

```
!-----
!                               LCLS MAGNETIC MEASUREMENT FACILITY
!                               LEITZ PMM 12106 CMM
!
!                               QUINDOS MEASUREMENT PROGRAM
!-----
```

```
! Q VERSION : Q7
! DATE      : 16-APR-2008
! PART      : QUADRUPOLE FIDUCIAL BENCH
! PROGRAM   : QUAD WIRE V2.0.WDB
! PROGRAM BY: E. REESE/S. Anderson
```

```
!-----
!                               --- CHANGE LOG ---
!-----
```

DATE	VERSION	MODIFICATIONS
15-apr-08	1.1	completed output of values to txt file, output wire csy tra to file
16-apr-08	2.0	fixed minor problems, verified output, add polarity input to menu
18-apr-08	2.1	moved calculation of final mech csy in front of calc of final wire csy
25-apr-08	2.2	put in work around for EVATRA command which caused program crash, added polarity to element name of TB file to be imported back for average calc.
29-apr-08	2.	CREATE REPORT button enabled, averages PLUS and MINUS runs and averages the points. outputs data to a file

```
!EDIT          (NAM=INPUT_DLP, TYP=DLP)
100:ShowDialog (DLP=INPUT_DLP, OBJ=UVS:INPUT)
      GETVALS  (OBJ=UVS:INPUT, TYP=UVS, CDS=(A,D,C), CHS=(~QID,~OP,~POL))
!      LISCHS (~QID,~OP,~POL)

IFTHENS      (ST1=~OP, ST2=CAL, TYP=EQ)
```

```

    MEAS$OP=200
ELSEIFS      (ST1=~OP, ST2=TABLE, TYP=EQ)
    MEAS$OP=300
ELSEIFS      (ST1=~OP, ST2=ADJ, TYP=EQ)
    MEAS$OP=400
ELSEIFS      (ST1=~OP, ST2=MAN, TYP=EQ)
    MEAS$OP=550
ELSEIFS      (ST1=~OP, ST2=AUTO, TYP=EQ)
    MEAS$OP=500
ELSEIFS      (ST1=~OP, ST2=CREATE, TYP=EQ)
    MEAS$OP=600
ELSEIFS      (ST1=~OP, ST2=EXIT, TYP=EQ)
    MEAS$OP=700
ENDIF

GOTO          (LAB=MEAS$OP)

```

```

!    1.0 --- calibrate initial probe cluster
200:CONTIN
    EDBUPD          (OBJ=EDBPRB:*(), DEL=Y)
    DfnArtefact    (NAM=MAG$BASE, DIA=24.9998, SAZ=135.0, SEL=45.0, SDM=12.0)
    USECMM          (NAM=EDBCMM:SAVE$CMM)
    UseArtefact    (NAM=MAG$BASE)
    REFPRB         (ZOF=-135.0, DIA=3.00, PRB=PRB(1), DEL=N, SNT=PMM)
    CALSPH         (NAM=PRB(2), DIA=5.00, SNT=Leitz PMMLeitz PMM, DEL=N, UAD=N)
    CALSPH         (NAM=PRB(3), DIA=3.00, SNT=Leitz PMMLeitz PMM, DEL=N, UAD=N)
    CALSPH         (NAM=PRB(4), DIA=3.00, SNT=Leitz PMMLeitz PMM, DEL=N, UAD=N)
    CALSPH         (NAM=PRB(5), DIA=3.00, SNT=Leitz PMMLeitz PMM, DEL=N, UAD=N)
    USECSY CSY
    MOVCM          (TYP=DLT, DST=(, ,300))

MECOL          (NAM=SAFE$POS, CSY=CSY)

GOTO 100

```

```

!    1.0 --- find the 4 tooling ball supports
300:CONTIN
    USEPRB PRB(1)

    T=4
    MEPLA          (NAM=TOP(T), CSY=CSY, MOD=NOE, ITY=GSS)
    MECIR          (NAM=OD(T), CSY=CSY, MOD=NOE, PTY=TOP(T))
    BLDCSY        (NAM=POST_CSY, TYP=CAR, SPA=TOP(T), SDR=+Z, XZE=OD(T), YZE=OD(T), ZZE=TOP(T))
~RESULT$PT=TB(T)
~RESULT$CSY=CSY
    INDPRC TB_CALC
    MOVCM          (TYP=DLT, DST=(, ,450))

    T=2
