EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN/MPS/CO 73-5
17 August, 1973

THE TRACKING OF PROTON TRAJECTORIES AROUND THE CPS

D.M. Lewis

Summary

During the course of a computational investigation into the behaviour of the proton beam after acceleration[1,2], the program BDP was written in CERN 6000 Fortran. This program is a more general version of that of O. Barbalat[3], which tracks proton trajectories in both horizontal and vertical planes for any given machine configuration containing dipoles, quadrupoles, sextupoles and octupoles; particle co-ordinates and lens strengths may be given in real units or in normalized units[4] for a theoretical approach.

* * *

/je

1. Introduction

The user reads the machine data in terms of locations of insertions, lens strengths and directions; the latter may be introduced in terms of the practical units of current for the standard PS elements or as normalized strengths; the program then determines all the necessary parameters. Initial proton co-ordinates are introduced in mm units (either real or normalized) and may be set to any specified straight section. The usual machine parameters must be supplied, viz. Q_R, Q_V, β_F, β_D, d/d_r (Q_R), the moment compaction factor α and the machine magnetic field, B. Closed orbit deviations may be introduced in any of those straight sections containing insertions and the lenses may be set off-centre if necessary. The program will track up to 20 proton trajectories for a maximum of 99 machine cycles. The proton co-ordinates are printed in mm units as required and a graphical print-out may be produced using the GD-3 system. Finally a post-mortem is available which allows a full trace of the computation to be recorded at each step.

2. Description of the program

All data is read in under the subroutine DATAIN, which also calculates all the necessary parameters for normalization, lens strengths, etc.

The program arranges the sequence of the insertions by means of the subroutine SEQUENS and computation commences with proton co-ordinates in straight section NINT. Traversal of the machine is carried out until straight section NTRACE is reached, when the data is printed and the proton co-ordinates will be printed each time the section NTRACE is reached. In the event of NINT or NTRACE coinciding with an insertion the co-ordinates are printed out before the lens traversal each time. The computation is performed in M.K.S. units but all output data is printed in mm units.

The CPS is divided into linear section of magnet units, each one separated by an insertion, and for each inter-insertional step around the machine, transfer matrices are calculated. These matrices are in normalized units and if ϕ_{ri} and ϕ_{zi} represent the calculated betatron phase jump for each of the NOPS steps, then NOPS matrices (M_i) are calculated with the form given by equation (1)

$$M_i = \begin{pmatrix} \cos\phi_{ri} & \sin\phi_{ri} & 0 & 0 \\ -\sin\phi_{ri} & \cos\phi_{ri} & 0 & 0 \\ 0 & 0 & \cos\phi_{zi} & \sin\phi_{zi} \\ 0 & 0 & -\sin\phi_{zi} & \cos\phi_{zi} \end{pmatrix} \qquad (1)$$

The use of normalized co-ordinates requires that normalization out at several straight sections; this is performed by the subroutine NORM which determines whether the sections are F or D focusing and then normalizes accordingly. Thus if real co-ordinates are represented by array CORD and the normalized co-ordinates by array CORDN, then normalization in an F section obeys the relation (2)

$$\begin{aligned} &\text{CORDN }(1,J) = \text{CORD }(1,J) && (x) \\ &\text{CORDN }(2,J) = \text{CORD }(2,J) \times \beta_F && (P_x) \\ &\text{CORDN }(3,J) = \text{CORD }(3,J) \times (\beta_F/\beta_D)^{\frac{1}{2}} && (z) \\ &\text{CORDN }(4,J) = \text{CORD }(4,J) \times (\beta_F \times \beta_D)^{\frac{1}{2}} && (P_z) \end{aligned} \qquad (2)$$

where J is subscript labelling of the proton.

$_F$ and $_D$ have their usual meaning. Similarly in a D section the relation (3) holds:

$$\begin{aligned} &\text{CORDN }(1,J) = \text{CORD }(1,J) \times (\beta_F/\beta_D)^{\frac{1}{2}} \\ &\text{CORDN }(2,J) = \text{CORD }(2,J) \times (\beta_F \times \beta_D)^{\frac{1}{2}} \\ &\text{CORDN }(3,J) = \text{CORD }(3,J) \times \beta_F \\ &\text{CORDN }(4,J) = \text{CORD }(4,J) \end{aligned} \qquad (3)$$

The traversal of a given number of magnet units is then performed by matrix multiplication in subroutine TRAVERS.

Each lens traversal obeys the thin lens approximation and transversal momentum jumps are then determined in terms of the real co-ordinates CORD. The lenses treated in the program are :

1. DIPOLES (subroutine DIPØL)

Momentum jumps obeying equation (4) are obtained

$$\Delta p = K_D \tag{4}$$

depending on the direction and polarity of the dipole. The convention adopted here is that positive currents in the dipole produce positive momentum jumps. The dipole characteristic strength is in Tesla.m Amp^{-1}.

2. QUADRUPOLES (subroutine QUATRA)

The momentum jumps obtained are:

$$\begin{aligned} \Delta P_x &= -K_Q X_Q \\ \Delta P_z &= +K_Q Z_Q \end{aligned} \tag{5}$$

where K_Q is the quadrupole strength previously calculated in subroutine DATAIN

$$\begin{aligned} X_Q &= X + R_D + R_{corr} - S_x \\ Z_Q &= Z + V_{corr} - S_z \end{aligned} \tag{τ}$$

X_Q, Z_Q are the effective real co-ordinates from the lens centre, having allowed for the momentum compaction of the beam (RD), closed orbit deflections (R_{corr}, V_{corr}) and off-centre positioning of the quadrupole (S_x, S_z). The sign convention adopted here for the quadrupole current is essentially that of Barbalat[3)], viz. positive quadrupole current gives rise to a focusing force in the radial plane. The characteristic strength of the normal quadrupoles in the ring are in units of Tesla.Amp^{-1}.

3. SEXTUPOLES (Subroutine SEXTRA)

Either mode of sextupole mounting may be used viz. normal or skew, and for the case of a skew sextupole, i.e. one with one pair of poles parallel to the horizontal plane[5)]

$$\Delta P_x = -(2X_s Z_s) \cdot K_s$$
$$\Delta P_z = +(Z_s^2 - X_s^2) \cdot K_s \qquad (7)$$

where X_s, Z_s have similar values to X_Q, Z_Q in (5) and again the lens strength K_s is determined in subroutine DATAIN. The convention here is that positive currents produce positive momenta jumps ΔP_z proportional to Z^2 or positive jumps in ΔP_x proportional to X^2 depending on which orientation is used. The characteristic strength of the sextupole in units of Tesla.(m.Amp)$^{-1}$ may be obtained in reference 6).

4. OCTUPOLES (subroutine OCTRA)

Momenta jumps are obtained

$$\Delta P_x = +(X_s^3 - 3\, X_s\, Z_s^2) \cdot k$$
$$\Delta P_z = +(Z_s^3 - 3\, Z_s\, X_s^2) \cdot k \qquad (8)$$

where X_s, Z_s have the usual values and again the sign convention is that positive octupole currents produce positive values for k. Typical values for the characteristic strength may be obtained from reference 6), the units being those of Tesla.Amp^{-1}.m^{-2}.

3. The introduction of data

Data is read under subroutine DATAIN and subsequently printed out at the end of the program. The necessary physical parameters are calculated and allocated to the relevent COMMON blocks. Input data is supplied in the form of the following cards:

Card 1: Format (I4,I1)

N, MPRINT

N is the total number of protons tracked (maximum 20) and MPRINT 1 allows for the option that both input co-ordinates and graph output co-ordinates are required as normalized units.

Card 2: Format (2 I4)

NINT, NTRACE

The straight section locations where input and output data respectively are required.

