Internal Report
DESY D3/39
October 1981

A Manual for EGS3 User Codes at DESY for Synchrotron Radiation Problems

Chiri Yamaguchi

A Manual for EGS3 User codes at DESY for Synchrotron Radiation Problem

Chiri Yamaguchi

A B S T R A C T

The electromagnetic casade shower code EGS (version 3) by Ford and Nelson is widely used around high energy electron machines. In order to use this code the user has to write his own MAIN user code and two subroutines, one of which is geometry subrotine HOWFAR and the other is output subroutine AUSGAB. This is a manual to show how to use some user codes that the author has prepared for solving synchrotron radiation problems during his stay at DESY.

1. INTRODUCTION

The electromagnetic cascade code EGS (Version 3), by Ford and Nelson[1], referred to as EGS3, has been used for various purposes. It is a package of computer programs that simulates the transport of electromagnetic cascade showers in various media using the Monte Carlo method. It outputs the fraction of energy deposited in various parts of the media. In order to use EGS3 the user has to prepare his own MAIN user code and two subroutines called HOWFAR and AUSGAB.

This is a manual to guide how to use some EGS3 user codes at DESY which have been prepared to calculate the absorbed dose in various parts of the accelerator components caused by the synchrotron radiation. They also output the photon energy spectrum or energy flux density at various regions of interest in the air spaces in the magnet as well as in the accelerator tunnel. Once a type of geometry is fixed the user does not need to change HOWFAR and AUSGAB. Thus most of the space in this manual has been spent for the handling of the MAIN user code.

Chapter 2 gives some basic equations useful for the absorbed dose calculation from the synchrotron radiation, and it is not important in terms of using EGS3 to solve problems. The reader may ignore it if he wants to save time. The EGS3 version at DESY is written in MORTRAN, and some fundamental knowledge of it is required to use the program. But it is not necessary to understand fully the MORTRAN Macros which appear at the beginning of the MAIN user code. A good introductory text for MORTRAN would be ref.2.

2. SOME USEFUL EQUATIONS USED IN THE PROGRAM

2.1 Synchrotron radiation

When a charged particle moves on a circular orbit with a highly relativistic velocity, it emits electromagnetic radiation, generally known as synchrotron radiation. The name originates from the fact that it was first observed at the 70 MeV General Electric synchrotron in Schenectady, U.S.A., in 1946. It has a continuous spectrum, which, in the case of PETRA energy regions, ranges from infrared to the hard X-ray region of a few hundred keV.

It is highly directional and is emitted along a tangent to the circulating particle orbit in a narrow cone of the angle m_0c^2/E in the laboratory system, where m_0 is the rest mass and E is the energy of the particle.

The radiation power loss ΔE per revolution of an electron, or a positron, in a circular orbit of radius R is expressed from the equation of Schwinger[3] as follows:

$$\text{Energy loss per revolution: } \Delta E(\text{keV}) = 88.46 \frac{[E(\text{GeV})]^4}{R(\text{m})} \qquad (1)$$

The bending radius of the PETRA dipole magnet is 192 m, thus ΔE = 38.5 MeV at E = 17 GeV. The energy loss per m, δE, of the bending magnet is obtained by dividing ΔE by $2\pi R$ or:

$$\text{Energy loss per m: } \delta E \ (\text{keV} \cdot \text{m}^{-1}) = \frac{\Delta E(\text{keV})}{2\pi R(\text{m})} \qquad (2)$$

The critical energy E_c is defined as the energy below which half the total power is radiated and half above.

$$\text{Critical energy: } E_c(\text{keV}) = 2.218 \frac{[E(\text{GeV})]^3}{R(\text{m})} \qquad (3)$$

The critical energy for PETRA at 17 GeV operation is 56.7 keV. Some useful figures for PETRA, HERA and LEP are listed in Table 1. The photon spectral density is given as follows:

$$\frac{d^2N}{dE \cdot dt} = \frac{5.320 \times 10^{11}}{[E(\text{GeV})]^2} G(x) \qquad (\text{photons MeV}^{-1}\text{s}^{-1}) \qquad (4)$$

$$\frac{d^2N}{dE \cdot dl} = \frac{17.75}{[E(\text{GeV})]^2} G(x) \qquad (\text{photons MeV}^{-1}\ \text{m}^{-1}) \qquad (5)$$

where $G(x)$ is the integral of a modified Bessel function $K_{5/3}$, and it is expressed as:

$$G(x) = \int_x^\infty K_{5/3}(t)\, dt, \quad x = \frac{E}{E_c} \cdot \qquad (6)$$

A new program for the efficient computation of eq. 6 has been developed[4].

2.2 Absorbed dose calculation in air region

The absorbed dose from the synchrotron radiation in various accelerator components is well assumed from the measured values with RPL (radio-photoluminescence) glass dosimeter. This is based on the fact that many of the accelerator components, in which the absorbed dose is to be calculated, can be presumed to be made from aluminum whose mass energy absorption coefficient is very closely the same as that of the glass dosimeter.

The glass dosimeter has a rather small dimension of 1 mmϕ × 6 mm. During the measurement it is surrounded with a few

mm thick plastic wall to obtain radiation equilibrium between primary photons and secondary electrons. On this condition the absorbed dose to the glass dosimeter set in the free air is calculated by the following equation[5]:

$$D(\text{rads}) = 1.60 \times 10^{-8} \Sigma \phi_i E_i \left(\frac{\mu_{en}}{\rho}\right)_i t \ , \qquad (7)$$

where ϕ_i = photon flux density ($cm^{-2}s^{-1}$) for i-th photon,
E_i = average photon energy (MeV) for the i-th photon,
$\left(\frac{\mu_{en}}{\rho}\right)_i$ = mass energy absorption coefficient (cm^2g^{-1}) of the glass dosimeter for energy E_i,
t = time (s) of measurement.

The mass energy absorption coefficient is given by

$$\frac{\mu_{en}}{\rho} = \frac{1}{\rho}\left[\tau(1-f) + \sigma\frac{\bar{E}}{h\nu} + \kappa\left(1 - \frac{2mc^2}{h\nu}\right)\right][1-G], \qquad (8)$$

where τ = photoelectric coefficient,
σ = total Compton cross section,
κ = pair production cross section,
f = fluorescent x-ray fraction,
G = fraction of energy lost by secondary elections in bremsstrahlung processes.

The product of $\phi_i E_i$ is called energy flux density. In EGS3 user code the photons are sorted into the energy bins which the user has to specify. It outputs energy carried by particle into"the region of interest" and the summed value over the whole bins is given as SWNP (see 3.2 "Step 8").

The value of SWNP can be used as a measure of the stochastic variation in the result (shown later). The mean value of mass

energy absorption coefficient ($^{\mu}en/\rho$) is used for this calculation within a good accuracy. The mass energy absorption for glass dosimeter is shown in ref. 6, which has been calculated based on its elemental composition from the table compiled by Storm and Israel[7].

The product $\phi_i E_i (^{\mu}en/\rho)_i t$ is equal to kerma, or kinetic energy released in matter, minus energy lost by secondary electrons in bremsstrahlung process. The program outputs this product as "Kerma" for the corresponding energy bins as well as the sum over whole bins. Note that since the values are given for the specified regions of interest, the values given by the program must be divided by the cross section of the corresponding region on x-y plane to normalize to a unit flux density. Since the calculation is made assuming an infinite length of the absorbing material in x-direction and the results are given per unit length (m), the cross section A on x-y plane is given as:

$$A = 100(\text{cm}) \times H(\text{cm}) \qquad (9)$$

where H = height of the region in y-direction.

The cross section of the region of interest, which is filled with air, on y-z plane should be nearly square in the present way of dose calculation (see ROI= 6, 24, 42, 96, ... in Fig. 2).

3. HOW TO MODIFY THE USER CODE FOR YOUR OWN USE

Many EGS3 user codes have been prepared by the author during his stay at DESY. Among them PETRA4 would be the most suitable code to be chosen as a text. Once the reader gets to be able to handle this code, he will easily do the same with other codes, too.

To begin with, let us pick up an actual problem.

Problem: We want to know the absorbed dose due to synchrotron radiation in the glass dosimeters which were set in the air gaps of the PETRA dipole magnet.

The position of the dosimeters and the cross section of the magnet are shown in Fig. 1. The bending radius of the dipole magnet is 190 m, and the energy of electron and positron is 17 GeV.

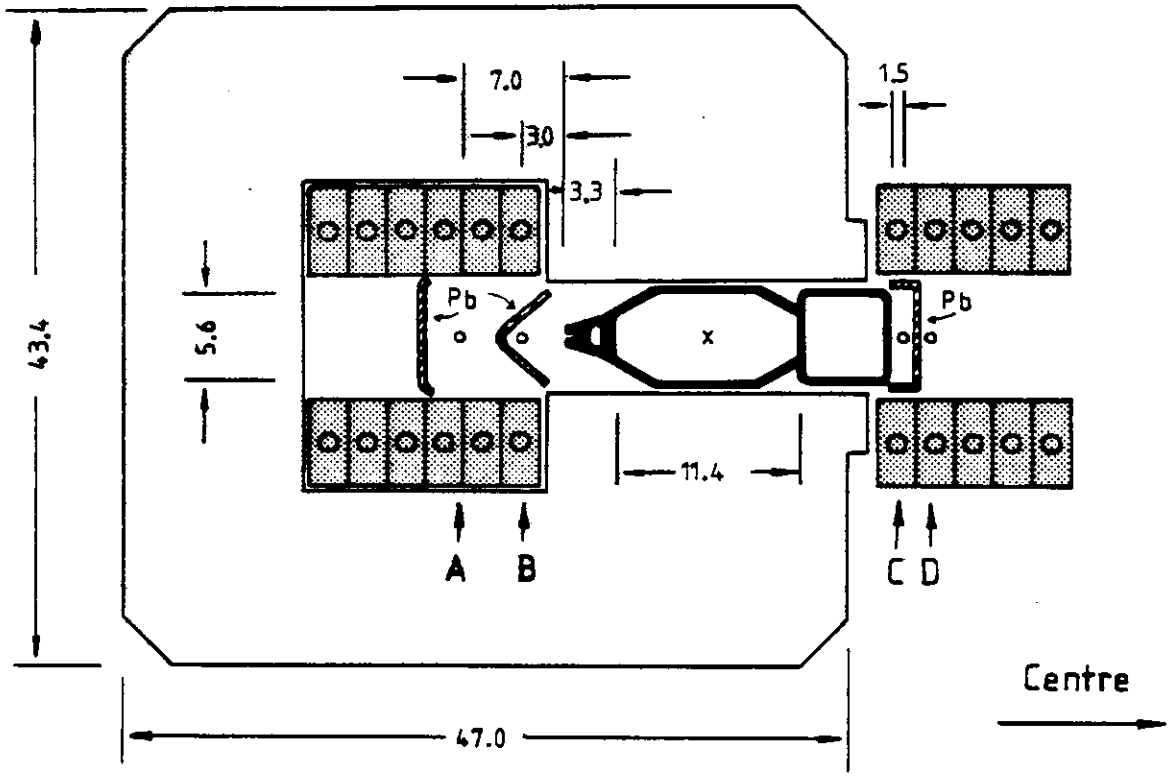

Fig. 1 Cross section of PETRA dipole magnet. The position of glass dosimeter in the magnet air gap is indicated by A, B, C and D.

We solve this problem as follows:

3.1 Geometrical configuration (2 dimensional)

At first we have to set up the geometrical configuration for the problem. We must define the regions whose absorbed doses and/or fraction of absorbed energy are to be calculated. The geometrical package HOWFAR in PETRA4 is "two" dimensional.

Fig. 2 gives a geometrical configuration which we are going to use to solve the given problem. The magnet is assumed to have an infinite length in x-direction, and the EGS3 code simulates the electromagnetic shower "three" dimensionally. For simplicity the planes that divide the figures are parallel or perpendicular from each other in the present case. We have to give each plane a "plane number". The number should start from 1 up to the number of planes. First, we have to number the vertical (x-y) planes from the left to the right, i.e. 1,2,..., 18 in this case. Then we have to continue the same process for the horizontal (x-y) planes, i.e. 19, 20,..., 28.

The planes divide the space into regions. We have to number the regions in order. Let us start from the left to the right (to z-direction) and the bottom to the top (y-direction). Note that the outer regions except region 1 must be numbered later as shown in the figure.

