VOLUME II

Part 2

GEODYN PROGRAMMER'S GUIDE

Contract No.: NAS 5-11735 - MOD 65
PCN 550-W-72416

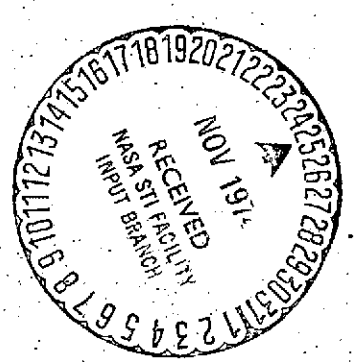
Prepared by

N.E. Mullins
N.C. Dao
T.V. Martin
C.C. Goad
N.L. Boulware
M.M. Chin

Wolf Research and Development Corporation
Riverdale, Maryland

For

Goddard Space Flight Center
Greenbelt, Maryland



October 1972

TABLE OF CONTENTS

		Page
	INTRODUCTION	i
1.0	INTRODUCTION TO THE GEODYN PROGRAM	1.0-1
2.0	GEODYN ENVIRONMENTAL REQUIREMENTS	2.0-1
3.0	DIAGRAM OF OVERLAY STRUCTURE	3.0-1
4.0	DIAGRAMS OF SUBROUTINE STRUCTURE	4.0-1
5.0	SUMMARY OF SUBPROGRAMS USED BY GEODYN	5.0-1
6.0	SUBROUTINE CROSS REFERENCE CHART	6.0-1
7.0	COMMON BLOCK CROSS REFERENCE CHART	7.0-1
8.0	PROGRAM DESCRIPTIONS	8.0-1
9.0	COMMON BLOCK DESCRIPTIONS	9.0-1

ORBIT

DESCRIPTION

ORBIT is the executive control routine for the orbit integration.

At the beginning of each arc it initializes required program constants as well as the variational partials at epoch. If epoch needs to be reset to a previous time, ORBIT negates the stepsize, and calls for COWELL integration backwards to the desired time. After backward integration is completed, ORBIT resets the stepsize to the proper positive quantity.

For each time point for which it is called, ORBIT performs the following tasks:

- If necessary, calls subroutine COWELL to integrate the orbit further.
- Calls INTRP to obtain values for the position, velocity. In the data reduction mode, variational partials associated with the orbit parameters are also calculated.
- Converts position and velocity from true equator and equinox of reference day to true equator and equinox of date using subroutine REFCOR.

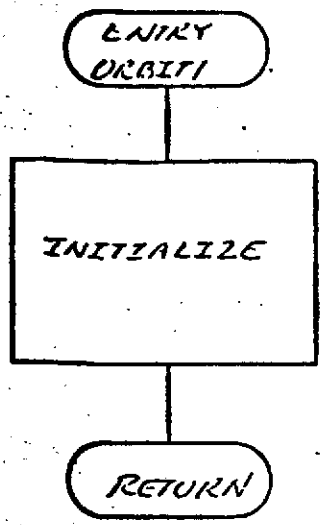
NAME	ORBIT	
ENTRY POINT	PURPOSE	
ORBITI	TO INITIALIZE	
ORBIT	TO RETURN SATELLITE STATE (POSITION & VELOCITY) AND FORCE MODEL PARTIALS AT THE CALLED TIME (DAY)	
CALLING SEQUENCE	CALL ORBITI(FCT,SUM,XPART)	
SYMBOL	TYPE	DESCRIPTION
FCT (3,1)	DP	INPUT - ACCELERATION ARRAY (ORBIT AND FORCE MODEL PARTIALS)
SUM (2,3,1)	DP	INPUT - SUM ARRAY USED BY SUMMED-COWELL INTEGRATOR & INTERPOLATOR
XPART (6,NCONMX,1)	DP	OUTPUT - ORBIT AND FORCE MODEL PARTIAL MATRIX
CALLING SEQUENCE	CALL ORBIT(DAY)	
SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT - DESIRED OUTPUT TIME
SUBROUTINES USED	COWELL	INTERP ELEM DATES REFCOR
COMMON BLOCKS	INTBLK	INITBK INTERP PRIOR1 XYZOUT
	CTIME	CELEM XYZ APARAM CONOUT
INPUT FILES	NONE	
OUTPUT FILES	PRINTER	
REFERENCES	"GEODYN SYSTEMS DESCRIPTION" VOLUME 1 - GEODYN DOCUMENTATION	

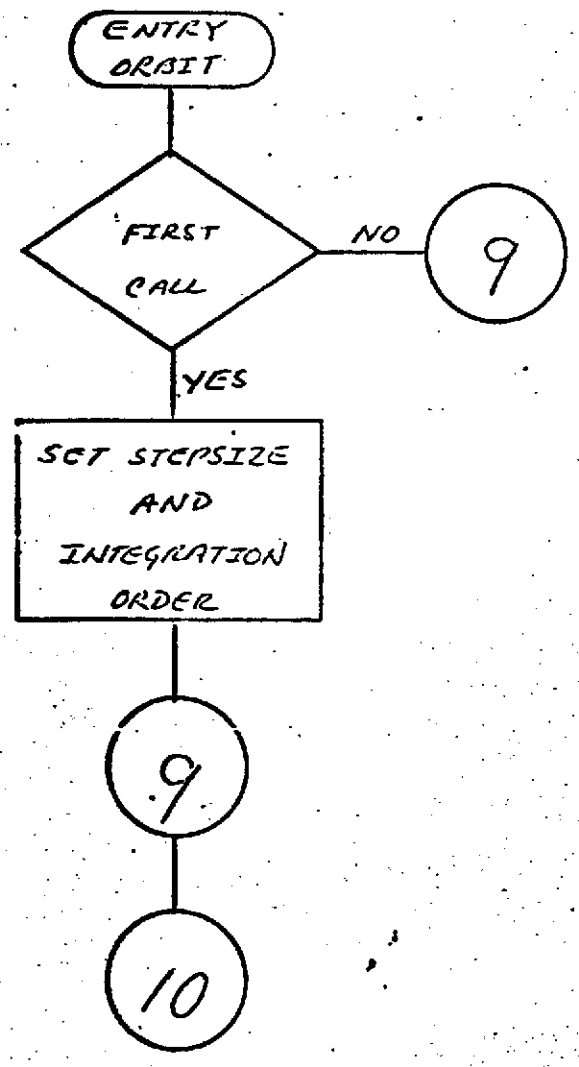
```
      SUBROUTINE ORBITI(FCT,SUM,XPART)                                  ORBI   44
      REAL*8 DAYS,DAY,SEC1,XPART(6,NCONMX,1),SUM(2,3,1),STEPSZ,FCT(3,1) ORBI   45
      REAL*8 ELCMST,GM,THETDS,AE,ACSC,DAYK,P(4),TIM(4),BODT,ORBE_       ORB2   46
      REAL*8 MSUN,MMOON,MSAT,GMZ,G,EC,APSM,APLM,PRESS,ASAT              ORBI   47
      REAL*8 EPSC,MBODY,CD,CUR,AMISS,ELMIN,FAC,DATAID,XYZFNT,           ORBI   48
     *  ERGPAR,ORBELA,JINT1,JINT2,PP(20,2),F(20,2),CC(20,2),C(20,2),     ORBI   49
     *  VCC(20,2),VC(20,2)                                               ORBI   50
      INTEGER SEAL,ADEA,ORDER,ICRDER(4)                                 ORBI   51
      LOGICAL TDS,ET,INITAL,VARSTP,NLVES*,INITC                         ORBI   52
      COMMON/INTBLK/THOUT1(4),THETDS,CM,AE,ACSC,FLAT(6),GMZ(6),P(2),    ORBI   53
     *  BODT(2),DC(3),APSM(2),APLM(2),PRESS,INITAL,NORBAT,THETGO(2),     ORBI   54
     *  MBODY(6),STEPSZ(4),FFREND(24),CRCOR(4),                          ORBI   55
```

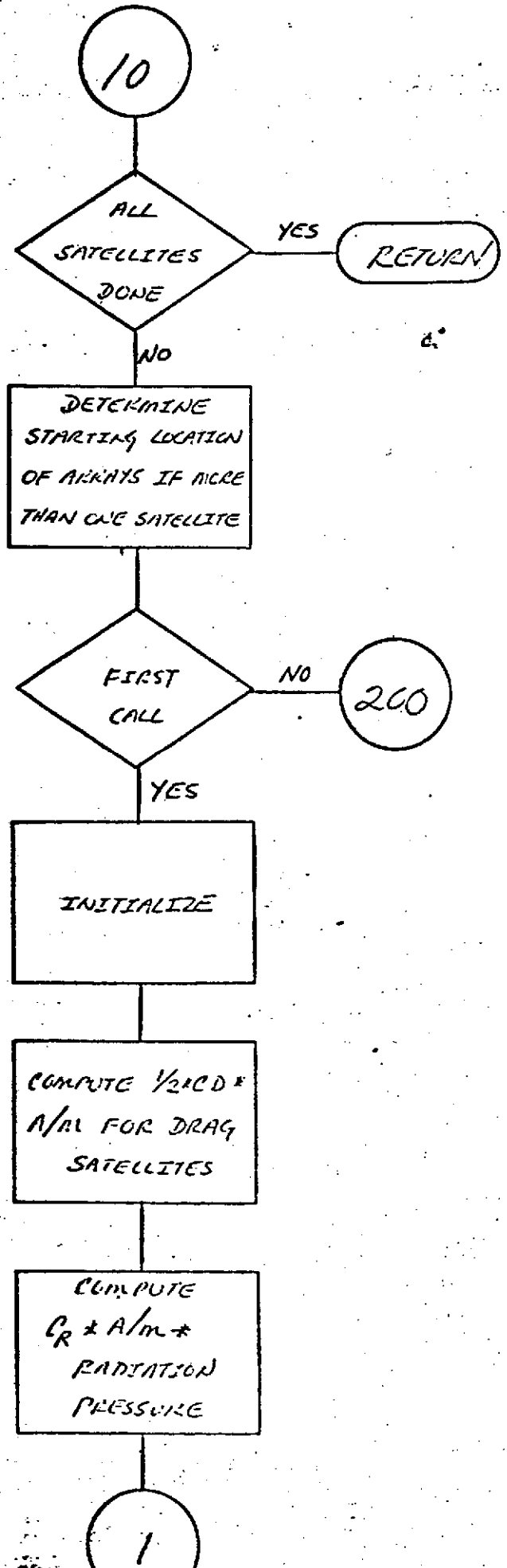
```
     *  ASAT(2),MSAT(2),VARSTP(2),HLVCSN(2),NECN(2),ACER(4),                ORBI  56
     *  SRAD(2),ISRAD(6),TORQFT,NBODY                                       ORBI  57
      COMMON/INITBK/IDAYMO,IYR4M,EPSLC,ICUMI(4),ORBEL(6,2),ICUMK(25)       ORBI  58
      COMMON/INTERP/ICUM(882),M(2,2)                                       ORBI  59
      COMMON/PRIORI/ELEMIN(6,2),IGLE(156),CD(2),CDD(2),EMISS(2)            ORBI  60
      COMMON/XYZOLT/XYZOLD(6,2),DERGPAR(6,2)                               ORBI  61
      COMMON/CTIMEC/DATABP,DAYBEF(2),DAYC,DAYSTP(17)                       ORBI  62
      COMMON/CELEM/ELEMST(6,2),ORBELA(6,2),IORBEL(5)                       ORBI  63
      COMMON/XYZ/XYZDUM(16),ISAT,IFORCE(2)                                 ORBI  64
      COMMON/APARAM/KKD(4),NSAT,KD(5)                                      ORBI  65
      COMMON/CONBLT/MINOUT(8),NECMX,IVAR(10)                               ORBI  66
      RETURN                                                               ORBI  67
      ENTRY ORBIT(DAY)                                                     ORBI  68
      INITO=INITAL                                                         ORBI  69
      IF(.NOT.INITAL)GO TO 9                                               ORBI  70
      IBAC=0                                                               ORBI  71
C SET STEPSIZE AND INTEGRATION ORDER                                       ORBI  72
      H(1)=STEPSZ(1)                                                       ORBI  73
      H(2)=STEPSZ(3)                                                       ORBI  74
      H(3)=STEPSZ(2)                                                       ORBI  75
      H(4)=STEPSZ(4)                                                       ORBI  76
      IORDER(1)=ORDER(1)                                                   ORBI  77
      IORDER(2)=ORDER(3)                                                   ORBI  78
      IORDER(3)=ORDER(2)                                                   ORBI  79
      IORDER(4)=ORDER(4)                                                   ORBI  80
      AESQ=AE**2                                                           ORBI  81
      DO 50 I=1,6                                                          ORBI  82
   50 GM2(I)=GM*MBODY(I)                                                   ORBI  83
    9 ISAT=0                                                               ORBI  84
      IPT1=1                                                               ORBI  85
      IDISP=1                                                              ORBI  86
   10 ISAT=ISAT+1                                                          ORBI  87
      ISATL1=ISAT-1                                                        ORBI  88
      IF(ISAT.GT.NSAT)RETURN                                               ORBI  89
C DETERMINE STARTING LOCATION OF ARRAYS IF MORE THAN 1 SATELLITE           ORBI  90
      IF(ISAT.GT.1)IDISP=IDISP+M(1,ISATL1)*(NECN(ISATL1)-1)*M(2,ISATL1)    ORBI  91
      IPT=1+(ISATL1)*2                                                     ORBI  92
      IF(ISAT.GT.1)IPT1=IPT1+NECN(ISATL1)                                  ORBI  93
      IF(.NOT.INITAL)GO TO 200                                             ORBI  94
C INITIALIZE                                                               ORBI  95
      M(1,ISAT)=0                                                          ORBI  96
      M(2,ISAT)=0                                                          ORBI  97
      TIM(IPT)=6.6404*DAYO                                                 ORBI  98
      TIM(IPT+1)=TIM(IPT)                                                  ORBI  99
      BC(ISAT)=0.0C                                                        ORBI 100
      IF(MSAT(ISAT).GT.0.00)BC(ISAT)=.5C*ASAT(ISAT)/MSAT(ISAT)             ORBI 101
C DRAG                                                                     ORBI 102
      B(ISAT)=BC(ISAT)*CD(ISAT)                                            ORBI 103
      BDOT(ISAT)=BC(ISAT)*CDD(ISAT)                                        ORBI 104
      APLM(ISAT)=0.00                                                      ORBI 105
      IF(MSAT(ISAT).GT.(.0C)APLM(ISAT)=ASAT(ISAT)*PRESS/MSAT(ISAT)         ORBI 106
C SOLAR REFLECTIVITIES                                                     ORBI 107
      APSM(ISAT)=APLM(ISAT)*EMISS(ISAT)                                    ORBI 108
      IF(NECN(ISAT).E.0.1)SRAD(ISAT)=0                                     ORBI 109
    1 DO 150 I=1,6                                                         ORBI 110
C INITIALIZE ORBIT                                                         ORBI 111
```

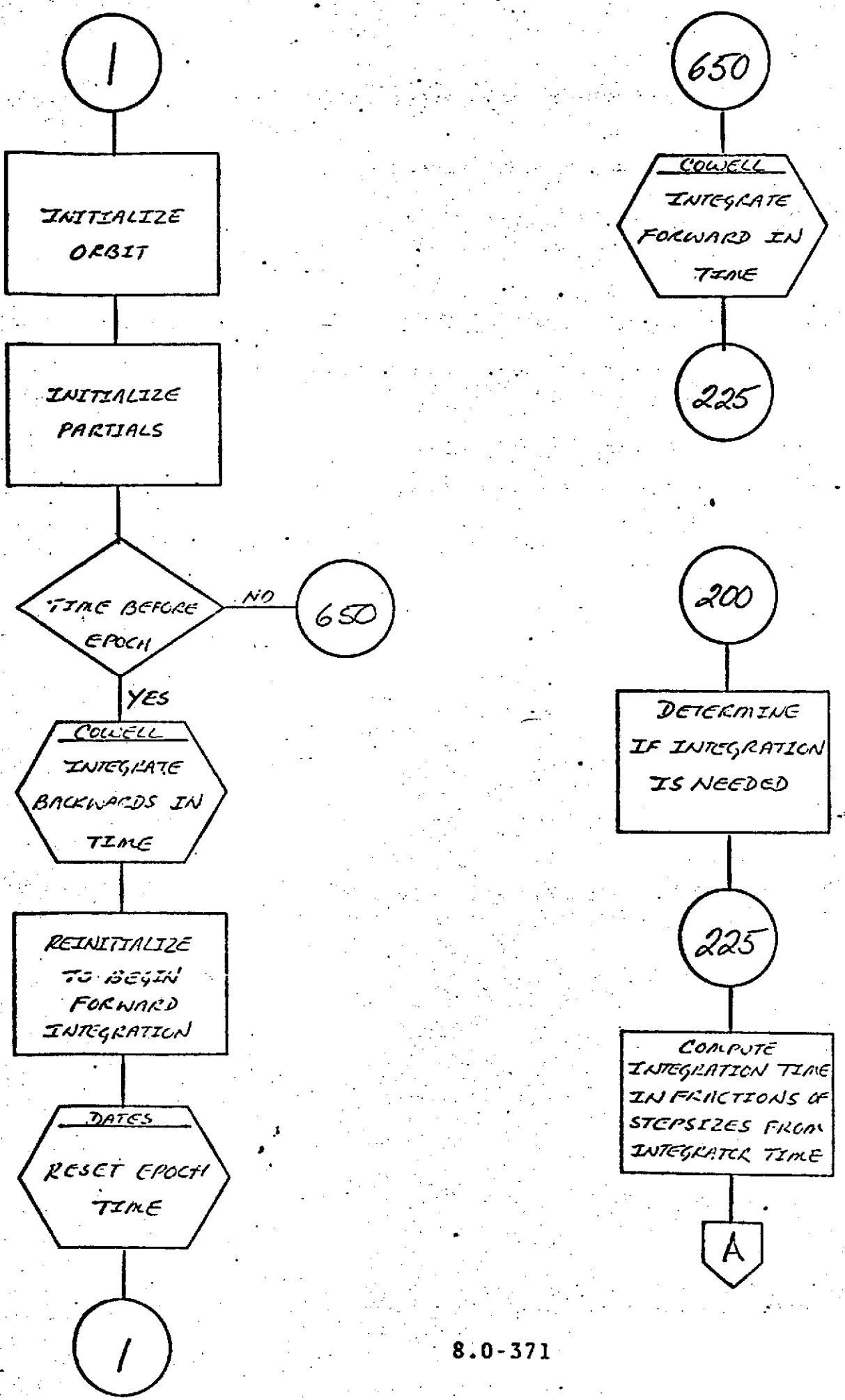
```
      XPART(I,1,ISAT)=ELEMST(I,ISAT)                                    ORBI 112
  150 CONTINUE                                                          ORBI 113
      IF(NEON(ISAT).EQ.1)GOTO190                                        ORBI 114
C INITIALIZE PARTIALS                                                   ORBI 115
      NNO=NEON(ISAT)                                                    ORBI 116
      DO 180 I=1,6                                                      ORBI 117
      DO 170 J=2,NNO                                                    ORBI 118
  170 XPART(I,J,ISAT)=0.D0                                              ORBI 119
  180 XPART(I,I+1,ISAT)=1.D0                                            ORBI 120
  190 IF(IBAC.EQ.ISAT)GOTO650                                           ORBI 121
      IF(DAYO.LE.DATAEP)GOTO650                                         ORBI 122
C INTEGRATE BACKWARDS IN TIME IF TIME POINT DESIRED PRECEDES EPOCH      ORBI 123
      H(IPT)=-H(IPT)                                                    ORBI 124
      CALL COWELL(DATAEP,XPART(1,1,ISAT),FCT(1,IDISP),IORDER(IPT),      ORBI 125
     *    H(IPT),TIM(IPT),SUM(1,1,IPT1),VARSTP(ISAT),-1.D0,M(1,ISAT),   ORBI 126
     *  1,ISAT,PP(1,ISAT),P(1,ISAT),CC(1,ISAT),C(1,ISAT),VCC(1,ISAT),    ORBI 127
     *  VC(1,ISAT))                                                     ORBI 128
      SEC1=(8.6404*DATAEP-TIM(IPT))/H(IPT)                              ORBI 129
      CALL INTRP(SEC1,H(IPT),IORDER(IPT),1,ELEMST(1,ISAT),FCT(1,IDISP), ORBI 130
     *  M(1,ISAT),SUM(1,1,IPT1))                                        ORBI 131
      CALL ELEM(ELEMST(1,ISAT),ORBELA(1,ISAT),1,.TRUE.,ORBELA(1,ISAT))  ORBI 132
      DO 240 I=1,6                                                      ORBI 133
      ORBEL(I,ISAT)=ORBELA(I,ISAT)                                      ORBI 134
      ELEMIN(I,ISAT)=ELEMST(I,ISAT)                                     ORBI 135
  240 CONTINUE                                                          ORBI 136
      IBAC=IBAC+1                                                       ORBI 137
      INITAL=.TRUE.                                                     ORBI 138
C HAVING FINISHED BACKWARD INTEGRATION, REINITIALIZE TO BEGIN FORWARD   ORBI 139
C INTEGRATION                                                           ORBI 140
      H(IPT)=-H(IPT)                                                    ORBI 141
      TIM(IPT)=8.6404*DATAEP                                            ORBI 142
      TIM(IPT+1)=TIM(IPT)                                               ORBI 143
      H(IPT)=DMIN1(H(IPT),STEPSZ(ISAT))                                 ORBI 144
      FAC=H(IPT)/STEPSZ(ISAT)                                           ORBI 145
      IF(FAC.NE.1.D0)H(IPT+1)=STEPSZ(ISAT+2)*FAC                        ORBI 146
      IF(IBAC.LT.NSAT)GOTO 1                                            ORBI 147
      DAYO=DATAEP                                                       ORBI 148
C RESET EPOCH TIME                                                      ORBI 149
      CALL DATES(DAYO,IEPYMD,IEPHM,EPSEC)                               ORBI 150
      GO TO 1                                                           ORBI 151
  650 CONTINUE                                                          ORBI 152
      CALL COWELL(DAY,XPART(1,1,ISAT),FCT(1,IDISP),IORDER(IPT),H(IPT),  ORBI 153
     *  TIM(IPT),SUM(1,1,IPT1),VARSTP(ISAT),1.D0,M(1,ISAT),NEON(ISAT),  ORBI 154
     *  ISAT,PP(1,ISAT),P(1,ISAT),CC(1,ISAT),C(1,ISAT),VCC(1,ISAT),     ORBI 155
     *  VC(1,ISAT))                                                     ORBI 156
      IF(ISAT.LT.NSAT)INITAL=INITC                                      ORBI 157
      GO TO 225                                                         ORBI 158
C DETERMINE IF INTEGRATION IS NEEDED                                    ORBI 159
  200 DINT1=8.6404*DAY                                                  ORBI 160
      DINT2=DINT1+2.D0*H(IPT+1)                                         ORBI 161
      DINT1=DINT1+H(IPT)+H(IPT)                                         ORBI 162
      IF(DINT1.GT.TIM(IPT).OR.(NEON(ISAT).GT.1.AND.DINT2.GT.            ORBI 163
     *  TIM(IPT+1)))CALL COWELL(DAY,XPART(1,1,ISAT),FCT(1,IDISP),       ORBI 164
     *  IORDER(IPT),H(IPT),TIM(IPT),SUM(1,1,IPT1),VARSTP(ISAT),         ORBI 165
     * 1.D0,M(1,ISAT),NEON(ISAT),ISAT,PP(1,ISAT),P(1,ISAT),CC(1,ISAT),  ORBI 166
     *  C(1,ISAT),VCC(1,ISAT),VC(1,ISAT))                               ORBI 167
```

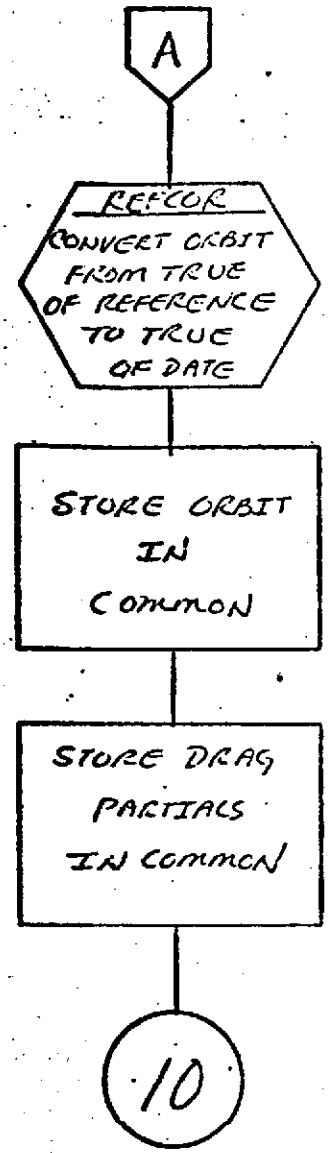
```
C COMPUTE INTEGRATION TIME IN FRACTIONS OF STEPSIZES FROM INTEGRATOR TIME     ORBI 168
C                                                                            ORBI 169
  225 SEC1=(2.6434*DAY-TIM(IPT))/H(IPT)                                      ORBI 170
      CALL INTRP(SEC1,H(IPT),ICRDER(IPT),1,XPART(1,1,ISAT),FCT(1,IDISP),     ORBI 171
     *  M(1,ISAT),SUM(1,1,IPT1))                                            ORBI 172
  250 IF(NCON(ISAT).LT.2)GO TO 300                                           ORBI 173
      SEC1=(3.6434*DAY-TIM(IPT+1))/H(IPT+1)                                  ORBI 174
      CALL INTRP(SEC1,H(IPT+1),IORDER(IPT+1),NCON(ISAT)-1,                   ORBI 175
     *  XPART(1,2,ISAT),FCT(1,IDISP+M(1,ISAT)),M(2,ISAT),                   ORBI 176
     *  SUM(1,1,IPT1+1))                                                    ORBI 177
  300 IF(TOREFT)GO TO 700                                                    ORBI 178
C CORRECT ORBIT FROM TRUE OF REFERENCE TO TRUE OF DATE                       ORBI 179
      CALL REFCOR(DAY,.FALSE.,XPART(1,1,ISAT))                               ORBI 180
      CALL REFCOR(DAY,.FALSE.,XPART(4,1,ISAT))                               ORBI 181
C STORE ORBIT IN COMMON                                                      ORBI 182
  700 DO 800 I=1,6                                                           ORBI 183
  800 XYZEND(I,ISAT)=XPART(I,1,ISAT)                                         ORBI 184
      IF(ADJR(ISAT).EQ.0)GOTO10                                              ORBI 185
C STORE DRAG PARTIALS IN COMMON                                              ORBI 186
      DO 900 I=1,6                                                           ORBI 187
  900 DRGPAR(I,ISAT)=XPART(I,8,ISAT)                                         ORBI 188
      GO TO 10                                                               ORBI 189
      END                                                                    ORBI 190
```


ENTRY ORBIT
INITIALIZE
RETURN


ENTRY ORBIT
FIRST CALL
NO
9
YES
SET STEPSIZE AND INTEGRATION ORDER
9
10


10
ALL SATELLITES DONE
YES
RETURN
NO
DETERMINE STARTING LOCATION OF ARRAYS IF MORE THAN ONE SATELLITE
FIRST CALL
NO
200
YES
INITIALIZE
COMPUTE ½ρCD*A/M FOR DRAG SATELLITES
COMPUTE CR*A/m* RADIATION PRESSURE
1


1
INITIALIZE ORBIT
INITIALIZE PARTIALS
TIME BEFORE EPOCH
NO
650
YES
COWELL
INTEGRATE BACKWARDS IN TIME
REINITIALIZE TO BEGIN FORWARD INTEGRATION
DATES
RESET EPOCH TIME
1
650
COWELL
INTEGRATE FORWARD IN TIME
225
200
DETERMINE IF INTEGRATION IS NEEDED
225
COMPUTE INTEGRATION TIME IN FRACTIONS OF STEPSIZES FROM INTEGRATOR TIME
A



NAME — OUTRAD

PURPOSE — CONVERTS INPUT IN RADIANS TO EITHER DEGREES, MINUTES AND SECONDS OR HOURS, MINUTES AND SECONDS

CALLING SEQUENCE — CALL OUTRAD(RAD,IH,IM,S,K)

SYMBOL	TYPE	DESCRIPTION
RAD	DP	INPUT - ANGLE TO BE CONVERTED IN RADIANS
IH	I	OUTPUT - SIGNED DEGREES OR HOURS
IM	I	OUTPUT - UNSIGNED MINUTES OF ARC OR TIME
S	DP	OUTPUT - UNSIGNED SECONDS OF ARC OR TIME
K	I	INPUT - SWITCH FOR OUTPUT ... K=1 OUTPUT WILL BE IN TIME UNITS K=2 OUTPUT WILL BE IN ARC UNITS

SUBROUTINES USED — NONE

COMMON BLOCKS — NONE

INPUT FILES — NONE

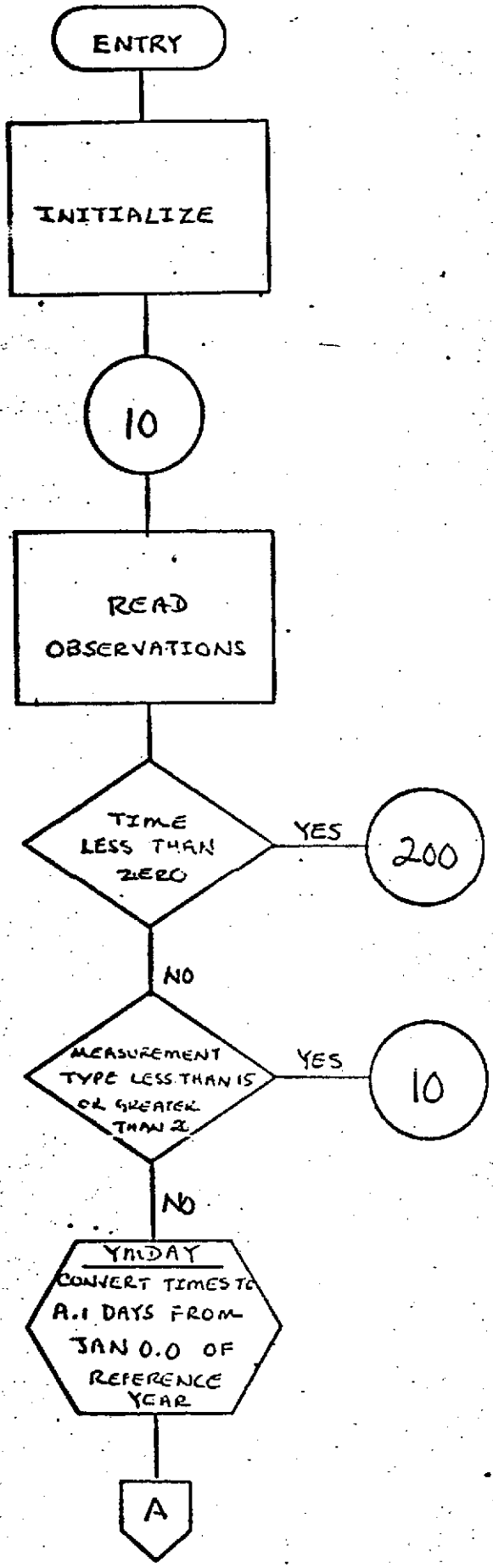
OUTPUT FILES — NONE

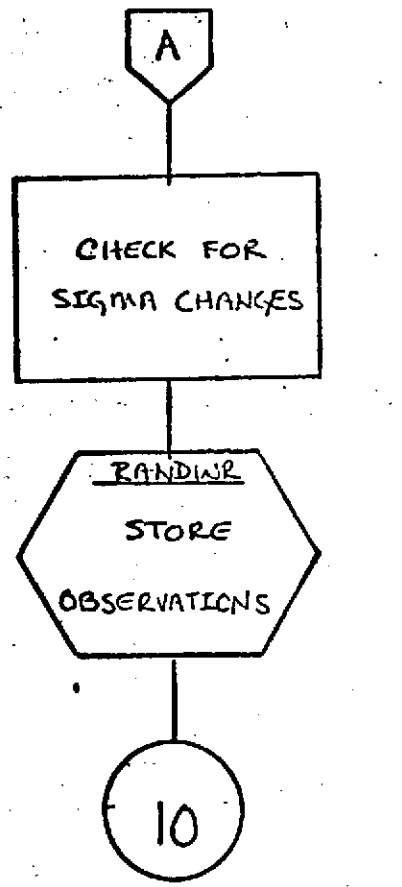
```
      SUBROUTINE OUTRAD(RAD,IH,IM,S,K)                                  OUTP  33
      REAL*8 RAD,S,RADIAN,SMIN                                          OUTP  34
C CONVERT TO DECIMAL DEGREES                                            OUTP  35
      RADIAN=RAD*57.295779500                                           OUTP  36
C IF OUTPUT IS TO BE IN HOURS, DIVIDE BY 15                             OUTP  37
      IF(K.EQ.2) RADIAN=RADIAN/15.D0                                    OUTP  38
C EXTRACT HOURS OR DEGREES                                              OUTP  39
      IH=RADIAN                                                         OUTP  40
      H=IH                                                              OUTP  41
C EXTRACT MINUTES                                                       OUTP  42
      SMIN=(RADIAN-H)*60.D0                                             OUTP  43
      IM=SMIN                                                           OUTP  44
C EXTRACT SECONDS                                                       OUTP  45
      S=(SMIN-IM)*60.000                                                OUTP  46
      RETURN                                                            OUTP  47
      END                                                               OUTP  48
```

