter of LCM2 as necessary. Finally, the transport overlay, MCRUN, sets the field length just long enough to contain all the cross sections. After that, the field length is not changed again.

In a continue-run only the IMCN and MCRUN overlays are involved. IMCN sets the field length long enough to contain all the cross sections, and MCRUN does not change it.

If you modify MCNP, make sure that any LCM Common Blocks you add are loaded before LCM1. This is accomplished by putting the name of the new Common Block before LCM1 in the list of LCM Common Blocks in the file of LOD commands. The name of the new common block must be put in the LCM statement in the SET block. A COMMON statement for the new block must also be included in the SET block. Beware of inadvertently adding an LCM Common Block as a result of calling a new ORDERLIB subroutine. If a new ORDERLIB subroutine declared an LCM Common Block, you must also specify it before LCM1, as was done with WINDBUF, QIRW789, and Q8QXMITL. Incidentally, the present end of LCM1 is quite close to the point where 19-bit addressing is required. If you add much to memory before LCM1, you will need to specify 19-bit addressing by putting parentheses around LCM1 in the LCM statement.

If for some reason you must add an LCM Common Block after LCM2, use the GROSS. option in the execute line message. This option causes the LCM field length to be the same as the length of the drop file. Difficulties with out-of-bounds references will go away, but so will good timesharing in the initiation phases of the run.

2. MCNP TALLY STRUCTURE

MCNP employs a variably dimensioned tallying scheme. Bin limits for all tallies are stored in array P; the actual tallies are stored in array FL. Data for

the variably dimensioned arrays P and FL are located by pointers which are found in several fixed dimensioned arrays, which are described in Table 1. Most other tally arrays and parameters are also described in Table 1.

The FL array may be subdivided into three blocks as illustrated in Table 2. The first block, BLOCK 1, contains tally information for the current source particle history only. After each source particle history is ended, the data of BLOCK1 are accumulated in BLOCK2 and the squares of the data in BLOCK1 are accumulated in BLOCK3. When it comes time to print the tallies, the desired currents, fluxes, etc. are simply the data in BLOCK2 divided by the number of source particle histories. The desired error estimates are readily computed from the data in BLOCK2 and BLOCK3. The way to find data in the FL array for various tally types is described in Table 3.

Tallies are manipulated by eleven subroutines in MCNP; these routines are described in Table 4. Although the variably dimensioned tallying scheme of MCNP may appear confusing, it permits storage of tallies in the minimum possible space.

Table 1

Summary of Tally Information and Pointer Arrays

Arrays	Description
AREA(i), i=1,I1	Surface areas for tallies 3 and 13, I1 = g*m*n where g, m and n are the number of entries on the CYL, CPHI, and F3 or F13 cards
ANORP(i), i=16	Hollerith information for tallies 1-16
ALP(i), i=1,3	Hollerith labeling information
DETN(i,j)	Detector parameters for detector i
DETP(i,j)	j=1/2/3/4 = x/y/z/$\frac{2}{3}$ r_o^3

APPENDIX C
Tally Structure

IDRECT	Direct contribution to detector flag, 2 if turned on; 1 otherwise
IFL(m)	IFL(m)+2**(I-1) m=problem cell number of cell which is flagged for tally type I
ICSTP(m)	Problem cell/surface number read from Fn tally card. m=ITPNT(n,1) to ITPNT(n,2)
ITPNT(n,m)	Address in ICSTP array for tally n; m=1/2=starting/ending address
ITAL3(i,j)	Surfaces and cells for tallies 3 and 13
IRES(i)	Locator of RESn card in P array
ISORC(i)	Flag for tally type i in set routine ITALLY data statement.
IPNT(j,k,l)	Pointer for bin limits in P array. j=tally type (1 through 16) k=1/2=starting/ending address l=1/2/3/4=energybins/time bins/cosine or CPHI bins/CYL bins
JFL(m)= JFL(m)+2**(i-1)	m=problem cell number of cell which is flagged for tally type i
LCB(i)	Location of tally data in FL array. i=IABS(LTAL(n))-1+J where n=tally type and J=number of surface/cell/detector

of interest. Example, suppose LTAL(1)=10, the F1 tally card is F1 2 4 and there are 20 surfaces in the problem. Then LCB[(i),i=10,29] = 0,LOC_2,0, LOC_4,0,0,...0, where LOC_2 is the location of the surface 4 data.

LTAL(n) — Location of Fn tally in LCB block. For *Fn tally, LTAL(n) < 0.

LTAL2(n) — Number of energy-time arrays for tally type n

LGTH(n,N) — Length of tally blocks for tally type n. If N_1 = number of energy bins, N_2 = number of time bins, N_3 = number of cosine bins. For n = 3 or 13, N_3 = number of CPHI bins. For n = 5 or 15, N_3 = 1/2 = no direct tallies/direct tallies. For n = 4, 5, 6, 14, 15, N_3 = 1/2 = no flagging flagging

Then LGH(n,1) = N_1+1

LGTH(n,2) = $(N_1+1)*(N_2+1)$

LGTH(N,3) = $n_3*(N_2+1)*(N_1+1)$

LGTH(n,4) = $N_4*N_3*(N_2+1)*(N_1+1)$

MXF — length of block in FL tally

MXF2 — 2*MXF

MXF3 — 3*MXF

NDETN — Number of neutron detectors

NDETP — Number of photon detectors

NTAL3 — Number of tally 3 or tally 13 surfaces

NTAL1 — First tally type used

NTAL2 — Last tally type used

APPENDIX C
Tally Structure

PILC(I,1) PILC(I,2)	Entries on PDETN and PDETP cards, probabilities of detector contribution
NDCN	Number of neutron cells contributing to detectors
P(L),L=I1,I2	Array containing tally bin limits (See Table 4) I1=IPNT(I,1,k); I2=IPNT(I,2,k) (see definition of IPNT)
STRT(K)	K=1/2/3 = lower bin limit for energy/time/cosine
TP(I)	Array of tallies and variances to be printed on a given output line
VECT(i),i=1,9	Vectors for tallies 3,13

Table 2

Overall Layout of the FL Array

Location	Parameter	Description	Comment
FL(1) . . FL(MXF)	x_i	all tallies for source particle history i	BLOCK1
FL(MXF+1) . . . . FL(MXF2)	$\sum_i X_i$	cumulative tallies	BLOCK2 at the completion of each source particle history the data of BLOCK1 are added to BLOCK2
FL(MXF2+1)			BLOCK3

.	$\sum_i X_i^2$	cumulative squares of tallies	at the completion of each source particle history all data in BLOCK1 are squared and added to BLOCK3
.			
.			
.			
FL(MXF3)			

MXF = total number of tallies for each source particle history
MXF2 = 2*MXF
MXF3 = 3*MXF

Throughout each source particle history tallies are accumulated in BLOCK1. The fluxes, currents, etc. printed by MCNP are simply the data of BLOCK2 divided by the number of source particles, NPS. The data in BLOCK3 is used to compute variances (statistical error).