    MEPLA          (NAM=TOP(T), CSY=CSY, MOD=NOE, ITY=GSS)
    MECIR          (NAM=OD(T), CSY=CSY, MOD=NOE, PTY=TOP(T))

```

```
BLDCSY          (NAM=POST_CSY, TYP=CAR, SPA=TOP(T), SDR=+Z, XZE=OD(T), YZE=OD(T), ZZE=TOP(T))
~RESULT$PT=TB(T)
~RESULT$CSY=CSY
INDPRC TB_CALC
MOVCMM          (TYP=DLT, DST=(, ,450))
```

```
T=1
MEPLA           (NAM=TOP(T), CSY=CSY, MOD=NOE, ITY=GSS)
MECIR           (NAM=OD(T), CSY=CSY, MOD=NOE, PTY=TOP(T))
BLDCSY          (NAM=POST_CSY, TYP=CAR, SPA=TOP(T), SDR=+Z, XZE=OD(T), YZE=OD(T), ZZE=TOP(T))
~RESULT$PT=TB(T)
~RESULT$CSY=CSY
INDPRC TB_CALC
MOVCMM          (TYP=DLT, DST=(, ,450))
```

```
T=3
MEPLA           (NAM=TOP(T), CSY=CSY, MOD=NOE, ITY=GSS)
MECIR           (NAM=OD(T), CSY=CSY, MOD=NOE, PTY=TOP(T))
BLDCSY          (NAM=POST_CSY, TYP=CAR, SPA=TOP(T), SDR=+Z, XZE=OD(T), YZE=OD(T), ZZE=TOP(T))
~RESULT$PT=TB(T)
~RESULT$CSY=CSY
INDPRC TB_CALC
MOVCMM          (TYP=DLT, DST=(, ,450))
```

```
COLPTS          (NAM=TB_PLA, CSY=CSY, DEL=Y, ELE=(TB(1), TB(2), TB(3)), TYP=PLA)
COLPTS          (NAM=TB_AXI, CSY=CSY, DEL=Y, ELE=(TB(1), TB(2)), TYP=AXI)
BLDCSY          (NAM=TABLE_CSY, TYP=CAR, SPA=TB_PLA, SDR=+Y, PLA=TB_AXI, PDR=+Z, XZE=TB(1))
TRAELE          (NEW=TB_NEW(4), TRA=TABLE_CSY, OLD=TB(4), RPL=N, EVA=Y)
```

```
MECOL           (NAM=SAFE$POS, CSY=CSY)
GOTO 100
```

```
! ---- measurement of quad tooling balls for initial location
```

```
! -- qtb-1
400:CONTIN
  SetActiveTool (NAM=PRB(1))
!   USEPRB PRB(1)
  DFNCMM        (NAM=OFF$2, POF=3, ENB=N, CPY=EDBCMM:SAVE$CMM)

  USECMM OFF$2
  QTB=1
  T=1
  MEPLA         (NAM=QUAD_P(QTB), CSY=TABLE_CSY, MOD=NOE, ITY=GSS)
  MECIR         (NAM=QUAD_C(QTB), CSY=TABLE_CSY, MOD=NOE, PTY=QUAD_P(QTB))
  BLDCSY        (NAM=POST_CSY, TYP=CAR, SPA=QUAD_P(QTB), SDR=+Z, XZE=QUAD_C(QTB), YZE=QUAD_C(QTB))
  USECMM SAVE$CMM
  ~RESULT$PT=QUAD_TB(T)
  ~RESULT$CSY=TABLE_CSY
  INDPRC TB_CALC

  USECMM OFF$2
  QTB=2
  T=2
  MEPLA         (NAM=QUAD_P(QTB), CSY=TABLE_CSY, MOD=NOE, ITY=GSS)
```

```

MECIR          (NAM=QUAD_C(QTB), CSY=TABLE_CSY, MOD=NOE, PTY=QUAD_P(QTB))
BLDCSY        (NAM=POST_CSY, TYP=CAR, SPA=QUAD_P(QTB), SDR=+Z, XZE=QUAD_C(QTB), YZE=QUAD_C(QTB))
USECMM SAVE$CMM
~RESULT$PT=QUAD_TB(T)
~RESULT$CSY=TABLE_CSY
INDPRC TB_CALC

USECMM OFF$2
QTB=3
T=3
MEPLA          (NAM=QUAD_P(QTB), CSY=TABLE_CSY, MOD=NOE, ITY=GSS)
MECIR          (NAM=QUAD_C(QTB), CSY=TABLE_CSY, MOD=NOE, PTY=QUAD_P(QTB))
BLDCSY        (NAM=POST_CSY, TYP=CAR, SPA=QUAD_P(QTB), SDR=+Z, XZE=QUAD_C(QTB), YZE=QUAD_C(QTB))
USECMM SAVE$CMM
~RESULT$PT=QUAD_TB(T)
~RESULT$CSY=TABLE_CSY
INDPRC TB_CALC

USECMM OFF$2
QTB=4
T=4
MEPLA          (NAM=QUAD_P(QTB), CSY=TABLE_CSY, MOD=NOE, ITY=GSS)
MECIR          (NAM=QUAD_C(QTB), CSY=TABLE_CSY, MOD=NOE, PTY=QUAD_P(QTB))
BLDCSY        (NAM=POST_CSY, TYP=CAR, SPA=QUAD_P(QTB), SDR=+Z, XZE=QUAD_C(QTB), YZE=QUAD_C(QTB))
USECMM SAVE$CMMH