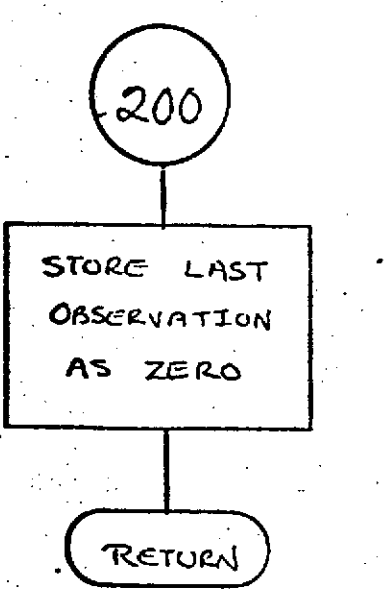
NAME	PCERD
PURPOSE	TO READ PCE FORMAT DATA
CALLING SEQUENCE	CALL PCERD
SUBROUTINES USED	RANDWR YMDAY TDIF
COMMON BLOCKS	APARAM CGEOS CONSTS CTIME INTBLK PREBLK SIGBLK
INPUT FILES	NONE
OUTPUT FILES	NONE
REFERENCES	"GEODYN SYSTEMS DESCRIPTION" - APPENDIX C VOLUME 3 - GEODYN DOCUMENTATION

```
      SUBROUTINE PCERD                                                  PCER  22
      IMPLICIT REAL*8 (A-H,O-Z)                                         PCER  23
      LOGICAL*1 OKSAT,VHFCHN,PREPRO                                     PCER  24
      LOGICAL NORATE                                                    PCER  25
      INTEGER*2 CULL,CHANEL,NMEAS,MTYPE,PRETYP,IMTYPE,ISTNO,ISATNO,ITYPEPCER  26
      INTEGER PRCNO                                                     PCER  27
      REAL TDIF,SIGSTD,DAYINT,SIGCHG,SGPRNT                             PCER  28
      COMMON/CGEOS/ISATID,IPREPR(251),APRC,NSIG,NCULL,SIGCHG(50),       PCER  29
     *   IMTYPE(50),ISTNO(50),CULL(2,100)                               PCER  30
      COMMON/CONSTS/CPI,DTOR,D2R,S2P                                    PCER  31
      COMMON/CTIME/DATARP,DAYREF,DAYL,DAYSTP,DAYINT(15)                 PCER  32
      COMMON/INTBLK/INTBK1(53),NORATE,INTBK2(78)                        PCER  33
      COMMON/PREBLK/DAY,OBS1,OBS2,SIG1,SIG2,SRENGX,ISTA,MTYPE,NMEAS,    PCER  34
     *   ISATNO,PRETYP,CHANEL,VHFCHN,PREPRO,PRCNO                       PCER  35
      COMMON/SIGBLK/SIGSTD(30),SGPRNT(30),ICBS,IDTAPE(3)                PCER  36
      ALTIME(DAY)=TDIF(4,3,DAY)/8.64D4                                  PCER  37
      IF(ICBS.LE.0) ICBS=20                                             PCER  38
C INITIALIZE                                                            PCER  39
      NUMBER=0                                                          PCER  40
      NORATE=.FALSE.                                                    PCER  41
      SIGB=0.0D0                                                        PCER  42
      ISTA=0                                                            PCER  43
      ISATNO=1                                                          PCER  44
      NMEAS=1                                                           PCER  45
      PRETYP=0                                                          PCER  46
      CHANEL=0                                                          PCER  47
      VHFCHN=.FALSE.                                                    PCER  48
      PREPRO=.FALSE.                                                    PCER  49
      SRENGX=0.0D0                                                      PCER  50
      OBS2=0.0D0                                                        PCER  51
C READ OBSERVATIONS                                                     PCER  52
   10 CALL RANDWR(ICBS,IDTAPE,ERR=10,END=200) MTYPE,IYMD,IHMS,SEC,OBS1,SIG1PCER  53
      MTYPE=MTYPE-1                                                     PCER  54
      IF(IYMD.LT.0) GO TO 200                                           PCER  55
```

```
      IF(MTYPE.LT.15.OR.MTYPE.GT.26) GO TO 10                          PCER  56
C CONVERT TIME TO A.1 DAYS FROM JAN 0.0 OF REFERENCE YEAR              PCER  57
      DAY=YMDAY(IYMD,IHM,SEC)                                          PCER  58
      DAY=DAY+ATIME(DAY)                                               PCER  59
      IF(DAY.LT.DAYSTP) GO TO 10                                       PCER  60
      IF(DAY.GT.DAYSTP) GO TO 200                                      PCER  61
      IF(SIG1.LE.0.0D0) SIG1=SIGSTO(MTYPE)                             PCER  62
      IF(NSIG.LE.0) GO TO 50                                           PCER  63
      NN1=0                                                            PCER  64
C CHECK FOR SIGMA CHANGES                                              PCER  65
      DO 20 I=1,NSIG                                                   PCER  66
      IF(ISTNO(I).EQ.0.AND.ITYPE(I).EQ.MTYPE) NN1=I                    PCER  67
   20 CONTINUE                                                         PCER  68
      IF(NN1.GT.0) SIG1=SIGCHG(NN1)                                    PCER  69
   50 IF(MTYPE.LT.21.AND.MTYPE.GT.17) SIG1=SIG1*1.0D-2                 PCER  70
      IF(MTYPE.EQ.22) SIG1=SIG1*1.0D-6                                 PCER  71
  100 IF(MTYPE.LE.22) GO TO 110                                        PCER  72
      OBS1=OBS1*0.2F                                                   PCER  73
      SIG1=SIG1*525                                                    PCER  74
  110 NUMBER=NUMBER+1                                                  PCER  75
      IF(NCELL.LE.0) GO TO 260                                         PCER  76
      DO 240 I=1,NCELL                                                 PCER  77
      IF(NUMBER-CELL(1,I)) 240,230,220                                 PCER  78
  220 IF(NUMBER.GT.CELL(2,I)) GO TO 240                                PCER  79
  230 SIG1=0.0D0                                                       PCER  80
  240 CONTINUE                                                         PCER  81
  260 RECNO=RECNO+1                                                    PCER  82
C STORE OBSERVATIONS                                                   PCER  83
      CALL RANDWR                                                      PCER  84
      GO TO 10                                                         PCER  85
C STORE LAST OBSERVATION AS ZERO                                       PCER  86
  200 MTYPE=0                                                          PCER  87
      RECNO=RECNO+1                                                    PCER  88
      CALL RANDWR                                                      PCER  89
      PRINT 2000,NUMBER,IOBS                                           PCER  90
      DAYSTP=DAY                                                       PCER  91
      RETURN                                                           PCER  92
 1000 FORMAT(1X,I2,I6,I4,F8.4,D24.16,D10.4)                            PCER  93
 2000 FORMAT(1H0//31X,I6,' OBSERVATIONS SELECTED FROM MASTER PCE DATA ',PCER  94
     *   'TAPE NUMBER',I3)                                            PCER  95
      END                                                              PCER  96
```


ENTRY
INITIALIZE
10
READ OBSERVATIONS
TIME LESS THAN ZERO
YES
200
NO
MEASUREMENT TYPE LESS THAN 1 OR GREATER THAN 2
YES
10
NO
YMDAY
CONVERT TIMES TO A.1 DAYS FROM JAN 0.0 OF REFERENCE YEAR
A


A
CHECK FOR SIGMA CHANGES
RANDWR
STORE OBSERVATIONS
10


200
STORE LAST OBSERVATION AS ZERO
RETURN

NAME PDEN

ENTRY POINT	PURPOSE
PDLN2	INITIALIZATION
PDEN	TC PRINT ADJUSTED SURFACE DENSITIES

CALLING SEQUENCE CALL PDEN2(BCENTR,AREA,DEN,ADJDEN,APSIG,SUM1,
DENCON,SAVSIG)

SYMBOL	TYPE	DESCRIPTION
BCENTR (2,1)	DP	INPUT - THE LATITUDE AND LONGITUDE OF THE ADJUSTED SURFACE DENSITY BLOCKS
AREA (4,1)	DP	INPUT - SURFACE DENSITY SUB-BLOCK AREAS
DEN (1)	DP	INPUT - SURFACE DENSITY VALUES
ADJDEN (1)	DP	INPUT & OUTPUT - ADJUSTED SURFACE DENSITY VALUES
APSIG (1)	DP	INPUT - APRIORI SIGMAS FOR ADJUSTED DENSITIES
SUM1 (1)	DP	INPUT - NORMAL MATRIX
DENCON (NCONST,1)	DP	INPUT - MATRIX OF CONSTRAINT EQUATION
SAVSIG (1)	DP	SCRATCH

CALLING SEQUENCE CALL PDEN(ICUTER)

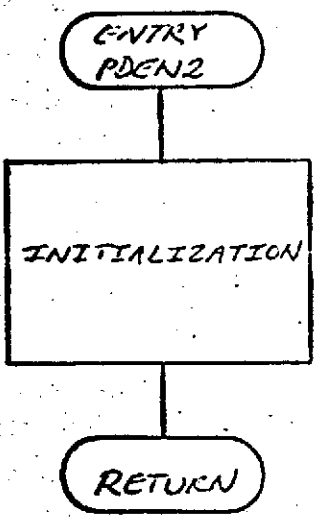
SYMBOL	TYPE	DESCRIPTION
ICUTER	I	INPUT - CUTER ITERATION

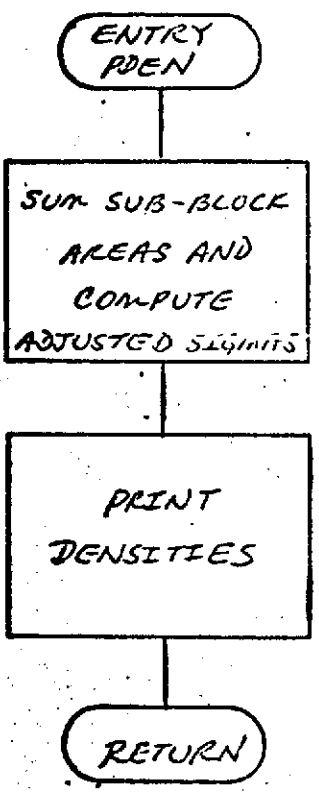
SUBROUTINES USED	NONE
COMMON BLOCKS	CPARAM TPEBLK
INPUT FILES	NONE
OUTPUT FILES	IOUT - PRINTER

SUBROUTINE PDEN2(BCENTR,AREA,DEN,ADJDEN,APSIG,SUM1,DENCON,SAVSIG) PDEN 55

```
      IMPLICIT REAL*8(A-H,O-Z)                                                  PDEN  56
      LOGICAL CMPGPR                                                            PDEN  57
      DIMENSION BCENTR(2,1),AREA(4,1),DEN(1),ADJDEN(1),APSIG(1),SUM1(1)         PDEN  58
      DIMENSION DENCON(NCONST,1),SAVSIG(1),INDS(4)                              PDEN  59
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,NGPCON,            PDEN  60
     *   NCSEST,CMPGPR,LIM1,LIM2,NDEN,NABLOK,NTIEST,NTIDEN,INNRS#,              PDEN  61
     *   NCONST,NDCONS                                                          PDEN  62
      COMMON/IPBLK/INTP,ICUT,ITAPES(10)                                         PDEN  63
      INDXND(I)=NDIM*(I-1)-(I*(I-1))/2                                          PDEN  64
      NCON2=NABLOK-NCONST                                                       PDEN  65
      INDS(1)=1                                                                 PDEN  66
      INDS(2)=NCON2                                                             PDEN  67
      INDS(3)=NABLOK+1                                                          PDEN  68
      INDS(4)=NDEN                                                              PDEN  69
      LM=2                                                                      PDEN  70
      IF(NDEN.GT.NABLOK) LM=4                                                   PDEN  71
      RETURN                                                                    PDEN  72
      ENTRY PDEN(IOUTER)                                                        PDEN  73
      I2=NDIM-3*NMAST-NABLOK+NCONST                                             PDEN  74
      DO 100 I=1,NABLOK                                                         PDEN  75
      I2=I2+1                                                                   PDEN  76
      IF(MOD(I,45).EQ.1)WRITE(ICUT,1000)IOUTER                                  PDEN  77
      IF(MOD(I,5).EQ.1)WRITE(ICUT,1002)                                         PDEN  78
      DEG=BCENTR(1,I)                                                           PDEN  79
      IDP=DEG                                                                   PDEN  80
      SP=(DEG-IDP)*60.D0                                                        PDEN  81
      IMP=SP                                                                    PDEN  82
      SP=(SP-IMP)*60.D0                                                         PDEN  83
      DEG=BCENTR(2,I)                                                           PDEN  84
      IDL=DEG                                                                   PDEN  85
      SL=(DEG-IDL)*60.0D0                                                       PDEN  86
      IML=SL                                                                    PDEN  87
      SL=(SL-IML)*60.D0                                                         PDEN  88
C SUM SUB-BLOCK AREAS AND COMPUTE ADJUSTED SIGMAS                               PDEN  89
      ASUM=AREA(1,I)+AREA(2,I)+AREA(3,I)+AREA(4,I)                              PDEN  90
      IF(I.GT.NCON2) GO TO 50                                                   PDEN  91
      I1=INDXND(I2)+I2                                                          PDEN  92
      ADJSIG=DSQRT(SUM1(I1))                                                    PDEN  93
      SAVSIG(I)=ADJSIG                                                          PDEN  94
      GO TO 100                                                                 PDEN  95
   50 SUM=0.0D0                                                                 PDEN  96
      SUMSQ2=0.0D0                                                              PDEN  97
      I1=I-NCON2                                                                PDEN  98
      DO 60 L=1,LM,2                                                            PDEN  99
      J1=INDS(L)                                                                PDEN 100
      J2=INDS(L+1)                                                              PDEN 101
      DO 60 J=J1,J2                                                             PDEN 102
      PDCDA=DENCON(I1,J)                                                        PDEN 103
      SUM=SUM+PDCDA*ADJDEN(J)                                                   PDEN 104
      IF(L.GT.1) GO TO 60                                                       PDEN 105
      SUMSQ2=SUMSQ2+(PDCDA*SAVSIG(J))**2                                        PDEN 106
   60 CONTINUE                                                                  PDEN 107
      ADJSIG=DSQRT(SUMSQ2)                                                      PDEN 108
      ADJDEN(I)=SUM                                                             PDEN 109
C PRINT DENSITIES                                                               PDEN 110
  100 WRITE(ICUT,1001)I,IDP,IMP,SP,IDL,IML,SL,ASUM,DEN(I),                      PDEN 111
```

```
     *   ADJDEN(I),APSIG(I),ADJSIG                                          PDEN 112
         RETURN                                                             PDEN 113
1000     FORMAT(1H1,40X,46HSURFACE DENSITIES ADJUSTED FOR OUTER ITERATION,  PDEN 114
     *   I2/      30X,15HCENTER OF BLOCK,14X,5HBLOCK,10X,5HVALUE,24X,       PDEN 115
     *   5HSIGMA/21X,35HGEOCENTRIC LATITUDE  EAST LONGITUDE,4X,            PDEN 116
     *   4HAREA,7X,6HA-PRIORI,7X,8HADJUSTED,7X,5HA-PRIORI,4X,8HADJUSTED/   PDEN 117
     *   13X,6HBLOCK ,2(4X,13HDEG MN SEC,4X,7H(KM**2),                     PDEN 118
     *   3(6X,9H(KG/M**2)),3X,9H(KG/M**2))                                 PDEN 119
1001     FORMAT(12X,I5,2X,2(I7,I3,F7.3),-6PF12.0,1X,1P2D15.6,1X,2D12.3)     PDEN 120
1002     FORMAT(1X)                                                         PDEN 121
         END                                                                PDEN 122
```





NAME	PDEN1
PURPOSE	TO PRINT INPUT SURFACE DENSITIES
CALLING SEQUENCE	CALL PDEN1
SUBROUTINES USED	NONE
COMMON BLOCKS	SRFBLK TPEBLK
INPUT FILES	NONE
OUTPUT FILES	IOUT - PRINTER

```
      SUBROUTINE PDEN1                                                  PDEN  18
      IMPLICIT REAL*8(A-H,O-Z)                                          PDEN  19
      REAL DLAT,DLON,DEN,APSIG                                          PDEN  20
      INTEGER*2 NLAT,NLON                                               PDEN  21
      COMMON/SRFBLK/SLAT(675),SLON(675),DLAT(675),DLON(675),DEN(675),   PDEN  22
     *   APSIG(675),NLAT(675),NLON(675),NSBLOK                          PDEN  23
      COMMON/TPEBLK/INTP,IOUT,ITAPES(10)                                PDEN  24
      NLINES=0                                                          PDEN  25
      IDEN2=0                                                           PDEN  26
      DO 100 K=1,2                                                      PDEN  27
      DO 100 I=1,NSBLOK                                                 PDEN  28
      IF(APSIG(I).LE.0.0) GO TO (100,20),K                              PDEN  29
      IF(K.EQ.2)GO TO 100                                               PDEN  30
   20 NLINES=NLINES+1                                                   PDEN  31
      IF(MOD(NLINES,45).EQ.1)WRITE(IOUT,1000)                           PDEN  32
      IF(MOD(NLINES,5).EQ.1)WRITE(IOUT,1003)                            PDEN  33
      IDEN1=IDEN2+1                                                     PDEN  34
      IDEN2=IDEN1+NLAT(I)*NLON(I)-1                                     PDEN  35
      ICP=SLAT(I)                                                       PDEN  36
      ICL=SLON(I)                                                       PDEN  37
      SP=(SLAT(I)-ICP)*60.D0                                            PDEN  38
      SL=(SLON(I)-ICL)*60.D0                                            PDEN  39
      IMP=SP                                                            PDEN  40
      IML=SL                                                            PDEN  41
      SP=(SP-IMP)*60.D0                                                 PDEN  42
      SL=(SL-IML)*60.D0                                                 PDEN  43
      BN=NLAT(I)*DLAT(I)                                                PDEN  44
      BE=NLON(I)*DLON(I)                                                PDEN  45
      BS=SLAT(I)-BN*0.5D0                                               PDEN  46
      BW=SLON(I)-BE*0.5D0                                               PDEN  47
      BN=BS+BN                                                          PDEN  48
      BE=BW+BE                                                          PDEN  49
      IF(K.EQ.1)WRITE(IOUT,1001)IDEN1,IDEN2,ICP,IMP,SP,ICL,IML,SL,      PDEN  50
     *   DLAT(I),DLON(I),NLAT(I),NLON(I),DEN(I),APSIG(I),BN,BS,BE,BW    PDEN  51
      IF(K.EQ.2)WRITE(IOUT,1002)IDEN1,IDEN2,ICP,IMP,SP,ICL,IML,SL,      PDEN  52
     *   DLAT(I),DLON(I),NLAT(I),NLON(I),DEN(I),BN,BS,BE,BW             PDEN  53
  100 CONTINUE                                                          PDEN  54
      RETURN
```

```
 100C  FORMAT(1HI,53X,27HSURFACE DENSITY BLOCKS USED/                    PDEN  55
     *    17X,22HCENTER OF MASTER BLOCK,7X,18HSIZE OF INCREMENTS,       PDEN  56
     *    3X,9HNUMBER OF,8X,15HSURFACE DENSITY,                         PDEN  57
     *    8X,23HMASTER BLOCK BOUNDARIES/                                PDEN  58
     *    11X,25HGEOCENTRIC LATITUDE  EAST LONGITUDE,3X,3HLAT,6X,3HLON, PDEN  59
     *    5X,10HINCREMENTS,7X,5HVALUE,9X,5HSIGMA,4X,12HNORTH  SOUTH,    PDEN  60
     *    3X,4HEAST,3X,4HWEST/                                          PDEN  61
     *    4X,6HBLOCKS,4X,2(1X,14HDEG MN SECONDS,2X),2(3X,3HDEG,4X),     PDEN  62
     *    2X,3HLAT,2X,3HLON,1X,2(5X,9H(KG/M**2)),4(3X,3HDEG,1X))        PDEN  63
 1001 FORMAT(2X,I3,3H TO,I4,2(3X,2I3,F7.3,1X),2(F8.2,1X),1X,2I5,1PE18.6,PDEN  64
     *    5I2.2,0P4F7.1)                                                PDEN  65
 1002 FORMAT(2X,I3,3H TO,I4,2(3X,2I3,F7.3,1X),2(F8.2,1X),1X,2I5,1PF18.6,PDEN  66
     *    2X,10HUNADJUSTED,0P4F7.1)                                     PDEN  67
 1003  FORMAT(1X)                                                       PDEN  68
       END                                                              PDEN  69
                                                                        PDEN  70
```

PLHOUT

DESCRIPTION

Subroutine PLHOUT converts an input Earth-fixed Cartesian station position and its associated Cartesian covariance matrix to the spherical geodetic coordinate system.

The order of computation is:

- Compute the spheroid height, h, using an iterative procedure.
- Compute the geodetic latitude ϕ, and the east longitude λ.
- Compute the partial derivatives of ϕ, λ, and h with respect to the rectangular coordinates.
- Invoke subroutine VCONV to convert the input covariance matrix.

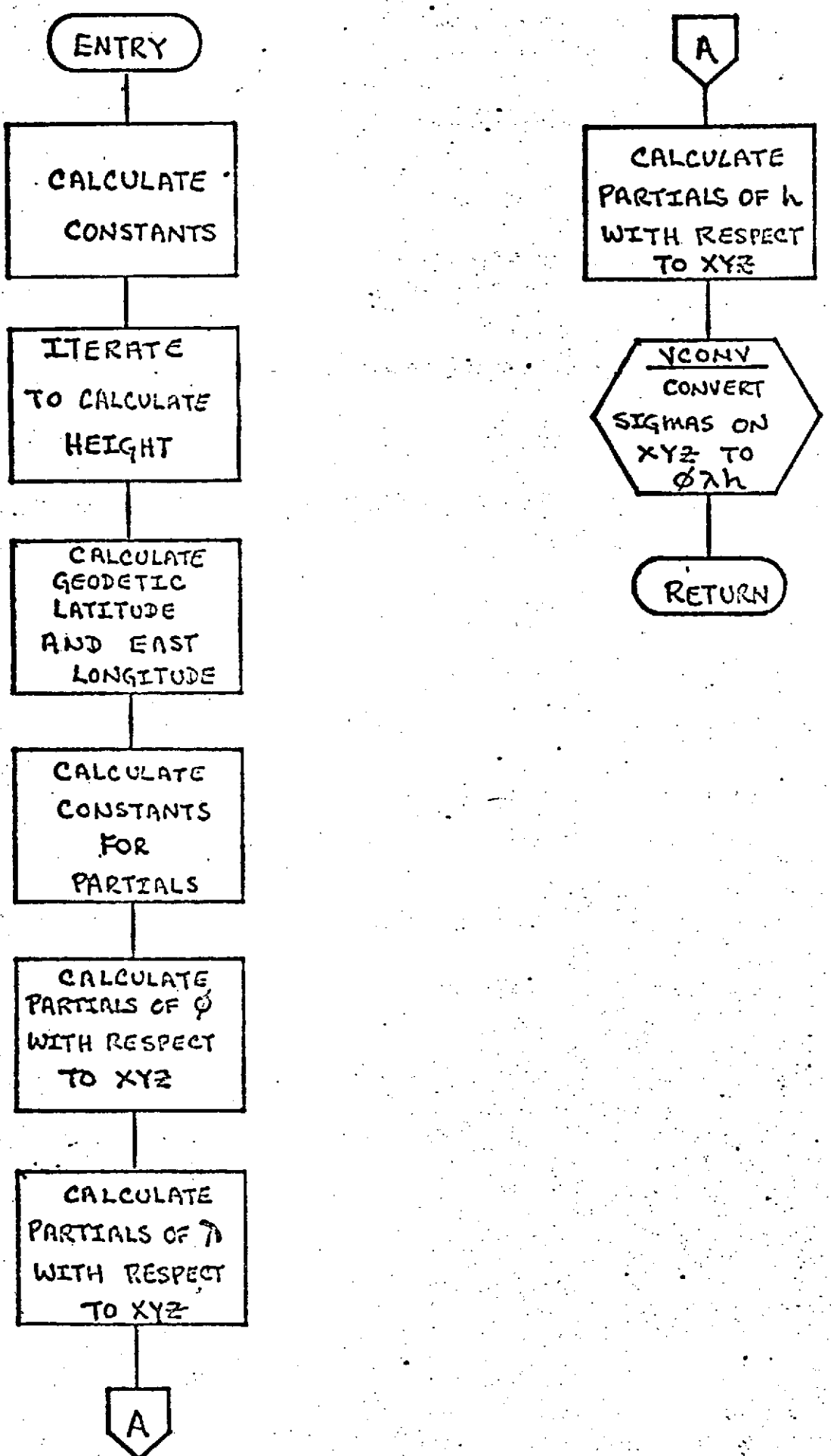
NAME	PLHOUT
PURPOSE	1) TO COMPUTE GEODETIC PHI, LAMBDA, H FROM GEOCENTRIC X, Y, Z 2) TO COMPUTE PARTIAL DERIVATIVES OF GEODETIC COORDINATES WITH RESPECT TO GEOCENTRIC COORDINATES
CALLING SEQUENCE	CALL PLHOUT(STAP,XYZSIG,PLHSIG,PHI,LAMBDA,H)

SYMBOL	TYPE	DESCRIPTION
STAP (3)	DP	INPUT - GEOCENTRIC X,Y,Z
XYZSIG (3,3)	R	INPUT - SIGMAS ON X,Y,Z
PLHSIG (3,3)	R	OUTPUT - SIGMAS ON PHI, LAMBDA , H
PHI	DP	OUTPUT - GEODETIC LATITUDE
LAMBDA	DP	OUTPUT - GEODETIC EAST LONGITUDE
H	R	OUTPUT - GEODETIC HEIGHT

SUBROUTINES USED	DARCTN VCONV
COMMON BLOCKS	INTBLK
INPUT FILES	NONE
OUTPUT FILES	NONE
REFERENCES	"GEODYN SYSTEMS DESCRIPTION" VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE PLHOUT(STAP,XYZSIG,PLHSIG,PHI,LAMBDA,H)                PLHO  42
      DIMENSION PLHSIG(3,3),XYZSIG(3,3),PARTL(3,3)                      PLHO  43
      REAL*8 PHI,LAMBDA,SINPHI,ZT,HI,XYSQ,T,ESQ,ESQ1,T1,N2,ESQSP,DARCTN,PLHO  44
     *       A2,STAP(3),SPSQ,CONV,PTXYSQ                                PLHO  45
      COMMON/INTBLK/G1(8),A2,A1SQ(2),FLAT,G2(118)                       PLHO  46
      REAL*8 FLAT                                                       PLHO  47
      DATA DELTA/.001/                                                  PLHO  48
C CALCULATE CONSTANTS                                                   PLHO  49
      ESQ=FLAT*(2.0-FLAT)                                               PLHO  50
      ESQ1=(1.0-FLAT)**2                                                PLHO  51
      T1=1.0-ESQ                                                        PLHO  52
      T=ESQ*STAP(3)                                                     PLHO  53
      XYSQ=STAP(1)**2+STAP(2)**2                                        PLHO  54
C ITERATIVE PROCEDURE FOR HEIGHT                                        PLHO  55
```

```
      DO 10 J=1,25                                                      PLHO  56
      ZT=STAP(3)+T                                                      PLHO  57
      H1=SQRT(XYSQ+ZT**2)                                               PLHO  58
      SINPHI=ZT/H1                                                      PLHO  59
      ESQSP=ESQ*SINPHI                                                  PLHO  60
      H2=AE/SQRT(1.0D0-ESQSP*SINPHI)                                    PLHO  61
      T1=H2*ESQSP                                                       PLHO  62
      IF(DABS(T1-T).LT.DELTA) GO TO 20                                  PLHO  63
   10 T=T1                                                              PLHO  64
C HEIGHT                                                                PLHO  65
   20 H=H1-H2                                                           PLHO  66
C GEODETIC LATITUDE                                                     PLHO  67
      RTXYSQ=DSQRT(XYSQ)                                                PLHO  68
      PHI=DATAN2(ZT,RTXYSQ)                                             PLHO  69
C EAST LONGITUDE                                                        PLHO  70
      LAMBDA=DARCTN(STAP(2),STAP(1))                                    PLHO  71
C CALCULATE CONSTANTS FOR PARTIALS                                      PLHO  72
      ZSQ=STAP(3)**2                                                    PLHO  73
      SPSQ=SINPHI**2                                                    PLHO  74
      COSPHI=DSQRT(1.0D0-SPSQ)                                          PLHO  75
      CONPHI=ESQI/((ESQI**2*XYSQ+ZSQ)*RTXYSQ)                           PLHO  76
      CONH=1.0D0-ESQ*SPSQ                                               PLHO  77
      CONH=CONH*DSQRT(CONH)                                             PLHO  78
      CONH=-ESQ*AE*ESQI*SINPHI*COSPHI/CONH-STAP(3)*COSPHI/SPSQ          PLHO  79
C PARTIALS OF PHI WITH RESPECT TO X,Y,Z                                 PLHO  80
      PARTL(1,1)=-STAP(1)*STAP(3)*CONPHI                                PLHO  81
      PARTL(2,1)=-STAP(2)*STAP(3)*CONPHI                                PLHO  82
      PARTL(3,1)=XYSQ*CONPHI                                            PLHO  83
C PARTIALS OF LAMBDA WITH RESPECT TO X,Y,Z                              PLHO  84
      PARTL(1,2)=-STAP(2)/XYSQ                                          PLHO  85
      PARTL(2,2)=STAP(1)/XYSQ                                           PLHO  86
      PARTL(3,2)=0.                                                     PLHO  87
C PARTIALS OF H WITH RESPECT TO X,Y,Z                                   PLHO  88
      PARTL(1,3)=CONH*PARTL(1,1)                                        PLHO  89
      PARTL(2,3)=CONH*PARTL(2,1)                                        PLHO  90
      PARTL(3,3)=CONH*PARTL(3,1)+1.0D0/SINPHI                           PLHO  91
C CONVERT SIGMAS ON X,Y,Z TO SIGMAS ON PHI, LAMBDA, H                   PLHO  92
      CALL VCONV(XYZSIG,PLHSIG,PARTL)                                   PLHO  93
      RETURN                                                            PLHO  94
      END                                                               PLHO  95
```


ENTRY
CALCULATE CONSTANTS
ITERATE TO CALCULATE HEIGHT
CALCULATE GEODETIC LATITUDE AND EAST LONGITUDE
CALCULATE CONSTANTS FOR PARTIALS
CALCULATE PARTIALS OF φ WITH RESPECT TO XYZ
CALCULATE PARTIALS OF λ WITH RESPECT TO XYZ
A
A
CALCULATE PARTIALS OF h WITH RESPECT TO XYZ
VCONV
CONVERT SIGMAS ON XYZ TO φλh
RETURN

NAME	POLE
PURPOSE	TO SELECT FROM A TABLE FOR A GIVEN INPUT DATE THE COORDINATES OF THE TRUE POLE
CALLING SEQUENCE.	CALL POLE(XP,YP,DAY)

SYMBOL	TYPE	DESCRIPTION
XP	DP	OUTPUT - X COORDINATE OF THE POLE
YP	DP	OUTPUT - Y COORDINATE OF THE POLE
DAY	DP	INPUT - TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR FOR THE ARC

SUBROUTINES USED	DJUL
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE
REFERENCES	'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE POLE(XP,YP,DAY)                                        POLE  31
      DOUBLE PRECISION XP,YP,DAY,D,DJUL,DFIRST,DLAST                    POLE  32
      DIMENSION X(687),Y(687),X1(117),X2(180),X3(181),X4(209),Y1(117),  POLE  33
     .   Y2(180),Y3(181),Y4(209)                                        POLE  34
      EQUIVALENCE (X1(1),X(1)),(Y1(1),Y(1)),(X2(1),X(118)),(Y2(1),Y(118)POLE  35
     .   ),(X3(1),X(298)),(Y3(1),Y(298)),(X4(1),X(479)),(Y4(1),Y(479))  POLE  36
C FIRST POINT IS ON 17 SEPT 1957, JULIAN DATE 2436099.5                 POLE  37
C POSITIONS ARE RELATIVE TO THE MEAN POLE  OF 1900 0.5                  POLE  38
C THE INCREMENT BETWEEN POINTS IS 10 DAYS                               POLE  39
      DATA X1/                                                          POLE  40
     X0.340,0.296,0.248,0.199,0.151,0.103,0.047,-.015,-.085,-.140,-.178,POLE  41
     X-.203,-.220,-.227,-.232,-.232,-.224,-.212,-.193,-.166,-.136,-.104,POLE  42
     X-.071,-.036,0.002,0.042,0.082,0.119,0.153,0.184,0.215,0.245,0.273,POLE  43
     X0.300,0.327,0.353,0.374,0.391,0.403,0.389,0.360,0.323,0.281,0.234,POLE  44
     X0.186,0.133,0.080,0.027,-.026,-.069,-.102,-.124,-.143,-.157,-.168,POLE  45
     X-.175,-.174,-.169,-.161,-.151,-.138,-.123,-.104,-.083,-.060,-.033,POLE  46
     X-.002,0.033,0.074,0.117,0.169,0.221,0.266,0.299,0.318,0.323,0.323,POLE  47
     X0.318,0.309,0.296,0.277,0.255,0.227,0.196,0.163,0.134,0.105,0.076,POLE  48
     X0.048,0.020,-.008,-.034,-.055,-.072,-.086,-.095,-.101,-.101,-.100,POLE  49
     X-.099,-.096,-.091,-.081,-.068,-.050,-.030,-.008,0.012,0.032,0.050,POLE  50
     X0.068,0.086,0.102,0.118,0.132,0.145,0.149/                        POLE  51
      DATA X2/                                                          POLE  52
     X .148, .147, .147, .147, .142, .142, .141, .140, .138, .135, .131,POLE  53
     X .127, .123, .118, .111, .101, .088, .075, .064, .055, .046, .037,POLE  54
     X .028, .019, .011, .003,-.003,-.008,-.012,-.015,-.018,-.021,-.024,POLE  55
```

```
X-.027,-.029,-.031,-.033,-.031,-.025,-.015,-.010, .002, .014, .026,POLE  56
X .038, .050, .062, .076, .091, .106, .119, .131, .142, .152, .161,POLE  57
X .169, .176, .181, .183, .179, .171, .161, .150, .138, .124, .108,POLE  58
X .091, .073, .054, .032, .009,-.014,-.036,-.056,-.068,-.088,-.098,POLE  59
X-.112,-.112,-.108,-.100,-.088,-.073,-.055,-.032,-.004, .027, .058,POLE  60
X .088, .117, .143, .167, .189, .208, .225, .238, .246, .249, .247,POLE  61
X .240, .227, .207, .180, .147, .109, .067, .023,-.019,-.057,-.092,POLE  62
X-.122,-.146,-.164,-.228,-.247,-.262,-.272,-.273,-.265,-.248,-.228,POLE  63
X-.202,-.171,-.137,-.105,-.075,-.046,-.018, .012, .049, .089, .129,POLE  64
X .163, .199, .209, .224, .235, .240, .236, .229, .217, .202, .183,POLE  65
X .157, .123, .085, .048, .018,-.010,-.039,-.073,-.105,-.146,-.176,POLE  66
X-.197,-.209,-.217,-.225,-.228,-.227,-.219,-.204,-.186,-.165,-.142,POLE  67
X-.117,-.088,-.052,-.014, .024, .062, .099, .132, .159, .176, .185,POLE  68
X .192, .197, .200, .199/                                          POLE  69
 DATA X3/                                                          POLE  70
X .195, .186, .167, .147, .126, .102, .076, .049, .020,-.012,-.043,POLE  71
X-.070,-.092,-.107,-.122,-.138,-.154,-.169,-.175,-.173,-.155,-.153,POLE  72
X-.141,-.126,-.108,-.089,-.069,-.048,-.027,-.007, .013, .032, .049,POLE  73
X .065, .081, .098, .113, .124, .127, .125, .121, .112, .097, .081,POLE  74
X .067, .056, .046, .034, .022, .010, .000,-.010,-.020,-.027,-.027,POLE  75
X-.021,-.022,-.022,-.023,-.023,-.021,-.019,-.017,-.013,-.009,-.001,POLE  76
X .008, .004,-.003,-.007,-.009,-.011,-.015,-.021,-.022,-.015,-.015,POLE  77
X-.014,-.009,-.004,-.006,-.006,-.006,-.004, .000, .004, .007, .009,POLE  78
X .011, .014, .018, .022, .028, .037, .047, .055, .062, .068, .074,POLE  79
X .079, .082, .080, .071, .059, .046, .037, .031, .024, .008,-.014,POLE  80
X-.037,-.068,-.101,-.134,-.146,-.149,-.149,-.137,-.119,-.098,-.078,POLE  81
X-.062,-.045,-.034,-.025,-.014,-.001, .014, .028, .046, .062, .080,POLE  82
X .099, .170, .134, .147, .152, .151, .137, .120, .098, .079, .068,POLE  83
X .050, .039, .029, .021, .004,-.020,-.044,-.074,-.103,-.129,-.151,POLE  84
X-.165,-.174,-.173,-.161,-.139,-.119,-.109,-.106,-.101,-.089,-.071,POLE  85
X-.051,-.029, .017, .055, .088, .119, .145, .168, .186, .199, .207,POLE  86
X .209, .205, .199, .191, .183/                                    POLE  87
 DATA X4/                                                          POLE  88
X .160, .120, .078, .048, .024, .000,-.029,-.057,-.085,-.111,-.133,POLE  89
X-.152,-.174,-.196,-.209,-.216,-.212,-.200,-.192,-.183,-.170,-.150,POLE  90
X-.129,-.104,-.061, .002, .060, .103, .140, .170, .198, .224, .245,POLE  91
X .266, .263, .264, .252, .229, .203, .179, .157, .135, .109, .087,POLE  92
X .055, .022,-.014,-.052,-.083,-.107,-.127/                        POLE  93
 DATA Y1/                                                          POLE  94
Y0.057,0.029,0.001,-.025,-.045,-.058,-.068,-.075,-.077,-.073,-.061,POLE  95
Y-.044,-.012,0.029,0.074,0.127,0.194,0.239,0.284,0.330,0.372,0.403,POLE  96
Y0.424,0.441,0.457,0.467,0.473,0.471,0.463,0.451,0.430,0.398,0.365,POLE  97
Y0.325,0.281,0.251,0.219,0.167,0.133,0.082,0.047,-.002,-.030,-.065,POLE  98
Y-.076,-.093,-.082,-.079,-.071,-.060,-.047,-.030,-.007,0.019,0.050,POLE  99
Y0.083,0.119,0.157,0.193,0.229,0.262,0.293,0.321,0.347,0.367,0.381,POLE 100
Y0.384,0.381,0.376,0.366,0.349,0.325,0.296,0.271,0.251,0.233,0.215,POLE 101
Y0.197,0.179,0.159,0.137,0.115,0.095,0.076,0.070,0.056,0.044,0.034,POLE 102
Y0.028,0.028,0.033,0.044,0.059,0.079,0.098,0.116,0.133,0.149,0.165,POLE 103
Y0.179,0.192,0.204,0.216,0.229,0.242,0.257,0.271,0.283,0.293,0.301,POLE 104
Y0.305,0.305,0.303,0.299,0.294,0.284,0.271/                        POLE 105
 DATA Y2/                                                          POLE 106
X .252, .235, .221, .209, .210, .199, .188, .177, .164, .149, .134,POLE 107
X .121, .110, .101, .099, .099, .100, .104, .112, .118, .128, .140,POLE 108
X .154, .170, .187, .203, .207, .210, .213, .215, .217, .219, .220,POLE 109
X .221, .222, .222, .223, .225, .230, .238, .258, .267, .273, .277,POLE 110
X .281, .285, .289, .289, .287, .283, .276, .267, .257, .245, .231,POLE 111
```

```
     X .217, .202, .185, .165, .142, .121, .105, .093, .083, .075, .070,POLE 112
     X .067, .068, .071, .078, .090, .106, .127, .153, .181, .207, .231,POLE 113
     X .259, .279, .29 , .307, .318, .328, .337, .344, .349, .351, .350,POLE 114
     X .344,                       .333, .317, .297, .272, .247, .222,POLE 115
     X .197, .172, .147, .122, .096, .070, .044, .020,-.001,-.017,-.025,POLE 116
     X-.025,-.017,-.003, .016, .040, .068, .101, .134, .162, .190, .220,POLE 117
     X .253, .289, .325, .359, .390, .417, .443, .465, .479, .482, .478,POLE 118
     X .472, .467, .460, .446, .424, .394, .360, .330, .301, .270, .235,POLE 119
     X .199, .155, .136, .111, .092, .074, .058, .043, .052, .054, .055,POLE 120
     X .057, .062, .075, .094, .120, .150, .180, .211, .245, .291, .315,POLE 121
     X .349, .383, .410, .425, .441, .454, .465, .468, .460, .448, .425,POLE 122
     X .401, .377, .350, .323, .299, .275, .249, .222/                  POLE 123
      DATA Y3/                                                          POLE 124
     Y .192, .163, .137, .117, .101, .087, .073, .061, .054, .052, .054,POLE 125
     Y .063, .074, .087, .104, .124, .149, .177, .202, .223, .243, .263,POLE 126
     Y .285, .304, .322, .340, .356, .361, .361, .361, .358, .354, .350,POLE 127
     Y .346, .341, .332, .316, .294, .273, .254, .235, .220, .206, .190,POLE 128
     Y .175, .169, .167, .165, .164, .164, .166, .169, .173, .177, .183,POLE 129
     Y .193, .200, .202, .204, .206, .210, .216, .220, .222, .216, .210,POLE 130
     Y .210, .207, .203, .204, .207, .209, .212, .219, .232, .246, .253,POLE 131
     Y .256, .265, .270, .272, .274, .276, .273, .269, .263, .261, .256,POLE 132
     Y .254, .254, .256, .258, .250, .236, .220, .208, .198, .188, .180,POLE 133
     Y .174, .168, .165, .162, .160, .158, .151, .145, .141, .133, .130,POLE 134
     Y .130, .139, .156, .186, .215, .238, .261, .282, .297, .307, .316,POLE 135
     Y .328, .337, .350, .360, .368, .372, .373, .371, .365, .357, .344,POLE 136
     Y .332, .317, .300, .273, .241, .205, .173, .147, .125, .115, .111,POLE 137
     Y .108, .106, .104, .102, .101, .106, .114, .126, .135, .146, .175,POLE 138
     Y .210, .201, .200, .202, .200, .371, .300, .399, .410, .421, .434,POLE 139
     Y .444, .446, .448, .442, .426, .402, .373, .342, .309, .276, .244,POLE 140
     Y .213, .183, .155, .126, .101/                                    POLE 141
      DATA Y4/                                                          POLE 142
     Y .078, .063, .053, .038, .023, .013, .013, .022, .035, .057, .085,POLE 143
     Y .117, .148, .179, .211, .253, .301, .340, .370, .397, .424, .452,POLE 144
     Y .480, .505, .519, .524, .520, .500, .476, .450, .421, .387, .351,POLE 145
     Y .317, .283, .249, .211, .175, .139, .118, .103, .092, .078, .062,POLE 146
     Y .048, .035, .026, .027, .034, .050, .074/                        POLE 147
C FIRST POINT 570917 *** LAST POINT 711204                              POLE 148
      DATA DFIRST/2436099.500/,                                         POLE 149
     .   DLAST/2441379.500/                                             POLE 150
      LOGICAL FRSTME/.TRUE./                                            POLE 151
      IF(.NOT.FRSTME) GO TO 20                                          POLE 152
      FRSTME = .FALSE.                                                  POLE 153
      IN=(DLAST-DFIRST)/10.D0+1.D0                                      POLE 154
      DO 10 I=1,IN                                                      POLE 155
      X(I)=X(I)*0.4848137E-5                                            POLE 156
      Y(I)=Y(I)*0.4848137E-5                                            POLE 157
   10 CONTINUE                                                          POLE 158
   20 CONTINUE                                                          POLE 159
      D = DJUL(DAY)                                                     POLE 160
      IF(D .GE. DFIRST) GO TO 30                                        POLE 161
      XP = X(1)                                                         POLE 162
      YP = Y(1)                                                         POLE 163
      RETURN                                                            POLE 164
   30 CONTINUE                                                          POLE 165
      IF(D .LT. DLAST) GO TO 40                                         POLE 166
       XP = X(IN)                                                       POLE 167
```

```
      YP = Y(IN)                                                        POLE 168
      RETURN                                                            POLE 169
   40 CONTINUE                                                          POLE 170
      D=(D-DFIRST)/10.0D0+1.0D0                                         POLE 171
C INTERPOLATE FOR CORRDINATES OF THE POLE                               POLE 172
      ID = D                                                            POLE 173
      IDP1 = ID + 1                                                     POLE 174
      D1=D-DFLOAT(IDP1)                                                 POLE 175
      D2=D-DFLOAT(ID)                                                   POLE 176
      YP = -D1*Y(ID) + D2*Y(IDP1)                                       POLE 177
      XP = -D1*X(ID) + D2*X(IDP1)                                       POLE 178
      RETURN                                                            POLE 179
      END                                                               POLE 180
```

