The M.I.T. Lunar and Planetary Ephemeris

by

P. J. Morgan and R. W. King
Department of Earth and Planetary Sciences
Massachusetts Institute of Technology
Cambridge, MA 02139 USA

1 Introduction

This document describes a software READ package for the export version of an M.I.T. planetary ephemeris tape. The tape itself contains as time-series the heliocentric orbits of the nine major planets, the geocentric orbit of the moon, the earth's nutation, and the moon's physical libration. Lunar and planetary coordinates are cartesian and are presently referenced to the 1950.0 mean equator and equinox. The tabular interval is one-half day for the moon, one day for Mercury, and two days for each of the remaining planets. Conventional nutation and libration angles are tabulated at half- day intervals.

The ephemeris tape is available at 800, 1600, or 6250 BPI on 9-track tape using the IBM EBCDIC character set. Records are of fixed length, 80 characters, and blocked using a factor of 100. There are two principal record groups on the tape. The first group of 61 records contains the 128 character header and 59 other records specifying the parameters used to calculate the ephemerides. The second group of record contains the positions and velocities of the bodies, and the nutation and libration angles, in block of 20 days.

Appendices A through L contain documentation and listings of the subroutines required to interpolate for any epoch the heliocentric coordinates (position, velocity, and acceleration) of the planets and the solar system barycenter, the geocentric coordinates of the moon, the two angles of the earth's nutation, and the three angles of the moon's physical libration. The subroutines are written in IBM FORTRAN, but are easily translatable for use on other machines. Appendix M lists an example of the first three data blocks from an ephemeris tape, and Appendix N gives sample output obtained using the READ package with this example.

2 APPENDIX A - SUBROUTINE PRTCRD

```
      SUBROUTINE PRTCRD (IFILE,JD,FRACT,LER,KISS,XPERT,XCNT,DPSI,
     1DEPS,LIBRAT)
C
C     MAIN ROUTINE TO COMPUTE POSITION,VELOCITY AND ACCELERATION OF
C     PLANETS AND SOLAR SYSTEM BARYCENTER RELATIVE TO CENTER
C     OF MASS OF SUN, MOON RELATIVE TO EARTH, THE EARTH'S NUTATION
C     AND THE MOON'S PHYSICAL LIBRATION.
C
C     LOGIC FOLLOWS THAT USED IN PEP AS WRITTEN BY M.ASH.
C     NO COMMON IS USED. THIS MAKES FOR EXTENSIVE CALLING SEQUENCES.
C
C     THE FIXED FORMAT MIT PLANETARY EPHEMERIS CONSISTS OF 3 TYPES OF
C     RECORD BLOCKS. BLOCK ONE HAS TWO RECORDS OF CHARACTERS
C     DESCRIBING THE EPHEMERIS IN ENGLISH OR AUSTRALIAN. BLOCK TWO
C     HAS 48 RECORDS DESCRIBING FIXED PARAMETERS & CONSTANTS ASSOCIATED
C     WITH THE EPHEMERIS. FOLLOWING BLOCKS 1 AND 2 THERE ARE N TYPE 3
C     BLOCKS EACH CONSISTING OF 261 RECORDS TABULATING FOR A 20-DAY
C     INTERVAL BODY POSITIONS AND BODY VELOCITIES, FOLLOWED BY
C     NUTATION AND LUNAR PHYSICAL LIBRATION ANGLES.  THE FORMAT OF
C     THE TAPE IS DESCRIBED IN MORE DETAIL IN THE DOCUMENTATION OF
C     THE NBODY SOURCE PROGRAM (NOT INCLUDED HERE).
C
C     NOTE (1) THIS ROUTINE SHOULD BE CALLED AS FOLLOWS:
C              THE FIRST CALL SHOULD BE WITH JD=0. THIS LOADS
C              FROM THE EPHEMERIS TAPE PARAMETERS AND
C              CONSTANTS FROM BLOCKS 1 & 2. IT ALSO LOADS SOME
C              ARRAYS WITH INITIAL VALUES AND THE LIMITS OF
C              APPLICABILITY OF THE DATA SET. ALL SUBSEQUENT CALLS
C              SHOULD BE MADE WITH JD .GT. 0.
C
C          (2) THIS CODE IS SET UP TO PERFORM NUMERICAL DIFFERENTIATION
C              TO OBTAIN FIRST AND SECOND DERIVATIVES WHEN REQUIRED.
C              CODE IS NOT IMPLEMENTED TO MAKE USE OF VELOCITY
C              INFORMATION ON THE NBODY TAPE. THESE DIFFERENCES ARE VERY
C              SMALL. THE FIRST DERIVATIVE FOR THE NUTATION TERMS IS
C              ALWAYS COMPUTED IF NUTATIONS ARE REQUESTED. THE FIRST
C              DERIVATIVE OF THE LIBRATIONS IS NEVER COMPUTED ALTHOUGH
C              THE ARRAYS AND LOGIC ARE CAPABLE OF SUPPORTING THIS
C              FEATURE.
C
C          (3) NOTE CAREFULLY THE DEFINITIONS OF JD AND FRACT, WHICH
C              SPECIFY THE EPOCH FOR WHICH EPHEMERIS INFORMATION
C              IS REQUIRED. FRACT IN PEP IS RECKONED FROM THE
C              MIDNIGHT PRECEDING THE BEGINNING (AT NOON) OF THE
C              JULIAN DAY JD.  THUS, PEP'S JD+FRACT IS ONE HALF DAY
C              GREATER THAN THE CONVENTIONAL JULIAN DATE. THE EPOCH
C              SHOULD BE SPECIFIED IN COORDINATE TIME (CT).
C
C          (4) UNITS: POSITIONS- AU
C                     VELOCITIES- AU/DAY
```

```
C                        ACCELERATIONS- AU/DAY**2
C                        ANGLES- RADIANS
C                        ANGULAR RATES- RADIANS/DAY
C
C
C     INPUT PARAMETERS ARE:
C
C  IFILE        :I*4  LOGICAL UNIT NUMBER OF EPHEMERIS TAPE/FILE
C  JD           :I*4  JULIAN DAY BEGINNING AT NOON OF DAY ON WHICH
C                     VECTOR IS REQUIRED.
C
C  FRACT        :R*8  FRACTION OF DAY FROM MIDNIGHT FOR WHICH
C                     VECTOR IS REQUIRED.
C  LER          :I*4  LER=0  POSITIONS ONLY ARE REQUIRED.
C                     LER=1  POSITIONS AND VELOCITIES ARE REQUIRED.
C                     LER=2  POSITIONS VELOCITIES AND ACCELERATIONS
C                     ARE REQUIRED.
C  KISS(13)     :I*4  KISS(I) .LT. 0  DO NOT COMPUTE VECTORS FOR BODY I.
C                     KISS(I) .GT. 0  COMPUTE VECTORS FOR THIS BODY.
C                     SPECIAL MEANING IS ATTACHED TO KISS(11) - KISS(13)
C                     WHICH OPERATE ANALOGOUSLY TO KISS(1) THROUGH KISS(10):
C                     KISS(11) IS FOR NUTATIONS.
C                     KISS(12) IS FOR LUNAR LIBRATIONS.
C                     KISS(13) IS FOR SOLAR SYSTEM BARYCENTER.
C
C     OUTPUT PARAMETERS ARE:
C
C  XPERT(9,10) :R*8  POSITION(1-3),VELOCITY(4-6),&  ACCELERATION(7-9) OF
C                    PLANETARY BODIES: MERCURY=1, VENUS=2, EARTH/MOON .
C                    BARYCENTER=3, MARS=4, JUPITER=5, SATURN=6,
C                    URANUS=8, PLUTO=9, MOON=10.  ALL POSITIONS  ARE
C                    HELIOCENTRIC EXCEPT FOR THE MOON, WHICH IS
C                    GEOCENTRIC.
C  XCNT(9)      :R*8  SOLAR SYSTEM BARYCENTER RELATIVE TO SUN
C  DPSI(2)      :R*4  NUTATION IN LONGITUDE AND FIRST DERIVATIVE
C  DEPS(2)      :R*4  NUTATION IN OBLIQUITY AND FIRST DERIVATIVE
C  LIBRAT(2,3) :R*4  LUNAR PHYSICAL LIBRATIONS: TAU, THETA (=I+RHO),
C                    SIGMA, AND FIRST DERIVATIVE.
C
C     REQUIRED SUBROUTINES ARE:
C
C  READF   TO READ BLOCK ONE OF FIXED FORMAT NBODY TAPE
C  READG   TO READ BLOCK TWO OF FIXED FORMAT NBODY TAPE.
C  READH   TO READ FIRST RECORD OF A TYPE 3 BLOCK OF RECORDS
C          FROM FIXED FORMAT NBODY TAPE (EPHEMERIS TAPE)
C  READI   TO READ RECORDS 2 THROUGH 261 OF A TYPE 3 BLOCK OF RECORDS ON
C          FIXED FORMAT NBODY TAPE.
C  LVECTR COMPUTES DO LOOP LIMITS FOR SHIFTING STORAGE AND READING
C          EPHEMERIS DATA INTO ARRAYS FROM EPHEMERIS TAPE.
C  SKIPF   TO SKIP FORWARDS OVER A BLOCK OF TYPE 3 RECORDS.
C  SKIPB   TO SKIP BACKWARDS OVER A BLOCK OF TYPE 3 RECORDS.
C  YPRTCD COMPUTES EVERETT INTERPOLATION Y-VECTORS USING EIGHTH
C          DIFFERENCES.
C  YNUTLB COMPUTES EVERETT INTERPOLATION Y-VECTORS FOR NUTATION AND
```

```
C         LIBRATION USING FOURTH DIFFERENCES.
C  PRTERP PERFORMS EVERETT EIGHTH DIFFERENCE INTERPOLATION FOR THE
C         MOON AND PLANETS.
C  NUTERP PERFORMS EVERETT FOURTH DIFFERENCE INTERPOLATION FOR
C         NUTATIONS AND LIBRATIONS.
C
C     PETER MORGAN SEPT 81
C
C     DECLARATIONS:
C
      INTEGER *4 NBDY,NPL(10),NCP(10),INTB(10),INTBD,KBDY(40)
C
      INTEGER *4 IFILE,JD,LER,JDBD1,JDBD2,JDBD0(10),NMOON,JVLBD,ITRT,NPG
     1,LV(7),JDBD(3),IVEL(3),MVEL(3),ITEMP1,ITEMP2,ITEMP3,IVL,MVL,NTAB,
     2KISS(13)
C
      REAL *4 NAME(6,12),EPSBD,PSID(120),EPSD(120),LIBRT(120,3)
     1,DPSI(2),DEPS(2),LIBRAT(2,3)
C
      REAL *8 FRACT,XPERT(9,10),XCNT(9),
     1BETA(6,10),MASS1(10),FRCT(3),MERC(6,30),BODY(6,15,
     28),MON(6,120),DTEMP1,P(4),DMOON(2),DMERC(2),DBODY(2),
     3YMOON(5,9,2),YMERC(5,9,2),YBODY(5,9,2,8),YEPS(3,2,2),YPSI(3,2,2),
     4YLIB(3,2,2,3),FRACT1,DIR,MEQINC,MASCNT
C
C     END DECLARATIONS SECTION
C
      IF(JD.NE.0) GO TO 80
      REWIND IFILE
C     READ HEADER RECORDS (BLOCK 1) OF EPHEMERIS TAPE
      CALL READF (IFILE)
C     READ TYPE 2 DATA BLOCK. THIS INFO CONSTANT FOR NBODY TAPE.
      CALL READG (IFILE,NBDY,NPL,NCP,INTB,JDBD1,JDBD2,JDBD0,BETA,MEQINC,
     1NAME,NMOON,NBDY1,INTBD,JVLBD,MASS1)
C     TEST FOR TAPE TO BE AU UNITS FOR MOON
      IF(NMOON.EQ.0) GO TO 20
      WRITE(6,10) NMOON
   10 FORMAT(/,1X,'THIS ROUTINE SETUP FOR DISTANCES TO MOON IN AU ONLY:
     1VALUE ON NBODY TAPE IS',I3,/,' PROGRAM TERMINATING-FATAL ERROR')
      STOP
C
C     SUM UP SOLAR SYSTEM MASSES FOR COMPUTATION OF SOLAR SYSTEM
C     BARYCENTER. NOTE MASS OF SUN=1.D0 MASS UNITS. COMPUTATIONAL
C     ERRORS WILL RESULT IF ALL BODIES NOT PRESENT.
   20 MASCNT=1.D0
      ITRIG=1
      ITRIG2=0
      DO 30 I=1,10
   30 ITRIG2=ITRIG2+KISS(I)
      IF(ITRIG2 .NE. 10 .OR. NBDY .NE. 10) GO TO 50
      DO 40 I=1,NBDY
   40 MASCNT=MASCNT+MASS1(I)
      GO TO 60
   50 ITRIG=0
```

```
C
C     FILL STORAGE WITH FIRST THREE BLOCKS OF TYPE 3 RECORD BLOCKS.
C
   60 NN=1
      DO 70 L=NN,3
      CALL LVECTR(L,LV)
      CALL READH(IFILE,L,JDBD,FRCT,IVL,MVL)
      CALL READI(IFILE,LV,IVL,MVL,NBDY,MERC,BODY,MON,PSID,EPSD,LIBRT)
   70 CONTINUE
      IF(JD.EQ.0) RETURN
   80 CONTINUE
C
C     FIRST TEST THAT THE REQUESTED EPOCH IS ON THE NBODY TAPE.
      IF(JD.GE.JDBD1+20 .AND. JD.LT.JDBD2-20) GO TO 100
      WRITE(6,90) JD,JDBD1,JDBD2
   90 FORMAT(/,1X,'PROGRAM TERMINATING DUE TO OUT OF RANGE JULIAN DATES'
     1,/,5X,'REQUESTED EVALUATION JD IS',I10,/,5X,'STARTING JD OF TAPE
     2IS',I10,' END JD OF TAPE IS ',I10)
      STOP
C
C     ORDINARILY THE EPHEMERIS TAPE IS INCREASING IN TIME, THE FORWARD
C     DIRECTION. INTEGRATION AND HENCE TAPE CAN BE  BACKWARDS IN
C     TIME. PARAMETER IDIR=1 FOR FORWARD-NORMAL MODE OR -1 FOR
C     BACKWARD  MODE. THIS ROUTINE WORKS FOR BOTH DIRECTIONS DEPENDING
C     ON THE SETTING OF PARAMETER IDIR WHICH IS AUTOMATIC FROM
C     JDBD1 AND JDBD2.
C
  100 IDIR=ISIGN(1,JDBD2-JDBD1)
      DIR=DFLOAT(IDIR)
C
C     RESET ITRIG USED TO TEST SOLAR SYSTEM BARYCENTER COMPUTATION
C     SINCE THIS COMPUTATION DEPENDENT ON KISS ARRAY.
      ITRIG=0
C
C     A BODY'S STATE IS GIVEN BY A VECTOR OF COORDINATES; ONE VECTOR OF
C     3 COORDINATES FOR POSITION, VELOCITY, AND ACCELERATION. INPUT
C     PARAMETER LER SPECIFIES THIS STATE.
C     PARAMETERS LIMVEL,LIMNUT & LIMLIB CONTROL THE 1ST & 2ND
C     DERIVATIVES OF A BODY POSITION & 1ST DERIVATIVE OF NUTATION AND
C     LIBRATION.
      LIMVEL=3*(LER+1)
      LIMNUT=IABS(KISS(11))
      IF(KISS(11).GT.0) LIMNUT=LIMNUT+1
      LIMLIB=IABS(KISS(12))
      IF(KISS(12).GT.0)LIMLIB=LIMLIB+1
C
C     THEORETICALLY IT IS POSSIBLE TO GENERATE PEP TAPES WITH TABULATION
C     INTERVALS OTHER THAN MERCURY=2 DAYS, VENUS THROUGH PLUTO=4 DAYS AND
C     EARTH'S MOON =0.5 DAY. SUCH A TAPE WILL REQUIRE STORAGE ALLOCATION
C     TO BE MODIFIED AND/OR TAPE FORMATS. THIS SUITE OF ROUTINES ASSUMES
C     THE ABOVE STANDARD CONDITIONS. NEVERTHELESS FOR STRANGE AUSTRALIAN
C     AMERICAN REASONS IT IS NECESSARY TO DEFINE THESE VALUES
C     EXPLICITLY. THE PRINCIPAL REASON IS TO COPE WITH FORWARD AND
C     BACKWARD TAPES.
```

```
C
C     DMOON(1), DMERC(1) & DBODY(1) CONTAIN TABULAR VALUE WHILE
C     DMOON(2), DMERC(2) & DBODY(2) CONTAIN SQUARE OF TABULAR INTERVAL.
C     LATTER REQUIRED BY EVERETT INTERPOLATOR ROUTINE PRTERP.
C
      DMOON(1)=0.5D0*DIR
      DMOON(2)=DMOON(1)**2
      DMERC(1)=2.0D0*DIR
      DMERC(2)=DMERC(1)**2
      DBODY(1)=4.0D0*DIR
      DBODY(2)=DBODY(1)**2
      NN=IDIR*(JD-JDBD(2))
C
C     THREE WAY BRANCH: NN .LT. 0 CORRECT RECORDS ARE BEHIND ON TAPE.
C                       NN=0 CORRECTLY POSITIONED
C                       NN .GT. 0 CORRECT RECORDS ARE AHEAD ON TAPE.
C
      IF(NN)250,300,110
C
C     CORRECT RECORDS ARE AHEAD ON TAPE.
C     HOW MANY RECORDS AHEAD ON TAPE?
C
  110 N=NN/20
      IF(N.EQ.0) GO TO 300
      MM=N-3
      IF(MM)140,270,120
  120 DO 130 I=1,MM
      CALL SKIPF(IFILE,ITEMP1,DTEMP1,ITEMP2,ITEMP3)
  130 CONTINUE
      GO TO 270
C
C     CORRECT MIDDLE RECORD IS NO MORE THAN 2 RECORD BLOCKS AHEAD ON
C     TAPE
C
C     ALL STORAGE MUST BE SHIFTED
C
C     THE L VECTOR RESULTS FROM THE DIFFERENT TABULATION INTERVALS
C     FOR THE BODIES. L1 & L2 ARE FOR MERCURY WHICH HAS 10 COLUMNS
C     L3 & L4 ARE FOR VENUS THROUGH PLUTO WHICH HAVE 5 COLUMNS WHILE
C     L5 & L6 ARE FOR EARTH'S MOON WHICH HAS 40 COLUMNS.
  140 L1 = 1
      L2 =10
      L3 = 1
      L4 = 5
      L5 = 1
      L6 =40
      M1 =N*10
      M2 =N*5
      M3 =N*40
      NN =2
      JDBD(1) =JDBD(N+1)
      FRCT(1)=FRCT(N+1)
      IVEL(1)=IVEL(N+1)
      MVEL(1)=MVEL(N+1)
```

```
      IVL    =IVEL(1)
      MVL    =MVEL(1)
C     START SHIFT LOOP FOR MERCURY THROUGH PLUTO SINCE FIRST VARIABLE IS
C     COMMON
  150 DO 190 I=1,IVL
C     SHIFT STORAGE FOR MERCURY
      DO 160 J=L1,L2
      MM  =J+M1
      MERC(I,J)  =MERC(I,MM)
  160 CONTINUE
      DO 180 J=L3,L4
      MM  =J+M2
      DO 170 K=1,8
      BODY(I,J,K)=BODY(I,MM,K)
  170 CONTINUE
  180 CONTINUE
  190 CONTINUE
C     SHIFT STORAGE FOR EARTH'S MOON
      DO 210 J=L5,L6
      MM  =J+M3
      DO 200 I=1,MVL
      MON(I,J)  =MON(I,MM)
  200 CONTINUE
  210 CONTINUE
C     SHIFT STORAGE FOR NUTATION COEFFICIENTS
C     THIS IS ALSO THE OUTER LOOP FOR LIBRATION COEFFICIENTS
      DO 230 J=L5,L6
      MM  =J+M3
      PSID(J) =PSID(MM)
      EPSD(J) =EPSD(MM)
C     SHIFT STORAGE (INNER LOOP) FOR LIBRATION COEFFICIENTS.
      DO 220 I=1,3
      LIBRT (J,I) = LIBRT (MM,I)
  220 CONTINUE
  230 CONTINUE
      GO TO(240,280),N
C     SHIFT POINTERS TO NEW CENTER RECORD FOR SUCCESSFUL INTERPOLATION
C
  240 JDBD(2) =JDBD(3)
      FRCT(2)=FRCT(3)
      IVEL(2)=IVEL(3)
      MVEL(2)=MVEL(3)
      IVL    =IVEL(2)
      MVL    =MVEL(2)
      N  = 2
      NN = 3
      L1 =11
      L2 =20
      L3 = 6
      L4 =10
      L5 =41
      L6 =80
      GO TO 150
C
```

```
C           CORRECT RECORDS ARE BEHIND ON THE TAPE
  250 NN =4-(NN+1)/20
      DO 260 I=1,NN
      CALL SKIPB(IFILE,ITEMP1,DTEMP1,ITEMP2,ITEMP3)
  260 CONTINUE
C
C           READ PERTURBING PLANET TAPE
  270 NN=1
  280 DO 290 L=NN,3
      CALL LVECTR(L,LV)
      CALL READH(IFILE,L,JDBD,FRCT,IVL,MVL)
      CALL READI(IFILE,LV,IVL,MVL,NBDY,MERC,BODY,MON,PSID,EPSD,LIBRT)
      IVEL(L)=IVL
      MVEL(L)=MVL
  290 CONTINUE
      NTMOON=-99
      NTBODY=-99
      NTMERC=-99
C     SETUP FOR MOON,NUTATION & LIBRATION INTERPOLATION NOW
      NN=(JD-JDBD(2))*IDIR
  300 FRACT1=FRACT*DIR
C     REMEMBER KISS: KISS(I)=-1 DON'T COMPUTE INTERPOLATED POSITION.
C                    KISS(I)= 1 COMPUTE INTERPOLATED POSITION.
C                    KISS(11)   FOLLOWS SAME RULES FOR NUTATION.
C                    KISS(12)   FOLLOWS SAME RULES FOR LIBRATION.
C                    KISS(13)   FOLLOWS SAME RULES FOR SOLAR SYSTEM
C                               BARYCENTER.
C     HENCE KISS(10) .GT. 0 MEANS COMPUTE MOON VECTOR.
C           KISS(11) .GT. 0 MEANS COMPUTE NUTATION.
C           KISS(12) .LT. 0 DO NOT COMPUTE LIBRATION.
C           KISS(13) .GT. 0 MEANS COMPUTE SOLAR SYSTEM BARYCENTER
C                           NOTE KISS(1) THROUGH KISS(10) MUST BE
C                           GREATER THAN ZERO FOR THIS CONDITION.
C
      IF(KISS(10).GT.0) GO TO 310
      IF(KISS(11).GT.0) GO TO 310
      IF(KISS(12).LE.0) GO TO 530
  310 NTAB=2*NN+1
      P(1) =FRACT-0.5D0
      IF(P(1) .LT. 0.0D0) GO TO 320
      NTAB =NTAB+IDIR
      GO TO 330
  320 P(1) =FRACT
  330 P(1) =P(1)/DMOON(1)
      N    =3
C
C           COMPUTE P VECTOR AND TABULAR INDICES
  340 IF(P(1) .GE. 0.0D0) GO TO 350
      P(1) =P(1)+1.0D0
      NTAB =NTAB-1
  350 MTAB =NTAB+1
      P(3) =1.0D0-P(1)
      P(2) =P(1)**2
      P(4)  =P(3)**2
```

```
      GO TO (660,540,360),N
C
C           REST OF SETUP FOR MOON, NUTATION, LIBRATION INTERPOLATION
  360 N    =10
      IF (NTAB.EQ.NTMOON) GO TO 490
      IF (MTAB.EQ.NTMOON) GO TO 380
      IF (NTAB.EQ.(NTMOON+1)) GO TO 370
      K1 = NTAB + 36
      K2 = 2
      K3 = 1
      GO TO 450
  370 K1 = MTAB + 36
      K3 = 2
      GO TO 390
  380 K1 = NTAB  + 36
      K3 = 1
  390 K2 = 1
      K4 = 3 - K3
C
C           SHIFT Y-VECTORS FOR MOON, NUTATION LIBRATION INTERPOLATION
       IF(KISS(10).LT.0) GO TO 410
      LIMVL =LIMVEL
      DO 400 I=1,5
      DO 400 J=1,LIMVL
  400 YMOON(I,J,K4) = YMOON(I,J,K3)
  410 IF(KISS(11).LE.0) GO TO 430
      DO 420 I=1,3
      DO 420 J=1,LIMNUT
      YEPS(I,J,K4)=YEPS(I,J,K3)
      YPSI(I,J,K4)=YPSI(I,J,K3)
  420 CONTINUE
  430 IF(KISS(12) .LE.0) GO TO 450
      DO 440 I=1,3
      DO 440 J=1,LIMLIB
      DO 440 K=1,3
      YLIB(I,J,K4,K)=YLIB(I,J,K3,K)
  440 CONTINUE
C
C           CALCULATE Y-VECTORS FOR MOON NUTATION LIBRATION INTERP.
  450 NTMOON = NTAB
      IF(KISS(10).LT.0) GO TO 460
      CALL YPRTCD(LER,MON(1,K1),YMOON(1,1,K3),K2)
  460 IF(KISS(11) .LE. 0) GO TO 470
      CALL YNUTLB (PSID(K1),YPSI(1,1,K3),K2,KISS(11))
      CALL YNUTLB (EPSD(K1),YEPS(1,1,K3),K2,KISS(11))
  470 IF(KISS(12) .LE.0) GO TO 490
      DO 480 I=1,3
      CALL YNUTLB(LIBRT(K1,I),YLIB(1,1,K3,I),K2,KISS(12))
  480 CONTINUE
C
C           PERFORM MOON, NUTATION, LIBRATION INTERPOLATIONS
  490  IF(KISS(10) .LT. 0) GO TO 500
       CALL PRTERP(P,LER,N,XPERT,DMOON,YMOON)
  500 IF(KISS(11) .LE. 0) GO TO 510
```

```
      CALL NUTERP(P,YPSI,DPSI,DMOON,KISS(11))
      CALL NUTERP(P,YEPS,DEPS,DMOON,KISS(11))
  510 IF(KISS(12) .LE. 0) GO TO 530
      DO 520 I=1,3
      CALL NUTERP(P,YLIB(1,1,1,I),LIBRAT(1,I),DMOON,0)
  520 CONTINUE
C
C     TRANSFORM THE LIBRATION ANGLES FROM THE TAU, RHO, I*(TAU-SIGMA)
C     SYSTEM TO THE TAU, THETA, SIGMA SYSTEM.
C     NOTE THAT FIRST DERIVATIVES ARE NOT COMPUTED AND ARE THEREFORE
C     ZERO.
      LIBRAT(1,2)=LIBRAT(1,2)+MEQINC
      LIBRAT(1,3)=LIBRAT(1,1)-LIBRAT(1,3)/MEQINC
C
C             SET UP VENUS THROUGH PLUTO INTERPOLATION
  530 NTAB =NN/4
      N    =NN-4*NTAB
      P(1) =N
      P(1) =(P(1)+FRACT1)/DBODY(1)
      NTAB =NTAB+1
      N    =2
      GO TO 340
  540 NGO = 3
      IF (NTAB.EQ.NTBODY) GO TO 600
      IF (MTAB.EQ.NTBODY) GO TO 560
      IF (NTAB.EQ.(NTBODY+1)) GO TO 550
      NGO = 1
      K1 = NTAB + 1
      K2 = 2
      K3 = 1
      GO TO 580
  550 K1 = MTAB + 1
      K3 = 2
      GO TO 570
  560 K1 = NTAB + 1
      K3 = 1
  570 K2 = 1
      NGO = 2
  580 NTBODY = NTAB
      GO TO 600
C
C
C             PERFORM VENUS THROUGH PLUTO INTERPOLATIONS
  590 N    =N+1
  600 IF(KISS(N) .LT. 0) GO TO 650
      K5 = N-1
      GO TO (630,610,640),NGO
  610 K4 = 3 - K3
      LIMVL =LIMVEL
      DO 620 I=1,5
      DO 620 J=1,LIMVL
  620 YBODY(I,J,K4,K5) = YBODY(I,J,K3,K5)
  630 CALL YPRTCD(LER,BODY(1,K1,K5),YBODY(1,1,K3,K5),K2)
  640 CALL PRTERP (P,LER,N,XPERT,DBODY,YBODY(1,1,1,K5))
```

```
  650 IF(N .LT. 9) GO TO 590
C
C           SET UP MERCURY INTERPOLATION
      IF(KISS(1) .LT. 0) GO TO 730
      NTAB =NN/2
      N    =NN-2*NTAB
      P(1) =N
      P(1) =(P(1)+FRACT1)/DMERC(1)
      NTAB =NTAB+1
      N    =1
      GO TO 340
C
C           PERFORM MERCURY INTERPOLATION
  660 IF (NTAB.EQ.NTMERC) GO TO 720
      IF (MTAB.EQ.NTMERC) GO TO 680
      IF (NTAB.EQ.(NTMERC+1)) GO TO 670
      K1 = NTAB+ 6
      K2 = 2
      K3 = 1
      GO TO 710
  670 K1 = MTAB + 6
      K3 = 2
      GO TO 690
  680 K1 = NTAB + 6
      K3 = 1
  690 K2 = 1
      K4 = 3 - K3
      LIMVL =LIMVEL
      DO 700 I=1,5
      DO 700 J=1,LIMVL
  700 YMERC(I,J,K4) = YMERC(I,J,K3)
  710 CALL YPRTCD(LER,MERC(1,K1),YMERC(1,1,K3),K2)
      NTMERC = NTAB
  720 CALL PRTERP(P,LER,N,XPERT,DMERC,YMERC)
C
C     COMPUTE SOLAR SYSTEM BARYCENTER. STOP PROGRAM IF EITHER
C     SUM OF KISS ARRAY 1 THROUGH 10 .NE. 10 OR ALL BODIES NOT ON TAPE.
C
  730 IF(KISS(13) .NE.1) GO TO 800
      ITRIG2=0
      DO 740 I=1,10
  740 ITRIG2=ITRIG2+KISS(I)
      IF (ITRIG2 .EQ. 10) ITRIG=1
      IF(ITRIG .EQ. 1 ) GO TO 760
      WRITE(6,750) NBDY,ITRIG
  750 FORMAT(/,1X,'FATAL ERROR OCCURS WHEN SOLAR SYSTEM BARYCENTER IS',
     1' ABOUT TO BE COMPUTED.',/,1X,'# OF BODIES ON TAPE WAS',I3,
     2/,1X,'SUM OF KISS ARRAY THROUGH 10 WAS',I3)
      STOP
C
C     FIRST ZERO OUT XCNT ARRAY SO THAT ACCUMULATION CAN BE DONE.
  760 DO 770 I=1,LIMVEL
  770 XCNT(I)=0.0D0
C
```

```
C     OUTER LOOP IS FOR POSITION, VELOCITY AND ACCELERATION. INNER
C     LOOP IS THE INDIVIDUAL BODY CONTRIBUTION SUM.
      DO 790 I=1,LIMVEL
      DO 780 J=1,9
      XCNT(I)=XCNT(I)+XPERT(I,J)*MASS1(J)
  780 CONTINUE
      XCNT(I)=XCNT(I)/MASCNT
  790 CONTINUE
C
  800 RETURN
      END
```