Card 3: Format (2F6.3)

BF, BD

The usual betatron amplitude is m.radian units.

Card 4: Format (2F6.3)

QR, QV

The horizontal and vertical Q values of the unperturbed machine.

Cards 5: to N+4 Format (4F10.5)

CORD (I,J) where I = 1,4 , J = 1,N

or depending if MPRINT equals 1

CORDN (I,J)

the proton co-ordinates X, P_x, Z, P_z in mm and mradian units.

Card N+5: Format (3I4)

NQUAD, NSEX, NDIP, NOCT

The total number of quadrupoles, sextupoles, dipoles and octupoles respectively required in the machine.

Cards: (the next NQUAD cards) Format (I3,F10.4,F10.6,2I1,F10.5,I1)

LSQ, AQUAD, QTS, KX, KDIR, SHIF, MOC

where

LSQ gives the straight section location of the quadrupole

AQUAD gives a value to either the quadrupole current in Amps or to its normalized strength

QTS the characteristic strength in units of $Tesla.Amp^{-1}$

KX =0 gives AQUAD the interpretation of quadrupole current
=1 gives AQUAD the interpretation of normalized strength

KDIR =1,2 determines whether the quadrupole normalized strength (only for the case KX=1) is in the horizontal or vertical direction

SHIF gives the magnitude in mms of the shift of the centre of the quadrupole from the centre of the vacuum chamber

MOC =1,2 determines whether the above shift is in the horizontal or vertical direction

Cards: (the next NSEX cards) (Format I3,F10.4,F10.6,3I1,F10.5,I1)

LSS, ASEX, STS, KS, KSIR, MOD, SHIS, MOT

LSS gives the straight section location for the sextupole

ASEC gives either the sextupole current (in Amps) or the normalized strength

STS the characteristic strength of the sextupole in units of $Tesla.Amp^{-1}.m^{-1}$

KS index for the interpretation of ASEX viz. KS=0 assigns the value of current (in Amps)

KSIR index for interpretation of the direction of the strength, i.e. horizontal or vertical for KSIR=1,2 respectively

MOD index to indicate whether one pair of poles lies on horizontal or vertical axes, MOD=1,2 respectively

SHIS gives in mm units the value of the shift in the sextupole centre away from the centre of the vacuum chamber

MOC index to indicate the direction of the above shift, MOC=1,2 for horizontal and vertical shifts respectively.

Cards: (the next NDIP cards) (Format (2I4,F10.5),I4)

LSD, DTS, KD, CND, JPN

LSD location of dipole

DTS characteristic stength of the dipole in $Tesla.m.Amp^{-1}$

KD index to indicate whether CND is the dipole current (KD=0) or the normalized strength (KD=1)

CND the dipole current in Amps or normalized strength

JPN index for the direction of the dipole viz. JPN=1,2 for horizontal and vertical directions respectively.

Cards: (the next NOCT cards) (Format I3,E10.3,F10.8,2I1,F10.5,I1)

LSR, AOCT, OTS, KO, KOIR, SHIO, MOO

LSO location of octupole

AOCT gives the value of either octupole current (in Amps) or the normalized strength

OTS the characteristic strength of the octupole in Tesla.Amp^{-1}.m^{-2}

KO index to decide whether AOCT refers to current or lens strength KO(10)=0,1 respectively

KOIR assigns the direction of the lens strength, KOIR=1,2 respectively for horizontal and vertical directions

SHIO the magnitude in mm of shift of the centre of the octupole

MOO index to assign the direction of the above shift.

Card: N + NQUAD + NSEX + NDIP + NOCT + 6 (Format F10.4)

RDISP

The position of the equilibrium orbit (in mms) from the centre of the vacuum chamber.

Card: N + SQUAD + NSEX + NDIP + NOCT + 7 (Format F10.4)

QRAD

The value of d/dr (QR) under the given machine conditions.

Card: N + NQUAD + NSEX + NDIP + NOCT + 8 (Format I4)

LF

The number of straight sections where closed orbit deviations are to be included.

Cards: (the next LF cards) (Format I4, 2F6.3)

LSTS, AMP, AMV

LSTS the straight section location for the closed orbit correction

AMP horizontal closed orbit deviation in (real) mm units

AMV vertical closed orbit deviation in (real) mm units.

Card: N + NQUAD + NSEX + NDIP + NOCT + LF + 9 (Format I4)

NREV

The number of machine revolutions to be tracked.

Card: N + SQUAD + NSEX + NDIP + NOCT + LF 10 (Format I1)

JPR

Index to determine whether a full trace is required viz. JPR=1.

Card: N + NQUAD + NSEX + NDIP + NOCT + LF + 11 (Format 2I1)

NXD, NZP

Indices to indicate whether graph outputs are required for horizontal (NXP=1) and vertical (NZP=1) planes.

4. Conclusion

The program described above produces a relatively convenient tool for the study of proton trajectories in the CPS after acceleration. A typical example of tracking 8 protons over 50 machine revolutions could be run as a CERN 6000 XJOB, the limit set only by the maximum line printer output tolerated; typical C.P. times are of the order of 10 seconds.

Acknowledgements

The author is extremely grateful to Ch. Steinbach for introducing him to this subject and for continued interest and support. The author would also like to thank O. Barbalat for several interesting discussions.