B.I.H.
61, Avenue de l'Observatoire
75 - PARIS 14ème

Circular D69

1 - UNIVERSAL TIME AND COORDINATES OF THE POLE

Date (0h UT) 1972	J.D. 2400000.5 +	smoothed values x 0".001	y 0".001	UT2-UTC 0.0001s	UT1-UTC 0.0001s	raw values x 0".001	y 0".001	UT1-UTC 0.0001s	UT1
June 1	41 469	-145	+356	-5252	-5553	-151	+341	-5541	-10.
6	474	-134	+366	-5416	-5710	-129	+371	-5743	
11	479	-120	+376	-5579	-5861	-114	+359	-5872	
16	484	-105	+385	-5740	-6006	- 92	+374	-6019	
21	489	- 89	+394	-5899	-6145	- 92	+421	-6165	
26	494	- 72	+402	-6057	-6279	- 51	+382	-6263	
July 1	499	- 54	+409	+3786	+3591	- 58	+407	+3616	

IAT-UTC is exactly 10s in June 1972
IAT-UTC is exactly 11s since 1972 July 1st, 0h UTC.

2 - EMISSION TIME OF TIME SIGNALS, for June 1972 (E = UTC-Signal in 0.0001s)

Signal	E	Signal	E	Signal	E
CHU	0	FTH42, FTK77, FTN87	0	NSS (o.c.)	+ 9
DAM, DAN, DAO	0	HEG	0	OLBS	(2
DCF77	0	IAM	0	OMA	(2
DGI	0	IBF	+ 3	PPE	- 5
DIZ	0	JJY	0	RWM (1)	0
FFH	0	LOL	- 5	VNG	0
FTA91	0	MSF	+ 1	WWV, WWVB, WWVH	0
		GBZ (3)	- 3	ZUO	(2

(1) and other signals from USSR (2) no data available
(3) corrected values : April 1972, E = - 3 ; May 1972, E = - 2

3 - COORDINATED UNIVERSAL TIME (approximation UTC(i) of UTC, kept by the laboratory
Ref. CCIR Recommendation 458, 1970)

a - From LORAN-C and Television pulses receptions

Date 1972 J.D. 2400000.5 +	June 11 41 479	June 21 41 489	July 1 41 499	
Laboratory i	UTC-UTC(i)	(unit : 1 µs)		
PTB (Braunschweig)	+ 2.9	+ 3.0	+ 2.9	
USNO (Washington)	- 6.6	- 6.5	- 6.2	
CP (Paris)	+ 1.6	+ 1.6	+ 1.6	
NBS (Boulder)	- 2.4	- 2.5	- 2.7	
RCO (Kerstmonceux)	+ 3.2	+ 3.8	+ 4.4	
NRC (Ottawa)	+ 0.8	+ 0.9	+ 1.2	
FOA (Stockholm)	+ 23.9	+ 26.3	+ 28.6	
DFI (Hamburg)	- 16.9	- 15.3	- 13.9	
OM (Neuchâtel)	+ 20.6	+ 20.7	+ 20.6	P. T.

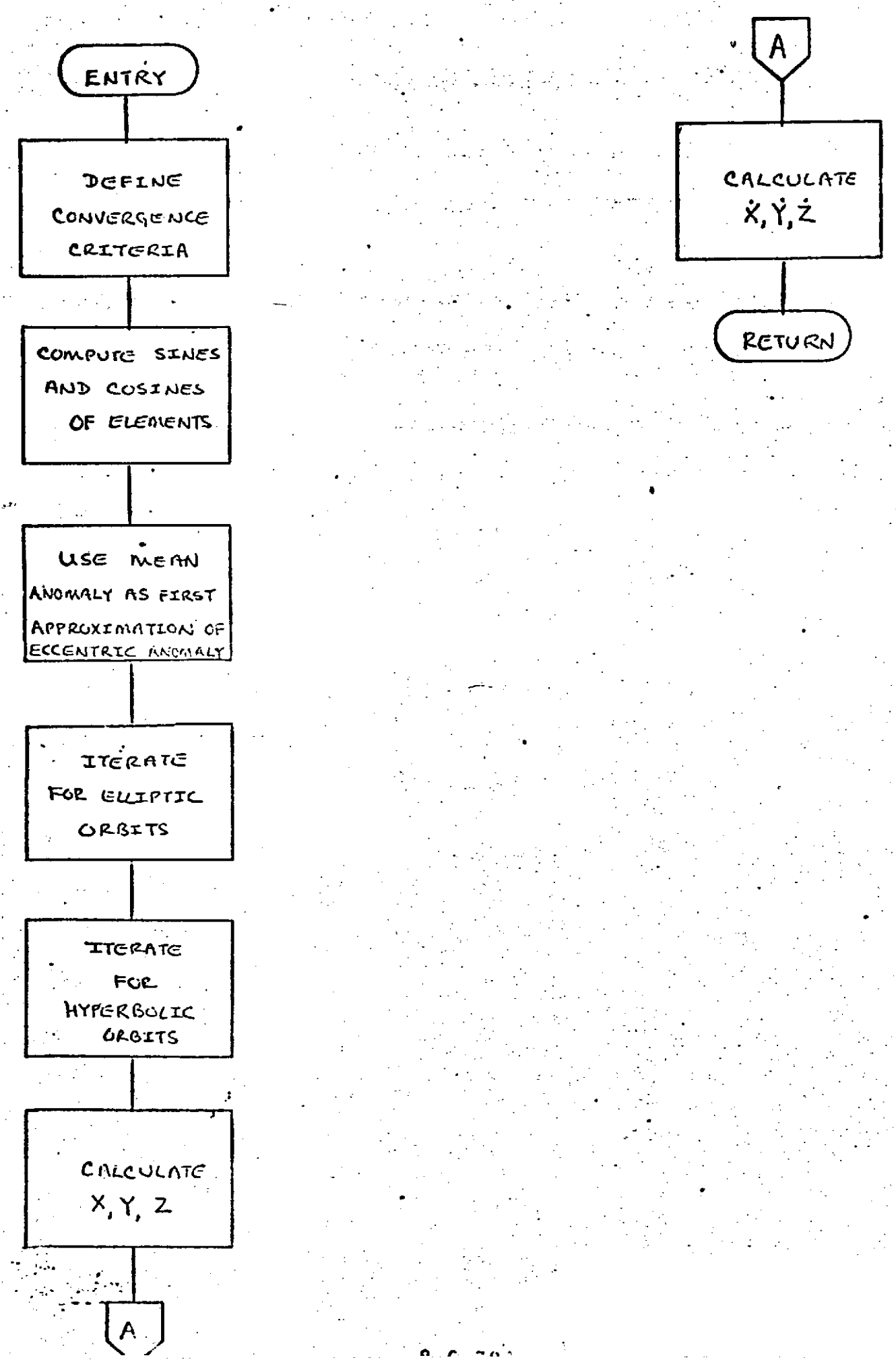
NAME	POSVEL
PURPOSE	TO CONVERT OSCULATING ORBITAL ELEMENTS TO INERTIAL POSITION AND VELOCITY VECTORS
CALLING SEQUENCE	POSVEL(XYZ,AEI,IDRAD)

SYMBOL	TYPE	DESCRIPTION
XYZ (6)	DP	OUTPUT - CARTESIAN ELEMENTS
AEI (6)	DP	INPUT - KEPLER ELEMENTS
IDRAD	I	INPUT - =2 MEANS INPUT IN RADIANS =1 MEANS INPUT IN DEGREES

SUBROUTINES USED	NONE
COMMON BLOCKS	CONSTS INTBLK
INPUT FILES	NONE
OUTPUT FILES	NONE
REFERENCES	"GEODYN SYSTEMS DESCRIPTION" VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE POSVEL(XYZ,AEI,IDRAD)                                  POSV  33
      IMPLICIT REAL*8 (A-H,O-Z)                                         POSV  34
      REAL RMSTOT                                                       POSV  35
      DOUBLE PRECISION INCL,MEAN,NODE                                   POSV  36
      DIMENSION AEI(6),XYZ(6),AEINPM(6),XYZXYZ(6)                       POSV  37
      COMMON/CONSTS/PI,TWOPI,RAD,RSEC                                   POSV  38
      COMMON/INTBLK/THDOTS(3),GM,AE(62)                                 POSV  39
      EQUIVALENCE (A,AEINPM(1)),(E,AEINPM(2)),(INCL,AEINPM(3)),         POSV  40
     *   (NODE,AEINPM(4)),(P,AEINPM(5)),(MEAN,AEINPM(6)),               POSV  41
     *   (X,XYZXYZ(1)),(Y,XYZXYZ(2)),(Z,XYZXYZ(3)),                     POSV  42
     *   (XDOT,XYZXYZ(4)),(YDOT,XYZXYZ(5)),(ZDOT,XYZXYZ(6))             POSV  43
C DEFINE CONVERSION CRITERIA                                            POSV  44
      DATA DELTA/0.1D-10/                                               POSV  45
      SCALE=1.0D0                                                       POSV  46
      IF(IDRAD.EQ.1) SCALE=RAD                                          POSV  47
      DO 10 I=1,6                                                       POSV  48
      AEINPM(I)=AEI(I)                                                  POSV  49
      IF(I.LE.2) GO TO 10                                               POSV  50
      AEINPM(I)=AEI(I)*SCALE                                            POSV  51
   10 CONTINUE                                                          POSV  52
      SQMUA=DSQRT(GM/DABS(A)**3)                                        POSV  53
      E2=E**2                                                           POSV  54
      ONEME2=DSQRT(DABS(1.0D0-E2))                                      POSV  55
```

```
C SINES AND COSINES OF THE ELEMENTS                                    POSV  56
      COSI=DCOS(INCL)                                                  POSV  57
      SINI=DSIN(INCL)                                                  POSV  58
      SINN=DSIN(NODE)                                                  POSV  59
      COSN=DCOS(NODE)                                                  POSV  60
      SINP=DSIN(P)                                                     POSV  61
      COSP=DCOS(P)                                                     POSV  62
C SET ECC. ANOM. EQUAL TO MEAN ANOM. FOR FIRST APROX.                  POSV  63
      ECC=MEAN                                                         POSV  64
C ITERATE                                                              POSV  65
      IF (E2.GE.1.0D0) GO TO 150                                       POSV  66
C ...FOR ELLIPTIC ORBITS                                               POSV  67
      DO 100 J=1,50                                                    POSV  68
      E00=ECC                                                          POSV  69
      SINECC=DSIN(E00)                                                 POSV  70
      COSECC=DCOS(E00)                                                 POSV  71
      ECOS=1.0D0-E*COSECC                                              POSV  72
      ECC=E00-(E00-E*SINECC-MEAN)/ECOS                                 POSV  73
      IF(DABS(E00-ECC).LT.DELTA) GO TO 200                             POSV  74
  100 CONTINUE                                                         POSV  75
      PRINT 1000                                                       POSV  76
      GO TO 200                                                        POSV  77
C ...FOR HYPERBOLIC ORBITS                                             POSV  78
  150 DO 160 J=1,100                                                   POSV  79
      E00=ECC                                                          POSV  80
      SINECC=DSINH(E00)                                                POSV  81
      COSECC=DCOSH(E00)                                                POSV  82
      ECOS=E*COSECC-1.0D0                                              POSV  83
      ECC=E00-(E*SINECC-E00-MEAN)/ECOS                                 POSV  84
      IF (DABS(E00-ECC).LT.DELTA) GO TO 200                            POSV  85
  160 CONTINUE                                                         POSV  86
      PRINT 1000                                                       POSV  87
  200 SPCN=SINP*COSN                                                   POSV  88
      CPSN=COSP*SINN                                                   POSV  89
      CPCN=COSP*COSN                                                   POSV  90
      SPSN=SINP*SINN                                                   POSV  91
      A2=DABS(A)*ONEME2                                                POSV  92
      AX=A*(CPCN-SPSN*COSI)                                            POSV  93
      AY=A*(SPCN*COSI+CPSN)                                            POSV  94
      AZ=A*SINP*SINI                                                   POSV  95
      BX=-A2*(SPCN+CPSN*COSI)                                          POSV  96
      BY=A2*(CPCN*COSI-SPSN)                                           POSV  97
      BZ=A2*COSP*SINI                                                  POSV  98
      C=ECCSECC-E                                                      POSV  99
      EDOT=SQMUA/ECOS                                                  POSV 100
C ...FOR X,Y,Z                                                         POSV 101
      X=AX*C+BX*SINECC                                                 POSV 102
      Y=AY*C+BY*SINECC                                                 POSV 103
      Z=AZ*C+BZ*SINECC                                                 POSV 104
C ...FOR XDOT,YDOT,ZDOT                                                POSV 105
      IF (E2.GE.1.0D0) SINECC=-SINECC                                  POSV 106
      XDOT=EDOT*(BX*COSECC-AX*SINECC)                                  POSV 107
      YDOT=EDOT*(BY*COSECC-AY*SINECC)                                  POSV 108
      ZDOT=EDOT*(BZ*COSECC-AZ*SINECC)                                  POSV 109
      DO 300 I=1,6                                                     POSV 110
  300 XYZ(I)=XYZXYZ(I)                                                 POSV 111
      RETURN                                                           POSV 112
 1000 FORMAT(1H1,31HECCENTRIC ANOMALY NOT CONVERGED)                   POSV 113
      END                                                              POSV 114
```


ENTRY
DEFINE CONVERGENCE CRITERIA
COMPUTE SINES AND COSINES OF ELEMENTS
USE MEAN ANOMALY AS FIRST APPROXIMATION OF ECCENTRIC ANOMALY
ITERATE FOR ELLIPTIC ORBITS
ITERATE FOR HYPERBOLIC ORBITS
CALCULATE X, Y, Z
A
A
CALCULATE Ẋ, Ẏ, Ż
RETURN

PRECES

DESCRIPTION

Subroutine PRECES generates the rotation matrix to precess a vector from the mean equator and equinox of an input date to the mean equator and equinox of 1950.0.

The precession angles are evaluated using polynomials derived by Simon Newcomb. The rotation matrices are evaluated by ROTMAT; the output rotation matrix is computed as the product of the three input matrices by MULMAT.

NAME	PRECES
PURPOSE	TO GENERATE THE MATRIX FOR PRECESSION FROM MEAN EQUATOR AND EQUINOX OF AN EPOCH TO MEAN EQUATOR AND EQUINOX OF 1950
CALLING SEQUENCE	CALL PRECES(DAY, X)

SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT - TIME IN DAYS FROM JAN 0.0 OF REFERENCE YEAR
X (3,3)	DP	OUTPUT - PRECESSION MATRIX

SUBROUTINES USED	ROTMAT MULMAT YMDAY
COMMON BLOCKS	INITBK
INPUT FILES	NONE
OUTPUT FILES	NONE
REFERENCES	"GEODYN SYSTEMS DESCRIPTION" VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE PRECES(DAY,X)                                          PREC  31
      REAL*8 DAY,DBASE,D,X(3,3),Z(3,3,3),ANGLE,YMDAY                   PREC  32
      REAL*8 COEF(3,3)/+.30595320465D-6,+.39720490D-14,+.191031D-20,    PREC  33
     *               -.26603999754D-6,+.15481180D-14,+.41390C2D-20,     PREC  34
     *               +.30595320465D-6,+.10974920D-14,+.178097D-20/      PREC  35
      INTEGER AXIS(3)/3,2,3/                                            PREC  36
      COMMON/INITBK/IG1(53),NOTIST,IG2(3)                               PREC  37
      LOGICAL NOTIST                                                    PREC  38
      IF(NOTIST) GO TO 10                                               PREC  39
      NOTIST=.TRUE.                                                     PREC  40
      DBASE=YMDAY(50C100.0,0.0D0)-.072D0                                PREC  41
10    D=DAY-DBASE                                                       PREC  42
      DO 30 I=1,3                                                       PPEC  43
      ANGLE=0.D0                                                        PPEC  44
      DO 20 J=1,3                                                       PREC  45
20     ANGLE=(ANGLE+COEF(4-J,I))*D                                      PREC  46
30    CALL ROTMAT(ANGLE,AXIS(I),Z(1,1,I))                               PREC  47
      CALL MULMAT(X,Z(1,1,3),Z(1,1,2),Z(1,1,1))                         PREC  48
      RETURN                                                            PPEC  49
      END                                                               PREC  50
```

PREDCT

DESCRIPTION

Subroutine PREDCT computes the residuals and partial derivatives for observations involving fewer than two tracking stations for the parameter estimation. In addition, PREDCT predicts the measurement values from a priori data and the geodetic spherical coordinates of the satellite.

The observation types by program index number are:

1) right ascension and declination
2) range (including sat-sat summed range)
3) range-rate (including sat-sat summed range rate)
4) altimeter height and height rate
5) ℓ and m direction cosines
6) X and Y angles
7) azimuth and elevation

The order of computation is as follows:

- Call ORBIT to obtain the satellite position, velocity, and variational partials.
- Call GRHRAN to obtain the right ascension of Greenwich and the station-satellite vector for observing station.
- Compute the equivalent for each measurement and the associated residual.
- Compute the Earth-fixed geometric partial derivatives of the calculated equivalents.

- Convert the partial derivatives to inertial coordinates and chain them back to epoch,

- If necessary compute the spheroid height, geodetic latitude, and east longitude of the satellite.

NAME PREDCT

ENTRY POINT	PURPOSE
PREDC1	INITIALIZATION
PREDCT	TO COMPUTE MEASUREMENTS, RESIDUALS (O-C), AND MEASUREMENT PARTIALS

CALLING SEQUENCE CALL PREDC1(EHAT,NHAT,ZHAT,PMPX0,PXPX0,NPARM, NEQNMX)

SYMBOL	TYPE	DESCRIPTION
EHAT (3,1)	DP	INPUT - STATION UNIT EAST VECTOR
NHAT (3,1)	DP	INPUT - STATION UNIT NORTH VECTOR
ZHAT (3,1)	DP	INPUT - STATION UNIT VERTICAL VECTOR
PMPX0 (NPARM,1)	DP	OUTPUT - MEASUREMENT PARTIALS
PXPX0 (6,NEQNMX,2)	DP	INPUT - SATELLITE STATE PARTIAL WRT EPOCH PARAMETERS
NPARM	I	INPUT - MAXIMUM NUMBER OF PARAMETERS PER MEASUREMENTS
NEQNMX	I	INPUT - NPARM+8

CALLING SEQUENCE CALL PREDCT(ISTA,DAY,RESID1,RESID2,DATASW)

SYMBOL	TYPE	DESCRIPTION
ISTA	I	INPUT - STATION INDEX
DAY	DP	INPUT & OUTPUT - MEASUREMENT TIME
RESID1	R	OUTPUT - FIRST MEASUREMENT RESIDUAL (O-C)
RESID2	R	OUTPUT - SECOND MEASUREMENT RESIDUAL
DATASW	L	INPUT - .TRUE. WHEN POSITION OF SATELLITE WANTED .FALSE. WHEN MEASUREMENT PARTIALS WANTED

SUBROUTINES USED	ORBIT	GRHRAN	PROCES	ELEM	CLEAR
	DOTPRD	DARCTN	XINERT	XEFIX	YINERT
	YEFIX				
COMMON BLOCKS	APARAM	CONSTS	CORR1	CUVECT	GNDTRK
	INITBK	INTBLK	PREBLK	XYZOUT	

INPUT FILES NONE

OUTPUT FILES NONE

REFERENCES *GEODYN SYSTEMS DESCRIPTION*
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE PREDC1(EHAT,NHAT,ZHAT,PMPX0,PXPX0,NPARM,NEQNMX)                PRED  67
      IMPLICIT REAL*8 (A-H,O-Z)                                                 PRED  68
      LOGICAL*1 VHFCHN,PREPRO,NOT1ST,TWOSTA                                     PRED  69
      LOGICAL NOEST,SATSW,SUBSAT,DATASW,TRKSW,SATSAT,ELEVSW                     PRED  70
      INTEGER*2 MTYPE,NMEAS,PRETYP,CHANEL,ISAT                                  PRED  71
      INTEGER RECNO,ADDR                                                        PRED  72
      DOUBLE PRECISION NHAT,LOVE                                                PRED  73
      DIMENSION PMSTA1(3),PMSTA2(3),OBSC(2),XYZDOT(3),PM(6,4),G(3),             PRED  74
     .   G2(3,3),R2(3,3),PARH2(3,3),PMSTA3(3,2),PMPX0(NPARM,1),                 PRED  75
     .   PXPX0(6,NEQNMX,2),PMSTA(3,2),EHAT(3,1),NHAT(3,1),ZHAT(3,1),            PRED  76
     .   URHOO(3),DR(3),H(3),HI(3),AEIXYZ(6,6),P12(3),PS2(3),PS1(3),            PRED  77
     .   PS3(3),P23(3),V12(3),VS2(3),VS1(3),VS3(3),V23(3)                       PRED  78
      COMMON/APARAM/NPAR,INPART(3),NSAT,NGPARC(5)                               PRED  79
      COMMON/CONSTS/OPI,DTWOPI,DRAD,DRSEC                                       PRED  80
      COMMON/CORB1/T,W,U,THETG,PERHT(2),APHT(2),PRD(2)                          PRED  81
      COMMON/CJVCCT/UHAT(3,2),XYZ(3,2),RXYZ(3,2),RENV(3,2),R(2),RSO(2),         PRED  82
     .   XYSQ(2)                                                               PRED  83
      COMMON/GNDTRK/SATLAT(2),SATLON(2),SATH(2),ELEV(2),SATSW                   PRED  84
      COMMON/INITBK/IG1(42),SUBSAT,IG2(5),MISLOG(9)                             PRED  85
      COMMON/INTBLK/THDOT1(2),THDT2S,GM,AE,AFSQ,FLAT,FSQ32,FFSQ32,              PRED  86
     .   GM3(49),NEQNS(2),ADDR(2,3),LOVE(4)                                    PRED  87
      COMMON/PREBLK/DAYSTA,OBSQ(2),SIG(2),SRENDX,ISN,MTYPE,NMEAS,               PRED  88
     .   ISAT,PRETYP,CHANEL,VHFCHN,PREPRO,RECNO                                PRED  89
      COMMON/XYZOUT/XYZI(6,4)                                                   PRED  90
      DATA NOT1ST/.FALSE./                                                      PRED  91
      DATA C/2.99792508/,DTOL/1.0D-9/                                           PRED  92
      EQUIVALENCE (F,FLAT),(PMSTA1(1),PMSTA(1,1)),(PMSTA2(1),PMSTA(1,2))        PRED  93
      EQUIVALENCE (PRD(1),TRKSW),(AEIXYZ(1,1),URHOO(1)),                        PRED  94
     .   (AEIXYZ(1,2),DR(1)),(AEIXYZ(1,3),H(1)),(AEIXYZ(1,4),HI(1))             PRED  95
      EQUIVALENCE (DTRANS,OBSQ(2)),(KKSAT,SIG(2)),(TIME2,SRENDX)                PRED  96
      RETURN                                                                    PRED  97
      ENTRY PREDCT(ISTA,DAY,RESID1,RESID2,DATASW)                               PRED  98
      IF(NOT1ST) GO TO 10                                                       PRED  99
C INITIALIZE                                                                    PRED 100
      C1=1.5D0*AE*F*F                                                           PRED 101
      C2=AE*F+C1                                                                PRED 102
      FLAT21=(1.0D0-FLAT)**2                                                    PRED 103
      NOT1ST=.TRUE.                                                             PRED 104
   10 CONTINUE                                                                  PRED 105
      SATSAT=(.NOT.DATASW).AND.(MTYPE.EQ.2.OR.MTYPE.EQ.3).AND.KKSAT.GT.0PRED 106
     .   .AND.KKSAT.LE.NSAT                                                    PRED 107
      ISAT1=ISAT                                                                PRED 108
      ISAT2=ISAT                                                                PRED 109
      INCR=1                                                                    PRED 110
      K2=0                                                                      PRED 111
```

```
      ELEVSW=PREPRO                                                    PRED 112
C IF SAT-SAT TRACKING THEN GO TO 300                                   PRED 113
      IF(SATSAT) GO TO 300                                             PRED 114
      TWOSTA=MTYPE.GT.26.AND..NOT.DATASW                               PRED 115
      NOEST=.FALSE.                                                    PRED 116
C FOR VLBI & AVERAGE RANGE RATE DATA SKIP INTEGRATOR CALL              PRED 117
      IF(TWOSTA) GO TO 50                                              PRED 118
C OBTAIN SATELLITE ORBIT                                               PRED 119
      CALL ORBIT(DAY)                                                  PRED 120
C OBTAIN G.A. GREENWICH & STATION - SATELLITE VECTORS                  PRED 121
      THETG=GPHRAN(DAY,ISTA)                                           PRED 122
C SKIP MEASUREMENTS & PARTIALS IF ONLY GROUND TRACK REQUESTED          PRED 123
      IF(DATASW) GO TO 2020                                            PRED 124
      IF(ISTA.EQ.0) GO TO 200                                          PRED 125
      NOEST=NPAR.EQ.0.OR.(SIG(1).EQ.0.0D0.AND.SIG(2).EQ.0.0D0)         PRED 126
   50 NP=3                                                             PRED 127
C COMPUTE ELEVATION                                                    PRED 128
      ENSQ=1.0D0-RENV(3,ISAT)**2                                       PRED 129
      EN=DSQRT(ENSQ)                                                   PRED 130
      ELEV(ISAT)=DATAN(RENV(3,ISAT)/EN)                                PRED 131
C IF VLBI OR AVERAGE RANGE RATE THEN GO TO 230                         PRED 132
      IF(TWOSTA) GO TO 230                                             PRED 133
C PROCESS DATA IF REQUESTED                                            PRED 134
      IF(PREPRO) CALL PROCES(ISTA,DAY,THETG)                           PRED 135
C IF TRANSIT TIME CORRECTION WAS MADE THEN RECOMPUTE ELEVATION         PRED 136
      IF(.NOT.ELEVSW) GO TO 190                                        PRED 137
      ENSQ=1.0D0-RENV(3,ISAT)**2                                       PRED 138
      EN=DSQRT(ENSQ)                                                   PRED 139
      ELEV(ISAT)=DATAN(RENV(3,ISAT)/EN)                                PRED 140
  190 GO TO (400,450,500,550,600,700,800),MTYPE                        PRED 141
C PCE DATA                                                             PRED 142
  200 DO 210 I=1,6                                                     PRED 143
  210 PM(I,1)=0.0D0                                                    PRED 144
      MT=MOD(MTYPE-15,6)+1                                             PRED 145
      PM(MT,1)=1.0D0                                                   PRED 146
      RESID1=OBSO(1)-XYZI(MT,ISAT)                                     PRED 147
      ELEV(ISAT)=0.5D0*DPI                                             PRED 148
      NP=6                                                             PRED 149
      IF(MTYPE.LT.21) GO TO 1900                                       PRED 150
C CALL ELEM FOR KEPLER DATA                                            PRED 151
      CALL ELEM(XYZI(1,ISAT),PMSTA1,3,.FALSE.,AEIXYZ)                  PRED 152
      RESID1=OBSO(1)-PMSTA1(MT)                                        PRED 153
      IF(DABS(RESID1).GT.DPI) RESID1=RESID1-DSIGN(DTWOPI,RESID1)       PRED 154
      DO 220 I=1,6                                                     PRED 155
  220 PM(I,1)=AEIXYZ(MT,I)                                             PRED 156
      GO TO 1900                                                       PRED 157
  230 KTYPE=MTYPE-26                                                   PRED 158
      GO TO (240,500,450,450),KTYPE                                    PRED 159
C TIME DELAY                                                           PRED 160
  240 RESID1=R(ISAT)/C                                                 PRED 161
      DO 250 I=1,3                                                     PRED 162
  250 PMSTA1(I)=-UHAT(I,ISAT)/C                                        PRED 163
      GO TO 1900                                                       PRED 164
C IF SAT-SAT DATA THEN COMPUTE UPLINK & DOWNLINK TRANSIT TIMES         PRED 165
  300 NOEST=NPAR.EQ.0.OR.SIG(1).EQ.0.0D0                               PRED 166
      NP=3                                                             PRED 167
```