Example:

Let x_n be a flux tally for source history n. Then after N source histories, the flux printed by MCNP is simply

$$\frac{1}{N}\sum_{n=1}^{N} X_n$$

and the variance is

$$\left[\frac{\sum_{n=1}^{N} X_n^2}{\left(\sum_{n=1}^{N} X_n\right)^2} - \frac{1}{N}\right]^{\frac{1}{2}}$$

APPENDIX C
Tally Structure

Table 3
Layout of the FL Array for Various Tally Types

Tallies 1 and 11: current across a surface

FL(IA+IB+(IC−1)*IJ+J1*(J−1)+I)
currents in cosine bin IC, time bin J and energy bin I

Tallies 2 and 12: flux integrated over surface

FL(IA+IB+J1*(J−1)+I)
fluxes in time bin J and energy bin I

Tallies 3 and 13: material heating at a surface

FL(IA+IC+K1*(k−1)+(IE−1)*IJ+J1*(J−1)+I)
flux surface heating in time bin J, energy bin I, cosine bin IE, CYL card bin K

Tallies 5, 5a, 15, and 15a: detectors

FL(IA+IG+IJ*(L−1)+J1*(J−1)+I)
flux in time bin J and energy bin I. L=1 for total contribution; L=2 for direct contribution to detectors

Tallies 4, 6, 7, 14, 16: track length in cell

FL(IA+ID+IJ*(L−1)+J1*(J−1)+I)
flux in time bin J and energy bin I.
L=1 if this tally is not flagged;
L=2 if this tally is flagged

Definition of Pointers

IA = 0 tallies for current source particle; see BLOCK1, Table 2.
= MXF accumulated tallies: BLOCK2, Table 2.
= MXF2 accumulated squares of tallies: BLOCK3, Table 2.

IB = LCB(LTAB+JJ) = location of tally information for tally type n and a specified surface on the Fn card.

Example: for surface tally card Fn s_1

LTAB = LTAL(n)−1

JJ = program surface number corresponding to problem surface number s_1. For *Fn s_1

LTAB = −LTAL(n)−1

For LTAL and LCB, see Table 1.

IG = LCB(LTAB+JJ) = location of tally information for JJ^{th} detector. LTAB=IABS(LTAL(n))−1 for n=5 or 15

IC = LCB(LTAB+JJ) = location of tally information for JJ^{th} surface specified on F3 or F13 card. LTAB=IABS(LTAL(n))−1 for n=3 or 13.

ID = LCB(LTAB+JJ) = location of tally information for tally type n and program cell JJ.

For cell tally card:

Fn c_1 c_2 c_3

LTAB=IABS(LTAL(n))−1 and JJ=program number of cell corresponding to problem cells c_1, c_2, or c_3.

J1 = N_E+1 where N_1 is the number of energy bins on the En card

I = energy bin number. 1≤I≤J1. If I=J1, this bin contains the sum over bins I, I=1,N_E

IJ = $(N_E+1)*(N_T+1)$ where N_T is the number of time bins on the T_N card.

J = time bin number. 1≤J≤N_T+1. If J=N_T+1 this bin contains the sum over bins J, J=1,N_T

IC = cosine bin for tallies 1 and 11. 1≤IC≤N_C where N_C is the number of cosine bins on the C1 or C11 cards.

IE = cosine bin for tallies 3 and 13. 1≤IC≤N_{C3}

where N_{C3} is the number of cosine bins on the CPHI card

Table 4
MCNP Tally Subroutines

Routine		Function
IMCN Overlay		
a.	RDPROB	reads tally specifications
b.	ITALLY	defines pointers and arrays for storing tallies
c.	TALLYPR	prints a summary of tally specifications
MCRUN Overlay		
a.	TALLYC	updates cell tallies for the current source particle every time a history moves in a cell
b.	TALLYS	updates surface tallies for the current source particle every time a surface is reached
c.	BIN	determines the correct bin in which to count a tally
d.	TALLYP	prints out block of tallies for all histories run so far
e.	PRTET	prints tallies in a time-energy array
f.	TALLYDN	tally neutron contribution to detectors
g.	TALLYDP	tally photon contribution to detectors
h.	TALSHFT	add contribution of source particle to tally

APPENDIX D

PLOT OVERLAY

The PLOT overlay of MCNP has two very useful functions: (1) it can be used to plot the problem geometry specified in the INP file and (2) it can be used to rotate and/or translate geometry in a coordinate system. The first feature is invaluable for debugging geometries, and the second feature can greatly simplify geometry setup. For example, if an ellipse is needed somewhere off the coordinate system axis and canted at 35°, much simpler than determining the surface coefficients for that ellipse would be to calculate the coefficients for an ellipse centered at the coordinate system origin and parallel to an axis. Then the MOVE feature of PLOT could be used to move the simple ellipse where you want it, the new surface coefficients and cards being determined automatically for you. Not only does MOVE handle individual surfaces, but it will transform complete geometries.

INPUT – General Notations

PLOT requires two sources of input:

(1) an MCNP initiation–run file (RUNTPE)
(2) control directives and data.

The control directives are input in the form of KEY Words, and data are input via LRLTRAN NAMELIST. These may be input to PLOT from a terminal (TTY) or may be input from an LTSS file. In either case the input lines are the same.

If you are not using a graphics TTY (such as a Tektronix) you should use the NOTEK. option on the MCNP execution line. The plots will be put in a local film file. If you are using a graphics terminal you will be prompted for box number, line speed (it's 960 now), and terminal type.

PLOT running under ORDER assumes the job is also running under BATCH. It then expects the PLOT control directives to be on input "cards" at the end of the MCNP INP input file – after the INP blank line terminator. The film file that is created is automatically given to the system for processing. If any card of the PLOT input is TTY, then all remaining input is done interactively at a terminal.

APPENDIX D
Plot

KEY WORDS

Key words are used to specify plotter tasks and to control code operation. The key words are:

```
VIEW  ZOOM  MOVE  DRAW
INPUT TTY   WHAT  END
```

Key words are entered into the first 10 spaces of a line. A key word may be spelled out entirely or its first character may be used; e.g. VIEW or V. Only the key words VIEW, ZOOM, and MOVE have an associated data list. No data should be entered on the key word line and only one data list may follow a key word. If input is made from a terminal (TTY), a prompt (a list of appropriate key words) will be given whenever PLOT expects another key word.

The following illustrates the use of key words and data lines:

```
V                              keyword: VIEW
PX = 0.   EX = 150.   A        data line
VIEW                           key word: VIEW
PY = 10.  A                    data line
Z                              keyword: ZOOM
F = 4.  A                      data line
E                              key word: END
```

Data Lists

Data related to a key word are entered separately on the lines that follow the key word. If input is from a terminal (TTY), a prompt will be given displaying the data list variables. Allowed abbreviations for variable names appear in parenthesis following the variable name. For example, BASIS (BA) indicates that BASIS or BA may be used to input values to BASIS.

Default Values for Data

As a general rule, data values previously defined are the default values for subsequent data entries. An exception occurs in the use of PX, PY, or PZ for VIEW. Current values may be examined by using WHAT.

Data Lines

Data lists are input via LRLTRAN NAMELIST. Some general rules concerning LRLTRAN NAMELIST follows:

(1) Variables are defined by equating the variable name to the value, for instance PX=44. defines PX. Dimensioned variables may be input be equating the variable name to a list of values, for instance BA = 1. 0. 0. will set BA(1) = 1., BA(2) = 0., and BA(3) = 0. Individual elements of an array may be defined by use of subscripts, for instance with BA(7) = .89 defines the seventh element of BA.

(2) Blanks are delimiters (they separate items); do not use commas.

(3) Real variables must be input with decimal points.

(4) Integer variables must be input without decimal points.

(5) A data list is terminated by a variable name not in the data list. Special characters are not allowed. Note: since the character *A* is not a variable of any of the data lists, it can be used as a terminator for all of the lists. Also note that by this rule a misspelled variable name is a data list terminator and trouble generally follows.

(6) Any subset of a data list, in any order, may be input.

Detection of Data Errors in PLOT

Data errors detected by PLOT (not namelist errors) will cause the input device to be the TTY. In case an input file was being read, reading the input file may be resumed by use of the key word INPUT.

YES OR NO REPLIES

Various queries from PLOT are answered with YES or NO. In such cases the prompt YES(Y) or NO(N) ( ) is given. This means that Y is accepted as YES and that N or the return key is accepted as NO.