REFERENCES

1. Ch. STEINBACH, Possibility of vertical dumping on a fixed block in the CPS, MPS/CO Note 73-2

2. O. BARBALAT, Décharge du faisceau PS - essai de synthèse du séminaire PS du 14 mars 1973, MPS/DL/Note 73-3

3. O. BARBALAT, Fortran programmes for the CPS resonant extraction calculations, MPS/Int.DL/69-1

4. H.G. HEREWARD, Effect of quadrupoles in the CPS, resonant extraction calculations, MPS/Int.DL/63-9

5. N. MARSHALL-KING, Particle dynamics in sextupole magnets, MPS/EP/26

6. E. REGENSTREIF, The CERN Proton Synchrotron, CERN 59-29

Distribution

PS Senior Staff
PSS and EiC's

```
      PROGRAM EDP (INPUT,OUTPUT,TAPE1=INPUT,TAPE2=OUTPUT,TAPE6)
      DIMENSION PS(4,4,10),CORD(4,20),CORDN(4,20),SHIF(10),MOCO(10),LSO(
     110),SHIO(10),MOO(10),LSQ(10),LSS(10),LSD(10),JPN(10),SHIS(10),MOT(
     210),MOD(10),XT(50,20),YT(50,20),LOC(10),NAT(10),NSTEP(10),RCO(20),
     3VCO(20),XPT(50,20),YPT(50,20)
      INTEGER PRI(10)
      REAL K5(10),K6(10),K9(10),K7(10)
      COMMON/H/NREV,K5,K6,K7,K9,JPN,SHIF,MOCO,SHIS,MOT,MOD,SHIO,MOO
      COMMON/P/JPR
      COMMON/V/RCO,VCO
      COMMON/X/NXP,NZP

C     READ IN DATA

      CALL DATAIN(CORD,CORDN,N)

C     SEQUENCE ARRANGEMENT OF ELEMENTS

      CALL SEQUENS (LOC,NAT,PRI,NSTEP,NOPS)
      CALL METRIC(CORD,N,4,1)
      CALL METRIC(CORDN,N,4,1)
      PRINT 1000

C     CALCULATION OF MAGNET MATRIX ELEMENTS

      DO 1 IOPS=1,NOPS
    1 CALL TRANSF (PS,NSTEP(IOPS),IOPS)
      PRINT 1000
      MREV=NREV+1
      DO 2 IREV=1,MREV
      DO 3 IOPS=1,NOPS
      RCORR=RCO(IOPS)
      VCORR=VCO(IOPS)
      NPRI=PRI(IOPS)
      IF (JPR.EQ.0) GO TO 9
      CALL METRIC (CORD ,N,4,2)
      CALL METRIC (CORDN,N,4,2)
      PRINT 1001,(((CORD(I,J),I=1,4),(CORDN(I,J),I=1,4)),J=1,N)
      CALL METRIC (CORD ,N,4,1)
      CALL METRIC (CORDN,N,4,1)

C     TRACE REQUIRED

    9 NOL=NAT(IOPS)
      GO TO (3,7,5,6,4,13),NOL
    7 CALL METRIC(CORD,N,4,2)
      CALL METRIC(CORDN,N,4,2)
      CALL DATAOUT (CORD,CORDN,N,IREV,MREV,XT,XPT,YT,YPT)
      CALL METRIC(CORD,N,4,1)
      CALL METRIC(CORDN,N,4,1)
      IF (IREV.EQ.MREV) GO TO 8
      GO TO 10

C     TRAVERSAL OF QUADRUOPLE

    5 CALL QUATRA (CORD,N,K5(NPRI),RCORR,VCORR,SHIF(NPRI),MOCO(NPRI))
      GO TO 10

C      TRAVERSAL OF SEXTUOPLE

    6 CALL SEXTRA (CORD,N,K6(NPRI),MOD(NPRI),RCORR,VCORR,SHIS(NPRI),MOT(
     1NPRI))
      GO TO 10

C.    TRAVERSAL OF DIPOLE

    4 JNOPS=JPN(NPRI)
      CALL DIPOL (CORD,N,LOC(IOPS),JNOPS,K7(NPRI))
      GO TO 10

C      TRAVERSAL OF OCTUPOLE

   13 CALL OCTRA (CORD,N,K9(NPRI),RCORR,VCORR,SHIO(NPRI),MOO(NPRI))
   10 CONTINUE

C     POST MORTEM OPTION

      IF (JPR.EQ.0) GO TO 11
      PRINT 1002,NAT(IOPS)
      CALL NORM(CORD,CORDN,N,2,LOC(IOPS))
      CALL METRIC (CORD ,N,4,2)
      CALL METRIC (CORDN,N,4,2)
      PRINT 1003,(((CORD(I,J),I=1,4),(CORDN(I,J),I=1,4)),J=1,N)
      CALL METRIC (CORD ,N,4,1)
```

```
      CALL METRIC (CORDN,N,4,1)
C     TRAVERSAL OF NSTEP(IOPS) MAGNET UNITS
   11 IF (NSTEP(IOPS).EQ.0) GO TO 3
      CALL TRAVERS (CORD,CORDN,PS,N,IOPS,LOC(IOPS),NSTEP(IOPS))
    3 CONTINUE
    8 CONTINUE
    2 CONTINUE
      IF (NXP.EQ.0.AND.NZP.EQ.0) GO TO 12
      CALL DRAW (XT,XPT,YT,YPT,N,MREV)
   12 PRINT 1000
 1000 FORMAT (1H1)
 1001 FORMAT (/,5X,'BDUMP1',/,8(3X,E12.4),/)
 1002 FORMAT (I8)
 1003 FORMAT (/,5X,'BDUMP2',/,8(3X,E12.4),/)
      STOP
      END
```

```
      SUBROUTINE DATAIN(CORD,CORDN,N)
      DIMENSION LSQ(10),LSS(10),AQUAD(10),ASEX(10),LSD(10),DTS(10),CND(1
     10),JPN(10),KD(10),LSO(10),SHIO(10),MOO(10),AOCT(10),OTS(10),KO(10)
     2,KOIR(10),QTS(10),STS(10),MOD(10),CORD(4,20),CORDN(4,20),LOC(10),N
     3TEP(10),LSTS(20),AMR(20),AMV(20),SHIS(10),MOT(10),KX(10),KDIR(10),
     4KS(10),KSIR(10),BMOD(2),RDI(2),SHIF(10),MOCO(10)
      REAL K1(10),K2(10),K3(10),K4(10),K5(10),K6(10),K7(10),K9(10),K10(1
     10),K11(10)
      INTEGER PRI(10)
      COMMON/A/NINT,NTRACE
      COMMON/B/NQUAD,LSQ,NSEX,LSS,NDIP,LSD,LSO,NOCT
      COMMON/C/BF,BD
      COMMON/D/QR,QV
      COMMON/E/C1,C2
      COMMON/F/RDISP
      COMMON/H/NREV,K5,K6,K7,K9,JPN,SHIF,MOCO,SHIS,MOT,MOD,SHIO,MOO
      COMMON/G/LF,LSTS,AMR,AMV
      COMMON/R/JPR
      COMMON/U/MPRINT
      COMMON/X/NXP,NZP

C     READS IN DATA AND IMMEDIATELLY PRINTS OUT ALL DATA AND NECESSARY
C     PARAMETERS

      DATA BMOD(1),BMOD(2)/'HORIZONTAL','  VERTICAL '/
      READ 2057,N,MPRINT
      READ 2001,NINT,NTRACE
      READ 2002,BF,BD
      READ 2002,QR,QV
      READ 2006,ALPHA
      IF (MPRINT.EQ.1) GO TO 2058
      READ 2004,((CORD(I,J),I=1,4),J=1,N)