3 APPENDIX B - SUBROUTINE LVECTR

```
      SUBROUTINE LVECTR(L,LV)
C
C     THIS SUBROUTINE COMPUTES FOR A GIVEN L A VECTOR OF DO LOOP LIMITS
C     USED IN READING BODY DATA INTO STORAGE. THE VECTOR IS LV AND IS OF
C     SIZE 7
C
C     PETER MORGAN SEPT 81
C
      INTEGER *4 LV(7)
      LV(1)=L
      LV(3)=L*10
      LV(2)=LV(3)-9
      LV(5)=L*5
      LV(4)=LV(5)-4
      LV(7)=L*40
      LV(6)=LV(7)-39
      RETURN
      END
```

)

4 APPENDIX C - SUBROUTINE SKIPF

```
      SUBROUTINE SKIPF(IFILE,JDBD,FRCT,IVL,MVL)
C
C     SUBROUTINE TO FORWARD SKIP BLOCKS OF RECORDS IN THE 80 COLUMN
C     EPHEMERIS TAPE. IFILE IS THE LOGICAL UNIT NUMBER OF THE TAPE/FILE.
C     A RECORD BLOCK IS 261 PHYSICAL TYPE3 3 RECORDS.
C     POSITION OF TAPE OR FILE IS GIVEN BY RETURN PARAMETERS,JDBD,FRCT,
C     IVL,AND MVL
C  IFILE        :I*4  LOGICAL UNIT NUMBER OF EPHEMERIS FILE
C  JDBD         :I*4  JULIAN DATE AT MIDNIGHT OF FIRST RECORD OF BLOCK
C                     OVER WHICH SKIPPING WILL OCCUR.
C  FRCT         :R*8  FRACTION OF DAY FROM MIDNIGHT OF FIRST RECORD OF
C                     BLOCK
C  IVL          :I*4  NUMBER OF COORDINATES FOR PLANETS (6 FOR POSITION
C                     & RATE)
C  MVL          :I*4  NUMBER OF COORDINATES FOR MOON (6 FOR POSITION
C                     & RATE)
C
C     PETER MORGAN SEPT 81
C
      REAL *8 FRCT
      DIMENSION JUNK(20)
      READ(IFILE,30) JDBD,FRCT,IVL,MVL
      DO 10 I=1,260
   10 READ(IFILE,20) JUNK
   20 FORMAT(20A4)
   30 FORMAT(I10,D25.18,2I5)
      RETURN
      END
```

5 APPENDIX D - SUBROUTINE SKIPB

```
      SUBROUTINE SKIPB(IFILE,JDBD,FRCT,IVL,MVL)
C
C     SUBROUTINE TO BACKWARD SKIP BLOCKS OF RECORDS IN THE 80 COLUMN
C     EPHEMERIS TAPE. IFILE IS THE LOGICAL UNIT NUMBER OF THE TAPE/FILE.
C     A RECORD BLOCK IS 261 PHYSICAL TYPE3 3 RECORDS.
C     POSITION OF TAPE OR FILE IS GIVEN BY RETURN PARAMETERS,JDBD,FRCT,
C     IVL,AND MVL
C  IFILE        :I*4  LOGICAL UNIT NUMBER OF EPHEMERIS FILE
C  JDBD         :I*4  JULIAN DATE AT MIDNIGHT OF FIRST RECORD OF BLOCK
C                     OVER WHICH SKIPPING HAS OCCURRED.
C  FRCT         :R*8  FRACTION OF DAY FROM MIDNIGHT OF FIRST RECORD OF
C                     BLOCK
C  IVL          :I*4  NUMBER OF COORDINATES FOR PLANETS (6 FOR POSITION
C                     & RATE)
C  MVL          :I*4  NUMBER OF COORDINATES FOR MOON (6 FOR POSITION
C                     & RATE)
C
C     PETER MORGAN SEPT 81
C
      REAL *8 FRCT
      DO 10 I=1,261
   10 BACKSPACE IFILE
      READ(IFILE,20) JDBD,FRCT,IVL,MVL
   20 FORMAT(I10,D25.18,2I5)
      BACKSPACE IFILE
      RETURN
      END
```

6 APPENDIX E - SUBROUTINE READF

```
      SUBROUTINE READF (IFILE)
C     SUBROUTINE TO REPOSITION EPHEMERIS TAPE TO BEGINNING OF TAPE AND
C     THEN READ TYPE 1 RECORD BLOCK.
C     TYPE ONE DATA RECORDS ARE TWO CHARACTER RECORDS DESCRIBING THE
C     EPHEMERIS TAPE IN ENGLISH OR AUSTRALIAN.
C     INPUT PARAMETERS ONLY IFILE THE LOGICAL UNIT NUMBER AT WHICH THE
C     EPHEMERIS TAPE IS MOUNTED. THERE ARE NO OUTPUT PARAMETERS.
C
C     PETER MORGAN SEPT 1981
C
      INTEGER *4 JUNK1(20),JUNK2(20)
      REWIND IFILE
      READ(IFILE,20,ERR=30) JUNK1
      READ(IFILE,20) JUNK2
      WRITE(6,10) JUNK1,(JUNK2(I),I=1,12)
   10 FORMAT(/,1X,'THE HEADER LABEL ON THE EPHEMERIS TAPE WAS:',
     1/,1X,20A4,/,1X,12A4)
      RETURN
   20 FORMAT(20A4)
   30  WRITE(6,40)
   40 FORMAT(/,1X,'AN ERROR OCCURRED ON OPENING THE EPHEMERIS TAPE'
     1,' PROGRAM ABORTING')
      STOP
      END
```

7 APPENDIX F - SUBROUTINE READG

```
      SUBROUTINE READG(IFILE,NBDY,NPL,NCP,INTB,JDBD1,JDBD2,JDBD0,BETA,
     1MEQINC,NAME,NMOON,NBDY1,INTBD,JVLBD,MASS1)
C     SUBROUTINE TO READ TYPE 2 BLOCKS OF RECORDS FROM FIXED FORMAT
C     EPHEMERIS TAPE.
C  IFILE          :I*4  LOGICAL UNIT NUMBER OF EPHEMERIS FILE.
C  NBDY           :I*4  NUMBER OF BODIES ON THE NBODY TAPE
C  NPL(NBDY)      :I*4  IDENTIFYING NUMBER FOR EACH BODY:NOTE VARIABLE
C                       DIMENSION.
C                       KEY: MERCURY=1, VENUS=2, EARTH-MOON BARYCENTER=3,
C                            MARS=4,JUPITER=5, SATURN=6, URANUS=7,
C                            NEPTUNE=8,PLUTO=9,MOON=10.
C  NCP(NBDY)      :I*4  IDENTIFYING NUMBER FOR CENTRAL BODY FOR EACH
C                       NPL(I): NOTE NORMALLY NCP(1) THROUGH NCP(9)=0 AND NCP
C                       NCP(10)=3.
C  INTB(NBDY)     :I*4  INTERVAL IN DAYS BETWEEN TABULAR POINTS FOR EACH
C                       BODY. NORMALLY MERCURY HAS INTB=2, VENUS THROUGH
C                       PLUTO HAVE INTB=4, MOON HAS INTB=-1
C                       (MEANING 2**(-1):0.5DAY).
C                       NOTE INTB .GT. 0 IS IN DAYS NOT POWERS OF 2.
C  JDBD1          :I*4  STARTING TIME OF TAPE.
C  JDBD2          :I*4  ENDING TIME OF TAPE.
C  JDBD0(NBDY)    :I*4  EPOCH OF INITIAL CONDITIONS OF INTEGRATION FOR
C                       BODY NPL(I).
C  BETA(6,NBDY)   :R*8  SIX INITIAL CONDITION FOR EACH BODY REFERENCED TO
C                       MEAN EQUATOR AND EQUINOX OF 1950.0.
C                       KEY: A(AU), E, INC(DEG), RA OF ASCENDING NODE(DEG)
C                            ARGUMENT OF PERIGEE(DEG), INITIAL MEAN
C                            ANOMALY(DEG).
C  MEQINC         :R*8  INCLINATION OF LUNAR EQUATOR TO ECLIPTIC
C                       FOR USE WITH LIBRATION ANGLES ON TAPE.
C  NAME(6,12  )   :R*4  24 CHARACTER TITLE FOR EACH BODY INTEGRATION
C                       PLUS THE NUTATION AND LIBRATION TABULATIONS. THE
C                       FORMAT IS FOR THE BODY TO BE CENTERED ON THE
C                       FIRST 8 CHARACTERS (++MARS++) FOLLOWED BY THE
C                       INTEGRATION RUN NUMBER AND DATE.
C  NMOON          :I*4  EQUALS 0 FOR PEP-GENERATED TAPES. MEANS LUNAR
C                       EPHEMERIS UNIT OF DISTANCE IS THE ASTRONOMICAL
C                       UNIT (AU).
C  NBDY1          :I*4  NUMBER OF BODIES (EXCLUDING MOON)
C  INTBD          :I*4  INTEGRATION STEP SIZE FOR NBDY INTEGRATION
C                       (DEFAULT IS 2 DAYS).
C  JVLBD          :I*4  SET EQUAL TO ZERO. INDICATES VELOCITY AS WELL AS
C                       POSITION ON TAPE AT EACH TABULAR POINT.
C  MASS1(NBDY)    :R*8  MASS OF BODY IN UNITS OF SOLAR MASS.
C
C       PETER MORGAN SEPT 81
C
      INTEGER *4 NPL(10),NCP(10),INTB(10)
      INTEGER *4 JDBD1,JDBD2,JDBD0(10)
      REAL *4 NAME(8,10)
```

```
      REAL *8 BETA(6,10),MASS1(10),MEQINC
      READ(IFILE,10) NBDY,(NPL(I),I=1,NBDY)
   10 FORMAT(I2,3X,10I3)
      READ(IFILE,20) (NCP(I),I=1,NBDY)
   20 FORMAT(10I3)
      READ(IFILE,20) (INTB(I),I=1,NBDY)
      READ(IFILE,30) JDBD1,JDBD2,(JDBD0(I),I=1,NBDY)
   30 FORMAT(2I10/10I8)
      READ(IFILE,40) ((BETA(I,J),I=1,6),J=1,NBDY)
   40 FORMAT(3D25.18)
      READ(IFILE,50) MEQINC
   50 FORMAT(D25.18)
      READ(IFILE,60) ((NAME(I,J),I=1,6),J=1,   12)
   60 FORMAT(6A4)
      READ(IFILE,70) NMOON,NBDY1,INTBD,JVLBD,(MASS1(I),I=1,NBDY)
   70 FORMAT(4I5,/,(D25.18))
      RETURN
      END
```

8 APPENDIX G - SUBROUTINE READH

```
      SUBROUTINE READH (IFILE,L,JDBD,FRCT,IVL,MVL)
C
C     SUBROUTINE TO READ HEADER RECORD FROM BLOCK OF TYPE 3 RECORDS.
C     A BLOCK OF TYPE 3 RECORDS CONSISTS OF A HEADER RECORD CONTAINING
C     PARAMETERS JDBD,FRCT,IVL AND MVL. THE SUBSEQUENT RECORDS CONTAIN
C     THE COORDINATES THEMSELVES.
C     NOTE: SUBROUTINE READI READS ALL THE SUBSEQUENT RECORDS OF THE
C     TYPE 3 BLOCK. THUS, TO ACCOMPLISH A SEQUENCE OF READS OF THE
C     EPHEMERIS TAPE YOU CALL READH - READI THEN READH - READI ETC.
C  IFILE        :I*4  LOGICAL UNIT NUMBER OF EPHEMERIS FILE
C  JDBD         :I*4  JULIAN DAY OF FIRST RECORD OF BLOCK.
C  FRCT         :R*8  FRACTION OF DAY FROM MIDNIGHT OF FIRST RECORD OF
C                     BLOCK.
C  IVL          :I*4  NUMBER OF COORDINATES FOR PLANETS (6 FOR POSITION
C                     & VELOCITY).
C  MVL          :I*4  NUMBER OF COORDINATES FOR MOON (6 FOR POSITION
C                     & VELOCITY).
C
C     PETER MORGAN SEPT 81
C
      REAL *8 FRCT(3)
      INTEGER *4 JDBD(3)
      READ(IFILE,10) JDBD(L),FRCT(L),IVL,MVL
   10 FORMAT(I10,D25.18,2I5)
      RETURN
      END
```

9 APPENDIX I - SUBROUTINE READI

```
      SUBROUTINE READI (IFILE,LV,IVL,MVL,NBDY,MERC,BODY,MON,PSID,EPSD,
     1LIBRT)
C
C     SUBROUTINE TO READ BODY (PLANET) DATA FROM 80 COLUMN TAPE/FILE.
C     THIS SUBROUTINE READS ONLY 260 TYPE 3 RECORDS OF 261 RECORD BLOCK
C     THE FIRST RECORD OF A TYPE 3 BLOCK IS A HEADER RECORD. IT IS
C     ASSUMED THAT THIS RECORD HAS BEEN READ BY SUBROUTINE READH
C     THE PARAMETERS IVL AND MVL ARE GATHERED FROM THIS HEADER RECORD.
C     THE PARAMETER NBDY GIVING THE NUMBER OF BODIES ON THE TAPE IS READ
C     FROM THE TYPE 2 RECORD BLOCK. FOR THE FIXED RECORD EPHEMERIS
C     TAPE THE NUMBER OF BODIES SHOULD BE 10; THAT IS, ALL PLANETS PLUS
C     THE MOON.
C
C     INPUT PARAMETERS:
C
C  IFILE        :I*4  LOGICAL UNIT NUMBER OF EPHEMERIS FILE
C  LV(7)        :I*4  VECTOR OF DO LOOP LIMITS INDICATING FIRST MIDDLE
C                     OR END RECORDS AND THEIR STORAGE POSITIONS.
C  IVL          :I*4  NUMBER OF COORDINATES FOR PLANETS (6 FOR POSITION
C                     AND RATE)
C  MVL          :I*4  NUMBER OF COORDINATES FOR MOON (6 FOR POSITION
C                     AND RATE)
C  NBDY         :I*4  NUMBER OF BODIES ON EPHEMERIS TAPE
C
C     OUTPUT PARAMETERS
C
C     THE OUTPUT ARRAYS ARE ALL DIMENSIONED TO HOLD 3 TYPE-3 RECORD
C     BLOCKS IN CORE. SINCE EACH BLOCK SPANS 20 DAYS, THE MAXIMUM
C     DIMENSION OF EACH ARRAY IS 3*(20/INTB), WHERE INTB IS THE
C     INTERVAL IN DAYS USED TO TABULATE THE BODY'S POSITION.
C
C  MERC(IVL,30) :R*8  COORDINATES OF MERCURY AT 2 DAY INTERVALS.
C
C  BODY(IVL,15,NBDY-2)
C               :R*8  COORDINATES OF PLANETS EXCEPTING MERCURY AND MOON
C                     TABULATED AT 4 DAY INTERVALS.
C  MON(MVL,120) :R*8  COORDINATES OF MOON TABULATED AT 0.5 DAY INTERVALS
C
C  PSID(120)    :R*4  NUTATION IN LONGITUDE- DELTA(PSI) TABULATED AT
C                     0.5 DAY INTERVALS.
C  EPSD(120)    :R*4  NUTATION IN OBLIQUITY- DELTA(EPSILON) TABULATED
C                     AT 0.5 DAY INTERVALS.
C  LIBRT(120,3) :R*4  LUNAR PHYSICAL LIBRATIONS: TAU, RHO, AND
C                     I*(TAU-SIGMA) TABULATED AT 0.5 DAY INTERVALS.
C                     NOTE REVERSAL OF ORDER OF ARRAY.
C
C      PETER MORGAN SEPT 1981
C
      INTEGER *4 NBDY,NBDY2
      INTEGER *4 LV(7)
```

```
      REAL *4 PSID(120),EPSD(120),LIBRT(120,3)
      REAL *8 MERC(6,30),BODY(6,15,8),MON(6,120)
      NBDY2=NBDY-2
      L=LV(1)
      L1=LV(2)
      L2=LV(3)
      L3=LV(4)
      L4=LV(5)
      L5=LV(6)
      L6=LV(7)
      DO 10 I=L1,L2
   10 READ(IFILE,20) (MERC(J,I),J=1,IVL)
   20 FORMAT(3D25.18)
      DO 30 I=1,NBDY2
   30 READ(IFILE,20) ((BODY(K,J,I),K=1,IVL),J=L3,L4)
      READ(IFILE,20) ((MON(I,J),I=1,MVL),J=L5,L6)
      READ(IFILE,40)(PSID(I),EPSD(I),I=L5,L6)
   40 FORMAT(2E15.8)
      READ(IFILE,50) ((LIBRT(J,I),I=1,3),J=L5,L6)
   50 FORMAT(3E15.8)
      RETURN
      END
```

10 APPENDIX I - SUBROUTINE YPRTCD

```
      SUBROUTINE YPRTCD(LER,X,Y,N)
C
C     SUBROUTINE TO COMPUTE EVERETT INTERPOLATION Y-VECTORS.
C
C     SUBROUTINE ADAPTED FROM YPRTCD WRITTEN BY M.ASH FOR PEP.
C
C     INPUT PARAMETERS ARE:
C
C  LER         :I*4  LER=0  POSITIONS ONLY ARE REQUIRED.
C                    LER=1  POSITIONS AND VELOCITIES ARE REQUIRED.
C                    LER=2  POSITIONS, VELOCITIES & ACCELERATIONS ARE
C                           REQUIRED
C  X(6,84)     :R*8  VECTOR TO BE INTERPOLATED.
C  N           :I*4
C
C     OUTPUT PARAMETERS ARE:
C
C  Y(5,9,41)   :R*8  INTERPOLATION Y-VECTORS.
C
C     PETER MORGAN SEPT 81
C
C     DECLARATIONS.
C
      REAL *8 X(6,84),Y(5,9,41),EVCF(25),FACT(9),F1,F2,F3,F4
      EQUIVALENCE (F1,FACT(1))
C           INTERPOLATION COEFFICIENTS
      DATA EVCF(1),EVCF(2),EVCF(3),EVCF(4),EVCF(5)
     1                     / 1.7873015873015873D 00,
     2-0.4960317460317460D 00, 0.1206349206349206D 00,
     3-0.1984126984126984D-01, 0.1587301587301587D-02/
      DATA EVCF(6),EVCF(7),EVCF(8),EVCF(9),EVCF(10)
     1                     /-0.9359567901234568D 00,
     2 0.6057098765432098D 00,-0.1632716049382716D 00,
     3 0.2779982363315696D-01,-0.2259700176366843D-02/
      DATA EVCF(11),EVCF(12),EVCF(13),EVCF(14),EVCF(15)
     1                     / 0.1582175925925926D 00,
     2-0.1171296296296296D 00, 0.4606481481481481D-01,
     3-0.8796296296296296D-02, 0.7523148148148148D-03/
       DATA EVCF(16),EVCF(17),EVCF(18),EVCF(19),EVCF(20)
     1                     /-0.9755291005291005D-02,
     2 0.7605820105820106D-02,-0.3505291005291005D-02,
     3 0.8597883597883598D-03,-0.8267195767195767D-04/
      DATA EVCF(21),EVCF(22),EVCF(23),EVCF(24),EVCF(25)
     1                     / 0.1929012345679012D-03,
     2-0.1543209876543210D-03, 0.7716049382716048D-04,
     3-0.2204585537918871D-04, 0.2755731922398589D-05/
      DATA FACT(2),FACT(3),FACT(4),FACT(5),FACT(6),FACT(7),FACT(8),
     1 FACT(9)           /3.0D0,5.0D0,7.0D0,9.0D0,2.0D0,4.0D0,6.0D0,
     28.0D0/
C
```

```
C           DETERMINATION OF POSITION Y-VECTORS
      DO 30 K=1,N
      NR =K+4
      DO 20 L=1,3
      F1 =X(L,NR+1)+X(L,NR-1)
      F2 =X(L,NR+2)+X(L,NR-2)
      F3 =X(L,NR+3)+X(L,NR-3)
      F4 =X(L,NR+4)+X(L,NR-4)
      DO 10 I=1,5
      J   =(I-1)*5+1
      Y(I,L,K) =EVCF(J)*X(L,NR)+(EVCF(J+1)*F1+(EVCF(J+2)*F2+(EVCF(J+3)*
     1       F3+EVCF(J+4)*F4)))
   10 CONTINUE
   20 CONTINUE
   30 CONTINUE
C
C           DETERMINATION OF VELOCITY,ACCELERATION Y-VECTORS (IF LER>0)
      IF(LER .LE. 0) GO TO 70
C           FUTURE OPTION TO CALCULATE VELOCITY Y-VECTORS FROM
C           COORDINATES IF THEY EXIST RATHER THAN BY NUMERICAL
C           DIFFERENTIATION
      DO 60 L=1,3
      J  =L+3
      M  =J+3
      DO 50 K=1,N
      Y(1,J,K) =Y(1,L,K)
      DO 40 I=2,5
      Y(I,J,K) =Y(I,L,K)*FACT(I)
      Y(I,M,K) =Y(I,J,K)*FACT(I+4)
   40 CONTINUE
   50 CONTINUE
   60 CONTINUE
   70 RETURN
      END
```

11 APPENDIX J - SUBROUTINE YNUTLB

```
      SUBROUTINE YNUTLB(NLB,YNLB,K2,KISS)
C
C     SUBROUTINE TO COMPUTE Y-VECTORS FOR NUTATION AND LIBRATION.
C
C     SUBROUTINE ADAPTED FROM YNUTLB WRITTEN BY M.ASH FOR PEP
C
C     INPUT PARAMETERS ARE:
C
C  NLB(120)     :R*4  VECTOR TO BE INTERPOLATED.
C  K2           :I*4
C  KISS         :I*4  A SWITCH PARAMETER TO DETERMINE IF
C                     FIRST DIFFERENTIALS ARE TO BE COMPUTED.
C
C     OUTPUT PARAMETERS ARE:
C
C  YNLB(3,2,2) :R*8  INTERPOLATION Y-VECTOR.
C
C     PETER MORGAN SEPT. 1981
C
C     DECLARATIONS.
      REAL *4 NLB(120)
      REAL *8 YNLB(3,2,2),F1,F2,FACT(3)
      REAL*8 EVCF(9) /1.5333333333333333D0,-3.0D-1, 3.3333333333333333D-2,
     1-5.8333333333333333D-1, 3.3333333333333333D-1,-4.1666666666666667D-2,
     2 5.0000000000000000D-2,-3.3333333333333333D-2, 8.3333333333333333D-3/
      DATA FACT /0.,3.,5./
C
      DO 20 K=1,K2
      NR=K+4
         F1 = DBLE(NLB(NR+1)) + DBLE(NLB(NR-1))
         F2 = DBLE(NLB(NR+2)) + DBLE(NLB(NR-2))
      DO 10 I=1,3
           J = 3*I - 2
      YNLB(I,1,K) = EVCF(J)*NLB(NR) + EVCF(J+1)*F1 + EVCF(J+2)*F2
   10 CONTINUE
   20 CONTINUE
C
      IF(KISS .LE. 0) GO TO 50
      DO 40 K=1,K2
      YNLB(1,2,K) =YNLB(1,1,K)
      DO 30 I=2,3
      YNLB(I,2,K) =YNLB(I,1,K)*FACT(I)
   30 CONTINUE
   40 CONTINUE
C
   50 RETURN
      END
```