APPENDIX D
Plot

KEY WORD FUNCTIONS AND DATA LISTS

Key word: VIEW

Data list: PX, PY, PZ, EXTENT (EX), ORIGIN (OR), BASIS (BA)

The VIEW option directs PLOT to plot the intersection of a rectangular plane section, a "window," with the MCNP cell boundaries; that is, an arbitrary cross-sectional view contained within a specified rectangle may be plotted.

The rectangular cross-sectional view in MCNP (x,y,z) geometry space is defined by giving the center of the window, the orientation of the window and by giving the extent of the window along each of its axes about the center. The orientation is specified by two vectors, one along each of the window axes.

This information is related to the input variables as follows:

ORIGIN(I=1,3) = (x,y,z) coordinates of the center of the window

BASIS (I=1,3) = (x,y,z) components of a vector along the horizontal axis of the window

BASIS (I=4,6) = (x,y,z) components of a vector along the vertical axis of the window

EXTENT(I=1,2) = the size of the window about ORIGIN along the horizontal and vertical axes. If only one value is input to EXTENT, then EXTENT(2) is taken to be the same as EXTENT(1)

The words *horizontal* and *vertical* refer to the plot window and not to (x,y,z) space. The window may have any orientation in the (x,y,z) geometry space.

USE OF THE INPUT VARIABLES PX, PY, PZ

These input variables provide an abbreviated way to define BASIS and ORIGIN for special planes:

PX=C defines the plane X=C {the plane parallel to the (y,z) plane through the point (C,0.,0.)} and directs PLOT to define

BASIS = 0. 1. 0. 0. 0. 1. and

ORIGIN = C 0. 0.

PY=C defines the plane Y=C {the plane parallel to the (x,z) plane through the point (0.,C,0.)} and directs PLOT to define

BASIS = 1. 0. 0. 0. 0. 1. and
ORIGIN = 0. C 0.

PZ=C defines the plane Z=C {the plane parallel to the (x,y) plane through the point (0.,0.,C)} and directs PLOT to define

BASIS = 0. 1. 0. 1. 0. 0. and
ORIGIN = 0. 0. C

The default definition of the zero components of ORIGIN may be overridden by explicitly entering ORIGIN on the data line. For example, the data line *PX=10. OR(2)=6. -8. EX=40. A* defines a window in the plane x=10 about the point (10, 6, -8).

Note: since only one plane window may be defined for each VIEW, only one from the above should be used at a time. With the use of PX, PY, or PZ, BASIS should not be used.

For each VIEW, two plots are produced, one with surface labels and one without surface labels. The extremes of the plot frame are not necessarily the values of EXTENT on input. The PLOT overlay fills the plot frame with the figure drawn, and the values of EXTENT listed on the plot frame are the coordinates of the corner tabs. The set of points from which the plots are drawn is retained until another VIEW line is read. The saving of these points enables the use of the ZOOM option (which is described later).

Comments on input:

EXTENT must always be defined, whether using a PX, PY, or PZ or using BASIS. EXTENT should not be made a lot bigger than is necessary.

ORIGIN is best placed in the middle of the region of the geometry to be plotted. The plot produced will be centered about ORIGIN.

Expanded views can be obtained with the VIEW option by placing ORIGIN in the center of interest and by making EXTENT small. This method of

course creates a new view. Expanded views made by using the ZOOM option are obtained from the existing points from the last VIEW.

Key word: ZOOM
Data list: FACTOR (F), CENTER (CE), THETA

The ZOOM option directs PLOT to plot an expanded view from the plotting points of the last VIEW plot. This option is a more efficient method of obtaining expanded views than is the method of using VIEW since the cell bounding lines are not recomputed. However, under extreme magnification, curves may be too rough and VIEW should be used. Any number of ZOOM's may follow a VIEW, each of the ZOOM plots being taken from the last VIEW. There is no ZOOM on a ZOOM plot.

The input variables define the ZOOM relative to the VIEW as follows:

FACTOR	= linear magnification
CENTER (I=1,2)	= coordinates, relative to the center of the VIEW plot, for the midpoint of the ZOOM plot
THETA	= rotation (in degrees, counterclockwise rotation being positive) of the ZOOM plot.

On the ZOOM plot, the titles ORIGIN and EXTENT correctly identify the center of the frame and the size of the frame. The values of the variables ORIGIN and EXTENT are not changed by ZOOM.

Note: since CENTER (CE) is defined relative to the VIEW frame, CE should be defined so that

$$-EXTENT(1) \leq CE(1) \leq EXTENT(1)$$
$$-EXTENT(2) \leq CE(2) \leq EXTENT(2)$$

Key word: MOVE
Data list: MAP, BASIS (BA), ORIGIN (OR)

The MOVE option does no plotting. MOVE produces new surface cards (in the LTSS file SURFACE) for a transformation of the MCNP problem geometry space. The transformations allowed are mappings between the original geometry space and another Cartesian coordinate system. No scaling of coordinates is allowed.

For the purpose of describing the input of MOVE, consider two coordinate systems, namely the (x,y,z) system and the (xp,yp,zp) system. The MCNP problem geometry may be considered to be in either of the systems. The following variables describe the relationship between the two systems as follows:

MAP = 1	The input surfaces are in the (x,y,z) system and the output surfaces are in the (xp,yp,zp) system
MAP = -1	The input surfaces are in the (xp,yp,zp) system and the output surfaces are in the (x,y,z) system
ORIGIN (I=1,3)	(x,y,z) coordinates of the (xp,yp,zp) origin
BASIS (I=1,3)	(x,y,z) components of a vector along the xp-axis
BASIS (I=4,6)	(x,y,z) components of a vector along the yp-axis
BASIS (I=7,9)	(x,y,z) components of a vector along the zp-axis

The two cases MAP = 1 and MAP = -1 are included for convenience since either the transformation or the inverse transformation may be known from direct geometric considerations.

Note: the input variables to MOVE are sufficient to define all rotations and translations as well as coordinate swapping.

Key word: DRAW
Data list: none

The key word DRAW directs PLOT to redraw the last plot frame displayed. The last plot frame may have been either a VIEW or a ZOOM. Displays on a Tektronix terminal are sometimes clobbered by operator messages or by line noise and can be redrawn with DRAW most efficiently.

Key Word: INPUT
Data list: none

The key word INPUT directs PLOT to resume reading the input file defined on the MCNP execution line.

Key word: TTY
Data list: none

The key word TTY directs PLOT to read input from the terminal keyboard. It may be placed in an input file to cause PLOT to switch to TTY input.

Key word: WHAT
Data list: none

APPENDIX D
Plot

The key word WHAT directs PLOT to list current values of all input variables as well as the message from the MCNP execution line.

Note: in general, the default values for input variables are those values previously defined. WHAT may be used to inspect those values.

Key word: END
Data list: none
The key word END terminates the PLOT overlay.

EXAMPLES

Consider a golf ball sitting on a tee. The geometry consists of a sphere, a cone, a plane and two ambiquity planes. The sphere has a diameter of 1 43/64 inch and is centered at (0,2,1.55). The cone has its axis parallel to the z-axis and has its vertex at (0,2,-.75). The plane is the plane z=0. The MCNP input deck follows:

```
a sphere atop a cone which penetrates a plane
1     0   -1,2,4,5
2     0   -2,3,4   1,1   -5   4
3     0   2,2   -3,4   4
4     0    2,2   1,1   3   -4
5     0    1,1   5

1     s    0   2   1.55   .8359375
2     k/z  0   2   -.75   .01733
3     pz   0
4     pz   -.75
5     pz   2

i     0  0  0  0  0
```

The following input lines to PLOT illustrate the use of VIEW, ZOOM and END.

```
V
PX=0.   EX=4.   A
Z
F=1.5   CE=2. 0.   A
V
```

```
PY=2.   EX=3.   A
Z
F=1.75   CE=0. 1.55   A
V
BA=1. 0. 0. 0. 1. 3.5   OR=0. 2. 0.   EX=2.5   A
E
```

If you try this for yourself, five plots will be drawn, three will be from the VIEW command:

(1) The first VIEW produces a plot in the plane x=0 about the point (0,0,0) and consequently the sphere and cone are to the right of center. The following ZOOM produces an enlarged plot about the point (2,0) relative to the VIEW frame.