      DO 2026 J=1,N
      DO 2026 K=1,4
 2026 CORDN(K,J)=0.0
      GO TO 2059
 2058 READ 2004,((CORDN(I,J),I=1,4),J=1,N)
      DO 2060 J=1,N
      DO 2060 K=1,4
 2060 CORD(K,J)=0.0
 2059 READ 2043,NQUAD,NSEX,NDIP,NOCT
      IF (NQUAD.EQ.0) GO TO 2029
      READ 2005,((LSQ(IQ),AQUAD(IQ),QTS(IQ),KX(IQ),KDIR(IQ),SHIF(IQ),MOC
     1O(IQ)),IQ=1,NQUAD)
 2029 IF (NSEX.EQ.0) GO TO 2030
      READ 2036,((LSS(IS),ASEX(IS),STS(IS),KS(IS),KSIR(IS),MOD(IS),SHIS(
     1IS),MOT(IS)),IS=1,NSEX)
 2030 IF (NDIP.EQ.0) GO TO 2044
      READ 2010,((LSD(ID),DTS(ID),KD(ID),CND(ID),JPN(ID)),ID=1,NDIP)
 2044 IF (NOCT.EQ.0) GO TO 2089
      READ 2076,((LSO(IO),AOCT(IO),OTS(IO),KO(IO),KOIR(IO),SHIO(IO),MOO(
     1IO)),IO=1,NOCT)
 2089 READ 2006,B
      READ 2006,RDISP
      READ 2006,QRAD
      READ 2000,LF
      IF (LF.EQ.0) GO TO 2031
      READ 2003,((LSTS(IF),AMR(IF),AMV(IF)),IF=1,LF)
 2031 READ 2000,NREV
      READ 2028,JPR
      READ 2074,NXP,NZP

C     CALCULATION OF MACHINE PARAMETERS

      C=0.299793
      RN=70.079
      P=B*RN*C*0.0001
      FP=ALPHA*RDISP/RN
      P=P*(1.0+FP)
      RN=RN+RDISP
      BRHO=RN*B*0.0001
      C1=SQRT(BF/BD)
      C2=SQRT(BF*BD)
      PRINT 2007
      PRINT 2014,B
      PRINT 2015,P
       PRINT 2008,NREV
      PRINT 2009,QR,QV,BF,BD
      PRINT 2075,ALPHA

C     CALCULATION OF QUADRUPLE CONSTANTS

      IF (NQUAD.EQ.0) GO TO 2032
      DO 2055 IQ=1,NQUAD
```

```
      MCC=MOCO(IQ)
      LQS=LSQ(IQ)/2
      LREM=LSQ(IQ)-(2*LQS)
      IF (LREM.EQ.0) GO TO 2039
      BQ=BF
     BQ1=BD
      GO TO 2040
 2039 BQ=BD
      BQ1=BF
 2040 CONTINUE
      IF (KX(IQ).EQ.0) GO TO 2067
      IF (KDIP(IQ).EQ.2) GO TO 2068
      K1(IQ)=AQUAD(IQ)
      GO TO 2064
 2068 K3(IQ)=AQUAD(IQ)
      K1(IQ)=(-1.0*K3(IQ)*BQ)/BQ1
      GO TO 2065
 2067 K1(IQ)=(-1.0*QTS(IQ)*AQUAD(IQ)*BQ)/BRHO
 2064 K3(IQ)=(-1.0*K1(IQ)*BQ1)/BQ
 2065 K5(IQ)=-1.0*K1(IQ)/BQ
      IF (KX(IQ).EQ.0) GO TO 2066
      AQUAD(IQ)=(-1.0*K1(IQ)*BRHO)/(QTS(IQ)*BQ)
 2066 PRINT 2011,IQ,LSQ(IQ),IQ,AQUAD(IQ)
      PRINT 2024,IQ,QTS(IQ)
      PRINT 2018,IQ,K1(IQ)
      PRINT 2041,K3(IQ)
      PRINT 2062,IQ,BMOD(MCC),SHIF(IQ)
      SHIF(IQ)=SHIF(IQ)/1000.0
 2055 CONTINUE

C     CALCULATION OF SEXTUPLE CONSTANTS

 2032 IF (NSEX.EQ.0) GO TO 2033
      DO 2012 IS=1,NSEX
      MS=MOD(IS)
      MT=MOT(IS)
      LSSS=LSS(IS)/2
      LSEM=LSS(IS)-(2*LSSS)
      IF (LSEM.EQ.0) GO TO 2037
      BS=BF
       BS1=SQRT((BD**3)/BF)
 2037 BS=SQRT((BD**3)/BF)
      BS1=BF
 2038 CONTINUE
      IF (KS(IS).EQ.0) GO TO 2070
      IF (KSIR(IS).EQ.2) GO TO 2071
      K2(IS)=ASEX(IS)
      K4(IS)=K2(IS)*BS1*(-1.0)/BS
      GO TO 2072
 2071 K4(IS)=ASEX(IS)
      K2(IS)=K4(IS)*BS*(-1.0)/BS1
 2072 K6(IS)=K2(IS)/BS*(-1.0**MS)
      ASEX(IS)=K6(IS)*BRHO*1000.0/STS(IS)
      GO TO 2073
 2070 K6(IS)=STS(IS)*ASEX(IS)/(BRHO*1000.0)
      K2(IS)=K6(IS)*BS*(-1.0**MS)
      K4(IS)=K6(IS)*BS1*(-1.0**(MS+1))
 2073 PRINT 2013,IS,LSS(IS),IS,ASEX(IS)
      PRINT 2027,IS,STS(IS)
      PRINT 2035,IS,BMOD(MS)
      PRINT 2019,IS,K2(IS)
      PRINT 2042,K4(IS)
      PRINT 2063,IS,BMOD(MT),SHIS(IS)
      SHIS(IS)=SHIS(IS)/1000.0
 2012 CONTINUE

C     CALCULATION OF DIPOLE CONSTANTS

 2033 IF (NDIP.EQ.0) GO TO 2049
      DO 2045 ID=1,NDIP
      LDD=LSD(ID)/2
      LDEM=LSD(ID)-(2*LDD)
      IF (LDEM.EQ.0.AND.JPN(ID).EQ.2) GO TO 2046
      IF (LDEM.EQ.1.AND.JPN(ID).EQ.1) GO TO 2046
      BDD=C2
      GO TO 2047
 2046 BDD=BF
 2047 PRINT 2048,ID,LSD(ID)
      KMOD=JPN(ID)
      PRINT 2056,ID,BMOD(KMOD)
      PRINT 2050,ID,DTS(ID)
      IF (KD(ID).EQ.1) GO TO 2051
      K7(ID)=CND(ID)*DTS(ID)/BRHO
      STRE=K7(ID)*BDD*1000.0
```

```
      PRINT 2053,ID,STRE
      PRINT 2052,ID,CND(ID)
      PRINT 2054
      GO TO 2045
 2051 PRINT 2053,ID,CND(ID)
      K7(ID)=CND(ID)/(BDD*1000.0)
      CND(ID)=K7(ID)*BRHO/DTS(ID)
      PRINT 2052,ID,CND(ID)
      PRINT 2054
 2045 CONTINUE
C        CALCULATION OF OCTUPOLE CONSTANTS
 2049 IF (NOCT.EQ.0) GO TO 2077
      DO 2078 IO=1,NOCT
      KR=KOIR(IO)
      KM=MOO(IO)
      LO=LSO(IO)/2
      LOEM=LSO(IO)-(2*LO)
      IF (LOEM.EQ.0) GO TO 2079
      BO=BF
      BO1=(BD**2)/BF
      GO TO 2080
 2079 BO=(BD**2)/BF
      BO1=BF
 2080 IF (KO(IO).EQ.0) GO TO 2081
      IF (KR.EQ.2) GO TO 2082
      K10(IO)=AOCT(IO)
      K11(IO)=K10(IO)*BO1/BO
      GO TO 2083
 2082 K11(IO)=AOCT(IO)
      K10(IO)=K11(IO)*BO/BO1
 2083 K9(IO)=(K10(IO)*1000000.0)/BO
      AOCT(IO)=K9(IO)*BRHO/OTS(IO)
      GO TO 2084
 2081 K9(IO)=OTS(IO)*AOCT(IO)/BRHO
      K10(IO)=K9(IO)*BO*0.000001
      K11(IO)=K9(IO)*BO1*0.000001