~RESULT$PT=QUAD_TB(T)
~RESULT$CSY=TABLE_CSY
INDPRC TB_CALC

COLPTS        (NAM=QUAD_TB$PLA, CSY=TABLE_CSY, DEL=Y, ELE=(QUAD_TB(4), QUAD_TB(3), QUAD_TB(2), QUAD_TB(1)))
COLPTS        (NAM=QUAD_TB$AXI, CSY=TABLE_CSY, DEL=Y, ELE=(QUAD_TB(2), QUAD_TB(1)), TYP=CAR)
BLDCSY        (NAM=QUAD_TB$CSY, TYP=CAR, SPA=QUAD_TB$PLA, SDR=+Y, PLA=QUAD_TB$AXI, PDR=+Z)

MECOL          (NAM=SAFE$POS, CSY=CSY)
!--- stripout extra characters from QUAD NAME
~QID6=~QID
~NULL=' '
SUBSTR         (NAM=~QID6, LM1=7, RSD=~NULL)

CONCAT        (NAM=~QUAD$DIR, STR=('C:\Quad Transfer Data\Undulator Quad Transformation\
CONCAT        (NAM=~QUAD$TRA, STR=('Quad', ' ',~QID6,'.TRA'), LMG=1, INI=Y)
CONCAT        (NAM=~FIL$CNV, STR=(~QUAD$DIR,~QUAD$TRA), LMG=1, INI=Y)
CNVFIL ~FIL$CNV

TRACSY        (NEW=QUAD_MECH$CSY, TRA=CSY$TRA_P, OLD=QUAD_TB$CSY, CAD=N)
FINDTRA       (NAM=QUAD_Z$LOC, CS1=TABLE_CSY, CS2=QUAD_MECH$CSY)
! ---- location of Quad Mech c\1 to table csy
GETVALS       (OBJ=QUAD_Z$LOC, TYP=TRA, RDS=(Z,X,Y) ,REA=(QUAD$Z,QUAD$X,QUAD$Y))
Z_CORR=253.1-QUAD$Z

CVREACHS      (NAM=~QUAD$Z, VAL=QUAD$Z, INT=N, ANG=N, SPZ=N, RLS=Y, RTZ=Y)
CVREACHS      (NAM=~QUAD$CORR, VAL=Z_CORR, INT=N, ANG=N, SPZ=N, RLS=Y, RTZ=Y)

!EDTMSG QUAD$Z
CPYMSG        (FRM=QUAD$Z, TO =QUAD$Z(1), DEL=Y)
RPLSTR        (NAM=MSG:QUAD$Z(1), FRM=zzzzzz, TO =~QUAD$Z)
RPLSTR        (NAM=MSG:QUAD$Z(1), FRM=aaaaaa, TO =~QUAD$CORR)

```

```

INQYESNO      (CHS=~ITR(1), MSG=QUAD$Z(1), RED=Y)

!EDTACT       (NAM=QUAD_CSY$YAX)
TRAELE        (NEW=YAX_TO_TABLE, TRA=TABLE_CSY, OLD=QUAD_CSY$YAX, RPL=N, EVA=Y)
GETVALS YAX_TO_TABLE.EVA(1), , D, ROLL$DEG
ROLL$MRAD=ROLL$DEG*((F_PI(1)/180)*1000)
CVREACHS      (NAM=~RMAD, VAL=ROLL$MRAD, INT=N, ANG=N, SPZ=N, RLS=Y, RTZ=Y)

ADJ$VAL=(( -198)*FTAN(ROLL$DEG))/2
CVREACHS      (NAM=~ADJ_R, VAL=ADJ$VAL, INT=N, ANG=N, SPZ=N, RLS=Y, RTZ=Y)
CVREACHS      (NAM=~ADJ_L, VAL=-1*ADJ$VAL, INT=N, ANG=N, SPZ=N, RLS=Y, RTZ=Y)

!EDTMSG       (NAM=ROLL$MSG)tund
CPYMSG        (FRM=ROLL$MSG, TO =ROLL$MSG(1), DEL=Y)
RPLSTR        (NAM=MSG:ROLL$MSG(1), FRM=xxxxxx, TO =~RMAD)
RPLSTR        (NAM=MSG:ROLL$MSG(1), FRM=yyyyyy, TO =~ADJ_R)
RPLSTR        (NAM=MSG:ROLL$MSG(1), FRM=zzzzzz, TO =~ADJ_L)

INQYESNO      (CHS=~ITR, MSG=ROLL$MSG(1), RED=Y)

REMEAS=0
IFTHENS       (ST1=~ITR(1), ST2=N, TYP=EQ, LEN=1)
  REMEAS=1
ELSEIFS       (ST1=~ITR, ST2=N, TYP=EQ, LEN=1)
  REMEAS=1
ENDIF

IFGOTO        (VAL=REMEAS, EQ0=100, GT0=400)

```

550:CONTIN

```

!EDTWKP       (NAM=DET_CALIBRATION)
! --- measure the tooling balls on the detectors --
! --- build local csy and of set along line between balls form ball 1 of each set
START         (WKP=DET_CALIBRATION, SER=~QID, TOP=N)
DFNCMM        (NAM=SLOW_OFF2, PSP=0.5, POF=1.5, ENB=Y, CPY=EDBCMM:SAVE$CMM)
USECMM        (NAM=SLOW_OFF2)
! -- first detector (X)
D=1
DX(1)=-47.34611
USEPRB PRB(3)
MESPH         (NAM=PD_TB1A(D), CSY=TABLE_CSY, ITY=GSS)