(2) The second VIEW produces a plot in the plane y=2 about the point (0,2,0) which is on the axis of the cone. The following ZOOM produces a plot about the point (0, 1.55) relative to the VIEW, a point at the center of the sphere.

(3) The third VIEW produces a plot in the plane $(x,y,z) = (0,2,0) + s(1,0,0) + t(0, 1/n, 3.5/n)$ where $n = \mathrm{sqrt}(1^2 + 3.5)$ and where (s,t) are coordinates of the plot plane. Note here that the BASIS vectors need not be unit vectors.

A second example involves MOVE. Given a plane containing the point (−234.57, 609.6, −1070.0) that is rotated 22.5° off the y−axis toward the z−axis. The problem is to find the equation of the plane (the plane is perpendicular to the yz−plane). The solution is to set up a new coordinate system with origin at (−234.57, 609.6, −1070.0) that is rotated 22.5° so that the plane is along the y−axis. Then MOVE is used to transform one coordinate system to the other. The INP file for the geometry follows:

```
transformation of a plane
1     0   1,2   -2,3
2     0  -2,3   -1,1
3     0   2,2,1   -3,4
4     0   3,3

1     pz 0
2     so 500
```

```
3     so 1000

i     0  0  0  0
```

The surface cards are in the rotated coordinate system. The spheres have been included for easy comparison before and after the MOVE. The keyword MOVE (or just M) is used and then the following input is typed in:

```
map=-1 or=-234.57 609.6 -1070.  ba=1. 0. 0. 0. .9238795326 .3826834325
0. -.3826834325 .9238795326 a
```

Note that lines are continued by not typing the "a" terminator until the end of the MOVE input. When the return key is hit in PLOT, the input line just typed is immediately displayed (if you are on a TTY terminal), so give PLOT time to do this before you resume the input. In this example, $\sin(22.5°) = .3826834325$ and $\cos(22.5°) = .9238795326$.

The new surface cards are printed at the terminal and also put in a local file SURFACE. The new surface cards are used to replace their pre-transformed equivalents. Using the new cards in this example, the input file containing the plane with the correct specifications is:

```
transformed plane
1     0  1,2  -2,3
2     0   -2,3  -1,1
3     0  2,2,1  -3,4
4     0  3,3

1     p  0  -.382683432   .923879532   -1221.83492
2     s    -234.57   609.6   -1070   500
3     s    -234.57   609.6   -1070   1000

i     0  0  0  0
```

APPENDIX E

VOLUME AND AREA CALCULATION

The volume option V of MCNP (see page 49) may be used to calculate the volumes of those problem cells whose boundary surfaces are symmetric with respect to the y-axis and whose domains do not consist of disjoint regions.

An alternate method is described here which will stochastically compute (ray tracing) both the volumes of cells which do not meet these criteria and also the areas of any of the problem surfaces (or subset of surfaces as might be specified for an F3 tally). The SRC3 source is used for this method, along with tallies F2, F3, or F4. The procedure is to run MCNP with this method using the IXR options. After the volumes and areas of interest are determined, they are added to the INP file on the VOL and AREA cards for a subsequent run with MCNP.

The run is made in Mode 0. The energy of the neutrons is not important and it is easiest to use a monoenergetic source (see page 18). The SRC3 source is started on a sphere with radius large enough to encompass the geometry of interest, and all the cells are entered as voids. The SRC3 source is specified with J set to the source surface problem number, M (the weight multiplier) set equal to one, and p and ν as default values of 0.5 and 0, respectively. The importance of all cells should be unity except for the cell beyond the source sphere which must have zero importance.

The volumes of cells of interest are obtained in a straightforward manner using the F4 track length estimator tally. The expected value of the track length estimator for a cell is

$$E(T) = \int_V \phi(F)\, dV$$

where $\phi(r)$ is the scalar flux and V is the volume of the cell. By setting densities equal to zero and M=1.0 on the SRC3 card, a uniform isotropic flux of unit strength (n/cm^2) is obtained inside the sphere, and the value of the tally is just the cell volume.

Cell areas can be obtained in a similar manner with the flux tallies

using the surface crossing estimator, F2. The F3 tally (with AREA entries set equal to 1) can be used to get areas of subsets of surfaces. This is often the easiest way to obtain values for the AREA card to be used in a later F3 tally of surface heating.

The above methods will generally give good results for volumes and areas in a small amount of running time. If the cells or sub-surfaces are very small, you may want to use a method which can remove the cell(s) of interest from the overall geometry and run in an isolated manner. The source sphere can then be reduced in size to increase the efficiency of the calculation. Bob Schrandt has an UPDATE patch which will allow you to do this.

APPENDIX F

MCNP CROSS-SECTION LIBRARIES

I. SUMMARY

Discussion of the Monte Carlo cross-section libraries is given in three parts: (1) description of the available libraries, (2) major details of the nuclear data and reactions, and (3) available visual information to aid the user in cross-section selection. A list of the library contents then follows.

II. LIBRARY DESCRIPTION

Continuous energy or pointwise neutron data for use in Monte Carlo calculations are available in five separate libraries. Each nuclide evaluation in the libraries has a unique identifier number in each of the libraries. Photon cross-section data are available in only one library, MCPLIB.

The Recommended Monte Carlo Cross Section (RMCCS) library contains cross sections for isotopes and elements which are, in our opinion, the best (and usually the most current) evaluations of nuclear data that have been processed. The RMCCS library has one, and only one, evaluation of each nuclide listed. With sufficient core storage the RMCCS library can handle most requirements that a user may need. On the other hand the Alternate Monte Carlo Cross Section (AMCCS) library contains different evaluations of many of the isotopes on the RMCCS library. AMCCS also contains cross-section information at different temperatures, and some evaluations have no photon production data. Cross-section needs for most problems can be satisfied using the RMCCS and AMCCS libraries. For the user who requires special or archival cross-section information two other libraries are available: XMCCS and UMCCS. Nuclides in these libraries may have more than one evaluation and should be used cautiously.

A fifth library, the Discrete Reaction Monte Carlo Cross Section (DRMCCS) library repeats much of the RMCCS and AMCCS data and some of the XMCCS and UMCCS data in "multigroup" form. The 240-group treatment applies only to neutron reaction cross sections; secondary angular and energy distributions are identical to those in the continuous energy RMCCS, AMCCS,

XMCCS and UMCCS libraries. DRMCCS cross sections are accessed whenever a DRXS card is used (see page 31). A great advantage of using the discrete cross sections is that computer cross-section storage requirements can be reduced (by a factor of two on the average) so that a timesharing environment is enhanced.

Each nuclide in the above libraries is identified by a unique nuclide identifier number called the ZAID number. See page 29 for details concerning the ZAID number. If a nuclide's ZAID number is listed on the DRXS card, then the cross sections are read off the DRMCCS library rather than off one of the continuous energy libraries. An index at the end of this Appendix gives the cross-section evaluations and libraries for each ZAID number. The list indicates which nuclides of the RMCCS, AMCCS, XMCCS, and UMCCS libraries have discrete reaction counterparts on the DRMCCS library file.