12 APPENDIX K - SUBROUTINE PRTERP

```
      SUBROUTINE PRTERP(P,LER,N,XPERT,DIST,Y)
C
C     EVERETT EIGHTH DIFFERENCE INTERPOLATION FOR MOON AND PLANETS.
C
C     SUBROUTINE ADAPTED FROM PRTERP WRITTEN BY M.ASH FOR PEP.
C
C     INPUT PARAMETERS ARE:
C
C  P(4)          :R*8  VECTOR OF INTERPOLATION COEFFICIENTS.
C  LER           :I*4  A SWITCH FOR INTERPOLATION OF VECTOR AND FIRST AND
C                      SECOND DERIVATIVES.
C  N             :I*4  THE NUMBER OF THE BODY BEING INTERPOLATED.
C  DIST(2)       :R*8  TABULAR INTERVAL OF EPHEMERIS AND SQUARE.
C  Y(5,9,2)      :R*8  VECTOR TO BE INTERPOLATED.
C
C     OUTPUT PARAMETERS ARE:
C
C  XPERT(9,10)   :R*8  INTERPOLATED POSITION VECTOR FOR BODY N;
C      (I,N)           I=1,3 FOR POSITIONS; I=4,6 FOR VELOCITIES AND
C                      I=6,9 FOR ACCELERATIONS.
C
C     WRITTEN BY P. MORGAN SEPT 1981.
C
C     DECLARATIONS:
C
      REAL *8 P(4),Y(5,9,2),DIST(2),XPERT(9,10)
C
C           DETERMINE POSITION COORDINATES FOR BODY N
      DO 10 I=1,3
      XPERT(I,N) = P(1)*(Y(1,I,2)+P(2)*(Y(2,I,2)+P(2)*(Y(3,I,2)
     1            +P(2)*(Y(4,I,2)+P(2)* Y(5,I,2)))))
     2            +P(3)*(Y(1,I,1)+P(4)*(Y(2,I,1)+P(4)*(Y(3,I,1)
     3            +P(4)*(Y(4,I,1)+P(4)* Y(5,I,1)))))
   10 CONTINUE
C
C           DETERMINE VELOCITY,ACCELERATION FOR BODY N (IF THERE IS REL)
      IF(LER .LE. 0) GO TO 30
      DO 20 I=4,6
      XPERT(I,N) =(Y(1,I,2(2,I,2)+P(2)*(Y(3,I,2)+P(2)*
1        (Y(4,I,2)+P(2)* Y(5,I,2))))
       2            -Y(1,I,1)-P(4)*(Y(2,I,1)+P(4)*(Y(3,I,1)+P(4)*
       3            (Y(4,I,1)+P(4)* Y(5,I,1))))))/DIST(1)
        IF(LER.LE.1) GO TO 20
      J  =I+3
      XPERT(J,N) =(P(1)*(Y(2,J,2)+P(2)*(Y(3,J,2)+P(2)*(Y(4,J,2)
     1            +P(2)* Y(5,J,2))))
     2            +P(3)*(Y(2,J,1)+P(4)*(Y(3,J,1)+P(4)*(Y(4,J,1)
     3            +P(4)* Y(5,J,1))))))/DIST(2)
   20 CONTINUE
   30 CONTINUE
```

```
C
      RETURN
      END
```

13 APPENDIX L - SUBROUTINE NUTERP

```
      SUBROUTINE NUTERP(P,YNLB,NLB,DMOON,KISS)
C
C     FOURTH DIFFERENCE INTERPOLATOR FOR NUTATIONS AND LIBRATIONS.
C
C     SUBROUTINE ADAPTED FROM NUTERP WRITTEN BY M.ASH FOR PEP.
C
C     INPUT PARAMETERS ARE:
C
C  P(4)          :R*8  VECTOR OF INTERPOLATION COEFFICIENTS.
C  YNLB(3,2,2)   :R*8  VECTOR TO BE INTERPOLATED.
C      (I,J,K)
C  DMOON(2)      :R*8  TABULATION INTERVAL AND SQUARE.
C  KISS          :I*4  SWITCH ON WHICH TO TEST IF FIRST DERIVATIVES
C                      ARE REQUIRED.
C
C     OUTPUT PARAMETERS ARE:
C
C  NLB(2)        :R*4  THE INTERPOLATED VALUE & IF REQUESTED FIRST
C                      DERIVATIVE.
C
C     WRITTEN BY PETER MORGAN SEPT 1981
C
C     DECLARATIONS
C
      REAL *8 P(4),YNLB(3,2,2),DMOON(2)
      REAL *4 NLB(2)
      NLB(1)=P(1)*(YNLB(1,1,2)+P(2)*(YNLB(2,1,2)+P(2)*
     1          YNLB(3,1,2))) +
     2          P(3)*(YNLB(1,1,1) + P(4)*(YNLB(2,1,1) + P(4)*
     3          YNLB(3,1,1)))
      IF(KISS .LT. 1) GO TO 10
      NLB(2) = (YNLB(1,2,2) + P(2)*(YNLB(2,2,2) + P(2)*
     1         YNLB(3,2,2)) -YNLB(1,2,1) - P(4)*(YNLB(2,2,1)
     2          + P(4)*YNLB(3,2,1)))/DMOON(1)
C
   10 RETURN
      END
```

14 APPENDIX M - FIRST 3 DATA BLOCKS ON EPHEMERIS TAPE

```
 N-BODY RUN 311  9-BODY INTEGRATION 1961-1980 (305A I.C. INNER, 308 I.C. OUTER)
 8/16/69 PERTURBING PLNT TAPE FROM N-BODY INTEG.
10     1  2  3  4  5  6  7  8  9 10
  0  0  0  0  0  0  0  0  0  3
  2  4  4  4  4  4  4  4  4 -1
   2439121   2439521
 2440001 2440001 2440001 2440001 2440001 2440001 2440001 2440001 2440001 2440001
 0.387098825101369295D+00 0.205614352134493597D+00 0.286033719000404290D+02
 0.108594625301948200D+02 0.669327719394875196D+02 0.908396522026607975D+02
 0.723328078763313995D+00 0.675754064669862198D-02 0.244668542978979495D+02
 0.797815831359027094D+01 0.123785191032075698D+03 0.274406563359057088D+03
 0.999982768892511392D+00 0.167527182067684693D-01 0.234433128994516800D+02
 0.675466880608359670D-03 0.102034242539411800D+03 0.139365833626903399D+03
 0.152360455094046898D+01 0.934578242475824911D-01 0.246929677929284672D+02
 0.334666957437566692D+01 0.332101320790162390D+03 0.898997578206112067D+02
 0.520294628935084180D+01 0.481917803219289896D-01 0.232530224521215381D+02
 0.326103059630796399D+01 0.103739429825117200D+02 0.141185858569590597D+03
 0.952604639118047181D+01 0.546172081500696897D-01 0.225732184087006367D+02
 0.596872764990932492D+01 0.875933767359338091D+02 0.289740456650754766D+03
 0.192744533863705989D+02 0.512420376766123094D-01 0.236714771204321082D+02
 0.184732451381537088D+01 0.168343798200289200D+03 0.699098259064799499D+01
 0.301136980094241373D+02 0.698422305132670972D-02 0.223148273747761685D+02
 0.351992886987806397D+01 0.540750735804183478D+02 0.177594434902879897D+03
 0.397469418884173180D+02 0.252237384045182297D+00 0.236555699451932995D+02
 0.437701311957991592D+02 0.181963168405029698D+03 0.329847527451677195D+03
 0.257151437464250897D-02 0.556154473576186692D-01 0.283968543863414773D+02
 0.331295920987404280D+01 0.226271247222358600D+03 0.154885695684920798D+03
 0.0268587D0
MERCURY RUN 311   8/16/69
 VENUS  RUN 311   8/16/69
 EMBARY RUN 311   8/16/69
  MARS  RUN 311   8/16/69
JUPITER RUN 311   8/16/69
 SATURN RUN 311   8/16/69
 URANUS RUN 311   8/16/69
NEPTUNE RUN 311   8/16/69
 PLUTO  RUN 311   8/16/69
  MOON  RUN 440-  8/ 7/72
 MROTAT  KOZIEL    1967
 EROTAT   WHAR     1981
    0      9     2     0
 0.165784802042999265D-06
 0.244784858587787177D-05
 0.304043689862058415D-05
 0.322715977668054342D-06
 0.954753469287681386D-03
 0.285779606767261058D-03
 0.436109899694723048D-04
 0.519210799584631359D-04
 0.551876379690949216D-06
```

```
 C.0
  2439121 0.0                          6    6
-0.395854899285230988D+00-0.727435820113880721D-01 0.171504999624891730D-02
-0.119918362350520725D-02-0.234502659677803330D-01-0.124349929766947804D-01
-0.394750288372177252D+00-0.118865303184654020D+00-0.230977672207935067D-01
 0.225801577335899250D-02-0.226149836865391545D-01-0.123428789105732191D-01
-0.387009894976345747D+00-0.162998459695493494D+00-0.475274335969820662D-01
 0.543497999546820477D-02-0.214717487677418793D-01-0.120570564776729542D-01
-0.373200820441902392D+00-0.204583871919465476D+00-0.712162291217521282D-01
 0.832648005697611288D-02-0.200755475134234028D-01-0.116064349822018785D-01
-0.353893470337354021D+00-0.243162330934154292D+00-0.938596311183944459D-01
 0.109338532502984333D-01-0.184715896082141779D-01-0.110153627413783909D-01
-0.329651563779756213D+00-0.278356449641765216D+00-0.115197621534192343D+00
 0.132620417640050914D-01-0.166967303302305824D-01-0.103040844553954685D-01
-0.301027086769691782D+00-0.309855396241126096D+00-0.135007029841151302D+00
 0.153175598894634713D-01-0.147809007241375317D-01-0.948929824676157056D-02
-0.268558612669156996D+00-0.337402385764509080D+00-0.153095007001016345D+00
 0.171071200545810289D-01-0.127484232787700904D-01-0.858471907395848190D-02
-0.232771888934792295D+00-0.360784649170332497D+00-0.169293593382885924D+00
 0.186367110979295649D-01-0.106191712365609767D-01-0.760160487030104876D-02
-0.194181944697757722D+00-0.379825562992656296D+00-0.183455287457601743D+00
 0.199109780770827501D-01-0.840957214559075964D-02-0.654923025119533413D-02
 0.186842948131895087D+00 0.638827084231456899D+00 0.276061865396564354D+00
-0.196036773844505520D-01 0.424360860706823537D-02 0.315044126771773847D-02
 0.107411922334066406D+00 0.651725397482830385D+00 0.286891069594740741D+00
-0.200700473986464038D-01 0.219811815205748335D-02 0.225817329999441113D-02
 0.266208161890726953D-01 0.656370114534958651D+00 0.294086998977074604D+00
-0.202828423297254923D-01 0.121211439526860629D-03 0.133573166664295974D-02
-0.545077950519326088D-01 0.652688707248148633D+00 0.297552380037115816D+00
-0.202384847804787478D-01-0.196047390885268995D-02 0.394893491224387774D-03
-0.134943799211159621D+00 0.640715780375117241D+00 0.297237737985714859D+00
-0.199367949369063900D-01-0.402007140175529749D-02-0.552246362891832955D-03
-0.650136773242734889D-01 0.900301298326657354D+00 0.390410650682383564D+00
-0.174481904461161177D-01-0.109855705921707510D-02-0.476362214943857557D-03
-0.134586746469618920D+00 0.893670669527233480D+00 0.387535396313632929D+00
-0.173239022026531074D-01-0.221545418625267230D-02-0.960699241606695884D-03
-0.203489847745717295D+00 0.882591987641219239D+00 0.382731268117244783D+00
-0.171133450729878999D-01-0.332161243471642073D-02-0.144037855108863525D-02
-0.271379897498071496D+00 0.867119806369172672D+00 0.376021924425743267D+00
-0.168175970657526005D-01-0.441124657288891400D-02-0.191289182727262048D-02
-0.337919103318082018D+00 0.847331639619109192D+00 0.367440980841090795D+00
-0.164382242392376648D-01-0.547866670116191161D-02-0.237577151224223408D-02
 0.108405084618917980D+01-0.771718199257734550D+00-0.383333805761574026D+00
 0.925213840242533440D-02 0.111321393041578171D-01 0.486136531175084853D-02
 0.112008133007132260D+01-0.726514499441550832D+00-0.363552184479094029D+00
 0.876022568268070272D-02 0.114664829133253321D-01 0.502804105131290837D-02
 0.115411033286563947D+01-0.680012715977277846D+00-0.343120986844399081D+00
 0.825157597688051369D-02 0.117810434502228579D-01 0.518608550830217552D-02
 0.118607308513463439D+01-0.632293593394967476D+00-0.322075536532728704D+00
 0.772728864496799884D-02 0.120750249231340537D-01 0.533510378830203465D-02
 0.121590933275673052D+01-0.583440935629914351D+00-0.300452682784186909D+00
 0.718852111744181128D-02 0.123476943606819984D-01 0.547472836607307452D-02
 0.296963721428718450D+00 0.469987843783317327D+01 0.200899780459065469D+01
-0.762319862525409619D-02 0.661151789440098862D-03 0.469963640513295451D-03
```

```
 0.266465861511739441D+00 0.470244005369087104D+01 0.201084217935312815D+01
-0.762568602683512051D-02 0.619658053700624712D-03 0.452223457362566246D-03
 0.235958595433071489D+00 0.470483571859911676D+01 0.201261559042258842D+01
-0.762790176421336563D-02 0.578176570943504079D-03 0.434481904356384880D-03
 0.205443009147515121D+00 0.470706548465743579D+01 0.201431803365064233D+01
-0.762984618999959507D-02 0.536708887889387446D-03 0.416739648612341146D-03
 0.174920187172895605D+00 0.470912941018166387D+01 0.201594950757439606D+01
-0.763151967164380721D-02 0.495256564593383563D-03 0.398997365201801473D-03
 0.939865110408989457D+01-0.182394766896974292D+01-0.116039746488520668D+01
 0.922338565618071346D-03 0.503674030558125090D-02 0.204262136552797621D-02
 0.940231562228465534D+01-0.180379599548001734D+01-0.115222395921025056D+01
 0.909919112108714646D-03 0.503909216932957749D-02 0.204412977379959054D-02
 0.940593044567175474D+01-0.178363496578778657D+01-0.114404444029125085D+01
 0.897491172608719571D-03 0.504141840447532193D-02 0.204562798518269731D-02
 0.940949554043931591D+01-0.176346468241925813D+01-0.113585894893593364D+01
 0.885054813268694983D-03 0.504371900896107613D-02 0.204711599154567587D-02
 0.941301087303403961D+01-0.174328524786837047D+01-0.112766752596328090D+01
 0.872610095641111088D-03 0.504599399961164034D-02 0.204859379483846889D-02
-0.177729734370423706D+02 0.383844239318491676D+01 0.193297540158749026D+01
-0.960973886416227586D-03-0.367312096459336741D-02-0.159582824302039735D-02
-0.177768104537297980D+02 0.382374833119819679D+01 0.192659130077736562D+01
-0.957534172712796821D-03-0.367390953492950336D-02-0.159622194869125402D-02
-0.177806337081453627D+02 0.380905112086437958D+01 0.192020562770401915D+01
-0.954092749701735397D-03-0.367469513518589363D-02-0.159661437437978742D-02
-0.177844431934402891D+02 0.379435077411569965D+01 0.191381838749860989D+01
-0.950649611523560819D-03-0.367547773557936892D-02-0.159700551252197704D-02
-0.177882389027486632D+02 0.377964730304383667D+01 0.190742958534375950D+01
-0.947204757056103238D-03-0.367625728753700370D-02-0.159739534550454537D-02
-0.195707391535787991D+02-0.216014602604430266D+02-0.835608679104130636D+01
 0.237079710506702399D-02-0.183485280544284967D-02-0.811298629320883014D-03
-0.195612543135713466D+02-0.216087979291454566D+02-0.835933131528533968D+01
 0.237162298837771576D-02-0.183398147427178118D-02-0.810963462388057579D-03
-0.195517661689501523D+02-0.216161321116576417D+02-0.836257449849897938D+01
 0.237244941319505185D-02-0.183310970750692834D-02-0.810628113940992090D-03
-0.195422747175319600D+02-0.216234628061869998D+02-0.836581633994575680D+01
 0.237327638821765692D-02-0.183223747621903899D-02-0.810292576824218056D-03
-0.195327799571050278D+02-0.216307900107849171D+02-0.836905683883937157D+01
 0.237410391740781375D-02-0.183136473268549500D-02-0.809956833831454101D-03
-0.305812492297079075D+02 0.349348320443577243D+01 0.103722951500296636D+02
-0.750595021831047494D-04-0.312845374423225591D-02-0.967176440897862799D-03
-0.305815473634011425D+02 0.348096905463394890D+01 0.103684256867072004D+02
-0.740071480116030364D-04-0.312862106936532582D-02-0.967555203567567576D-03
-0.305818412853167025D+02 0.346845423658723018D+01 0.103645547085343408D+02
-0.729536120865895982D-04-0.312878786380731681D-02-0.967933865886072401D-03
-0.305821309907103291D+02 0.345593875246929638D+01 0.103606822159232244D+02
-0.718988856995990796D-04-0.312895409827438145D-02-0.968312420536132916D-03
-0.305824164748092251D+02 0.344342260461128169D+01 0.103568082093364704D+02
-0.708429648788333363D-04-0.312911972468944938D-02-0.968690850149689711D-03
 0.167449358457460822D-02-0.185226178945920250D-02-0.104416267147451879D-02
 0.443756046291836315D-03 0.323938380982394301D-03 0.115717376242538003D-03
 0.188642556933437349D-02-0.168027743857807049D-02-0.980564439860606393D-03
 0.403192221007736659D-03 0.363331290086340794D-03 0.138424235210601557D-03
 0.207694248750386624D-02-0.148962616879642971D-02-0.906006603531797346D-03
 0.358169050899078388D-03 0.398551348171793289D-03 0.159522602818547594D-03
```

0 224392591629318816D-02	-0.128247364776005530D-02	-0.821341954635063879D-03
0.309134144585264576D-03	0.429287102577088247D-03	0.178820724187841763D-03
0.238548500313726222D-02	-0.106113330797978444D-02	-0.727515652192706338D-03
0.256549930896220527D-03	0.455258324847993066D-03	0.196140611306813071D-03
0.249996364829227805D-02	-0.828051650378017505D-03	-0.625558756189130861D-03
0.200893023516349982D-03	0.476212394043977298D-03	0.211316370046303899D-03
0.258594777836768910D-02	-0.585795256683854305D-03	-0.516582587400411103D-03
0.142655210913422082D-03	0.491921139342326667D-03	0.224192600611633498D-03
0.264227350376754433D-02	-0.337039189041788919D-03	-0.401773830038127007D-03
0.823459809873288025D-04	0.502178509894271217D-03	0.234623057358048812D-03
0.266803688959790645D-02	-0.845562713010205436D-04	-0.282390199707510117D-03
0.204964165296617598D-04	0.506799439397036223D-03	0.242469760312127702D-03
0.266260596774183850D-02	0.168793436892709684D-03	-0.159756401397819679D-03
-0.423357839799937815D-04	0.505620270914684052D-03	0.247602757171597535D-03
0.262563547159322088D-02	0.420073843527054604D-03	-0.352600013618605801D-04
-0.105560502824166302D-03	0.498501100956959145D-03	0.249900740140301969D-03
0.255708457477860307D-02	0.666285751084854977D-03	0.896532661365691732D-04
-0.168549079100799517D-03	0.485330383273055509D-03	0.249252723747607409D-03
0.245723764837786786D-02	0.904382713585690553D-03	0.213485337965146361D-03
-0.230628474088672594D-03	0.466032089893090651D-03	0.245560982289160521D-03
0.232672770363839679D-02	0.113129116031494202D-02	0.334693481878889712D-03
-0.291076510449990574D-03	0.440575642269645578D-03	0.238745420309354428D-03
0.216656174700067047D-02	0.134393619530126957D-02	0.451700526262479316D-03
-0.349119163175749066D-03	0.408988677395877233D-03	0.228749495430010522D-03
0.197814673702686051D-02	0.153927445952677084D-02	0.562908446888643480D-03
-0.403931069949811899D-03	0.371372479521355154D-03	0.215547716728315185D-03
0.176331420949902564D-02	0.171433527794632377D-02	0.666716074128012517D-03
-0.454640574942391138D-03	0.327919568620082418D-03	0.199154591063326976D-03
0.152434096338465553D-02	0.186627095080680179D-02	0.761541569497767431D-03
-0.500340663060071798D-03	0.278932486254628866D-03	0.179634676049248048D-03
0.126396254697567806D-02	0.199241643654345525D-02	0.845850083525010680D-03
-0.540107004047487888D-03	0.224842277632172016D-03	0.157113124342085939D-03
0.985375761582443083D-03	0.209035777886078022D-02	0.918186621776378249D-03
-0.573023924266670014D-03	0.166224594407663626D-03	0.131785790520697611D-03
0.692226159486071047D-03	0.215800746215843834D-02	0.977213603301325452D-03
-0.598218392816367295D-03	0.103810844162413258D-03	0.103927665351915659D-03
0.388576724421245336D-03	0.219368351208433141D-02	0.102175191642574133D-02
-0.614901035318453565D-03	0.384915442160492746D-04	0.738981762172702302D-04
0.788547422718424992D-04	0.219618775117836647D-02	0.105082352384293533D-02
-0.622411853703576758D-03	-0.286908227939190077D-04	0.421418399770940502D-04
-0.232224617397356189D-03	0.216487741741791463D-02	0.106369295244677887D-02
-0.620266935257850780D-03	-0.965614560540217353D-04	0.918296622420738488D-05
-0.539757326787618480D-03	0.209972365385492529D-02	0.105990447119945453D-02
-0.608201294510064177D-03	-0.163846082220336889D-03	-0.243863616414233166D-04
-0.838768859432152984D-03	0.200135046547399615D-02	0.103931156988428918D-02
-0.586202474964417808D-03	-0.229215888750582003D-03	-0.579249741800129199D-04
-0.112434598449140768D-02	0.187104879533708004D-02	0.100209562534248124D-02
-0.554529949099888582D-03	-0.291341434335349724D-03	-0.907669165713583670D-04
-0.139177393556140142D-02	0.171076238409410722D-02	0.948771419072078905D-03
-0.513716801684769783D-03	-0.348950682712163059D-03	-0.122249334440163783D-03
-0.163666968326199960D-02	0.152304479689158997D-02	0.880178374641640317D-03
-0.464552479817391026D-03	-0.400885288479341890D-03	-0.151741219039144447D-03
-0.185510210012899122D-02	0.131098997698113058D-02	0.797457825423079068D-03
-0.408048078046005470D-03	-0.446149293058681231D-03	-0.178670058762925873D-03

```
-0.204369117670161903D-02 0.107814136856514836D-02 0.702018042114846170D-03
-0.345388081325942583D-03-0.483945507699221858D-03-0.202543763386775975D-03
-0.219968086621682898D-02 0.828386553141942264D-03 0.595489885424696009D-03
-0.277874148246415743D-03-0.513696800467547887D-03-0.222966014871350607D-03
-0.232098311108600968D-02 0.565845168630636894D-03 0.479676615431832672D-03
-0.206867059266604120D-03-0.535051780622630105D-03-0.239644248937616260D-03
-0.240619356643678023D-02 0.294757595069650873D-03 0.356501517885449105D-03
-0.133732385680182693D-03-0.547876457166276553D-03-0.252390528720035649D-03
-0.245458199200464580D-02 0.193807001291327332D-04 0.227956653918419926D-03
-0.597940453081303254D-04-0.552234935113645180D-03-0.261116414659324982D-03
-0.246606192986191866D-02-0.256104804557687616D-03 0.960553466642578850D-04
 0.137018621399954823D-04-0.548362923018154631D-03-0.265823437098073169D-03
-0.244114504012588724D-02-0.527672753539968947D-03-0.372098351104701893D-04
 0.856122770828739143D-04-0.536637802449964189D-03-0.266590922655600339D-03
-0.238088543370163749D-02-0.791514774112451113D-03-0.169902685340733343D-03
 0.154912180526135961D-03-0.517548450536955507D-03-0.263562779803040508D-03
-0.228681873478433595D-02-0.104408889031682377D-02-0.300172376051943083D-03
 0.220700993045410511D-03-0.491667161805482960D-03-0.256934522043116313D-03
-0.216089967281052613D-02-0.128215493123892585D-02-0.426276235429528096D-03
 0.282202410098315014D-03-0.459625112351593596D-03-0.246941408624778362D-03
-0.71567731E-04 0.17581551E-04
-0.71410119E-04 0.17703380E-04
-0.71301634E-04 0.17829734E-04
-0.71242073E-04 0.17956394E-04
-0.71229442E-04 0.18079329E-04
-0.71260220E-04 0.18194492E-04
-0.71329210E-04 0.18298291E-04
-0.71429778E-04 0.18387393E-04
-0.71553965E-04 0.18458944E-04
-0.71692426E-04 0.18510807E-04
-0.71834904E-04 0.18541323E-04
-0.71970251E-04 0.18549748E-04
-0.72086841E-04 0.18536011E-04
-0.72172712E-04 0.18501130E-04
-0.72216179E-04 0.18447070E-04
-0.72206094E-04 0.18377104E-04
-0.72132665E-04 0.18295759E-04
-0.71988252E-04 0.18208753E-04
-0.71768212E-04 0.18123159E-04
-0.71472197E-04 0.18046674E-04
-0.71104645E-04 0.17987448E-04
-0.70675698E-04 0.17953207E-04
-0.70200855E-04 0.17950675E-04
-0.69700531E-04 0.17984552E-04
-0.69198373E-04 0.18057224E-04
-0.68719484E-04 0.18167979E-04
-0.68287889E-04 0.18313076E-04
-0.67924178E-04 0.18486026E-04
-0.67643530E-04 0.18677965E-04
-0.67454224E-04 0.18878665E-04
-0.67357192E-04 0.19077357E-04
-0.67346191E-04 0.19263854E-04
-0.67408997E-04 0.19429281E-04
-0.67528774E-04 0.19566971E-04
```

```
-0 67686196E-04 0.19672618E-04
-0.67861081E-04 0.19744461E-04
-0.68033973E-04 0.19783285E-04
-0.68187597E-04 0.19791783E-04
-0.68307447E-04 0.19774394E-04
-0.68382637E-04 0.19736515E-04
-0.29572984E-04 0.66689006E-03 0.25000260E-03
-0.20040854E-04 0.69389748E-03 0.16567609E-03
-0.10501899E-04 0.70999842E-03 0.78005905E-04
-0.10310823E-05 0.71473373E-03-0.11429607E-04
 0.82940587E-05 0.70789666E-03-0.10098286E-03
 0.17393744E-04 0.68954471E-03-0.18898462E-03
 0.26187743E-04 0.65999944E-03-0.27379254E-03
 0.34594283E-04 0.61983778E-03-0.35383669E-03
 0.42532469E-04 0.56987535E-03-0.42766635E-03
 0.49923154E-04 0.51113963E-03-0.49398746E-03
 0.56690100E-04 0.44483598E-03-0.55169943E-03
 0.62762832E-04 0.37231040E-03-0.59991935E-03
 0.68076537E-04 0.29500504E-03-0.63800090E-03
 0.72574665E-04 0.21441400E-03-0.66554314E-03
 0.76210563E-04 0.13203872E-03-0.68238983E-03
 0.78949917E-04 0.49343056E-04-0.68862061E-03
 0.80770551E-04-0.32283933E-04-0.68453397E-03
 0.81664824E-04-0.11156326E-03-0.67062164E-03
 0.81640057E-04-0.18735419E-03-0.64753951E-03
 0.80719503E-04-0.25867228E-03-0.61607338E-03
 0.78942379E-04-0.32470445E-03-0.57710265E-03
 0.76364144E-04-0.38481038E-03-0.53156400E-03
 0.73054398E-04-0.43851952E-03-0.48041530E-03
 0.69097863E-04-0.48552058E-03-0.42460416E-03
 0.64591601E-04-0.52564265E-03-0.36504259E-03
 0.59643236E-04-0.55883639E-03-0.30258019E-03
 0.54369666E-04-0.58514485E-03-0.23799510E-03
 0.48893940E-04-0.60468121E-03-0.17198172E-03
 0.43342850E-04-0.61759888E-03-0.10515147E-03
 0.37844424E-04-0.62407018E-03-0.38037688E-04
 0.32525146E-04-0.62426412E-03 0.28893031E-04
 0.27507049E-04-0.61832997E-03 0.95225419E-04
 0.22905893E-04-0.60638785E-03 0.16057603E-03
 0.18828272E-04-0.58852346E-03 0.22457245E-03
 0.15370184E-04-0.56479033E-03 0.28683222E-03
 0.12615109E-04-0.53521921E-03 0.34694397E-03
 0.10632816E-04-0.49983081E-03 0.40445081E-03
 0.94789611E-05-0.45865658E-03 0.45883795E-03
 0.91942038E-05-0.41175820E-03 0.50952751E-03
 0.98045675E-05-0.35925419E-03 0.55587874E-03
```