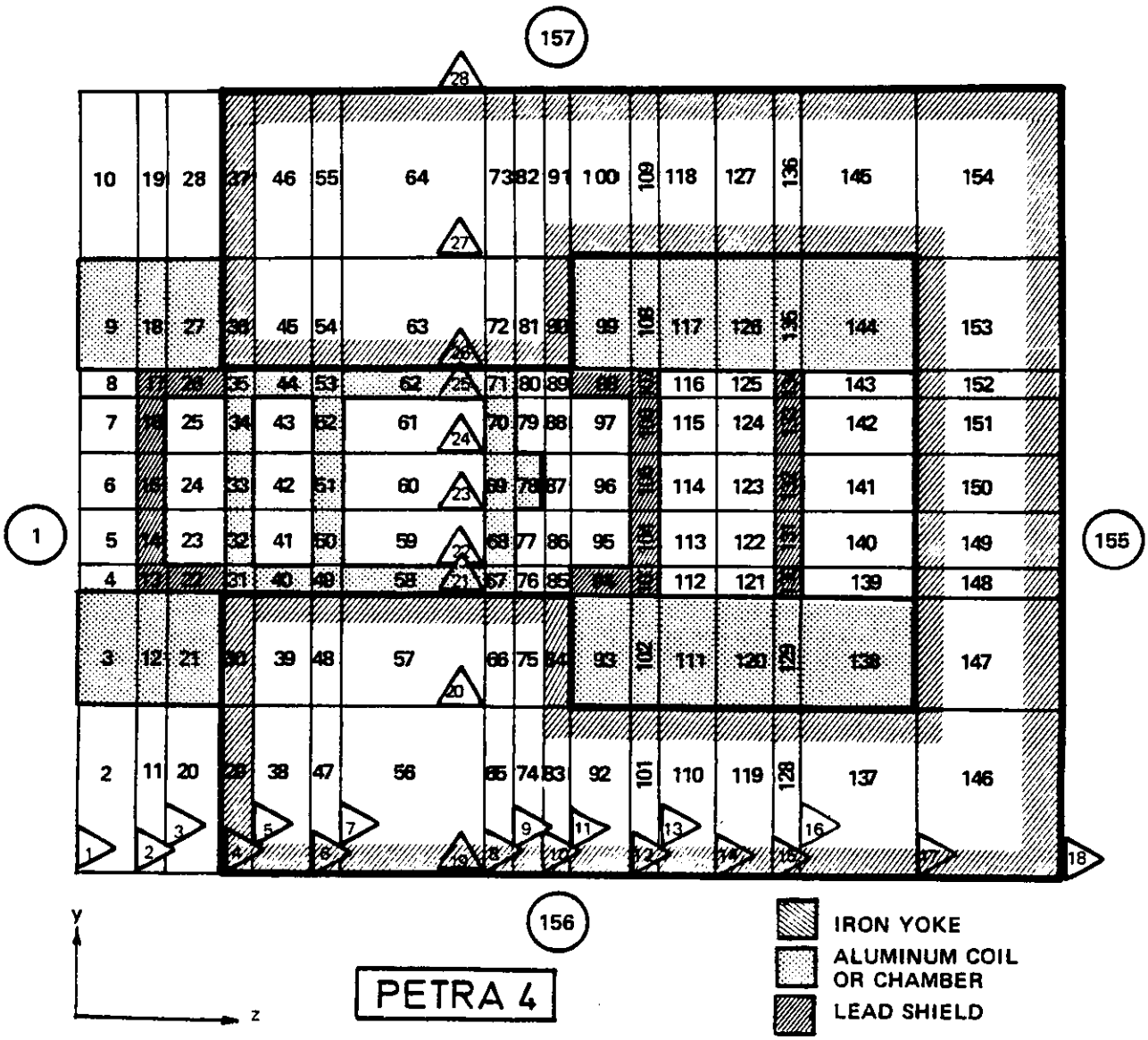

Fig. 2 Geometrical configuration used in the EGS user code PETRA4.

3.2 Modification of the main user code

We have to modify the MAIN user code depending on our calculation requests. The main user code consists of "eight" steps. So, let us follow it according to these steps. The steps are numbered in the program. In this process we will check even some variables or strings which we usually do not have to change.

```
%I4
*    PETRA4 --EGS USER CODE TO CALCULATE THE ENERGY DEPOSITION*
*           CAUSED BY SYNCHROTRON RADIATION THAT IS SCATTERED *
*           FROM THE VACUUM CHAMBER OF PETRA AND TO GET LEPTON*
*           SPECTRA FOR PARTICLES LEAVING THE GEOMETRY*
*           C.YAMAGUCHI/W.R.NELSON*
*                                          FRACTION*
*           DESY/SLAC*
*           13 APRIL 1981/REV. 7 MAI 1981*
*****************************************************************
```

"Step 1"

%'$MXEBINS'='46'

This string specifies the "maximum number of energy bins",and usually we do not have to change it.

We have to change the number if we want to increase the number of energy bins more than the present number 46, which is large enough for general purposes.

```
0
0     "STEP 1"
0
0
0
0     "EGS3MACX---15 FEB 1980 - 1246 (SLAC/CERN) WRN"
0     "EXTENSION SET OF MACROS FOR EGS3MAC"
0
      %'$MXPLNS'='100'   "MAXIMUM NUMBER OF PLANES"
0
0     "PLADTA---COMMON BLOCK FOR $PLANE1 AND $PLANE2 MACROS"
      %';COMIN/PLADTA/;'=';COMMON/PLADTA/PCOORD(3,$MXPLNS),PNORM(3,$MXPLNS);'
0
0     "$PLANE1---MACRO REPLACEMENT FOR SUBROUTINE PLANE1"
      %'$PLANE1(#,#,#,#);'='UDOTA=PNORM(1,#1)*U(NP)+PNORM(2,#1)*V(NP)+PNORM(3,
      #1)*W(NP); UDOTAP=#2*UDOTA; IF(UDOTA.EQ.0.0)<#3=2;>ELSEIF(UDOTAP.LT.0.0)
      <#3=0;> ELSE <#3=1; #4=(PNORM(1,#1)*(PCOORD(1,#1)-X(NP))+PNORM(2,#1)*
      (PCOORD(2,#1)-Y(NP))+PNORM(3,#1)*(PCOORD(3,#1)-Z(NP)))/UDOTA;>'
      "NOTE: EVERYWHERE $PLANE1 IS USED ONE MUST INCLUDE COMIN/PLADTA,STACK/"
0
0     "$PLANE2---MACRO REPLACEMENT FOR SUBROUTINE PLANE2"
      %'$PLANE2(#,#,#,#,#);'='$PLANE1(#1,#3,IHIT,TVAL); IF(IHIT.EQ.1) <
      $CHGTR(TVAL,#2);> ELSEIF(IHIT.EQ.0) <$PLANE1(#4,#6,IHIT,TVAL);
      $CHGTR(TVAL,#5);>'   "NOTE: EVERYWHERE $PLANE2 IS USED ONE MUST INCLUDE
      COMIN/EPCONT,PLADTA,STACK/"
0
      %'$MXCYLS'='75'   "MAXIMUM NUMBER OF CYLINDERS"
      %'$DELCYL'='1.0E-6'   "CLOSEST ALLOWABLE DISTANCE TO CYLINDER SURFACE"
0
0     "CYLDTA---COMMON BLOCK FOR $CYLNDR MACRO"
      %';COMIN/CYLDTA/;'=';COMMON/CYLDTA/CYRAD2($MXCYLS);'
0
0     "$CYLNDR---MACRO REPLACEMENT FOR SUBROUTINE CYLNDR"
      %'$CYLNDR(#,#,#,#);'='#3=1; ACYL=U(NP)*U(NP)+V(NP)*V(NP); BCYL=X(NP)*
      U(NP)+Y(NP)*V(NP); CCYL=X(NP)*X(NP)+Y(NP)*Y(NP)-CYRAD2(#1);
      ARGCY=BCYL*BCYL-ACYL*CCYL;
      IF(ACYL.EQ.0.0) <#3=0;> ELSEIF(ABS(CCYL).LT.$DELCYL) <
      IF(#2.NE.0) <IF(BCYL.LT.0.0) <#4=-2.*BCYL/ACYL;> ELSE <#4=0.0;>> ELSE
      <IF(BCYL.LT.0.0) <#4=0.0;> ELSE <#3=0;>>>
      ELSEIF(ARGCY.LT.0.0) <#3=0;>   ELSE <
      ROOTCY=SQRT(ARGCY); IF(BCYL.GT.0.0) <#4=(-BCYL-ROOTCY)/ACYL;> ELSE
      <#4=-CCYL/(BCYL-ROOTCY);> IF(#4.LT.0.0) <
      IF(BCYL.LT.0.0) <#4=(-BCYL+ROOTCY)/ACYL;> ELSE <#4=-CCYL/(BCYL+ROOTCY);>
      IF(#4.LT.0.0) <#3=0;>>>'
      "NOTE: EVERYWHERE $CYLNDR IS USED ONE MUST INCLUDE COMIN/CYLDTA,STACK/"
0
0     ;    "INSERTED EXTRA SEMI-COLON HERE TO FLUSH INPUT BUFFER"
0
      %'$MXCONES'='75'   "MAXIMUM NUMBER OF CONES"
      %'$DELCON'='1.0E-4'   "CLOSEST ALLOWABLE DISTANCE TO CONE SURFACE"
0
0     "CONDTA---COMMON BLOCK FOR $CONE MACRO"
      %';COMIN/CONDTA/;'=';COMMON/CONDTA/COALP2($MXCONES),SMALLL($MXCONES);'
0
0     "$CONE---MACRO REPLACEMENT FOR SUBROUTINE CONE"
      %'$CONE(#,#,#,#);'='#3=1; ZMINL=Z(NP)-SMALLL(#1);
0     ACON=U(NP)*U(NP)+V(NP)*V(NP)-COALP2(#1)*W(NP)*W(NP);
0     BCON=X(NP)*U(NP)+Y(NP)*V(NP)-COALP2(#1)*ZMINL*W(NP);
0     CCON=X(NP)*X(NP)+Y(NP)*Y(NP)-COALP2(#1)*ZMINL*ZMINL;
0     IF(ACON.EQ.0.0) <IF(BCON.EQ.0.0)<#3=0;> ELSE <TCONE=-CCON/2./BCON;
0     IF(TCONE.LT.0.0)<#3=0;>>>ELSE<ACCON=ACON*CCON;BCON2=BCON*BCON;ARGCO=
      BCON2-ACCON;IF(ABS(CCON).LT.$DELCON) <IF(#2.NE.0) <IF(BCON.LE.0.0) <
      TCONE=-2.*BCON/ACON; IF(TCONE.LT.0.0) <#3=0;>> ELSE <TCONE=0.0;>>
      ELSE <IF(BCON.LT.0.0) <TCONE=0.0;> ELSE <#3=0;>>>
      ELSEIF(ARGCO.LT.0.0) <#3=0;> ELSE <ROOTCO=SQRT(ARGCO);
      IF(BCON.GT.0.0) <TCONE1=(-BCON-ROOTCO)/ACON;> ELSE <TCONE1=
      -CCON/(BCON-ROOTCO);> IF(BCON.LT.0.0) <TCONE2=
      (-BCON+ROOTCO)/ACON;> ELSE <TCONE2=-CCON/(BCON+ROOTCO);>
      IF(TCONE1.LT.0.0) <IF(TCONE2.LT.0.0) <#3=0;> ELSE <TCONE=TCONE2;>>
      ELSE <IF(TCONE2.LT.0.0) <TCONE=TCONE1;>
      ELSE <TCONE=AMIN1(TCONE1,TCONE2);>>>>'
      "NOTE: EVERYWHERE $CONE IS USED ONE MUST INCLUDE COMIN/CONDTA,STACK/"
0
0     ;    "INSERTED EXTRA SEMI-COLON HERE TO FLUSH INPUT BUFFER"
0
0     "$CHGTR---MACRO REPLACEMENT FOR SUBROUTINE CHGTR"
      %'$CHGTR(#,#);'=';IF(#1.LE.USTEP) <USTEP=#1; IRNEW=#2;>'
0     "NOTE: EVERYWHERE $CHGTR IS USED ONE MUST INCLUDE COMIN/EPCONT/"
0
0     "$FINVAL---MACRO REPLACEMENT FOR SUBROUTINE FINVAL"
      %'$FINVAL(#,#,#,#);'='#2=X(NP)+#1*U(NP); #3=Y(NP)+#1*V(NP);
0     #4=Z(NP)+#1*W(NP);'   "NOTE: EVERYWHERE $FINVAL IS USED ONE MUST
0     INCLUDE COMIN/STACK/"
0
0     "END OF EGS3 MACRO EXTENSION (EGS3MACX)"
0
      %'$MXEBINS'='40'
```

DIMENSION XXX(30, 45);

We do not have to change this string usually. This is used at "Step 8" in order to save and output "Kerma" data (see Chapter 2) for each energy bin.

Dimension of XXX(A,B) is given as follows:

A$\geq$ NROI, the number of regions of interest (see "Step 2"),
B$\geq$ Number of energy bins eventually used.

A=30, B=45, or XXX(30, 45) is large enough for general purposes.