USEPRB PRB(5)
MESPH         (NAM=PD_TB2A(D), CSY=TABLE_CSY, ITY=GSS)
COLPTS        (NAM=PD_LINE(D), CSY=TABLE_CSY, DEL=Y, ELE=(PD_TB2A(D), PD_TB1A(D)), TYP=AXI)
BLDCSY        (NAM=PD_CSY(D), TYP=CAR, SPA=PD_LINE(D), SDR=+X, PLA=TB_PLA, PDR=+Y, XZE=PD_TB
TRAOBJ        (NEW=PD_SHIFT$CSY(D), OLD=PD_CSY(D), SHX=DX(1), AXI=+Z, DTY=CSY)
! -- second detector (Y)
D=2
DY(2)=-47.51918
USEPRB PRB(5)
MESPH         (NAM=PD_TB1(D), CSY=TABLE_CSY, ITY=GSS)
MESPH         (NAM=PD_TB2(D), CSY=TABLE_CSY, ITY=GSS)
COLPTS        (NAM=PD_LINE(D), CSY=TABLE_CSY, DEL=Y, ELE=(PD_TB2(D), PD_TB1(D)), TYP=AXI)

```

```

BLDCSY          (NAM=PD_CSY(D), TYP=CAR, SPA=PD_LINE(D), SDR=+Y, PLA=TB_AXI, PDR=+Z, XZE=PD_TB
TRAOBJ          (NEW=PD_SHIFT$CSY(D), OLD=PD_CSY(D), SHY=DY(2), AXI=+Z, DTY=CSY)

! -- third detector (Y)
D=3
DY(3)=-47.66513
USEPRB PRB(3)
MESPH          (NAM=PD_TB2(D), CSY=TABLE_CSY, ITY=GSS)
MESPH          (NAM=PD_TB1(D), CSY=TABLE_CSY, ITY=GSS)
COLPTS        (NAM=PD_LINE(D), CSY=TABLE_CSY, DEL=Y, ELE=(PD_TB2(D), PD_TB1(D)), TYP=AXI)
BLDCSY        (NAM=PD_CSY(D), TYP=CAR, SPA=PD_LINE(D), SDR=+Y, PLA=TB_AXI, PDR=+Z, XZE=PD_TB
TRAOBJ        (NEW=PD_SHIFT$CSY(D), OLD=PD_CSY(D), SHY=DY(3), AXI=+Z, DTY=CSY)

! -- fourth detector (X)
D=4
DX(4)=-47.43996
USEPRB PRB(3)
MESPH          (NAM=PD_TB1A(D), CSY=TABLE_CSY, ITY=GSS)
USEPRB PRB(5)
MESPH          (NAM=PD_TB2A(D), CSY=TABLE_CSY, ITY=GSS)
COLPTS        (NAM=PD_LINE(D), CSY=TABLE_CSY, DEL=Y, ELE=(PD_TB2A(D), PD_TB1A(D)), TYP=AXI)
BLDCSY        (NAM=PD_CSY(D), TYP=CAR, SPA=PD_LINE(D), SDR=+X, PLA=TB_PLA, PDR=+Y, XZE=PD_TB
TRAOBJ        (NEW=PD_SHIFT$CSY(D), OLD=PD_CSY(D), SHX=DX(4), AXI=+Z, DTY=CSY)

USECSY        (NAM=CSY)
MOVCMM        (TYP=DLT, DST=(, ,100))
! ---- build detector csy

! --- define element at center of each detector csy
DFNELE        (NAM=XD$PT(1), TYP=POI, CSY=PD_SHIFT$CSY(1), DEL=N)
DFNELE        (NAM=YD$PT(2), TYP=POI, CSY=PD_SHIFT$CSY(2), DEL=N)
DFNELE        (NAM=YD$PT(3), TYP=POI, CSY=PD_SHIFT$CSY(3), DEL=N)
DFNELE        (NAM=XD$PT(4), TYP=POI, CSY=PD_SHIFT$CSY(4), DEL=N)

! --- move x detector center points into table csy and
! --- create temp wire axis between center of x detectors
TRAELE        (NEW=US_X$PT, TRA=TABLE_CSY, OLD=XD$PT(1), EVA=Y)
TRAELE        (NEW=DS_X$PT, TRA=TABLE_CSY, OLD=XD$PT(4), EVA=Y)
COLPTS        (NAM=TEMP$WIRE, CSY=TABLE_CSY, DEL=Y, ELE=(DS_X$PT,US_X$PT), TYP=AXI)

! --- project and intersect y detector axis onto temp wire axis
INTPNT        (NAM=US_X$PT, CSY=TABLE_CSY, EL1=PD_LINE(2), EL2=TEMP$WIRE, PPI=IM)
INTPNT        (NAM=DS_X$PT, CSY=TABLE_CSY, EL1=PD_LINE(3), EL2=TEMP$WIRE, PPI=IM)

! --- move y detector center points into table csy and
TRAELE        (NEW=US_Y$PT, TRA=TABLE_CSY, OLD=YD$PT(2))
TRAELE        (NEW=DS_Y$PT, TRA=TABLE_CSY, OLD=YD$PT(3))