 2084 PRINT 2085,IO,LSO(IO),IO,AOCT(IO)
      PRINT 2086,IO,OTS(IO)
      PRINT 2087, K10(IO),K11(IO)
      PRINT 2088,IO,EMOD(KM),SHIO(IO)
      SHIO(IO)=SHIO(IO)/1000.0
 2078 CONTINUE
 2077 PRINT 2017,RDISP
      PRINT 2025,QRAD
      PRINT 2061
      RDI(1)=RDISP
      CALL METRIC(RDI,1,1,1)
      RDISP=RDI(1)
      IF (LF.EQ.0) GO TO 2034
      PRINT 2021
      PRINT 2022,((LSTS(IST),AMR(IST),AMV(IST)),IST=1,LF)
 2034 PRINT 2016,N,NINT
      MPRONT=2-MPRINT
      CALL NORM (CORD,CORDN,N,MPRONT,NINT)
      CALL NORM(CORD,CORDN,N,2,NINT)
      DO 2023 J=1,N
 2023 PRINT 2020, (CORD(I,J),I=1,4),(CORDN(I,J),I=1,4)
      QR=QR+(RDISP*QRAD)
 2000 FORMAT (I4)
 2001 FORMAT (2I4)
 2002 FORMAT (2F6.3)
 2003 FORMAT (I4,2F6.3)
 2004 FORMAT (4F10.5)
 2005 FORMAT (I3,F10.4,F10.6,2I1,F10.5,I1)
 2006 FORMAT (F10.4)
 2007 FORMAT (1H1,/5X,'INPUT DATA',/,5X,'**********',//)
 2008 FORMAT (5X,'NUMBER OF REVOLUTIONS',19X,'=',I6)
 2009 FORMAT (5X,'MACHINE VALUE OF QR',21X,'=',F11.4,/,5X,'MACHINE VALUE
     1 OF QV',21X,'=',F11.4,/,5X,'VALUE OF BF',29X,'=',F11.4,/,5X,'VALUE
     2 OF BD',29X,'=',F11.4)
 2010 FORMAT (2(I4,F10.5),I4)
 2011 FORMAT(5X,'SECTION NO OF QUADRUOPLE NO',I2,11X,'=',I6,/,5X,'CURREN
     1T IN QUADRUOPLE NO',I2,14X,'=',F11.4,' AMPS')
 2013 FORMAT (5X,'SECTION NO OF SEXTUOPLE NO',I2,12X,'=',I6,/,5X,'CURREN
     2T IN SEXTUOPLE NO ',I2,14X,'=',F11.4,' AMPS')
 2014 FORMAT (5X,'MAGNETIC FIELD',26X,'=',F11.4,' GAUSS')
 2015 FORMAT(5X,'MOMENTUM',32X,'=',F11.4,' GEV/C',//)
 2016 FORMAT (5X,'INITIAL CO-ORDINATES OF ',I3,' PARTICLES AT STRAIGHT S
     1ECTION NO',I3,/,5X,'***************************************
     2****************          ',/,5X,'X(MM)',9X,'XP(MRAD)',8X,'Z(MM)',
     39X,'ZP(MRAD)', 8X,'X(MM)',9X,'XP(MMEQ)',8X,'Z(MM)',9X,'ZP(MMEQ)',/
     4)
```

```
2017 FORMAT (5X,'DISPLACEMENT RADIAL POSITION',12X,'=',F11.4)
2018 FORMAT (5X,'FORCE OF QUADRUOPLE NO',I2,'IN RADIAL PLANE =',F13.6)
2019 FORMAT (5X,'FORCE OF SEXTUOPLE NO',I3,'IN RADIAL PLANE ','=',F13.6
    1)
2020 FORMAT (2X,8(F11.6,4X))
2021 FORMAT (5X,'CLOSED ORBIT',/,5X,'STRAIGHT SECTION',5X,'RADIAL DISPL
    1ACEMENT(MM)',5X,'VERTICAL DISPLACEMENT(MM)')
2022 FORMAT (12X,I3,17X,F7.3,22X,F7.3,//)
2024 FORMAT (5X,'CONSTANT OF QUADRUOPLE NO',I2,13X,'=',F13.6,'TESLA PER
    1 AMP')
2025 FORMAT (5X,'COEFFICIENT OF QR VARIATION',13X,'=',F11.4,' PER MM',/
    1)
2027 FORMAT (5X,'CONSTANT OF SEXTUOPLE NO',I2,14X,'=',F13.6,' TESLA PER
    2 AMP METRE')
2028 FORMAT (I1)
2035 FORMAT (5X,'SEXTUOPLE NO',I2,' AXIS OF POLES    ',9X,'=',1X,A10)
2036 FORMAT (I3,F10.4,F10.6,3I1,F10.5,I1)
2041 FORMAT (5X,'FORCE OF QUADRUOPLE IN VERTICAL PLANE',3X,'=',F11.4)
2042 FORMAT (5X,'FORCE OF SEXTUOPLE IN VERTICAL PLANE',4X,'=',F14.7)
2043 FORMAT (4I4)
2048 FORMAT (/,5X,'SECTION NO OF DIPOLE NO',I2,15X,'=',I6)
2050 FORMAT (5X,'CONSTANT OF DIPOLE NO',I2,17X,'=',F13.6,' TESLA METRE
    1 PER AMP')
2052 FORMAT (5X,'CURRENT IN DIPOLE NO',I2,18X,'=',F11.4,' AMPS')
2053 FORMAT (5X,'STRENGTH OF DIPOLE NO',I2,17X,'=',F13.6)
2054 FORMAT (/)
2056 FORMAT (5X,'AXIS OF DIPOLE NO',I2,21X,'=',A10)
2057 FORMAT (I4,I1)
2061 FORMAT (///)
2062 FORMAT (5X,'DIRECTION OF SHIFT QUADRUOPLE NO',I2,6X,'=',A10,/,5X,'
    1MAGNITUDE OF SHIFT',22X,'=',F11.4,' MM',/)
2063 FORMAT (5X,'DIRECTION OF SHIFT SEXTUOPLE NO',I2,7X,'=',A10,/,5X,'M
    1AGNITUDE OF SHIFT',22X,'=',F11.4,' MM',/)
2074 FORMAT (2I1)
2075 FORMAT (5X,'MOMENTUM COMPACTION FACTOR',14X,'=',F11.4,/)
2076 FORMAT (I3,E10.3,F10.5,2I1,F10.5,I1)
2085 FORMAT (5X,'SECTION NO OF OCTOPOLE NO',I2,13X,'=',I6,/,5X,'CURRENT
    1 IN OCTOPOLE NO',I2,16X,'=',F11.4,' AMPS')
2086 FORMAT (5X,'CONSTANT OF OCTOPOLE NO',I2,15X,'=',F15.8,' TESLA PER
    1 AMP.METRE2')
2087 FORMAT (5X,'FORCE OF OCTOPOLE IN HORIZONTAL PLANE',3X,'=',F16.9,/,
    15X,'FORCE OF OCTOPOLE IN VERTICAL PLANE',5X,'=',F16.9)
2088 FORMAT (5X,'DIRECTION OF SHIFT OCTOPOLE NO ',I2,8X,'= ',A10,/,5X,'
    1MAGNITUDE OF SHIFT',22X,'=',F11.4,' MM',/)
     RETURN
     END
```

```
      SUBROUTINE SEQUENS(LOC,NAT,PRI,NSTEP,NOPS)
      DIMENSION LSQ(10),LSS(10),LOC(10),NAT(10),NSTEP(10),LSTS(20),AMR(2
     10),AMV(20),RCO(20),VCO(20),NCQ(10),NCS(10),LSD(10),NCD(10),NCO(10)
     2,LSO(10)
      INTEGER PRI(10)
      COMMON/A/NINT,NTRACE
      COMMON/B/NQUAD,LSQ,NSEX,LSS,NDIP,LSD,LSO,NOCT
      COMMON/G/LF,LSTS,AMR,AMV
      COMMON/V/RCO,VCO

C     CONTROLS THE SEQUENCE OF OPERATIONS , VIZ MAGNET TRAVERSAL
C     QUADRUOPLE AND SEXTUOPLE , PRINTS OUT THE SEQUENCE

      NTRACE=NTRACE-NINT
      IF (NTRACE.GE.0) GO TO 3001
      NC1=1