REAL EBDTA (25)/ ..... /;

This gives the energy bin data. In "Step 2" we will point the lower and higher energy bins that will specify the energy region of photons treated in the calculation. Of course, it is possible to overwrite the data. The number of data should be less or equal to '$MXEBINS' (see above).
Energies should be given in MeV

INTEGER MEDARR (24,4)/ ..... /;

This is to specify the array of the media with which we are going to fill our geometry. Each medium string consists of 24 characters and it should be exactly the same as that in the filed "PEGS data" (ref. 1). Therefore,

```
0
      %'$MXEBINS'='46'
0
0     "NTALY1---COMMON BLOCK FOR KEEPING COUNT OF ETALY1-EVENTS"
      %';COMIN/NTALY1/;'='COMMON/NTALY1/NSUM(4,$MXREG,5);'
0
0     ;COMIN/BOUNDS,DEBUG,ETALY1,MEDIA,MISC,NTALY1,PLADTA,RANDOM,
0     UPHIOT,USEFUL/;
0     COMMON/LINES/NLINES,NWRITE,NCOUNT,ILINES;
0     COMMON/PASSIT/NNP(30,3,$MXEBINS),WNP(30,3,$MXEBINS),IRGI(30),
0     NROI,ESAM1,NREG,NYBIN,NZBIN,INDY,INDZ,IRLP;
0     COMMON/EDATA/EBIN($MXEBINS),NEBIN;
0     REAL YSORT1($MXEBINS),THSPEC($MXEBINS),EAVE($MXEBINS);
0     REAL*8 FINT1,ERRDUM,SMALRA,SMALR,SMALR1,SMALR2;
0     REAL*8 EI,EKIN,TOTKE,WTINDP,WNP,SWNP,XXX;
0     INTEGER NEBLO,NEBHI;   "ENERGY SAMPLING BIN POINTERS"
0
0     DIMENSION XXX(30,46);     " NOTICE DIMENSION XXX( NROI, NO OF EBDTA )"
0
0     REAL EBDTA(25)/
0     0.01  ,0.015,0.02  ,0.03  ,0.04  ,0.05  ,0.06  ,0.08  ,          00011900
0     0.1   ,0.15  ,0.2   ,0.3   ,0.4   ,0.5   ,0.6   ,0.8   ,          00012000
0     1.0   ,1.5   ,2.0   ,3.0   ,4.0   ,5.0   ,6.0   ,8.0   ,          00012100
0     10.0/;                                                           00012200
0
0     INTEGER ISORT1($MXEBINS);
0
0     "THE FOLLOWING STATEMENT DETERMINES THE MEDIA TO BE USED IN PROBLEM"
0
0     INTEGER MEDARR(24,4)/
0     $S'FE  (AP=1 KEV)',I1$'  ',
0     $S'PB  (AP=1 KEV)',I1$'  ',
0     $S'AL  (AP=1 KEV)',I1$'  ',
0     $S'AIR AT NTP  (AP=1 KEV)',3$'  '/;
0
```

```
$S'FE (AP=1 KEV)', 11x' ' is not equal to
$S'FE (AP=1KEV)', 12x' '
```

because the former string has a blank between AP=1 and KEV while the latter does not.

Notice: Change X in MEDARR (24,X) according to the number of the media which you "eventually" use (a must at DESY!).

"Step 2"

```
NEBLO=1;
NEBHI=19;
```

These numbers specify the lower and higher energy bins in the EBDTA(25)/...../ ("Step 1"). Now the lower energy pointer is 1 and the higher energy pointer is 19. Therefore, the lower energy bin is 0.01 MeV and the higher energy bin is 2.0 MeV. These values specify 19-(1-1)=19 successive energy bins in the 25 data in the present case.

```
REAL MEADTA (18)/...../;
```

We must give "mass energy absorption coefficent" data for the RPL glass dosimeter. The data should correspond to the energy bins we have specified in "Step 1". The number of "MEADTA", therefore, is equal to the number of energy bins we have specified just above. Now it is 19-1=18.

```
0
0    EXTERNAL FUNI;
0    EXTERNAL GOVERR;
0
0    "STEP 2"
0
0    "THE FOLLOWING INPUT DATA HAVE TO BE CHANGED EVENTUALLY"
0
0    NEBLO=1;  "LOWER ENERGY BIN POINTER"
0    NEBHI=19; "HIGHER ENERGY BIN POINTER"
0
0    "NEXT, DATA OF MASS ENERGY ABSORPTION COEFFICIENT FOR RPL GLASS"
0
0    REAL MEADTA(18)/11.0,  4.0 ,  1.6, 0.86, 0.46, 0.27, 0.15,0.087,
0    0.052,0.036,0.032,0.030,0.030,0.030,0.029,0.028,
0    0.026,0.024/;
```

```
NYBIN=9;
NZBIN=17;
```

As explained by the comments in the program, they are the number of geometrical bins in y and z directions. We have to change these numbers according to our geometry.

```
NMED=4;
```

The number of media we have specified for "MEDARR(24,X)" in "Step 1". Now X is 4, and so is NMED.

```
INTEGER IROID(21)/...../;
```

We have to specify the regions of interest where we want to find the absorbed dose to the glass dosimeter. The numbers may not be in order. The maximum number is 30 at present. See also "DIMENSION XXX(30,45)" in "Step 1".

Notice: When you increase the number of data more than 30, do not forget to change "COMMON/PASSIT/NNP(30,3,$MXEBINS),..../" in "Step 1" and in subroutines HOWFAR and AUSGAB.

```
NROI=21;
```

We have to give the number of the regions of interest. Thus, the number should be equal to X in IROID(X) specified just above.

```
0    IWTON=1;   "IWTON=0 FOR WTI=1.0"
0    "                =1 FOR WTI CALCULATED FROM SPECTRUM"
0
0    NYBIN=9; "NUMBER OF BINS IN Y-DIRECTION"
0    NZBIN=17; "NUMBER OF BINS IN Z-DIRECTION"
0    NMED=4; "NUMBER OF MEDIA"
0    DO J=1,NMED <DO I=1,24 <MEDIA(I,J)=MEDARR(I,J);>>
0
0
0    "SPECIFY THE REGION OF INTEREST"
0
0    INTEGER IROID(21)/1,6,23,24,25,42,60,87,95,96,97,113,114,115,
0    122,123,124,141,155,156,157/;                                  00016305
0
0    NROI=21;  "GIVE THE NUMBER OF REGIONS OF INTEREST: MAX IS 30)"
0    DO I=1,NROI< IROI(I)=IROID(I);>
0    IRLP=0;    "INITIALIZATION :  USED IN AUSGAB"
0
```

MED(1)=0; etc.

Here, we have to fill all the regions in our geometry with the corresponding media.
Media number "0" specifies "vacuum".
Other numbers correspond to the media array X in MEDARR(24,X) which we have specified in "Step 1" above.
Let us pay attention not to forget to fill all the regions with the appropriate media. Better, first fill all the regions with vacuum, air or whatever you think it is convenient to refill (overwrite) them soon after with proper media. With this process we will be able to avoid a mis-filling. We had better check once again after all fillings.

"Step 3"

Nothing is to be changed.

```
0
0   "CALCULATE SOME USEFUL INDICES"
0
0   NPLAN=NYBIN+NZBIN+2;   "NUMBER OF PLANES"
0   NREG=NYBIN*NZBIN+4;    "THE ACTUAL NUMBER OF REGIONS IN CALCULATION"
0   NYPLN=NYBIN+1;         "THE NUMBER OF PLANES IN Y-DIRECTION"
0   NZPLN=NZBIN+1;         "THE NUMBER OF PLANES IN Z-DIRECTION"
0
0   "NEXT, ASSIGN MEDIA TO VARIOUS REGIONS"
0
0   MED(1)=0; DO I=156,157 <MED(I)=0;>  "VACUUM OUTSIDE THE ARRAY"
0   DO I=41,43    <MED(I)=0;>           "VACUUM CHAMBER"
0   DO I=59,61    <MED(I)=0;>
0
0   DO I=29,146,9 <MED(I)=1;>           "FE : MAGNET MATERIAL"
0   DO I=30,84,9  <MED(I)=1;>
0   DO I=36,90,9  <MED(I)=1;>
0   DO I=37,154,9 <MED(I)=1;>
0   DO I=147,153  <MED(I)=1;>
0
0   DO I=13,17    <MED(I)=2;>           "LEAD SHIELDING"
0   MED(22)=2;  MED(26)=2;
0   DO I=103,107  <MED(I)=2;>
0   MED(94)=2;  MED(98)=2;
0   DO I=130,134  <MED(I)=2;>
0
0   DO I=3,21,9   <MED(I)=3;>           "AL COILS"
0   DO I=9,27,9   <MED(I)=3;>
0   DO I=93,138,9 <MED(I)=3;>
0   DO I=99,144,9 <MED(I)=3;>
0
0   DO I=31,67,9  <MED(I)=3;>           "AL CHAMBER"
0   DO I=35,71,9  <MED(I)=3;>
0   DO I=32,34    <MED(I)=3;>
0   DO I=60,62    <MED(I)=3;>
0   DO I=68,70    <MED(I)=3;>
0   MED(78)=3;
0   MED(40)=1;  MED(44)=1;              "TAKE TI GETTER INTO ACCOUNT"
0
0   DO I= 2,20,9  <MED(I)=4;>           "AIR"
0   DO I= 4, 8    <MED(I)=4;>
0   DO I=10,28,9  <MED(I)=4;>
0   DO I=23,25    <MED(I)=4;>
0   DO I=76,77    <MED(I)=4;>
0   DO I=79,80    <MED(I)=4;>
0   DO I=85,89    <MED(I)=4;>
0   DO I=96,97    <MED(I)=4;>
0   DO I=112,116  <MED(I)=4;>
0   DO I=121,125  <MED(I)=4;>
0   DO I=139,143  <MED(I)=4;>
0
0
0   "DUNIT UNCHANGED, WORKING CM UNITS BY DEFAULT"
0
0   "STEP 3"
0
0   CALL HATCH;
```

"Step 4"

Now, we have to define the coordinates and unit normals of the planes.

PCOORD(I,J)

PCOORD(I,J) represents the coordinate of the planes. I=1,2, and 3 correspond to x,y, and z axes, respectively. Thus, a set of PCOORD(1,J), PCOORD(2,J), and PCOORD(3,J) gives x,y, and z-coordinates of the "J-th" plane.

In the present program all the coordinates PCOORD(I,J) are initialized to null.

Now, we have to give our data to the PCOORD(I,J). Since all the planes in our present geometry are made to be parallel to either x-y or x-z plane, we have to specify only z and y coordinates of the planes.

So, let us at first fill z-coordinates PCOORD(3,J) from J=2 to J=(number of planes parallel to x-y plane), or 18 at present, and y-coordinates PCOORD(2,J) from J=18+2=20 to J=28 for now. The PCOORD(3,1) and PCOORD(2,19) have been already set to null by the initialization above mentioned.

Notice: Coordinates must be given in cm.

```
0
0    "STEP 4"
0
0    "NOW, ESTABLISH THE COORDINATES AND UNIT NORMALS OF THE VARIOUS PLANES"
0
0    DO J=1,NPLAN <
1         DO I=1,3 <PCOORD(I,J)=0.0;PNORM(I,J)=0.0;>
1         >
0
0
0    PCOORD(3,2)= 2.0;
0    PCOORD(3,3)= 2.3;
0    PCOORD(3,4)= 4.3;
0    PCOORD(3,5)= 5.85;    "TVACL=0.55 CM STRUCTURE OF GETTER PUMP CONSIDERED"
0    PCOORD(3,6)= 9.6;
0    PCOORD(3,7)=10.0;
0    PCOORD(3,8)=21.6;
0    TVACR=0.4;                          "THICKNESS OF RIGHT VACUUM CHAMBER WALL"
0    PCOORD(3,9)=21.6+TVACR;
0    TVACNS=1.0;                         "THICKNESS OF VACUUM CHAMBER NOSE"
0    PCOORD(3,10)=21.6+TVACR+TVACNS;
0    PCOORD(3,11)=24.8;
0    PCOORD(3,12)=26.8;
0    TPB=0.45;                           "EFFECTIVE THICKNESS OF LEAD SHIELDING"
0    PCOORD(3,13)=26.8+TPB;
0    PCOORD(3,14)=29.2;
0    PCOORD(3,15)=31.2;
0    PCOORD(3,16)=31.1;
0    PCOORD(3,17)=40.3;
0    PCOORD(3,18)=61.3;
0
0
0    PCOORD(2,20)=11.7;
0    PCOORD(2,21)=17.7;
0    PCOORD(2,22)=18.1;
0    PCOORD(2,23)=20.6;
0    PCOORD(2,24)=22.2;
0    PCOORD(2,25)=24.7;
0    PCOORD(2,26)=25.1;
0    PCOORD(2,27)=31.1;
0    PCOORD(2,28)=42.8;
0
```

PNORM(I,J)

Normally no need to be changed.

PNORM (I,J) signifies the "outward" normal vectors of the planes. Therefore, a set of PNORM(1,J), PNORM(2,J) and PNORM(3,J) gives an outward normal vector to the "J-th" plane.

In the program all the values of PNORM(I,J) are initially set to null.

Since all the planes are at present parallel to either x-y or x-z plane, as mentioned above, it follows that:

PNORM(1,1)=0, PNORM(2,1)=0, PNORM(3,1)=1,
PNORM(1,2)=0, PNORM(2,2)=0, PNORM(3,2)=1,
.....
up to the number of planes parallel to x-y plane, i.e. up to J=18 at present, then:

PNORM(1,19)=0, PNORM(2,19)=1, PNORM(3,19)=0,
PNORM(1,20)=0, PNORM(2,20)=1, PNORM(3,20)=0,
.....
until the number of the plane, or until J=28 for now.