! --- extract coordinate vales of points and assign to variables
GETVALS       (OBJ=DS_X$PT, RDS=(X,Z), REA=(DS$X,DS$Z))
GETVALS       (OBJ=DS_Y$PT, RDS=Y, REA=DS$Y)
GETVALS       (OBJ=US_X$PT, RDS=(X,Z), REA=(US$X,US$Z))
GETVALS       (OBJ=US_Y$PT, RDS=Y, REA=US$Y)

! --- create wire endpoints by creating points with extracted coordinates
DFNELE        (NAM=DS_WIRE$PT, CSY=TABLE_CSY, DEL=N)
PUTVALS       (OBJ=DS_WIRE$PT, TYP=ELE, RDS=(X,Y,Z), VAL=(DS$X,DS$Y,DS$Z))

```

```

DFNELE          (NAM=US_WIRE$PT, CSY=TABLE_CSY, DEL=N)
PUTVALS        (OBJ=US_WIRE$PT, TYP=ELE, RDS=(X,Y,Z), VAL=(US$X,US$Y,US$Z))

! --- create final wire axis
COLPTS        (NAM=WIRE$AXI, CSY=TABLE_CSY, DEL=Y, ELE=(US_WIRE$PT,DS_WIRE$PT), TYP=AXI)
!EDTAPT  WIRE$AXI
! --- perpendicual distance from pin wire axis to detector points
!EDTACT      (NAM=WIRE_PIN$DS)
!EDTACT      (NAM=WIRE_PIN$US)
!COLPTS      (NAM=PIN_WIRE, CSY=TABLE_CSY, DEL=Y, ELE=(WIRE_PIN$DS,WIRE_PIN$US), TYP=AXI)
PEPNTAXI     (NAM=US_DT_OFF, CSY=TABLE_CSY, EL1=US_WIRE$PT, EL2=PIN_WIRE.$AXI)
PEPNTAXI     (NAM=DS_DT_OFF, CSY=TABLE_CSY, EL1=DS_WIRE$PT, EL2=PIN_WIRE.$AXI)

GETVALS      (OBJ=US_DT_OFF.EVA(1), RDS=D, REA=OFF_X)
GETVALS      (OBJ=US_DT_OFF.EVA(2), RDS=D, REA=OFF_Y)
GETVALS      (OBJ=US_DT_OFF.EVA(3), RDS=D, REA=OFF_Z)
DFNELE      (NAM=US$OFFSET, DEL=N)
PUTVALS      (OBJ=US$OFFSET, RDS=(X,Y,Z), VAL=(OFF_X,OFF_Y,OFF_Z))

GETVALS      (OBJ=DS_DT_OFF.EVA(1), RDS=D, REA=OFF_X)
GETVALS      (OBJ=DS_DT_OFF.EVA(2), RDS=D, REA=OFF_Y)
GETVALS      (OBJ=DS_DT_OFF.EVA(3), RDS=D, REA=OFF_Z)
DFNELE      (NAM=DS$OFFSET, DEL=N)
PUTVALS      (OBJ=DS$OFFSET, RDS=(X,Y,Z), VAL=(OFF_X,OFF_Y,OFF_Z))

! ---- measure quad tooling balls in final location
! -- qtb-1

USEPRB PRB(1)
DFNCMM      (NAM=OFF$2, POF=3, ENB=N, CPY=EDBCMM:SAVE$CMM)

USECMM OFF$2
QTB=1
T=1
MEPLA      (NAM=QUAD_P(QTB), CSY=TABLE_CSY, MOD=NOE, ITY=GSS)
MECIR      (NAM=QUAD_C(QTB), CSY=TABLE_CSY, MOD=NOE, PTY=QUAD_P(QTB))
BLDCSY     (NAM=POST_CSY, TYP=CAR, SPA=QUAD_P(QTB), SDR=+Z, XZE=QUAD_C(QTB), YZE=QUAD_C(QTB))
USECMM SAVE$CMM
~RESULT$PT=QUAD_TB(T)
~RESULT$CSY=TABLE_CSY
INDPRC TB_CALC

USECMM OFF$2
QTB=2
T=2
MEPLA      (NAM=QUAD_P(QTB), CSY=TABLE_CSY, MOD=NOE, ITY=GSS)
MECIR      (NAM=QUAD_C(QTB), CSY=TABLE_CSY, MOD=NOE, PTY=QUAD_P(QTB))
BLDCSY     (NAM=POST_CSY, TYP=CAR, SPA=QUAD_P(QTB), SDR=+Z, XZE=QUAD_C(QTB), YZE=QUAD_C(QTB))
USECMM SAVE$CMM
~RESULT$PT=QUAD_TB(T)
~RESULT$CSY=TABLE_CSY
INDPRC TB_CALC

USECMM OFF$2
QTB=3
T=3

```

```

MEPLA          (NAM=QUAD_P(QTB), CSY=TABLE_CSY, MOD=NOE, ITY=GSS)
MECIR          (NAM=QUAD_C(QTB), CSY=TABLE_CSY, MOD=NOE, PTY=QUAD_P(QTB))
BLDCSY        (NAM=POST_CSY, TYP=CAR, SPA=QUAD_P(QTB), SDR=+Z, XZE=QUAD_C(QTB), YZE=QUAD
USECMM SAVE$CMM
~RESULT$PT=QUAD_TB(T)
~RESULT$CSY=TABLE_CSY
  INDPRC TB_CALC

USECMM OFF$2
QTB=4
T=4
MEPLA          (NAM=QUAD_P(QTB), CSY=TABLE_CSY, MOD=NOE, ITY=GSS)