15 APPENDIX N - TEST OUTPUT LISTING FOR JD 243938.7500

```
NBODY TAPE WAS MOUNTED AT LOGICAL UNIT #  1
REQUIRED JD IS   2439381 FRACTION OF JD IS     0.7500000000
VALUE OF LER IS  2 KISS ARRAY IS   1   1   1   1   1   1   1   1   1   1   1   11

THE XPERT ARRAY: NBODY #  1
-0.382763934741254808D+00   0.244080567052931338D-02  0.406316880018126559D-01
-0.718518963146612313D-02  -0.239785641245379060D-01 -0.121029359294319797D-01
 0.198598011955958818D-02  -0.126614988308146975D-04 -0.210817443269550905D-03

THE XPERT ARRAY: NBODY #  2
-0.492628428405401900D+00   0.464405387375869005D+00  0.240382527847328428D+00
-0.147807805227472928D-01  -0.130968995570729073D-01 -0.496813554278154653D-02
 0.393127144286469117D-03  -0.370602159108962483D-03 -0.191829080477380405D-03

THE XPERT ARRAY: NBODY #  3
 0.988735747484167662D+00   -0.171532112639045969D+00 -0.743864953193943412D-01
 0.291608647434330229D-02    0.154537069009273702D-01  0.670136538285305908D-02
-0.287154436315266708D-03    0.498145160438153602D-04  0.216023937288520704D-04

THE XPERT ARRAY: NBODY #  4
-0.516691021211064347D+00    0.138448529167213685D+01  0.649365243643462153D+00
-0.127080873950444536D-01   -0.312496941749311214D-02 -0.109323936947103881D-02
 0.363523675121017043D-04   -0.974072853096694253D-04 -0.456875759664516512D-04

THE XPERT ARRAY: NBODY #  5
-0.166435997833359361D+01    0.452596688364770738D+01  0.198232891387487253D+01
-0.724062579045067420D-02   -0.195498559399252293D-02 -0.661712644308231289D-03
 0.347818773703325270D-05   -0.945939713084861476D-05 -0.414320870408499977D-05

THE XPERT ARRAY: NBODY #  6
 0.953213545059109957D+01   -0.495292984627549684D+00 -0.616825749662326103D+00
 0.962016777579799973D-04   -0.513545960949777103D-02  0.211912111272918239D-02
-0.322215151483188135D-05    0.158446163647416392D-06  0.204613689711597625D-06

THE XPERT ARRAY: NBODY #  7
-0.179941144172494951D+02    0.287455734875452773D+01  0.151376956646840233D+01
-0.734712103336015413D-03   -0.371794980413469190D-02 -0.161864320725303300D-02
 0.875004059977432300D-06   -0.148823427227519094D-06 -0.776018050434669016D-07

THE XPERT ARRAY: NBODY #  8
-0.189455379596428664D+02   -0.220723692570429044D+02 -0.856473126662139062D+01
 0.242435236097223719D-02   -0.177681461720185412D-02 -0.788915357083100723D-03
 0.204131863473881897D-06    0.224760387596502039D-06 -0.866570466252752758D-07

THE XPERT ARRAY: NBODY #  9
-0.305917648792498298D+02    0.267645856764286250D+01  0.101169237908013934D+02
-0.540149327626287206D-05   -0.313790843312830321D-02 -0.991495102068836896D-03
 0.271105278387094660D-06   -0.331839533362652875D-07 -0.930542054122031540D-07

THE XPERT ARRAY: NBODY # 10
```

```
-0.184081458878085997D-02   -0.132964069811136081D-02  0.808977652470840272D-03
-0.386118342744651906D-03   -0.454758755466053307D-03 -0.201914196201896884D-03
 0.116969183869004316D-03   -0.854762520576093422D-04 -0.520599521943820081D-04
```

THE SOLAR SYSTEM BARYCENTRE ARRAY

```
-0.648690241847625095D-03    0.316151852824177008D-02  0.134384792068807025D-02
-0.682429685982397033D-05   -0.645105958918003548D-06 -0.132415450906337949D-06
 0.287911236384219270D-08   -0.977020444390244370D-08 -0.434978705016568347D-08
```

NUTATION AND LIBRATION ARRAYS

```
 1  -0.59444268E-04   0.31827120E-04  -0.27167588E-03   0.26300143E-01
     0.52425005E-02
 2  -0.97796601E-07   0.37475161E-06   0.0               0.0               0.0
```

A Real Time Display for Satellite Ranging

by

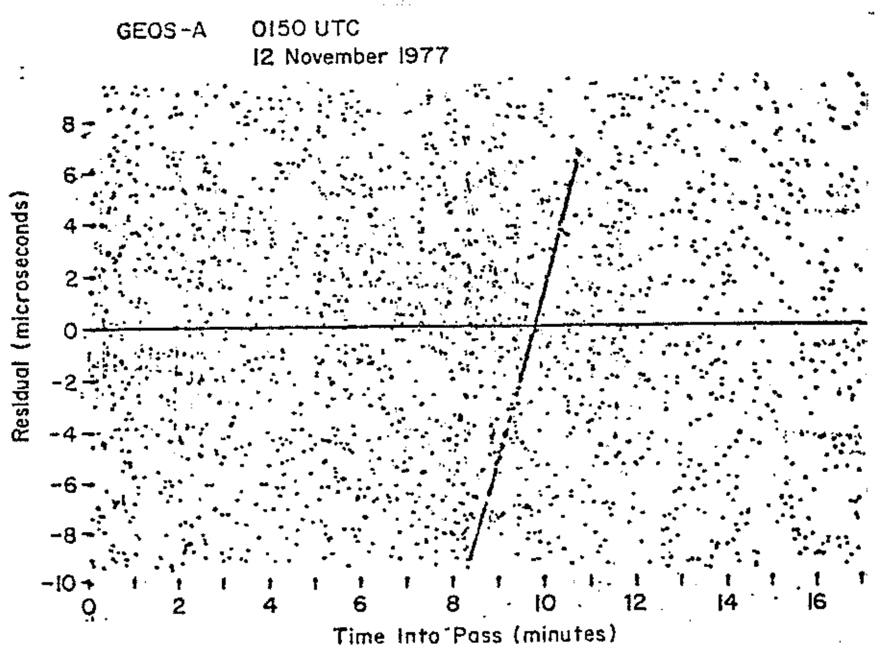
J. Rayner
Department of Physics and Astronomy
University of Maryland
College Park, Maryland

ABSTRACT

The low signal to noise ratio present in single photoelectron ranging prevents direct discrimination between satellite returns and noise. In order to provide a real time indication of success, the University of Maryland has developed a display program which plots residual range against time into the satellite pass. Since the satellite returns are correlated, they show up as a smooth curve which stands out from the random distribution of points due to noise. In practice this has provided very useful feedback for adjusting the pointing and time bias. A stand-alone version of the program is also available for viewing the data after a pass.

Working at the single photoelectron level implies a relatively poor signal to noise ratio. In general, there will be many noise events for each satellite return. This prevents the use of photomultiplier pulses as an indication of satellite returns. Fortunately, the relative stability of the real returns relative to the random distribution of noise events allows the real returns to be distinguished. The problem is then to display the data so that the real returns can be easily identified. We have taken several approaches to this problem in the various laser ranging timing systems that we have designed. The McDonald Lunar ranging system was limited to a teletype for output, but only fired every three seconds, allowing each residual range to be printed. All residual ranges lying within five nanoseconds of a previous residual were flagged and the teletype bell rung. Though simple, this system proved most useful and was quite well received. For the Maui ranging system, the computer video display was used to display a real time histogram of the residual ranges, allowing returns to be identified as a buildup in one or two bins. The parameters of the histogram could be changed in real time, allowing part of the histogram to be expanded once returns had been detected.

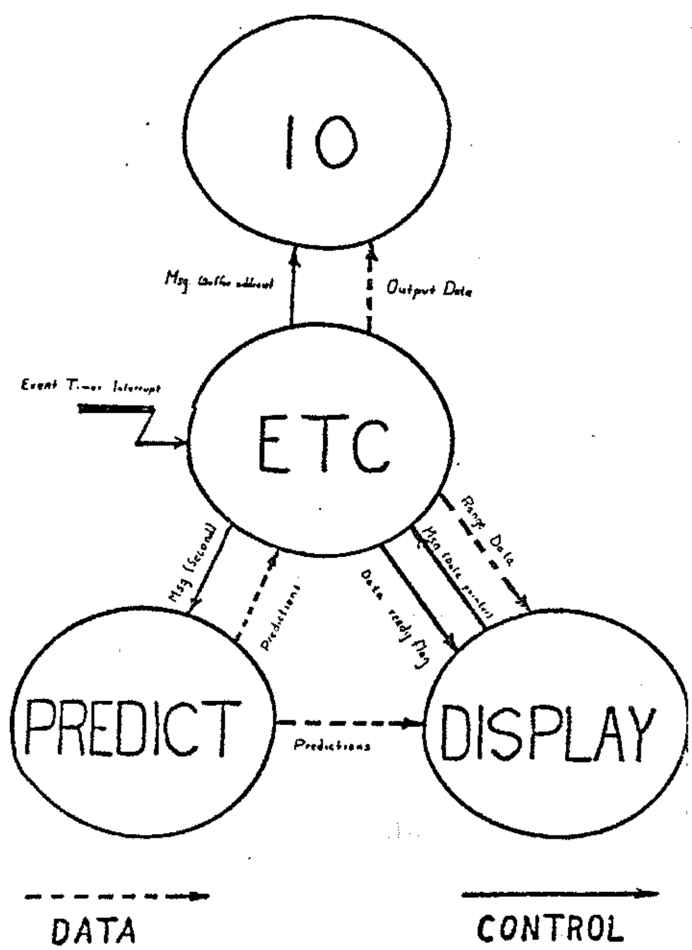
The high repetition rate and relative uncertainty of satellite predictions made a histogram less desirable for our present satellite system. A display of residual range versus time into the pass was chosen instead. As can be seen in the following illustration, satellite returns can be readily identified as a line against the scattered background points due to noise events.



The University of Maryland ranging system is distinguished by three features. We fire at the relatively high rate of 30 shots per second; we operate at the single photoelectron level; and we use an event timer rather than a time interval meter to time the returns. Our computer system consists of a NOVA 2/10 computer with 64 Kbytes of main memory, and 2.4 Mbytes of disk. It has a hardware multiply/divide unit but no hardware floating point. A

Tektronics 4013 graphics terminal is used for operator communication and graphic output, and a digi-data 9-track tape drive is used for bulk data transfer.

Although simple sequential processing is no longer possible, and must be replaced by multi-task processing, there are no significant problems handling 30 shots per second. The only critical point is assuring that the data output keeps up with data acquisition. The ranging program is structured as four cooperating tasks. In order of priority, they are: event timer control, data output, range prediction, and real time display. The various tasks are synchronized by the exchange of intertask messages and share data through common locations. With the exception of the event timer control and output tasks which are time critical, the program is written in FORTRAN.



The event timer control task is synchronized to the laser by an external interrupt from the event time. The start data from the event timer is then read. From this, the event timer control task calculates the expected return time using linear interpolation on predictions supplied by the range prediction task. There are two buffers for range prediction data. When the control task notes that the second has changed, it switches the buffer from which it takes its range predictions, and sends a message to the range prediction task. The prediction task then has a second to calculate the prediction for the following second and place it in the free buffer. The calculated return time is sent to the event timer, causing the range gate to be activated just before the expected return. Activation of the range gate causes another external interrupt that wakes up the control task which then reads the return data from the event timer and stores it in the output buffer. When the output buffer is full, the control task gets a new buffer and sends a message to the output task telling it to write out the old buffer. The output task then writes the buffer to disk and then returns it to the ready queue. Any time not used by the above tasks is available to the display task. When the display task is ready to process a new point it sets a flag which is tested by the control task. When it finishes processing a shot the control task checks the display flag; if set, it copies the data for the shot into a buffer accessible to the display task and sends a message to the display task. On receipt of this message, the display task picks up the data and calculates a new point for the real time display.

The ranging program described above is just one of a complex of programs which make up the Maryland satellite ranging system. These include a post processor which calculates ranges from the raw data and writes then to an IBM/360 compatible tape for later processing; a display program which can re-display the range data using various parameters and scales; and various clock setting and monitoring programs.

```
DISPLAY DGRID CD NXTR FIT4 DFLT2 RANGE PASS EVAL TRANS TDATA FILES ^
CHAR PBYTE PLOT.LB FORT.LB
C  THIS PROGRAM DISPLAYS THE RESIDUAL RANGES FOR A SATELLITE PASS
C    CALL LINE--DISPLAY <SATELLITE> [<STATION> [<DATA FILE>]]
C    IF <STATION> IS OMITTED 48INCH WILL BE USED
C    IF <DATA FILE> IS OMITTED RANGE$DATA WILL BE USED
C    GLOBAL
C      COEF--RANGE PREDICTION DATA (DESCRIBED IN EVAL)
C      PTATA--DISPLAY DATA (DESCRIBED IN DGRID)
C    LOCAL
C      DELTA--RANGE OFFSET(MICROSECONDS)
C      SCALE--VERTICAL SCALE FOR DISPLAY(MICROSEC/DIV)
C      TBIAS--TIME BIAS ON SATELLITE PREDICTION(SECONDS)
C      N--NUMBER OF ARGS ON CALL LINE
C      TJD--TIME(FRACTIONAL JULIAN DAY)
C      T--TIME(SECOND OF THE DAY)
C      IR1--RAW DATA(START TIME)
C      IR2--RAW DATA(STOP TIME)
C      TS--START TIME FOR PASS(SECOND OF THE DAY)
C      TE--END TIME FOR PASS(SECOND OF THE DAY)
C      T1--START TIME(FLOTING POINT)
C      T2--STOP TIME(FLOATING POINT)
```

```
C       MR--MEASURED RANGE(NANO SECONDS)
C       TLEAD--ONE WAY RANGE TO SATELLITE(SECONDS)
C       CR--CALCULATED RANGE(NONOSECONDS)
C       RR--RESIDUAL RANGE(NANOSECONDS)
         COMPILER DOUBLE PRECISION
         INTEGER IR1(7),IR2(11)
         REAL MR
         COMMON DUM1(4),DELTA
         COMMON /COEF/T0,N,COEF(10,3),TSCALE,TBIAS
         COMMON /PDATA/TS,TE


C       READ DELTA,SCALE,TBIAS
         ACCEPT 'DELTA(MICROSEC)=',DELTA
         ACCEPT 'SCALE(MICROSEC/DIV)=',SCALE
         ACCEPT 'TBIAS(SEC)=',TBIAS
C       GET DATA FROM CALL LINE AND ASSIGN FILES
         CALL FILES(N)
         IF(N.LE.0.OR.N.GT.3) STOP BAD CALL LINE
         IF(N.EQ.1) CALL FOPEN(2,'48INCH')
         IF(N.NE.3) CALL FOPEN(3,'RANGE$DATA')
C       GET FIRST POINT
         READ BINARY(3) IR1
         REWIND 3
C       CALCULATE FRACTIONAL JULIAN DAY AND SECOND OF THE DAY
         TJD=IR1(1)
         IF(TJD.LT.0.) TJD=TJD+65536.
         T=IR1(2)
         IF(T.LT.0.) T=T+65536.
         TJD=TJD/2.+T/86400.
         IF(MOD(IR1(1),2).EQ.0) T=T+43200.
C       INITALIZE PREDICTION ROUTINE
         CALL PASS(TJD)
         TS=86400.*DMOD(TS+.5,1.)
         TE=86400.*DMOD(TE+.5,1.)
C       DRAW AND LABEL AXIS FOR DISPLAY
         CALL DGRID(DELTA,SCALE,XS)
         SCALE=SCALE/.078
         TLEAD=.03
C       PLOT NEXT RESIDUAL RANGE
C        GET RAW DATA FOR NEXT POINT
3000     CALL NXTR(IR1,IR2,3,$4000)
C          CALCULATE TIME
           T=IR1(2)
           IF(IR1(2).LT.0) T=T+65536.
           IF(T.LT.T0) T=T+43200.
           IF(T.LT.TS) GO TO 3000
           IF(T.GT.TE) GO TO 4000
C          CALCULATE MEASURED RANGE
           T1=100.*DFLT2(IR1(3),IR1(4))-IR1(5)/20.
           T2=100.*DFLT2(IR2(3),IR2(4))-IR2(5)/20.
           T=T+T1/1D9-.6777215
           MR=T2-T1                  ;MEASURED RANGE
           IF(MR.LT.0.) MR=MR+1D9
```

```
C         CALCULATE PREDICTED RANGE
          TLEAD=RANGE(T+TLEAD)
          CR=1000.*(2D6*TLEAD-DELTA)
C         CALCULATE RESIDUAL RANGE
          RR=MR-CR                ;RESIDUAL RANGE
C         PLOT RESIDUAL RANGE
          IX=(T-TS)/XS
          RR=RR/SCALE
          IF(RR.GT.0..AND.RR.LT.780.) CALL PNTABS(IX,IFIX(RR))
          GO TO 3000
C       END OF DATA FILE--CLOSE UP SHOP
4000    CONTINUE
        CALL MOVABS(0,0)
        CALL ANMODE
        TYPE
        END
C  DGRID DRAWS AND LABELS THE AXIS FOR THE REAL TIME DISPLAY
C    INPUT
C      DELTA--RANGE OFFSET(MICROSECONDS)
C      YS--VERTICAL SCALE
C    OUTPUT
C      XS--HORIZONTAL SCALE
C    GLOBAL
C      PDATA
C        TS--START TIME(MINIUTS)
C        TE--END TIME(MINUTS)
C    LOCAL
C      T--TIME OF DAY(SECONDS)
C      IT--INTEGER TIME OF DAY(SECONDS)
C
        COMPILER DOUBLE PRECISION
        SUBROUTINE DGRID(DELTA,YS,XS)
        COMMON /NFILE/NFILE(6,3)
        COMMON /PDATA/TS,TE
        INTEGER D,M,Y,DX

C     INITALIZE PLOT ROUTINES AND CLEAR SCREEN
        CALL INITT(100)
C     CALCULATE START OF PASS IN MINUTES AND SECONDS
        T=86400.*DMOD(TS+.5,1.)
        IH=T/3600.
        IM=DMOD(T,3600.)/60.
        IT=1440.*DMOD(TE+.5,1.)
        IT=IT-T/60.+1.
        IF(IT.LT.0) IT=IT+1440
C     CALCULATE HORIZONTIAL SCALE
        XS=60.*IT/1024.
        DX=60./XS
C     CALCULATE CIVIL DATE FROM JULIAN DAY
        CALL CD(D,M,Y,INT(TS-.5))
C     LABEL DISPLAY
        WRITE(10,101) NFILE(1,1),M,D,Y,IH,IM,YS
101       FORMAT(10X,S12,I2,2('/',I2),2X,I2,':',I2,
     1      '  SCALE=',F6.3,' MICROSEC/DIV')
```

```
        IY=78.*DELTA/YS
C      PLOT RESIDUAL RANGE EQUALS 0 LINE
        IF(IY.LT.0.OR.IY.GT.780) GO TO 300
          CALL MOVABS(0,IY)
          CALL DRWABS(1023,IY)
C       PLOT HORIZONTAL SCALE
300     DO 309 IX=0,1024,DX
          CALL MOVABS(IX,0)
          CALL DRWABS(IX,5)
309       CONTINUE
C       PLOT VERTICAL SCALE
        DO 319 IY=0,780,78
          CALL MOVABS(0,IY)
          CALL DRWABS(5,IY)
319       CONTINUE
        TS=T
        TE=TS+60*IT
        RETURN
        END
C  CD CALCULATES THE CIVIL DATE (DAY,MONTH,YEAR) FROM THE JULIAN DAY
C    INPUT
C      JD--MODIFIED JULIAN DAY
C    OUTPUT
C      DAY--DAY OF THE MONTH
C      MNTH--MONTH OF THE YEAR
C      YR--YEAR-1900
C
        SUBROUTINE CD(DAY,MNTH,YR,JD)
        INTEGER DAY,MNTH,YR,YR4,YR0,DAY0
        INTEGER DAYS
        COMMON /CDDATA/DAY0,YR0,DAYS(13)
        DATA DAYS/0,31,59,90,120,151,181,212,243,273,304,334,366/
        DATA DAY0/-144/,YR0/68/

        DAY=JD-DAY0
        YR4=DAY/1461
        DAY=MOD(DAY,1461)-1
        YR=DAY/365
        IF(DAY.EQ.-1) YR=0
        DAY=MOD(DAY,365)+1
        IF(YR.NE.0) GO TO 20
          IF(DAY.EQ.59) GO TO 40
          IF(DAY.LT.59) DAY=DAY+1
20      DO 29 MNTH=1,12
          IF(DAY-DAYS(MNTH+1).LE.0) GO TO 30
29      CONTINUE
        STOP
30      DAY=DAY-DAYS(MNTH)
        GO TO 45
40      MNTH=2
        DAY=29
45      YR=4*YR4+YR+YR0
        RETURN
        END
```

```
      SUBROUTINE NXTR(IR1,IR2,NFILE,EOF)
      INTEGER IR1(7),IR2(11),JUNK(4)
      COMMON /NXTRC/ICNT
      DATA ICNT/1/
      READ BINARY(NFILE,END=90)IR1,IR2
      ICNT=ICNT+1
      IF(ICNT.LE.14) RETURN
      ICNT=1
      READ BINARY(NFILE,END=90)JUNK
      RETURN
90    RETURN EOF
      END
;FUNCTION TO CONVERT TWO INTEGER ARGUMENTS TAKEN
;  AS A DOUBLE PRECISION WORD TO DOUBLE PRECISION
;  FLOATING POINT.
;    THE FIRST ARGUMENT MUST BE LESS THAN 256
        .TITL   DFLT2
        .ENT    DFLT2
        .EXTU
        .NREL
        3
DFLT2:  JSR     @.CPYL ;GET ARGUMENTS
        LDA     0,@.T+1,3 ;HIGH ORDER 8 BITS
        LDA     1,@.T+2,3 ;LOW 16 BITS
        LDA     2,EXP   ;EXPONENT=>2^24
        ADD     2,0     ;INSERT EXPONENT
        LDA     2,.T,3  ;LOC(RESULT)
        STA     0,0,2   ;RETURN RESULT
        STA     1,1,2
        SUB     1,1
        STA     1,2,2   ;FILL OUT DOUBLE PRECISION
        STA     1,3,2
        FRET            ;RETURN
EXP:    43000
.T=-167
        .END
C  RANGE RETURNS THE RANGE TO THE SATELLITE IN SECONDS
C
C    INPUTS
C      T--TIME OF DAY IN SECONDS
C    LOCAL
C      DELT--TIME INTO PASS IN SECONDS
C      P--POSITION VECTOR
        COMPILER DOUBLE PRECISION
        PARAMETER C=299792.5
        FUNCTION RANGE(T)
        COMMON /COEF/TO,N,COEF(10,3),TSCALE,TBIAS
        REAL P(3)

        DELT=T-TO
C     CALCULATE POSITION IN EARTH CENTERED INERTIAL FRAME
        CALL EVAL(P,DELT)
C     TRANSFORM TO STATION CENTERED ROTATING FRAME
        CALL TRANS(P,DELT)
```

```
C     CALCULATE RANGE
        RANGE=SQRT(P(1)*P(1)+P(2)*P(2)+P(3)*P(3))/C
C       AZ=57.2957795*ATAN2(P(1),P(2))
C       EL=57.2957795*ATAN2(P(3),SQRT(P(1)*P(1)+P(2)*P(2)))
        RETURN
        END
C  PASS INITALIZES THE VARIABLES USED IN THE SATALITE RANGE CALCULATION
    IT USES TEST DATA IF NO SATELLITE WAS SPECIFIED (N=0)
C
C  INPUT
C    JD--DATE IN FRACTIONAL JULIAN DAYS
C  GLOBAL
C    COEF--SATELLITE PASS DATA(DESCRIBED IN EVAL)
C    STATION--STATION POSITION INFORMATION(DESCRIBED IN TRANS)
C    PDATA--DISPLAY DATA(DESCRIBED IN DGRID)
C
        COMPILER DOUBLE PRECISION
        PARAMETER D1900=-24979.5
        SUBROUTINE PASS(T)
        COMMON T1
        COMMON/COEF/TO,N,COEF(10,3),TSCALE,TBIAS
        COMMON/STATION/SLON,SLAT,CLAT,DELV,DELN,LHAO
        COMMON /PDATA/TS,TE
        REAL LHAO
        INTEGER HR,MIN,SEC
        DATA TS/1310.8416666/,TE/1310.859027777/

        IF(N.EQ.0) GO TO 20
C         READ STATION LOCATION DATA
          READ BINARY(2) SLON,SLAT,CLAT,DELV,DELN
C         READ SATELLITE ORBIT DATA FOR NEXT PASS
10        IF(T.LT.TE) GO TO 20
            READ BINARY(1) TS,TE,TO,N,((COEF(I,J),I=1,N),J=1,3),TSCALE
            GO TO 10
C       MODIFY BEGINING AND ENDING TIMES FOR DISPLAY
20      ACCEPT 'TRIM(MIN) ',T1,T2
          TS=TS+T1/1440.
          TE=TE-T2/1440.
C       CALCULATE LOCAL HOUR ANGLE AT START OF THE PASSS
        ACCEPT 'DELPSI=',DELPSI
        HR=TO/3600.
        MIN=DMOD(TO/60.,60.)+.2
        SEC=DMOD(TO,60.)+.2
        WRITE(10,21) HR,MIN,SEC
21        FORMAT(' PASS STARTS AT ',I2,':',I2,':',I2)
        FCEN=(INT(TS-D1900)+.5)/36525.
        LHAO=DMOD(23925.836+FCEN*(8640184.542+.0929*FCEN)+DELPSI,86400.)
        LHAO=DMOD(LHAO+1.0027379093*TO-SLON,86400.)/13750.987083
        IF(TO.GE.86400.) TO=TO-86400.
        IF(N.NE.0) RETURN
          N=6
          TO=T1
          TYPE 'TO=',TO
          DT=TE-TS
```

```
          TS=T
          TE=TS+DT
        RETURN
        END
C  EVAL CALCULATES THE SATELLITE POSITION IN INERTIAL SPACE
C    FROM THREE N'TH ORDER POLYNOMIALS
C    INPUT
C      DELT--TIME INTO PASS IN SECONDS
C    OUTPUT
C      P--X Y Z POSITION IN EARTH CENTERED INERTIAL FRAME
C    GLOBAL
C      COEF
C        TO--START OF PASS(SECOND OF THE DAY)
C        N--ORDER OF PREDICTION POLYNOMIALS
C        COEF--COEFFICIENTS OF PREDICTION POLYNOMIALS
C        TSCALE--SCALING FACTOR FROM SECONDS FOR ARGUMENT
C          FOR POLYNOMIALS
C        TBIAS--ALONG TRACK BIAS IN SECONDS
C
        COMPILER DOUBLE PRECISION
        SUBROUTINE EVAL(P,DELT)
        COMMON /COEF/ TO,N,COEF(10,3),TSCALE,TBIAS
        REAL P(3)

        T=(DELT+TBIAS)/TSCALE
        DO 90 J=1,3
          POLY=COEF(1,J)
          DO 9 I=2,N
            POLY=POLY*T+COEF(I,J)
9           CONTINUE
          P(J)=POLY
90        CONTINUE
        RETURN
        END
C  TRANS TRANSLATES THE SATELLITE POSITION VECTOR
C    FROM A EARTH CENTERED INERTIAL FRAME
C    TO A STATION CENTERED ROTATING FRAME
C
C    INPUT
C      P--EARTH CENTERED XYZ POSITION VECTOR
C      DELT--TIME INTO PASS IN SECONDS
C    OUTPUT
C      P--STATION CENTERED ENV POSITION VECTOR
C    GLOBAL
C      STATION
C        SLON--STATION LONGITUDE
C        SLAT--SINE OF STATION LATITUDE
C        CLAT--COSINE OF STATION LATITUDE
C        DELV--VERTICAL DEVIATION
C        DELN--NORTH DEVIATION
C        LHAO--LOCAL HOUR ANGLE AT START OF PASS
C    LOCAL
C      LHA--LOCAL HOUR ANGEL
C      RPERS--EARTH ROTATION RATE(RADIANS/SEC)
```

```
C
      COMPILER DOUBLE PRECISION
      PARAMETER RPERS=7.292115854D-5
      SUBROUTINE TRANS(P,T)
      COMMON  /STATION/ SLON,SLAT,CLAT,DELV,DELN,LHAO
      REAL LHAO,LHA,P(3)

      LHA=RPERS*T+LHAO
      SLHA=SIN(LHA)
      CLHA=COS(LHA)
      XE1=P(1)*CLHA+P(2)*SLHA
      P(1)=-P(1)*SLHA+P(2)*CLHA
      P(2)=-XE1*SLAT+P(3)*CLAT-DELN
      P(3)=XE1*CLAT+P(3)*SLAT-DELV
      RETURN
      END
      COMPILER DOUBLE PRECISION
      BLOCK DATA
      COMMON /COEF/TO,N,COEF(10,3),TSCALE,TBIAS
      COMMON /STATION/SLON,SLAT,CLAT,DELV,DELN,LHAO
      REAL LHAO
      DATA N/O/COEF/
     1  -9.0192424D-6,-2.3252777D-4,6.4891695D-2,
     2   3.3866396D0,-1.7610474D2,-3.020634D3,4*0D0,
     3  -1.3092129D-6,1.5710605D-3,2.6791398D-2,
     4  -8.9414493D0,-1.0266486D2,8.0625833D3,4*0D0,
     5   1.525832D-5,-4.3730091D-4,-1.1845482D-1,
     6  -8.451897D-1,3.364123D2,7.0712201D2,4*0D0/
      DATA SLON,SLAT,CLAT/-67961.22933,.6295994624,.776919891/
      DATA DELV,DELN/6369.699554,-20.91386051/
      DATA TO/29520./
      DATA TBIAS,TSCALE/0.,60./
      END
C  FILES READS THE ARGUMENTS FROM THE PROGRAM CALL LINE
C    AND OPENS FILES 1...N TO THOSE FILENAMES
C    THE FILE NAMES ARE STORED IN NFILE
C    OUTPUT
C      N--THE NUMBER OF FILES
C    GLOBAL
C      NFILE
C        NFILE--THE FILE NAMES
C
      SUBROUTINE FILES(N)
      INTEGER CHAR
      COMMON /NFILE/NFILE(6,3)

      CALL FOPEN(0,'COM.CM')
      DO 99 I=1,12
        IF(CHAR(0,$300).EQ.0) GO TO 100
99      CONTINUE
100   DO 199 I=1,4
        JUNK=CHAR(0,$300)
199     CONTINUE
      N=0
```

```
        DO 299 J=1,3
          DO 209 I=1,12
            ICHAR=CHAR(0,$300)
            CALL PBYTE(ICHAR,NFILE(1,J),I)
            IF(ICHAR.EQ.0) GO TO 210
209         CONTINUE
210       N=N+1
          CALL FOPEN(N,NFILE(1,J))
          DO 219 I=1,4
            JUNK=CHAR(0,$300)
219         CONTINUE
299       CONTINUE
300     RETURN
        END
        .TITL   CHAR
        .ENT    CHAR
        .EXTU
        .NREL
.T=-167
.V=.T+200
        3
CHAR:   JSR     @.CPYL
        LDA     2,@.T+1,3 ;GET UNIT
        LDA     0,.IOCA ;GET CHANNEL TABLE
        MOVZL   2,2
        ADD     0,2
        LDA     2,0,2   ;GET CHANNEL
        LDA     0,BPTR
        SUBZL   1,1     ;GENERATE 1
        .SYSTM          ;READ CHARACTER
        .RDS    77
        JMP     ERR
        LDA     0,BUF
        MOVS    0,0
        STA     0,@.T,3         ;RETURN THE RESULT
        FRET
BPTR:   2*BUF
BUF:    0
ERR:    LDA     0,P6    ;HANDLE ABNORMAL RETURNS
        SUB     2,0,SZR ;EOF?
        JMP     ERR1    ;  NO
        JSR     @.AFRTN         ;  YES-TAKE EOF RETURN
        @.V+2
ERR1:   .SYSTM          ;  NO-ABORT PROGRAM
        .ERTN
        HALT            ;NEVER GET HERE
P6:     6
        .END

        .TITL   PBYTE
        .ENT    PBYTE
        .EXTU
        .NREL
.T=-167
```

```
        3
PBYTE:  JSR     @.CPYL
        LDA     1,@.T,3 ;GET DATA
        LDA     0,@.T+2,3 ;GET BYTE POSITION
        SUBZL   2,2     ;GENERATE ONE
        SUB     2,0     ;CHANGE TO 0 ORIGN
        LDA     2,MSK   ;=377
        AND     2,1     ;CLEAN UP INPUT BYTE
        MOVZR   0,0,SNC ;WHICH BYTE?
        MOVS    1,1,SKP ;  LEFT-FLIP INPUT
        MOVS    2,2     ;  RIGHT-FLIP MASK
        LDA     3,.T+1,3 ;GET BASE ADDRESS
        ADD     0,3     ;GET TARGET ADDRESS
        STA     3,TARG  ;SAVE IT
        LDA     3,0,3   ;GET TARGET WORD
        AND     2,3     ;CLEAR HALF
        ADD     1,3     ;INSERT NEW BYTE
        STA     3,@TARG        ;PUT IT BACK
        FRET
TARG:   0
MSK:    377
        .END
```