"Step 5"

```
NNP(IOUT, IQ2, I)=0;
WNP(IOUT, IQ2, I)=0.0D0;
```

Initialization of NNP and WNP is made here. For notations of these arrays see "Step 8".

```
0
0   DO J=1,NZPLN <PNORM(3,J)=1.0;>
0   NYBEG=NZPLN+1;
0   DO J=NYBEG,NPLAN <PNORM(2,J)=1.0;>
0
0   "WRITE-OUT SOME THICKNESS PARAMETERS"
0
0   OUTPUT TVACR,TVACNS,TPB;
0   (////,' PETRA4'./
0   '    TVACR=',F5.2,' TVACNS=',F5.2,'  TPB=',F5.2,//);
0
0
0   "WRITE-OUT COORDINATES OF PLANES"
0
0   OUTPUT; (////,' COORDINATES OF PLANES:',//);
0   DO J=1,NPLAN <
1       OUTPUT J,(PCOORD(I,J),I=1,3);
1       (' J=',I3,2X,3G16.7);>
0
0   "WRITE-OUT NORMAL VECTORS OF PLANES"
0
0   OUTPUT; (////,' NORMAL VECTORS FOR PLANES:',//);
0   DO J=1,NPLAN <
1       OUTPUT J,(PNORM(I,J),I=1,3);
1       (' J=',I3,2X,3G16.7);>
0
0
0   "STEP 5"
0
0   "INITIALIZE COUNTERS NEXT"
0   ILINES=0;
0   NCOUNT=0;
0
0   "INITIALIZE COUNTERS, ETC., FOR INCIDENT SPECTRUM"
0
0   NEOUTI=0;
0   DO I=1,$MXEBINS <YSORTI(I)=0.0;  ISORTI(I)=0;>
0
0   TOTKE=0.D0;  "INITIALIZE TOTAL K.E."
0
0   OUTPUT; ('1ENERGY SORTING BINS (END-POINTS):',//);
0   NEBIN=NEBHI-NEBLO+1;  "NUMBER OF ENERGY END-POINTS"
0   DO I=1,NEBIN <
1       EBIN(I)=EBDTA(NEBLO+I-1);
1       OUTPUT I,EBIN(I);  (' EBIN(',I2,')=',G15.5);
1       >
0
0   NEBIN=NEBIN-1;      "TO CHANGE FROM END-POINTS TO BINS"
0
0   "NOW CALL ECNSV1 IN ORDER TO ZERO-OUT ARRAYS"
0   CALL ECNSV1(0,NREG,DUMMY);
0   CALL NTALLY(0,NREG);
0
0   "NEXT, INITIALIZE ARRAYS FOR SCORING LEPTON ENERGY SPECTRA"
0
0   DO IOUT=1,NRO1<
1       DO IQ2=1,3<
2           DO I=1,NEBIN<
3               NNP(IOUT,IQ2,I)=0;     "INITIALIZE"
3               WNP(IOUT,IQ2,I)=0.0D0;
3               >>>
0
```

"Step 6"

Many variables must be specified in this step.

```
EGEV=21.0;
```

We have to give an electron energy in GeV.

```
BIGR=192.05;
```

Bending radius in m.

```
IQI=0;
```

Incoming particle identification.
No need to be changed for the moment.

IQI= -1 for electron
IQI= 0 for photon
IQI= 1 for positron

```
XI=0.0;
YI=(PCOORD(2,23)+PCOORD(2,24))/2.;
ZI=PCOORD(3,8)+0.001;
```

We have to give the entrance coordinates in cm. So long as the present geometry is used there is no need to modify the above equations.

The second term in the right hand side of the last equation is added to specify clearly the region where the synchrotron radiation hit the vacuum chamber wall. Without this figure the incoming particle might impinge in the opposite

```
0
0   "STEP 6"
0
0   "SYNCHROTRON RADIATION SPECTRUM DATA NEXT"
0
0   EGEV=17.0;  "ELECTRON BEAM ENERGY (GEV)"
0   BIGR=192.06;  " BEND RADIUS (METERS)"
0   ECRIT=2.218E-3*EGEV**3/BIGR;   "CRITICAL ENERGY (MEV)"
0
0   ESAM1=EBDTA(NEBLO);     "LOWER SAMPLING ENERGY (MEV)"
0   ESAM2=EBDTA(NEBHI);     "UPPER SAMPLING ENERGY (MEV)"
0   DELSAM=ESAM2-ESAM1;
0   SMALR1=ESAM1/ECRIT;
0   SMALR2=ESAM2/ECRIT;
0
0   IF(IWTON.EQ.1)<"CALCULATE FNORM BECAUSE WTI TO BE CALCULATED"
1
1       FINT1=DCADRE(FUN1,SMALR1,SMALR2,1.D-60,1.D-5,ERRDUM,IER);
1       IF(IER.GT.66)<OUTPUT IER,SMALR1,SMALR2,FINT1,ERRDUM;
2            (' DCADRE CODE =',I4,' FROM ',D20.10,' TO ',D20.10,
2            ' FINT1=',D20.10,'+-',D20.10);>
1
1       C1=1.496E23*1.6E-19/2.0/PI/EGEV**2;
1
1       FNORM=ECRIT*C1*FINT1;
1       >
0
0   ELSE<"WEIGHT WILL BE UNITY AND FNORM=1"
1       FINT1=1.0;
1       FNORM=1.0;
1       >
0
0   "NOTE:  FNORM WILL BE DIVIDED BY THE SUM OF THE WEIGHTS LATER"
0   "       (CALCULATION IS FOR ONE ELECTRON AND ONE METER OF BEND)"
0
0   IQI=0;   "INCOMING PARTICLE ID"
0
0   XI=0.0;                              "ENTRANCE COORDINATES (CM)"
0   YI=(PCOORD(2,23)+PCOORD(2,24))/2.;
0   ZI=PCOORD(3,8)+0.0001;
```

region across the plane (now plane number J=8) because any computer has a round error in computation, which is far smaller than the above value.

IRI=69;

We have to specify the region where synchrotron radiation enter. Region number 69 is the incident region at present.

THRAD=0.0244;

Incident angle in radians of photons to the vacuum chamber with respect to the negative x-direction.

IXX=123456781

Rundom number generation seed.
This number must end in odd number!

"Step 7"

We do not have to change anything in this step for the moment.

```
U
0   DO I=1,NZPLN <TVAR=ZI-PCOORD(3,I); IF(TVAR.LE.0.) <INDZ=I;EXIT;>>
0   DO I=NYBEG,NPLAN <TVAR=PCOORD(2,I)-YI; IF(TVAR.LE.0.) <INDY=I;EXIT;>>
0
0   IRI=69; "ENTRANCE REGION"
0   THRADI=0.0244;  "INCIDENT ANGLE IN RADIANS"
0
0   THETAI=THRADI*180./PI;  "INCIDENT ANGLE IN DEGREES"
0   UI=COS(THRADI); "X-DIRECTION COSINE"
0   VI=0.0;  "Y-DIRECTION COSINE"
0   WI=SIN(THRADI);  "Z-DIRECTION COSINE"
0   IDINC=-1; "ID (IARG) FOR INCIDENT PARTICLE I.D."
0   NWRITE=10;  "NO. OF SHOWER CASES TO PRINT OUT"
0   NLINES=100; "NO. OF LINES TO PRINT OUT"
0   IXX=123456781; "RANDOM NO. GENERATOR SEED....MUST END IN ODD NO"
0   IXXST=IXX; "TO KEEP TRACK OF STARTING SEED"
0   TYMOUT=16.0;  "A PARAMETER TO DEFINE TIME PRIOR TO CLEAN-UP"
0   NCASES=1000000; "NO. OF CASES TO RUN---LET TIMER CODE DETERMINE OPTIMUM"
0
0   "WRITE-OUT TITLES"
0
0   OUTPUT; ('1ENERGY/COORDINATES/DIRECTION COSINES/ETC.',/,
0   '    (FIRST HUNDRED OR SO LINES)    ',//);
0
0   OUTPUT; (5X,'E',12X,'X',12X,'Y',12X,'Z',12X,'U',12X,'V',
0   12X,'W',9X,'IQ',3X,'IR',1X,'IARG',4X,'WT',/);
0
0   "STEP 7"
0
0   WTISUM=0.0;
0
0   DO I=1,NCASES <"START OF SHOWER GENERATION LOOP"
1
1       "FIRST, DETERMINE IF THERE IS ENOUGH TIME LEFT"
1
1       CALL TIMEL(TYMNOW);
1       IF(TYMNOW.LT.TYMOUT) <EXIT;> "EXIT THE SHOWER CALL LOOP"
1       "                              AND OUTPUT THE RESULTS"
1
1       "SAMPLE ENERGY UNIFORMLY OVER ENERGY RANGE"
1
1       $RANDOMSET RANNO;
1       EVAL=ESAMI+RANNO*DELSAM;
1       EI=EVAL;
1
1       IF(IWTGN.EQ.1)<"CALCULATE THE WEIGHT"
2           SMALR=EVAL/ECRIT;
2           WTI=GOVERR(SMALR);
2           WTINDP=WTI;
2           EKIN=EI*WTINDP;
2           >
1
1       ELSE<"WEIGHT IS UNITY"
2           EKIN=EI;
2           WTI=1.0;  "WEIGHT OF INCIDENT PARTICLE"
2           >
1
1       WTISUM=WTISUM + WTI;
1
1       CALL ESORT(EVAL,IEBIN);
1       IF(IEBIN.NE.0) <
2           YSORTI(IEBIN)=YSORTI(IEBIN) + WTI;
2           ISORTI(IEBIN)=ISORTI(IEBIN) + 1;
2           >
1       ELSE <"COUNT THE ONES THAT COULD NOT BE SORTED"
2           NEOUTI=NEOUTI + 1;
2           >
1
1       IF(NCOUNT.LE.NWRITE.AND.ILINES.LE.NLINES) <
2           OUTPUT EI,XI,YI,ZI,UI,VI,WI,
2           IQI,IRI,IDINC,WTI;   (7G13.5,314,G13.5);>
1
1       CALL SHOWER(IQI,EI,XI,YI,ZI,UI,VI,WI,IRI,WTI);
1
1       TOTKE=TOTKE+EKIN;  "KEEP SUM OF K.E"
1       NCOUNT=NCOUNT+1;   "COUNT THE NUMBER OF SHOWER CASES GENERATED"
1       ILINES=ILINES+1;  "COUNT THE NUMBER OF LINES PRINTED OUT"
1       IXXEND=IXX;  "THIS KEEPS TRACK OF THE LAST RANDOM NUMBER INTEGER"
1       "                THAT WAS USED---IN CASE WE NEED IT LATER ON"
1
1       "END OF SHOWER GENERATION LOOP">
0
```

"Step 8"

Output of the results is controlled in this step. We do not have to change any parts unless we want to modify the output format.

WNP(IOUT, IQ2, I)

WNP(IOUT, IQ2, I) array registers the sum over each energy bin of the individual "weight" with which each lepton, i.e. electron, photon or positron is transported. The array consists of IOUT, IQ2 and I,
where

IOUT = array of the regions of interest, ROI. Maximum value of IOUT is NROI (see "Step 2").

IQ2 = 1 for electron,
= 2 for photon,
= 3 for positron.

I = energy bin number. The maximum of I is equal to the number of energy bins.

The output of WNP(IOUT, IQ2, I) for IQ2=1, 2, and 3 gives the energy fraction spectrum for electrons, photons and positrons, respectively.

NNP(IOUT, IQ2, I)