MECIR          (NAM=QUAD_C(QTB), CSY=TABLE_CSY, MOD=NOE, PTY=QUAD_P(QTB))
BLDCSY        (NAM=POST_CSY, TYP=CAR, SPA=QUAD_P(QTB), SDR=+Z, XZE=QUAD_C(QTB), YZE=QUAD
USECMM SAVE$CMMH
~RESULT$PT=QUAD_TB(T)
~RESULT$CSY=TABLE_CSY
  INDPRC TB_CALC

COLPTS        (NAM=QUAD_TB$PLA, CSY=TABLE_CSY, DEL=Y, ELE=( QUAD_TB(3), QUAD_TB(2), QUAD
COLPTS        (NAM=QUAD_TB$AXI, CSY=TABLE_CSY, DEL=Y, ELE=(QUAD_TB(2), QUAD_TB(1)), TYP=
BLDCSY        (NAM=QUAD_TB$CSY, TYP=CAR, SPA=QUAD_TB$PLA, SDR=+Y, PLA=QUAD_TB$AXI, PDR=X
TRACSY        (NEW=QUAD_MECH$CSY, TRA=CSY$TRA, OLD=QUAD_TB$CSY, CAD=N)

TRAELE        (NEW=YAX_TO_TABLE, TRA=TABLE_CSY, OLD=QUAD_CSY$YAX, RPL=N, EVA=Y)
GETVALS YAX_TO_TABLE.EVA(1), , D, ROLL$DEG
ROLL$MRAD=ROLL$DEG*((F_PI(1)/180)*1000)
DFNELE        (NAM=QUAD$ROLL, DEL=N)
PUTVALS        (OBJ=QUAD$ROLL, RDS=(X,Y,Z), VAL=(ROLL$MRAD,0,0))
CVREACHS      (NAM=~RMAD, VAL=ROLL$MRAD, INT=N, ANG=N, SPZ=N, RLS=Y, RTZ=Y)
  FINDTRA      (NAM=QUAD_Z$LOC, CS1=TABLE_CSY, CS2=QUAD_MECH$CSY)
  EVATRA       (PNT=QUAD_Z$LOC, TRA=QUAD_Z$LOC, MOD=EVA)
!LISCHS ~RMAD
! measure quad tb 5 in quad tb csy

USEPRB PRB(2)
T=5
~RESULT$PT=QUAD_TB(T)
~RESULT$CSY=TABLE_CSY
USECMM          (NAM=OFF$2)
MEPLA (NAM=TBA_PLA(T), CSY=QUAD_TB$CSY, ITY=GSS)
MECYL          (NAM=TBA_CYL(T), CSY=QUAD_TB$CSY, ITY=CY3)
BLDCSY (NAM=TBA_CSY(T), TYP=CAR, SPA=TBA_PLA(T), SDR=+Z, XZE=TBA_CYL(T), YZE=TBA_CYL(T),
DFNELE (NAM=PLA_25P4(T), TYP=PLA, CSY=TBA_CSY(T), DEL=N)
PUTVALS (OBJ=PLA_25P4(T), RDS=(X,Y,Z,U,V,W), VAL=(0,0,25.4,0,0,1))
INTPNT (NAM=~RESULT$PT, CSY=~RESULT$CSY, EL1=PLA_25P4(T), EL2=TBA_CYL(T).$AXI, PPI=NO, C

! --- get the sigma of TB_PLA() and TB_CYL() and put into fields D&E of pnt for tb loc.
DO (NAM=I, BGN=1, END=5, DLT=1)
  GETVALS (OBJ=TBA_PLA(I), RDS=F, REA=P$SIG)
  GETVALS (OBJ=TBA_CYL(I), RDS=F, REA=C$SIG)
  PUTVALS (OBJ=QUAD_TB(I), RDS=(D,E), VAL=(P$SIG,C$SIG))
ENDDO

```

```

MECOL                (NAM=SAFE$POS, CSY=CSY)

!--- update quad mech csy in ref to balls 1,2,3,5 to table csy
COLPTS              (NAM=QUAD_TB$PLA, CSY=TABLE_CSY, DEL=Y, ELE=(QUAD_TB(3), QUAD_TB(2), QUAD_TB(1)), TYP=CAR)
COLPTS              (NAM=QUAD_TB$AXI, CSY=TABLE_CSY, DEL=Y, ELE=(QUAD_TB(2), QUAD_TB(1)), TYP=CAR)
BLDCSY              (NAM=QUAD_TB$CSY, TYP=CAR, SPA=QUAD_TB$PLA, SDR=+Y, PLA=QUAD_TB$AXI, PDR=X)

TRACSY              (NEW=QUAD_MECH$CSY, TRA=CSY$TRA_F, OLD=QUAD_TB$CSY, CAD=N)
FINDTRA             (NAM=QUAD_Z$LOC, CS1=TABLE_CSY, CS2=QUAD_MECH$CSY)
!***** EVATRA causes program crash *****
!   EVATRA          (PNT=QUAD_MECH$LOC, TRA=QUAD_Z$LOC, MOD=EVA)

!---- work around for EVATRA which causes program crash -----
DFNELE              (NAM=QUAD_MECH$LOC, TYP=POI, CSY=TABLE_CSY, DEL=*)
GETVALS             (OBJ=QUAD_Z$LOC, TYP=TRA, RDS=(X,Y,Z), REA=(TX,TY,TZ))
PUTVALS             (OBJ=QUAD_MECH$LOC, RDS=(X,Y,Z), VAL=(TX,TY,TZ))