```
      J=ISAT                                                                    PRED 168
      ISAT1=MIND(J,KKSAT)                                                       PRED 169
      ISAT2=MAXO(J,KKSAT)                                                       PRED 170
  301 IPRE=0                                                                    PRED 171
      IF(.NOT.PREPRO) GO TO 380                                                 PRED 172
      IPRE=PRETYP/10                                                            PRED 173
      PRETYP=PRETYP-IPRE*10                                                     PRED 174
      IF(IPRE.GT.0) GO TO 310                                                   PRED 175
      CALL ORBIT(DAY)                                                           PRED 176
      THETG=GRHRAN(DAY,ISTA)                                                    PRED 177
      DO 302 I=1,3                                                              PRED 178
  302 PS2(I)=RXYZ(I,ISAT)                                                       PRED 179
      KK=0                                                                      PRED 180
      DAYP=DAY                                                                  PRED 181
  304 R12=0.0D0                                                                 PRED 182
      DO 306 I=1,3                                                              PRED 183
      P12(I)=PS2(I)-RXYZ(I,KKSAT)                                               PRED 184
  306 R12=R12+P12(I)**2                                                         PRED 185
      R12=DSQRT(R12)                                                            PRED 186
      DTRANS=DAY+R12/(C*8.64D4)                                                 PRED 187
      DT=DABS(DTRANS-DAYP)                                                      PRED 188
      IF(DT.LE.DTOL) GO TO 380                                                  PRED 189
      KK=KK+1                                                                   PRED 190
      IF(KK.GT.5) GO TO 308                                                     PRED 191
      DAYP=DTRANS                                                               PRED 192
      CALL ORBIT(DTRANS)                                                        PRED 193
      THETG=GRHRAN(DAY,ISTA)                                                    PRED 194
      GO TO 304                                                                 PRED 195
  308 DT=DT*8.64D4                                                              PRED 196
      PRINT 3000,DT,DTOL                                                        PRED 197
  380 KK=0                                                                      PRED 198
      IF(ELEVSW) TIME2=DAY                                                      PRED 199
  382 DAYP=TIME2                                                                PRED 200
      CALL ORBIT(TIME2)                                                         PRED 201
      THETG=GRHRAN(DAY,ISTA)                                                    PRED 202
      DO 384 I=1,3                                                              PRED 203
  384 PS3(I)=RXYZ(I,KKSAT)                                                      PRED 204
      IF(.NOT.ELEVSW) GO TO 390                                                 PRED 205
      R23=0.0D0                                                                 PRED 206
      DO 386 I=1,3                                                              PRED 207
      P23(I)=PS3(I)-PS2(I)                                                      PRED 208
  386 R23=R23+P23(I)**2                                                         PRED 209
      R23=DSQRT(R23)                                                            PRED 210
      TIME2=DAY-R23/(C*8.64D4)                                                  PRED 211
      DT=DABS(TIME2-DAYP)                                                       PRED 212
      IF(DT.LE.DTOL) GO TO 390                                                  PRED 213
      KK=KK+1                                                                   PRED 214
      IF(KK.LE.5) GO TO 382                                                     PRED 215
      DT=DT*8.64D4                                                              PRED 216
      PRINT 3000,DT,DTOL                                                        PRED 217
  390 IF(MTYPE.NE.3) GO TO 310                                                  PRED 218
      VS3(1)=XEFIX(XYZI(4,KKSAT),XYZI(5,KKSAT))+THDT2S*XYZ(2,KKSAT)             PRED 219
      VS3(2)=YEFIX(XYZI(4,KKSAT),XYZI(5,KKSAT))-THDT2S*XYZ(1,KKSAT)             PRED 220
      VS3(3)=XYZI(6,KKSAT)                                                      PRED 221
  310 KK=0                                                                      PRED 222
  318 DAYP=DTRANS                                                               PRED 223
```

```
      CALL ORBIT(DTRANS)                                                    PRED 224
      THETG=GRHRAN(DAY,ISTA)                                                PRED 225
      RS1=0.000                                                             PRED 226
      DO 320 I=1,3                                                          PRED 227
      PS1(I)=RXYZ(I,KKSAT)                                                  PRED 228
  320 RS1=RS1+PS1(I)**2                                                     PRED 229
      RS1=DSQRT(RS1)                                                        PRED 230
      IF(IPRE.LE.0) GO TO 324                                               PRED 231
      KK=KK+1                                                               PRED 232
      IF(KK.GT.5) GO TO 322                                                 PRED 233
      DTRANS=DAY-RS1/(C*8.64D4)                                             PRED 234
      DT=DABS(DAYP-DTRANS)                                                  PRED 235
      IF(DT.GT.DTOL) GO TO 318                                              PRED 236
      GO TO 323                                                             PRED 237
  322 DT=DT*8.64D4                                                          PRED 238
      PRINT 3000,DT,DTOL                                                    PRED 239
  323 DAY=DTRANS                                                            PRED 240
  324 ENSQ=1.000-RENV(3,KKSAT)**2                                           PRED 241
      EN=DSQRT(ENSQ)                                                        PRED 242
      ELEV(ISAT)=DATAN(RENV(3,KKSAT)/EN)                                    PRED 243
      ELEV(KKSAT)=ELEV(ISAT)                                                PRED 244
      PREPRO=PREPRC.AND.PRETYP.GT.0                                         PRED 245
      IF(.NOT.PREPRO) GO TO 325                                             PRED 246
      J=ISAT                                                                PRED 247
      ISAT=KKSAT                                                            PRED 248
      CALL PROCES(ISTA,DTRANS,THETG)                                        PRED 249
      THETG=GRHRAN(DAY,ISTA)                                                PRED 250
      ISAT=J                                                                PRED 251
  325 IF(MTYPE.NE.3) GO TO 326                                              PRED 252
      NP=6                                                                  PRED 253
      VS1(1)=XEFIX(XYZI(4,KKSAT),XYZI(5,KKSAT))+THDT2S*XYZ(2,KKSAT)         PRED 254
      VS1(2)=YEFIX(XYZI(4,KKSAT),XYZI(5,KKSAT))-THDT2S*XYZ(1,KKSAT)         PRED 255
      VS1(3)=XYZI(6,KKSAT)                                                  PRED 256
  326 KK=0                                                                  PRED 257
  328 DAYP=DAY                                                              PRED 258
      CALL ORBIT(DAY)                                                       PRED 259
      THETG=GRHRAN(DAY,ISTA)                                                PRED 260
      R12=0.000                                                             PRED 261
      DO 330 I=1,3                                                          PRED 262
      PS2(I)=RXYZ(I,ISAT)                                                   PRED 263
      P12(I)=PS2(I)-PS1(I)                                                  PRED 264
  330 R12=R12+P12(I)**2                                                     PRED 265
      R12=DSQRT(R12)                                                        PRED 266
      IF(IPRE.LE.0) GO TO 334                                               PRED 267
      KK=KK+1                                                               PRED 268
      IF(KK.GT.5) GO TO 332                                                 PRED 269
      DAY=DTRANS-R12/(C*8.64D4)                                             PRED 270
      DT=DABS(DAYP-DAY)                                                     PRED 271
      IF(DT.GT.DTOL) GO TO 328                                              PRED 272
      GO TO 331                                                             PRED 273
  332 DT=DT*8.64D4                                                          PRED 274
      PRINT 3000,DT,DTOL                                                    PRED 275
      GO TO 331                                                             PRED 276
  334 RS1I=1.000/RS1                                                        PRED 277
      R12I=1.000/R12                                                        PRED 278
      IF(MTYPE.NE.3) GO TO 338                                              PRED 279
```