This array registers the absolute number of leptons that drop into each energy bin. The notations for IOUT, IQ2 and I are the same as above.

```
0
0    "STEP 8"
0
0    IF(TOTKE.EQ.0.D0) <"SOMETHING IS WRONG.....OUTPUT INFO. AND STOP"
1        OUTPUT TOTKE,EKIN,EI,PRM,NCOUNT;
1        (' $$$$$ TOTKE,EKIN,EI,PRM=',4G15.7,5X,'NCOUNT=',I6);
1        STOP;>
0
0    "OTHERWISE, WRITE-OUT VARIOUS QUANTITIES OF INTEREST"
0
0    OUTPUT NCOUNT,NCASES,IQI,TOTKE:  ('1SUMMARY:',//,
0    I8,' CASES OUT OF ',I8,' WERE COMPLETED.',//,
0    ' INCIDENT PARTICLE TYPE=',I2,/,
0    ' TOTAL K.E. IN RUN=',G15.7,' MEV');
0
0    OUTPUT EGEV,BIGR,ECRIT:  (/,' SYNCHROTRON SPECTRUM INPUT DATA:',//,
0    ' EGEV=',G16.5,' GEV',5X,'BIGR=',G15.5,' METERS',5X,
0    'ECRIT=',G16.5,' MEV');
0
0    FNORM=FNORM/WTISUM;
0
0    OUTPUT ESAM1,ESAM2,SMALR1,SMALR2,
0    FINT1,WTISUM,FNORM;
0    (//,' SPECTRUM SAMPLING INFORMATION:',
0    /,' ESAM1=',G16.5,' MEV',3X,'ESAM2=',G16.5,' MEV',//,
0    ' SMALR1=',G15.5,3X,'SMALR2=',G15.5,/,
0    ' FINT1=',G16.5,3X,'WTISUM=',G15.5,3X,'FNORM=',G16.5);
0
0
0    "OUTPUT INCIDENT SPECTRUM SAMPLED"
0
0    OUTPUT:  (//,' INCIDENT SPECTRUM (SAMPLED AND THEORETICAL)',
0    ' (PHOTONS/MEV/METER/ELEC):',/);
0
0    DO I=1,NEBIN<
1        EAVE(I)=(EBIN(I)+EBIN(I+1))/2.0;
1        SMALRA=EAVE(I)/ECRIT;
1        THSPEC(I)=C1*GOVERR(SMALRA);
1        DELTAE=EBIN(I+1)-EBIN(I);
1        YSORTI(I)=FNORM*YSORTI(I)/DELTAE;
1        ISORTI(I)=ISORTI(I)/DELTAE;
1        OUTPUT EBIN(I),EBIN(I+1),YSORTI(I),ISORTI(I),THSPEC(I);
1        (G15.5,' TO',G16.5,' MEV',5X,G15.5,6X,'(',I9,' EVENTS/MEV)',
1        3X,'S=',G15.5);
1        >
0
0    OUTPUT NEOUTI:  (///,' NEOUT=NO. THAT COULD NOT BE ENERGY-SORTED=',I6);
0
0    CALL ECNSV1(1,NREG,TOTKE); "THIS CALCULATES AND PRINTS OUT THE ENERGY"
0    "                          CONSERVATION RESULTS"
0
0    CALL NTALLY(1,NREG);
0
0    "FOLLOWING STRINGS ARE FOR GETTING LEPTON ENERGY SPECTRUM"
0
0    OUTPUT IXXST; (///////,' RANDOM NO. GENERATION SEED =', I10 );
0
0    DO IOUT=1,NROI<
1        SWNP=0.0D0;
1        SXXX=0.0D0;
1        IIROI=IROI(IOUT);
1        OUTPUT IIROI;( //, ' FRACTION OF ENERGY CARRIED BY PARTICLE',
1        ' INTO THE REGION OF INTEREST: IROI=',
1        I3,//,28X,
1        ' ELECTRON',3X,' PHOTON',5X,' POSITRON',10X,
1        ' ELECTRON',3X,' PHOTON',5X,' POSITRON',6X,
1        ' KERMA(I)'/);
1        DO I=1,NEBIN<
2            DELTAE=EBIN(I+1)-EBIN(I);
2
2            DO IQ2=1,3<
3                WNP(IOUT,IQ2,I)=WNP(IOUT,IQ2,I)/TOTKE;
3                SWNP=SWNP+WNP(IOUT,IQ2,I);
3                >
2
2            XXX(IOUT,I)=MEADTA(I)*WNP(IOUT,2,I);
2            SXXX=SXXX+XXX(IOUT,I);
2
2            OUTPUT EBIN(I),EBIN(I+1),(WNP(IOUT,IQ2,I),IQ2=1,3),
2            (NNP(IOUT,IQ2,I),IQ2=1,3),XXX(IOUT,I);
2            (F9.3,' TO',F9.3,' MEV',6G12.4,10X,1G12.4);
2            >
1
1        OUTPUT SWNP,SXXX; (/, ' TOTAL ENERGY FRACTION THAT THE',
1        ' PARTICLE BRINGS IN :   SWNP=', G12.4,10X,
1        ' KERMA*(A CM**2)/MEV/EL.=',G12.4);
1
1
1        >
0
```

Note: The dimensions of WNP and NNP are defined in "Step 1" as:

WNP(30,3,$MXEBINS), and
NNP(30,3,$MXEBINS).

Do not forget that they are also transferred through COMMON/PASSIT/...../ to subroutine HOWFAR and AUSGAB.

SWNP

This variable is to output the sum of WNP over the whole energy bins. The value of SWNP is used as one of the criteria for the error estimation in the Monte Carlo calculation. When it is lower than 1×10^{-5}, the error is usually pretty large.

XXX (IOUT, I)

This array is to output the value of "Kerma" for the individual energy bins.

IOUT = array of the regions of interest, ROI.
Maximum value of IOUT is NROI (see "Step 2").
I = energy bin number.

SXXX

This variable is to sum up the values of "Kerma" over the whole energy bins. The value is used for the calculation of the absorbed dose to the RPL glass dosimeter (Chapter 2).

```
      "FOLLOWING STRINGS ARE FOR GETTING LEPTON ENERGY SPECTRUM"

      OUTPUT IXXST; (///////,' RANDOM NO. GENERATION SEED =', I10 );

      DO IOUT=1,NROI<
           SWNP=0.0D0;
           SXXX=0.0D0;
           IIROI=IROI(IOUT);
           OUTPUT IIROI;( //, ' FRACTION OF ENERGY CARRIED BY PARTICLE',
           ' INTO THE REGION OF INTEREST: IROI=',
           I3,//,28X,
           ' ELECTRON',3X,' PHOTON',5X,' POSITRON',10X,
           ' ELECTRON',3X,' PHOTON',5X,' POSITRON',6X,
           ' KERMA(I)'/);
           DO I=1,NEBIN<
                DELTAE=EBIN(I+1)-EBIN(I);

                DO IQ2=1,3<
                     WNP(IOUT,IQ2,I)=WNP(IOUT,IQ2,I)/TOTKE;
                     SWNP=SWNP+WNP(IOUT,IQ2,I);
                     >

                XXX(IOUT,I)=MEADTA(I)*WNP(IOUT,2,I);
                SXXX=SXXX+XXX(IOUT,I);

                OUTPUT EBIN(I),EBIN(I+1),(WNP(IOUT,IQ2,I),IQ2=1,3),
                (NNP(IOUT,IQ2,I),IQ2=1,3),XXX(IOUT,I);
                (F9.3,' TO',F9.3,' MEV',6G12.4,10X,1G12.4);
                >

           OUTPUT SWNP,SXXX; (/, ' TOTAL ENERGY FRACTION THAT THE',
           ' PARTICLE BRINGS IN :   SWNP=', G12.4,10X,
           ' KERMA*(A CM**2)/MEV/EL.=',G12.4);


           >

      STOP;
      END;
```

"SUBROUTINE HOWFAR"

So long as the present two dimensional HOWFAR is used, there is normally nothing to be changed in the subroutine HOWFAR. But for further modification it would be useful to do a brief survey of the HOWFAR. This subroutine is a geometry routine and it defines "how far" the particle is transported in each process. For detailed knowledge the reader is required to read ref. 1. When the way of numbering the regions and planes is different from what is mentioned above, the subroutine HOWFAR must be modified.

```
IRL = IR(NP);
```

IRL is a local variable for IR(NP). IR(NP) is the index of the particle's current region. NP is the stack pointer or the number of particles on the stack.

```
IOUTRG = NREG-2;
```

NREG is the number of total regions. Thus, as we can see easily from the equation, IOUTRG is the number of the outer-most region which has the smallest region number (except region number 1).

Since NREG=157 at the moment, IOUTRG=157-2 =155. The region number 155 corresponds to the region opposite to region 1 (see Fig. 2).

```
IF(IRL.EQ.1.OR.IRL.GE.IOUTRG) <IDISC=1; RETURN; >
```

This string is used to discard the particles which come out into the regions outside the magnet component, which makes a computation faster.

```
0    SUBROUTINE HOWFAR;
0
0    COMIN/DEBUG,EPCONT,PLADTA,STACK/;
0    COMMON/PASSIT/HNP(30,3,$MXEBINS),WNP(30,3,$MXEBINS),IROI(30),
0    NROI,ESAM1,NREG,NYBIN,NZBIN,INDY,INDZ,IRLP;                          00050700
0    REAL*8 WNP;
0
0    IRL=IR(NP);   "SET LOCAL VARIABLE"
0    IOUTRG=NREG-2;   "IOUTRG : THE FIRST OUTER REGION"
0
0    IF(IRL.EQ.1.OR.IRL.GE.IOUTRG) <IDISC=1; RETURN;>
0
0    I=(IRL-2)/NYBIN+1;   "COLUMN NUMBER"
0    J=IRL-1-NYBIN*(I-1);      "ROW NUMBER"
0
0    NPL1=I+1;   NPL2=1;      "Z-DIRECTION"
0    IF(I.LT.NZBIN) <NRG1=IRL+NYBIN;> ELSE <NRG1=IOUTRG;>
0    IF(I.GT.1) <NRG2=IRL-NYBIN;> ELSE <NRG2=1;>
0    IF(NPL1.GT.INDZ) <$PLANE2(NPL1,NRG1,1,NPL2,NRG2,-1);>
0    ELSE <$PLANE2(NPL2,NRG2,-1,NPL1,NRG1,1);>
0
0    NPL1=NZBIN+2+J; NPL2=NPL1-1;   "Y-DIRECTION"
0    IF(J.LT.NYBIN) <NRG1=IRL+1;> ELSE <NRG1=IOUTRG+2;>
0    IF(J.GT.1) <NRG2=IRL-1;> ELSE <NRG2=IOUTRG+1;>
0    IF(NPL1.GT.INDY) <$PLANE2(NPL1,NRG1,1,NPL2,NRG2,-1);>
0    ELSE <$PLANE2(NPL2,NRG2,-1,NPL1,NRG1,1);>
0
0
0    RETURN;
0    END;
0
     XE
```

The strings which follow below are to find the region into which the current particle will enter.

$PLANE2

This is a macro replacement for subroutine PLANE2 (see "Step 1"). The particle's current region IRL is surrounded by four planes, i.e. NPL1 and NPL2 in both y and z directions (see Fig. 3). Around IRL there are four regions specified by NRG1 and NRG2 in both y and z directions. The subroutine PLANE2 is used to find the region into which the current particle will enter. For detail the reader is requested to read APPENDIX UC of ref. 1.

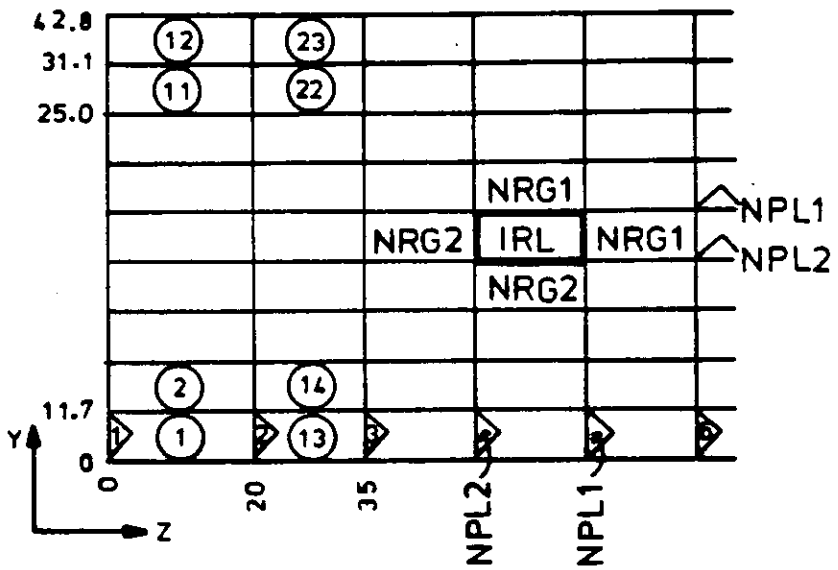

Fig. 3 Relation among particle's current region IRL and its four surrounding planes, i.e. NPL1 and NPL2 in both y and z directions. Four regions specified by NRG1 and NRG2 in both y and z directions are also shown (see subroutine HOWFAR).

```
SUBROUTINE HOWFAR;

COMIN/DEBUG,EPCON1,PLADTA,STACK/;
COMMON/PASSIT/NNP(30,3,$MXEBINS),WNP(30,3,$MXEBINS),IROI(30),
NROI,ESAM1,NREG,NYBIN,NZBIN,INDY,INDZ,IRLP;
REAL*8 WNP;

IRL=IR(NP);   "SET LOCAL VARIABLE"
IOUTRG=NREG-2;   "IOUTRG : THE FIRST OUTER REGION"

IF(IRL.EQ.1.OR.IRL.GE.IOUTRG) <IDISC=1; RETURN;>

I=(IRL-2)/NYBIN+1;   "COLUMN NUMBER"
J=IRL-1-NYBIN*(I-1);      "ROW NUMBER"

NPL1=I+1;   NPL2=I;      "Z-DIRECTION"
IF(I.LT.NZBIN) <NRG1=IRL+NYBIN;> ELSE <NRG1=IOUTRG;>
IF(I.GT.1) <NRG2=IRL-NYBIN;> ELSE <NRG2=1;>
IF(NPL1.GT.INDZ) <$PLANE2(NPL1,NRG1,1,NPL2,NRG2,-1);>
ELSE <$PLANE2(NPL2,NRG2,-1,NPL1,NRG1,1);>

NPL1=NZBIN+2+J; NPL2=NPL1-1;   "Y-DIRECTION"
IF(J.LT.NYBIN) <NRG1=IRL+1;> ELSE <NRG1=IOUTRG+2;>
IF(J.GT.1) <NRG2=IRL-1;> ELSE <NRG2=IOUTRG+1;>
IF(NPL1.GT.INDY) <$PLANE2(NPL1,NRG1,1,NPL2,NRG2,-1);>
ELSE <$PLANE2(NPL2,NRG2,-1,NPL1,NRG1,1);>


RETURN;
END;

%E
```

"SUBROUTINE AUSGAB"