Finally, the user may use any other library file of his own simply by setting SPAREC1 or SPARED1 = *file name* on the MCNP execution line (see page 47).

III. NUCLEAR DATA AND REACTIONS

Cross-section information in the Monte Carlo library comes principally from three sources: Evaluated Nuclear Data File (ENDF/B), the Evaluated Nuclear Data Library (ENDL), and the Atomic Weapons Research Establishment (AWRE) library. The ENDF/B evaluation is a national cross-section effort that has contributors from the thermal and fast nuclear reactor programs, the controlled thermonuclear reactor program, and the weapons program. ENDF/B evaluations change every few years. The ENDL evaluations come from Lawrence Livermore Laboratory and generally represent efforts connected with the weapons program at that laboratory. A much older evaluation is the AWRE library which comes from Great Britain. The AWRE library is no longer updated, since the British now use ENDF/B; it is kept in the Monte Carlo library for reproduceability of old problems.

MCNP has a continuous energy treatment of nuclear data--even through the resonance region. All the reactions available in a particular data set, such as ENDF/B, are accounted for in MCNP.

The actual cross-section libraries used by MCNP are binary files in the "ACE" (A Compact ENDF) format. The differences between ENDF/B data (our latest source of cross sections) and the more compact binary ACE data are (1) all interpolation is linear-linear, (2) resonance cross sections

are represented as pointwise data which is Doppler broadened to a specific temperature, (3) pointwise heating numbers are added, (4) secondary angular distribution data are stored in tabular form as equally probable cosines of the scattering angle, (5) secondary neutron energy distribution data are described in any of ten different ways prescribed by distribution laws, (6) scattering is assumed to be isotropic if no angular data are provided for an inelastic collision, (7) fission $\bar{\nu}$'s are represented either in tabular form or as polynomial functions, and (8) photon production data are added in tabular form.

Processing of ENDF/B-formatted data into ACE format is usually done with either the MINX/ETOPL/MCENDF code sequence or with the NJOY-MCNJOY code.

The neutron cross sections and angular distributions for the Monte Carlo library have been generated from the ENDF/B, ENDL and AWRE libraries under the constraint that the resulting library is accurate with linear-linear interpolation to within a relative error E_1 for the cross sections and E_2 for the angular distributions. Linear interpolation is used between energy points with a few hundred to several thousand tabulated energy points, depending on the isotope. Cross sections are added at a sufficient number of points to insure that the linear-linear constraint reproduces the original cross-section evaluation to within E_1. Furthermore, cross sections for all reactions are given in the same energy grid. This faithful preservation of the ENDF/B ENDL and AWRE libraries has the advantage that the Monte Carlo cross-section libraries can be used with confidence for general applications throughout the laboratory. It suffers from the disadvantage that the cross-section sets are much larger than necessary for some applications.

The angular distributions for secondary neutrons from inelastic reactions are stored on a coarse energy grid. These distributions are obtained from the laws prescribed in the particular cross-section evaluations and are preserved in the Monte Carlo libraries. Linear interpolation from data given at discrete energies yields the secondary energy distribution for an incident neutron energy.

The fission reaction can be treated with one of two methods. If the total fission cross section alone is given, then that cross section is used with assumed behavior for the breakup into (n,f), (n,n')f, and (n,2n)f. However, if the reactions (n,f), (n,n')f and (n,2n)f are explicitly tabulated, they are used. In both cases the associated angular, neutron/fission and secondary energy distributions of the END/B, ENDL and

AWRE libraries are preserved and are sampled continuously in the Monte Carlo codes.

Photon production cross-section data (as determined from secondary energy distribution information) are included in the library for some isotopes. These production cross sections are given on the same energy grid as the total cross section. The energy spectra for photons produced by neutron interactions are given in terms of 20 equally probable photon energies for each of 30 incident neutron energy groups. The photons are assumed to be born isotropic in angular distribution.

When a neutron collides with an isotope, heat (energy) is deposited locally by recoil of charged particles and residual nuclei. The average local heating number is tabulated in the cross-section file at each energy point. The total particle heating is the sum of the local heating and the total photon energy deposited in each cell after the photon has been properly transported. If there are no photon production data for the isotope, the photon energy is included in the local heating number.

At this writing, only one set of photon cross sections (MCPLIB) is available for use in MCNP. In this file, the natural logs of the total and partial cross sections are tabulated as a function of the natural log of the incident photon energy. Average photon heating is also tabulated on this same logarithmic energy grid. Incoherent and coherent atomic form factors as a function of momentum transfer are included for the calculation of secondary angular distributions. Fluorescence data are also available. The principal source for the photon data on this file is the compilation of Storm and Israel (below 15 MeV) and the ENDF/B photon interaction files (above 15 MeV). The PLIB code was used to access the ENDF/B cross sections, calculate the heating values, and merge the two data sources. PLIB is currently being updated to process all photon interaction data available in ENDF/B.

The only alteration of data performed by MCNP is the expunging of data outside the energy range of interest for a given problem and the thermal treatment applied to low energy cross sections. These modifications are problem dependent.

IV. VISUAL INFORMATION

To assist the user in cross-section selection two sources of information describing the cross sections in the Monte Carlo libraries are available. First, there are microfiche cards available which display the

Hollerith and BCD information for all isotopes in the Monte Carlo libraries. Second, there is an index of the cross sections which are on each of the RMCCS, AMCCS, DRMCCS, XMCCS, or UMCCS libraries. This index is at the end of this Appendix and lists the nuclide identification number (ZAID) and its library location. Also on the index is pertinent Hollerith information in addition to an important parameter SIZE. SIZE is the total, unexpunged, size of the cross-section set and warns the user of the LCM storage requirements. In this list *DRXS* indicates discrete reaction cross sections are available; *CPXS*, gamma production cross sections are available; and Ψ, total fission $\bar{\nu}$ not available.

Graphic information which compares the master evaluations (such as ENDF/B, ENDL, and AWRE) and the ACE representation of the evaluation is available on 35mm film. Other plotting packages are being developed to help the MCNP user in cross-section selection.