```
      VS2(1)=XEFIX(XYZI(4,ISAT),XYZI(5,ISAT))+THDT2S*XYZ(2,ISAT)          PRED 280
      VS2(2)=YEFIX(XYZI(4,ISAT),XYZI(5,ISAT))-THDT2S*XYZ(1,ISAT)          PRED 281
      VS2(3)=XYZI(6,ISAT)                                                 PRED 282
      DO 336 I=1,3                                                        PRED 283
      V12(I)=(VS2(I)-VS1(I))*R12I                                         PRED 284
  336 VS1(I)=VS1(I)*RS1I                                                  PRED 285
      RRS12=DOTPRD(P12,V12)+DOTPRD(PS1,VS1)                               PRED 286
  338 DO 340 I=1,3                                                        PRED 287
      P12(I)=P12(I)*R12I                                                  PRED 288
  340 PS1(I)=PS1(I)*RS1I                                                  PRED 289
      IF(NPARM.GT.0) CALL CLEAR(PMPXC,NPARM,4)                            PRED 290
      KK=1                                                                PRED 291
      JJ=3                                                                PRED 292
      IF(ISAT.EQ.ISAT1) GO TO 345                                         PRED 293
      KK=3                                                                PRED 294
      JJ=1                                                                PRED 295
  345 R23=0.000                                                           PRED 296
      RS3=0.000                                                           PRED 297
      DO 351 I=1,3                                                        PRED 298
      P23(I)=PS3(I)-PS2(I)                                                PRED 299
      R23=R23+P23(I)**2                                                   PRED 300
  351 RS3=RS3+PS3(I)**2                                                   PRED 301
      R23=DSQRT(R23)                                                      PRED 302
      RS3=DSQRT(RS3)                                                      PRED 303
      IF(MTYPE-2) 350,350,360                                             PRED 304
C COMPUTE RESIDUAL & PARTIALS FOR SUMMED RANGE                            PRED 305
  350 RESID1=OBSQ(1)-0.500*(RS1+R12+R23+RS3)                              PRED 306
      IF(NOEST) GO TO 2010                                                PRED 307
      DO 352 I=1,3                                                        PRED 308
  352 PMSTA1(I)=-PS1(I)                                                   PRED 309
      PM(1,KK)=XINERT(P12(1),P12(2))                                      PRED 310
      PM(2,KK)=YINERT(P12(1),P12(2))                                      PRED 311
      PM(1,JJ)=XINERT(PS1(1),PS1(2))-PM(1,KK)                             PRED 312
      PM(2,JJ)=YINERT(PS1(1),PS1(2))-PM(2,KK)                             PRED 313
      PM(3,KK)=P12(3)                                                     PRED 314
      PM(3,JJ)=PS1(3)-P12(3)                                              PRED 315
      GO TO 1930                                                          PRED 316
C COMPUTE RESIDUAL & PARTIALS FOR SUMMED RANGE RATE                       PRED 317
  360 RS3I=1.000/RS3                                                      PRED 318
      R23I=1.000/R23                                                      PRED 319
      DO 361 I=1,3                                                        PRED 320
      V23(I)=(VS3(I)-VS2(I))*R23I                                         PRED 321
  361 VS3(I)=VS3(I)*RS3I                                                  PRED 322
      RESID1=OBSQ(1)-0.500*(RRS12+DOTPRD(P23,V23)+DOTPRD(PS3,VS3))        PRED 323
      IF(NOEST) GO TO 2010                                                PRED 324
      DP=DOTPRD(PS1,VS1)                                                  PRED 325
      DO 362 I=1,3                                                        PRED 326
  362 PMSTA1(I)=-VS1(I)+DP*PS1(I)                                         PRED 327
      PM(4,KK)=XINERT(P12(1),P12(2))                                      PRED 328
      PM(5,KK)=YINERT(P12(1),P12(2))                                      PRED 329
      PM(6,KK)=P12(3)                                                     PRED 330
      PM(4,JJ)=XINERT(PS1(1),PS1(2))-PM(4,KK)                             PRED 331
      PM(5,JJ)=YINERT(PS1(1),PS1(2))-PM(5,KK)                             PRED 332
      PM(6,JJ)=PS1(3)-P12(3)                                              PRED 333
      DP=DOTPRD(P12,V12)                                                  PRED 334
      DO 364 I=1,3                                                        PRED 335
```

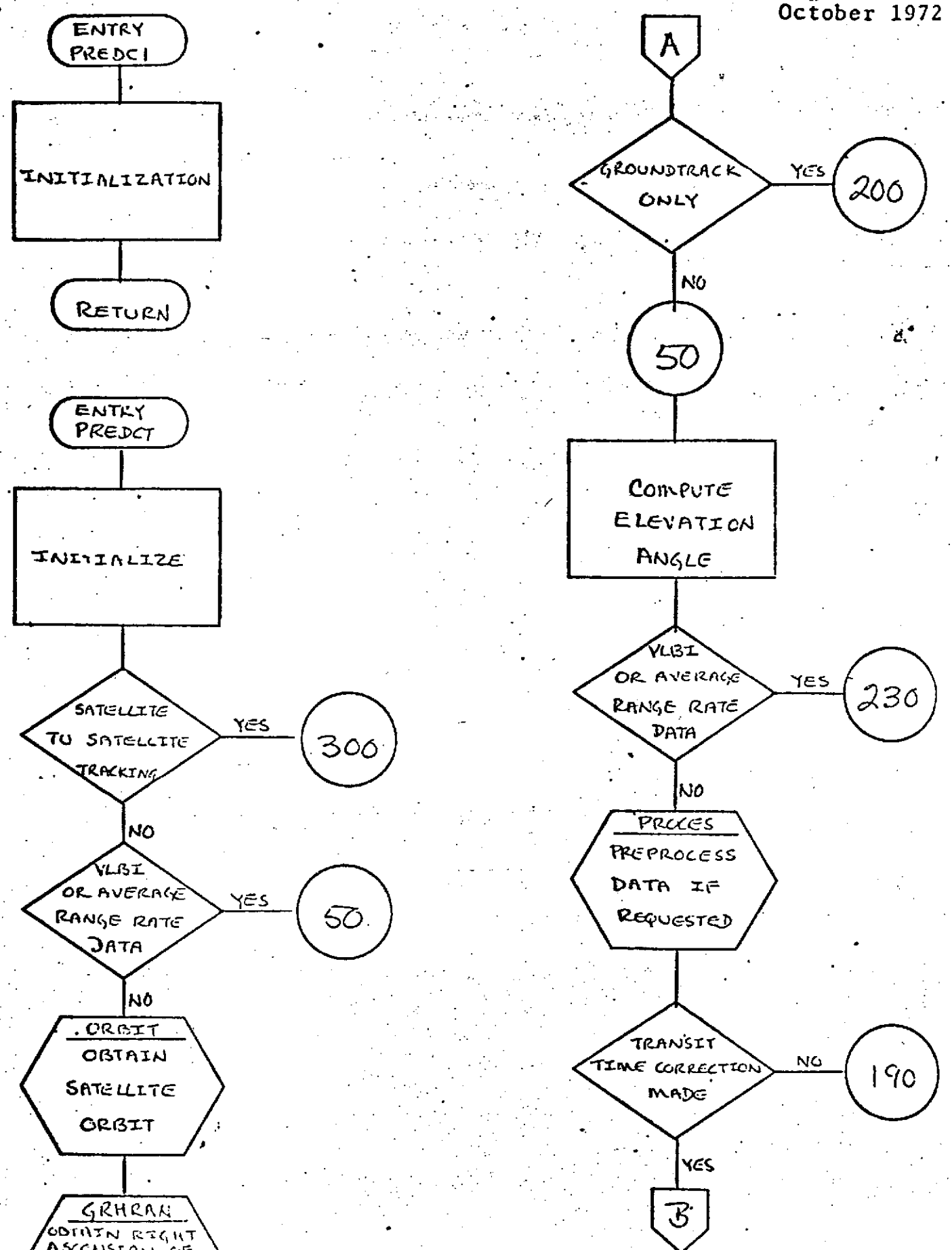
```
  364 PMSTA2(I)=V12(I)-DP*P12(I)                                          PRED 336
      PM(1,KK)=XINERT(PMSTA2(1),PMSTA2(2))                                PRED 337
      PM(2,KK)=YINERT(PMSTA2(1),PMSTA2(2))                                PRED 338
      PM(3,KK)=PMSTA2(3)                                                  PRED 339
      PM(1,JJ)=-XINERT(PMSTA1(1),PMSTA1(2))-PM(1,KK)                      PRED 340
      PM(2,JJ)=YINERT(PMSTA1(1),PMSTA1(2))-PM(2,KK)                       PRED 341
      PM(3,JJ)=-PMSTA1(3)-PMSTA2(3)                                       PRED 342
      PM(1,KK)=PM(1,KK)-THDT2S*PM(5,KK)                                   PRED 343
      PM(2,KK)=PM(2,KK)+THDT2S*PM(4,KK)                                   PRED 344
      PM(1,JJ)=PM(1,JJ)-THDT2S*PM(5,JJ)                                   PRED 345
      PM(2,JJ)=PM(2,JJ)+THDT2S*PM(4,JJ)                                   PRED 346
      GO TO 1930                                                          PRED 347
C RIGHT ASCENSION AND DECLINATION                                         PRED 348
  400 XY=DSQRT(XYSQ(ISAT))                                                PRED 349
      OBSC(1)=DATAN2(UHAT(2,ISAT),UHAT(1,ISAT))+THETG                     PRED 350
      OBSC(1)=OBSO(1)-DMOD(OBSC(1),DTWOPI)                                PRED 351
      IF(DABS(OBSC(1)).GT.DPI) OBSC(1)=OBSC(1)-DSIGN(DTWOPI,OBSC(1))      PRED 352
      RESID1=OBSC(1)                                                      PRED 353
      OBSC(2)=OBSO(2)-DATAN(RXYZ(3,ISAT)/XY)                              PRED 354
      IF(DABS(OBSC(2)).GT.DPI) OBSC(2)=OBSC(2)-DSIGN(DTWOPI,OBSC(2))      PRED 355
      RESID2=OBSC(2)                                                      PRED 356
      IF (.NOT.TRKSW) GO TO 425                                           PRED 357
      URHOO(1)=DCOS(OBSO(1)-THETG)*DCOS(OBSO(2))                          PRED 358
      URHOO(2)=DSIN(OBSO(1)-THETG)*DCOS(OBSO(2))                          PRED 359
      URHOO(3)=DSIN(OBSO(2))                                              PRED 360
      XYZDOT(1)=XEFIX(XYZI(4,ISAT),XYZI(5,ISAT))+THDT2S*XYZ(2,ISAT)       PRED 361
      XYZDOT(2)=YEFIX(XYZI(4,ISAT),XYZI(5,ISAT))-THDT2S*XYZ(1,ISAT)       PRED 362
      XYZDOT(3)=XYZI(6,ISAT)                                              PRED 363
      DOTPS=DOTPRD(UHAT(1,ISAT),XYZ(1,ISAT))/DOTPRD(URHOO,XYZ(1,ISAT))    PRED 364
      DO 405 I=1,3                                                        PRED 365
      DR(I)=R(ISAT)*DOTPS*URHOO(I)-RXYZ(I,ISAT)                           PRED 366
      I2=MOD(I,3)+1                                                       PRED 367
      I3=MOD(I2,3)+1                                                      PRED 368
      H(I)=XYZ(I2,ISAT)*XYZDOT(I3)-XYZ(I3,ISAT)*XYZDOT(I2)                PRED 369
  405 HI(I)=XYZI(I2,ISAT)*XYZI(I3+3,ISAT)-XYZI(I3,ISAT)*XYZI(I2+3,ISAT)   PRED 370
      T=DOTPRD(DR,XYZDOT)                                                 PRED 371
      W=DOTPRD(DR,H)                                                      PRED 372
      SINU=-(XYZI(2,ISAT)*HI(2)+XYZI(1,ISAT)*HI(1))*HI(3)+XYZI(3,ISAT)*   PRED 373
     .    (HI(1)**2+HI(2)**2)                                            PRED 374
      COSU=(-XYZI(1,ISAT)*HI(2)+XYZI(2,ISAT)*HI(1))*DSQRT(HI(1)**2+       PRED 375
     .    HI(2)**2+HI(3)**2)                                             PRED 376
      U=DARCTN(SINU,COSU)                                                 PRED 377
  425 IF (NOEST) GO TO 2010                                               PRED 378
C CALCULATE PARTIALS FOR ESTIMATION                                       PRED 379
      RSOXY=RXYZ(3,ISAT)/(RSO(ISAT)*XY)                                   PRED 380
      PMSTA1(1)=RXYZ(2,ISAT)/XYSQ(ISAT)                                   PRED 381
      PMSTA1(2)=-RXYZ(1,ISAT)/XYSQ(ISAT)                                  PRED 382
      PMSTA1(3)=0.D0                                                      PRED 383
      PMSTA2(1)=RXYZ(1,ISAT)*RSOXY                                        PRED 384
      PMSTA2(2)=RXYZ(2,ISAT)*RSOXY                                        PRED 385
      PMSTA2(3)=-XY/RSO(ISAT)                                             PRED 386
      GO TO 1930                                                          PRED 387
C RANGE                                                                   PRED 388
  450 RESID1=OBSO(1)-R(ISAT)                                              PRED 389
      IF(NOEST) GO TO 2010                                                PRED 390
      DO 455 I=1,3                                                        PRED 391
```

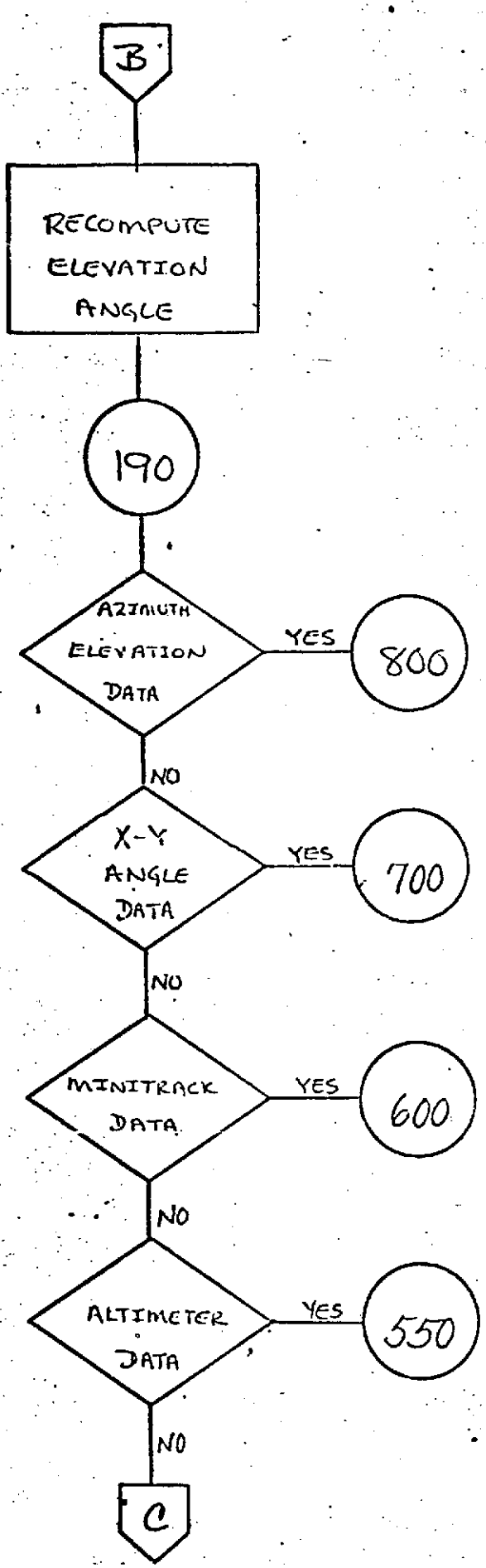
```
  455 PMSTA1(I)=-UHAT(I,ISAT)                                          PRED 392
      GO TO 1900                                                       PRED 393
C RANGE RATE                                                           PRED 394
  500 XYZDOT(1)=XEFIX(XYZI(4,ISAT),XYZI(5,ISAT))+THDT2S*XYZ(2,ISAT)    PRED 395
      XYZDOT(2)=YEFIX(XYZI(4,ISAT),XYZI(5,ISAT))-THDT2S*XYZ(1,ISAT)    PRED 396
      XYZDOT(3)=XYZI(6,ISAT)                                           PRED 397
      OBSC(1)=DOTPRD(XYZDOT,UHAT(1,ISAT))                              PRED 398
      RESID1=OBSO(1)-OBSC(1)                                           PRED 399
      IF(NOEST) GO TO 2010                                             PRED 400
      NP=6                                                             PRED 401
      DO 505 I=1,3                                                     PRED 402
      PMSTA1(I)=-(XYZDOT(I)-OBSC(1)*UHAT(I,ISAT))/R(ISAT)              PRED 403
  505 PMSTA3(I,1)=-UHAT(I,ISAT)                                        PRED 404
      IF(.NOT.TWOSTA) GO TO 1900                                       PRED 405
      RESID1=OBSC(1)*SRFNDX/C                                          PRED 406
      DO 520 I=1,3                                                     PRED 407
      PMSTA1(I)=PMSTA1(I)*SRFNDX/C                                     PRED 408
  520 PMSTA3(I,1)=PMSTA3(I,1)*SRFNDX/C                                 PRED 409
      GO TO 1900                                                       PRED 410
  550 CONTINUE                                                         PRED 411
C ALTIMETER MEASUREMENTS                                               PRED 412
      NP=6                                                             PRED 413
      ELEV(ISAT)=0.5D0*DPI                                             PRED 414
      C3=(2.0D0*C2-4.0D0*C1*UHAT(3,ISAT)**2)*UHAT(3,ISAT)              PRED 415
      RR=1.0D0/R(ISAT)                                                 PRED 416
      OBSC(1)=R(ISAT)-AE-(C1*UHAT(3,ISAT)**2-C2)*UHAT(3,ISAT)**2       PRED 417
      RESID1=OBSO(1)-OBSC(1)                                           PRED 418
      DO 552 I=1,3                                                     PRED 419
  552 G(I)=-UHAT(3,ISAT)*UHAT(I,ISAT)*RR                               PRED 420
      G(3)=G(3)+RR                                                     PRED 421
C ALTIMETER PARTIALS                                                   PRED 422
      DO 555 J=1,3                                                     PRED 423
  555 PMSTA1(J)=-(UHAT(J,ISAT)+C3*G(J))                                PRED 424
      XYZDOT(1)=XEFIX(XYZI(4,ISAT),XYZI(5,ISAT))+THDT2S*XYZ(2,ISAT)    PRED 425
      XYZDOT(2)=YEFIX(XYZI(4,ISAT),XYZI(5,ISAT))-THDT2S*XYZ(1,ISAT)    PRED 426
      XYZDOT(3)=XYZI(6,ISAT)                                           PRED 427
C ALTIMETER RATE                                                       PRED 428
      OBSC(2)=-DOTPRD(PMSTA1,XYZDOT)                                   PRED 429
      RESID2=OBSO(2)-OBSC(2)                                           PRED 430
      IF (NOEST) GO TO 2010                                            PRED 431
      C4=-RR*RR                                                        PRED 432
      DO 558 J=1,3                                                     PRED 433
  558 G2(J,J)=(1.0D0-3.0D0*UHAT(J,ISAT)**2)*UHAT(3,ISAT)*C4            PRED 434
      G2(3,3)=G2(3,3)+2.0D0*UHAT(3,ISAT)*C4                            PRED 435
      G2(1,2)=-3.0D0*UHAT(1,ISAT)*UHAT(2,ISAT)*UHAT(3,ISAT)*C4         PRED 436
      G2(2,1)=G2(1,2)                                                  PRED 437
      DO 6667 I=1,2                                                    PRED 438
      G2(I,3)=(1.0D0-3.0D0*UHAT(3,ISAT)**2)*UHAT(I,ISAT)*C4            PRED 439
 6667 G2(3,I)=G2(I,3)                                                  PRED 440
      DO 553 I=1,3                                                     PRED 441
      DO 554 J=1,3                                                     PRED 442
  554 R2(I,J)=-UHAT(I,ISAT)*UHAT(J,ISAT)*RR                            PRED 443
  553 R2(I,I)=R2(I,I)+RR                                               PRED 444
      C5=-12.0D0*C1*UHAT(3,ISAT)**2+2.0D0*C2                           PRED 445
      DO 556 I=1,3                                                     PRED 446
      PMSTA3(I,1)=0.0D0                                                PRED 447
```

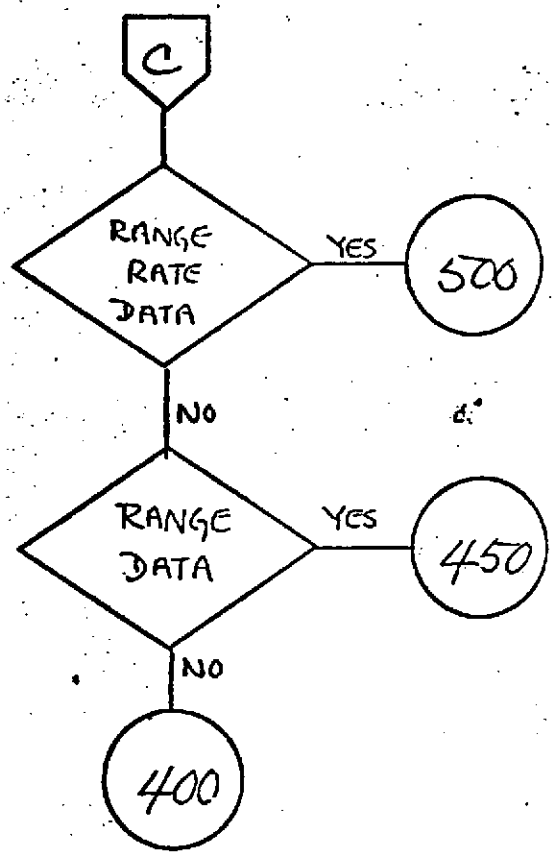
```
      DO 556 J=1,3                                                    PRED 448
C SECOND PARTIALS OF ALTIMETER WRT X,Y,Z                              PRED 449
  556 PARH2(I,J)=R2(I,J)+C3*G2(I,J)+G (I)*C5*G (J)                    PRED 450
      DO 557 I=1,3                                                    PRED 451
C PARTIAL HDOT,WRT X,Y,Z,XDOT,YDOT,ZDOT                               PRED 452
      PMSTA2(I)=0.D0                                                  PRED 453
      PMSTA3(I,2)=PMSTA1(I)                                           PRED 454
      DO 557 J=1,3                                                    PRED 455
  557 PMSTA2(I)=PMSTA2(I)-XYZDOT(J)*PARH2(I,J)                        PRED 456
      GO TO 1900                                                      PRED 457
C DIRECTION COSINES                                                   PRED 458
  600 RESID1=OBSO(1)-RENV(1,ISAT)                                     PRED 459
      RESID2=OBSO(2)-RENV(2,ISAT)                                     PRED 460
      IF(NOEST) GO TO 2010                                            PRED 461
      DO 605 I=1,3                                                    PRED 462
      PMSTA1(I)=-(EHAT(I,ISTA)-RENV(1,ISAT)*UHAT(I,ISAT))/R(ISAT)     PRED 463
  605 PMSTA2(I)=-(NHAT(I,ISTA)-RENV(2,ISAT)*UHAT(I,ISAT))/R(ISAT)     PRED 464
      GO TO 1900                                                      PRED 465
C X-Y ANGLES                                                          PRED 466
  700 EZSQ=1.D0-RENV(2,ISAT)**2                                       PRED 467
      EZ=DSQRT(EZSQ)                                                  PRED 468
      RESID1=OBSO(1)-DATAN(RENV(1,ISAT)/RENV(3,ISAT))                 PRED 469
      RESID2=OBSO(2)-DATAN(RENV(2,ISAT)/EZ)                           PRED 470
      IF(NOEST) GO TO 2010                                            PRED 471
      REZ=R(ISAT)*EZ                                                  PRED 472
      REZSQ=R(ISAT)*EZSQ                                              PRED 473
      DO 705 I=1,3                                                    PRED 474
      PMSTA1(I)=(ZHAT(I,ISTA)*RENV(1,ISAT)-EHAT(I,ISTA)*RENV(3,ISAT))/ PRED 475
     *   REZSQ                                                        PRED 476
  705 PMSTA2(I)=-(NHAT(I,ISTA)-RENV(2,ISAT)*UHAT(I,ISAT))/REZ         PRED 477
      GO TO 1900                                                      PRED 478
C AZIMUTH & ELEVATION ANGLES                                          PRED 479
  800 RESID1=OBSO(1)-DARCTN(RENV(1,ISAT),RENV(2,ISAT))                PRED 480
      IF(DABS(RESID1).GT.DPI) RESID1=RESID1-DSIGN(DTWOPI,RESID1)      PRED 481
      RESID2=OBSO(2)-ELEV(ISAT)                                       PRED 482
      IF(NOEST) GO TO 2010                                            PRED 483
      REN=R(ISAT)*EN                                                  PRED 484
      RENSQ=R(ISAT)*ENSQ                                              PRED 485
      DO 805 I=1,3                                                    PRED 486
      PMSTA1(I)=-(RENV(2,ISAT)*EHAT(I,ISTA)-RENV(1,ISAT)*NHAT(I,ISTA))/ PRED 487
     *   RENSQ                                                        PRED 488
  805 PMSTA2(I)=-(ZHAT(I,ISTA)-RENV(3,ISAT)*UHAT(I,ISAT))/REN         PRED 489
C CONVERT EARTH FIXED PARTIALS TO INERTIAL                            PRED 490
 1900 CALL CLEAR(PMPXO,NPARM,2*NMEAS)                                 PRED 491
      IF(MTYPE.GT.14.AND.MTYPE.LT.27) GO TO 1940                      PRED 492
      DO 1925 K=1,NMEAS                                               PRED 493
      PM(1,K)=-XINERT(PMSTA(1,K),PMSTA(2,K))                          PRED 494
      PM(2,K)=-YINERT(PMSTA(1,K),PMSTA(2,K))                          PRED 495
      PM(3,K)=-PMSTA(3,K)                                             PRED 496
      PMPXO(NPARM-5,K)=PMSTA(1,K)                                     PRED 497
      PMPXO(NPARM-4,K)=PMSTA(2,K)                                     PRED 498
      PMPXO(NPARM-3,K)=PMSTA(3,K)                                     PRED 499
      PMPXO(NPARM-7,K)=1.D0                                           PRED 500
      IF(NP.LT.4) GO TO 1925                                          PRED 501
      PM(4,K)=-XINERT(PMSTA3(1,K),PMSTA3(2,K))                        PRED 502
      PM(5,K)=-YINERT(PMSTA3(1,K),PMSTA3(2,K))                        PRED 503
```

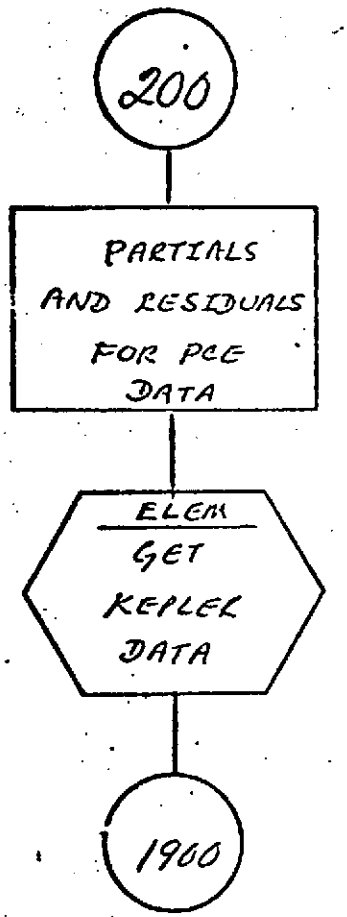
```
      PM(6,K)=-PMSTA3(3,K)                                                  PRED 504
      IF(MTYPE.NE.3.AND.MTYPE.NE.4.AND.MTYPE.NE.28) GO TO 1925              PRED 505
      IF(MTYPE.EQ.4.AND.K.EQ.1) GO TO 1925                                  PRED 506
      PM(1,K)=PM(1,K)-THDT2S*PM(5,K)                                        PRED 507
      PM(2,K)=PM(2,K)+THDT2S*PM(4,K)                                        PRED 508
 1925 CONTINUE                                                              PRED 509
      GO TO 1940                                                            PRED 510
 1930 INCR=ISAT2-ISAT1                                                      PRED 511
      PMPX0(NPARM-5,1)=PMSTA1(1)                                            PRED 512
      PMPX0(NPARM-4,1)=PMSTA1(2)                                            PRED 513
      PMPX0(NPARM-3,1)=PMSTA1(3)                                            PRED 514
      PMPX0(NPARM-7,1)=1.0D0                                                PRED 515
C CHAIN INSTANTANEOUS PARTIALS BACK TO EPOCH                                PRED 516
 1940 ISATNO=ISAT1                                                          PRED 517
 1945 L1=(ISATNO-1)*6                                                       PRED 518
      DO 1950 I=1,6                                                         PRED 519
      L1=L1+1                                                               PRED 520
      DO 1950 K=1,NMEAS                                                     PRED 521
      K1=K+K2                                                               PRED 522
      DO 1950 J=1,NP                                                        PRED 523
 1950 PMPX0(L1,K1)=PMPX0(L1,K1)+PM(J,K1)*PXPX0(J,I,ISATNO)                  PRED 524
      I1=6                                                                  PRED 525
      J1=NSAT*6                                                             PRED 526
      DO 1970 I=1,3                                                         PRED 527
      L1=ADDR(ISATNO,I)                                                     PRED 528
      IF(L1.LE.0) GO TO 1970                                                PRED 529
      L1=J1+L1                                                              PRED 530
      I1=I1+1                                                               PRED 531
      DO 1960 K=1,NMEAS                                                     PRED 532
      K1=K+K2                                                               PRED 533
      DO 1960 J=1,NP                                                        PRED 534
 1960 PMPX0(L1,K1)=PMPX0(L1,K1)+PM(J,K1)*PXPX0(J,I1,ISATNO)                 PRED 535
 1970 CONTINUE                                                              PRED 536
      I1=I1+1                                                               PRED 537
      I2=NEQNS(ISATNO)-1                                                    PRED 538
      IF(I1.GT.I2) GO TO 2000                                               PRED 539
      L1=MAX0(ADDR(1,1),ADDR(2,1),ADDR(1,2),ADDR(2,2),ADDR(1,3),           PRED 540
     *   ADDR(2,3))+J1                                                      PRED 541
      DO 1980 I=I1,I2                                                       PRED 542
      L1=L1+1                                                               PRED 543
      DO 1980 K=1,NMEAS                                                     PRED 544
      K1=K+K2                                                               PRED 545
      DO 1980 J=1,NP                                                        PRED 546
 1980 PMPX0(L1,K1)=PMPX0(L1,K1)+PM(J,K1)*PXPX0(J,I,ISATNO)                  PRED 547
 2000 K2=K2+2                                                               PRED 548
      ISATNO=ISATNO+INCR                                                    PRED 549
      IF(ISATNO.EQ.ISAT2) GO TO 1945                                        PRED 550
 2010 ELEV(ISAT)=ELEV(ISAT)/DRAD                                            PRED 551
C TEST FOR GROUND TRACK WANTED                                              PRED 552
      IF(.NOT.SUBSAT) RETURN                                                PRED 553
C EAST LONGITUDE OF SATELLITE IN DEGREES                                    PRED 554
 2020 DO 2050 J=1,NSAT                                                      PRED 555
      SATLON(J)=(DATAN2(XYZI(2,J),XYZI(1,J))-THETS)/DRAD                    PRED 556
      SATLON(J)=DMOD(SATLON(J)+7.202,3.602)                                 PRED 557
C GEODETIC LONGITUDE OF SATELLITE IN DEGREES                                PRED 558
      XYSQ(J)=XYZI(1,J)**2+XYZI(2,J)**2                                     PRED 559
```

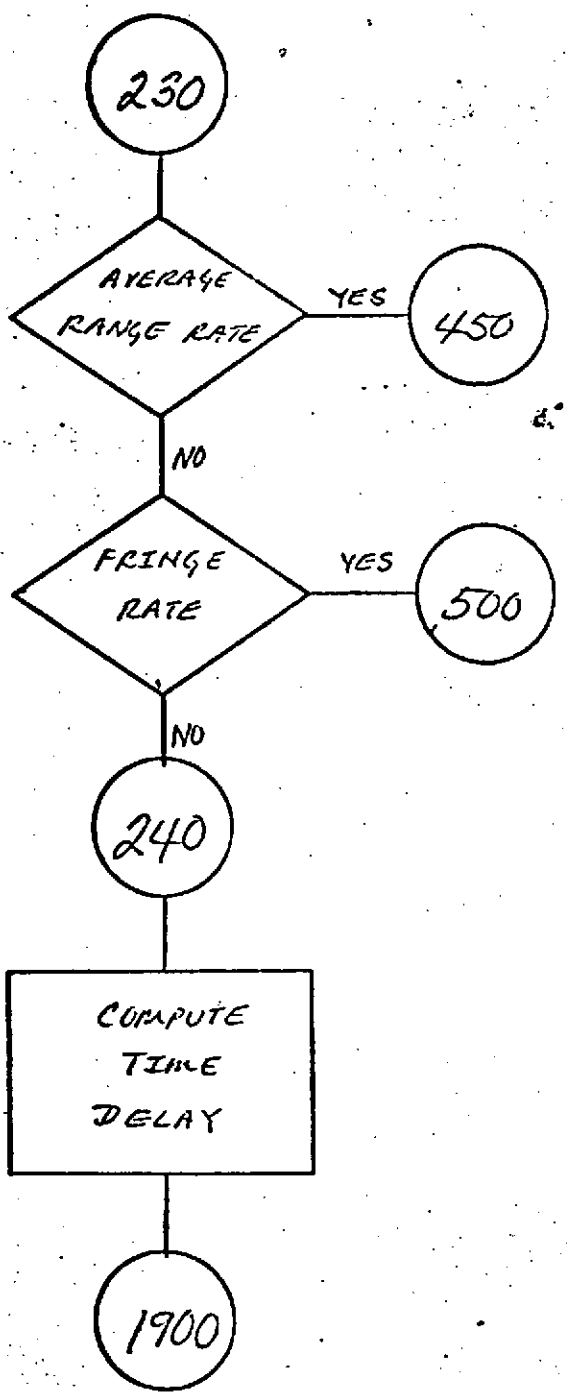
```
      SATLAT(J)=XYZI(3,J)/(DSQRT(XYSQ(J))*FLAT21)                    PRED 560
      SATLAT(J)=DATAN(SATLAT(J))/DRAD                                PRED 561
C SATELLITE HEIGHT IN METERS                                         PRED 562
      RSAT=DSQRT(XYSQ(J)+XYZI(3,J)**2)                               PRED 563
      SPSISQ=(XYZI(3,J)/RSAT)**2                                     PRED 564
 2050 SATH(J)=(RSAT-AE)-(FSQ32*SPSISQ**2-FFSQ32*SPSISQ)              PRED 565
      RETURN                                                         PRED 566
 3000 FORMAT(' $**********$ ACCEPTED TRANSIT TIME ERROR AFTER SIX ',  PRED 567
     .   'ITERATIONS =',E12.5,' SECONDS. GREATER THAN',E12.5,        PRED 568
     .   ' DAYS $**********$')                                       PRED 569
      END                                                            PRED 570
```

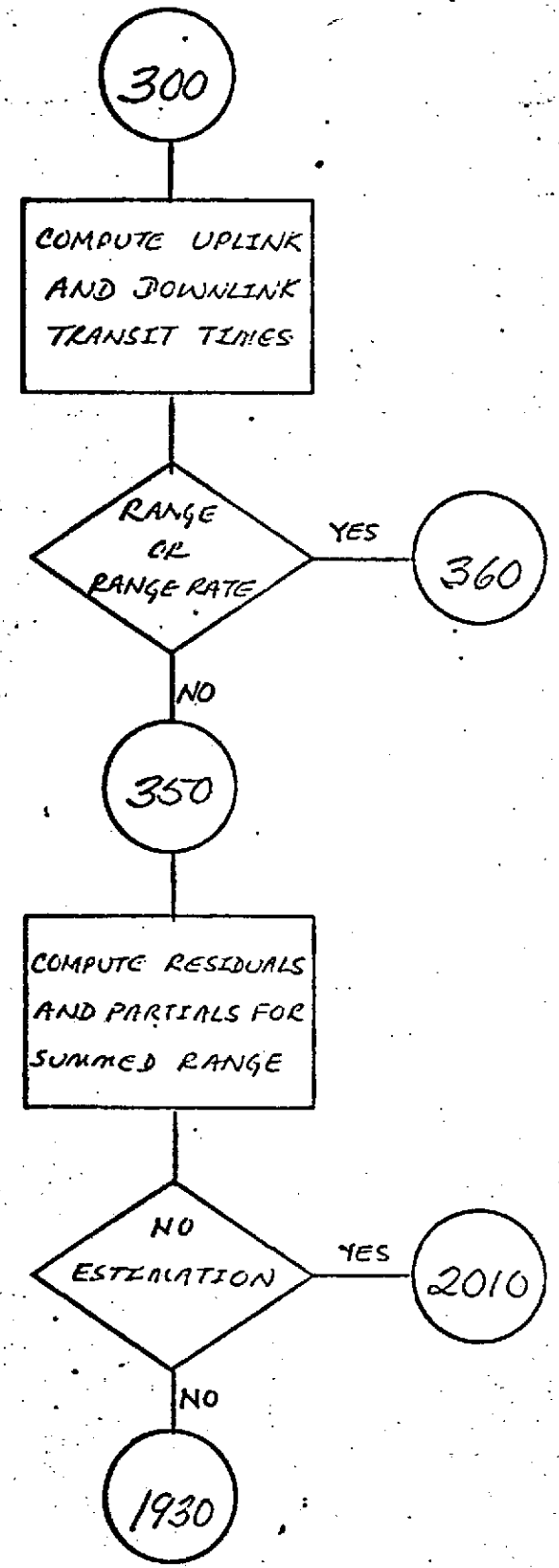

ENTRY PREDCI
INITIALIZATION
RETURN
ENTRY PREDCT
INITIALIZE
SATELLITE TO SATELLITE TRACKING
YES
300
NO
VLBI OR AVERAGE RANGE RATE DATA
YES
50
NO
.ORBIT
OBTAIN SATELLITE ORBIT
GRHRAN
OBTAIN RIGHT ASCENSION OF GREENWICH AND STATION TO SATELLITE VECTOR
A
GROUNDTRACK ONLY
YES
200
NO
50
COMPUTE ELEVATION ANGLE
VLBI OR AVERAGE RANGE RATE DATA
YES
230
NO
PREPROCESS DATA IF REQUESTED
TRANSIT TIME CORRECTION MADE
NO
190
YES
B

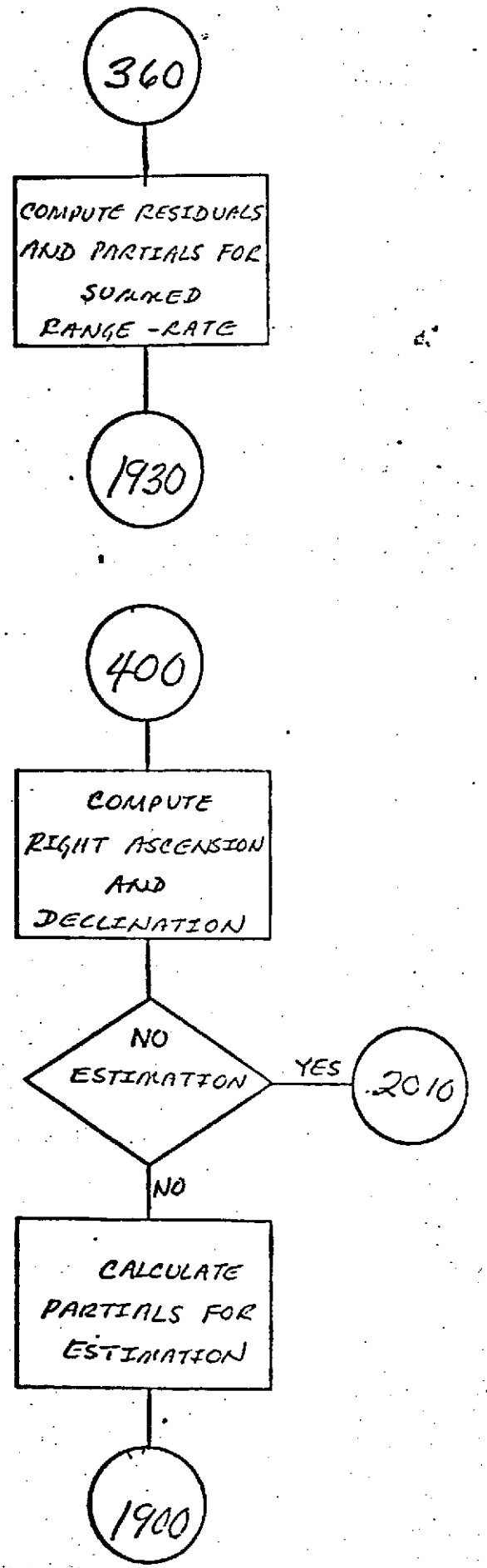

B
RECOMPUTE ELEVATION ANGLE
190
AZIMUTH ELEVATION DATA
YES
800
NO
X-Y ANGLE DATA
YES
700
NO
MINITRACK DATA
YES
600
NO
ALTIMETER DATA
YES
550
NO
C

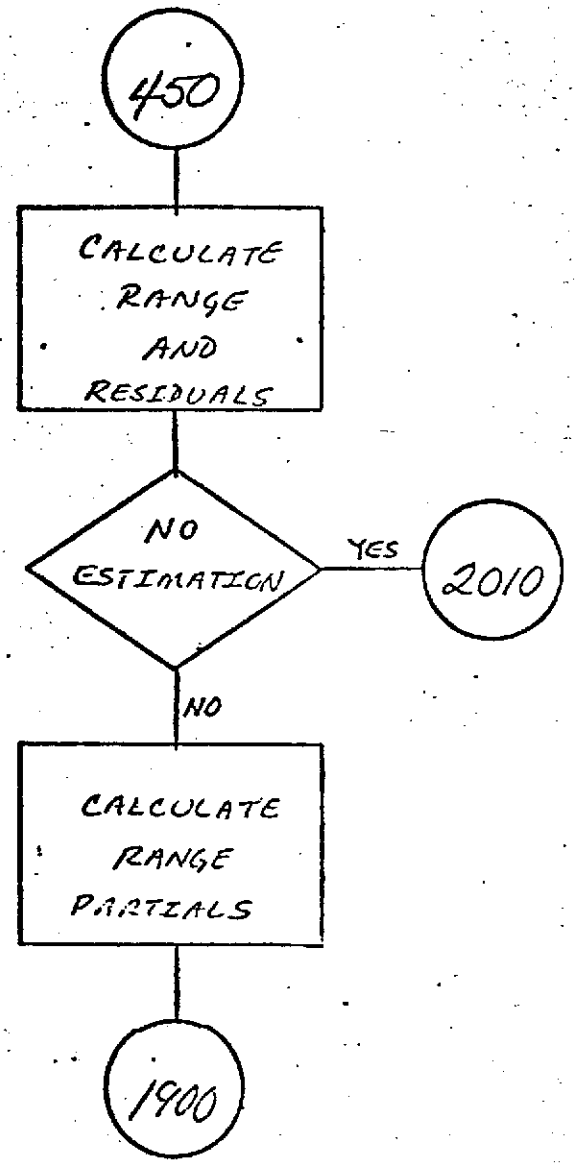

C
RANGE RATE DATA
YES
500
NO
RANGE DATA
YES
450
NO
400

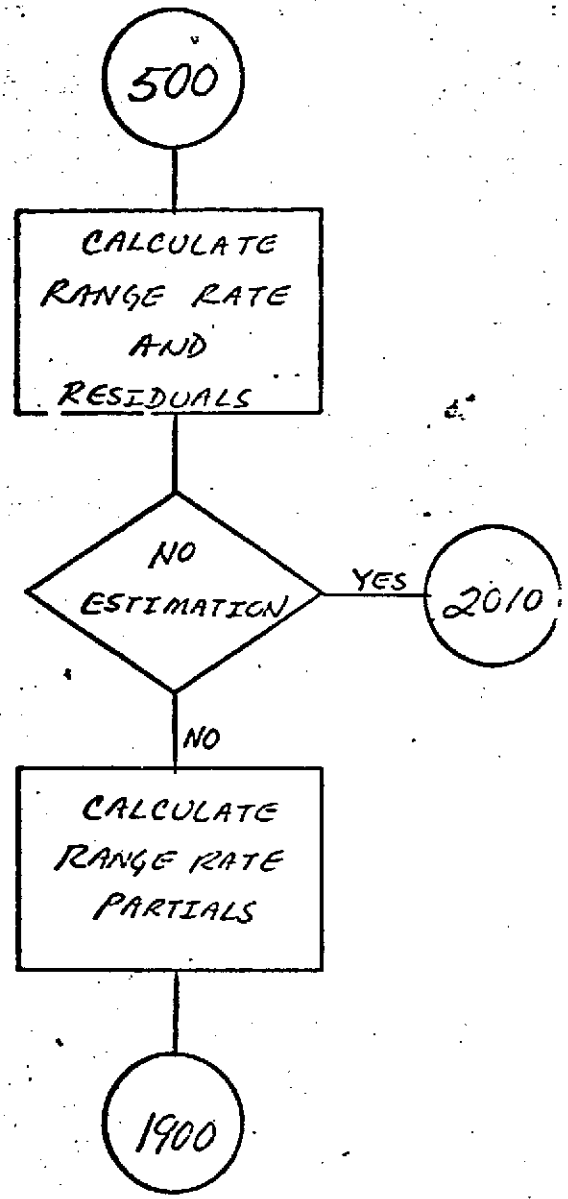

200
PARTIALS AND RESIDUALS FOR PCE DATA
ELEM
GET KEPLER DATA
1900

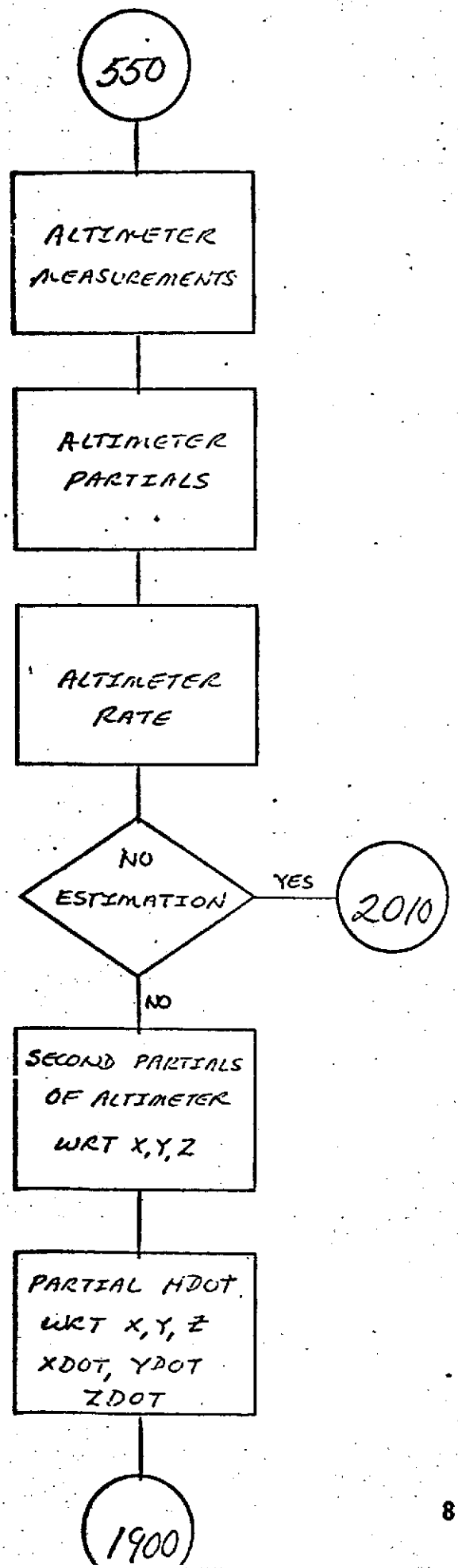

230
AVERAGE RANGE RATE
YES
450
NO
FRINGE RATE
YES
500
NO
240
COMPUTE TIME DELAY
1900

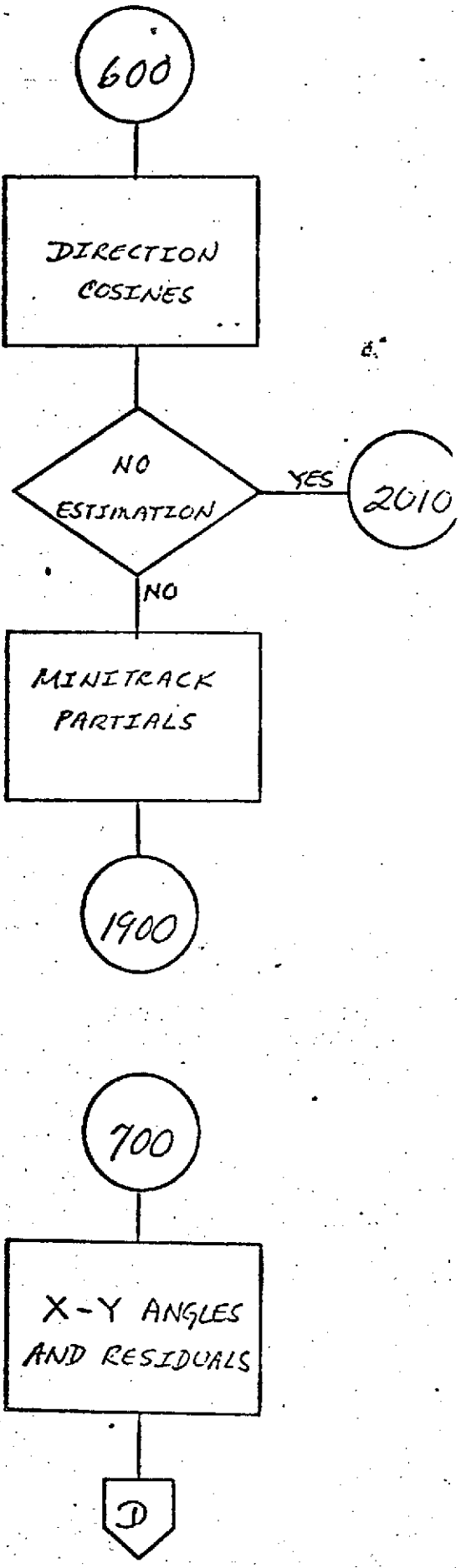

300
COMPUTE UPLINK AND DOWNLINK TRANSIT TIMES
RANGE OR RANGE RATE
YES
360
NO
350
COMPUTE RESIDUALS AND PARTIALS FOR SUMMED RANGE
NO ESTIMATION
YES
2010
NO
1930

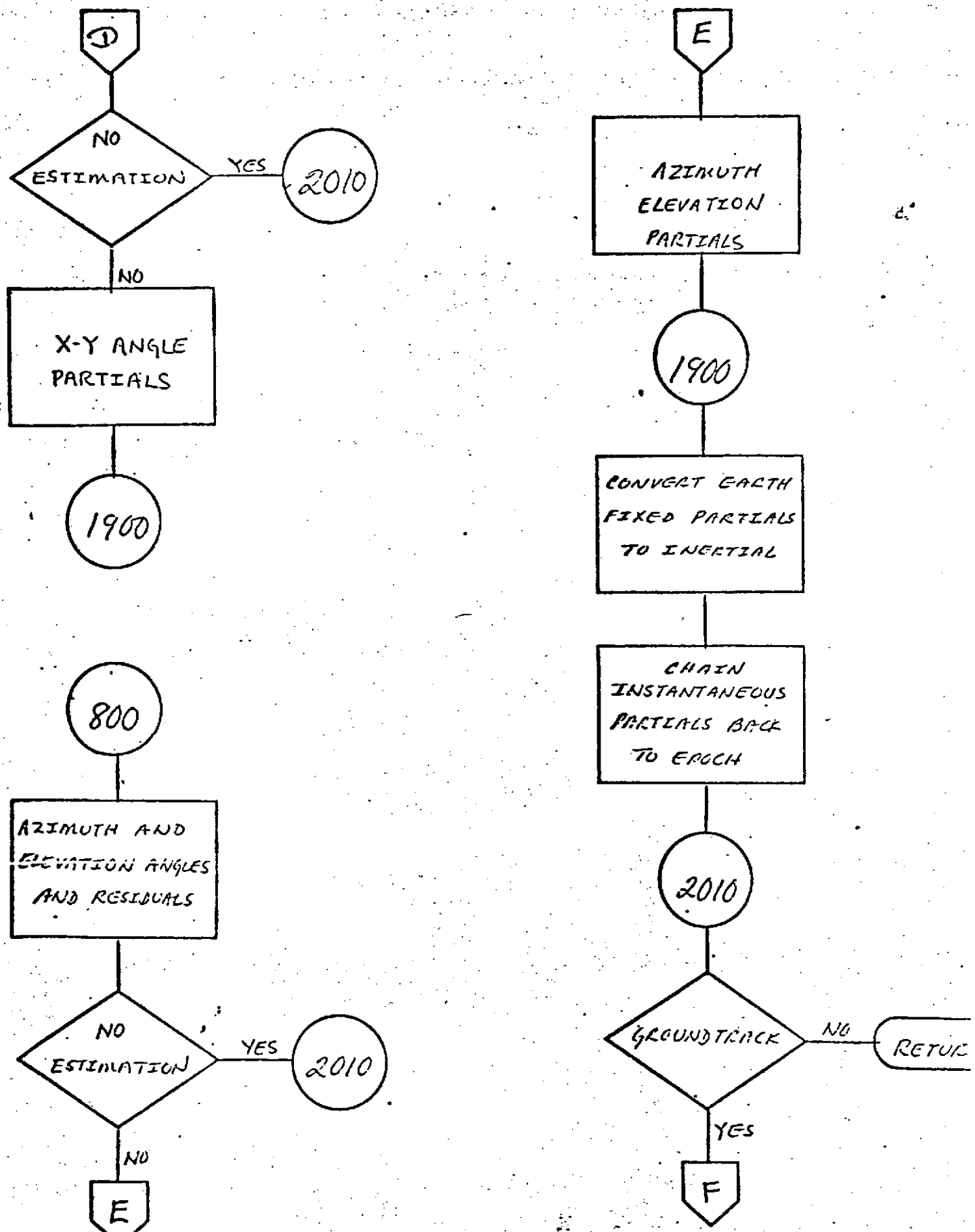

360
COMPUTE RESIDUALS AND PARTIALS FOR SUMMED RANGE-RATE
1930
400
COMPUTE RIGHT ASCENSION AND DECLINATION
NO ESTIMATION
YES
2010
NO
CALCULATE PARTIALS FOR ESTIMATION
1900

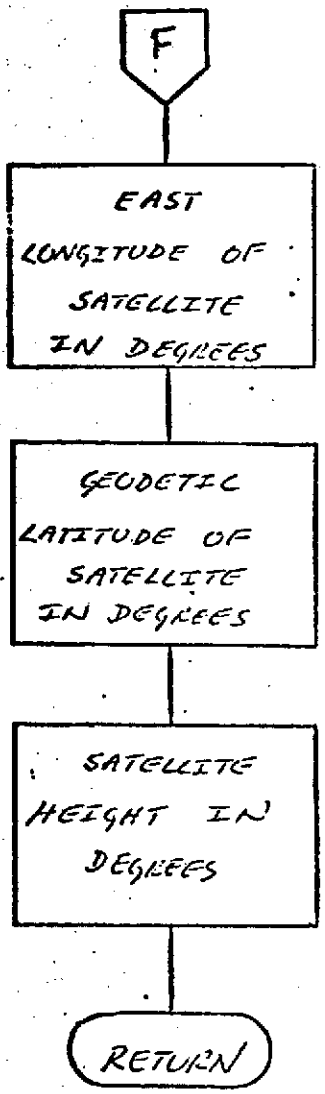

450
CALCULATE RANGE AND RESIDUALS
NO ESTIMATION
YES
2010
NO
CALCULATE RANGE PARTIALS
1900


500
CALCULATE RANGE RATE AND RESIDUALS
NO ESTIMATION
YES
2010
NO
CALCULATE RANGE RATE PARTIALS
1900


550
ALTIMETER MEASUREMENTS
ALTIMETER PARTIALS
ALTIMETER RATE
NO ESTIMATION
YES
2010
NO
SECOND PARTIALS OF ALTIMETER WRT X, Y, Z
PARTIAL HDOT WRT X, Y, Z XDOT, YDOT ZDOT
1900


600
DIRECTION COSINES
NO ESTIMATION
YES
2010
NO
MINITRACK PARTIALS
1900
700
X-Y ANGLES AND RESIDUALS
D


D
NO ESTIMATION
YES
2010
NO
X-Y ANGLE PARTIALS
1900
800
AZIMUTH AND ELEVATION ANGLES AND RESIDUALS
NO ESTIMATION
YES
2010
NO
E
E
AZIMUTH ELEVATION PARTIALS
1900
CONVERT EARTH FIXED PARTIALS TO INERTIAL
CHAIN INSTANTANEOUS PARTIALS BACK TO EPOCH
2010
GROUNDTRACK
NO
RETUR
YES
F


F
EAST LONGITUDE OF SATELLITE IN DEGREES
GEODETIC LATITUDE OF SATELLITE IN DEGREES
SATELLITE HEIGHT IN DEGREES
RETURN

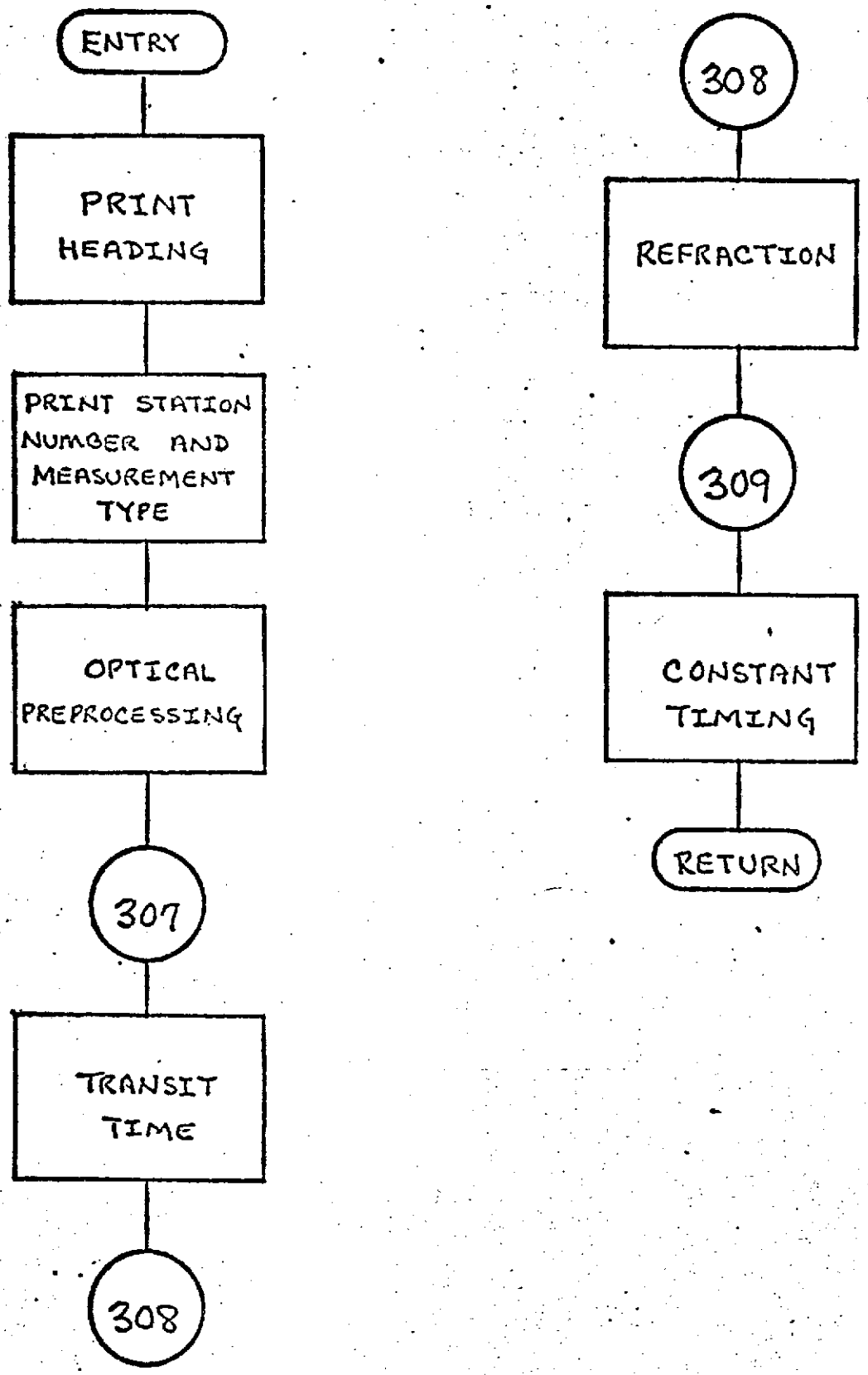
NAME	PRNTPR
PURPOSE	TO PRINT THE OBSERVATION PREPROCESSING REQUESTED
CALLING SEQUENCE	CALL PRNTPR(OUTP,ATYPE)

SYMBOL	TYPE	DESCRIPTION
OUTP	I	OUTPUT - PRINTER
ATYPE (31)	DP	OUTPUT - MEASUREMENT TYPE NAMES

SUBROUTINES USED	NONE
COMMON BLOCKS	CGEOS
INPUT FILES	NONE
OUTPUT FILES	OUTP - PRINTER

```
      SUBROUTINE PRNTPR(OUTP,ATYPE)                                          PRNT  25
      REAL*8 ATYPE(31),NAME1,NAME2,ALL,TYPES,BLANK                           PRNT  26
      INTEGER OUTP                                                           PRNT  27
      DIMENSION DUM(2)                                                       PRNT  28
      COMMON/CGEOS/DUM,IPREPR(4,50),RFINDX(2,50),INDPRE(2,50),NOPRPR,        PRNT  29
     *       G1(202)                                                         PRNT  30
      INTEGER*2 IPREPR,INDPRE                                                PRNT  31
      DATA MAXTPE/31/,ALL/5H  ALL/,TYPES/5HTYPES/,BLANK/1H /                 PRNT  32
C PRINT HEADING                                                              PRNT  33
      WRITE(OUTP,10700)                                                      PRNT  34
      DO 320 I=1,NOPRPR                                                      PRNT  35
      WRITE(OUTP,10710)                                                      PRNT  36
      L=INDPRE(2,I)                                                          PRNT  37
      IF(L.EQ.0.OR.(L.GT.14.AND.L.LT.27)) L=MAXTPE                           PRNT  38
C PRINT STATION NUMBER & MEASUREMENT TYPE                                    PRNT  39
      NAME1=ALL                                                              PRNT  40
      NAME2=TYPES                                                            PRNT  41
      IF(L.NE.MAXTPE) NAME1=ATYPE(L)                                         PRNT  42
      IF(L.NE.MAXTPE) NAME2=BLANK                                            PRNT  43
      IF(L.LT.8) NAME2=ATYPE(L+7)                                            PRNT  44
      IF(INDPRE(1,I)) 320,250,225                                            PRNT  45
  225 WRITE(OUTP,10701) INDPRE(1,I),NAME1,NAME2                              PRNT  46
      GO TO 275                                                              PRNT  47
  250 WRITE(OUTP,10702) NAME1,NAME2                                          PRNT  48
  275 CONTINUE                                                               PRNT  49
      DO 305 J=1,4                                                           PRNT  50
      IF(IPREPR(J,I).GT.0) GO TO (307,308,309,310),J                         PRNT  51
  305 CONTINUE                                                               PRNT  52
      GO TO 320                                                              PRNT  53
C TRANSIT TIME                                                               PRNT  54
  307 WRITE(OUTP,10703)                                                      PRNT  55
```