There is no need to change this subroutine. It is used to sort and score the particles which enter the regions of interest that the user have specified in "Step 2".

```
0    SUBROUTINE AUSGAB(IARG);
0
0    COMIN/DEBUG,EPCONT,ETALY1,NTALY1,STACK/;
0    COMMON/LINES/NLINES,NWRITE,NCOUNT,ILINES;
0    COMMON/PASSIT/NNP(30,3,$MXEBINS),WNP(30,3,$MXEBINS),IROI(30),
0    NROI,ESAM1,NREG,NYBIN,NZBIN,INDY,INDZ,IRLP;                    00053320
0    COMMON/EDATA/EBIN($MXEBINS),NEBIN;
0    REAL*8 WTINDP,WNP;
0
0    IRL=IR(NP);       "SET LOCAL VARIABLE"
0    WTINDP=WT(NP);   "DEFINE DOUBLE PRECISION WEIGHT"
0
0    ESUM(IQ(NP)+2,IR(NP),IARG+1)=ESUM(IQ(NP)+2,IR(NP),IARG+1)+EDEP*WTINDP;
0    NSUM(IQ(NP)+2,IR(NP),IARG+1)=NSUM(IQ(NP)+2,IR(NP),IARG+1) + 1;
0
0    IF(NCOUNT.LE.NWRITE.AND.ILINES.LE.NLINES) <
1        OUTPUT E(NP),X(NP),Y(NP),Z(NP),U(NP),V(NP),W(NP),
1        IQ(NP),IR(NP),IARG,WT(NP);    (7G13.6,3I4,G13.6);
1
1        ILINES=ILINES+1;>
0
0    "*****************************************  FOR LEPTON SPECTRA  *****"
0    DO I=1,NROI<
1        IROIL=IROI(I);      "SET LOCAL VARIABLE"
1
1        IF(IRL.NE.IROIL) <IRLP=0; NEXT;>
1        IF(IRL.NE.IRLP.AND.(IARG.EQ.0.OR.IARG.EQ.3))<
2            IOUT=I; IRLP=IRL "KEEP PREVIOUS IRL"; GO TO :ADD:;
2            >
1        RETURN;
1        >
0
0    RETURN;
0    :ADD:
0    IQ2=IQ(NP)+2;      "IQ2=1 FOR ELECTRON"
0    "                       =2      PHOTON"
0    "                       =3      POSITRON"
0
0    IF(E(NP).GT.ESAM1) <MEBIN=NEBIN+1;
1        DO I=2,MEBIN <DE=E(NP)-EBIN(I);
2            IF(DE.LE.0.)<
3                NNP(IOUT,IQ2,I-1)=NNP(IOUT,IQ2,I-1)+1;
3                WNP(IOUT,IQ2,I-1)=WNP(IOUT,IQ2,I-1)+WTINDP*E(NP);
3                EXIT;
3                > >   >
0
0    RETURN;
0    END;
0
```

3.3 Execution of the program

The member PETRA4 is executed by submitting the member 'DO3CYA.EGS.S(XUC)'. The list of the member XUC is given below.

```
DO3CYA.EGS.S(XUC)

//DO3CYA76 JOB CLASS=L,TIME=(16,00)
//*MAIN ORG=EXT,LINES=(5)
// EXEC NEWFAST
// EXEC MORTRAN
%MACRO MMACRO
%MACRO PETRA4
%MACRO BLKDATA
// EXEC FCLG,LEVEL=8,LIB1='DO3CYA.EGS.L',LIB2='R01UTL.CERNLIB',
// REGION.GO=900K
//FORT.SYSIN DD DSN=&&FORTIN,DISP=SHR
//GO.FT12F001 DD UNIT=FAST,DISP=SHR,DSN=DO3CYA.EGSUC20
//GO.FT08F001 DD DUMMY
```

3.4 How to read the output

COORDINATES OF PLANES

The x,y, and z coordinates of the planes specified in "Step 2" are listed. J is the plane number.

NORMAL VECTORS FOR PLANES

The normal vectors of the planes are listed. J is the plane number.

```
DUNIT REQUESTED&USED ARE:    1.000000E+00    1.000000E+00 (CM.)
EGS SUCCESSFULLY MATCHED FOR      4 MEDIA.


PETRA4
  TVACR= 0.40 TVACNS= 1.00   TPB= 0.45


COORDINATES OF PLANES:

J=  1        .0              .0              .0
J=  2        .0              .0              2.000000
J=  3        .0              .0              2.300000
J=  4        .0              .0              4.300000
J=  5        .0              .0              5.850000
J=  6        .0              .0              9.600000
J=  7        .0              .0              10.00000
J=  8        .0              .0              21.60001
J=  9        .0              .0              22.00000
J= 10        .0              .0              23.00000
J= 11        .0              .0              24.80000
J= 12        .0              .0              26.80000
J= 13        .0              .0              27.25000
J= 14        .0              .0              29.20000
J= 15        .0              .0              31.20000
J= 16        .0              .0              31.10001
J= 17        .0              .0              40.30000
J= 18        .0              .0              51.30000
J= 19        .0              .0              .0
J= 20        .0              11.70000        .0
J= 21        .0              17.70000        .0
J= 22        .0              18.10001        .0
J= 23        .0              20.60001        .0
J= 24        .0              22.20000        .0
J= 25        .0              24.70000        .0
J= 26        .0              25.10001        .0
J= 27        .0              31.10001        .0
J= 28        .0              42.80000        .0

NORMAL VECTORS FOR PLANES:

J=  1        .0              .0              1.000000
J=  2        .0              .0              1.000000
J=  3        .0              .0              1.000000
J=  4        .0              .0              1.000000
J=  5        .0              .0              1.000000
J=  6        .0              .0              1.000000
J=  7        .0              .0              1.000000
J=  8        .0              .0              1.000000
J=  9        .0              .0              1.000000
J= 10        .0              .0              1.000000
J= 11        .0              .0              1.000000
J= 12        .0              .0              1.000000
J= 13        .0              .0              1.000000
J= 14        .0              .0              1.000000
J= 15        .0              .0              1.000000
J= 16        .0              .0              1.000000
J= 17        .0              .0              1.000000
J= 18        .0              .0              1.000000
J= 19        .0              1.000000        .0
J= 20        .0              1.000000        .0
J= 21        .0              1.000000        .0
J= 22        .0              1.000000        .0
J= 23        .0              1.000000        .0
J= 24        .0              1.000000        .0
J= 25        .0              1.000000        .0
J= 26        .0              1.000000        .0
J= 27        .0              1.000000        .0
J= 28        .0              1.000000        .0
```

ENERGY SORTING BINS (END-POINTS):

Energy sorting bins specified in "Step 2" are listed.

ENERGY SORTING BINS (END-POINTS):

```
EBIN( 1)=      .10000E-01
EBIN( 2)=      .15000E-01
EBIN( 3)=      .20000E-01
EBIN( 4)=      .30000E-01
EBIN( 5)=      .40000E-01
EBIN( 6)=      .50000E-01
EBIN( 7)=      .60000E-01
EBIN( 8)=      .80000E-01
EBIN( 9)=      .10000
EBIN(10)=      .15000
EBIN(11)=      .20000
EBIN(12)=      .30000
EBIN(13)=      .40000
EBIN(14)=      .50000
EBIN(15)=      .60000
EBIN(16)=      .80000
EBIN(17)=      1.0000
EBIN(18)=      1.5000
EBIN(19)=      2.0000
```

ENERGY/COORDINATES/DIRECTION COSINES/ETC.:

First hundred or so lines of variables are printed.

X, Y, Z	Position of particles in units established by DUNIT (usually in cm).
U, V, W	Direction cosines of particles.
IQ	Integer charge of particle (-1 for electron, 0 for photon, +1 for positron).
IR	Index of particle's current region.
IARG	Integer argument that indicates the situation under which the output subroutine AUSGAB is being called (see Table 4.6.1 of ref. 1).
WT	Statistical weight of current particle.

ENERGY/COORDINATES/DIRECTION COSINES/ETC.
(FIRST HUNDRED OR SO LINES)

E	X	Y	Z	U	V	W	IQ	IR	IARG	WT
1.3242	.0	21.400	21.600	.99970	.0	.24398E-01	0	69	-1	.91074E-11
1.3242	.0	21.400	21.600	.99970	.0	.24398E-01	0	69	0	.91074E-11
1.3242	16.366	21.400	22.000	.99970	.0	.24398E-01	0	78	0	.91074E-11
.70031	21.654	21.400	22.125	.66003	.73639	-.14862	0	78	0	.91074E-11
.46396	21.691	21.442	22.120	-.89132E-01	.97458	.20556	0	78	0	.91074E-11
.46396	21.622	22.200	22.280	-.89132E-01	.97458	.20556	0	79	0	.91074E-11
.46396	21.393	24.700	22.607	-.89132E-01	.97458	.20556	0	80	0	.91074E-11
.46396	21.357	25.100	22.892	-.89132E-01	.97458	.20556	0	81	0	.91074E-11
.46396	21.310	25.613	23.000	-.89132E-01	.97458	.20556	0	90	0	.91074E-11
.20226	21.214	26.666	23.220	.85897	-.26527	-.43794	0	90	0	.91074E-11
.14148	21.477	26.577	23.066	-.58153	-.49276	-.64729	0	90	0	.91074E-11
.14148	21.400	26.511	23.000	-.58153	-.49276	-.64729	0	81	0	.91074E-11
.71120D-02	20.601	26.004	22.334	-.58153	-.49276	-.64729	0	81	4	.91074E-11
.64537	20.601	26.004	22.334	-.58153	-.49276	-.64729	-1	81	2	.91074E-11
.67176	21.477	26.577	23.066	.99796	.62617E-01	.11702E-01	-1	90	2	.91074E-11
.77271	21.214	26.666	23.220	-.37109	.87268	.31736	-1	90	2	.91074E-11
.74735	21.691	21.442	22.120	.92340	.11650	-.36572	-1	78	2	.91074E-11
1.1349	21.654	21.400	22.125	.85024	-.50891	.13469	-1	78	2	.91074E-11
.34110	.0	21.400	21.600	.99970	.0	.24398E-01	0	69	-1	.64512E-03
.34110	.0	21.400	21.600	.99970	.0	.24398E-01	0	69	0	.64512E-03
.21220	.84243	21.400	21.621	.69997E-01	.56693	.82078	0	69	0	.64512E-03
.21220	.87478	21.662	22.000	.69997E-01	.56693	.82078	0	78	0	.64512E-03
.21220	.94121	22.200	22.779	.69997E-01	.56693	.82078	0	79	0	.64512E-03
.21220	.96007	22.383	23.000	.69997E-01	.56693	.82078	0	86	0	.64512E-03
.21220	1.1136	23.896	24.800	.69997E-01	.56693	.82078	0	97	0	.64512E-03
.21220	1.2499	24.700	26.398	.69997E-01	.56693	.82078	0	96	0	.64512E-03
.14664	1.2539	24.732	26.446	-.69991	.57150	-.42836	0	98	0	.64512E-03
.88000D-01	1.2468	24.736	26.441	-.69991	.57150	-.42836	0	98	4	.64512E-03
.56965	1.2468	24.738	26.441	-.69991	.57150	-.42836	-1	98	2	.64512E-03
.57656	1.2539	24.732	26.446	.44001	-.13668	.88783	-1	98	2	.64512E-03
.63990	.84243	21.400	21.621	.84677	-.31234	-.43060	-1	69	2	.64512E-03
1.4760	.0	21.400	21.600	.99970	.0	.24398E-01	0	69	-1	.57193E-12
1.4760	.0	21.400	21.600	.99970	.0	.24398E-01	0	69	0	.57193E-12
.23990	2.0674	21.400	21.651	-.78818	-.59748	.14756	0	69	0	.57193E-12
.23990	1.0321	20.000	21.649	-.78818	-.59748	.14756	0	68	0	.57193E-12
.23990	.22339	19.587	22.000	-.78818	-.59748	.14756	0	77	0	.57193E-12
.23990	-2.2689	16.100	22.466	-.78818	-.59748	.14756	0	76	0	.57193E-12
.23990	-2.7535	17.700	22.561	-.78818	-.59748	.14756	0	75	0	.57193E-12
.23990	-6.1161	15.938	23.000	-.78818	-.59748	.14756	0	84	0	.57193E-12
.16315	-5.5610	15.602	23.063	-.60970	.73344	-.30051	0	84	0	.57193E-12
.16316	-5.7291	15.804	23.000	-.60970	.73344	-.30051	0	75	0	.57193E-12
.12973	-6.1016	16.253	22.816	-.37666	-.39329	-.83870	0	75	0	.57193E-12
.71120D-02	-6.2622	16.064	22.414	-.37666	-.39329	-.83870	0	75	4	.57193E-12
.63362	-6.2822	16.064	22.414	-.37666	-.39329	-.83870	-1	75	2	.57193E-12
.54442	-6.1018	16.253	22.816	-.26947	.90886	.31632	-1	75	2	.57193E-12
.58775	-5.5610	15.602	23.063	-.30856	-.90567	.29074	-1	84	2	.57193E-12
1.7491	2.0874	21.400	21.651	.99632	.85689E-01	.39350E-03	-1	69	0	.57193E-12
1.4990	2.1303	21.404	21.651	.74886	.30347	-.58915	-1	69	2	.57193E-12
1.4622	.0	21.400	21.600	.99970	.0	.24398E-01	0	69	-1	.75911E-12
1.4622	.0	21.400	21.600	.99970	.0	.24398E-01	0	69	0	.75911E-12
.92352	5.4518	21.400	21.733	.72254	.68918	.64489E-01	-1	69	2	.75911E-12
1.0497	5.4518	21.400	21.733	.86307	-.50505	-.69457E-02	0	69	0	.75911E-12
1.0497	6.8189	20.600	21.724	.86307	-.50505	-.69457E-02	0	68	0	.75911E-12
.26257	6.1859	19.800	21.714	-.79703	-.44773	-.40530	0	68	0	.75911E-12
.26257	7.9612	19.674	21.600	-.79703	-.44773	-.40530	0	59	0	.75911E-12
.26257	6.1596	16.100	20.175	-.79703	-.44773	-.40530	0	58	0	.75911E-12

INCIDENT SPECTRUM:

Sampled and theoretical spectra of synchrotron radiation are listed. For each energy bin the photon number is given, which is normalized to a photon energy width of 1 MeV, 1m of magnet length, and one electron. Since the sampling is done equally over the energy region which the user has specified, the figures in the parenthesis, i.e. EVENTS/MEV, for the individual energy bins should be in the same order of magnitude.