      NTRACE=NTRACE+100
 3001 IF (NQUAD.EQ.0) GO TO 3024
      DO 3002 IQ=1,NQUAD
      LSQ(IQ)=LSQ(IQ)-NINT
      IF (LSQ(IQ).GE.0) GO TO 3002
      NCQ(IQ)=1
      LSQ(IQ)=LSQ(IQ)+100
 3002 CONTINUE
 3024 IF (NSEX.EQ.0) GO TO 3025
      DO 3003 IS=1,NSEX
      LSS(IS)=LSS(IS)-NINT
      IF (LSS(IS).GE.0) GO TO 3003
      NCS(IS)=1
      LSS(IS)=LSS(IS)+100
 3003 CONTINUE
 3025 IF (NDIP.EQ.0) GO TO 3026
       DO 3023 ID=1,NDIP
      LSD(ID)=LSD(ID)-NINT
      IF (LSD(ID).GE.0) GO TO 3023
      NCD(ID)=1
      LSD(ID)=LSD(ID)+100
 3023 CONTINUE
 3026 IF (NOCT.EQ.0) GO TO 3034
      DO 3035 IO=1,NOCT
      LSO(IO)=LSO(IO)-NINT
      IF (LSO(IO).GE.0) GO TO 3035
      NCO(IO)=1
      LSO(IO)=LSO(IO)+100
 3035 CONTINUE
 3034 NOPS=2+NSEX+NQUAD+NDIP+NOCT
      IOPS=1
      LOC(IOPS)=0
      NAT(IOPS)=1
      PRI(IOPS)=1
      DO 3032 MK=1,100
      LK=MK-1
      IF (LK.NE.NTRACE) GO TO 3005
      IOPS=IOPS+1
      LOC(IOPS)=NTRACE
      NAT(IOPS)=2
      PRI(IOPS)=1
 3005 DO 3006 IQ=1,NQUAD
      IF (LK.NE.LSQ(IQ)) GO TO 3006
      IOPS=IOPS+1
      LOC(IOPS)=LSQ(IQ)
      NAT(IOPS)=3
      PRI(IOPS)=IQ
 3006 CONTINUE
      DO 3004 IS=1,NSEX
      IF (LK.NE.LSS(IS)) GO TO 3004
      IOPS=IOPS+1
      LOC(IOPS)=LSS(IS)
      NAT(IOPS)=4
      PRI(IOPS)=IS
 3004 CONTINUE
      DO 3028 ID=1,NDIP
      IF (LK.NE.LSD(ID)) GO TO 3028
      IOPS=IOPS+1
      LOC(IOPS)=LSD(ID)
      NAT(IOPS)=5
      PRI(IOPS)=ID
 3028 CONTINUE
      DO 3040 IO=1,NOCT
      IF (LK.NE.LSO(IO)) GO TO 3040
      IOPS=IOPS+1
      LOC(IOPS)=LSO(IO)
      NAT(IOPS)=6
      PRI(IOPS)=IO
```

```
3040 CONTINUE
3032 CONTINUE
     MOPS=NOPS+1
     LOC(MOPS)=100
     DO 3007 IOPS=1,NOPS
     NSTEP(IOPS)=LOC(IOPS+1)-LOC(IOPS)
3007 CONTINUE
     NTRACE=NTRACE+NINT
     IF (NC1) GO TO 3008
     NTRACE=NTRACE-100
3008 DO 3009 IQ=1,NQUAD
     LSQ(IQ)=LSQ(IQ)+NINT
     IF (NCQ(IQ)) GO TO 3009
     LSQ(IQ)=LSQ(IQ)-100
3009 CONTINUE
     DO 3010 IS=1,NSEX
     LSS(IS)=LSS(IS)+NINT
     IF (NCS(IS)) GO TO 3010
     LSS(IS)=LSS(IS)-100
3010 CONTINUE
     DO 3029 ID=1,NDIP
     LSD(ID)=LSD(ID)+NINT
     IF (NCD(ID)) GO TO 3029
     LSD(ID)=LSD(ID)-100
3029 CONTINUE
     DO 3038 IO=1,NOCT
     LSO(IO)=LSO(IO)+NINT
     IF (NCO(IO)) GO TO 3038
     LSO(IO)=LSO(IO)-100
3038 CONTINUE
     PRINT 3011
     DO 3012 IOPS=1,NOPS
     IIOPS=NAT(IOPS)
     GO TO (3013,3014,3015,3016,3030,3041),IIOPS
3013 LOC(1)=NINT
     PRINT 3017, LOC(1)
     GO TO 3021
3014 LOC(IOPS)=NTRACE
     PRINT 3018,LOC(IOPS)
     GO TO 3021
3015 LAC=LOC(IOPS)+NINT
     IF (LAC.GT.100) LAC=LAC-100
     LOC(IOPS)=LAC
     PRINT 3019,PRI(IOPS),LOC(IOPS)
     GO TO 3021
3016 LAC=LOC(IOPS)+NINT
     IF (LAC.GT.100) LAC=LAC-100
     LOC(IOPS)=LAC
     PRINT 3020 ,PRI(IOPS),LOC(IOPS)
     GO TO 3021
3030 LAC=LOC(IOPS)+NINT
     IF (LAC.GT.100) LAC=LAC-100
     LOC(IOPS)=LAC
     PRINT 3031,PRI(IOPS),LOC(IOPS)
     GO TO 3021
3041 LAC=LOC(IOPS)+NINT
     IF (LAC.GT.100) LAC=LAC-100
     LOC(IOPS)=LAC
     PRINT 3039,PRI(IOPS),LOC(IOPS)
3021 PRINT 3022,NSTEP (IOPS)
3012 CONTINUE
     DO 3000 IOPS=1,NOPS
     DO 3033 KF=1,LF
     IF (LSTS(KF).NE.LOC(IOPS)) GO TO 3033
     RCO(IOPS)=AMR(KF)
     VCO(IOPS)=AMV(KF)
     GO TO 3000
3033 CONTINUE
3027 RCO(IOPS)=0.0
     VCO(IOPS)=0.0
3000 CONTINUE
3011 FORMAT (1H1,//,5X,'SUMMARY OF EJECTION COMPONENTS',/5X,'**********
    1*********************',//)
3017 FORMAT (5X,'ENTRY CONDITIONS (REAL) FOR STRAIGHT SECTION NO    =',I
    13)
3018 FORMAT (5X,'TRACE CONDITIONS SET FOR STRAIGHT SECTION NO',6X,'=',I
    13)
3019 FORMAT (5X,'QUADRUOPLE NO',I3,' SET IN STRAIGHT SECTION NO',7X,'='
    1,I3)
3020 FORMAT (5X,'SEXTUOPLE NO',I3,' SET IN STRAIGHT SECTION NO',8X,'=',
    1I3)
3022 FORMAT (5X,'NO OF MAGNETS TRAVERSED',48X,'=',I3,/)
3031 FORMAT (5X,'DIPOLE NO ',I3,'SET  IN STRAIGHT SECTION NO',10X,'=',I
    13)
```

```
3039 FORMAT (5X,'OCTOPOLE NO ',I3,'SET  IN STRAIGHT SECTION NO  ',06X,'
    1=',I3)
     RETURN
     END
```

```
      SUBROUTINE METRIC (Z,NM,MM,KN)
      DIMENSION Z(MM,NM)
C     CONVERT FROM MMS TO METRES ACCORDING TO THE VALUE OF KN
      IF (KN.EQ.2) GO TO 8003
      DO 8000 J=1,NM
      DO 8000 K=1,MM
 8000 Z(K,J)=Z(K,J)*0.001
      GO TO 8002
 8003 DO 8001 J=1,NM
      DO 8001 K=1,MM
 8001 Z(K,J)=Z(K,J)*1000.0
 8002 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE QUATRA (CORD,N,CONSQ,RCORR,VCORR,SHIFT,MOC)
      DIMENSION CORD(4,N)
      COMMON/F/RDISP
C
C     QUADRUOPLE TRAVERSAL USING THIN LENS APPROXIMATION
C
      DO 9000 J=1,N
      XQ=CORD(1,J)+RDISP+RCORR-((SHIFT*(2-MOC)))