```
      GO TO 305                                                          PRNT  56
  308 IF(IPREPR(1,I).NE.0) WRITE(OUTP,10710)                             PRNT  57
C REFRACTION                                                             PRNT  58
      A=RFINDX(1,I)                                                      PRNT  59
      IF(A.LE.0.) A=328.6                                                PRNT  60
      WRITE(OUTP,10704) A                                                PRNT  61
      GO TO 305                                                          PRNT  62
C CONSTANT TIMING                                                        PRNT  63
  309 IF(IPREPR(1,I)+IPREPR(2,I).NE.0) WRITE(OUTP,10710)                 PRNT  64
      WRITE(OUTP,10705) RFINDX(2,I)                                      PRNT  65
      GO TO 305                                                          PRNT  66
C OPTICAL                                                                PRNT  67
  310 WRITE(OUTP,10708)                                                  PRNT  68
  320 CONTINUE                                                           PRNT  69
      RETURN                                                             PRNT  70
10700 FORMAT(1H1,50X,25HPREPROCESSING CORRECTIONS/1H0,44X,7HSTATION,     PRNT  71
     1      21X,10HCORRECTION/1H ,36X,6HNUMBER,5X,7HTYPE(S),12X,        PRNT  72
     2      7HTYPE(S),11X,5HVALUE)                                      PRNT  73
10701 FORMAT(37X,I5,3X,A6,2X,A6)                                         PRNT  74
10702 FORMAT(38X,3HALL,4X,A6,2X,A6)                                      PRNT  75
10703 FORMAT(1H+,63X,12HTRANSIT TIME)                                    PRNT  76
10704 FORMAT(1H+,63X,10HREFRACTION,F12.1,8H N UNITS)                     PRNT  77
10705 FORMAT(1H+,63X,11HTIMING BIAS,3PF11.1,13H MILLISECONDS)            PRNT  78
10708 FORMAT(1H ,63X,7HOPTICAL)                                          PRNT  79
10710 FORMAT(1H )                                                        PRNT  80
      END                                                                PRNT  81
```


ENTRY
PRINT HEADING
PRINT STATION NUMBER AND MEASUREMENT TYPE
OPTICAL PREPROCESSING
307
TRANSIT TIME
308
308
REFRACTION
309
CONSTANT TIMING
RETURN

PROCES

DESCRIPTION

PROCES is a subroutine designed specifically to complete the GEODYN preprocessing.

PROCES makes the following preprocessing corrections:

- Transit Time,
- Annual Aberration,
- Diurnal Aberration,
- Parallactic Refraction,
- Tropospheric Refraction,
- Range Transponder Corrections.

The specific corrections applied by PROCES are dependent upon indicators input to PROCES through COMMON storage.

NAME PROCES

ENTRY POINT PURPOSE

PROCS1 INITIALIZATION

PROCES TO COMPLETE PREPROCESSING OF SATELLITE MEASUREMENTS

CALLING SEQUENCE CALL PROCS1(ISTANO,STAXYZ,RLAT,RLON)

SYMBOL	TYPE	DESCRIPTION
ISTANO (1)	I*2	INPUT - TRACKING STATION NUMBERS
STAXYZ (3,1)	DP	INPUT - TRACKING STATION CARTESIAN COORDINATES
RLAT (1)	DP	INPUT - TRACKING STATION LATITUDES
RLON (1)	DP	INPUT - TRACKING STATION LONGITUDES

CALLING SEQUENCE CALL PROCES(ISTA,DAY,THETG)

SYMBOL	TYPE	DESCRIPTION
ISTA	I	INPUT - TRACKING STATION INDEX
DAY	DP	INPUT & OUTPUT - MEASUREMENT TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
THETG	DP	INPUT & OUTPUT - RIGHT ASCENSION OF GREENWICH

SUBROUTINES USED ORBIT, GRHRAN, REFION, DOTPRD, NUMBR4, OBSDOT, EPHEM, EQN, DJUL

COMMON BLOCKS CGEOS, CUVECT, CEPHEM, GNDTRK, CONSTS, PREBLK

INPUT FILES NONE

OUTPUT FILES NONE

REFERENCES "GEODYN SYSTEMS DESCRIPTION" VOLUME 1 - GEODYN DOCUMENTATION

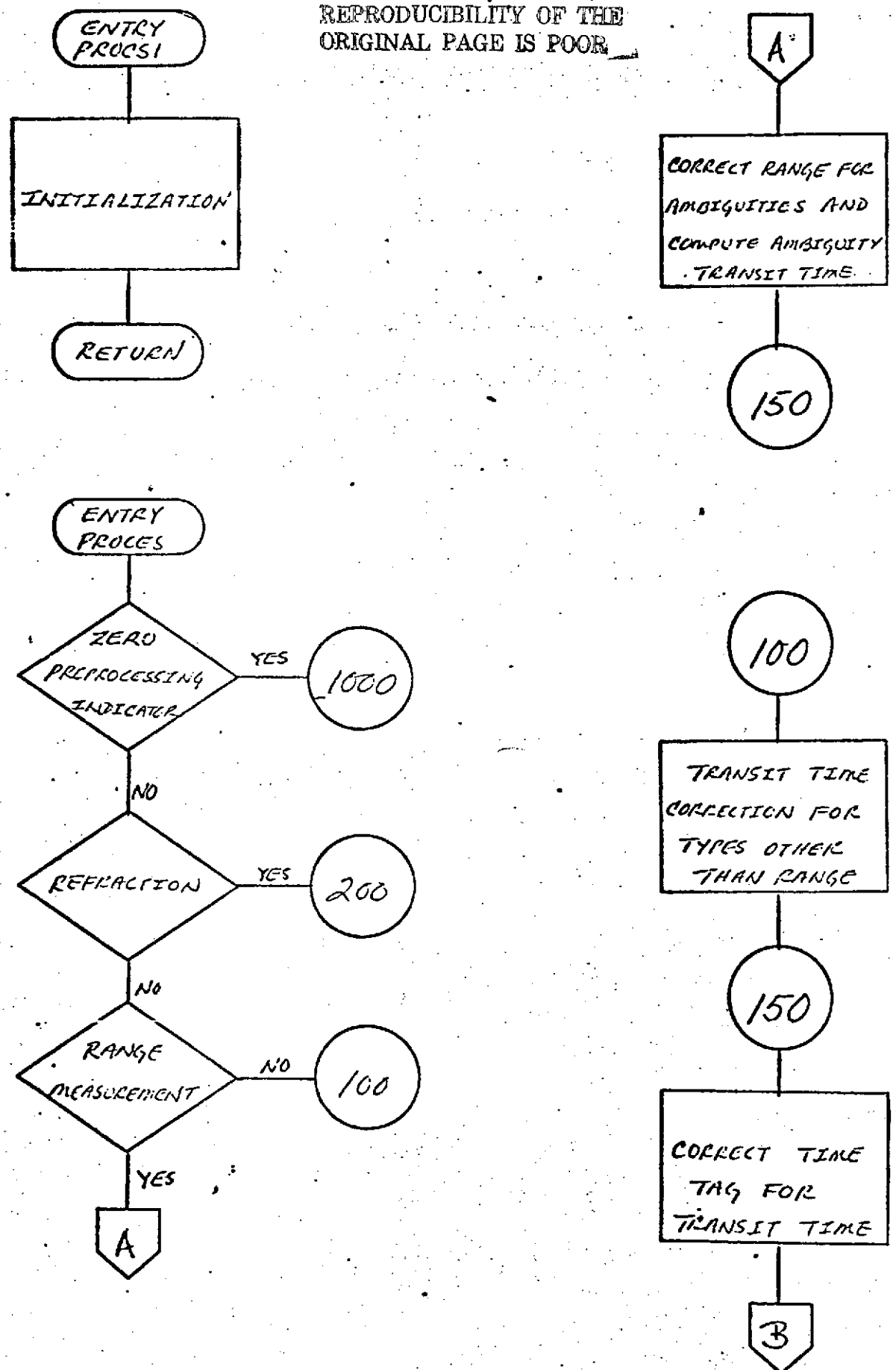
```
      SUBROUTINE PROCS1(ISTANO,STAXYZ,RLAT,RLON)
      IMPLICIT REAL*8 (A-H,O-Z)
      LOGICAL*1 VHFCHN,PREPRO
      LOGICAL SATSW
```

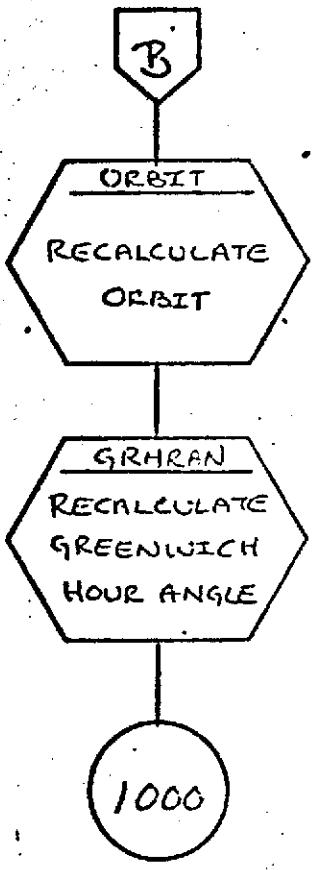
```
      INTEGER RECNO                                                          PROC  56
      INTEGER*2 MTYPE,NMEAS,PRETYP,CHANEL,ISAT1,IPRE,ISTANO                  PROC  57
      DOUBLE PRECISION L                                                     PROC  58
      REAL TRANSP,REFION,EON,OBSDOT                                          PROC  59
      DIMENSION PSAT(3),STASAT(3),ISTANO(1),TRANSP(3,2,9),KSATNO(27),        PROC  60
     . STAXYZ(3,1),RLAT(1),RLON(1),GATE(3),PRETYP(2),STPSAT(3)               PROC  61
      COMMON/CGEOS/ISATID(2),IPREPR(453)                                     PROC  62
      COMMON/CJVECT/UHAT(3,2),XYZ(3,2),RXYZ(3,2),PENV(3,2),R(2),RSQ(2),      PROC  63
     .    XYSQ(2)                                                           PROC  64
      COMMON/CEPHEM/A0(4),SUNX,SUNY,IG2(1324)                                PROC  65
      COMMON/GNDTRK/SAT_LAT(2),SATLON(2),SATH(2),ELEV(2),SATSW               PROC  66
      COMMON/CONSTS/DPI,DTWOPI,D2R,S2R                                       PROC  67
      COMMON/PREBLK/DAYSTA,OBSO1,OBSO2,SIG1,SIG2,SRENDX,ISN,MTYPE,           PROC  68
     .    NMEAS,ISAT1,IPRE,CHANEL,VHFCHN,PREPRO,RECNO                       PROC  69
      DATA NSATNO/27/,GATE/18737031.3D0,4684257.8D0,936851.6D0/,              PROC  70
     .    VLIGHT/2.99792508/,STDNDX/328.5D0/,PCONST/0.84323350-2/            PROC  71
      DATA KSATNO/65891,650891,650C91,                                       PROC  72
     .            68021,680C21,6800201,                                      PROC  73
     .            64541,640541,6405401,                                      PROC  74
     .            65811,650811,6508101,                                      PROC  75
     .            66491,660491,6604901,                                      PROC  76
     .            67731,670731,6707301,                                      PROC  77
     .            68141,680141,6801401,                                      PROC  78
     .            69511,69511,6905101,                                       PROC  79
     .            65911,65911,6905111/                                       PROC  80
      DATA TRANSP/                                                           PROC  81
C  GEOS A                                                                    PROC  82
     .    0.3677E4,2*0.,0.306E4,2*0.,                                       PROC  83
C  GEOS B                                                                    PROC  84
     .    0.3689E4,2*0.,0.3684E4,2*0.,                                      PROC  85
C  OGO 1                                                                     PROC  86
     .    0.3636E4,2*0.,0.3710E4,2*0.,                                      PROC  87
C  OGO 2                                                                     PROC  88
     .    0.352E4,2*0.,0.373E4,2*0.,                                        PROC  89
C  OGO 3                                                                     PROC  90
     .    0.3519E4,0.8508E-2,0.2376E-6,0.3731E4,-.5597E-2,0.3535E-6,        PROC  91
C  OGO 4                                                                     PROC  92
     .    0.3599E4,0.6014E-2,0.4384E-6,0.3740E4,0.9844E-2,0.4587E-6,        PROC  93
C  OGO 5                                                                     PROC  94
     .    0.3797E4,2*0.,0.3880E4,2*0.,                                      PROC  95
C  OGO 6                                                                     PROC  96
     .    0.3720E4,0.6738E-2,0.2410E-6,0.3732E4,-.6299E-2,0.2430E-6,        PROC  97
     .    0.3927E4,-.6929E-2,0.4664E-6,0.7282E4,-.2003E-2,-.4685E-6/        PROC  98
      RETURN                                                                 PROC  99
      ENTRY PROCES(ISTA,DAY,THETG)                                           PROC 100
      PRETYP(1)=IPRE/10                                                      PROC 101
      PRETYP(2)=IPRE-PRETYP(1)*10                                            PROC 102
      PREPRO=.FALSE.                                                         PROC 103
      DO 1000 I=1,2                                                          PROC 104
      IF(PRETYP(I).EQ.0) GO TO 1000                                          PROC 105
      IF(I.EQ.2) GO TO 200                                                   PROC 106
      IF(MTYPE.NE.2) GO TO 100                                               PROC 107
      I1=PRETYP(1)                                                           PROC 108
C TRANSIT TIME                                                               PROC 109
      I2=(R(ISAT1)-OBSO1)/GATE(I1)+0.500                                     PROC 110
      IF(I2.EQ.0) GO TO 1000                                                 PROC 111
```

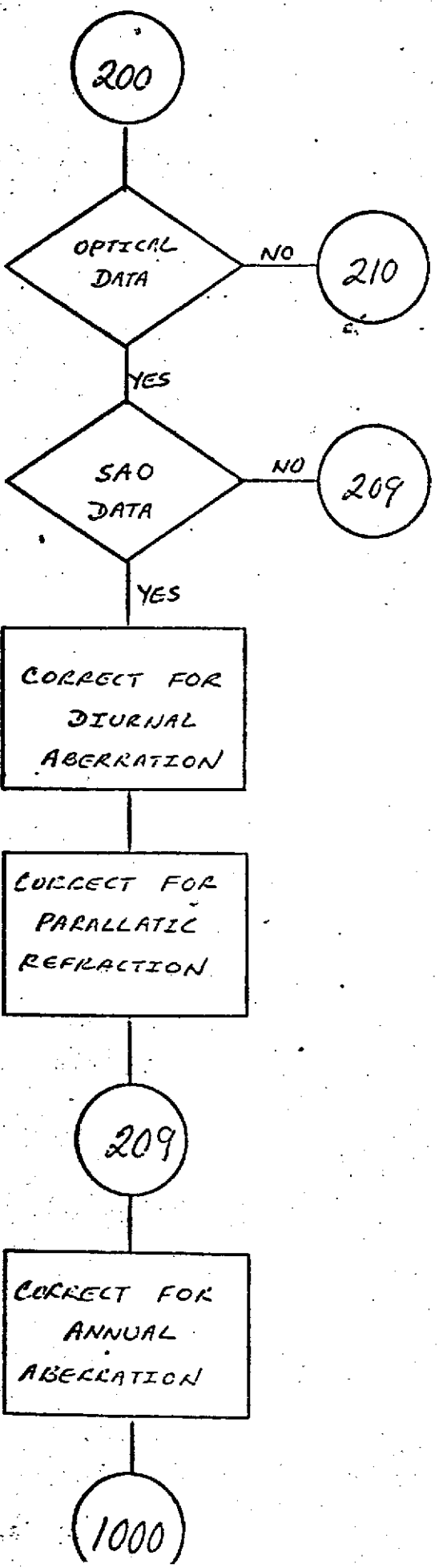
```
      OBSO1=OBSO1+GATE(I1)*DFLOAT(I2)                                   PROC 112
      TRANTM=DFLOAT(I2)*GATE(I1)/VLIGHT                                 PROC 113
      GO TO 150                                                         PROC 114
  100 TRANTM=R(ISAT1)/VLIGHT                                            PROC 115
  150 DAY=DAY-TRANTM/86400.D0                                           PROC 116
C RECALCULATE ORBIT & GREENWICH HOUR ANGLE                              PROC 117
      CALL ORBIT(DAY)                                                   PROC 118
      THETG=GHAAN(DAY,ISTA)                                             PROC 119
      GO TO 1000                                                        PROC 120
  200 IF(MTYPE.NE.1) GO TO 210                                          PROC 121
      IF(ISTANO(ISTA).LT.9000.AND.ISTANO(ISTA).NE.8009) GO TO 209       PROC 122
C DIURNAL ABERRATION                                                    PROC 123
      T2= IDINT(DAY)                                                    PROC 124
      UT=DAY-T2                                                         PROC 125
      DCOSLT=DCOS(FLAT(ISTA))                                           PROC 126
      H=UT*24.D0+9.8565D0*IDINT(UT*24.D0)/3.6D3                         PROC 127
      H=H*DTWOPI/24.D0+RLON(ISTA)+THETG-OBSO1                           PROC 128
      OBSO2=OBSO2+0.0213D0*S2R*15.D0*DCOSLT*DSIN(H)*DSIN(OBSO2)         PROC 129
      OBSO1=OBSO1+0.322D0*S2R*DCOSLT*DCOS(H)/DCOS(OBSO2)                PROC 130
C PARALLACTIC REFRACTION                                                PROC 131
      PSAT(1)=UHAT(2,ISAT1)                                             PROC 132
      PSAT(2)=-UHAT(1,ISAT1)                                            PROC 133
      PSAT(3)=0.D0                                                      PROC 134
      STASAT(1)=STAXYZ(2,ISTA)*XYZ(3,ISAT1)                             PROC 135
     .         -STAXYZ(3,ISTA)*XYZ(2,ISAT1)                             PROC 136
      STASAT(2)=-STAXYZ(1,ISTA)*XYZ(3,ISAT1)                            PROC 137
     .          +STAXYZ(3,ISTA)*XYZ(1,ISAT1)                            PROC 138
      STASAT(3)=STAXYZ(1,ISTA)*XYZ(2,ISAT1)                             PROC 139
     .         -STAXYZ(2,ISTA)*XYZ(1,ISAT1)                             PROC 140
      T2=PSAT(1)**2+PSAT(2)**2                                          PROC 141
      UT=STASAT(1)**2+STASAT(2)**2+STASAT(3)**2                         PROC 142
      T2=DSQRT(T2)                                                      PROC 143
      UT=DSQRT(UT)                                                      PROC 144
      DO 205 J=1,3                                                      PROC 145
      PSAT(J)=PSAT(J)/T2                                                PROC 146
  205 STASAT(J)=STASAT(J)/UT                                            PROC 147
      CO=DOTPRD(PSAT,STASAT)                                            PROC 148
      STPSAT(1)=UHAT(3,ISAT1)*PSAT(2)-UHAT(2,ISAT1)*PSAT(3)             PROC 149
      STPSAT(2)=-UHAT(3,ISAT1)*PSAT(1)+UHAT(1,ISAT1)*PSAT(3)            PROC 150
      STPSAT(3)=UHAT(2,ISAT1)*PSAT(1)-UHAT(1,ISAT1)*PSAT(2)             PROC 151
      UT=DSQRT(STPSAT(1)**2+STPSAT(2)**2+STPSAT(3)**2)                  PROC 152
      SO=DOTPRD(STPSAT,STASAT)/UT                                       PROC 153
      Z=0.25D0*DTWOPI-ELEV(ISAT1)                                       PROC 154
      IF(ELEV(ISAT1).GT.0.0D0) GO TO 206                                PROC 155
      Z=0.0D0                                                           PROC 156
      PRINT 10000                                                       PROC 157
10000 FORMAT('0**********$ ELEVATION NEGATIVE. MAXIMUM CORRECTION ',    PROC 158
     1   'FOR PARALLACTIC REFRACTION USED. $*********'/)                PROC 159
  206 RCOSZ=R(ISAT1)*DCOS(Z)                                            PROC 160
      IF(RCOSZ.GT.1.1D6) GO TO 207                                      PROC 161
      H=1.0D0-DEXP(-1.38D-4*RCOSZ)                                      PROC 162
      GO TO 208                                                         PROC 163
  207 H=1.0D0                                                           PROC 164
  208 DR=0.435D0*4.84813D0*DTAN(Z)*H/RCOSZ                              PROC 165
      OBSO2=OBSO2+DR*CO                                                 PROC 166
      OBSO1=OBSO1+DR*SO/T2                                              PROC 167
```

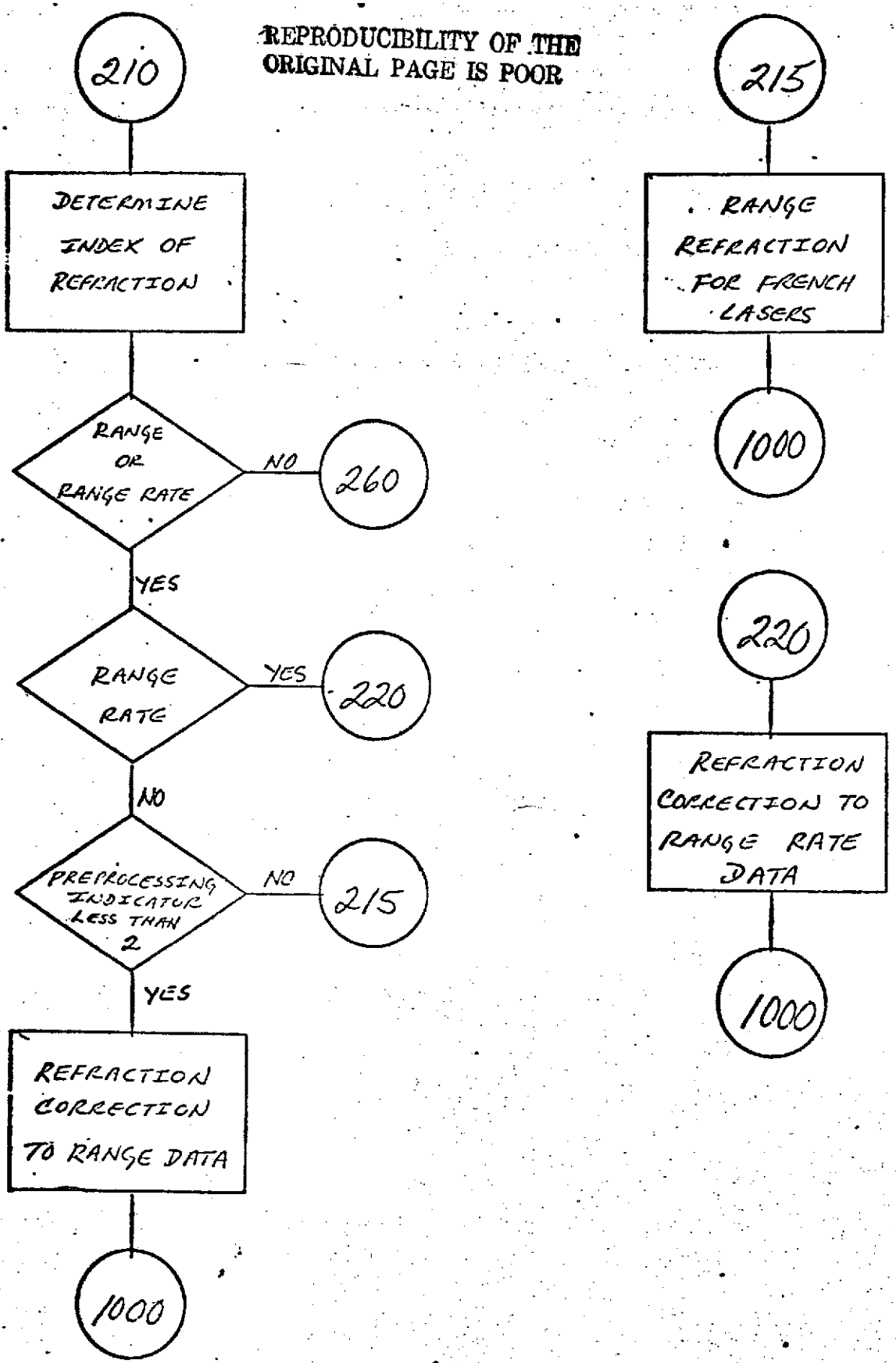
```
C ANNUAL ABERRATION                                                    PROC 168
  209 J=ISTAND(ISTA)                                                   PROC 169
      IF(.NOT.(J.EQ.8009.OR.J.EQ.8015.OR.J.EQ.8019.OR.J.EQ.8030.OR.    PROC 170
     *    J.EQ.8014.OR.J.EQ.8018.OR.J.EQ.8023.OR.J.EQ.8027.OR.        PROC 171
     *    J.EQ.8035.OR.J.EQ.8042.OR.J.EQ.8051.OR.J.EQ.8052.OR.        PROC 172
     *    J.EQ.9431.OR.J.EQ.9432)) GO TO 1000                         PROC 173
      DJ=DJUL(DAY)                                                     PROC 174
      CO=SUN(DJ,SO,OR,E)                                               PROC 175
      CALL EPHEM(DAY,.FALSE.)                                          PROC 176
      CE=DCOS(E)                                                       PROC 177
      L=DATAN2(SUNY,CE*SUNX)                                           PROC 178
      CL=DCOS(L)                                                       PROC 179
      SL=CL*SUNY/(SUNX*CE)                                             PROC 180
      CA=DCOS(OBSO1)                                                   PROC 181
      SA=DSIN(OBSO1)                                                   PROC 182
      CD=DCOS(OBSO2)                                                   PROC 183
      SD=DSIN(OBSO2)                                                   PROC 184
      TE=DSIN(E)/CE                                                    PROC 185
      OBSO1=OBSO1-20.5D0*S2R*(CA*CL*CE+SA*SL)/CD                       PROC 186
      OBSO2=OBSO2-20.5D0*S2R*(CL*CE*(TE*CD-SA*SD)+CA*SD*SL)            PROC 187
      GO TO 1000                                                       PROC 188
  210 SINE=RENV(3,ISAT1)                                               PROC 189
      SRINDX=STDNDX                                                    PROC 190
      IF(SRFNDX.NE.0.D0) SRINDX=SRFNDX                                 PROC 191
      IF(MTYPE.GT.3) GO TO 260                                         PROC 192
      VCORR=RCONST*SRINDX                                              PROC 193
      IF(VHFCHN) VCORR=VCORR+REFION(MTYPE,ISTA,DAY)                    PROC 194
      IF(MTYPE.EQ.3) GO TO 220                                         PROC 195
      IF(PRETYP(2).GT.2) GO TO 215                                     PROC 196
C RANGE REFRACTION                                                     PROC 197
      OBSO1=OBSO1-VCORR/(0.02600+SINE)                                 PROC 198
      GO TO 1000                                                       PROC 199
  215 COTE=DSQRT(1.000-SINE**2)/SINE                                   PROC 200
C RANGE REFRACTION FOR FRENCH LASER                                    PROC 201
      OBSO1=OBSO1-SRFNDX/(SINE+COTE*1.0D-3)                            PROC 202
      GO TO 1000                                                       PROC 203
C APPLY REFRACTION CORRECTION TO RANGE RATE DATA                       PROC 204
  220 RATE1=OBSDOT(7,ISTA,EDOT)                                        PROC 205
      OBSO1=OBSO1+VCORR*EDOT*DSQRT(1.000-SINE**2)/(0.02600+SINE)**2    PROC 206
      GO TO 1000                                                       PROC 207
  260 CCSE2=1.00-SINE**2                                               PROC 208
      COSE=DSQRT(CCSE2)                                                PROC 209
      VCORR=1.D-6*SRINDX                                               PROC 210
      IF(VHFCHN.OR.MTYPE.EQ.5) VCORR=VCORR+REFION(MTYPE,ISTA,DAY)      PROC 211
      DELTAE=-VCORR/(.01644D0+.9300*SINE/COSE)                         PROC 212
      IF(MTYPE.EQ.7) GO TO 290                                         PROC 213
      SINA=RENV(3,ISAT1)/COSE                                          PROC 214
      COSA=RENV(2,ISAT1)/COSE                                          PROC 215
      IF(MTYPE.EQ.6) GO TO 270                                         PROC 216
C APPLY REFRACTION CORRECTION TO MINITRACK DATA                        PROC 217
      OBSO1=OBSO1-SINA*SINE*DELTAE                                     PROC 218
      OBSO2=OBSO2-COSA*SINE*DELTAE                                     PROC 219
      GO TO 1000                                                       PROC 220
C APPLY REFRACTION CORRECTION TO X-Y ANGLE DATA                        PROC 221
  270 OBSO1=OBSO1-SINA*DELTAE/(SINE**2+SINA**2*COSE2)                  PROC 222
      OBSO2=OBSO2-COSA*SINE*DELTAE/DSQRT(1.000-COSE2*COSA**2)          PROC 223
```

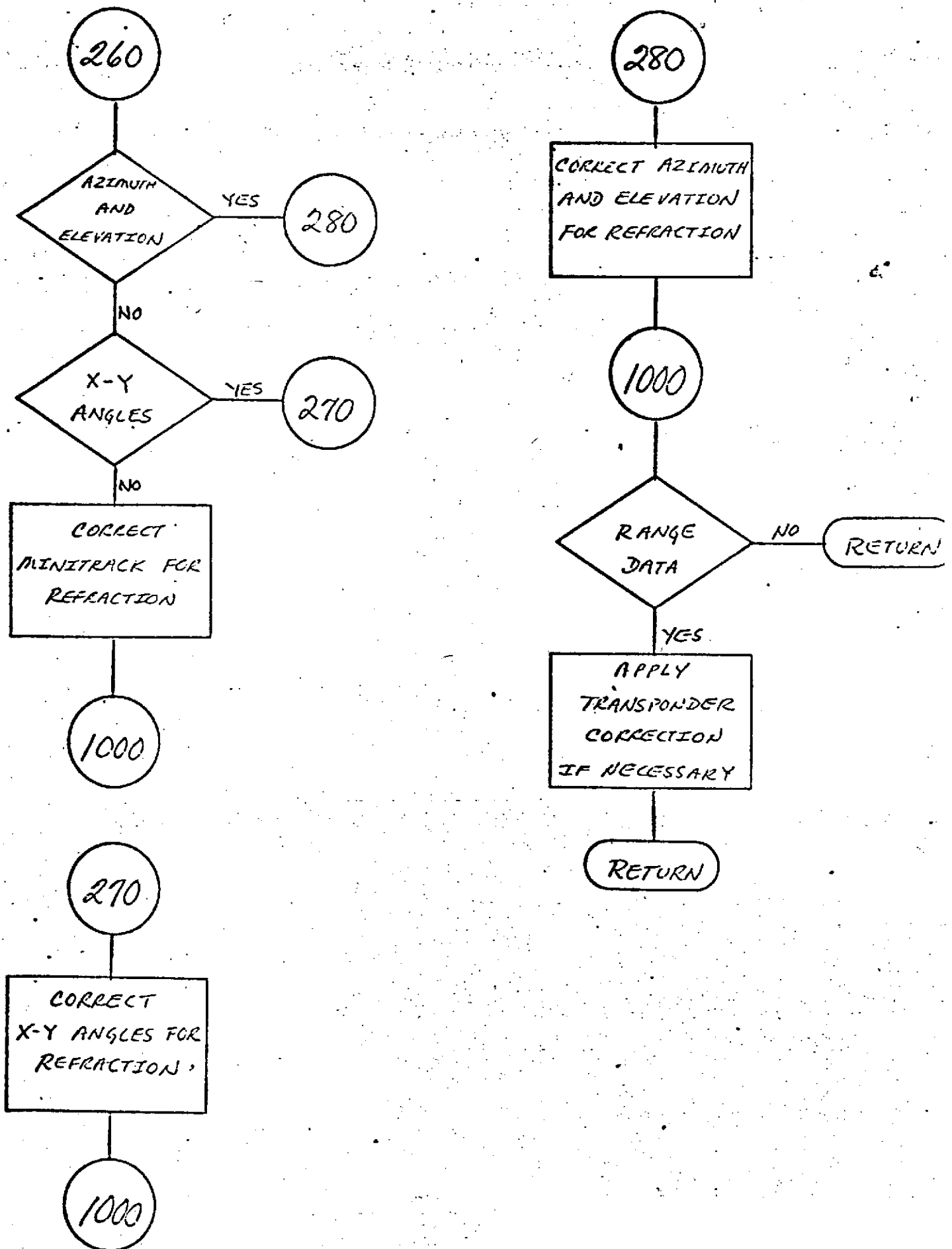
```
      GO TO 1000                                                        PPOC 224
C APPLY REFRACTION CORRECTION TO ELEVATION ANGLE DATA                   PPOC 225
  280 OBS02=OBS02+DELTAE                                                PPOC 226
 1000 CONTINUE                                                          PPOC 227
      IF(MTYPE.NE.2.OR.CHANEL.EQ.0) RETURN                              PPOC 228
C APPLY TRANSPONDER DELAY CORRECTION TO RANGE DATA                      PPOC 229
      ISAT=NUMBR4(ISATID(ISAT1),KSATNO,NSATNO)                          PPOC 230
      IF(ISAT.EQ.0) RETURN                                              PPOC 231
      ISAT=(ISAT+2)/3                                                   PPOC 232
      IF(ISAT.EQ.8.AND.CHANEL.GT.3) GO TO 265                           PPOC 233
      GO TO 290                                                         PPOC 234
  265 ISAT=ISAT+1                                                       PPOC 235
      CHANEL=CHANEL-3                                                   PPOC 236
      IF(CHANEL.GT.1) RETURN                                            PPOC 237
  290 CHANEL=(CHANEL+2)/2                                               PPOC 238
      RDOT=OBSDOT(2,ISTA,RDOT2)                                         PPOC 239
      DELR=0.00                                                         PPOC 240
      DO 300 I=1,3                                                      PPOC 241
  300   DELR=DELR+RDOT**(I-1)*TRANSP(I,CHANEL,ISAT)                     PPOC 242
      DELR=-DELR*.5D-9*VLIGHT                                           PPOC 243
      OBS01=OBS01+DELR                                                  PPOC 244
      RETURN                                                            PPOC 245
      END                                                               PPOC 246
```




B
ORBIT
RECALCULATE ORBIT
GRHRAN
RECALCULATE GREENWICH HOUR ANGLE
1000


200
OPTICAL DATA
NO
210
YES
SAO DATA
NO
209
YES
CORRECT FOR DIURNAL ABERRATION
CORRECT FOR PARALLATIC REFRACTION
209
CORRECT FOR ANNUAL ABERRATION
1000




260
AZIMUTH AND ELEVATION
YES
280
NO
X-Y ANGLES
YES
270
NO
CORRECT MINITRACK FOR REFRACTION
1000
270
CORRECT X-Y ANGLES FOR REFRACTION
1000
280
CORRECT AZIMUTH AND ELEVATION FOR REFRACTION
1000
RANGE DATA
NO
RETURN
YES
APPLY TRANSPONDER CORRECTION IF NECESSARY
RETURN

RANDOM

DESCRIPTION

RANDOM is a subroutine used to read, write and read/rewrite satellite observation data.

Since each measurement must be rewritten after it has been preprocessed, direct access methods of I/O are required and since large quantities of data are processed, it is necessary to use a large data buffer.

RANDOM buffers the observation data and calls the direct access reading and writing system subroutines DREAD and DWRITE to perform the I/O operations.

NAME	RANDOM
ENTRY POINT	PURPOSE
RANDRD	TO UNBLOCK & READ OBSERVATIONS FROM RANDOM ACCESS FILE
RANDWR	TO BLOCK & WRITE OBSERVATIONS ON RANDOM ACCESS FILE
CALLING SEQUENCE	CALL RANDRD
CALLING SEQUENCE	CALL RANDWR
SUBROUTINES USED	ERROR
COMMON BLOCKS	PREBLK
INPUT FILES	IDISK - RANDOM ACCESS DATA FILE
OUTPUT FILES	IDISK - RANDOM ACCESS DATA FILE