APPENDIX F

td-6 monte carlo neutron cross-section library 6feb78

zaid	library	mat	size
z = 1	**********		
1001.01	xmccs h-1 lll-howerton 1/73	501	3250
	gpxs = from xsec 12/14/72.		
1001.02	xmccs h-1 lasl dec 1970	5990	2011
	gpxs = from xsec 12/14/72.		
1001.03	xmccs h-1 endf/b-iii (t301)	1148	2459
	gpxs = from xsec 12/14/72.		
1001.04	rmccs h-1 endf/b4 t404 rev.1 11/75	1269	2459
	drxs gpxs = endf/b		2760
1001.30	amccs h-1 endl 7101 t=0k emin=1.0e-4	7101	1444
	gpxs = endf/b		
1002.01	xmccs h-2 lll-howerton 1/73	502	2122
	gpxs = from xsec 12/14/72.		
1002.02	rmccs h-2 uk-lasl dec 1967	2254	3007
	gpxs = from xsec 12/14/72.		
1002.03	xmccs h-2 endf/b-iii (t301)	1120	1963
	gpxs = from xsec 12/14/72.		
1002.04	amccs d-2 endf/b4 t402 rev.1 15oct75	1120	2144
	drxs gpxs = endf/b		2780
1003.01	xmccs h-3 lll-howerton 1/73	503	1111
	gpxs = none		
1003.02	umccs h-3 uk-lasl dec 1967	2252	1483
	drxs gpxs = none		2336
1003.03	rmccs h-3 endf/b-iv t=300k 2/21/78	1169	2114
	gpxs = none		
1003.30	amccs h-3 endl 7103 t=0k emin=1.0e-4	7103	1026
	gpxs = none		

```
z =  2 ***************************************************************

  2000.01  rmccs he endf/b-iii (t301) 20feb74          1088    1705
           drxs  gpxs = none                                     1976

  2003.01  xmccs he-3 iii-howerton 1/73                 504    1320
                gpxs = none

  2003.02  xmccs he-3       uk  jan 1971               7220     818
                gpxs = none

  2003.03  rmccs he-3 endf/b-iii (t301) 20feb74        1146    1517
           drxs  gpxs = none                                     1916

  2004.01  umccs he-4 iii-howerton 1/73                 505    1217
           drxs  gpxs = none                                     2141

  2004.02  xmccs he-4       lasl 1965                    22    1162
                gpxs = none

  2004.03  rmccs he-4 endf/b-iv t=300k 2/21/78          1270    2407
                gpxs = none

  2004.30  emccs he4  endl 7105 t=0k emin=1.0e-4        7105    1267
                gpxs = none

z =  3 ***************************************************************

  3006.01  xmccs li-6       awre april 1965            2214    3443
                gpxs = endf/b iii  march 1973

  3006.02  xmccs li-6 endf/b-iii (t301)                1115    3997
                gpxs = endf/b iii  march 1973

  3006.04  xmccs  li-6 endf/b-iv                       1271    4615
                gpxs = endf/b

  3006.10  rmccs li-6 lasl sublib mat 101 t=0 4/76      101    8294
           drxs  gpxs = endf/b                                   6566

  3006.30  xmccs li-6 endl 7106 0k 9/76 emin=1.0-4     7106    2940
                gpxs = endf/b
```

APPENDIX F

```
  3007.01  xmccs li-7 lll-howerton 1/73                507   2787
                gpxs = endf/b iii  march 1973

  3007.02  xmccs li-7       awre april 1965           2215   3097
                gpxs = endf/b iii  march 1973

  3007.03  xmccs li-7 endf/b-iii (t301)                1116   3562
                gpxs = endf/b iii  march 1973

  3007.05  rmccs li-7 endf/b4 t404 rev.1 10/75         1272   3751
           drxs  gpxs = endf/b                                 3647

z =  4  ***************************************************************

  4009.01  xmccs be-9 lll-howerton 1/73                509   3843
                gpxs = from xsec 12/14/72.

  4009.02  umccs be-9 endf/b-iii (t301) 20feb74        1154   3321
           drxs  gpxs = from xsec 12/14/72.                    4093

  4009.03  rmccs be-9 lasl sublibrary t=300k 2/16/78    104   7883
                gpxs = lasl sublibrary

z =  5  ***************************************************************

  5000.01  rmccs b lll-howerton 1/73                   510   5058
           drxs  gpxs =.198*gpxs for b-10, 1/73 seamon          3975

  5010.01  xmccs b-10 lll-howerton 1/73                511   5359
                gpxs = june 6, 1973 seamon

  5010.02  amccs b-10 endf/b-iii (t301) 20feb74        1155   2589
                gpxs = june 6, 1973 seamon

  5010.03  rmccs  b-10 endf/b-iv                       1273   9241
           drxs  gpxs = endf/b                                 5522

  5011.01  xmccs b-11 lll-howerton 1/73                 565   1516
                gpxs = none

  5011.02  rmccs b-11 endf/b-iii (t301) 20feb74        1160   5134
           drxs  gpxs = none                                   2390
```

z = 6 **********

6012.01	xmccs c-12 uk jan 1969	3006	3098
	gpxs = from xsec 12/14/72.		
6012.02	xmccs c-12 endf/b-iii (t301)	1165	6942
	gpxs = from xsec 12/14/72.		
6012.03	xmccs c-12 endf/b-iv	1274	7567
	gpxs = endf/b		
6012.04	xmccs c-12 webster-11+mat 1165 elas ang	6965	2430
	gpxs = none		
6012.10	rmccs c-12 lasl sublib mat 102 4/76	102	8309
	drxs gpxs = endf/b		6068
6012.30	xmccs c-12 endl 7112 0k 9/76 emin=1.0-4	7112	2815
	gpxs = endf/b		

z = 7 **********

7014.01	xmccs n-14 lll-howerton 1/73	513	9586
	gpxs = mat=4133 mod4. p.young 7/73		
7014.02	xmccs n-14 lasl-lrl aug 1970	5513	9211
	gpxs = mat=4133 mod4. p.young 7/73		
7014.04	rmccs n-14 endf/b-iv	1275	21553
	drxs gpxs = endf/b		9698
7014.30	amccs n-14 endl 7113 0k 9/76 emin=1.0-4	7113	5148
	gpxs = endf/b		

z = 8 **********

8016.01	xmccs o-16 lll-howerton 1/73	514	5540
	gpxs = mat=4134 mod2. p.young 7/73		
8016.02	xmccs o-16 lrl april 1971	5514	4791
	gpxs = mat=4134 mod2. p.young 7/73		
8016.04	rmccs o-16 endf/b-iv	1276	21823

```
          drxs  gpxs = endf/b                                    10332

  8016.30 amccs o-16 endl 7114 0k 9/76 emin=1.0-4   7114   4565
          drxs  gpxs = endf/b                                    4607

z =  9  **********************************************************

  9019.01 amccs f-19 lll-howerton 1/73               515   3101
          drxs  gpxs= endf/b4 mat=1277 shlaph t411 2/75    4062

  9019.02 xmccs f-19 endf/b4 mat=1277 t411 2/75     1277  26334
                gpxs = endf/b4 mat=1277 shlaph t411 2/75

  9019.03 rmccs  f-19 endf/b-iv                     1277  24464
          drxs  gpxs = endf/b                                    8772

z = 11  **********************************************************

 11023.01 rmccs na-23 lll-howerton 1/73             516   6816
          drxs  gpxs =endf/b-iii (mat 1156)  28june72      4940

z = 12  **********************************************************

 12000.01 xmccs mg lll-howerton 1/73                517   5137
                gpxs = none

 12000.02 rmccs mg endf/b-iii (t302) 20feb74        1014   3771
          drxs  gpxs = none                                      2742

z = 13  **********************************************************

 13027.01 amccs al-27 lll-howerton 1/73             518   3915
                gpxs = from xsec 12/14/72.

 13027.02 xmccs al-27     awre  april 1965          2035   5532
                gpxs = from xsec 12/14/72.

 13027.03 xmccs al-27 endf/b-iii (t302)             1135  18038
                gpxs = from xsec 12/14/72.

 13027.04 rmccs  al-27 endf/b-iv                    1193  32517
          drxs  gpxs = endf/b                                    9546
```

```
z = 14  ********************************************************

 14000.01  amccs si lll-howerton 1/73                    519  12371
           drxs  gpxs = from xsec 12/14/72.                     4477

 14000.02  rmccs si endf/b-iii (t302) 20feb74           1151  21632
           drxs  gpxs = from xsec 12/14/72.                     6819

z = 15  ********************************************************

 15031.01  rmccs p-31 lll-howerton 1/73                  520   2842
           drxs  gpxs = none                                    3238

z = 16  ********************************************************

 16032.01  rmccs s-32 lll-howerton 1/73                  521   3252
           drxs  gpxs = none                                    3057

z = 17  ********************************************************

 17000.01  amccs cl lll-howerton 1/73                    522   9854
                 gpxs= endf/b-iii (mat 1149) 6/17/72

 17000.02  rmccs cl endf/b-iii (t302) 20feb74           1149  38371
           drxs  gpxs = endf/b-iii(mat 1149) 6/17/72            8061

z = 18  ********************************************************

 18000.01  rmccs argon lll-howerton 1/73                 523   2029
           drxs  gpxs = none                                    2743

z = 19  ********************************************************

 19000.01  rmccs k lll-howerton 1/73                     524   7436
           drxs  gpxs = endf/b-iii (mat 1150)   15june72        3976

z = 20  ********************************************************

 20000.01  amccs ca lll-howerton 1/73                    525   9623
           drxs  gpxs = endf/b-iii (mat 1152)   16june72        4303

 20000.10  rmccs ca endf/b4 1165 0k 9/76 emin=1.-5      1195  24085
           drxs  gpxs = endf/b                                  9022
```