Orienting a Transportable Alt-Azimuth Telescope

by

Randall L. Ricklefs
Department of Astronomy
University of Texas
Austin, Tx 78712

ABSTRACT

The Transportable Laser Ranging System utilizes a telescope mount modeling system which allows tracking within 15 arcsec of a nominal satellite path from a remote, often poorly prepared site within a few hours of setup. The program allows the operator to select pointing targets, usually stars whose positions are available from a machine-readable star catalog, to point to these objects, and to record their predicted and observed positions. The same program can then perform a non-linear least-squares fit of the pointing data from up to 20 objects to obtain specified combinations of the 9 mount parameters. The mount model and its effectiveness are also discussed.

1 Introduction

The Transportable Laser Ranging System (TLRS) is designed to be driven to various sites and to be ready to range satellites within hours. This requires a quick method of determining the orientation of the telescope so that the satellite can be tracked accurately enough to obtain data.

2 Orientation Technique

The orientation process requires that one have a model for the behavior of the telescope under the influence of such effects as optical and mechanical misalignment, flexure, and encoder zero points as well as a method of acquiring pointing data and solving for the model parameters. Program ORIENT, diagramed in Figures 1-3, fills these needs.

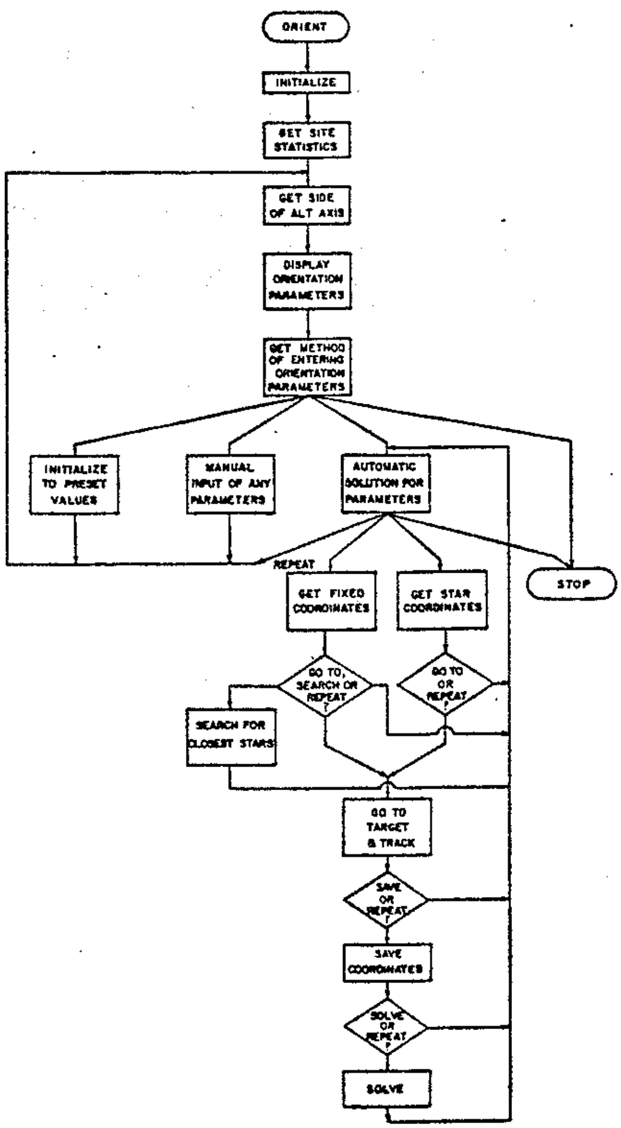


Figure 1: ORIENT Flow Diagram

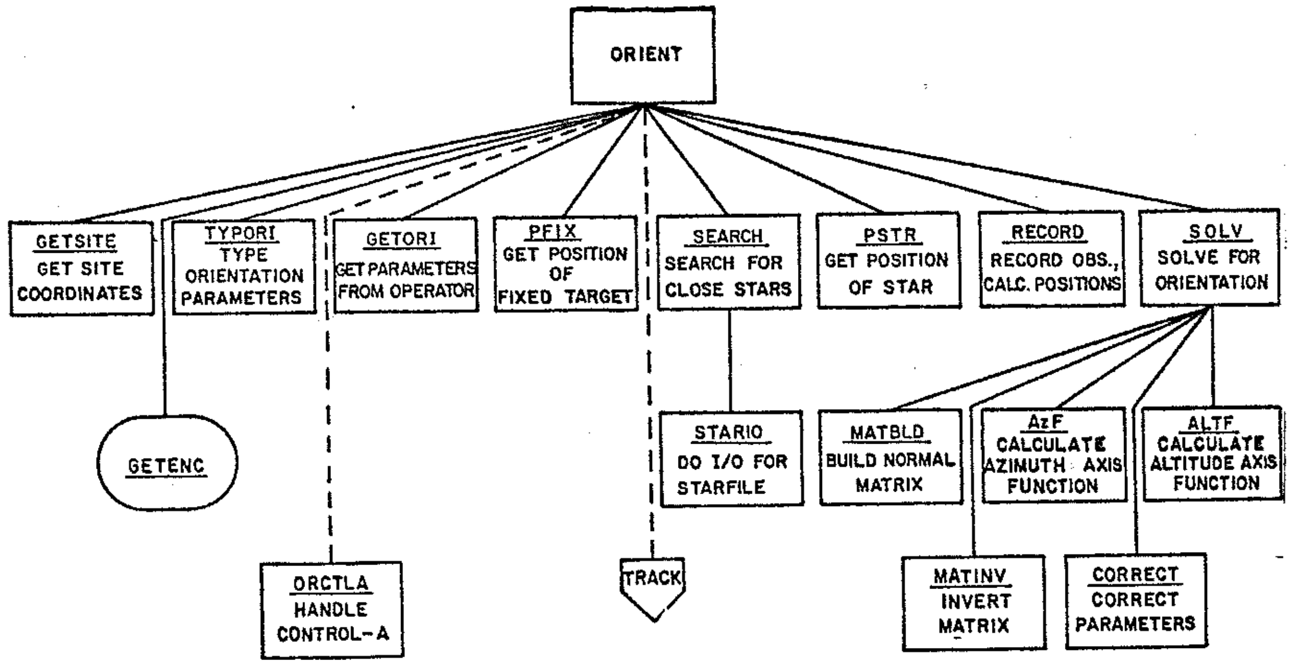


Figure 2: ORIENT Hierarchy Diagram

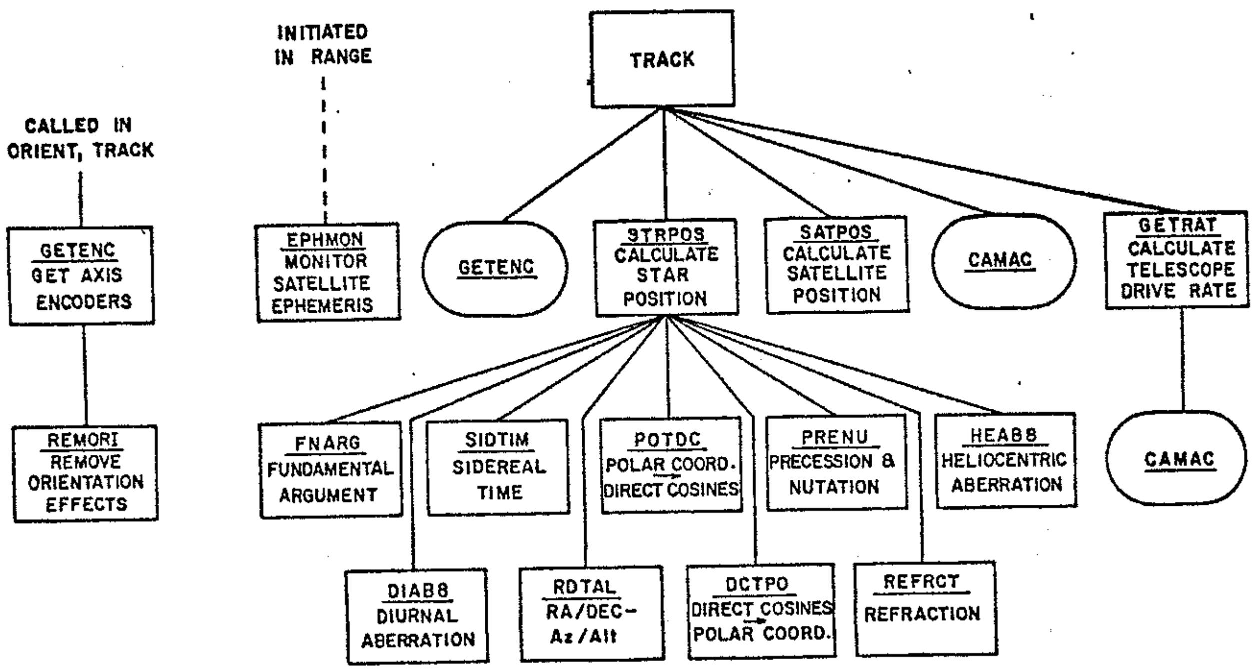


Figure 3: Tracking Subsystem Hierarchy Diagram

This interactive program queries the operator for site coordinates once at each site, for side of the altitude axis--experience has shown that one set of parameters will not perform adequately on both sides of the axis, and for method of model parameter input--manual entry or fit of observations. The operator can choose parts of the sky in which stars are to be observed i.e., everywhere for an initial global fit, and along the satellite path for fine timing prior to a pass, and let the the computer scan its catalog to find stars near these positions.

The operator can command the computer to go to a particular star and wait until it arrives. The operator then centers the star on the fiducial on the CRT monitor displaying the telescope field, and automatically records the predicted and observed positions.

After recording the desired number of star positions, the operator can have the system solve for the desired mount parameters. During initial orientation at a site, the operator may spend quite a while finding the first star. When its position has been recorded, he solves for the axis encoder offsets, the 2 largest terms, and bootstrap his way to other stars, constantly solving for more parameters, until 10-20 stars have been observed and all the significant parameters have been fit.

3 Mount Model

The biggest difficulty in developing the system was determining the terms for the analytical mount model. The basic equations can be derived by following a beam of light down the telescope tube and mathematically rotating the beam through small errors in mirror and telescope misalignments as well as azimuth and altitude. If one cancels all cross terms as one goes, the problem becomes manageable and one arrives at the first-order effects of misalignment. One difficulty is that the cross terms may be significant for a very badly misaligned optical system. In fact, the whole advantage of having an analytical model is that misalignments can be spotted immediately from the results of a fit and corrected. For instance, if the tilt parameters are large (>.5 degree or so), the crew can adjust the tower to make that parameter reasonable.

It turns out that the theoretical model does not adequately describe our mount. The final TLRS mount model, shown in Figure 4, contains several empirical terms.
The sin(altitude) term is a major factor not determined analytically; its origin is still not known. The latch flexure is a small magnitude term which is due to the massive beam director deforming 3 small latches holding it to the tower. The encoder wobble terms are historical: a complaint on the poor altitude axis pointing arose after the alt-axis encoder backlash spring broke and wrapped itself around the shaft in such a way as to bend it. This does point out that a good model can account for almost any error!

To insure an adequate set of parameters, it has been necessary to fine-tune the fit prior to each pass by fitting stars along the satellite path. The values of the model parameters do change significantly from pass to pass. In fact, if one applies successive sets of parameters to the same predicted position, the total corrections vary by 3 to 4 arc minutes in both axes. This implies either that our model is missing some important terms, that there are significant variations in the tower and support surfaces due to thermal effects or settling, or that picking stars along a particular path biases the parameters quite significantly towards one part of the sky. Note that only

$$\begin{aligned} Az &= Az_I + X_1 \\ &+ (-X_3 \sin Az_I - X_4 \cos Az_I + \\ &\quad X_5 \cos 3Az_I + X_6) * \tan Alt_C \\ &+ X_7 \sec Alt_C \end{aligned}$$

$$\begin{aligned} Alt &= Alt_I + X_2 \\ &- X_3 \cos Az_I + X_4' \sin Az_I \\ &+ X_8 \sin Alt_C \\ &+ X_9 \cos [32.8 (Alt_C - X_{10})] \end{aligned}$$

WHERE THE COEFFICIENTS REPRESENT:

Az, Alt - CORRECTED ANGLES,

Az_I, Alt_I - ANGLES READ FROM ENCODERS,

X_1, X_2 - OFFSETS FROM ENCODER ZERO POINTS TO AZ AND ALT ZERO POINTS,

Alt_C - $Alt_I + X_2$, APPROXIMATE CORRECTED ALT,

X_3, X_4 - TILTS OF TELESCOPE ALONG 2 PERPENDICULAR HORIZONTAL AXES,

X_5 - FLEXURE IN 3 LATCHES ATTACHING BEAM DIRECTOR TO TOWER,

X_6 - NON-ORTHOGONALITY OF AZ AND ALT AXES,

X_7 - TRANSVERSE MISALIGNMENT IN ALT MIRROR,

X_8 - EMPIRICAL TERM,

X_9 - WOBBLE IN ALT ENCODER SHAFT, AND

X_{10} - PHASE OF THIS WOBBLE.

Figure 4: TLRS Mount Model

5-20 stars are used in any one of these fits.

4 MLRS Improvements

The McDonald Laser Ranging System (MLRS) contains hardware and software similar to that on TLRS. The software changes include use of the FK4 catalog (obtained on magnetic tape from USNO) for pointing instead of extracts from the American Ephemeris and Nautical Almanac's "Mean Places of Stars, 1978.0" (AENA, 1976). Also, the star position routines have been improved to include proper motion, more nutation terms, and a corrected diurnal aberration routine. The new star search algorithm allows picking the initial radius of search, initial faintest magnitude, and a preference for stars that are brightest or closest to the nominal position. The least squares routine has been generalized for all spherical coordinate systems and for a maximum of 20 model parameters and 200 observations.

An additional new feature is the "auto star" mode in which the program automatically and successively picks stars in a grid across the sky and points to them. When the computer-readable Reticon camera becomes operational, the operator will be able to instruct the program to "observe 100 stars" and leave hands off while the system does the rest.

5 Software Sets

The software sets accompanying this paper are the star position package and the non-linear least-squares package. The former was written mainly by Mark A. Powell and was, in part, installed, debugged, and enhanced by the author. Effects included are coordinate transformations, sidereal time (ASAENA,1961), fundamental arguments (APAENA VI, APAENA XV, Danby,1962), proper motion (Van de Kamp,1964), precession and nutation (ESAENA,1961), heliocentric and diurnal aberration (ESAENA,1961, Woolard and Clemence,1966), and refraction (Marini and Murray,1973). With the exception of the diurnal aberration, all effects are treated rigorously. The error in star positions due to approximations in this software is apparently less than 0.5 arcsec. STARDEM is the test driver which generates two corrected apparent star positions. Comparison coordinates are from Apparent Places of Fundamental Stars, 1980 (APFS ,1978).

The least squares package is tailored to a spherical coordinate system having longitudinal (RA,azimuth, or X) and latitudinal (Dec,azimuth, or Y) coordinates. It also allows use of more than one model to describe the mount. This is done with the MODEL variable which is usually recorded with each observation. This can be used in FUNC to select the proper model for the optical path the observation pertains to. One should refer to the paper by Jefferys (Jefferys,1980) for a thorough, general treatment of the method of least-squares. One can use any standard matrix inversion routine to handle the matrix of normal equations. The one we use is by Bevington (Bevington,1969). The TLRS mount model and a sample data set are used by LSQDEM for the test case.

6 Conclusion

The TLRS mount orientation program has been operating successfully in the field for almost two years. The goals for the program have been met: at several sites LAGEOS returns were obtained on the first pass attempted. On at least one occasion, this was within a few hours of arrival. There is more to be learned, however, about modeling the mount's performance. Time could be saved if a single global fit could be used for both front and back sides of the axis for periods of days or weeks. If the causes of the time-variation of the fits are thermal instability or settling, there may be little that can be done to improve the situation.

7 References

American Ephemeris and Nautical Almanac, 1978, Nautical Almanac Office, U.S. Naval Observatory, Washington, U.S. Government Printing Office, 1976.

Apparent Places of Fundamental Stars, 1980, Heidelberg, Astronomisches Rechen Institut, 1978.

Astronomical Papers of the American Ephemeris and Nautical Almanac VI, Nautical Almanac Office, U.S. Naval Observatory, Washington, U.S.

Government Printing Office.

Astronomical Papers of the American Ephemeris and Nautical Almanac XV, Nautical Almanac Office, U.S. Naval Observatory, Washington, U.S.Government Printing Office.

Bevington. P.R., Data Reduction and Error Analysis for the Physical Sciences, New York, McGraw-Hill, 1969.

Explanatory Supplement to the American Ephemeris and Nautical Almanac. London, Her Majesty's Stationery Office, 1961.

Danby, J.M., Fundamentals of Celestial Mechanics, New York, MacMillan, 1962.

Jefferys, W.H., Astron J. 85(2), February,1980.

Marini, J.W. and C.W. Murray, Correction of Laser Ranging Tracking Data for Atmospheric Refraction at Elevations Above 10 Degrees, NASA Technical Document X-591-73-3531 (1973).

Woolard, E.W., and G.M. Clemence, Spherical Astronomy, New York and London, Academic Press, 1966.

Van de Kamp, P., Elements of Astromechanics, San Francisco, W.H. Freeman, 1964.