! --- build final wire csy

BLDCSY              (NAM=FINAL$WIRE_QR, TYP=CAR, SPA=WIRE$AXI, SDR=+Z, PLA=QUAD_MECH$CSY.$YDI, PDR=X)
BLDCSY              (NAM=FINAL$WIRE_TR, TYP=CAR, SPA=WIRE$AXI, SDR=+Z, PLA=TABLE_CSY.$YDI, PDR=X)
DFNELE              (NAM=QUAD$ZERO, TYP=POI, CSY=FINAL$WIRE_TR, DEL=N)
TRAELE              (NEW=QUAD$ZERO_T, TRA=TABLE_CSY, OLD=QUAD$ZERO, EVA=Y)

! --- translate quad tooling ball coordinates into final csys

DO                  (NAM=I, BGN=1, END=5, DLT=1)
  TRAELE            (NEW=QUAD_TB$QR(I), TRA=FINAL$WIRE_QR, OLD=QUAD_TB(I), RPL=N)
  TRAELE            (NEW=QUAD_TB$TR(I), TRA=FINAL$WIRE_TR, OLD=QUAD_TB(I), RPL=N)
ENDDO

! ---- get ave of coordinates for balls 1-4
DO                  (NAM=I, BGN=1, END=4, DLT=1)
  GETVALS           (OBJ=QUAD_TB$QR(I), RDS=(X,Y,Z), REA=(QR$X(I),QR$Y(I),QR$Z(I)))
  GETVALS           (OBJ=QUAD_TB$TR(I), RDS=(X,Y,Z), REA=(TR$X(I),TR$Y(I),TR$Z(I)))
ENDDO

TB$QR_X=(QR$X(1)+QR$X(2)+QR$X(3)+QR$X(4))/4
TB$QR_Y=(QR$Y(1)+QR$Y(2)+QR$Y(3)+QR$Y(4))/4
TB$QR_Z=(QR$Z(1)+QR$Z(2)+QR$Z(3)+QR$Z(4))/4
DFNELE              (NAM=QR$Y(1234), CSY=FINAL$WIRE_QR, DEL=*)
PUTVALS             (OBJ=QR$Y(1234), RDS=(X,Y,Z), VAL=(TB$QR_X,TB$QR_Y,TB$QR_Z))

TB$TR_X=(TR$X(1)+TR$X(2)+TR$X(3)+TR$X(4))/4
TB$TR_Y=(TR$Y(1)+TR$Y(2)+TR$Y(3)+TR$Y(4))/4
TB$TR_Z=(TR$Z(1)+TR$Z(2)+TR$Z(3)+TR$Z(4))/4
DFNELE              (NAM=TR$Y(1234), CSY=FINAL$WIRE_TR, DEL=*)

```

```

PUTVALS          (OBJ=TR$Y(1234), RDS=(X,Y,Z), VAL=(TB$TR_X,TB$TR_Y,TB$TR_Z))

!--- collect tb coordinates and ave of 1,2,3,4 into final csy
COLPTS          (NAM=QUAD$T_BALLS, CSY=FINAL$WIRE_QR, DEL=Y, ELE=(QUAD_TB$QR(1), QUAD_TB$
! --- load tooling ball coordinates from mech. fid.
USECSY          (NAM=FINAL$WIRE_QR)
DELELE NOM$TB,N$
CONCAT          (NAM=~QUAD$DIR, STR=('C:\Quad Transfer Data\Undulator Quad Transformation\
CONCAT          (NAM=~QUAD$TB_FILE, STR=('Quad', ' ',~QID6, ' ','Tooling Balls','.txt'), L
CONCAT          (NAM=~FIL$CNV, STR=(~QUAD$DIR,~QUAD$TB_FILE), LMG=1, INI=Y)
CNVFIL ~FIL$CNV

DO              (NAM=I, BGN=1, END=5, DLT=1)
  COLAPT        (NAM=QUAD$MECH(I), CSY=FINAL$WIRE_QR, DEL=Y, PTS=NOM$TB.ACT.PT(I), FRS=
ENDDO

DO              (NAM=I, BGN=1, END=5, DLT=1)
  GETVALS      (OBJ=NOM$TB.ACT.PTS(I), RDS=(X,Y,Z), REA=(MECH$X,MECH$Y,MECH$Z))
  GETVALS      (OBJ=QUAD_TB$QR(I), TYP=ELE, RDS=(X,Y,Z), REA=(ACT_X,ACT_Y,ACT_Z))
  DFNELE      (NAM=M_W$DIFF(I), CSY=FINAL$WIRE_QR, DEL=N)
  PUTVALS      (OBJ=M_W$DIFF(I), RDS=(X,Y,Z), VAL=(ACT_X-MECH$X,ACT_Y-MECH$Y,ACT_Z-MEC
ENDDO

! --- get run number from WKP and append to file name

CONCAT          (NAM=~CMM$DIR, STR=('C:\CMM DATA\ ',QUAD,' ', ~QID,'\ '), LMG=1, INI=Y)
CONCAT          (NAM=~CMM$FILE, STR=(~CMM$DIR,'CMM RUN', ' ',~QID,' ',~POL,'.TXT'), LMG=1,
CONCAT          (NAM=~T_BALL$FILE, STR=(~CMM$DIR,'TOOLING BALLS', ' ',~QID,' ',~POL,'.TXT'