```
      SUBROUTINE RANDRD                                                 RAND  26
      IMPLICIT REAL*8 (A-H,O-Z)                                         RAND  27
      COMMON/PREBLK/A(16),IREC                                          RAND  28
      INTEGER A,BUF(16,113),NREC/113/,KBUF/1/,NWORDS/16/                RAND  29
      LOGICAL WSWTCH/.FALSE./                                           RAND  30
      EQUIVALENCE (DISK,IDISK)                                          RAND  31
      DATA IDISK/12/                                                    RAND  32
      K=1                                                               RAND  33
      GO TO 10                                                          RAND  34
      ENTRY RANDWR                                                      RAND  35
      WSWTCH=.TRUE.                                                     RAND  36
      K=2                                                               RAND  37
10    IBUF=(IREC-1)/NREC+1                                              RAND  38
      IF(IBUF.NE.KBUF) GO TO (50,90),K                                  RAND  39
20    JREC=IREC-(IBUF-1)*NREC                                           RAND  40
      GO TO (30,70),K                                                   RAND  41
C UNBLOCK DATA                                                          RAND  42
30    DO 40 I=1,NWORDS                                                  RAND  43
40    A(I)=BUF(I,JREC)                                                  RAND  44
      RETURN                                                            RAND  45
50    IF(WSWTCH)CALL DWRITE(IDISK,KBUF,BUF)                             RAND  46
      KBUF=IBUF                                                         RAND  47
      WSWTCH=.FALSE.                                                    RAND  48
C READ BLOCK                                                            RAND  49
      CALL DREAD(IDISK,KBUF,BUF,&60)                                    RAND  50
      GO TO 20                                                          RAND  51
60    CALL ERROR(11,IDISK)                                              RAND  52
      RETURN                                                            RAND  53
C BLOCK DATA                                                            RAND  54
70    DO 80 I=1,NWORDS                                                  RAND  55
```

```
   80     BUF(I,JREC)=A(I)                                          RAND  56
          RETURN                                                    RAND  57
C WRITE BLOCK                                                       RAND  58
   90     CALL DWRITE(IDISK,KBUF,BUF)                               RAND  59
          KBUF=IBUF                                                 RAND  60
          GO TO 20                                                  RAND  61
 1000     FORMAT(1X,Z8,I6,20A4,4(1X,Z8))                            RAND  62
          END                                                       RAND  63
```

REARG

DESCRIPTION

REARG recomputes back values of acceleration and the sum array for given changes in stepsize. If the step is to be increased, first REARG will double the stepsize and then reduce to the desired stepsize. When the step is decreased, interpolation is used to obtain midpoint values for the integration values, followed by calls to F and VEVAL to evaluate the accelerations at each midpoint.

NAME REARG

PURPOSE TO RECOMPUTE BACK VALUES OF ACCELERATION AND THE SUM ARRAY FOR GIVEN CHANGES IN STEPSIZE

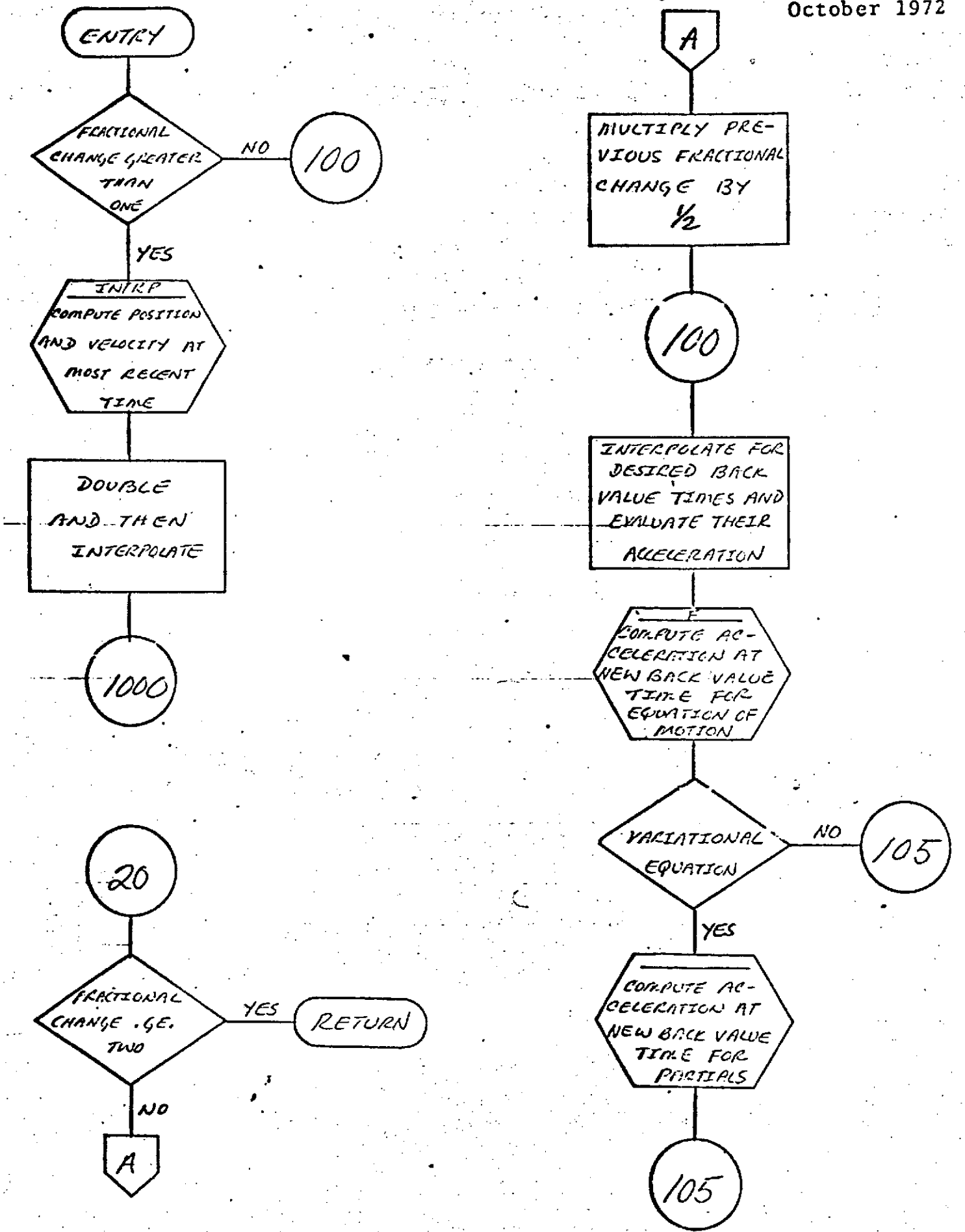
CALLING SEQUENCE CALL REARG(M,M1,IORDR1,IORDER,H1,H2,T1,T2,FCT,NN, SUM,SUM1,COEFP,COEFV,Y,Y1,VAR,FCT1,P)

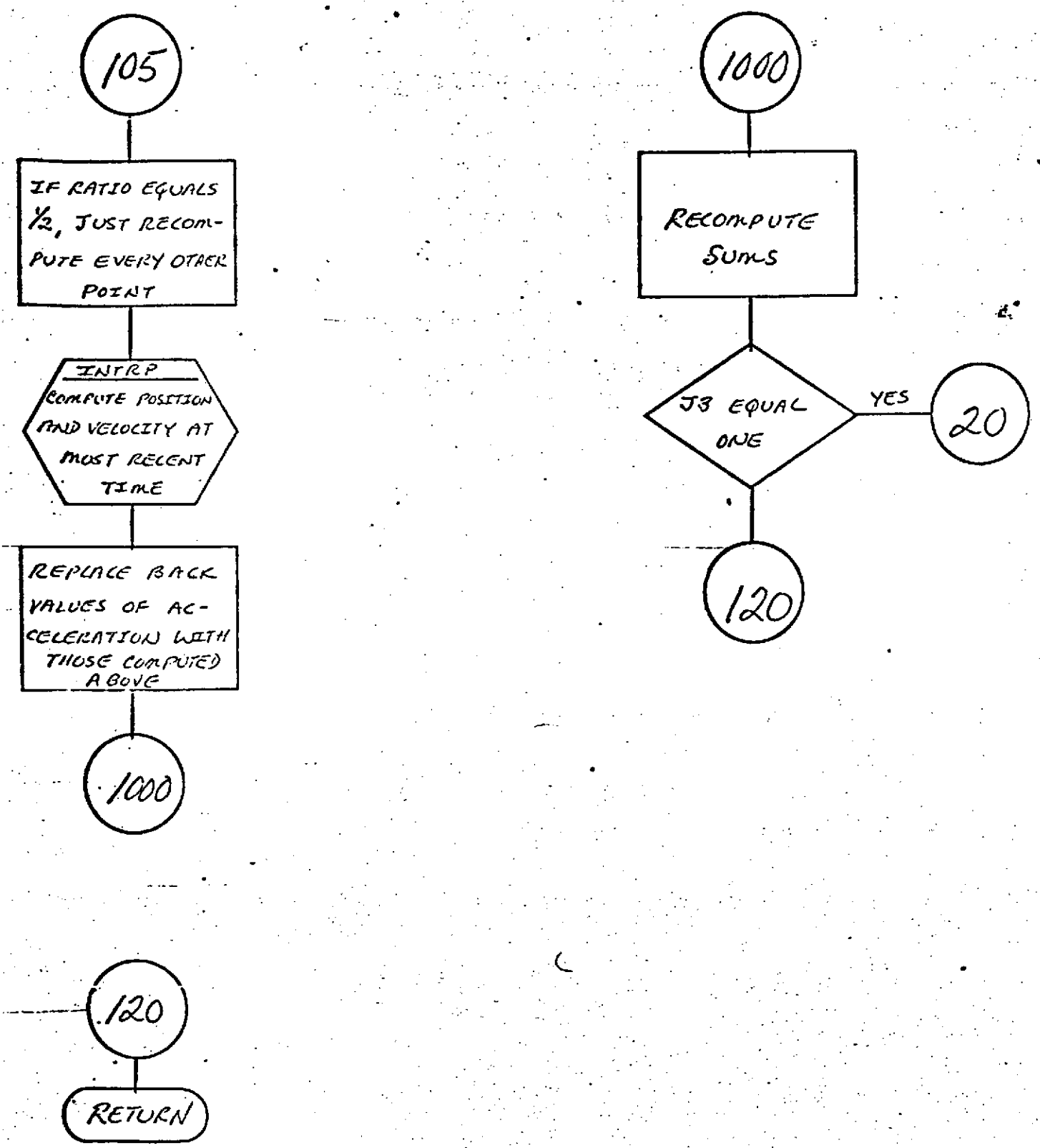
SYMBOL	TYPE	DESCRIPTION
M	I	INPUT - DISPLACEMENT VALUES USED BY INTEGRATOR
M1	I	INPUT - DISPLACEMENT VALUES USED BY INTEGRATOR
IORDR1	I	INPUT - ORDER
IORDER	I	INPUT - ORDER
H1	DP	INPUT - STEPSIZE
H2	DP	INPUT - STEPSIZE
T1	DP	INPUT - TIME
T2	DP	INPUT - TIME
FCT (3,1)	DP	INPUT - BACK VALUE ARRAY OF ACCELERATION
NN	I	INPUT - NUMBER OF EQUATIONS
SUM (2,3,1)	DP	OUTPUT - SUM ARRAY USED BY INTEGRATOR AND INTERPOLATOR
SUM1 (2,3)	DP	OUTPUT - SUM ARRAY USED BY INTEGRATOR AND INTERPOLATOR
COEFP (1)	DP	INPUT - POSITION COEFFICIENTS
COEFV (1)	DP	INPUT - VELOCITY COEFFICIENTS
Y (6,1)	DP	INPUT - ARRAY OF STATE VARIABLES
Y1 (6)	DP	INPUT - ARRAY OF STATE VARIABLES
VAR	L	INPUT - VARIATIONAL EQUATION SWITCH
FCT1 (3)	DP	INPUT - BACK VALUES OF ACCELERATION
P	DP	INPUT - FRACTIONAL STEPSIZE CHANGE

SUBROUTINES USED	INTRP F VEVAL
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	PRINTER

```
      SUBROUTINE REARG(M,M1,IORDR1,IORDER,H1,H2,T1,T2,FCT,NN,SUM,        REAR  68
     *  SUM1,COEFP,COEFV,Y,Y1,     VAR,FCT1,P)                           REAR  69
      IMPLICIT REAL*8 (A-H,O-Z)                                         REAR  70
      LOGICAL SWITCH,VAR,BACK                                           REAR  71
      DIMENSION FCT(3,1),ACC(3),SUM(2,3,1),SUM1(2,3)  ,COEFP(1),COEFV(1)REAR  72
     *  ,Y(6,1),Y1(6)  ,FCT1(3)                                         REAR  73
C TEMPORARY PRINT STATEMENT                                             REAR  74
      H=P*H2                                                            REAR  75
      WRITE(6,1) VAR,P,H                                                REAR  76
    1 FORMAT(1X,3G20.8)                                                 REAR  77
      H=H2                                                              REAR  78
      FACT=P                                                            REAR  79
      NEQN=1                                                            REAR  80
      IF(VAR)NEQN=NN-1                                                  REAR  81
      IOL2=IORDER-2                                                     REAR  82
      IOL1=IOL2+1                                                       REAR  83
      IF(FACT.LT.1.D0)GO TO 100                                         REAR  84
C COMPUTE POSITIONS & VELOCITY AT INTEGRATOR TIME                       REAR  85
    5 CALL INTRP(0.D0,H2,IORDER,NEQN,Y,FCT,M,SUM)                       REAR  86
    9 DO 10 I=1,IOL2                                                    REAR  87
      K1=M -I                                                           REAR  88
      K2=M-2*I                                                          REAR  89
      DO 10 L=1,NEQN                                                    REAR  90
      LL=(L-1)*M                                                        REAR  91
      FCT(1,LL+K1)=FCT(1,LL+K2)                                         REAR  92
      FCT(2,LL+K1)=FCT(2,LL+K2)                                         REAR  93
      FCT(3,LL+K1)=FCT(3,LL+K2)                                         REAR  94
   10 CONTINUE                                                          REAR  95
   11 H=H+H                                                             REAR  96
      H2=H2+H2                                                          REAR  97
      IF(.NOT.VAR)H1=H1+H1                                              REAR  98
      J3=1                                                              REAR  99
C RECOMPUTE SUMS                                                        REAR 100
      GO TO 1000                                                        REAR 101
   20 CONTINUE                                                          REAR 102
      IF(FACT.GE.2.D0) RETURN                                           REAR 103
      FACT=FACT*.5D0                                                    REAR 104
  100 SWITCH = .FALSE.                                                  REAR 105
      DELTAH=FACT*H                                                     REAR 106
      IF((H-DELTAH-DELTAH).EQ.0.D0)SWITCH=.TRUE.                        REAR 107
      I=0                                                               REAR 108
C INTERPOLATE FOR DESIRED BACK VALUE TIMES                              REAR 109
  101 I=I+1                                                             REAR 110
      S1=-DFLOAT(I )*DELTAH                                             REAR 111
```

```
      TIM=T2+S1                                                          REAR 112
      S=(TIM-T1)/H1                                                      REAR 113
      CALL INTRP(S,H1,IORDR1,1,Y1,FCT1,M1,SUM1)                          REAR 114
      IF(VAR)CALL F(TIM,YI,ACC,.TRUE.)                                   REAR 115
      IF(.NOT.VAR)CALL F(TIM,YI,FCT(1,M-IOL2-I),.FALSE.)                 REAR 116
      IF(.NOT.VAR)GO TO 105                                              REAR 117
      S=(TIM-T2)/H2                                                      REAR 118
      CALL INTRP(S,H2,IORDER,NEQN,Y,FCT,M ,SUM)                          REAR 119
  102 CALL VEVAL(Y,FCT(1,M-IOL2-I),6,.TRUE.,M )                          REAR 120
C IF RATIO EQUALS 1/2, JUST RECOMPUTE EVERY OTHER POINT                  REAR 121
  105 IF(.NOT.SWITCH)GO TO 110                                           REAR 122
      I=I+1                                                              REAR 123
      IF(I.GT.IOL2)GO TO 110                                             REAR 124
      LL=M-IOL2-I                                                        REAR 125
      LLL=M-(I/2)                                                        REAR 126
      FCT(1, LL)=FCT(1,LLL)                                              REAR 127
      FCT(2, LL)=FCT(2,LLL)                                              REAR 128
      FCT(3, LL)=FCT(3,LLL)                                              REAR 129
  110 IF(I.LT.IOL2)GO TO 101                                             REAR 130
C COMPUTE POSITION & VELOCITY AT INTEGRATOR TIME                         REAR 131
  111 CALL INTRP(0.D0,H2,IORDER,NEQN,Y,FCT,M ,SUM)                       REAR 132
C REPLACE BACK VALUES                                                    REAR 133
      DO 115 L=1,NEQN                                                    REAR 134
      LM=L*M                                                             REAR 135
      DO 115 I=1,IOL2                                                    REAR 136
      LL=LM-I                                                            REAR 137
      LLL=LL-IOL2                                                        REAR 138
      FCT(1,LL)=FCT(1,LLL)                                               REAR 139
      FCT(2,LL)=FCT(2,LLL)                                               REAR 140
      FCT(3,LL)=FCT(3,LLL)                                               REAR 141
  115 CONTINUE                                                           REAR 142
      H=DELTAH                                                           REAR 143
      H2=DELTAH                                                          REAR 144
      IF(.NOT.VAR)H1=DELTAH                                              REAR 145
      J3=2                                                               REAR 146
C RECOMPUTE SUMS                                                         REAR 147
      GO TO 1000                                                         REAR 148
  120 CONTINUE                                                           REAR 149
      RETURN                                                             REAR 150
 1000 CONTINUE                                                           REAR 151
      HS=H**2                                                            REAR 152
      DO 1030 N=1,NEQN                                                   REAR 153
      NM=N*M                                                             REAR 154
      LLL=NM +1                                                          REAR 155
      DO 1030 J=1,2                                                      REAR 156
      A=0.D0                                                             REAR 157
      B=0.D0                                                             REAR 158
      DO 1029 I=1,IOL2                                                   REAR 159
      LL=LLL-I                                                           REAR 160
      A=A-COEFV(I)*FCT(J,LL)                                             REAR 161
      B=B-COEFP(I)*FCT(J,LL)                                             REAR 162
 1029 CONTINUE                                                           REAR 163
 1028 A=A-COEFV(IOL1)*FCT(J, NM-IOL2)+FCT(J,NM)                          REAR 164
      A=Y(J+3,N)/H+A                                                     REAR 165
      SUM(1,J,N)=A                                                       REAR 166
      SUM(2,J,N)=Y(J,N)/HS   +A+B                                        REAR 167
 1030 CONTINUE                                                           REAR 168
      GO TO (2),120),J3                                                  REAR 169
      RETURN                                                             REAR 170
      END                                                                REAR 171
```


ENTRY
FRACTIONAL CHANGE GREATER THAN ONE
NO
100
YES
INTRP
COMPUTE POSITION AND VELOCITY AT MOST RECENT TIME
DOUBLE AND THEN INTERPOLATE
1000
20
FRACTIONAL CHANGE .GE. TWO
YES
RETURN
NO
A
A
MULTIPLY PREVIOUS FRACTIONAL CHANGE BY ½
100
INTERPOLATE FOR DESIRED BACK VALUE TIMES AND EVALUATE THEIR ACCELERATION
F
COMPUTE ACCELERATION AT NEW BACK VALUE TIME FOR EQUATION OF MOTION
VARIATIONAL EQUATION
NO
105
YES
COMPUTE ACCELERATION AT NEW BACK VALUE TIME FOR PARTIALS
105


105
IF RATIO EQUALS ½, JUST RECOMPUTE EVERY OTHER POINT
INTRP
COMPUTE POSITION AND VELOCITY AT MOST RECENT TIME
REPLACE BACK VALUES OF ACCELERATION WITH THOSE COMPUTED ABOVE
1000
120
RETURN
1000
RECOMPUTE SUMS
J3 EQUAL ONE
YES
20
120

REFCOR

DESCRIPTION

Subroutine REFCOR rotates a vector between the true equator and equinox of a reference date and the true equator and equinox of date.

This routine invokes subroutines PRECES and NUTATE at the start of each arc to obtain the precession rotation matrix from the reference to 1950.0 and the nutation rotation matrix from mean to true equator and equinox of the reference date. These latter are then inverted and combined to rotate from the mean equator and equinox of 1950.0 to the reference time.

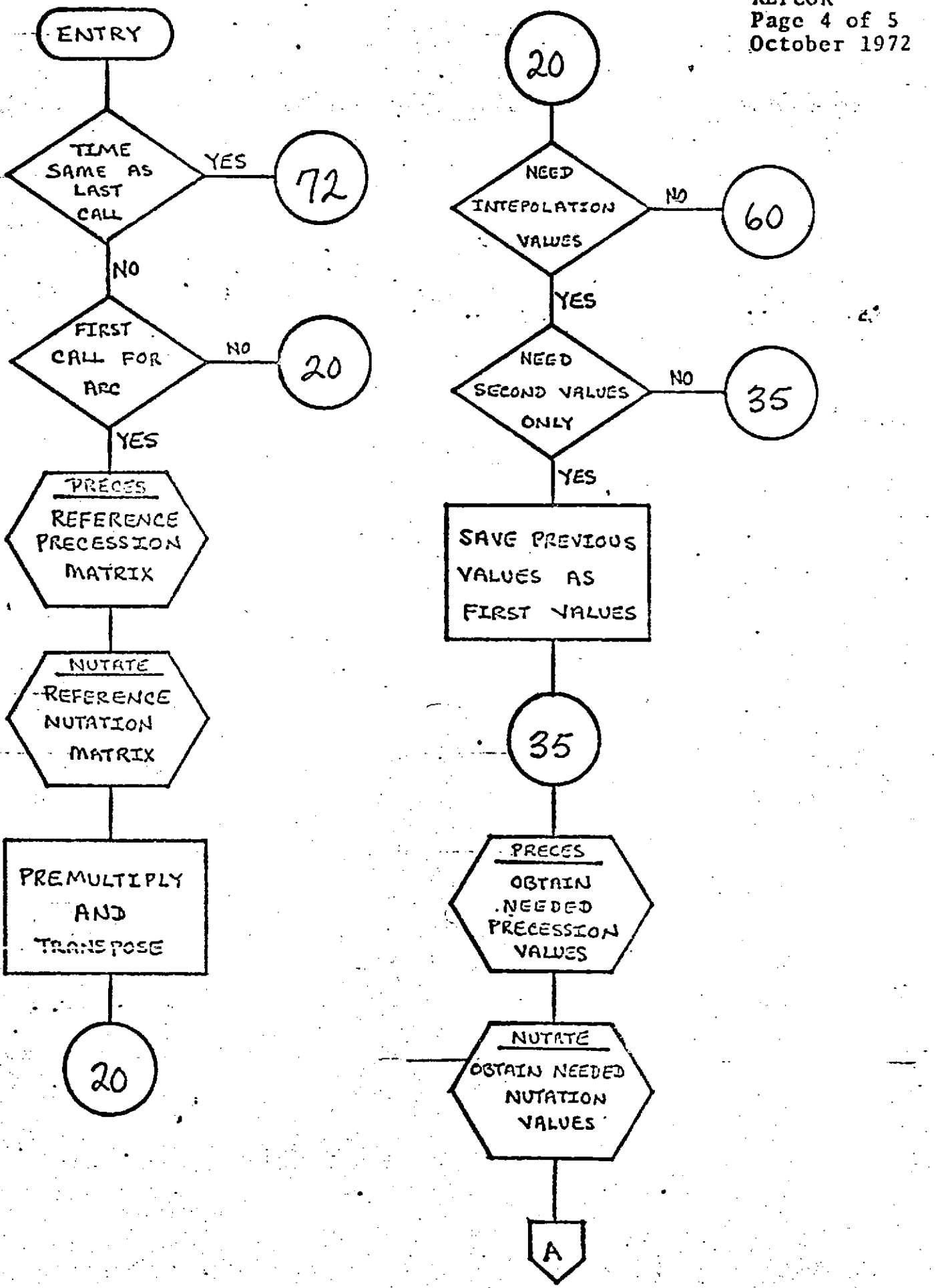
The routine obtains rotation matrices from PRECES and NUTATE for consecutive days, stepping in time as required by the input time. These matrices are combined with the above rotation matrix from mean equator and equinox of 1950.0 to the reference time to produce a rotation matrix for 0 hours on each consecutive day.

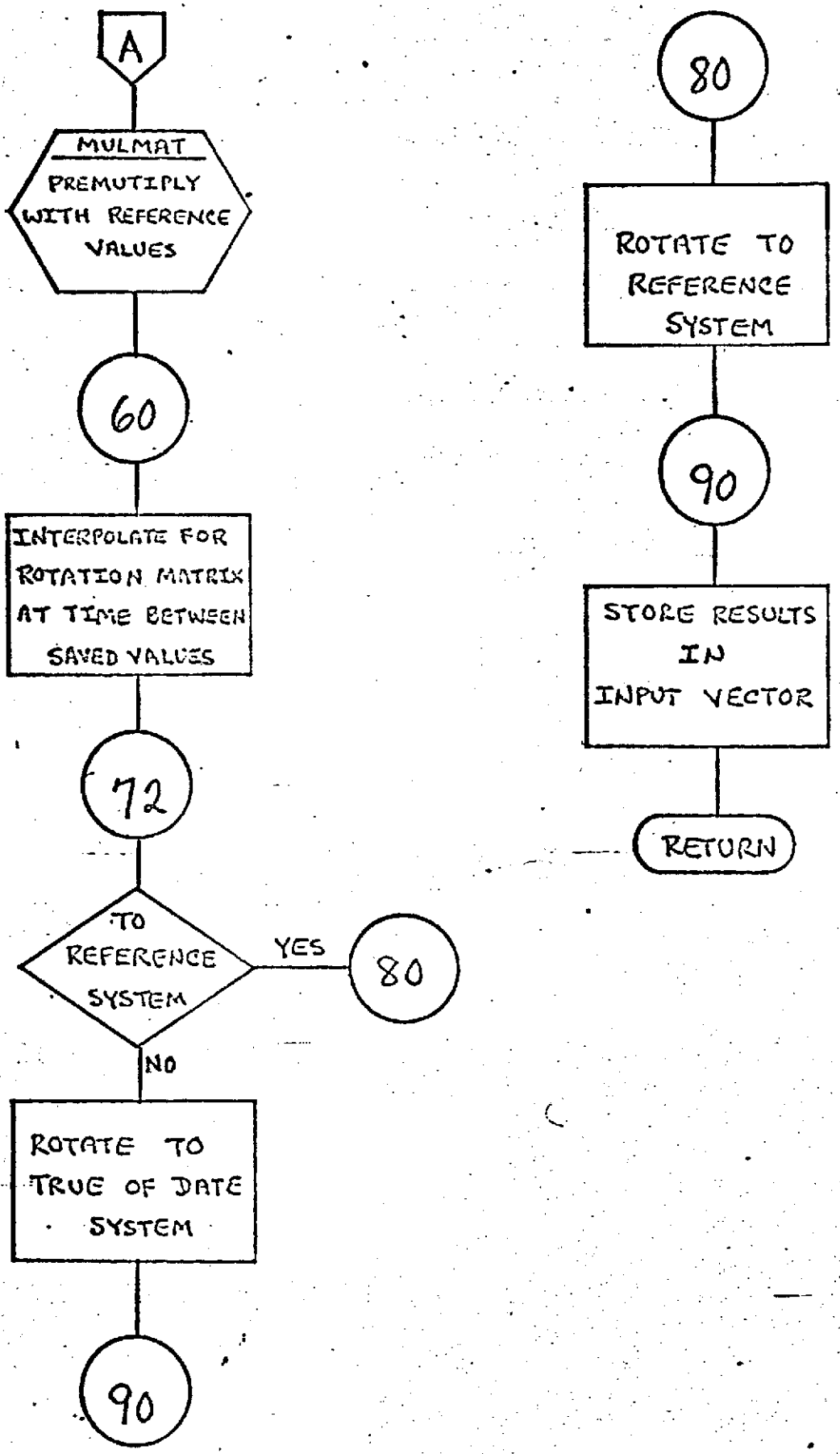
REFCOR linearly interpolates between these two consecutive rotation matrices to obtain the appropriate rotation matrix for the time of date. The input vector is then rotated either to or from the reference system.

NAME	REFCOR	
PURPOSE	TO PRECESS A VECTOR BETWEEN THE TRUE EQUATOR AND EQUINOX OF A REFERENCE TIME AND THE TRUE EQUATOR AND EQUINOX OF DATE	
CALLING SEQUENCE	CALL REFCOR(DAY,TO,X)	
SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT - NUMBER OF DAYS FROM JAN 0.0 OF REFERENCE YEAR
TO	L	INPUT - .TRUE. - CONVERT FROM TRUE OF DATE TO TRUE OF REFERENCE DATE .FALSE. - CONVERT FROM TRUE OF REFERENCE DATE TO TRUE OF DATE
X (3)	DP	INPUT - VECTOR TO BE CONVERTED OUTPUT - CONVERTED VECTOR
SUBROUTINES USED	PRECES NUTATE MULMAT	
COMMON BLOCKS	INITBK CTIME	
INPUT FILES	NONE	
OUTPUT FILES	NONE	
REFERENCES	'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION	

```
      SUBROUTINE REFCOR(DAY,TO,X)                                       REFC  3
      REAL*8 P(3,3),N(3,3),REF(3,3),PRE(3,3,2),X(3),Y(3),DAY,DAYP,      REFC  3
     *       DAYREF,PREINT(3,3)                                         REFC  3
      REAL*8 DAYSV/-1.0D0/                                              REFC  3
      INTEGER IDAY(2)/2*-1/                                             REFC  4
      COMMON/INITBK/IG2(54),NOTIST,IG3(2)                               REFC  4
      LOGICAL NOTIST                                                    REFC  4
      COMMON/CTIME/DATAEP(2),DAYREF,G1(19)                              REFC  4
      LOGICAL TO                                                        REFC  4
      IF(.NOT.NOTIST) DAYSV=-1.0D0                                      REFC  4
      IF(DAY.EQ.DAYSV) GO TO 72                                         REFC  4
      IF(NOTIST) GO TO 20                                               REFC  4
      NOTIST=.TRUE.                                                     REFC  4
C GET PRECESSION & NUTATION FOR REFERENCE TIME                          REFC  4
      CALL PRECES(DAYREF,P)                                             REFC  5
      CALL NUTATE(DAYREF,N)                                             REFC  5
C PREMUTIPLY & TRANSPO                                                  REFC  5
      DO 10 I=1,3                                                       REFC  5
      DO 10 J=1,3                                                       REFC  5
      REF(I,J)=0.D0                                                     REFC  5
```

```
      DO 10 K=1,3                                                    REFC  56
 10   REF(I,J)=P(J,K)*N(K,I)+REF(I,J)                                REFC  57
C ARE VALUES NEEDED FOR INTERPOLATION ?                              REFC  58
 20   ID=DAY                                                         REFC  59
      IF(ID.EQ.IDAY(1)) GO TO 60                                     REFC  60
      INEED=1                                                        REFC  61
C 2 VALUES OR ONLY THE SECOND ?                                      REFC  62
      IF(ID.NE.IDAY(2)) GO TO 35                                     REFC  63
      INEED=2                                                        REFC  64
C I NEED ONLY THE SECOND...SAVE THE PREVIOUS VALUES                  REFC  65
      DO 30 I=1,3                                                    REFC  66
      DO 30 J=1,3                                                    REFC  67
 30   PRE(I,J,1)=PRE(I,J,2)                                          REFC  68
      IDAY(1)=IDAY(2)                                                REFC  69
C GET PRECESSION & NUTATION VALUES FOR DAY OF INTEREST               REFC  70
 35   DO 50 M=INEED,2                                                REFC  71
      IDAY(M)=ID+M-1                                                 REFC  72
      DAYP=IDAY(M)                                                   REFC  73
      CALL PRECES(DAYP,P)                                            REFC  74
      CALL NUTATE(DAYP,N)                                            REFC  75
C PREMULTIPLY WITH REFERENCE VALUES                                  REFC  76
 50   CALL MULMAT(PRE(1,1,M),REF,P,N)                                REFC  77
C OBTAIN INTERPOLATED ROTATION MATRIX                                REFC  78
 60   DAYP=DAY-DFLOAT(IDAY(1))                                       REFC  79
      DO 70 I=1,3                                                    REFC  80
      DO 70 J=1,3                                                    REFC  81
 70   PREINT(I,J)=PRE(I,J,1)+DAYP*(PRE(I,J,2)-PRE(I,J,1))            REFC  82
C ROTATE INPUT VECTOR                                                REFC  83
 72   IF(TO) GO TO 80                                                REFC  84
C ROTATE FROM REFERENCE SYSTEM                                       REFC  85
      DO 75 I=1,3                                                    REFC  86
      Y(I)=0.000                                                     REFC  87
      DO 75 J=1,3                                                    REFC  88
 75   Y(I)=Y(I)+PREINT(J,I)*X(J)                                     REFC  89
      GO TO 90                                                       REFC  90
C ROTATE TO REFERENCE SYSTEM                                         REFC  91
 80   DO 85 I=1,3                                                    REFC  92
      Y(I)=0.000                                                     REFC  93
      DO 85 J=1,3                                                    REFC  94
 85   Y(I)=Y(I)+PREINT(I,J)*X(J)                                     REFC  95
C STORE RESULT IN INPUT VECTOR                                       REFC  96
 90   DO 100 I=1,3                                                   REFC  97
100   X(I)=Y(I)                                                      REFC  98
      DAYSV=DAY                                                      REFC  99
      RETURN                                                         REFC 100
      END                                                            REFC 101
```


ENTRY
TIME SAME AS LAST CALL
YES
72
NO
FIRST CALL FOR ARC
NO
20
YES
PRECES
REFERENCE PRECESSION MATRIX
NUTATE
REFERENCE NUTATION MATRIX
PREMULTIPLY AND TRANSPOSE
20
20
NEED INTEPOLATION VALUES
NO
60
YES
NEED SECOND VALUES ONLY
NO
35
YES
SAVE PREVIOUS VALUES AS FIRST VALUES
35
PRECES
OBTAIN NEEDED PRECESSION VALUES
NUTATE
OBTAIN NEEDED NUTATION VALUES
A


A
MULMAT
PREMUTIPLY WITH REFERENCE VALUES
60
INTERPOLATE FOR ROTATION MATRIX AT TIME BETWEEN SAVED VALUES
72
TO REFERENCE SYSTEM
YES
80
NO
ROTATE TO TRUE OF DATE SYSTEM
90
80
ROTATE TO REFERENCE SYSTEM
90
STORE RESULTS IN INPUT VECTOR
RETURN

NAME		REFION
PURPOSE		DUMMY IONOSPHERIC REFRACTION SUBROUTINE
CALLING SEQUENCE		X=REFION(MTYPE,ISTA,DAY)
SYMBOL	TYPE	DESCRIPTION
MTYPE	I	INPUT - MEASUREMENT TYPE
ISTA	I	INPUT - STATION INDEX
DAY	DP	INPUT - TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
REFION	R	OUTPUT - IONOSHEPIC REFRACTION VARIATION
SUBROUTINES USED		NONE
COMMON BLOCKS		NONE
INPUT FILES		NONE
OUTPUT FILES		NONE