8 APPENDIX A

```
        COMPILER DOUBLE PRECISION
C       ** STAR DEM **
C
C       MLRS/TLRS STAR PREDICTION SOFTWARE DEMONSTATION PROGRAM
C
C       R.L. RICKLEFS
C       UNIVERSITY OF TX, MCDONALD OBS.
C
C  REMEMBER TO LOAD BDMATH TO DEFINE COMMON /MATH/!
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$RLR 10/81
        INTEGER IC(8)
        REAL JD,LST
        COMMON /STCMN/ CATEP
        COMMON /SITE/ SITENO,MOVENO, LONG,GDLAT,LAT,HGT
        INTEGER SITENO
        REAL LONG, LAT, GDLAT, HGT
        COMMON /FUNAR/ ARG(12)
        COMMON /TARGET/ RA,DEC ,PMR,PMD,PMRD,PMDD,NSTAR
        COMMON /DIALOG/ IFILLO,IPTR,LINE(80),DW(2),DPNUM
        INTEGER IPTR,LINE,DW,SCAN
        COMMON /ENVPAR/ PRES, TEMP, FILL1(2)
        COMMON /MATH/ PIO2,PI,TWOPI,RDPDG,RDPHR,RDPAS,RDPTS
C
C  1950.0 IS DEFAULT CATALOG EPOCH
        DATA CATEP /2433282.423/
        DATA LONG,GDLAT,LAT,HGT  /4*0./
C****************************************************************
C
C       INITIALIZATION
C
C****************************************************************
C  OPEN OUTPUT FILE
        CALL OPEN(12,"STARSTAT",2,IERR)
        JD = 2444446.8
        INDX = 0
10      INDX = INDX+1
        GO TO (11,12,13), INDX
C  CANOPUS - FK4# 245
11      RA = 1.6704613902
        DEC = -.9192234655
        .290 * RDPTS
  PMD = 2.22 * RDPAS
  PMRD = .004 * RDPTS
  PMDD = -.04 * RDPAS
  IC(1) = 6
  IC(2) = 23
  IC(3) = 29
  IC(4) = 059
  IC(5) = -52
  IC(6) = 40
  IC(7) = 61
```

```
      IC(8) = 970
      GO TO 20
C  ETA UMA - FK4# 09
12    RA = 3.6022355123
      DEC = 0.8650245926
      PMR = -1.285 * RDPTS
      PMD =  -1.41 * RDPAS
      PMRD = .014 * RDPTS
      PMDD = -.08 * RDPAS
      IC(1) = 13
      IC(2) = 46
      IC(3) = 45
      IC(4) = 337
      IC(5) = 49
      IC(6) = 25
      IC(7) = 0
      IC(8) = 570
      GO TO 20
C  DONE
13    CALL FCLOS(12)
      STOP
C  SET UP FUNDAMENTAL ARGUMENTS FOR PROCESSING OF STAR POSITION
20    CALL FNARG(CATEP,JD)
C  WRITE THE RESULTS
      RLONG = LONG/RDPHR
      RLAT = LAT/RDPDG
      RGDLAT = GDLAT/RDPDG
      RRA = RA/RDPHR
      RDEC = DEC/RDPDG
      RPMR = PMR/RDPTS
      RPMD = PMD/RDPAS
      RPMRD = PMRD/RDPTS
      RPMDD = PMDD/RDPAS
      WRITE(12,1005) RLONG, RGDLAT, RLAT, TEMP,PRES, JD, ARG,
    X                RRA,RDEC, RPMR,RPMRD, RPMD,RPMDD,CATEP
1005  FORMAT(" SITE:"/10X,"LONG:   ",F10.6 /
    X        10X,"GDLAT: ",F10.5 /10X,"LAT:   ",F10.5//
    X        " ENVIRONMENT:"/10X,"TEMP:  ",F5.0/10X,"PRES: ",F6.0//
    X        " TIME (JD): ",F15.6//
    X        " FUNDAMENTAL ARGUMENTS (RADIANS AND UNITLESS QUANTITIES):"/
    X        10X,"OBLIQUITY OF ECLIPTIC: ",10X,E20.10/
    X        10X,"PRECESS (ZETA): ",17X,E20.10/
    X        10X,"PRECESS (ZEE=RA OF NODE): ",7X,E20.10/
    X        10X,"PRECESS (THETA=INCL OF EQUATOR): ",E20.10/
    X        10X,"NUTATE (D PSI): ",17X,E20.10/
    X        10X,"NUTATE (D ETA): ",17X,E20.10/
    X        10X,"GEO X OF SUN: ",19X,E20.10/
    X        10X,"GEO Y OF SUN: ",19X,E20.10/
    X        10X,"GEO Z OF SUN: ",19X,E20.10/
    X        10X,"ECC OF SUN: ",21X,E20.10/
    X        10X,"TRUE ANOM OF SUN: ",15X,E20.10/
    X        10X,"GEOM LONG OF SUN: ",15X,E20.10//
    X        " STAR INFO:"/
    X        " (RA,DEC IN DEG,HOURS; CORRECTIONS IN SEC OF ARC,TIME) "/
```

```
     X           10X,"RA:  ",F11.7/10X,"DEC: ",F11.7/
     X           10X,"MU RA:  ",F9.3, 10X,"MU DOT RA:  ",F9.3/
     X           10X,"MU DEC: ",F9.3, 10X,"MU DOT DEC: ",F9.3//
     X           10X,"CATALOG EPOCH (JD): "F15.6)

C  DETERMINE SIDEREAL TIME
        CALL SIDTIM(JD, LONG, LST)
        RLST = LST/RDPHR
        WRITE(12,1010) RLST
1010    FORMAT(/" LOCAL SIDEREAL TIME (HOURS): ",F11.7)
C  CORRECT FOR PROPER MOTION
        CALL PRMTN(RA,DEC, PMR,PMD,PMRD,PMDD, (JD-CATEP)/36534.22,PRA,PDEC)
        IF (PRA.LT.0.) PRA = PRA+TWOPI
        RRA=PRA/RDPHR
        RDEC=PDEC/RDPDG
        DRA=(PRA-RA)/RDPTS
        DDEC=(PDEC-DEC)/RDPAS
        WRITE(12,1012) RRA,RDEC,DRA,DDEC
1012    FORMAT(/" RESULTS OF PROPER MOTION:"/
     X           5X,"NEW RA,DEC: ",F11.7,2X,F10.6/5X," DRA,DDEC:",F9.2,5X,F8.2)
C  CONVERT RA, DEC TO DIRECTION COSINES
        CALL POTDC(1, PRA,PDEC, X,Y,Z)
C  PRECESS & NUTATE
        CALL PRENU(X,Y,Z,XP,YP,ZP)
        CALL DCTPO(1,XP,YP,ZP,TRA,TDEC)
        IF (TRA.LT.0.) TRA = TRA+TWOPI
        RRA = TRA/RDPHR
        RDEC = TDEC/RDPDG
        DRA = (TRA-PRA)/RDPTS
        DDEC = (TDEC-PDEC)/RDPAS
        WRITE(12,1015) RRA,RDEC, DRA,DDEC
1015    FORMAT(/" RESULTS OF PRECESSION & NUTATION: "/
     X           5X,"NEW RA,DEC: ",F11.7,2X,F10.6/5X,"DRA,DDEC: ",F8.2,5X,F8.2)
C  APPLY HELIOCENTRIC ABERRATION
        CALL HEAB8(XP,YP,ZP, 1 ,X,Y,Z)
        CALL DCTPO(1,X,Y,Z,TPRA,TPDEC)
        IF (TPRA.LT.0.) TPRA = TPRA+TWOPI
        RRA = TPRA/RDPHR
        RDEC = TPDEC/RDPDG
        DRA = (TPRA-TRA)/RDPTS
        DDEC = (TPDEC-TDEC)/RDPAS
        WRITE(12,1020) RRA,RDEC, DRA,DDEC
1020    FORMAT(/" RESULTS OF HELIOCENTRIC ABERRATION:"/
     X           5X,"NEW RA,DEC: ",F11.7,2X,F10.6/5X,"DRA,DDEC: ",F8.2,5X,F8.2)
C  CORRECT FOR DIURNAL ABERRATION
        CALL DIABR(X,Y,Z, LST,LAT, XD,YD,ZD)
        CALL DCTPO(1, XD,YD,ZD, TRA,TDEC)
        IF (TRA.LT.0.) TRA = TRA+TWOPI
        RRA = TRA/RDPHR
        RDEC= TDEC/RDPDG
        DRA = (TRA-TPRA)/RDPTS
        DDEC = (TDEC-TPDEC)/RDPAS
        WRITE(12,1025) RRA,RDEC,DRA,DDEC
1025    FORMAT(/" RESULTS OF DIURNAL ABERRATION:"/
```

```
     X          5X,"NEW RA,DEC: ",F11.7,2X,F10.6/5X,"DRA,DDEC: ",F8.2,5X,F8.2)
C  CONVERT TO HH/DD MM SS
      CALL RDTHR(TRA,IH,IM,TSEC)
      CALL RDTDG(TDEC,ID,IMN,ASEC)
      WRITE(12,1027) IH,IM,TSEC,ID,IMN,ASEC
1027  FORMAT(5X,"CON RA,DEC: ",2(2I3,F7.3,5X))
      WRITE(12,1026) IC
1026  FORMAT(/" COMPARISON POSITION (W/O DIURNAL ABERRATION):"/
     X          17X,2(3I3,".",I3,5X))
C  CONVERT RA/DEC TO AZ/ALT
      CALL RDTAL(XD,YD,ZD, GDLAT, LST, XP,YP,ZP)
C  CONVERT AZ/ALT DIRECTIONS TO AZ/ALT
      CALL DCTPO(1, XP,YP,ZP, AZ,ALT)
C  CALCULATE REFRACTION (2ND & 4TH PARAMETERS ARE DUMMY RANGE)
      CALL REFRCT(ALT,0., FALT,DUMMY)
      DALT = (FALT-ALT)/RDPAS
      RAZ = AZ/RDPDG
      RALT = ALT/RDPDG
      RFALT = FALT/RDPDG
      WRITE(12,1030) RAZ,RALT,RFALT,DALT
1030  FORMAT(/" RESULTS OF REFRACTION:"/
     X          5X,"AZ,ALT: ",2F11.6/
     X          5X,"REFRACTED ALT: ",F11.6/5X,"DALT: ",F8.2//)
      GO TO 10
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE SIDTIM(JD, LONG, LST)
C  CALCULATE LOCAL APPARENT SIDEREAL TIME (LST)
C  AT MODIFIED JULIAN DATE (MJD)+UNIVERSAL TIME (UT)
C  AT THE SITE WITH LONGITUDE LONG.
C  THE METHOD IS DOCUMENTED ON PAGE 84 OF THE EXPLANATORY SUPPLEMENT
C  (ESAENA).
C
C       ** FNARG MUST BE CALLED BEFORE SIDTIM **
C
C               INPUT:  JD       - JULIAN DATE (DECIMAL DAYS)
C                       LONG     - EAST LONGITUDE OF SITE(RADIANS)
C
C               << /FUNAR/ >>
C                       DPSI     - DELTA PSI, NUTATION IN LONGITUDE AT
C                                        TIME JD.
C                       EPSLN    - OBLIQUITY OF ECLIPTIC AT TIME JD
C
C               << /SIDCOM/ >>
C                       A,B,C    - COEFFICIENTS FOR SIDEREAL TIME CALCS.
C                       STEPOK   - STARTING EPOCH OF SID. TIM SERIES
C                                        (JAN 0.5, 1900)
C
C               OUTPUT: LST      - LOCAL SIDEREAL TIME (RADIANS)
C
C               COMMONS: SIDCOM
```

```
C                           MATH   -- COMMONLY USED MATH CONSTANTS
C                                                                  RLR
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C
        COMMON /FUNAR/ EPSLN,ZETA,ZEE,THETA,DPSI,DETA,
     1          XS,YS,ZS,ESUN,TSAN,GELS
C
        COMMON /SIDCOM/ A, B, C, STEPOK
C
        COMMON /MATH/ PIO2, PI, TWOPI, RDPDG, RDPHR, FILL(2)
        REAL JD, LONG, LST
C
C  COEFFICIENTS FOR SIDEREAL TIME (A IN HOURS, B&C IN SEC OF TIME)
        DATA A, B, C/6.646065556, 8640184.542, 0.0929/
C
C  START EPOCH OF SIDEREAL TIME SERIES (JAN 0.5, 1900)
        DATA STEPOK /2415020.0/
C
        TU = (JD - STEPOK)/36525.
C
C  EVALUATE SERIES FOR LMST AT GREENWICH
        GST = AMOD(A + (B*TU + C*TU**2.)/3600., 24.) * RDPHR
C
C  MODIFY RESULT FOR FRACTIONAL DAY, EAST LONGITUDE OF SITE,
C  AND EQUATION OF EQUINOXES. THE LATTER CONVERTS RESULT FROM
C  MEAN TO APPARENT SIDEREAL TIME.
        UTF = AMOD(JD-INT(JD)+.5, 1.)
        LST = UTF*TWOPI + GST + LONG + DPSI*COS(EPSLN)
        LST = AMOD(LST, TWOPI)
C
        RETURN
        END




        COMPILER DOUBLE PRECISION
        SUBROUTINE REFRCT(ALT,RANGE, ALTR,RANGER)
C
C  REFRACT SATELLITE ALTITUDE AND RANGE.
C  ALGORITHMS TAKEN FROM GODDARD LETTER OF 10/20/78,"LASER SOFTWARE NOTES"
C   FROM J. FRISKEY.
C  ORIGINALLY FROM "CORRECTION OF LASER RANGING TRACKING DATA FOR
C  ATMOSPHERIC REFRACTION AT ELEVATIONS ABOVE 10 DEGREES",
C  A NASA TECH DOCUMENT BY J.W. MARINI, &C.W. MURRAY,JR.
C
C       INPUT:  ALT     -- ALTITUDE IN RADIANS
C               RANGE   -- RANGE IN NSEC (TWO WAY)
C
C       << /ENVPAR/ >>
C               BARO    -- BAROMETRIC PRESSURE IN MILLIBARS
C               TEMP    -- ATMOSPHERIC TEMPERATURE IN DEGREES KELVIN
C               HUMID   -- ATMOSPHERIC HUMIDITY IN %. SMALL EFFECT ON
C                          REFRACTION, HENCE LEFT AT 50%.
```

```
C                LAMBDA  -- OBSERVING WAVELENGTH IN MICRONS (10000 ANG)
C
C       OUTPUT: ALTR    -- ALTITUDE + REFRCTION IN ALTITUDE
C               RANGER  -- RANGE + REFRACTION IN RANGE
C
C       COMMON: ENVPAR  -- ENVIRONMENTAL PARAMETERS
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
        REAL NS, Q, E, LAMBDA,LAMBD2
        COMMON /ENVPAR/ BARO, TEMP, HUMID, LAMBDA
        DATA BARO, TEMP, HUMID, LAMBDA /1000., 293., 50., .5320/
C
C  CALCULATE WAVELENGTH DEPENDENCE
        LAMBD2 = LAMBDA*LAMBDA
        F = 0.965 + (0.0164+0.000228/LAMBD2)/LAMBD2
CX      F = 1.
C
        SINALT = SIN(ALT)
        Q = 1./(SINALT + SQRT(SINALT*SINALT + 0.0025284))
C
C  PARTIAL PRESSURE OF WATER VAPOR
C
        DTEMP = TEMP - 273.15
        E = 0.0611*HUMID*10.**(7.5*DTEMP/(237.3 + DTEMP))
C
C  SURFACE REFRACTIVITY
        NS = 80.343*F*(BARO-0.140647*E)/TEMP
C
C  CORRECTION FOR ALTITUDE (RADIANS)
        DALT = Q*COS(ALT)*(6.01161E-4 + 1.92056E-6*(NS - 313.))
C
C  CORRECTION FOR RANGE (NSEC)
C    WHERE THE FIRST TERM CONTAINS THE NSEC/M CONVERSION (3.335640485)
C    ...BUT FIRST, BE SURE SINALT DOESNT CAUSE PROBLEMS AT ALT = ODEG
        IF (SINALT.EQ.0) SINALT = 1.0E-6
        DRANGE = .0078621046*F*BARO/SINALT
C
X       WRITE(12,1111) ALT,DALT,RANGE,DRANGE
X1111   FORMAT(10X,4E20.10)
C  CALCULATE CORRECTED VALUES
C
        ALTR = ALT + DALT
        RANGER = RANGE + DRANGE
C
C
        RETURN
        END
        COMPILER DOUBLE PRECISION


        SUBROUTINE FNARG(FEPOK,EPOCH)

C  FNARG CALCULATES THE FUNDAMENTAL ARGUMENT FOR PRECESSION,
```

```
C  NUTATION, HELIOCENTRIC ABERRATION, AND HELIOCENTRIC
C  PARALLAX AND PLACES THEM IN A COMMON STORAGE BLOCK FINAR
C
C            INPUT: FEPOK - FIXED EPOCH IN MODIFIED JULIAN DATE
C                   EPOCH - EPOCH OF OBSERVATION
C
C            OUTPUT: NONE
C
C            COMMON BLOCKS: FUNAR
C
C            CALLS:  RVTRD
C                    MD2PI
C                    POTDC
C                    ECTRD
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

      COMMON /FUNAR/ EPSLN,ZETA,ZEE,THETA,DPSI,DETA,
     1          XS,YS,ZS,ESUN,TSAN,GELS
      COMMON /FNTEMP/ EPS(4),ZET(4),ZE,THET(4),LL(5),LLP(5),FF(5),DD(5),
     1          OM(5),PSI(24),ETA(8),AMN(5),PRTS(9),PRTC(9),SML(4),ESN(3),
     2          EQC(4),RDLS(8),SNLT,EQLN,GP1(4),GP3(4),GP4(3),
     3          EPO,CNTRY,EPTO,TCNTRY
      REAL LL,LLP,L,LP, ARGNU(12)
C     MEAN OBLIQUITY OF THE ECLIPTIC OF DATE-ES-AENA P.98
      DATA EPS / 0.4093197474, -2.271109689E-4,
     1          -2.862339973E-8, 8.779006138E-9 /
C  PRECESSION QUANTITIES - ES-AENA PP. 29-30
C  PI/2 - ZETA = RA OF ASCENDING NODE OF EQUATOR OF EPOCH T
C  RECKONED FROM EQUINOX OF TO
      DATA ZET / 1.117131925E-2, 6.767998988E-6,
     1          1.464137317E-6, 8.72664626E-8 /
C  PI/2 + ZEE = RA OF NODE RECKONED FROM EQUINOX OF T
      DATA ZE / 3.834876218E-6 /
C  THETA = INCLINATION OF EQUATOR OF T TO EQUATOR OF TO
      DATA THET / 9.718972599E-3, -4.1354607E-6,
     1          -2.065306282E-6, -2.036217461E-7 /
C  NUTATION QUANTITIES - ES-AENA P.44
C  L
      DATA LL /5.168000346, 1325., 3.470571655,
     X          1.6042E-4, 2.511E-7/
C  LP (L')
      DATA LLP /6.25658358, 99., 6.266600316,
     X          -2.62E-6,  5.818E-8/
C  F=MEAN ARGUMENT OF LATITUDE OF MOON AT EPOCH
      DATA FF / 1.963650549E-1, 1342., 1.431608937,
     1          -5.604446154E-5, -5.817764173E-9 /
C  D = MEAN ELONGATION OF MOON FROM SUN AT EPOCH
      DATA DD / 6.121523943, 1236., 5.360154261,
     1          -2.506486731E-5, 3.296733032E-8 /
C  OMEGA = LONGITUDE OF MEAN ASCENDING NODE OF MOON AT EPOCH
      DATA OM / 4.523601515, -5., -2.341219711,
     1          3.626406335E-5, 3.878509449E-8 /
C  DPSI = NUTATION IN LONGITUDE AT EPOCH
```

```
      DATA PSI / -8.354649E-5, -8.4212E-8,  1.0123E-6,   9.7E-11,
     X          -6.17119E-6,  -6.30E-10,   6.1135E-7,  -1.50E-9,
     X          -2.410E-7,     5.82E-10,   1.038E-7,   -2.4E-10,
     X           6.012E-8,     4.8E-11,   -9.8757E-7,  -9.7E-11,
     X           3.272E-7,     4.8E-11,   -1.658E-7,   -1.9E-10,
     X          -1.265E-7,     0.,        -7.224E-8,    0./
C  DETA = NUTATION IN OBLIQUITY AT EPOCH
      DATA ETA / 4.46513E-5, 4.41E-9, -4.383E-7,  1.9E-10,
     1          2.6771E-6, -1.41E-9,  4.286E-7, -2.4E-10/
C  DATA FOR COMPUTATION OF THE TRUE ANOMALY AND
C  TRUE GEOCENTRIC LONGITUDE AND POSITION OF THE SUN
C  AMAN = AVERAGE MEAN ANOMALY. P.60 APAENA XV
      DATA AMN / 6.25658358, 99., 6.266600316,
     1         -2.617993878E-6, -5.817764173E-8 /
C  PERTS = PERTURBATION ON SUN > 1ARCSEC. P. 64 APAENA XV SINES
      DATA PRTS / 2.5E-5, -2.E-5, 1.4E-5, -8.E-6,
     1         7.E-6, 7.883070455E-2, -1.3E-5, -1.3E-5, -7.E-6 /
C  PERTC = PERTURBATION ON SUN > 1 ARCSEC. P.64 APAENA XV COSINE
C  TERMS.
      DATA PRTC /1.1E-5, -2.3E-5, 9.E-6, 7.E-6,
     1 6.E-6, 1.8E-5, 1.E-5, -8.E-6, -3.5E-5/
C  SUNML = SUNS MEAN LONGITUDE . APAENA XV, P.59.
      DATA SML /4.881627934, 100., 1.342027295E-2,
     1 5.279620987E-6/
C  EQNCN = EQUATION OF CENTER FROM DANBY P.337
      DATA EQC /2., -.25, 1.25, 1.0833333333/
C  RADSN = RADIUS VECTOR OF THE SUN APAENA XV P 66.
      DATA RDLS /3.057E-5, -7.27412E-3, -9.138E-5,
     1 3.593E-6, 4.021E-6, 1.814E-5, 5.822E-6, -7.067E-6/
C  SNLAT = LATITUDE OF THE SUN APAENA XV P.65
      DATA SNLT /2.792526803E-6/
C  EQLUN = LUNAR INEQUALITY IN LONGITUDE. APAENA VI, P.18.
      DATA EQLN /3.128987498E-5/
C  GPP1 = MEAN ANOMALY OF VENUS. APAENA XV P.64.
      DATA GP1 /3.710626228, 162., 3.452328892,
     1 2.244687344E-5/
C  ESUN = ECCENTRICITY OF SUNS ORBIT APAENA VI, P.9.
      DATA ESN /1.675104E-2, -4.18E-5, -1.26E-7/
C  GPP3 = MEAN ANOMALY OF MARS. APAENA XV, P.64.
      DATA GP3 /5.576840378, 53., 1.04472791,
     1 3.156137064E-6/
C  GPP4 = MEAN ANONALY OF JUPITER APAENA XV, P.64.
      DATA GP4 /3.93288906, 8., 2.699885163/
C  INITIAL EPOCH: JAN 0.5 ET, 1900, JD2415020.0
      DATA EPO /2415020.0/
C  INITIAL EPOCH FOR TROPICAL CENTURY=1900.0 = 2415020.313
      DATA EPTO /2415020.313/
C
C  CENTURY = 36525 JULIAN(OR EPHEMERIS) DAYS.
      DATA CNTRY /36525./
C  TROPICAL CENTURY = 36524.22 DAYS
      DATA TCNTRY /36524.22/
C-----------------------------------------------------------------------
C  FIXED EPOCH IN CENTURIES SINCE EPO.
```

```
      T0 = (FEPOK-EPT0)/TCNTRY
      TT = (EPOCH-EPT0)/TCNTRY
      TT2 = TT*TT
      TT3 = TT2*TT
C
C  TOTAL CENTURIES SINCE EP0.
      T = (EPOCH-EP0)/CNTRY
      T2 = T*T
      T3 = T2*T
C
C  TOTAL CENTURIES SINCE FEPOK (FOR USE IN PRECESSION CALCULATIONS)
      TP = (EPOCH-FEPOK)/TCNTRY
      TP2 = TP*TP
      TP3 = TP2*TP
C
C  COMPUTE OBLIQUITY OF ECLIPTIC
      EPSLN = EPS(1) + EPS(2)*TT + EPS(3)*TT2 + EPS(4)*TT3
C
C  COMPUTE ZETA
      ZETA = (ZET(1) + ZET(2)*T0)*TP + ZET(3)*TP2 + ZET(4)*TP3
C  COMPUTE ZEE
      ZEE = ZETA + ZE*TP2
C  COMPUTE THETA
      THETA = (THET(1) + THET(2)*T0)*TP + THET(3)*TP2 + THET(4)*TP3
C  COMPUTE F,D,OMEGA,AMAN,SUNML,GPP1,GPP3,GPP4 - ALL ANGULAR
C  VARIABLES.
      L = LL(1) + RVTRD(LL(2),T) + LL(3)*T + LL(4)*T2 + LL(5)*T3
      CALL MD2PI(L)
      LP= LLP(1) + RVTRD(LLP(2),T) + LLP(3)*T + LLP(4)*T2 + LLP(5)*T3
      CALL MD2PI(LP)
      F = FF(1) + RVTRD(FF(2),T) + FF(3)*T + FF(4)*T2 + FF(5)*T3
      CALL MD2PI(F)
      D = DD(1) + RVTRD(DD(2),T) + DD(3)*T + DD(4)*T2 + DD(5)*T3
      CALL MD2PI(D)
      OMEGA = OM(1) + RVTRD(OM(2),T) + OM(3)*T + OM(4)*T2 + OM(5)*T3
      CALL MD2PI(OMEGA)
      AMAN = AMN(1) + RVTRD(AMN(2),T) + AMN(3)*T + AMN(4)*T2 +
     1 AMN(5)*T3
      CALL MD2PI(AMAN)
      SUNML = SML(1) + RVTRD(SML(2),T) + SML(3)*T + SML(4)*T2
      CALL MD2PI(SUNML)
      GPP1 = GP1(1) + RVTRD(GP1(2),T) + GP1(3)*T + GP1(4)*T2
      CALL MD2PI(GPP1)
      GPP3 = GP3(1) + RVTRD(GP3(2),T) + GP3(3)*T + GP3(4)*T2
      CALL MD2PI(GPP3)
      GPP4 = GP4(1) + RVTRD(GP4(2),T) + GP4(3)*T
      CALL MD2PI(GPP4)
C  COMPUTE DPSI, DETA
      ARGNU(1)  = OMEGA
      ARGNU(2)  = 2.*OMEGA
      ARGNU(3)  = 2.*(F-D+OMEGA)
      ARGNU(4)  = LP
      ARGNU(5)  = LP + ARGNU(3)
      ARGNU(6)  =-LP + ARGNU(3)
```

```
      ARGNU(7)  = ARGNU(3) - OMEGA
      ARGNU(8)  = 2.*(F+OMEGA)
      ARGNU(9)  = L
      ARGNU(10) = 2.*F + OMEGA
      ARGNU(11) = L + ARGNU(8)
      ARGNU(12) = L - 2.*D
      DPSI = 0.
      DO 5 J=1,12
          K = (J-1)*2+1
          DPSI = DPSI + (PSI(K) + PSI(K+1)*T)*SIN(ARGNU(J))
5     CONTINUE
      DETA = (ETA(1) + ETA(2)*T)*COS(ARGNU(1))
     X      + (ETA(3) + ETA(4)*T)*COS(ARGNU(2))
     X      + (ETA(5) + ETA(6)*T)*COS(ARGNU(3))
     X      + (ETA(7) + ETA(8)*T)*COS(ARGNU(8))
C  CORRECT OBLIQUITY OF ECLIPTIC FOR NUTATION
      EPSLN = EPSLN+DETA
C  COMPUTE GEOMETRIC LONGITUDE OF SUN
C  1) COMPUTE SIN PERTURBATIONS TO LONGITUDE
      SINP = PRTS(1)*SIN(4.*AMAN - 8.*GPP3 + 3.*GPP4)
     1  + PRTS(2)*SIN(AMAN-GPP1) + PRTS(3)*SIN(2.*AMAN-2.*GPP1)
     2  + PRTS(4)*SIN(3.*AMAN-2.*GPP1) + PRTS(5)*SIN(13.*AMAN
     3  - 8.*GPP1+PRTS(6)*T) + PRTS(7)*SIN(GPP4)
     4  + PRTS(8)*SIN(2.*AMAN-2.*GPP4) + PRTS(9)*SIN(AMAN-GPP4)
C  2) COMPUTE COS PERTURBATIONS TO LONGITUDE
      COSP = PRTC(1)*COS(AMAN-GPP1) + PRTC(2)*COS(2.*AMAN-2.*GPP1)
     1  + PRTC(3)*COS(3.*AMAN-2.*GPP1) + PRTC(4)*COS(4.*AMAN-3.*GPP1)
     2  + PRTC(5)*COS(13.*AMAN-8.*GPP1+PRTS(6)*T)
     3  + PRTS(6)*COS(4.*AMAN-8.*GPP3+3.*GPP4)
     4  + PRTC(7)*COS(2.*AMAN-2.*GPP3) + PRTC(8)*COS(AMAN-2.*GPP3)
     5  + PRTC(9)*COS(AMAN-GPP4)
C  3) COMPUTE ECCENTRICITY OF SUN
      ESUN = ESN(1) + ESN(2)*T + ESN(3)*T2
C  4) COMPUTE LUNAR INEQUALITY IN LONGITUDE
      EQLUN = EQLN*SIN(D)
C  5) COMPUTE MEAN ANOMALY W/ PERTURBATION & NUTATION
      TAMAN = AMAN + EQLUN + DPSI
      CALL MD2PI(TAMAN)
C
C     COMPUTEUTE THE GEOMETRIC LONGITUDE W/ PERTURBATION
      GELS = SUNML + EQNCN + EQLUN + SINP + COSP + DPSI
      CALL MD2PI(GELS)
C
C  COMPUTE TRUE ANOMALY OF SUN
      TSAN = TAMAN + EQNCN
      CALL MD2PI(TSAN)
C  COMPUTE LATITUDE(GEOCENTRIC) OF SUN
      SNLAT = SNLT*SIN(F)
C  COMPUTE RADIUS VECTOR TO SUN
      RADSL = RDLS(1) + RDLS(2)*COS(AMAN) + RDLS(3)*COS(2.*AMAN)
     1  + RDLS(4)*COS(2.*AMAN-2.*GPP1) + RDLS(5)*COS(2.*AMAN-2.*GPP3)
     2  + RDLS(6)*COS(AMAN) + RDLS(7)*SIN(2.*AMAN-2.*GPP1)
     3  + RDLS(8)*SIN(AMAN-GPP4)
      RADSN = 10.**RADSL
```

```
C  COMPUTE GEOCENTRIC EQUATORIAL CARTESIAN COORDINATES IN A.U.
C  OF SUN
      IHAND = 1
      CALL POTDC(IHAND,GELS,SNLAT,X,Y,Z)
      CALL ECTRD(X,Y,Z,EPSLN,X,Y,Z)
      XS = X*RADSN
      YS = Y*RADSN
      ZS = Z*RADSN
      RETURN
      END



      COMPILER DOUBLE PRECISION
      SUBROUTINE DIABR(X,Y,Z,ST,LAT,XP,YP,ZP)

C  DIABR CORRECTS POSITION IN EQUATORIAL COORDIANTES FOR EFFECTS OF
C  DIURNAL ABERRATION.
C  NOTE: SIMPLE SERIES IS USED DUE TO SMALL SIZE OF CORRECTION
C
C  INPUT:
C       X,Y,Z - DIRECTION COSINES OF EQUATORIAL COORDS.
C       ST    - LOCAL SIDEREAL TIME IN RADIANS
C       LAT   - LATITUDE OF SITE IN RADIANS
C  OUTPUT:
C       X,Y,Z - CORRECTED EQUATORIAL DIRECTION COSINES.
C
C  REFERENCE: ESAENA, P.49.
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ RLR 1/81
      REAL LAT
      COMMON /MATH/ FILL(5),RDPAS,RDPTS
      CALL DCTPO(1,X,Y,Z,RA,DEC)
      HA = ST-RA
      COSLAT= COS(LAT)
      DRA = 0.0213*COSLAT*COS(HA)/COS(DEC)
      DDEC= 0.320 *COSLAT*SIN(HA)*SIN(DEC)
      RA = RA + DRA*RDPTS
      DEC= DEC+ DDEC*RDPAS
      CALL POTDC(1,RA,DEC,XP,YP,ZP)
      RETURN
      END



     COMPILER DOUBLE PRECISION
     SUBROUTINE HEAB8(X,Y,Z,ICTFG,XP,YP,ZP)
C CORRECTS THE DIRECTION COSINES OF A POSITION IN EQUATORIAL
C    COORDINATES FOR THE EFFECTS OF HELIOCENTRIC ABERRATION.
C X,Y,Z-POSITION IN EQUATORIAL COORDINATES.
C ICTFG-CATALOG FLAG-IF >0 POSITION FROM A CATALOG,I.E. NO E TERMS.
C XP,YP,ZP-CORRECTED POSITION IN EQUATORIAL COORDINATES.
C CALLS:SYSTEM:COS,SIN,ATAN2,SQRT;USER:RDTEC,ROTAZ,ROTAX,ECTRD
C USES COMMON BLOCK FUNAR.
```

```
C DOCUMENTATION:WOOLARD AND CLEMENCE.
C RADIUS IS CONSERVED.
C FNARG MUST BE CALLED PRIOR TO THE USE OF HEAB8 TO SET THE
C    FUNDAMENTAL ARGUMENTS TSUN,ESUN,EPSLN,GELS.
      COMMON/FUNAR/EPSLN,ZETA,ZEE,THETA,DPSI,DETA,XS,YS,ZS,ESUN,TSUN,
     1 GELS
C TSUN-TRUE ANOMALY OF SUN IN RADIANS.
C ESUN-ECCENTRICITY OF SUNS ORBIT
C EPSLN-OBLIQUITY OF THE ECLIPTIC IN RADIANS.
C GELS-GEOMETRIC LONGITUDE OF THE SUN IN RADIANS
C KAP IS THE CIRCULAR VELOCITY OF THE EARTH /SPEED OF LIGHT/SQRT(U-E**2)
      COMMON /MATH/ PIO2,PI,TWOPI,FILL(4)
      COMMON /H8COMM/ KAP
      REAL KAP, K
      DATA KAP /9.935346999E-5/
C COMPUTE THE LONGITUDE OF THE APEX.
      CSTA=COS(TSUN)
      APXL=PI+GELS+ATAN2((1.+ESUN*CSTA),(ESUN*SIN(TSUN)))
      K=KAP/SQRT(1.-ESUN*ESUN)
C CHECK  THE CATALOG FLAG.
      ETRM=0.
      IF(ICTFG.LE.0)ETRM=ESUN
C MOVE  X AXIS TO THE APEX OF THE MOTION.
      CALL RDTEC(X,Y,Z,EPSLN,XP,YP,ZP)
      CALL ROTAZ(XP,YP,ZP,APXL,XP,YP,ZP)
C GET THE POSITION AND THE X AXIS IN THE X-Y PLANE
      APLT=ATAN2(ZP,YP)
      CALL ROTAX(XP,YP,ZP,APLT,XP,YP,ZP)
C COMPUTE THE CORRECTION TO SECOND ORDER WITH ETERMS IF NEEDED.
      TETA=ATAN2(YP,XP)
      SNTH=SIN(TETA)
      VC2=(1.+ETRM*ETRM-2.*ETRM*COS(TSUN))*K*K
      DTET=SQRT(VC2)*SNTH-VC2*SNTH*COS(TETA)
C MAKE THE CORRECTION.
      CALL ROTAZ(XP,YP,ZP,DTET,XP,YP,ZP)
C GET BACK TO EQUATORIAL COORDINATES.
      APLTN=-APLT
      CALL ROTAX(XP,YP,ZP,APLTN,XP,YP,ZP)
      APXLN=-APXL
      CALL ROTAZ(XP,YP,ZP,APXLN,XP,YP,ZP)
      CALL ECTRD(XP,YP,ZP,EPSLN,XP,YP,ZP)
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE ECTRD(X,Y,Z,EPS,XP,YP,ZP)
C CONVERTS THE DIRECTION COSINES OF A POSITION IN THE ECLIPTIC
C    COORDINATES SYSTEM TO THE DITECTION COSINES OF THE POSITION IN THE
C    EQUATORIAL COORDINATE SYSTEM.
C X,Y,Z-POSITION IN ECLIPTIC COORDINATES.
C EPS-OBLIQUITY OF THE ECLIPTIC OF DATE IN RADIANS.
C XP,YP,ZP-POSITION IN EQUATORIAL COORDINATES.
```

```
C CALLS:USER:ROTAX.
C DOCUMENTATION:EICHHORN
C RADIUS IS CONSERVED.
      EPSN=-EPS
      CALL ROTAX(X,Y,Z,EPSN,XP,YP,ZP)
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE RDTEC(X,Y,Z,EPS,XP,YP,ZP)
C CONVERTS THE DIRECTION COSINES OF A POSITION IN THE EQUATORIAL
C    COOORDINATES SYSTEM TO THE DIRECTION COSINES OF THE POSITION
C    IN THE ECLIPTIC COORDINATE SYSTEM.
C X,Y,Z-POSITION IN EQUATORIAL COORDINATES.
C EPS-OBLIQUITY OF THE ECLIPTIC OF DATE IN RADIANS.
C XP,YP,ZP-POSITION IN ECLIPTIC COORDINTAES.
C CALLS:USER:ROTAX
C DOCUMENTATION: EICHHORN
C RADIUS IS CONSERVERED.
      CALL ROTAX(X,Y,Z,EPS,XP,YP,ZP)
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE PRENU(X,Y,Z,XP,YP,ZP)
C APPLIES THE CORRECTION FOR PRECESSION AND NUTATION TO THE DIRECTION
C    COSINES OF A POSITION IN EQUATORIAL COORDINATES.
C X,Y,Z-POSITION IN EQUATORIAL COORDINATES.
C CXP,YP,ZP-CORRECTED POSITION IN EQUATORIAL COORDINATES.
C CALLS:USER:ROTAZ,ROTAY,ROTAX.
C CONTAINS COMMON BLOCK FUNAR.
C DOCUMENTATION: ES-AENA.
C FNARG MUST BE CALLED PRIOR TO THE CALL TO PRENU TO SET THE
C    FUNDAMENTAL ARGUMENTS EPSLN,ZETA,ZEE,THETA,DPSI,DETA.
C RADIUS IS CONSERVED.
      COMMON /FUNAR/ EPSLN,ZETA,ZEE,THETA,DPSI,DETA,XS,YS,ZS,ESUN,TSAN,G
     CELS
C EPSLN-OBLIQUITY OF THE ELIPTIC OF DATE IN RADIANS.
C ZETA,ZEE,THETA-PRECESSIONAL QUATNTIES IN RADIANS
C DPSI-NUTATION IN LONGITUDE IN RADIANS.
C DETA-NUTATION IN OBLIQUITY IN RADIANS.
C CORRECT FOR PRECESSION.
      ZETAN=-ZETA
      ZAN=-ZEE
      CALL ROTAZ(X,Y,Z,ZETAN,XP,YP,ZP)
      CALL ROTAY(XP,YP,ZP,THETA,XP,YP,ZP)
      CALL ROTAZ(XP,YP,ZP,ZAN,XP,YP,ZP)
C CORRECT FOR NUTATION.
      DEPS=DETA-EPSLN
      CALL ROTAX(XP,YP,ZP,-DEPS,XP,YP,ZP)
```

```
      CALL ROTAZ(XP,YP,ZP,-DPSI,XP,YP,ZP)
      CALL ROTAX(XP,YP,ZP,-EPSLN,XP,YP,ZP)
      RETURN
      END


        COMPILER DOUBLE PRECISION
        SUBROUTINE PRMTN(RA,DEC,PMR,PMD,PMPR,PMPD,DT,RAP,DECP)

C  PROPER MOTION CALCULATIONS
C  INPUT:
C       RA,DEC  -- RADIANS
C       RMR,PMPR-- PM & DPM/DT IN RADIANS
C       PMD,PMPD-- PM & DPM/DT IN RADIANS
C       DT      -- FRACTION OF A TROPICAL CENTURY SINCE EPOCH OF CATALOG
C                       (TROPICAL CENTURY = 36524.22 DAY)
C  OUTPUT:
C       RAP,DECP-- CORRECTED POSITIONS IN RADIANS
C
C  REFERENCE: VAN DE KAMP, ELEMENTS OF ASTROMECHANICS
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ RLR 1/81

        DRA = -(PMR+PMPR*DT)*DT
        DDEC=  (PMD+PMPD*DT)*DT
        CALL OFSET(RA,DEC,DRA,DDEC,RAP,DECP)
        RETURN
        END


        COMPILER DOUBLE PRECISION
        SUBROUTINE OFSET (AZ,AL,DAZ,DAL,AZP,ALP)

C       OFSET CHANGES A POSITION AZ,AL BY ROTATING AMOUNTS DAZ,DAL
C       ABOUT THE AZIMUTH AND ITS SPECIFIC ALTITIUDE AXES RESPECTIVELY.
C       THE FINAL POSITION AZP,ALP IS ALSO POLAR ALT-AZIMUTH COORDINATES.
C
C       MARK A. POWELL - OCTOBER 28, 1976       COPY BY SJB 7/25/80


        IHAND=1
        CALL POTDC(IHAND,AZ,AL,X,Y,Z)
        CALL ROTAZ(X,Y,Z,AZ,X,Y,Z)
        CALL ROTAY(X,Y,Z,DAL,X,Y,Z)
        DDAZ=DAZ-AZ
        CALL ROTAZ(X,Y,Z,DDAZ,X,Y,Z)
        CALL DCTPO(IHAND,X,Y,Z,AZP,ALP)
        RETURN
        END
```

```
        COMPILER DOUBLE PRECISION
        SUBROUTINE HEPAR(X,Y,Z,PAR,XP,YP,ZP)
C
C       CORRECT A STAR'S POSITION FOR THE EFFECTS OF PARALLAX.
C
C               INPUT:  X,Y,Z    - POSITION IN DIRECTION COSINES OF
C                                  EQUATORIAL COORDINATE SYSTEM.
C                       PAR      - TOTAL PARALLAX IN RADIANS.
C
C               OUTPUT: XP,YP,ZP- CORRECTED POSITIONS.
C
C               COMMON: FUNAR    -FUNDAMENTAL ARGUMENTS.
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
        COMMON /FUNAR/ EPSLN,ZETA,ZEE,THETA,DPSI,DETA,
     1          XS,YS,ZS,ESUN,TSAN,GELS
        DIS = COS(PAR)/SIN(PAR)
        XP = X*DIS + XS
        YP = Y*DIS + YS
        ZP = Z*DIS + ZS
        RAD = SQRT(XP*XP + YP*YP + ZP*ZP)
        XP = XP/RAD
        YP = YP/RAD
        ZP = ZP/RAD
        RETURN
        END


      COMPILER DOUBLE PRECISION
      SUBROUTINE ALTRD(X,Y,Z,LAT,LST,XP,YP,ZP)
C CONVERTS THE DIRECTION COSINES OF A POSITION IN THE ALTITUDE,AZIMUTH
C    COORDINATE SYSTEM TO THE DIRECTION COSINES OF THE POSITION IN
C    EQUATORIAL COORDINATES.
C X,Y,Z-POSITION IN ALT-AZIMUTH COORDINATES.
C LAT-GEOCENTRIC LATITUDE OF THE OBSERVER IN RADIANS.
C LST-LOCAL SIDEREAL TIME IN RADIANS
C XP,YP,ZP-POSITION IN EQUATORIAL COORDINATES.
C CALLS:USER:ALTHD,HDTRD
C DOCUMENTATION:EICHHORN
C RADIUS IS CONSEREVED
      REAL LAT,LST
      CALL ALTHD(X,Y,Z,LAT,XP,YP,ZP)
      CALL HDTRD(XP,YP,ZP,LST,XP,YP,ZP)
      RETURN
      END

      COMPILER DOUBLE PRECISION
      SUBROUTINE RDTAL (X,Y,Z,LAT,LST,XP,YP,ZP)
C CONVERTS THE DIRECTION COSINES OF A POSITION IN THE EQUATORIAL
C    COORDINATE SYSTEM TO THE DIRECTION COSINES IN ALTITUDE,AZIMUTH
C    COORDINATES.
C X,Y,Z-POSITION IN EQUATORIAL COORDINATES.
C LAT-OBSERVERS GEOCENTRIC LATITUDE IN RADIANS.
```

```
C LST-LCOAL SIDEREAL TIME IN RADIANS.
C XP,YP,ZP-POSITION IN ALTITUDE,AZIMUTH COORDINATES.
C CALLS:USER:RDTHD,HDTAL
C DOCUMENTATION:EICHHORN
C RADIUS IS CONSERVERED.
      REAL LAT,LST
      CALL RDTHD(X,Y,Z,LST,XP,YP,ZP)
      CALL HDTAL(XP,YP,ZP,LAT,XP,YP,ZP)
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE ALTHD(X,Y,Z,LAT,XP,YP,ZP)
C
C    CONVERTS THE DIRECTION COSINES OF A SET OF ALT-AZIMUTH COORDINATES
C      TO THE DIRECTION COSINES OF THE HOUR ANGLE - DECLINATION
C      COORDINATES.   MARK A. POWELL - OCTOBER 28, 1976
C
C      LAT IS THE LATITUDE (GEOCENTRIC) OF THE OBSERVER.
C
      REAL LAT
      COMMON /MATH/ PIO2,PI,TWOPI, FILL(4)
      COLAT=LAT-PIO2
      CALL ROTAZ(X,Y,Z,PI,XP,YP,ZP)
      CALL ROTAY(XP,YP,ZP,COLAT,XP,YP,ZP)
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE HDTAL(X,Y,Z,LAT,XP,YP,ZP)
C
C    CONVERTS THE DIRECTION COSINES OF A SET OF HOUR ANGLE - DECLINATION
C      COORDINATES TO THE DIRECTION COSINES OF THE ALT-AZIMUTH
C      COORDINATES.   MARK A. POWELL - OCTOBER 28, 1976
C
C      LAT IS THE LATITUDE (GEOCENTIRC) OF THE OBSERVER.
C
      REAL LAT
      COMMON /MATH/ PIO2,PI,TWOPI,FILL(4)
      COLAT=PIO2-LAT
      CALL ROTAY(X,Y,Z,COLAT,XP,YP,ZP)
      CALL ROTAZ(XP,YP,ZP,-PI,XP,YP,ZP)
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE HDTRD(X,Y,Z,LST,XP,YP,ZP)
C     CONVERT THE DIRECTION COSINES OF A POSITION IN THE HOUR ANGLE, DECLIN-
```

```
C       ATION COORDINATE SYSTEM TO TH DIRECTION COSINES OF THE POSITION IN
C       THE EQUATORIAL COORDINATE SYSTEM.
        REAL LST
        YT = -Y
        CALL ROTAZ(X,YT,Z,-LST,XP,YP,ZP)
        RETURN
        END
```

```
      COMPILER DOUBLE PRECISION
      SUBROUTINE RDTHD(X,Y,Z,LST,XP,YP,ZP)
C CONVERTS THE DIRECTION COSINES OF A POSITION IN THE EQUATORIAL COOR-
C    DINATES SYSTEM TO THE DIRECTION COSINES OF THE POSITION IN
C    H.A.,DECLINATION COORDINATES.
C X,Y,Z-POSITION IN EQUATORIAL COORDINATES
C LST-LOCAL SIDEREAL TIME IN RADIANS.
C XP,YP,ZP-POSITION IN H.A.,DEC. COORDINATES
C CALLS: USER: ROTAZ
C DOCUMENTATION: EICHHORN
C RADIUS IS CONSERVERED.
      REAL LST
      CALL ROTAZ(X,Y,Z,LST,XP,YP,ZP)
      YP=-YP
      RETURN
      END
```

```
      COMPILER DOUBLE PRECISION
      SUBROUTINE POTDC(IHAND,XL,XC,X,Y,Z)
C
C    CONVERTS POLAR COORDINATES TO DIRECTION COSINES. HANDEDNESS OF THE
C      COORDINATES SYSTEM IS CHANGED IF IHAND{0.
C    MARK A. POWELL - OCTOBER 28, 1976
C
      CSXC=COS(XC)
      X=CSXC*COS(XL)
      Y=CSXC*SIN(XL)
      Z=SIN(XC)
      IF (IHAND.LT.0) Y=-Y
      RETURN
      END
```

```
      COMPILER DOUBLE PRECISION
      SUBROUTINE DCTPO(IHAND,X,Y,Z,XL,XC)
C
C    CONVERTS DIRECTION COSINES TO POLAR COORDINATES. HANDEDNESS OF THE
C      POLAR COORDINATES IS CHANGED IF IHAND{0.
C    MARK A. POWELL - OCTOBER 28, 1976
C
      XL=ATAN2(Y,X)
```

```
      R=SQRT(X*X+Y*Y)
      XC=ATAN2(Z,R)
      IF (IHAND.LT.0) XL=-XL
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE RDTDG(X,ID,IM,XS)
C
C    CONVERT RADIANS TO DEGREES, ARCMINUTES AND ARCSECONDS.
C    MARK A. POWELL - OCTOBER 28, 1976
C
      COMMON /MATH/ FILL1(3), RDPDG, FILL2(3)
      XS=X/RDPDG
      ID=IFIX(XS)
      XS=ABS((XS-FLOAT(ID))*60.)
      IM=IFIX(XS)
      XS=(XS-FLOAT(IM))*60.
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE RDTHR(X,IH,IM,XS)
C
C    CONVERT RADIANS TO HOURS,MINUTES AND SECONDS.
C    MARK A. POWELL - OCTOBER 28, 1976
C
      COMMON /MATH/ FILL1(4),RDPHR,FILL2(2)
      XS=X/RDPHR
      IH=IFIX(XS)
      XS=ABS((XS-FLOAT(IH))*60.)
      IM=IFIX(XS)
      XS=(XS-FLOAT(IM))*60.
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE MD2PI(X)
C DOES MODULO 2 PI FOR X
C X-ANGLE IN RADIANS.
C CALLS:SYSTEM:ATAN2,ABS,SIGN.
      COMMON /MATH/ PIO2,PI,TWOPI,FILL(4)
    1 IF(ABS(X).LT.TWOPI) GO TO 2
      X=X-SIGN(TWOPI,X)
      GO TO 1
    2 RETURN
      END
```

```
      COMPILER DOUBLE PRECISION
      FUNCTION RVTRD(R,X)
C CONVERTS MANY REVOLUTIONS TO MODULO 2 PI.
C R-NUMBER OF REVOLUTIONS.
C X-MULTIPLIER OF REVOLUTINONS.
C CALLS:SYSTEM:ATAN2,AINT
C DOCUMENTATION:NONE
C DESIGNED FOR USE WITH FNARG.
      COMMON /MATH/ PIO2,PI,TWOPI,FILL(4)

      RV=R*X
      FR=RV-AINT(RV)
      RVTRD=FR*TWOPI
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE ROTAX(X,Y,Z,ANGLE,XP,YP,ZP)
C
C     ROTATE ANGLE AMOUNT ABOUT  X AXIS.
C     MARK A. POWELL - OCTOBER 28, 1976
C
      CSA=COS(ANGLE)
      SNA=SIN(ANGLE)
      YS=Y
      XP=X
      YP=Y*CSA+Z*SNA
      ZP=Z*CSA-YS*SNA
      RETURN
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE ROTAY(X,Y,Z,ANGLE,XP,YP,ZP)
C
C     ROTATE ANGLE AMOUNT ABOUT Y AXIS.
C     MARK A. POWELL - OCTOBER 28, 1976
C
      CSA=COS(ANGLE)
      SNA=SIN(ANGLE)
      ZS=Z
      YP=Y
      ZP=Z*CSA+X*SNA
      XP=X*CSA-ZS*SNA
      RETURN
      END
```

```
      COMPILER DOUBLE PRECISION
      SUBROUTINE ROTAZ(X,Y,Z,ANGLE,XP,YP,ZP)
C
C    ROTATE ANGLE AMOUNT ABOUT Z AXIS.
C    MARK A. POWELL - OCTOBER 28, 1976
C
      CSA=COS(ANGLE)
      SNA=SIN(ANGLE)
      XS=X
      ZP=Z
      XP=X*CSA+Y*SNA
      YP=Y*CSA-XS*SNA
      RETURN
      END
```

```
SITE:
         LONG:      0.000000
         GDLAT:     0.00000
         LAT:       0.00000

ENVIRONMENT:
         TEMP:   273.
         PRES:   800.

TIME (JD):  2444446.800000

FUNDAMENTAL ARGUMENTS (RADIANS AND UNITLESS QUANTITIES):
         OBLIQUITY OF ECLIPTIC:                 0.4091006810D  0
         PRECESS (ZETA):                        0.3415916548D -2
         PRECESS (ZEE=RA OF NODE):              0.3416274858D -2
         PRECESS (THETA=INCL OF EQUATOR):       0.2969972493D -2
         NUTATE (D PSI):                       -0.4819099457D -4
         NUTATE (D ETA):                       -0.3607581390D -4
         GEO X OF SUN:                         -0.5465765127D  0
         GEO Y OF SUN:                          0.7853086957D  0
         GEO Z OF SUN:                          0.3404819428D  0
         ECC OF SUN:                            0.1671728155D -1
         TRUE ANOM OF SUN:                      0.3505416968D  1
         GEOM LONG OF SUN:                      0.2139092112D  1

STAR INFO:
(RA,DEC IN DEG,HOURS; CORRECTIONS IN SEC OF ARC,TIME)
         RA:     6.3806925
         DEC: -52.6676250
         MU RA:       0.290          MU DOT RA:      0.004
         MU DEC:      2.220          MU DOT DEC:    -0.040

         CATALOG EPOCH (JD):  2433282.423000

LOCAL SIDEREAL TIME (HOURS):   3.4757140

RESULTS OF PROPER MOTION:
```

```
    NEW RA,DEC:    6.3807172   -52.667438
     DRA,DDEC:      0.09           0.67

RESULTS OF PRECESSION & NUTATION:
    NEW RA,DEC:    6.3919567   -52.686564
    DRA,DDEC:     40.46        -68.86

RESULTS OF HELIOCENTRIC ABERRATION:
    NEW RA,DEC:    6.3914018   -52.683962
    DRA,DDEC:     -2.00           9.37

RESULTS OF DIURNAL ABERRATION:
    NEW RA,DEC:    6.3914089   -52.683914
    DRA,DDEC:      0.03           0.18
    CON RA,DEC:    6 23 29.072      -52 41   2.089

COMPARISON POSITION (W/O DIURNAL ABERRATION):
                   6 23 29. 59      -52 40 61.970

RESULTS OF REFRACTION:
    AZ,ALT:  152.212627  25.977116
    REFRACTED ALT:   26.004286
    DALT:    97.81


SITE:
         LONG:      0.000000
         GDLAT:     0.00000
         LAT:       0.00000

ENVIRONMENT:
         TEMP:   273.
         PRES:   800.

TIME (JD):  2444446.800000

FUNDAMENTAL ARGUMENTS (RADIANS AND UNITLESS QUANTITIES):
         OBLIQUITY OF ECLIPTIC:                  0.4091006810D  0
         PRECESS (ZETA):                         0.3415916548D -2
         PRECESS (ZEE=RA OF NODE):               0.3416274858D -2
         PRECESS (THETA=INCL OF EQUATOR):        0.2969972493D -2
         NUTATE (D PSI):                        -0.4819099457D -4
         NUTATE (D ETA):                        -0.3607581390D -4
         GEO X OF SUN:                          -0.5465765127D  0
         GEO Y OF SUN:                           0.7853086957D  0
         GEO Z OF SUN:                           0.3404819428D  0
         ECC OF SUN:                             0.1671728155D -1
         TRUE ANOM OF SUN:                       0.3505416968D  1
         GEOM LONG OF SUN:                       0.2139092112D  1

STAR INFO:
(RA,DEC IN DEG,HOURS; CORRECTIONS IN SEC OF ARC,TIME)
         RA:   13.7595261
         DEC:  49.5622583
```

```
        MU RA:      -1.285          MU DOT RA:       0.014
        MU DEC:     -1.410          MU DOT DEC:     -0.080

        CATALOG EPOCH (JD):  2433282.423000

LOCAL SIDEREAL TIME (HOURS):   3.4757140

RESULTS OF PROPER MOTION:
    NEW RA,DEC:  13.7594174   49.562137
     DRA,DDEC:     -0.39        -0.44

RESULTS OF PRECESSION & NUTATION:
    NEW RA,DEC:  13.7793077   49.411814
    DRA,DDEC:    71.61      -541.16

RESULTS OF HELIOCENTRIC ABERRATION:
    NEW RA,DEC:  13.7792679   49.416826
    DRA,DDEC:    -0.14        18.04

RESULTS OF DIURNAL ABERRATION:
    NEW RA,DEC:  13.7792596   49.416797
    DRA,DDEC:    -0.03        -0.10
    CON RA,DEC:  13 46 45.335      49 25  0.470

COMPARISON POSITION (W/O DIURNAL ABERRATION):
                 13 46 45.337      49 25  0.570

RESULTS OF REFRACTION:
    AZ,ALT:   20.206438 -35.975412
    REFRACTED ALT:  -25.968247
    DALT: 36025.80
```