CONCAT          (NAM=~QUAD$FILE, STR=('C:\CMM DATA\QUAD', ' ',~QID,, ' ',~POL,'.WDB'), LMG
MAKEDIR         (DIR=~CMM$DIR, OPT=C)
! --- output tooling ball data to file
OPEN            (FIL=~CMM$FILE, DEV=D1, STA=NEW, ACC=A)
OUTPUT          (NAM=$SCOTT, DEV=D1, TYP=ELE, STY=ACT, MSK=ACT$OUT)
CLOSE          (DEV=D1)
! --- add polarity to element name so mane is unique
CONCAT          (NAM=~T$BALL_ELE, STR=(QUAD$T_BALLS_,~POL), LMG=1, INI=Y)
CPYAPT         (FRM=QUAD$T_BALLS, TO =~T$BALL_ELE, DEL=Y)
FMTOBJ         (FIL=~T_BALL$FILE, NAM=~T$BALL_ELE, STA=NEW, TYP=ELE, STY=APT, DSC=(X,Y,Z)

!--- output tra from tooling ball csy to wire csy
CONCAT          (NAM=~QUAD$DIR, STR=('C:\Quad Transfer Data\Undulator Quad Transformation\
CONCAT          (NAM=~QUAD$TRA, STR=('Quad', ' ',~QID6,' ',WIRE,'.TRA'), LMG=1, INI=Y)
CONCAT          (NAM=~FIL$TRA, STR=(~QUAD$DIR,~QUAD$TRA), LMG=1, INI=Y)

FINDTRA        (NAM=QUAD_WIRE$LOC, CS1=QUAD_TB$CSY, CS2=FINAL$WIRE_QR)
FMTOBJ         (FIL=~FIL$TRA, NAM=QUAD_WIRE$LOC, TYP=TRA)

!--- save complete wdb
CONCAT          (NAM=~QUAD$FILE, STR=('C:\CMM DATA\QUAD', ' ',~QID,, ' ',~POL,'.WDB'), LMG
SAVE           (FIL=~QUAD$FILE)

GOTO 100

```

```

! --- calculate and ouput average of plus and minus tooling ball values
600:CONTIN
  USECSY CSY
  DELELE      (NAM=(QUAD$T_BALLS_MINUS, QUAD$T_BALLS_PLUS), CNF=N)
  DELELE      (NAM=ELE:AVE$TB(), CNF=N)
  CONCAT      (NAM=~T_BALL$FILE_P, STR=(~CMM$DIR, 'TOOLING BALLS', ' ', ~QID, ' ', PLUS, '.TX
  CONCAT      (NAM=~T_BALL$FILE_M, STR=(~CMM$DIR, 'TOOLING BALLS', ' ', ~QID, ' ', MINUS, '.T
  CNVFIL ~T_BALL$FILE_P
  CNVFIL ~T_BALL$FILE_M

  DO          (NAM=I, BGN=1, END=5, DLT=1)
    COLAPT    (NAM=AVE$TB(I), CSY=CSY, DEL=Y, PTS=QUAD$T_BALLS_MINUS, FRS=I, LST=I)
    COLAPT    (NAM=AVE$TB(I), CSY=CSY, DEL=N, PTS=QUAD$T_BALLS_PLUS, FRS=I, LST=I, TY
    PUTVALS   (OBJ=AVE$TB(I), RDS=(D,E,F), VAL=(0,0,0))
  ENDDO

! ---- get ave of coordinates for balls 1-4 and bld point
  DO          (NAM=I, BGN=1, END=5, DLT=1)
    GETVALS   (OBJ=AVE$TB(I), RDS=(X,Y,Z), REA=(AV$X(I),AV$Y(I),AV$Z(I)))
  ENDDO

  TB$AV_X=(AV$X(1)+AV$X(2)+AV$X(3)+AV$X(4))/4
  TB$AV_Y=(AV$Y(1)+AV$Y(2)+AV$Y(3)+AV$Y(4))/4
  TB$AV_Z=(AV$Z(1)+AV$Z(2)+AV$Z(3)+AV$Z(4))/4
  DFNELE     (NAM=AVE$Y(1234), CSY=CSY, DEL=*)
  PUTVALS    (OBJ=AVE$Y(1234), RDS=(X,Y,Z), VAL=(TB$AV_X,TB$AV_Y,TB$AV_Z))

  CONCAT      (NAM=~T_BALL$AVE, STR=(~CMM$DIR, 'TOOLING BALLS', ' ', ~QID, ' ', AVERAGE, '.TX

! --- output tooling ball data to file
  OPEN        (FIL=~T_BALL$AVE, DEV=D1, STA=NEW, ACC=A)
  OUTPUT      (NAM=(AVE$TB(1),AVE$TB(2),AVE$TB(3),AVE$TB(4),AVE$TB(5),AVE$Y(1234)), DEV=
  CLOSE       (DEV=D1)

GOTO         (LAB=100)

700:STOP
!EDTTXT ACT$OUT_TXT
!CNVTXT ACT$OUT_TXT
!EDTQUE $SCOTT

!EDTTXT EVA$OUT_HDR
!EDTTXT EVA$OUT_TXT
!CNVTXT EVA$OUT_TXT

```