SUMMARY:

16627 CASES OUT OF 1000000 WERE COMPLETED.

INCIDENT PARTICLE TYPE= 0
TOTAL K.E. IN RUN= 16.65961 MEV

SYNCHROTRON SPECTRUM INPUT DATA:
EGEV= 17.000 GEV BIGR= 192.05 METERS ECRIT= .66741E-01 MEV

SPECTRUM SAMPLING INFORMATION:
ESAM1= .10000E-01 MEV ESAM2= 2.0000 MEV
SMALR1= .17624 SMALR2= 35.248
FINT1= .90119 WTISUM= 405.74 FNORM= .16613E-02

INCIDENT SPECTRUM (SAMPLED AND THEORETICAL) (PHOTONS/MEV/METER/ELEC):

.10000E-01	TO	.15000E-01	MEV	19.610	(	6200	EVENTS/MEV)	S=	26.330
.15000E-01	TO	.20000E-01	MEV	17.304	(	7400	EVENTS/MEV)	S=	18.237
.20000E-01	TO	.30000E-01	MEV	12.902	(	8199	EVENTS/MEV)	S=	12.396
.30000E-01	TO	.40000E-01	MEV	9.5641	(	9400	EVENTS/MEV)	S=	8.1950
.40000E-01	TO	.50000E-01	MEV	8.9066	(	7999	EVENTS/MEV)	S=	5.7649
.50000E-01	TO	.60000E-01	MEV	4.4802	(	8500	EVENTS/MEV)	S=	4.2080
.60000E-01	TO	.80000E-01	MEV	3.0531	(	8650	EVENTS/MEV)	S=	2.7411
.80000E-01	TO	.10000	MEV	1.9265	(	9349	EVENTS/MEV)	S=	1.6286
.10000	TO	.15000	MEV	.82865	(	8440	EVENTS/MEV)	S=	.70919
.15000	TO	.20000	MEV	.26341	(	8199	EVENTS/MEV)	S=	.23744
.20000	TO	.30000	MEV	.62201E-01	(	8129	EVENTS/MEV)	S=	.50901E-01
.30000	TO	.40000	MEV	.89312E-02	(	8520	EVENTS/MEV)	S=	.71604E-02
.40000	TO	.50000	MEV	.12632E-02	(	8109	EVENTS/MEV)	S=	.10636E-02
.50000	TO	.60000	MEV	.20163E-03	(	8349	EVENTS/MEV)	S=	.16302E-03
.60000	TO	.80000	MEV	.17692E-04	(	8160	EVENTS/MEV)	S=	.10142E-04
.80000	TO	1.0000	MEV	.47368E-06	(	8710	EVENTS/MEV)	S=	.26056E-06
1.0000	TO	1.5000	MEV	.47981E-08	(	8256	EVENTS/MEV)	S=	.45779E-09
1.5000	TO	2.0000	MEV	.60261E-12	(	8426	EVENTS/MEV)	S=	.57117E-13

NEOUT=NO. THAT COULD NOT BE ENERGY-SORTED= 6

ENERGY DEPOSITION SUMMARY

The fractions of deposited energy are listed for each region according to the kind of particle.

IQ Integer change of particle. IQ = -1 for electron, 0 for photon, and 1 for positron.

IARG Integer argument that indicates the situation under which the output subroutine AUSGAB was called (see Table 4.6.1 of ref. 1).

SUMMARY OF EVENT COUNT

The count of events occurred in each region is listed according to the kind of particle. For significations of IQ and IARG see above.

ENERGY DEPOSITION SUMMARY FOR PARTICLES WITH IQ=-1

REGION	IARG 0	1	2	3	4	ROW SUM
1	.0	.0	.0	.0	.0	.0
2	.0	.0	.0	.0	.0	.0
3	.0	.0	.2552288D-08	.0	.0	.2552288D-08
4	.0	.0	.0	.0	.0	.0
5	.0	.0	.0	.0	.0	.0
6	.0	.0	.0	.0	.0	.0
7	.0	.0	.0	.0	.0	.0
8	.0	.0	.0	.0	.0	.0
9	.0	.0	.1455167D-04	.0	.0	.1455167D-04
10	.0	.0	.0	.0	.0	.0
11	.0	.0	.0	.0	.0	.0
12	.0	.0	.1484114D-06	.0	.0	.1484114D-06
13	.0	.0	.1240989D-05	.0	.0	.1240989D-05
14	.9358402D-14	.0	.4437897D-03	.0	.0	.4437897D-03
15	.5559742D-17	.0	.6659082D-03	.0	.0	.6659082D-03

SUMMARY OF EVENT COUNT FOR PARTICLES WITH IQ=-1

REGION	IARG 0	1	2	3	4	ROW SUM
1	0	0	0	0	0	0
2	0	0	0	0	0	0
3	0	0	60	0	0	60
4	0	0	0	0	0	0
5	0	0	0	0	0	0
6	0	0	0	0	0	0
7	0	0	0	0	0	0
8	0	0	0	0	0	0
9	0	0	29	0	0	29
10	0	0	0	0	0	0
11	0	0	0	0	0	0
12	0	0	13	0	0	13
13	0	0	7	0	0	7
14	6	0	258	0	0	266
15	3	0	167	0	0	170

FRACTION OF ENERGY CARRIED BY PARTICLE

The fraction of energy carried by particle into the region of interest (ROI) is listed for each energy bin. The summation of the individual fraction over the whole energy bins and particles is given as SWNP. The value of SWNP can be used as a criterion of the error estimation. When it is smaller than 1×10^{-5}, the error in the result is usually large.

On the right hand side of the output are listed for each energy bin the counts of events occurred in the region of interest.

KERMA*(A CM**2)/MEV/EL.

This value gives us Kerma multiplied by the cross-sectional area (A cm^2) on x-y coordinate plane of ROI and normalized to a unit incident energy (MEV) and one electron. The Kerma to RPL glass dosimeter is, therefore, obtained by dividing the above value by the crossectional area of ROI and multiplying it by the synchrotron radiation energy loss per turn and the number of electrons and/or positrons per m.

KERMA(I) shows the individual components to KERMA*(A CM**2)/MEV/EL. from each energy bin.

RANDOM NO. GENERATION SEED = 123456781

FRACTION OF ENERGY CARRIED BY PARTICLE INTO THE REGION OF INTEREST: IROI= 1

		ELECTRON	PHOTON	POSITRON	ELECTRON	PHOTON	POSITRON	KERMA(I)
0.010 TO	0.015 MEV	.0	.0	.0	0	0	0	.0
0.015 TO	0.020 MEV	.0	.0	.0	0	0	0	.0
0.020 TO	0.030 MEV	.0	.0	.0	0	0	0	.0
0.030 TO	0.040 MEV	.0	.0	.0	0	0	0	.0
0.040 TO	0.050 MEV	.0	.0	.0	0	0	0	.0
0.050 TO	0.060 MEV	.0	.0	.0	0	0	0	.0
0.060 TO	0.080 MEV	.0	.3808D-10	.0	0	2	0	.5712D-11
0.080 TO	0.100 MEV	.0	.2592D-13	.0	0	2	0	.2255D-14
0.100 TO	0.150 MEV	.0	.1364D-06	.0	0	9	0	.7093D-08
0.150 TO	0.200 MEV	.0	.1408D-04	.0	0	10	0	.5071D-06
0.200 TO	0.300 MEV	.0	.1455D-04	.0	0	44	0	.4656D-06
0.300 TO	0.400 MEV	.0	.1924D-04	.0	0	60	0	.5771D-06
0.400 TO	0.500 MEV	.0	.1236D-06	.0	0	48	0	.3708D-07
0.500 TO	0.600 MEV	.0	.4612D-08	.0	0	48	0	.1384D-07
0.600 TO	0.800 MEV	.0	.2335D-07	.0	0	52	0	.6771D-09
0.800 TO	1.000 MEV	.0	.2903D-09	.0	0	31	0	.8129D-11
1.000 TO	1.500 MEV	.0	.2319D-10	.0	0	42	0	.6029D-12
1.500 TO	2.000 MEV	.0	.2731D-16	.0	0	2	0	.6555D-18

TOTAL ENERGY FRACTION THAT THE PARTICLE BRINGS IN : SWNP= .4973D-04 KERMA*(A CM**2)/MEV/EL.= .1608E-05

FRACTION OF ENERGY CARRIED BY PARTICLE INTO THE REGION OF INTEREST: IROI= 6

		ELECTRON	PHOTON	POSITRON	ELECTRON	PHOTON	POSITRON	KERMA(I)
0.010 TO	0.015 MEV	.0	.0	.0	0	0	0	.0
0.015 TO	0.020 MEV	.0	.0	.0	0	0	0	.0
0.020 TO	0.030 MEV	.0	.0	.0	0	0	0	.0
0.030 TO	0.040 MEV	.0	.0	.0	0	0	0	.0
0.040 TO	0.050 MEV	.0	.0	.0	0	0	0	.0
0.050 TO	0.060 MEV	.0	.0	.0	0	0	0	.0
0.060 TO	0.080 MEV	.0	.1889D-16	.0	0	1	0	.2834D-17
0.080 TO	0.100 MEV	.0	.0	.0	0	0	0	.0
0.100 TO	0.150 MEV	.0	.1013D-10	.0	0	2	0	.5269D-12
0.150 TO	0.200 MEV	.0	.0	.0	0	0	0	.0
0.200 TO	0.300 MEV	.0	.7763D-09	.0	0	6	0	.2484D-10
0.300 TO	0.400 MEV	.0	.3629D-06	.0	0	11	0	.1089D-06
0.400 TO	0.600 MEV	.0	.6666D-07	.0	0	13	0	.2000D-08
0.600 TO	0.800 MEV	.0	.1517D-06	.0	0	16	0	.4551D-08
0.800 TO	1.000 MEV	.0	.5193D-08	.0	0	11	0	.1506D-09
1.000 TO	1.500 MEV	.0	.7413D-12	.0	0	6	0	.2076D-13
1.500 TO	2.000 MEV	.0	.5192D-12	.0	0	10	0	.1360D-13
		.0	.0	.0	0	0	0	.0

TOTAL ENERGY FRACTION THAT THE PARTICLE BRINGS IN : SWNP= .3854D-05 KERMA*(A CM**2)/MEV/EL.= .1156E-06

FRACTION OF ENERGY CARRIED BY PARTICLE INTO THE REGION OF INTEREST: IROI= 23

		ELECTRON	PHOTON	POSITRON	ELECTRON	PHOTON	POSITRON	KERMA(I)
0.010 TO	0.015 MEV	.0	.0	.0	0	0	0	.0
0.015 TO	0.020 MEV	.0	.0	.0	0	0	0	.0
0.020 TO	0.030 MEV	.0	.0	.0	0	0	0	.0

3.5 An example of absorbed dose calculation.

Let us calculate the absorbed dose, per integrated current of one Ah to the glass dosimeter placed in ROI=96 of Fig.2. assuming that the y x z dimension of the region is 1.6 x 2.0 cm^2 and the computer output [KERMA (A CM**2)/MEV/EL.] takes a value of 2.83 x $10^{-3}(cm^2\ e^{-1})$.