9 APPENDIX B

```
      COMPILER DOUBLE PRECISION

C  THIS IS A TEST DRIVER FOR THE NON-LINEAR LEAST-SQUARES SUB-SYSTEM.
C  THE INPUT DATA IS IN AN ABBREVIATED FORMAT: MLRS DATA INCLUDES A
C  HEADER LINE, STAR NUMBER,OPTICAL PATH, TIME OF OBSERVATION,
C  AND PRE-FIT AND POST-FIT RESIDUALS.
C
C  INPUT (ALL ANGULAR VARIABLES ARE IN RADIANS)
C      MODEL    -- MODEL NUMBER TO BE USED INN "FUNC".
C                  DIFFERENT NUMBERS REPRESENT DIFFERENT LIGHT PATHS OR
C                  EXPERIMENNTAL MODELS.
C      OLON,OLAT- OBSERVED TARGET POSITION IN COORDINATE SYSTEM LATITUDINAL
C                  AND LONGITUDINAL AXES (E.G., AZ/ALT, X/Y). THESE ARE
C                  RAW ENCODER READINGS CONVERTED TO RADIANS.
C      CLON,CLAT- CALCULATED TARGETS POSITIONS.
C      NOBS     -- NUMBER OF POSITIONS OBSERVED.
C      SOLVEC--LIST OF PARAMETERS (BY NUMBER) TO BE IN SOLUTION (IN ANY ORDER)
C      NCOEF    -- NUMBER OF PARAMETERS IN SOLUTION.
C      ORIRAD   -- INITIAL ORIENTATION PARAMETERS IN RADIANS. THESE ARE
C                  USED AS "FIRST GUESS" BY FITTING ROUTINES.
C
C  OUTPUT
C      ORIRAD   -- FINAL ORIENTATION PARAMETER VALUES.
C      OCLONP,  -- FINAL OBSERVED-CALCULATED RESIDUALS (POST-FIT)
C      OCLATP
C
C      STANDARD DEVIATION OF EACH ITERATION TO THE FIT IS WRITTEN TO
C      THE OPERATOR CONSOLE, AND A FINAL SUMMARY IS WRITTEN TO LINE PRINTER.
C
C  PERIPHERAL DEVICES ASSUMED:
C      UNIT 8  -- INPUT DATA FILE
C      UNIT 10 -- CONSOLE OUTPUT
C      UNIT 11 -- CONSOLE INPUT
C      UNIT 12 -- LINE PRINTER OUTPUT
C
C  LOAD THE FOLLOWING ROUTINES:
C      LSQDEM,BDMATH,BDPARM,LSOLN,LSINT,DNORMEQ,SOLCOR,FUNC
C      RECAP & MATRIX INVERSION ROUTINE
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ RLR 10/81
      COMMON /OBSVTN/ MODEL(200),OLON(200),OLAT(200), CLON(200),CLAT(200),
     X                OCLON(200),OCLAT(200), OCLONP(200),OCLATP(200),NOBS
      COMMON /OBSINF/ HDRLIN(40), NSTAR(200),IHR(200),IMIN(200),ISEC(200)
      INTEGER HDRLIN,NSTAR,IHR,IMIN,ISEC
      INTEGER DATNAM(7)
      COMMON /HSCMN/ SOLVEC(20),NCOEF
      INTEGER SOLVEC,NCOEF
      COMMON /TELCOR/ ORFILL(20), ORIRAD(20), NP
      DATA SOLVEC /1,2,3,4,5,6,7,8,12*0/, NCOEF /8/
      DATA ORIRAD /20*0./
      DATA MODEL /20*1/
C-----------------------------------------------------------------------
```

```
C  OPEN REPORT FILE FOR TEST
       CALL OPEN(12,"LSQOP",2,IERR)
C  GET THE DATA FILE NAME
5      WRITE(10,1000)
1000   FORMAT("  DATA FILE NAME?: ", Z)
       READ(11,1010) DATNAM
1010   FORMAT(7A2)
       IF (DATNAM(1).NE."  ") GO TO 25
C
9      WRITE(10,1001)
1001   FORMAT(8X, "INAPPROPRIATE RESPONSE")
       GO TO 5
C
C  OPEN THE DATA FILE
25     CALL OPEN(8,DATNAM,2,IERR)
         IF (IERR.NE.1) CALL ERROR(5240,IERR)
       NOBS = 1
       REWIND 8
C  READ THE POINTING DATA
50     READ(8,1020,END=100) OLON(NOBS),OLAT(NOBS),CLON(NOBS),CLAT(NOBS)
1020   FORMAT(4(E12.8,2X))
       NOBS= NOBS+ 1
       IF (NOBS.GT.200) GO TO 100
       GO TO 50
C
C  CLOSE OUT THE DATA FILE
100    NOBS= NOBS- 1
C  SOLVE FOR COEFFICIENTS WHOSE NUMBERS ARE IN SOLVEC
C  NOTE: SOLVEC IS NORMALLY OPERATOR SELECTED FOR FLEXIBILITY.
       CALL LSOLN(SOLVEC,NCOEF,NPASS)
C  PRINT THE RESULTS AND CLOSE IT ALL UP.
       CALL RECAP(-NPASS)
       CALL FCLOS(12)
       CALL FCLOS(8)
       STOP
       END

       COMPILER DOUBLE PRECISION
       BLOCK DATA
C  ** BDMATH **
C                BLOCK DATA FOR MATH COMMON
C
       COMMON /MATH/ PIO2,PI,TWOPI,RDPDG,RDPHR,RDPAS,RDPTS
C
C  ** MATH COMMON **
       DATA PIO2  /1.570796326794897/  ;PI/2
       DATA PI    /3.141592653589793/  ;PI
       DATA TWOPI /6.283185307179586/  ;2*PI
       DATA RDPDG /1.745329251994330E-2/;RADS/DEGREE
       DATA RDPHR /2.617993877991494E-1/;RADS/HOUR
       DATA RDPAS /4.848136811      E-6/;RADS/ARC SEC
       DATA RDPTS /7.272205217      E-5/;RADS/TIME SEC
C
       END
```

```
      BLOCK DATA

C  ** BDPARM **
C  BLOCK DATA FOR ORIENTATION PARAMETERS
      COMMON /LABELS/ ORILIS(20,20),MMM,NNN
      INTEGER ORILIS
      COMMON /ORILIM/ ORLO(20), ORHI(20)
      COMMON /TELCOR/ ORIENT(20), OPARMS(20), NPARM
C
      DATA MMM,NNN /20,20/
      DATA ORILIS
     1 /"1 AZIMUTH AXIS ENCODER OFFSET          "
     2 ,"2 ALTITUDE AXIS ENCODER OFFSET         "
     3 ,"3 X AXIS ROTATION COMPONENT            "
     4 ,"4 Y AXIS ROTATION COMPONENT            "
     6 ,"5 TRANSVERSE MISALIGNMENT              "
     8 ,"6 LATERAL MISALIGNMENT                 "
     7 ,"7 AZ & ALT AXIS NONORTHOGONALITY       "
     8 ,"8 LATCH FLEXURE                        "
     5 ,"9 ALTITUDE ENCODER WOBBLE              "
     9 ,"10 ALTITUDE ENCODER PHASE (ENTER ONLY) "
     1 ,"11 AZIMUTH STOW POSITION               "
     2 ,"12 ALTITUDE STOW POSITION              "
     X /
C
      DATA ORLO /-360., -105., 7*-2., -11., -360., -105./
C
      DATA ORHI /+360.,  -75., 7*+2., +11., +360.,  +90./
C
      DATA NPARM /10/
      END


      COMPILER DOUBLE PRECISION
      SUBROUTINE LSOLN(ISOLVEC,ICOEF,NPASS)
C
C       ROUTINE NSOLN CONTROLS THE NON-LINEAR LEAST
C  SQUARES SOLUTION FOR THE TELESCOPE ORIENTATION PARAMETERS USING
C  CALCULATED AND OBSERVED POSITIONS OF <= 200 STARS OR GROUND TARGETS
C  THE PROGRAM ITERATES UNTIL THE NEW CORRECTION TO THE
C  ORIENTATION PARAMETERS ARE "SUFFICIENTLY SMALL" -- 1.E-6 RADIANS, NOW.
C
C  SEE ANY TEXT DEALING WITH LEAST SQUARES FOR A DISCUSSION OF
C  THEORY OF THIS METHOD.
C
C  INPUT:
C       <<FROM /OBSVTN/ >>
C          OLON,OLAT -- OBSERVED POSITIONS  (AZ & ALT) UNEFFECTED
C                       BY ORIENTATION PARAMETERS.
C
C          CLON,CLAT -- CALCULATED POSITIONS IN RADIANS. (CALC.
C                       POSITIONS ARE NEVER EFFECTED BY ORIENTATION
C                       PARAMETERS.)
C
```

```
C          NOBS        -- NUMBER OF STARS RECORDED.
C
C  INTERMEDIATE VALUES:
C       <<FROM /LSTSQR/ >>
C          NMLMAT      -- THE NORMAL MATRIX AND INVERTED TO OBTAIN
C                         CORRECTIONS TO PARAMETERS
C
C          COEFO       -- ORIGINAL ORIENTATION PARAMETERS USED TO GET NEW FIT.
C
C          COCORR      -- THE SOLUTION OF THE LEAST SQUARES FIT. THESE ARE
C                         CORRECTIONS ADDED TO THE COEFO.
C
C          F           -- THE NUMERICAL DERIVATIVE OF THE CURRENT STAR
C                         POSITIONS W.R.T. EACH PARAMETER.
C
C          LHS         -- POST-FIT O-C (OBSERVED-CALCULATED) RESIDUALS FOR
C                         EACH OBSERVATION.
C
C          W           -- WEIGHT GIVEN EACH OBSERVATION.
C
C          SD          -- STANDARD DEVIATION (SQUARED) OF THE FIT.
C
C          COVMAT      -- COVARIANCE MATRIX OF THE FIT. (NOTE: ITH DIAGONAL
C                         ELEMENT IS STANDARD DEVIATION SQUARED OF
C                         ITH PARAMETER.)
C
C          NCOEF       -- NUMBER OF COEFFICIENTS BEING SOLVED FOR
C                         IN THE FIT.
C
C  OUTPUT:
C       <<FROM /TELCOR/ >>
C          ORIRAD      -- NEW ORIENTATION PARAMETERS IN RADIANS.
C                         SEE MNLORI FOR DESCRIPTION OF INDIVIDUAL PARAMETERS
C
C  CALLS:         SCAN
C                 DNORMEQ
C                 MATINV
C                 SOLCOR
C                 ERROR(532-,IERR)
C
C  COMMONS:       LSTSQR
C                 TELCOR
C                 MATH    -- COMMONLY USED MATH CONSTANTS
C
C                                                                    RLR
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

      COMMON /LSTSQR/ NMLMAT(20,20), COEFO(20), COCORR(20), COVMAT(20,20),
     X                LHS(20),SD, SOLVEC(20), NCOEF
      REAL NMLMAT,LHS
      INTEGER SOLVEC,NCOEF
      COMMON /TELCOR/ ORIENT(20), ORIRAD(20), NPARM
      COMMON /MATH/ FILLM1(3),RDPDG,FILLM2(3)
      INTEGER ISOLVEC(20),ICOEF
```

```
C-----------------------------------------------------------------------
        DO 10 II=1,20
10      SOLVEC(II) = ISOLVEC(II)
        NCOEF = ICOEF
        NPASS = 0
        WRITE(10,1000)
1000    FORMAT(/4X,"ITERATION",8X,"STANDARD DEVIATION"   /1X,47("-")/)
C
C  INITIALIZE VARIABLES
C
100     NPASS = NPASS + 1
        CALL LSINT(ORIRAD)
C  SET UP NORMAL EQNS
        CALL DNORMEQ
C  INVERT NORMAL MATRIX TO GET SOLUTION
        CALL MATINV(NMLMAT,NCOEF,DET)
        IF (DET.NE.0.) GO TO 150
            TYPE " ** DETERMINATE=0 - ACCEPT OLD SOLUTION OR TAKE MORE READINGS"
            NOBS = 0
            RETURN
C
C  CORRECT COEFFICIENTS
150     CALL SOLCOR
        SDD = SQRT(SD)/RDPDG
        WRITE(10,1010) NPASS, SDD
1010    FORMAT(8X,I2,15X,F10.4)
X       CALL RECAP(NPASS)
C  CHECK TO SEE THAT WE HAVE MET TOLERANCES
        DO 200 II=1,NCOEF
                IF (ABS(COCORR(II)).GT.1.E-6) GO TO 100
200     CONTINUE
C  FINISHED!
        RETURN
        END

        COMPILER DOUBLE PRECISION
        SUBROUTINE LSINT(BCOEF)
C  LSINT INITIALIZES ARRAYS FOR THE NON-LINEAR LEAST SQUARES SUBSYSTEM.
C  IT ZEROES:
C       NMLMAT
C       COCORR
C       LHS
C  AND SETS COEFO TO BCOEF, THE INITIAL VALUES OF THE PARAMETERS IN THE
C  FIT.
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ RLR 10/80
        DIMENSION BCOEF(20)
        COMMON /LSTSQR/ NMLMAT(20,20), COEF0(20), COCORR(20), COVMAT(20,20),
     X                  LHS(20),SD, SOLVEC(20), NCOEF
        REAL NMLMAT, LHS
        INTEGER SOLVEC,NCOEF
C-----------------------------------------------------------------------
        SD = 0.
        DO 10 II=1,20
```

```
            DO 20 JJ=1,20
20          NMLMAT(II,JJ) = 0.
            COCORR(II) = 0.
            LHS(II)    = 0.
            COEFO(II)  = BCOEF(II)
10      CONTINUE
        RETURN
        END

        COMPILER DOUBLE PRECISION
        SUBROUTINE DNORMEQ
C
C  DOUBLE NORMAL EQUATIONS
C
C  THIS ROUTINE TAKES A SET OF GENERALIZED LONGITUDES AND LATITUDES
C  AS WELL AS THE NAME OF THE PARTIAL DERIVATIVE FUNCTION (USED FOR
C  BOTH COORDINATES) AND CREATES THE LEAST SQUARES NORMAL MATRIX.
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ RLR 10/80
        COMMON /OBSVTN/ MODEL(200), OLON(200),OLAT(200), CLON(200),CLAT(200),
     X                OCLON(200),OCLAT(200), OCLONP(200),OCLATP(200), NOBS
        COMMON /LSTSQR/ NMLMAT(20,20), COEFO(20), COCORR(20), COVMAT(20,20),
     X                LHS(20), SD, SOLVEC(20),NCOEF
        INTEGER SOLVEC,NCOEF
        REAL NMLMAT, LHS
        COMMON /MATCMN/ DLODCO(20),DLADCO(20), WLO,WLA
        COMMON /MATH/ PIO2,PI,TWOPI,RDPDG,RDPHR, FILLXX(2)
C-----------------------------------------------------------------
        DO 500 N=1,NOBS
C  FIND PREFIT OBSERVED-CALCULATED (O-C) RESIDUALS
        CALL FUNC(OLON(N),OLAT(N),MODEL(N),COEFO,0,0.,RLONO,RLATO)
        OCLONP(N) = CLON(N)-RLONO
        IF (ABS(OCLONP(N)).GT.PI) OCLONP(N) = OCLONP(N)-SIGN(TWOPI,OCLONP(N))
        OCLATP(N) = CLAT(N)-RLATO
        IF (ABS(OCLATP(N)).GT.PI) OCLATP(N) = OCLATP(N)-SIGN(TWOPI,OCLATP(N))
C  FIND PARTIALS W.R.T. THIS PAIR OF OBSERVATIONS
C  SOLVEC ALLOWS SOLVING FOR A SELECTED SET OF COEFFICIENTS.
        DO 100 II=1,NCOEF
            CALL FUNC(OLON(N),OLAT(N),MODEL(N),COEFO,SOLVEC(II),+1.E-5,RLON1,RLA
            CALL FUNC(OLON(N),OLAT(N),MODEL(N),COEFO,SOLVEC(II),-1.E-5,RLON2,RLA
            DLODCO(II) = (RLON1-RLON2)/2.E-5
            DLADCO(II) = (RLAT1-RLAT2)/2.E-5
100     CONTINUE
C  SET WEIGHT
C  LONG ACCURACY DECREASES WITH LATITUDE
        WLO = 1.*COS(RLATO)
C  LAT ACCURACY IS CONSTANT
        WLA = 1.
C  UPDATE NORMAL MATRIX
        DO 200 II=1,NCOEF
            DO 150 JJ=1,NCOEF
                NMLMAT(II,JJ) = NMLMAT(II,JJ)
     X                        + DLODCO(II)*DLODCO(JJ)*WLO
     X                        + DLADCO(II)*DLADCO(JJ)*WLA
```

```
150     CONTINUE
C ... AND LEFT-HAND-SIDE OF NORMAL EQNS
        LHS(II) = LHS(II)
     X              + DLODCO(II)*OCLONP(N)*WLO
     X              + DLADCO(II)*OCLATP(N)*WLA
200     CONTINUE
C  UPDATE SIGMA
        SD = SD + OCLONP(N)*OCLONP(N)*WLO
     X          + OCLATP(N)*OCLATP(N)*WLA
500     CONTINUE
        RETURN
        END


        COMPILER DOUBLE PRECISION
        SUBROUTC         NSTARS -- NUMBER OF STARS RECORDED
C
C    <<FROM /LSTSQR/>>
C         NCOEF  -- NUMBER OF COEFFICIENTS IN THE SOLUTION.
C
C         COEFO  -- ORIGINAL COEFFICIENTS (IN RADIANS).
C
C         SD     -- STANDARD DEVIATION (SQUARED) IN RADIANS (SQUARED)
C
C         NMLMAT -- SOLUTION MATRIX
C
C  OUTPUT:
C
C    <<FROM /LSTSQR/>>
C         COCORR -- CORRECTIONS TO OLD PARAMETERS.
C
C         COVMAT -- COVARIANCE (ERROR) MATRIX
C
C    <<FROM /TELCOR/>>
C         ORIRAD -- NEW ORIENTATION PARAMETERS IN RADIANS.
C
C  COMMONS:     STARPS
C               LSTSQR
C               TELCOR
C
C                                                                  RLR
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C
        COMMON /OBSVTN/ MODEL(200), OLON(200),OLAT(200), CLON(200),CLAT(200),
     X                  OCLON(200),OCLAT(200),OCLONP(200),OCLATP(200),NOBS
C
        COMMON /LSTSQR/ NMLMAT(20,20), COEFO(20), COCORR(20), COVMAT(20,20),
     X          LHS(20),SD,SOLVEC(20),NCOEF
        INTEGER SOLVEC,NCOEF
        REAL NMLMAT,LHS
C
        COMMON /TELCOR/ ORIENT(20), ORIRAD(20), NPARM
C
C  CALCULATE ESTIMATE FOR STANDARD DEVIATION WITH NOBS-NCOEF
```

```
C   DEGREES OF FREEDOM
C
        NDF = 2*NOBS-NCOEF
        IF (NDF.GT.0) SD = SD/FLOAT(NDF)
C
C  CREATE COVARIANCE MATRIX, CORRECTIONS TO COEFS
C
        DO 200 II=1,NCOEF
                DO 100 JJ=1,NCOEF
                        COVMAT(II,JJ) = SD*NMLMAT(II,JJ)
                        COCORR(II) = COCORR(II) + NMLMAT(JJ,II)*LHS(JJ)
100             CONTINUE
C
C     CALC NEW COEFN
        SUBROUTINE FUNC(AZ,ALT, MODEL, COEFO, L, DIFF, AZF,ALTF)
C  AZIMUTH FUNCTION USED IN NUMERICAL DIFFERENTATION
C  OF THE OBSERVATIONS
C
C               AZF = XLEVEL TERM + YLEVEL TERM
C                       + NONORTHOGONALITY TERM
C                       + TRANSVERSE TERM
C                       + AZIMUTH OFFSET + ORIGINAL AZ
C
C  INPUT:       AZ,ALT -- OBSERVED POSITIONS OF STARS IN RADIANS
C
C               MODEL  -- PARTICULAR MODEL OR LIGHT PATH. NOT USED HERE.
C
C               COEFO  -- CURRENT APPROX. OF THE ORIENTATION PARAMETERS
C
C               L      -- THE PARAMETER WHICH WE DIFFERENTIATE
C                         WITH RESPECT TO.
C
C               DIFF   -- THE AMOUNT BY WHICH THE PARAMETER IS BEING
C                         "WIGGLED" IN DIFFERENTIATION.
C
C  OUTPUT:      AZF    -- THE VALUE OF THIS AXIS
C                          WITH THE "COEFO" PARAMETERS AND "DIFF" OFFSET
C                          OF THE LTH PARAMETER FOR THE POSITIONS
C                          "AZ" & "ALT".
C
C  COMMONS:     PARTC
C
C                                                                  RLR
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
C
        REAL COEFO(20)
        COMMON /PARTC/ DELTA(20)
        DATA DELTA /20*0.0/
C
        IF (L.EQ.0) GO TO 10
            CHOLD = COEFO(L)
            COEFO(L) = COEFO(L)+DIFF
C
10      AZF = -((COEFO(4)+DELTA(4))*SIN(AZ)
```

```
     1        +(COEFO(3)+DELTA(3))*COS(AZ))
     2                              *TAN(ALT+COEFO(2)+DELTA(2))
     3        +(COEFO(7)+DELTA(7)
     3        +(COEFO(8)+DELTA(8))*COS(3.*AZ))
     3                              *TAN(ALT+COEFO(2)+DELTA(2))
     4        +(COEFO(5)+DELTA(5))/COS(ALT+COEFO(2)+DELTA(2))
     5        +(COEFO(1)+DELTA(1))
     6        +AZ
C
      ALTF = -(COEFO(4)+DELTA(4))*COS(AZ)
     1         +(COEFO(3)+DELTA(3))*SIN(AZ)
     2         +(COEFO(6)+DELTA(6))*SIN(ALT+COEFO(2)+DELTA(2))
     5         +(COEFO(9)+DELTA(9))
     6                 *COS(32.8*(ALT+COEFO(2)+DELTA(2)-COEFO(10)))
     3         +(COEFO(2)+DELTA(2))
     4         +ALT
C
C
      IF(COS(ALT+COEFO(2)+DELTA(2)) .LE. 0.) ALTF = ALTF -
     2          2.*(COEFO(6)+DELTA(6))*SIN(ALT+COEFO(2)+DELTA(2))
C
      IF (L.NE.0) COEFO(L) = CHOLD
C
      RETURN
      END



      COMPILER DOUBLE PRECISION
      SUBROUTINE RECAP(NPASS)

C  WRITES RECAP OF THIS ITERATION OF THE SOLUTION. SPECIFICALLY, IT PRINTS:
C       1) THE POSTFIT (O-C) RESIDUALS,
C       2) THE NEW CORRECTIONS TO THE COEFFICIENTS,
C       3) THE NEW COEFFICIENTS,
C       4) THE STANDARD DEVIATION
C
C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$ RLR 10/80
      COMMON // IFILLB(2),DATFIL(7)
      INTEGER DATFIL
      COMMON /MATH/ PIO2,PI,TWOPI,RDPDG,RDPHR, RDPAS,RDPTS
      COMMON /OBSVTN/ MODEL(200),OLON(200),OLAT(200),CLON(200),CLAT(200),
     X               OCLON(200),OCLAT(200),OCLONP(200),OCLATP(200),NOBS
      COMMON /OBSINF/ HDRLIN(40),IFILL(800)
      INTEGER HDRLIN
      COMMON /TELCOR/ COFDEG(20),COFRAD(20),NPARM
      COMMON /LSTSQR/ NMLMAT(20,20),COEFO(20),COCORR(20),COVMAT(20,20),
     X               LHS(20), SD, SOLVEC(20),NCOEF
      INTEGER SOLVEC
      REAL NMLMAT,LHS
      COMMON /LABELS/ LABLIS(20,20),M,N
      COMMON /MATCMN/ DLODCO(20),DLADCO(20),WLO,WLA
      REAL TEMP(200)
C------------------------------------------------------------------
```

```
      IF (NPASS.GT.1) GO TO 10
C         CALL PRTIME(12)
          WRITE(12,1007) HDRLIN
1007      FORMAT(" DATA FILE HEADER:"/ 1X,40A2)
          WRITE(12,1002)
1002      FORMAT(/31X,"OBSERVATIONS"/
     X        3X,"#",2X,"MODEL",10X,"OBSERVED",16X,"PREDICTED"/
     X        11X,2(3X,"LONGITUDE",4X,"LATITUDE")/1X,58("-")/)
          DO 5 II=1,NOBS
              OLO = OLON(II)/RDPDG
              OLA = OLAT(II)/RDPDG
              CLO = CLON(II)/RDPDG
              CLA = CLAT(II)/RDPDG
              WRITE(12,1001) II,MODEL(II),OLO,OLA,CLO,CLA
1001          FORMAT(1X,I3,3X,I3,1X,4(2X,F10.5))
5         CONTINUE
          WRITE(12,1000) (SOLVEC(II),II=1,NCOEF)
1000      FORMAT(/" PARAMETERS IN SOLUTION: ", 20(I2,1X))
C          WRITE(10,1003)
C1003      FORMAT(/4X,"ITERATION",8X,"STANDARD DEVIATION"/4X,35("-"))
C  CONVERT POSTFIT RESIDUALS TO ARCSEC AND PRINT
10    MPASS = IABS(NPASS)
      WRITE(12,1004) MPASS
1004  FORMAT(/" // PASS NUMBER ",I2/)
      DO 15 KK=1,NOBS
15    TEMP(KK) = OCLONP(KK)/RDPAS
      WRITE(12,1005) (TEMP(II),II=1,NOBS)
1005  FORMAT(/" LONGITUDINAL POSTFIT O-C RESIDUALS (ARCSEC):",
     X        40(/5(2X,E15.5)))
      DO 16 KK=1,NOBS
16    TEMP(KK) = OCLATP(KK)/RDPAS
      WRITE(12,1006) (TEMP(II),II=1,NOBS)
1006  FORMAT(/" LATITUDINAL POSTFIT O-C RESIDUALS (ARCSEC):",
     X        40(/5(2X,E15.5)))
C  CONVERT CORRECTIONS AND COEFFICIENTS TO DEGREES AND PRINT
      DO 17 KK=1,NPARM
      TEMP(KK) = COCORR(KK)/RDPDG
17    COFDEG(KK) = COFRAD(KK)/RDPDG
      WRITE(12,1010) ((TEMP(II),COFDEG(II),(LABLIS(JJ,II),JJ=1,M)),II=1,NPARM)
1010  FORMAT(/"   CORRECTION   NEW VALUE     COEFFICIENT"
     X        20(/1X,2(2X,F10.4),5X,20A2))
C  NOW FINISH OFF WITH THE STANDARD DEVIATION.
      SDD = SQRT(SD)/RDPDG
      WRITE(12,1020) SDD
1020  FORMAT(/" STANDARD DEVIATION OF NEW FIT: ", F10.4," DEGREE")
C      WRITE(10,1022) MPASS,SDD
C1022   FORMAT(8X,I2,15X,F10.4)
      RETURN
      END
```