APPENDIX F

```
z = 22  ************************************************************

 22000.01  amccs ti lll-howerton 1/73                    526   9259
           drxs  gpxs = none                                   2926

 22000.02  xmccs ti        uk  april 1965                2190  3884
                 gpxs = none

 22000.11  rmccs ti endf/b4 1286 t=300.0k 5/77           1286  10644
           drxs  gpxs = endf/b                                 3743

z = 23  ************************************************************

 23000.30  rmccs v endf/b4 1196 0k 9/76 emin=1.0-5       1196  6456
           drxs  gpxs = endf/b                                 4449

z = 24  ************************************************************

 24000.01  amccs cr        uk  may 1966                  2045  3667
           drxs  gpxs = none                                   3422

 24000.11  rmccs cr endf/b4 t=300.0 24feb76              1191  38240
           drxs  gpxs = endf/b                                 11613

 24000.12  xmccs cr endf/b4 t=900.0 24feb76              1191  51663
                 gpxs = endf/b

z = 25  ************************************************************

 25055.01  rmccs mn-55 lll-howerton 1/73                  527   3586
           drxs  gpxs = none                                   2967

z = 26  ************************************************************

 26000.01  xmccs fe lll-howerton 1/73                     528   4102
                 gpxs = from xsec 12/14/72.

 26000.03  xmccs fe   dna  mat=4180mod2,  feb 1974       4180  62886
                 gpxs = from xsec 12/14/72.

 26000.11  rmccs fe endf/b4 t=300.0 24feb76              1192  54104
           drxs  gpxs = endf/b                                 8698
```

```
 26000.12  xmccs fe endf/b4 t=900.0 24feb76          1192  57638
           gpxs = endf/b

 26000.30  amccs fe endl 7132 0k 9/76 emin=1.0-4     7132  23179
           drxs  gpxs = endf/b                              3925

z = 28  ***********************************************************

 28000.01  amccs ni         uk  oct 1965            2046   5714
           drxs  gpxs = none                                3244

 28000.11  rmccs ni endf/b4 t=300.0 24feb76          1190  35192
           drxs  gpxs = endf/b                              5504

 28000.12  xmccs ni endf/b4 t=900.0 24feb76          1190  40842
           gpxs = endf/b

 28058.01  rmccs ni-58 lll-howerton 1/73             529   5347
           drxs  gpxs = none                                3375

z = 29  ***********************************************************

 29000.01  amccs cu lll-howerton 1/73                530   3629
           drxs  gpxs = from xsec 12/14/72.                 4273

 29000.02  xmccs cu         uk 1967                  2249   6476
           gpxs = from xsec 12/14/72.

 29000.10  rmccs cu endf/b-iv 1295 0k 9/76           1295  14703
           drxs  gpxs = endf/b                              8456

z = 31  ***********************************************************

 31000.01  rmccs ga lll-howerton 1/73                531   3730
           drxs  gpxs = none                                2775

z = 40  ***********************************************************

 40000.01  amccs zr         uk  nov 1965            2009   3837
           drxs  gpxs = none                                1877

 40000.02  rmccs zr endl mat7141 howerton-lll 3/75   7141  10312
           drxs  gpxs = endl 3/75 mat7141 w/laphan0 5/75    4157
```

APPENDIX F

```
z = 41  ************************************************************

 41093.01  amccs nb-93 lll-howerton 1/73                532   5880
           drxs  gpxs = none                                      3138

 41093.30  rmccs nb-93 endl 7143 0k 9/76 emin=1.-4      7143  29725
           drxs  gpxs = endf/b                                    5751

z = 42  ************************************************************

 42000.01  rmccs mo lll-howerton 1/73                    533   5714
           drxs  gpxs = t-2 phlag program  april 1973             3796

z = 48  ************************************************************

 48000.01  rmccs cd lll-howerton 1/73                    534   7690
           drxs  gpxs = none                                      2839

z = 50  ************************************************************

 50000.01  rmccs sn lll-howerton 1/73                    535   2332
           drxs  gpxs = none                                      2993

 50999.02  rmccs fission products crude endl 1/73        558   1647
           drxs  gpxs = none                                      2708

z = 56  ************************************************************

 56138.01  rmccs ba-138 lll-howerton 1/73                536   2606
           drxs  gpxs = none                                      3069

z = 63  ************************************************************

 63000.01  rmccs eu lll-howerton 1/73                    537   3133
           drxs  gpxs = none                                      2885

z = 64  ************************************************************

 64000.01  rmccs gd lll-howerton 1/73                    538   3206
           drxs  gpxs = none                                      2946

z = 67  ************************************************************
```

```
 67165.01  rmccs ho-165 lll-howerton 1/73              539   3626
           drxs  gpxs = none                                 3157

z = 73  ***********************************************************

 73181.01  amccs ta-181    lrl  oct 1966              8731   2431
           drxs  gpxs = none                                 2968

 73181.02  rmccs ta-181 lll-howerton 1/73              540  18114
           drxs  gpxs = none                                 4240

z = 74  ***********************************************************

 74000.01  rmccs w lll-howerton 1/73                   541   3237
           drxs  gpxs = from mats 328,329,330,331 11sep73    3896

 74182.10  rmccs w-182 endf/b4 t=0.0 24feb76          1128  33247
           drxs  gpxs = endf/b                               5766

 74183.10  rmccs w-183 endf/b4 t=0.0 24feb76          1129  27816
           drxs  gpxs = endf/b                               6971

 74184.10  rmccs w-184 endf/b4 t=0.0 24feb76          1130  27996
           drxs  gpxs = endf/b                               5985

 74186.10  rmccs w-186 endf/b4 t=0.0 24feb76          1131  30916
           drxs  gpxs = endf/b                               6054

z = 78  ***********************************************************

 78000.01  rmccs pt lll-howerton 1/73                  566  10313
           drxs  gpxs = none                                 3049

z = 79  ***********************************************************

 79197.01  xmccs au-197 lll-howerton 1/73              542   3561
                 gpxs = none

 79197.02  xmccs au-197    uk  oct 1965               2222   2660
                 gpxs = none

 79197.10  rmccs au-197 endl 7163 t=0. 4/76           7163  19844
           drxs  gpxs = endf/b                               4519
```