      YQ=CORD(3,J)+VCORR-(SHIFT*(MOC-1))
      CORD(2,J)=CORD(2,J)-(CONSQ*XQ)
 9000 CORD(4,J)=CORD(4,J)+(CONSQ*YQ)
      RETURN
      END
```

```
      SUBROUTINE NORM(CORD,CORDN,N,NN,NST1)
      DIMENSION CORD(4,N),CORDN(4,N)
      COMMON/C/BF,BD
      COMMON/E/C1,C2
      COMMON/R/JPR
C     NORMALIZES PARTICLE CO-ORDINATES AND VICE VERSA ACCORDING TO THE
C     VALUE OF NN AND STRAIGHT SECTION
      NST2=NST1/2
      NREM=NST1-(NST2*2)
      IF (NREM.EQ.0) GO TO 5005
C      F FOCUCING
      D1=1.0
      D2=BF
      D3=C1
      D4=C2
      IF (JPR.EQ.0) GO TO 5006
      PRINT 5007,NST1
      GO TO 5006
C      D FOCUSING
 5005 D1=C1
      D2=C2
      D3=1.0
      D4=BF
      IF (JPR.EQ.0) GO TO 5006
      PRINT 5008,NST1
 5006 IF (NN.EQ.2) GO TO 5002
 5000 DO 5001 J=1,N
      CORD(1,J)=CORDN(1,J)/D1
      CORD(2,J)=CORDN(2,J)/D2
      CORD(3,J)=CORDN(3,J)/D3
       CORD(4,J)=CORDN(4,J)/D4
 5001 CONTINUE
      RETURN
 5002 DO 5003 J=1,N
      CORDN(1,J)=CORD(1,J)*D1
      CORDN(2,J)=CORD(2,J)*D2
      CORDN(3,J)=CORD(3,J)*D3
      CORDN(4,J)=CORD(4,J)*D4
 5003 CONTINUE
 5007 FORMAT (5X,'NORM F',I5)
 5008 FORMAT (5X,'NORM D',I5)
      RETURN
      END
```

```
      SUBROUTINE TRAVERS(CORD,CORDN,PS,N,IOPS,NLOC,NNSTEP)
      DIMENSION CORD(4,N),PS(4,4,10),Y(4,10),CORDN(4,N)
      COMMON/R/JPR
C     MATRIX OPERATIONS FOR MAGNET TRAVERSAL
      IF (JPR.EQ.0) GO TO 6012
      PRINT 6003,IOPS
      PRINT 6011,NNSTEP
 6012 CONTINUE
      CALL NORM(CORD,CORDN,N,2,NLOC)
      DO 6006 J=1,N
      DO 6006 I=1,4
 6006 Y(I,J)=0.0
      DO 6000 J=1,N
      DO 6000 I=1,4
      DO 6000 K=1,4
 6000 Y(I,J)=Y(I,J)+(PS(I,K,IOPS)*CORDN(K,J))
      DO 6004 J=1,N
      DO 6004 I=1,4
 6004 CORDN(I,J)=0.0
      DO 6001 J=1,N
      DO 6001 I=1,4
 6001 CORDN(I,J)=Y(I,J)
      NLIC=NLOC+NNSTEP
      CALL NORM(CORD,CORDN,N,1,NLIC)
      IF (JPR.EQ.0) GO TO 6002
      PRINT 6010
 6002 CONTINUE
 6003 FORMAT (/,5X,'IOPS  = ',I3)
 6010 FORMAT (5X,'END OF TRAVERS')
 6011 FORMAT (5X,'NSTEP =',I3)
      RETURN
      END
```

```
      SUBROUTINE TRANSF(PS,JST,J)
      DIMENSION PS(4,4,10)
      COMMON/D/QR,QV
C     TO CALCULATE THE TRANSFER MATRICES FOR MAGNET PERIODS (NSET/2)
C     MATRIX ELEMENTS SET FOR USE WITH NOTMALIZED CO-ORDINATES
      IF (J.NE.1) GO TO 195
      PRINT 194
  195 PI=3.14159
      R=QR*PI*0.02*JST
      V=QV*PI*0.02*JST
      PRINT 196,JST,R,V
      PS(1,1,J)=COS(R)
      PS(2,1,J)=-SIN(R)
      PS(3,1,J)=0.0
      PS(4,1,J)=0.0
      PS(1,2,J)=-PS(2,1,J)
      PS(2,2,J)=PS(1,1,J)
      PS(3,2,J)=0.0
      PS(4,2,J)=0.0
      PS(1,3,J)=0.0
      PS(2,3,J)=0.0
      PS(3,3,J)=COS(V)
      PS(4,3,J)=-SIN(V)
      PS(1,4,J)=0.0
      PS(2,4,J)=0.0
      PS(3,4,J)=-PS(4,3,J)
      PS(4,4,J)=PS(3,3,J)
      DO 197 I=1,4
  197 PRINT 199,(PS(I,K,J),K=1,4)
      PRINT 198
  194 FORMAT (5X,'MAGNET TRANSFER MATRICES',/,5X,'**********************
     1**',//)
  196 FORMAT (5X,'NO OF MAGNET UNITS TRAVERSED = ',I4,/5X,'RADIAL PHASE
     2JUMP',13X,'=',F10.4,/,5X,'VERTICAL PHASE JUMP',11X,'=',F10.4,/)
  198 FORMAT (//)
  199 FORMAT (4(5X,F10.5))
      RETURN
      END
```

```
      SUBROUTINE DATAOUT (CORD,CORDN,N,IREV,MREV,XT,XPT,YT,YPT)
      DIMENSION XT(MREV,N),XPT(MREV,N),YT(MREV,N),YPT(MREV,N)
      DIMENSION CORD(4,N),CORDN(4,N)
      DIMENSION ICRED(50),NCRED(50)
      COMMON/U/MPRINT
      IF (IREV.GT.1) GO TO 7006
      PRINT 7017
 7017 FORMAT          (5X,'X(MM)',9X,'XP(MRAD)',8X,'Z(MM)',9X,'ZP(MRAD)',
     18X,'X(MM)',9X,'XP(MMEQ)',8X,'Z(MM)',9X,'ZP(MMEQ)',/,5X,'**********
     2**************************************************************
     3*********************************',/)
      DO 7007 J=1,N
      ICRED(J)=0
 7007 NCRED(J)=200
 7006 CONTINUE
      JREV=IREV-1
      PRINT 7000,JREV
 7000 FORMAT (//,5X,'CYCLE NO = ',I3)
      DO 7002 J=1,N
      XMAX=SQRT((CORDN(1,J)**2)+(CORDN(2,J)**2))
      ZMAX=SQRT((CORDN(3,J)**2)+(CORDN(4,J)**2))
      IF (XMAX.GE.100.0.OR.ZMAX.GE.100.0) GO TO 7003
      IF (ICRED(J).EQ.1) GO TO 7008
      IF (MPRINT.EQ.0) GO TO 7012
      XT (IREV,J)=CORDN(1,J)
      XPT(IREV,J)=CORDN(2,J)
      YT (IREV,J)=CORDN(3,J)
      YPT(IREV,J)=CORDN(4,J)
      GO TO 7015
 7012 XT (IREV,J)=CORD(1,J)
      XPT(IREV,J)=CORD(2,J)
      YT (IREV,J)=CORD(3,J)
      YPT(IREV,J)=CORD(4,J)
 7015 PRINT 7001,((CORD(K,J),K=1,4),(CORDN(K,J),K=1,4))
      GO TO 7002
 7003 CONTINUE
      DO 7004 K=1,4
      CORD(K,J)=0.0
 7004 CORDN(K,J)=0.0
      XT (IREV,J)=0.0
      XT(IREV,J)=0.0
      XPT(IREV,J)=0.0
      YT(IREV,J)=0.0