-0.94764035	2.05865310	0.85946807	0.41295492	-0.00008342	-0.000124
1.79861926	2.26065368	-2.67738781	0.61500631	-0.00007932	-0.000039

1.98691368	2.39341470	-2.48897172	0.74776513	-0.00004787	-0.000085
1.51131678	2.87810115	-2.96[illegible]34952	1.23246766	-0.00007308	-0.000158
1.47684547	2.33471098	-2.99907615	0.68917725	-0.00006555	0.000056
0.72826524	2.82681082	2.53686517	1.18132021	0.00017184	-0.000022
0.19022918	3.02674254	2.00080639	1.38136093	0.00001675	0.000038
-0.70493316	2.65921451	1.10277337	1.01384986	-0.00015939	0.000053
0.08570418	2.19720883	1.89284098	0.55166436	-0.00013322	0.000041
-0.97465996	1.97152179	0.83237986	0.32594292	-0.00011641	0.000023
-1.41203096	2.77474467	0.39606017	1.12942015	-0.00006360	0.000046
-1.36167928	2.57611555	0.44605433	0.93072816	0.00008396	0.000022
-1.79847559	2.25372876	0.00894293	0.60821791	0.00015262	-0.000018
-2.89200844	2.20135612	-1.08452317	0.55585962	0.00027847	0.000049
-2.27913927	2.59498427	-0.47141947	0.94960052	0.00012220	0.000027
-2.31506644	2.02266846	-0.50764951	0.37715784	0.00027477	0.000068
-1.66882743	2.77524272	0.13920039	1.12991441	-0.00008761	0.000039
-0.32192704	2.04730314	1.48507246	0.40167077	-0.00019113	-0.000021

DATA FILE HEADER:

		OBSERVATIONS			
#	MODEL	OBSERVED		PREDICTED	
		LONGITUDE	LATITUDE	LONGITUDE	LATITUDE
1	1	-54.29579	117.95213	49.24389	23.66057
2	1	103.05329	129.52591	-153.40302	35.23727
3	1	113.84177	137.13256	-142.60757	42.84379
4	1	86.59207	164.90305	-169.78742	70.61520
5	1	84.61701	133.76909	-171.83441	39.48695
6	1	41.72652	161.96433	145.35167	67.68466
7	1	10.89933	173.41957	114.63776	79.14615
8	1	-40.38969	152.36177	63.18426	58.08932
9	1	4.91049	125.89079	108.45180	31.60804
10	1	-55.84390	112.95988	47.69185	18.67515
11	1	-80.90341	158.98116	22.69258	64.71101
12	1	-78.01848	147.60055	25.55703	53.32680
13	1	-103.04506	129.12915	0.51239	34.84832
14	1	-165.69988	126.12841	-62.13860	31.84841
15	1	-130.58506	148.68165	-27.01035	54.40810
16	1	-132.64354	115.89037	-29.08617	21.60955
17	1	-95.61677	159.00969	7.97559	64.73933
18	1	-18.44506	117.30183	85.08838	23.01404

PARAMETERS IN SOLUTION: 1 2 3 4 5 6 7 8

// PASS NUMBER 4

LONGITUDINAL POSTFIT O-C RESIDUALS (ARCSEC):
-0.16690D 2 -0.15613D 2 -0.95963D 1 -0.10901D 2 -0.12517

0.38702D 2	0.90600D 1	-0.34610D 2	-0.27012D 2	-0.22973
-0.16932D 2	0.14618D 2	0.30109D 2	0.57058D 2	0.21600
0.56954D 2	-0.22672D 2	-0.38580D 2		

LATITUDINAL POSTFIT O-C RESIDUALS (ARCSEC):

-0.24625D 2	-0.94444D 1	-0.19171D 2	-0.33552D 2	0.10830
-0.41916D 1	0.90959D 1	0.12230D 2	0.96590D 1	0.59878
0.10122D 2	0.53132D 1	-0.38382D 1	0.84729D 1	0.47813
0.13108D 2	0.83359D 1	-0.31201D 1		

CORRECTION	NEW VALUE	COEFFICIENT
-0.0000	103.5014	1 AZIMUTH AXIS ENCODER OFFSET
0.0000	-94.2954	2 ALTITUDE AXIS ENCODER OFFSET
0.0000	-0.0033	3 X AXIS ROTATION COMPONENT
-0.0000	0.0010	4 Y AXIS ROTATION COMPONENT
0.0000	0.0358	5 TRANSVERSE MISALIGNMENT
-0.0000	0.0214	6 LATERAL MISALIGNMENT
-0.0000	0.0057	7 AZ & ALT AXIS NONORTHOGONALITY
-0.0000	-0.0002	8 LATCH FLEXURE
0.0000	0.0000	9 ALTITUDE ENCODER WOBBLE
0.0000	0.0000	10 ALTITUDE ENCODER PHASE (ENTER ONLY)

STANDARD DEVIATION OF NEW FIT: 0.0064 DEGREE