APPENDIX F

```
z = 82  ************************************************************

 82000.01  amccs pb lll-howerton 1/73                    543   3148
           drxs  gpxs = from xsec 12/14/72.                      3902

 82000.02  xmccs pb         awre  april 1965            2026   2192
                gpxs = from xsec 12/14/72.

 82000.10  rmccs pb endf/b-iv 0k 9/76                    1288  21052
           drxs  gpxs = endf/b                                 11376

z = 90  ************************************************************

 90232.01  xmccs th-232 lll-howerton 1/73 Ψ              544   4239
                gpxs = none

 90232.02  xmccs th-232 awre april 1965 Ψ               2022   2433
                gpxs = none

 90232.10  rmccs th-232 endl 7165 13apr76 t=0.0 Ψ       7165  31235
           drxs  gpxs = endf/b                                  4237

z = 92  ************************************************************

 92233.01  xmccs u-233 lll-howerton 1/73 Ψ              545   4591
                gpxs = none

 92233.10  rmccs u-233 endl 7166 13apr76 t=0.0 Ψ       7166   7945
           drxs  gpxs = endf/b                                  4393

 92234.01  xmccs u-234 lll-howerton 1/73 Ψ              546   2750
                gpxs = none

 92234.10  rmccs u-234 endl 7167 13apr76 t=0.0 Ψ       7167   3008
           drxs  gpxs = endf/b                                  4581

 92235.01  amccs u-235 lll-howerton 1/73 Ψ              547  12770
           drxs  gpxs = mat=7052 lll 10/73 t=0. 6/75            4985

 92235.04  xmccs u-235 endf/b4 t407 t=3.0e+3 2/75       1261  29516
                gpxs = u-235 t=3.0e+3 9jun75

 92235.05  xmccs u-235 endf/b4 t407 t=3.0e+4 2/75       1261  18573
```

	gpxs = u-235 t=3.0e+4 9jun75		
92235.06	xmccs u-235 endf/b4 t407 t=6.0e+5 2/75	1261	12560
	gpxs = u-235 t=6.0e+5 9jun75		
92235.07	xmccs u-235 endf/b4 t407 t=1.2e+7 2/75	1261	11268
	gpxs = u-235 t=1.2e+7 9jun75		
92235.08	xmccs u-235 endf/b4 t407 t=0.0e+0 2/75	1261	42923
	gpxs = u-235 t=0.0 9jun75		
92235.09	xmccs u-235 endf/b4 t407 t=3.0e+2k 2/75	1261	41638
	gpxs = u-235 t=3.0e+2 9jun75		
92235.10	amccs u-235 endf/b4 407 t=0. 1/76	1261	42716
	drxs gpxs = endf/b		7236
92235.11	amccs u-235 endf/b4 407 t=300. 1/76	1261	41332
	drxs gpxs = endf/b		7236
92235.15	amccs u-235 endf/b4 407 t=12.e+6 1/76	1261	11267
	drxs gpxs = endf/b		7236
92235.18	amccs u-235 endf/b4 t=3000.0 24feb76	1261	29254
	drxs gpxs = endf/b		7236
92235.19	rmccs u-235 endf/b4 t=3*e+4 24feb76	1261	18385
	drxs gpxs = endf/b		7236
92235.20	amccs u-235 endf/b4 t=6*e+5 24feb76	1261	12497
	drxs gpxs = endf/b		7236
92235.30	amccs u-235 endl 0k 14sep76 Ψ	7168	17940
	drxs gpxs = endf/b		4931
92236.01	rmccs u-236 lll-howerton 1/73 Ψ	548	3153
	drxs gpxs = none		3908
92237.01	rmccs u-237 lll-howerton 1/73 Ψ	549	2478
	drxs gpxs = none		3604
92238.01	amccs u-238 lll-howerton 1/73 Ψ	550	4703
	drxs gpxs = endf/b (stewart,hunter) 12/72		5115

APPENDIX F

92238.04	xmccs	u-238 endf/b4 t409 t=3.0e+4k 2/75	1262	32918
		gpxs = u-238 t=3.0e+4 9jun75		
92238.05	xmccs	u-238 endf/b4 t409 t=6.0e+5 2/75	1262	18803
		gpxs = u-238 t=6.0e+5 9jun75		
92238.06	xmccs	u-238 endf/b4 t409 t=1.2e+7 2/75	1262	10399
		gpxs = u-238 t=1.2e+7 9jun75		
92238.12	umccs	u-238 endf/b4 300k 8/77 10%thin	1262	50412
		gpxs = endf/b		
92238.13	rmccs	u-238 endf/b4 t409 t=3.e+4 1/76	1262	32895
	drxs	gpxs = endf/b		6799
92238.15	amccs	u-238 endf/b4 t409 t=12.e+6 1/76	1262	10351
	drxs	gpxs = endf/b		6864
92238.20	amccs	u-238 endf/b4 t=6e+5 24feb76	1262	18721
	drxs	gpxs = endf/b		6858
92238.30	amccs	u-238 endl 0k 14sep76 Ψ	7171	15110
	drxs	gpxs = endf/b		6639
92239.01	rmccs	u-239 lll-howerton 1/73 Ψ	551	3227
	drxs	gpxs = none		3767
92240.01	rmccs	u-240 lll-howerton 1/73 Ψ	552	2777
	drxs	gpxs = none		3833

z = 94 **********

94238.01	rmccs	pu-238 lll-howerton 1/73 Ψ	553	2588
	drxs	gpxs = none		3495
94239.01	amccs	pu-239 lll-howerton 1/73 Ψ	554	6077
	drxs	gpxs = endf mat 304 stewart,hunter 12/72		5772
94239.02	xmccs	pu-239 endf/b4 t407 t=3.0e+3 2/75	1264	40464
		gpxs = pu-239 t=3.0e+3 9jun75		
94239.03	xmccs	pu-239 endf/b4 t407 t=3.0e+4 2/75	1264	25460
		gpxs = pu-239 t=3.0e+4 9jun75		

94239.04	xmccs pu-239 endf/b4 t407 t=6.0e+5 2/75	1264	13633
	gpxs = pu-239 t=6.0e+5 9jun75		
94239.05	xmccs pu-239 endf/b4 t407 t=1.2e+7 2/75	1264	11349
	gpxs = pu-239 t=1.2e+7 9jun75		
94239.06	xmccs pu-239 endf/b4 t407 t=0.0e+0 2/75	1264	41167
	gpxs = pu-239 t=0.0 9jun75		
94239.07	xmccs pu-239 endf/b4 t407 t=3.0e+2 2/75	1264	34659
	gpxs = pu-239 t=3.0e+2 9jun75		
94239.15	amccs pu-239 endf/b4 407 t=12.e+6 1/76	1264	11297
	drxs gpxs = endf/b		7880
94239.16	amccs pu-239 endf/b4 t=0 24feb76	1264	41153
	drxs gpxs = endf/b		7880
94239.17	amccs pu-239 endf/b4 t=300 24feb76	1264	34631
	drxs gpxs = endf/b		7880
94239.18	amccs pu-239 endf/b4 t=3000 24feb76	1264	40421
	drxs gpxs = endf/b		7880
94239.19	rmccs pu-239 endf/b4 t=3*e+4 24feb76	1264	25417
	drxs gpxs = endf/b		7880
94239.20	amccs pu-239 endf/b4 600000k 24feb76	1264	13590
	drxs gpxs = endf/b		7880
94239.99	xmccs pu-239 lrl 1/65 Ψ	942	3833
	gpxs =endf mat 304 stewart,hunter 12/72		
94240.01	amccs pu-240 lll-howerton 1/73 Ψ	555	3653
	drxs df mat 305 stewart hunter 12/72		4640
94240.12	rmccs pu-240 endf/b-iv 900k 14sep76	1265	41821
	drxs gpxs = endf/b		5911
94241.01	rmccs pu-241 lll-howerton 1/73 Ψ	556	3443
	drxs gpxs = none		3955

z = 95 **********

APPENDIX F

```
 95242.01   rmccs am-242 111-howerton 1/73                557    5777
            drxs  gpxs = none                                       3641

z = 99  ************************************************************

 99003.01   umccs bm-3 boredum test data                    1      1039
                  gpxs = none


note************************************************************

zaid  = td-6 cross-section library identification number
mat   = evaluators material identification no.
size = words10 in cross-section set
length of rmccs =   884170
length of amccs =   485593
length of xmccs =   797864
length of umccs =    51659
Ψ = total ν̄ not available for nuclide
```