From eq. 1 the energy loss per revolution, ΔE, at 17 GeV operation is:

$$\Delta E = 88.46 \frac{[E(GeV)]^4}{R(m)} = \frac{88.5 \times 17^4}{192}$$

$$= 3.85 \times 10^4 \text{ (keV)}$$
$$= 38.5 \text{ (MeV)} \qquad (9)$$

Since 1 Ah = 3600 C, the average number of electrons and/or positrons, N_e, per one meter of beam orbit is:

$$N_e = \frac{1}{2\pi R} \frac{3600\ C}{1.6\times10^{-19}C} = \frac{2.25\times10^{22}}{2\times3.14\times192}$$

$$= 1.87\times10^{19} \text{ per m.} \qquad (10)$$

The cross-sectional area A(cm^2) on the x-y plane of the ROI which has a length of 1m in x-direction is:

$$A = 1.6 \times 100 = 160 \ (cm^2) \qquad (11)$$

Therefore, Kerma K for the ROI is:

$$K = \Delta E \cdot N_e \ |\text{KERMA*(A CM**2)/MEV/EL.}|/A$$
$$= 38.5 \cdot (1.87 \times 10^{19}) \cdot (2.83 \times 10^{-3})/160 \qquad (12)$$
$$= 1.27 \times 10^{16} \text{ (MeV).}$$

The absorbed dose D in rads to the RPL glass dosimeter is:

$$D = (1.60\times10^{-8}) \cdot (1.27\times10^{16})$$
$$= 2.03\times10^{8} \text{ (rad)}$$

References:

1. R.L. Ford and W.R.Nelson, The EGS code system: Computer program for the Monte Carlo simulation of electromagnetic cascade showers (Version 3), SLAC Report No. 210 (1978)

2. A.J. Cook and L.J. Shustek, A User's guide to MORTRAN 2, CTGM No. 165, SLAC (1975)

3. J. Schwinger, On the classical radiation of accelerated electrons, Phys. Rev. 75, 1912-1925 (1949)

4. H.H. Umstätter, Efficient computation of synchrotron radiation spectrum, CERN/PS/SM/81-13 (1981)

5. K.R. Kase and W.R. Nelson, Concept of radiation dosimetry, SLAC Rep. No. 153 (1972)

6. C. Yamaguchi, Comparison of absorbed dose to RPL glass dosimeter calculated by EGS code and kerma, DESY Internal Report, DESY D3/40 (1981)

7. E. Strom and H.I. Israel, photon cross sections from 1 keV to 100 MeV for elements Z = 1 to Z = 100, Atom. Data and Nucl. Data Tables 7, 565 (1970)

A P P E N D I X

Some other geometrical configurations for the dipole magnets of HERA, LEP and PETRA are given. PETRA6 and LEP2 are the user codes for calculating the radial dose distribution in the corresponding accelerator tunnels.

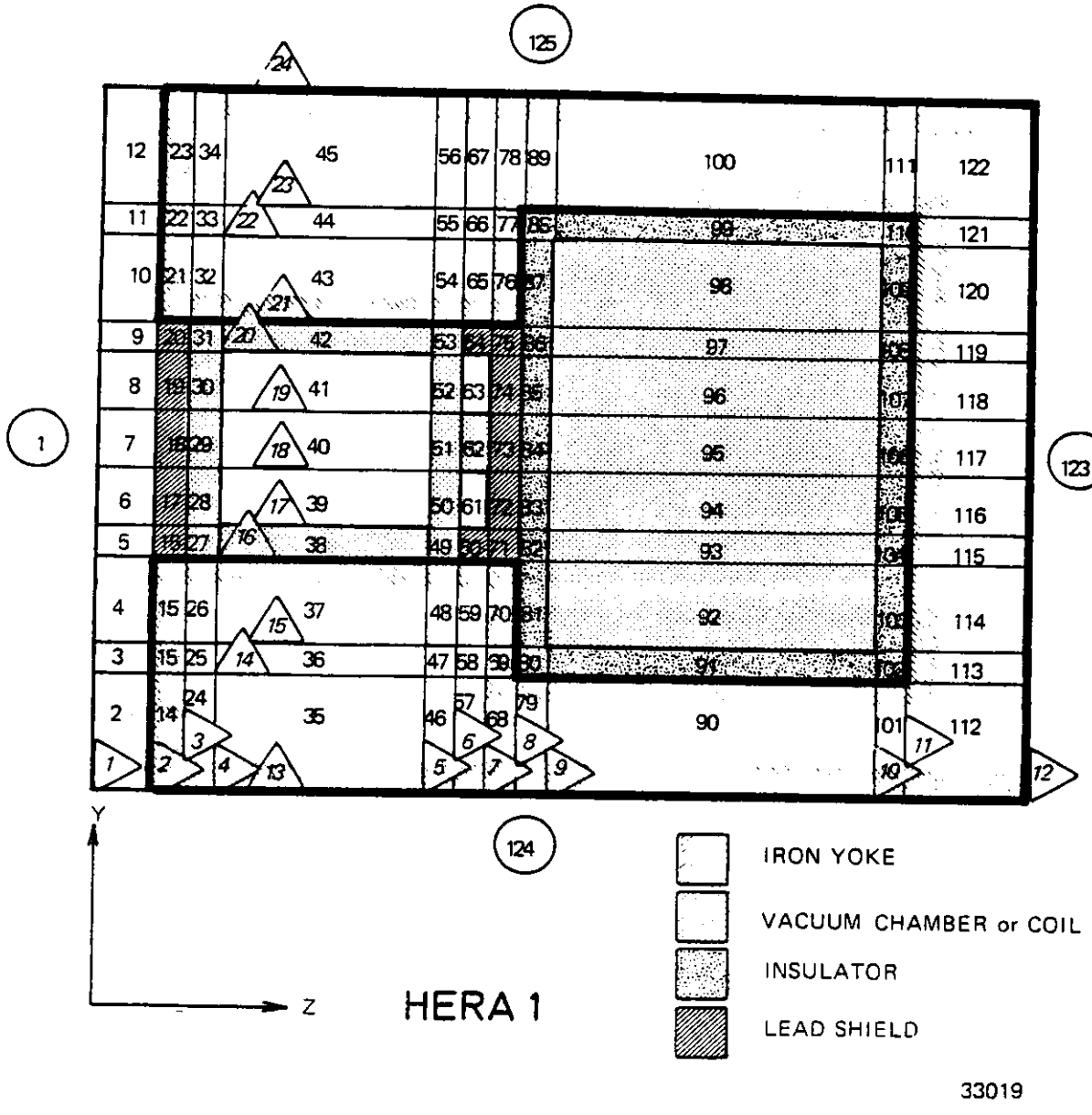

Fig. 4 Geometrical configuration for user code HERA1.

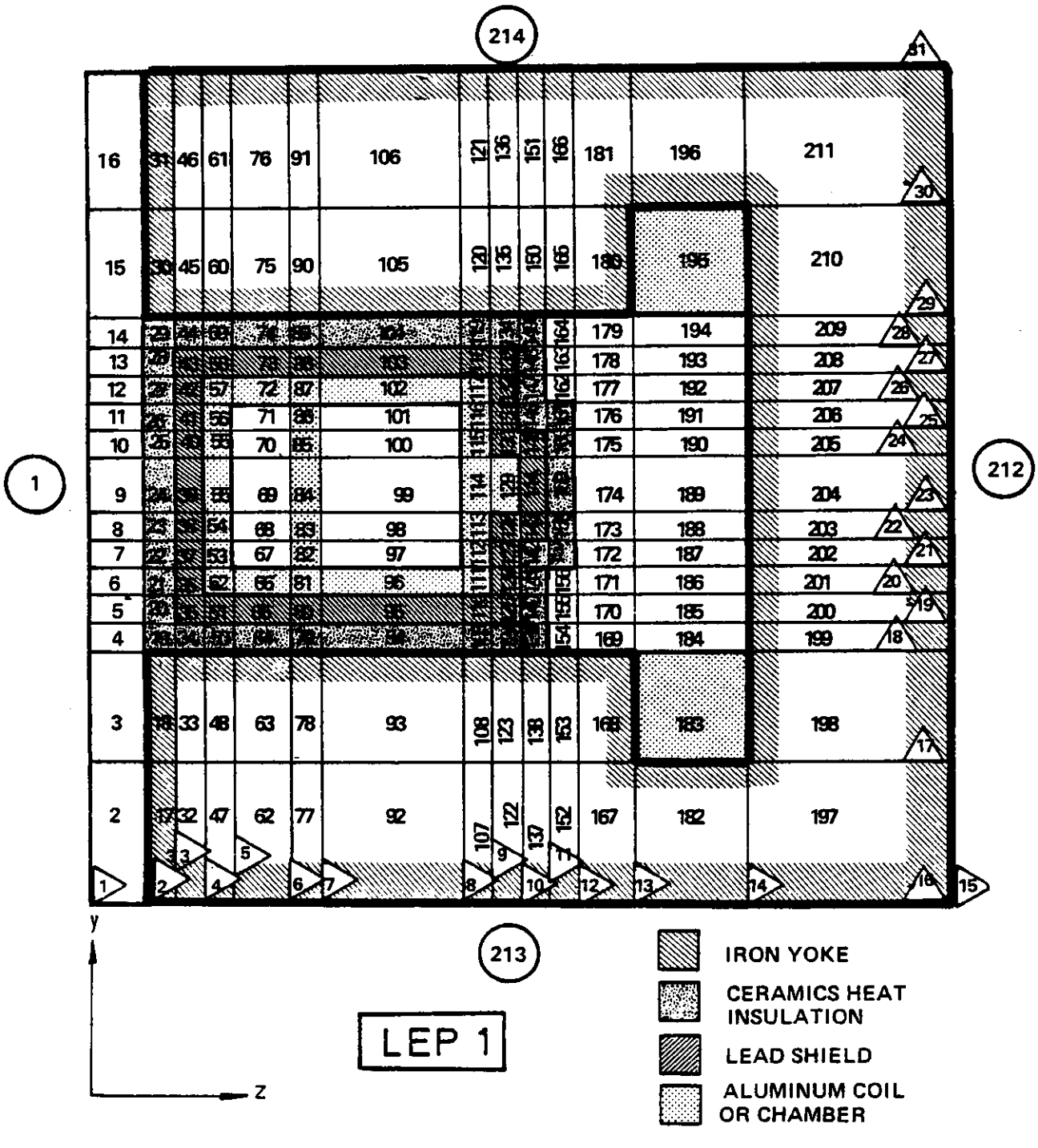

Fig. 5 Geometrical configuration for user code LEP1.

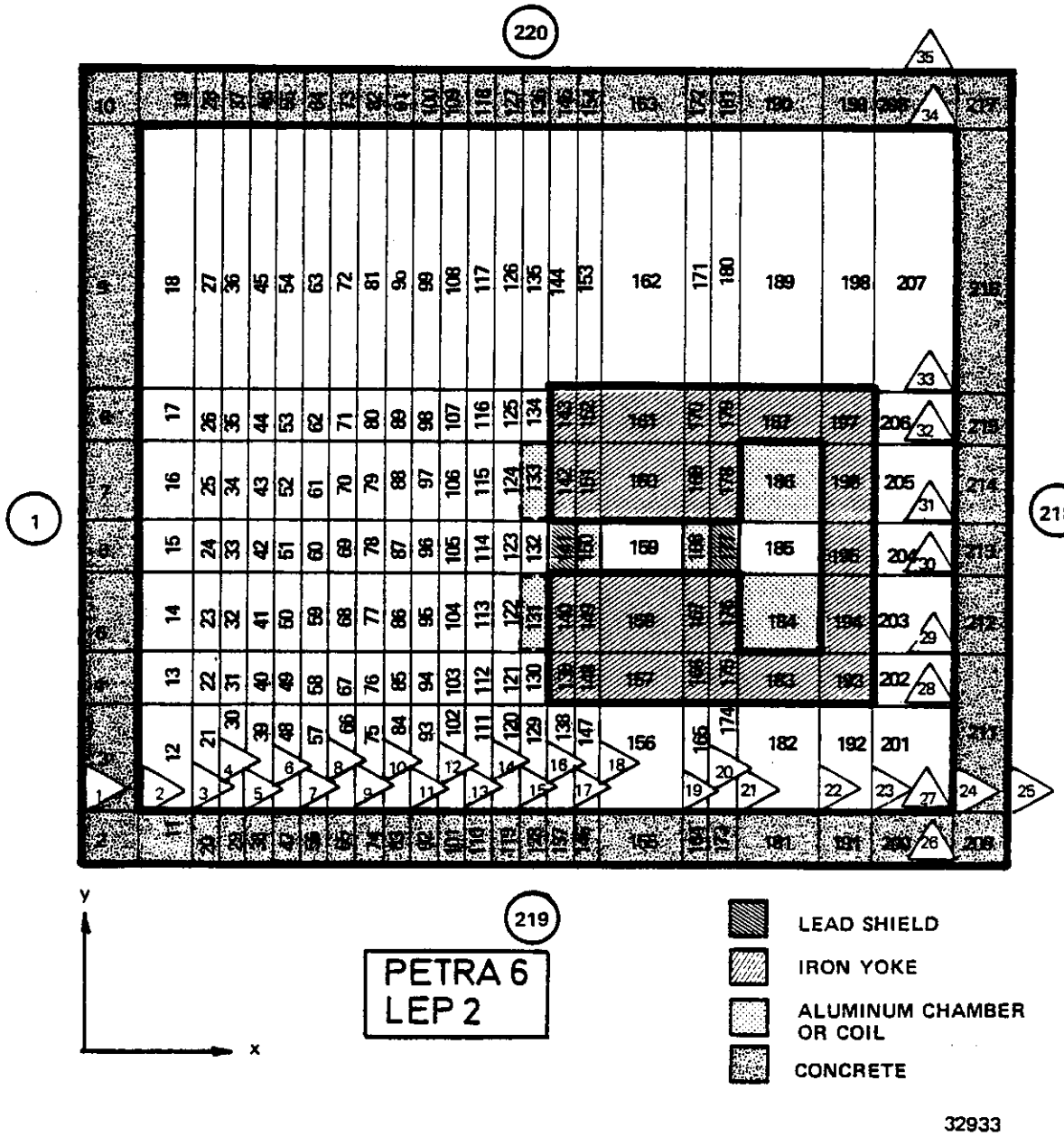

Fig. 6 Geometrical configuration for user code PETRA6 and LEP2.