      YPT(IREV,J)=0.0
      PRINT 7005,J
 7005 FORMAT (5X,'PARTICLE NO ',I2,' LOST')
      ICRED(J)=1
      NCRED(J)=IREV-1
      GO TO 7002
 7008 PRINT 7009,NCRED(J)
 7009 FORMAT (35X,'PARTICLE ALREADY LOST FROM TRACE ON CYCLE NO',I5)
      DO 7011 K=1,4
      CORD(K,J)=0.0
      CORDN(K,J)=0.0
 7011 CONTINUE
      XT(IREV,J)=0.0
      XPT(IREV,J)=0.0
      YT(IREV,J)=0.0
      YPT(IREV,J)=0.0
 7002 CONTINUE
 7001 FORMAT (4X,8(F7.3,8X))
      RETURN
      END
```

```
      SUBROUTINE SEXTRA (CORD,N,CONTS,MODE,RCORR,VCORR,SHITF,MOCS)
      DIMENSION CORD(4,N)
      COMMON/F/RDISP
C     SEXTUOPLE TRAVERSAL USING THIN LENS APPROXIMATION
C      MODE 1 CORRESPONDS TO SEXTUOPLE POLES ALONG THE XAXIS
C      ***  CONVENTION  ***
C      POSITIVE SEXTUOPLE CURRENT
C      NEGATIVE K2  NORMALIZED
C      POSITIVE K4  NORMALIZED
C      POSITIVE K6  UNNORMALIZED
      ADV=VCORR-(SHITF*(MOCS-1))
      ADR=RCORR+RDISP-(SHITF*(2-MOCS))
      DO 9011 J=1,N
      CORD(1,J)=CORD(1,J)+ADR
 9011 CORD(3,J)=CORD(3,J)+ADV
      CONSS=CONTS*1000.0
      IF (MODE.EQ.1) GO TO 9015
      L1=1
      L2=2
      L3=3
      L4=4
      GO TO 9016
 9015 L1=3
      L2=4
      L3=1
      L4=2
 9016 DO 9010 J=1,N
      XS=CORD(L1,J)
      ZS=CORD(L3,J)
      CORD(L2,J)=CORD(L2,J)+(((XS**2)-(ZS**2))*CONSS)
      CORD(L4,J)=CORD(L4,J)-((XS*ZS)*2.0*CONSS)
 9010 CONTINUE
      DO 9014 J=1,N
      CORD(1,J)=CORD(1,J)-RDISP
 9014 CONTINUE
      DO 9012 J=1,N
      CORD(1,J)=CORD(1,J)-ADR
 9012 CORD(3,J)=CORD(3,J)-ADV
      RETURN
      END
```

```
      SUBROUTINE DIPOL (CORD,N,LOCO,JNOPS,COND)
      DIMENSION CORD(4,N)
      JD2=(JNOPS*2)
      DO 9500 J=1,N
9500  CORD(JD2,J)=CORD(JD2,J)+COND
      RETURN
      END
```

```
      SUBROUTINE OCTRA (CORD,N,CONO,RCORR,VCORR,SHI,MD)
      DIMENSION CORD(4,N)
      COMMON/F/RDISP
C
C     OCTOPOLE TRAVERSAL USING THIN LENS APPROXIMATION
C
      ADX=RCORR+RDISP-(SHI*(2-MD))
      ADZ=VCORR-(SHI*(MD-1))
      DO 9400 J=1,N
      XS=ADX+CORD(1,J)
      ZS=ADZ+CORD(3,J)
      CORD(2,J)=CORD(2,J)+(CONO*XS*((XS**2)-(3.0*(ZS**2))))
      CORD(4,J)=CORD(4,J)+(CONO*ZS*((ZS**2)-(3.0*(XS**2))))
 9400 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DRAW (XT,XPT,YT,YPT,N,MREV)
      DIMENSION XT(MREV,N),XPT(MREV,N),YT(MREV,N),YPT(MREV,N),XA(100),YU
     1(100),XD(2),YD(2),XTEXT(3),YTEXT(3),XINT(1),YINT(1),XFIN(1),YFIN(1
     2)
      COMMON/A/NINT,NTRACE
      COMMON/X/NXP,NZP
      DATA XTEXT(1),XTEXT(2),XTEXT(3)/'     HORIZO','NTAL PHASE',' SPACE
     1  '/
      DATA YTEXT(1),YTEXT(2),YTEXT(3)/'      VERTI','CAL PHASE ','SPACE
     1   '/
      DATA DX,DN/60.0,-60.0/
      DB=DN-20.0
      DA=DX+20.0
      IF (NXP.EQ.0) L1=2
      IF (NZP.EQ.0) L2=1
      LREV=MREV-2
      CALL TVBGN(6)
      DO 9997 J=1,N
      DO 9996 L=L1,L2
      DO 9999 JREV=1,LREV
      IREV=JREV+1
      IF (L.EQ.2) GO TO 9995
      XA(JREV)=XT(IREV,J)
      YU(JREV)=XPT(IREV,J)
      GO TO 9994
 9995 XA(JREV)=YT(IREV,J)
      YU(JREV)=YPT(IREV,J)
 9994 XABS=ABS(XA(JREV))
      YABS=ABS(YU(JREV))
      IF (XABS.LE.DX.AND.YABS.LE.DX) GO TO 9999
      XA(JREV)=0.0
      YU(JREV)=0.0
 9999 CONTINUE
      IF (L.EQ.2) GO TO 9986
      XINT(1)=XT(1,J)
      YINT(1)=XPT(1,J)
      XFIN(1)=XT(MREV,J)
      YFIN(1)=XPT(MREV,J)
      GO TO 9985
 9986 XINT(1)=YT(1,J)
      YINT(1)=YPT(1,J)
      XFIN(1)=YT(MREV,J)
      YFIN(1)=YPT(MREV,J)
 9985 XABS=ABS(XFIN(1))
      YABS=ABS(YFIN(1))
      IF (XABS.LT.DX.OR.YABS.LT.DX) GO TO 9984
      XFIN(1)=0.0
      YFIN(1)=0.0
 9984 CONTINUE
      CALL TVRNG('USER',DB,DB,DA,DA)
      CALL TVAXIS ('X',DN,DN,DX,+1,4)
      CALL TVAXIS ('X',DN,DX,DX,+1,4)
      CALL TVAXIS ('Y',DX,DN,DX,+1,4)
      CALL TVAXIS ('Y',DN,DN,DX,+1,4)
      CALL TVLBL ('X',DN,DN,DX,+1,4,6H(F6.1),6)
      CALL TVLBL ('Y',DN,DN,DX,+1,4,6H(F6.1),6)
      IF (L.EQ.2) GO TO 9988
      CALL TVMTXT(0.0,(DX+10.0),XTEXT,30)
      GO TO 9987
 9988 CALL TVMTXT(0.0,(DX+10.0),YTEXT,30)
 9987 DO 9998 KL=1,2
      IF (KL.EQ.2) GO TO 9993
      XD(1)=DN
      XD(2)=DX
      GO TO 9991
 9993 YD(1)=DN
      YD(2)=DX
 9991 DO 9998 LD=1,3
      IF (KL.EQ.2) GO TO 9992
      YD(1)=DN+(DX*LD/2.0)
      YD(2)=YD(1)
      GO TO 9989
 9992 XD(1)=DN+(DX*LD/2.0)
      XD(2)=XD(1)
 9989 CALL TVDRAW (XD,YD)
 9998 CONTINUE
      CALL TVPLOT (XA,YU,LREV)
      CALL TVPLOT (XINT,YINT,1,'*')
      IF (XFIN(1).EQ.0.00) GO TO 9983
      CALL TVPLOT (XFIN,YFIN,1,'*')
 9983 IF (J.EQ.N.AND.L.EQ.2) GO TO 9990
      CALL TVNEXT
 9996 CONTINUE
 9997 CONTINUE
```

```
9990 CALL TVEND
     RETURN
     END
```