```
      FUNCTION REFION(MTYPE,ISTA,DAY)                                   REFI  29
      REFION=0.                                                         REFI  30
      RETURN                                                            REFI  31
      END                                                               REFI  32
```

NAME RESPAR

ENTRY POINT PURPOSE

RESPR1 INITIALIZATION

RESPAR TO CALCULATE THE PARTIALS OF GEOPOTENTIAL RESONANCE COEFFICIENTS REQUESTING ADJUSTMENT

CALLING SEQUENCE CALL RESPR1(INDXCS,GPSIG,GPPAR)

SYMBOL	TYPE	DESCRIPTION
INDXCS (3,1)	I*2	INPUT - INDICES OF ADJUSTED GEOPOTENTIAL COEFFICIENTS
GPSIG (1)	DP	INPUT - SIGMAS ON ADJUSTED GEOPOTENTIAL COEFFICIENTS
GRPAR (3,1)	DP	OUTPUT - PARTIALS OF FORCE MODEL PARAMETERS

CALLING SEQUENCE CALL RESPAR

SUBROUTINES USED NONE

COMMON BLOCKS CPARAM INTBLK VRBLOK XYZ

INPUT FILES NONE

OUTPUT FILES NONE

REFERENCES 'GEODYN SYSTEMS DESCRIPTION'
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE RESPR1(INDXCS,GPSIG,GRPAR)                              RESP  39
      IMPLICIT REAL*8 (A-H,O-Z)                                          RESP  40
      LOGICAL CMPGPR                                                     RESP  41
      INTEGER*2 INDXCS                                                   RESP  42
      DIMENSION INDXCS(3,1),GPSIG(1),GRPAR(3,1)                          RESP  43
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,NGPCOM,     RESP  44
     *   NCSEST,CMPGPR,LIM1,LIM2,NOEN,NDENST,NTIDST,NTIDEN,INNRSW,       RESP  45
     *   NCONST,NDCONS                                                   RESP  46
      COMMON/INTBLK/THDOT1(58),NEON(2),ADDR(7)                           RESP  47
      COMMON/VRBLOK/XYSQ,CSML(31,2),U1D(3),P(33,30),AORN(30),            RESP  48
     *   TPSIM(39)                                                       RESP  49
      COMMON/XYZ/X,Y,Z,XDOT,YDOT,ZDOT,R,RSQ,ISAT,IFORCE(2)               RESP  50
      EQUIVALENCE (CP,P(2,1))                                            RESP  51
      RETURN                                                             RESP  52
      ENTRY RESPAR                                                       RESP  53
      C1=TPSIM(2)/RSQ                                                    RESP  54
      C2=CP/P                                                            RESP  55
```

```
      II=NEON(ISAT)-NCSEST-7-NTIDEN                                    RESP  56
      DO 100 I=1,NCSEST                                                RESP  57
      IC=INDXCS(1,I)                                                   RESP  58
      IF(IC.LT.1) GO TO 100                                            RESP  59
      II=II+1                                                          RESP  60
      ICC=3-IC                                                         RESP  61
      N=INDXCS(2,I)                                                    RESP  62
      M=INDXCS(3,I)                                                    RESP  63
      M1=M+1                                                           RESP  64
      C3=AORN(N)*GPSIG(I)                                              RESP  65
      C4=C3*CSML(M1,IC)                                                RESP  66
      FCP=-C4*P(M1,N)*DFLOAT(N+1)/R                                    RESP  67
      FCL=C3*CSML(M1,ICC)*P(M1,N)*DFLOAT(M)                            RESP  68
      IF(IC.EQ.1) FCL=-FCL                                             RESP  69
      FCP=C4*(P(M1+1,N)-TPSIM(M1)*P(M1,N))                             RESP  70
      C3=FCR/R                                                         RESP  71
      C4=C3-C1*FCP                                                     RESP  72
      C5=FCL/XYSQ                                                      RESP  73
      GRPAR(1,II)=X*C4-Y*C5                                            RESP  74
      GRPAR(2,II)=Y*C4+X*C5                                            RESP  75
      GRPAR(3,II)=Z*C3+FCP*C2                                          RESP  76
  100 CONTINUE                                                         RESP  77
      RETURN                                                           RESP  78
      END                                                              RESP  79
```

NAME	RFTMCD
PURPOSE	TO CHECK AN 80 CHARACTER ALPHANUMERIC STRING TO DETERMINE IF THE FIRST 6 CHARACTERS ARE NUMERALS AND THE REMAINING 74 CHARACTERS ARE BLANKS
CALLING SEQUENCE	X=RFTMCD(A)

SYMBOL	TYPE	DESCRIPTION
A	I*2	INPUT - CHARACTER STRING
RFTMCD	L	OUTPUT - .TRUE. WHEN FIRST 6 CHARACTERS ARE NUMERALS AND REMAINING 74 ARE BLANKS

SUBROUTINES USED	NONE
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      LOGICAL FUNCTION RFTMCD(A)                                        RFTM  27
      INTEGER*2 A(1),NUMBRS(10),BL                                      RFTM  28
      DATA NUMBRS/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/,3L/1H /      RFTM  29
      RFTMCD=.FALSE.                                                    RFTM  30
      DO 20 I=1,6                                                       RFTM  31
      DO 10 J=1,10                                                      RFTM  32
      IF(A(I).EQ.NUMBRS(J)) GO TO 20                                    RFTM  33
   10 CONTINUE                                                          RFTM  34
      RETURN                                                            RFTM  35
   20 CONTINUE                                                          RFTM  36
      DO 30 I=7,80                                                      RFTM  37
      IF(A(I).NE.BL) RETURN                                             RFTM  38
   30 CONTINUE                                                          RFTM  39
      RFTMCD=.TRUE.                                                     RFTM  40
      RETURN                                                            RFTM  41
      END                                                               RFTM  42
```

NAME — RMSCMP

PURPOSE — TO COMPUTE RMS, RND, AND MEAN FROM SUMMED INFORMATION

CALLING SEQUENCE — CALL RMSCMP(NSUM,ASUM,NTYPE)

SYMBOL	TYPE	DESCRIPTION
NSUM (3,1)	I	INPUT - MEASUREMENT TYPE, NUMBER OF RESIDUALS, AND NUMBER OF WEIGHTED RESIDUAL RATIOS FOR THIS PARTICULAR STATION, SATELLITE, AND MEASUREMENT TYPE
ASUM (8,1)	R	INPUT - SUMMING ARRAYS FOR THIS PARTICULAR STATION SATELLITE & MEASUREMENT TYPE
NTYPE	I	INPUT - MAXIMUM NUMBER OF POSSIBLE TYPES FOR THIS STATION & SATELLITE (MAX. = 4 FOR TYPES 1-14 = 2 FOR TYPES 27-30)

SUBROUTINES USED — NONE

COMMON BLOCKS — CSTINF

INPUT FILES — NONE

OUTPUT FILES — NONE

REFERENCES — 'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE RMSCMP(NSUM,ASUM,NTYPE)                                RMSC  37
      DIMENSION NSUM(3,1),ASUM(8,1)                                     RMSC  38
      COMMON/CSTINF/MEASNO(4),NORS(4),RDMEAN(4),RMSO(4),RND(4),         RMSC  39
     .   MEASWT(4),WTMEAN(4),RMSWTO(4),WTRND(4),TYPRMS(30),NOTYPE(2,30),RMSC  40
     .   BSUM(6,12),RMSALL(30),NOALL(30),NOWTOB,LBASE                  RMSC  41
      DO 300 I=1,NTYPE                                                  RMSC  42
      J=NSUM(3,I)                                                       RMSC  43
      IF(J.NE.0) GO TO 100                                              RMSC  44
      NORS(I)=0                                                         RMSC  45
      RETURN                                                            RMSC  46
  100 NORS(I)=NSUM(1,I)                                                 RMSC  47
      MEASWT(I)=NSUM(2,I)                                               RMSC  48
      MEASNO(I)=J                                                       RMSC  49
      TYPRMS(J)=TYPRMS(J)+ASUM(4,I)                                     RMSC  50
      NOTYPE(2,J)=NOTYPE(2,J)+MEASWT(I)                                 RMSC  51
      XN=NORS(I)                                                        RMSC  52
      RDMEAN(I)=ASUM(1,I)/XN                                            RMSC  53
      RMSO(I)=0.                                                        RMSC  54
      IF(NORS(I).LT.10) GO TO 200                                       RMSC  55
```

```
      RMSO(I)=SQRT(ASUM(3,I)/(XN-1.))                                   RMSC  56
      RND(I)=(ASUM(3,I)-ASUM(1,I)**2/XN)/ASUM(5,I)                      RMSC  57
      RND(I)=(2.*RND(I)-1.)/SQRT((XN-2.)/(XN**2-1.))                    RMSC  58
  200 IF(MEASWT(I).EQ.0) GO TO 300                                      RMSC  59
      XN=MEASWT(I)                                                      RMSC  60
      WTMEAN(I)=ASUM(2,I)/XN                                            RMSC  61
      IF(MEASWT(I).LT.10) GO TO 300                                     RMSC  62
      RMSWTO(I)=SQRT(ASUM(4,I)/(XN-1.))                                 RMSC  63
      WTRND(I)=(ASUM(4,I)-ASUM(2,I)**2/XN)/ASUM(6,I)                    RMSC  64
      WTRND(I)=(2.*WTRND(I)-1.)/SQRT((XN-2.)/(XN**2-1.))                RMSC  65
  300 CONTINUE                                                          RMSC  66
      RETURN                                                            RMSC  67
      END                                                               RMSC  68
```

NAME	ROTMAT
PURPOSE	TO GENERATE A ROTATION MATRIX FROM AN ANGLE AND AXIS OF POTATION
CALLING SEQUENCE	CALL ROTMAT(THETA,I1,X)

SYMBOL	TYPE	DESCRIPTION
THETA	DP	INPUT - THE ROTATION ANGLE
I1	I	INPUT - THE ROTATION AXIS
X (3,3)	DP	OUTPUT - THE ROTATION MATRIX

SUBROUTINES USED	NONE
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE ROTMAT(THETA,I1,X)                                    ROTM  28
      REAL*8 THETA,X(3,3)                                              ROTM  29
      I2=MOD(I1,3)+1                                                   ROTM  30
      I3=MOD(I2,3)+1                                                   ROTM  31
      X(I1,I1)=1.D0                                                    ROTM  32
      X(I1,I2)=0.D0                                                    ROTM  33
      X(I1,I3)=0.D0                                                    ROTM  34
      X(I2,I1)=0.D0                                                    ROTM  35
      X(I3,I1)=0.D0                                                    ROTM  36
      X(I2,I2)=DCOS(THETA)                                             ROTM  37
      X(I3,I3)=X(I2,I2)                                                ROTM  38
      X(I2,I3)=DSIN(THETA)                                             ROTM  39
      X(I3,I2)=-X(I2,I3)                                               ROTM  40
      RETURN                                                           ROTM  41
      END                                                              ROTM  42
```

NAME — SATCLC

PURPOSE — TO APPLY SATELLITE CLOCK CORRECTIONS TO GEOS-I OPTICAL DATA

CALLING SEQUENCE — SATCLC(FIRST,DAY)

SYMBOL	TYPE	DESCRIPTION
FIRST	L	INPUT & OUTPUT - INITIALIZATION SWITCH
DAY	DP	INPUT - UNCORRECTED MEASUREMENT TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR FOR THE ARC
SATCLC	R	OUTPUT - CLOCK CORRECTION IN SECONDS

SUBROUTINES USED — YMDAY

COMMON BLOCKS — NONE

INPUT FILES — NONE

OUTPUT FILES — NONE

```
      FUNCTION SATCLC(FIRST,DAY)                                          SATC  28
      DIMENSION DAYNO(126),DAYNO2( 37),CLC(126),CLC2( 37)                 SATC  29
      EQUIVALENCE (DAYNO(127),DAYNO2(1)),(CLC(127),CLC2(1))               SATC  30
      REAL*8 LAUNCH,CDAY,ODAY,DAY,YMDAY                                   SATC  31
      LOGICAL FIRST                                                       SATC  32
      DATA NCLOCK/163/                                                    SATC  33
      DATA DAYNO/                                                         SATC  34
     *      9.000,   9.721,  10.641,  13.562,  14.562,  15.304,  15.567,SATC  35
     *     16.571,  17.579,  17.579,  19.312,  19.575,  20.579,  20.579,SATC  36
     *     21.583,  22.500,  23.587,  24.508,  25.513,  26.513,  27.429,SATC  37
     *     29.000,  30.442,  31.442,  32.446,  33.450,  34.367,  47.409,SATC  38
     *     48.054,  49.054,  50.325,  51.329,  52.152,  52.241,  54.895,SATC  39
     *     55.895,  58.900,  60.095,  60.829,  60.829,  64.835,  65.845,SATC  40
     *     66.845,  67.845,  67.845,  68.212,  68.762,  70.771,  70.771,SATC  41
     *     71.771,  71.771,  73.779,  73.779,  74.050,  74.696,  74.696,SATC  42
     *     78.704,  78.704,  79.712,  79.988,  80.713,  85.642,  85.642,SATC  43
     *     85.837,  86.650,  89.033,  89.654,  93.575,  93.575,  94.571,SATC  44
     *     94.571,  97.071,  97.591, 100.600, 101.520, 101.520, 102.516,SATC  45
     *    102.516, 103.346, 103.521, 105.441, 105.795, 106.446, 106.800,SATC  46
     *    108.904, 108.804, 109.371, 109.720, 110.725, 110.725, 111.737,SATC  47
     *    111.921, 112.733, 113.733, 113.733, 114.387, 114.746, 114.929,SATC  48
     *    115.737, 116.395, 116.745, 117.658, 117.658, 118.291, 118.658,SATC  49
     *    119.666, 119.666, 122.075, 122.654, 135.191, 135.538, 136.645,SATC  50
     *    136.645, 137.455, 137.454, 138.104, 138.104, 138.545, 139.545,SATC  51
     *    164.104, 164.929, 170.937, 170.937, 171.962, 171.962, 177.792/SATC  52
      DATA DAYNO2/                                                        SATC  53
     *    177.792, 191.108, 191.562, 261.092, 261.092, 262.092, 268.462,SATC  54
     *    269.020, 282.896, 282.896, 283.891, 283.891, 284.900, 284.900),SATC  55
```

```
     *    289.263,  289.883,  316.916,  317.916,  324.854,  324.854,  326.858,SATC  56
     *    326.858,  358.550,  358.550,  373.396,  373.396,  375.137,  375.229,SATC  57
     *    377.054,  378.050,  379.052,  379.052,  382.983,  382.983,  382.983,SATC  58
     *    383.983,  388.000/                                                  SATC  59
      DATA CLC/                                                              SATC  60
     *      -7.20,    -8.20,    -0.35,    -0.03,     2.80,     2.70,     1.10,SATC  61
     *       1.10,     1.10,     0.20,     0.20,     1.00,     1.00,     1.30,SATC  62
     *       1.10,     1.10,     1.00,     2.10,     2.20,     2.28,     0.10,SATC  63
     *      -0.10,    -0.90,    -0.90,     0.10,     0.30,    -0.20,     0.20,SATC  64
     *      -0.80,    -0.85,    -0.70,    -0.70,    -0.30,    -0.30,    -1.10,SATC  65
     *       0.10,     0.10,    -0.10,    -0.60,    -0.10,     0.50,     0.80,SATC  66
     *       0.10,     0.40,     0.10,     0.10,     0.10,     0.50,    -0.10,SATC  67
     *       0.30,    -0.30,    -0.30,    -0.10,    -2.10,    -2.00,     0.10,SATC  68
     *      -0.30,     0.25,     0.20,     0.20,    -0.30,    -0.60,     0.20,SATC  69
     *      -0.20,     0.20,     0.10,    -0.10,    -0.10,    -2.50,    -2.50,SATC  70
     *      -0.20,    -0.30,    -0.20,     0.30,    -2.60,    -0.50,     3.60,SATC  71
     *       0.20,     0.20,    -0.20,    -2.00,     0.20,     0.20,    -0.50,SATC  72
     *      -0.40,     0.20,     0.20,     0.30,     0.90,     0.40,     0.40,SATC  73
     *       0.10,    -0.60,     0.30,    -0.40,    -0.20,    -0.20,    -0.20,SATC  74
     *      -0.50,    -0.45,     0.10,     0.40,    -0.10,     0.10,     0.20,SATC  75
     *       0.60,    -0.40,     0.10,    -0.20,    -0.10,    -0.30,    -0.30,SATC  76
     *       1.70,     1.70,    -0.10,     0.10,     1.70,     1.70,     0.30,SATC  77
     *       0.10,     0.10,     0.40,    -0.50,    -0.50,    -0.40,    -0.30/SATC  78
      DATA CLC2/                                                             SATC  79
     *       0.0 ,    -0.30,    -0.10,    -0.30,     5.20,     0.20,     0.10,SATC  80
     *       0.10,    -0.20,    -5.00,    -4.90,     0.50,     0.50,    -0.10,SATC  81
     *      -0.10,    -0.10,    -0.40,    -0.20,    -0.40,    -0.20,    -0.40,SATC  82
     *      +0.20,    -0.45,    +0.19,    -0.32,    -0.01,    -0.03,    +0.05,SATC  83
     *      +0.07,    -0.50,    -0.50,    +0.23,    +0.40,    +0.22,    +0.30,SATC  84
     *      +0.24,    +0.50/                                                  SATC  85
C DEFINE REFERENCE TIME                                                      SATC  86
      IF(FIRST) LAUNCH=YMDAY(651108.0,0.D0)                                  SATC  87
      FIRST=.FALSE.                                                          SATC  88
      ODAY=DAY+8.64D4                                                        SATC  89
      CDAY=IDINT(ODAY+2.D0)/4*4                                              SATC  90
C SET TIME TO MULTIPLE OF 4 SECONDS                                          SATC  91
      SATCLC=CDAY-CDAY+.50-3                                                 SATC  92
      REFTIM=DAY-LAUNCH                                                      SATC  93
      DO 10 I=2,NCLOCK                                                       SATC  94
      IF(REFTIM.GT.DAYNO(I)) GO TO 10                                        SATC  95
C INTERPOLATE                                                                SATC  96
20    RATE=(CLC(I)-CLC(I-1))/(DAYNO(I)-DAYNO(I-1))                           SATC  97
      SATCLC=SATCLC+(RATE*(REFTIM-DAYNO(I-1))+CLC(I-1))/1.E3                 SATC  98
      RETURN                                                                 SATC  99
10    CONTINUE                                                               SATC 100
      RETURN                                                                 SATC 101
      END                                                                    SATC 102
```

NAME	SATCL2
PURPOSE	TO APPLY SATELLITE CLOCK CORRECTIONS TO GEOS-II OPTICAL DATA
CALLING SEQUENCE	X=SATCL2(FIRST,DAY)

SYMBOL	TYPE	DESCRIPTION
FIRST	L	INPUT & OUTPUT - INITIALIZATION SWITCH
DAY	D2	INPUT - UNCORRECTED MEASUREMENT TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR FOR THE ARC
SATCL2	F	OUTPUT - CLOCK CORRECTION IN SECONDS

SUBROUTINES USED	SATC21 SATC22 YMDAY
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      FUNCTION SATCL2(FIRST,DAY)                                          SATC  29
      DIMENSION DAYNO(500),DAYNO1(116),DAYNO2(51),DAYNO3(114),            SATC  29
     .   DAYNO4(114),DAYNO5(114),CLC(500),CLC1(116),CLC2(51),CLC3(114),   SATC  30
     .   CLC4(114),CLC5(114),DAYS(3)                                      SATC  31
      EQUIVALENCE (DAYNO1(1),DAYNO(1)), (DAYNO2(1),DAYNO(116)),           SATC  32
     .   (DAYNO3(1),DAYNO(167)), (DAYNO4(1),DAYNO(281)),                  SATC  33
     .   (DAYNO5(1),DAYNO(395)), (CLC1(1),CLC(1)),                        SATC  34
     .   (CLC2(1),CLC(116)), (CLC3(1),CLC(167)), (CLC4(1),CLC(281)),      SATC  35
     .   (CLC5(1),CLC(395))                                               SATC  36
      DATA DAYS/232.2979,457.4813,799.9319/                               SATC  37
      DATA DAYNO1/                                                        SATC  38
     .     51.0264,  52.0396,  52.7632,  57.0529,  54.0647,  55.0798,     SATC  39
     .     55.5479,  55.6514,  56.1146,  56.6511,  57.0335,  57.6750,     SATC  40
     .     58.2437,  58.6910,  59.0589,  60.0750,  61.0833,  63.7278,     SATC  41
     .     63.8197,  64.8239,  64.8236,  65.8426,  66.8472,  66.7258,     SATC  42
     .     67.2868,  67.8583,  68.1007,  68.8324,  68.8351,  69.0375,     SATC  43
     .     70.0625,  71.1035,  72.0792,  72.0625,  72.8451,  73.0160,     SATC  44
     .     73.0903,  73.8583,  74.0265,  74.1042,  74.8715,  75.0417,     SATC  45
     .     75.1174,  75.8847,  76.0549,  76.1312,  76.8215,  77.0574,     SATC  46
     .     77.8347,  77.9118,  78.0819,  78.8496,  78.9259,  79.0903,     SATC  47
     .     79.8618,  80.1076,  81.8986,  80.8750,  81.8118,  81.8882,     SATC  48
     .   82.1563,  82.8750,  82.9771,  83.0715,  84.8513,  84.8611,       SATC  49
     .     85.0379,  85.8496,  85.8993,  86.0722,  86.1111,  86.8021,     SATC  50
     .     87.8795,  87.8153,  87.8817,  88.0514,  88.8235,  88.7048,     SATC  51
     .     89.8417,  89.7187,  91.0833,  90.1868,  90.8588,  91.1014,     SATC  52
     .     91.8830,  91.8587,  92.1146,  92.8125,  92.8819,  92.8757,     SATC  53
     .     93.0521,  93.1278,  93.8837,  94.1417,  94.8819,  94.7083,     SATC  54
     .     95.1785,  95.8472,  95.8496,  95.7222,  96.0833,  96.8583,     SATC  55
```

```
     96.7373,  97.5965,  97.6722,  97.7486,  98.1180,  98.6111,   SATC  56
     98.6647,  99.0562,  99.1312,  99.5222,  99.6986, 100.0687/   SATC  57
 DATA DAYN02/                                                     SATC  58
     100.1451, 100.6354, 101.2919, 101.6486, 101.7250, 102.1951,  SATC  59
     102.5618, 103.6750, 103.7581, 104.1215, 104.6172, 104.6937,  SATC  60
     105.1347, 105.6257, 105.7021, 106.1486, 106.6339, 106.7153,  SATC  61
     107.0354, 107.1419, 107.6576, 108.6694, 108.6697, 109.6940,  SATC  62
     110.6965, 110.6972, 111.7104, 112.7336, 113.6611, 114.6679,  SATC  63
    114.6681,115.6742,116.6743,116.7501,116.1285,116.6201,114.6958,SATC  64
     117.1417, 117.6333, 117.7090, 118.1549, 118.6465, 118.7222,  SATC  65
     119.6646, 119.7361, 120.1055, 120.6715, 120.7493, 121.1197,  SATC  66
     121.6561, 121.7625/                                          SATC  67
 DATA DAYN03/                                                     SATC  68
     122.1319, 122.6893, 122.7764, 123.1451, 123.6368, 123.7125,  SATC  69
     124.1583, 124.6590, 124.7326, 125.1722, 125.6632, 125.7389,  SATC  70
     126.1090, 126.6764, 126.7528, 127.1322, 127.6836, 127.7660,  SATC  71
     128.1354, 128.7028, 129.7111, 129.7792, 130.1446, 130.6403,  SATC  72
     129.7160, 130.0868, 130.1618, 130.7292, 131.0993, 131.1750,  SATC  73
     131.6567, 132.0502, 132.6799, 132.7562, 133.1267, 133.6931,  SATC  74
     133.7594, 134.1289, 134.7063, 134.7826, 135.1521, 135.6437,  SATC  75
     135.7194, 135.7250, 136.1660, 136.6569, 136.7326, 137.1785,  SATC  76
     137.6701, 137.7465, 138.1140, 138.1924, 138.6843, 139.1302,  SATC  77
     139.6965, 139.7729, 140.1433, 140.7097, 140.7860, 141.1555,  SATC  78
     141.6479, 141.7229, 142.1687, 142.6611, 142.7368, 143.1910,  SATC  79
     143.6736, 143.7500, 144.1194, 144.6968, 144.7632, 145.1326,  SATC  80
     145.7000, 146.1451, 146.7139, 146.8310, 147.1533, 147.7271,  SATC  81
     147.8042, 148.1722, 148.6646, 148.7403, 149.7465, 149.1854,  SATC  82
     149.6778, 149.7535, 149.7613, 150.1386, 150.6910, 150.7667,  SATC  83
     151.1761, 151.7042, 151.7815, 152.1436, 152.7180, 152.7937,  SATC  84
     152.1518, 153.7305, 154.7486, 155.1989, 155.6812, 155.7569,  SATC  85
     156.2021, 156.6944, 156.7708, 156.7797, 157.1336, 157.7095,  SATC  86
     157.7540, 158.1521, 158.7278, 158.7972, 159.1553, 159.7340/  SATC  87
 DATA DAYN04/                                                     SATC  88
     159.8111, 160.1785, 160.6723, 160.7472, 161.7562, 161.1024,  SATC  89
     161.6956, 161.7611, 162.2056, 162.6986, 162.7743, 163.1431,  SATC  90
     163.2187, 163.7118, 164.1565, 164.2326, 164.7250, 165.1687,  SATC  91
     165.7382, 165.8146, 166.1974, 166.1913, 166.7514, 167.1208,  SATC  92
     167.1958, 167.7646, 168.2090, 168.7779, 169.2222, 169.7157,  SATC  93
     169.7917, 170.1597, 170.2301, 170.7265, 170.7382, 171.1722,  SATC  94
     171.7417, 171.8181, 172.1814, 172.7569, 173.8312, 177.1903,  SATC  95
     173.6931, 173.7681, 174.2125, 174.7903, 175.1530, 175.2257,  SATC  96
     175.7187, 176.1632, 176.2396, 176.7319, 177.1764, 177.2528,  SATC  97
     177.7451, 178.1889, 178.7590, 178.8354, 179.2328, 179.7722,  SATC  98
     180.1410, 180.2160, 180.7854, 181.2292, 181.7239, 181.7986,  SATC  99
     182.1667, 182.2430, 182.7361, 182.7472, 183.1730, 183.2562,  SATC 100
     183.7493, 183.8257, 184.1931, 184.2701, 184.7625, 184.8390,  SATC 101
     185.1319, 185.2062, 185.7767, 186.7029, 187.2396, 187.3029,  SATC 102
     185.1701, 186.2465, 180.7306, 189.7437, 189.7583, 190.1965,  SATC 103
     190.2736, 190.7569, 191.7792, 192.2229, 192.7370, 192.7970,  SATC 104
     193.1811, 193.2701, 193.7340, 193.8369, 194.1736, 194.2500,  SATC 105
     194.7437/ 195.7565, 195.8326, 196.2100, 196.2771, 196.7701,  SATC 106
     197.7433, 198.1521, 198.2264, 198.7965, 199.2336, 199.7340/  SATC 107
 DATA DAYN05/                                                     SATC 108
     199.8397, 200.1778, 200.7472, 201.8299, 201.1913, 201.7624,  SATC 109
     202.7736, 202.8510, 203.1431, 203.2167, 203.2944, 203.7858,  SATC 110
     204.2432, 204.1555, 204.2269, 204.8200, 204.8771, 205.1682,  SATC 111
```

```
     205.2430,  205.7375,  205.9132,  206.1812,  206.2549,  206.8271,    SATC *12
     207.1944,  207.2701,  207.7539,  207.8403,  208.1340,  208.2050,    SATC *13
     208.2340,  208.7771,  208.8535,  209.2291,  209.2979,  209.7003,    SATC *14
     209.9574,  210.1597,  210.2333,  210.2114,  210.7295,  210.9042,    SATC *15
     211.1732,  211.2472,  211.7417,  211.8174,  212.1847,  212.2404,    SATC *16
     212.7949,  212.8316,  213.1970,  213.2736,  213.7541,  213.8437,    SATC *17
     214.0472,  214.1282,  214.1471,  214.2111,  214.2875,  214.7812,    SATC *18
     214.8576,  215.1507,  215.2243,  215.2347,  215.3014,  215.7946,    SATC *19
     215.7986,  215.8709,  216.1622,  216.2375,  216.3153,  216.8076,    SATC *20
     217.1757,  217.2507,  217.7441,  217.8208,  218.1849,  218.2639,    SATC *21
     218.7593,  220.7989,  221.8278,  221.8349,  221.7979,  222.2410,    SATC *22
     222.8111,  222.8882,  223.1760,  223.2542,  223.8213,  223.1934,    SATC *23
     224.2473,  224.8375,  225.2812,  225.7750,  225.8514,  226.2187,    SATC *24
     226.7882,  226.8546,  227.2243,  227.3093,  227.8014,  228.2644,    SATC *25
     228.8146,  228.8917,  229.1846,  229.8278,  230.1965,  230.2715,    SATC *26
     230.8417,  231.2090,  231.2847,  231.8540,  232.2222,  232.2979/    SATC *27
 DATA CLC1/                                                              SATC *28
      0.390,     0.326,     0.246,     0.200,     0.110,    -0.105,      SATC *29
     -0.130,    -0.050,    -0.250,    -0.336,    -0.401,    -0.477,      SATC *30
     -0.542,    -0.689,    -0.940,    -0.370,     0.450,     0.450,      SATC *31
      0.733,     0.670,    -0.150,    -0.164,    -0.150,    -0.150,      SATC *32
      0.466,     0.500,     0.576,     0.643,     0.576,     0.127,      SATC *33
     -0.090,     0.130,     0.184,     0.198,     0.212,     0.254,      SATC *34
      0.279,     0.211,     0.260,     0.237,    -0.265,    -0.762,      SATC *35
     -0.770,    -0.197,    -0.038,    -0.060,    -0.060,     0.021,      SATC *36
      0.037,     0.031,     0.060,     0.063,     0.138,     0.154,      SATC *37
      0.131,     0.164,     0.151,     0.152,     0.141,     0.164,      SATC *38
      0.113,     0.155,     0.138,     0.093,     0.093,     7.244,      7.251,SATC *39
      7.202,     7.277,     7.276,     0.072,     0.049,     0.107,      SATC *40
      0.132,     0.131,     0.156,     0.150,     0.125,     0.131,      SATC *41
      0.167,     0.160,     0.194,     0.174,     0.178,     0.175,      SATC *42
      0.230,     0.256,     0.251,     0.309,     0.344,     0.336,      SATC *43
      0.345,     0.355,     0.437,     0.486,     0.412,     0.439,      SATC *44
      0.462,     0.515,    -0.360,    -0.325,    -0.314,    -0.244,      SATC *45
     -0.246,    -0.164,    -0.146,    -0.135,    -0.105,    -0.040,      SATC *46
     -0.073,     0.017,     0.030,     0.025,    -0.039,     0.154/      SATC *47
 DATA CLC2/                                                              SATC *48
      0.167,     0.146,     0.220,     0.197,     0.225,     0.250,      SATC *49
     -0.320,    -0.349,     0.399,     0.465,     0.511,     0.195,      SATC *50
      0.255,     0.296,     0.331,     0.370,     0.452,     0.450,      SATC 151
      0.471,     0.492,     0.020,     0.021,     0.022,     0.245,      SATC 152
     -0.096,    -0.066,     0.729,     0.089,    -0.100,    -0.100,      SATC *53
     15.253,    15.253,    -0.257,    -0.268,    -0.241,    -0.242,      SATC *54
     -0.217,    -0.198,    -0.143,    -0.153,    -0.171,    -0.141,      SATC *55
     -0.130,    -0.161,    -0.140,    -0.147,    -0.045,    -0.076,      SATC *56
     -0.057,    -0.045,    -0.039/                                       SATC *57
 DATA CLC3/                                                              SATC *58
     -0.052,    -0.093,     0.050,    -0.025,    -0.014,    -0.054,      SATC *59
     -0.065,    -0.073,    -0.079,    -0.135,    -0.069,    -0.086,      SATC *60
     -0.143,    -0.143,    -0.148,    -0.023,    -0.037,    -0.034,      SATC *61
     -0.311,     0.268,    -0.077,    -0.041,    -0.019,    -0.011,      SATC *62
      0.0  ,     0.060,     0.078,     0.070,     0.117,     0.122,      SATC *63
      0.175,     0.150,     0.174,     0.173,     0.214,     0.272,      SATC *64
      0.248,     0.303,     0.314,     0.207,     0.337,     0.309,      SATC *65
      0.157,     0.161,     0.214,     0.151,     0.137,     0.134,      SATC *66
      0.126,     0.105,     0.097,     0.123,     0.090,     0.103,      SATC *67
```

```
     *      0.071,    0.098,   -0.059,   -0.078,   -0.044,   -0.140,   SATC 168
     *     -0.188,   -0.177,   -0.140,   -0.164,   -0.158,   -0.151,   SATC 169
     *     -0.190,   -0.201,   -0.214,   -0.239,   -0.242,   -0.314,   SATC 170
     *     -0.330,   -0.362,   -0.342,   -0.347,   -0.351,   -0.349,   SATC 171
     *     -0.354,   -0.418,   -0.370,   -0.361,    0.476,    0.420,   SATC 172
     *      0.432,    0.449,   -0.076,   -0.050,   -0.035,   -0.043,   SATC 173
     *     -0.026,   -0.029,   -0.012,   -0.051,   -0.016,   -0.019,   SATC 174
     *      0.031,    0.016,    0.134,   -0.137,   -0.219,   -0.260,   SATC 175
     *     -0.360,   -0.415,   -0.448,   -0.304,   -0.029,   -0.140,   SATC 176
     *     -0.154,   -0.115,   -0.264,   -0.249,   -0.240,   -0.330/   SATC 177
     DATA CL24/                                                        SATC 178
     *     -0.258,   -0.295,   -0.325,   -0.318,   -0.033,   -0.076,   SATC 179
     *     -0.059,   -0.050,   -0.082,   -0.103,   -0.100,   -0.164,   SATC 180
     *     -0.112,   -0.128,   -0.123,   -0.140,   -0.144,   -0.172,   SATC 181
     *     -0.125,   -0.113,   -0.260,   -0.237,   -0.180,   -0.198,   SATC 182
     *     -0.212,   -0.175,   -0.207,   -0.213,   -0.195,   -0.208,   SATC 183
     *     -0.213,   -0.237,   -0.176,   -0.198,    0.245,    0.229,   SATC 184
     *      0.212,    0.198,    0.192,    0.156,    0.213,    0.129,   SATC 185
     *     -0.091,    0.191,    0.117,    0.142,    0.374,    0.314,   SATC 186
     *      0.337,    0.313,    0.299,    0.304,    0.307,    0.266,   SATC 187
     *      0.744,    0.709,    0.764,    0.766,    0.236,    0.053,   SATC 188
     *     -0.285,   -0.116,   -0.136,   -0.054,   -0.434,   -0.420,   SATC 189
     *     -0.450,   -0.463,   -0.506,    0.232,    0.150,    0.124,   SATC 190
     *      0.017,    0.012,   -0.105,   -0.147,   -0.264,   -0.307,   SATC 191
     *     -0.355,   -0.363,   -0.384,    0.451,    0.314,    0.026,   SATC 192
     *      0.118,    0.113,    0.448,    0.485,   -0.293,   -0.246,   SATC 193
     *     -0.344,   -0.186,    0.456,    0.505,    0.358,   -0.434,   SATC 194
     *     -0.476,   -0.415,   -0.372,   -0.368,   -0.332,   -0.329,   SATC 195
     *     -0.281,   -0.313,   -0.332,   -0.274,   -0.249,   -0.212,   SATC 196
     *     -0.108,   -0.130,   -0.100,   -0.408,   -0.473,   -0.447/   SATC 197
     DATA CL36/                                                        SATC 198
     *     -0.445,   -0.421,   -0.402,   -0.405,   -0.419,   -0.327,   SATC 199
     *     -0.150,   -0.155,   -0.188,   -0.150,   -0.132,   -0.140,   SATC 200
     *     -0.175,   -0.200,   -0.174,   -0.141,   -0.134,   -0.148,   SATC 201
     *     -0.183,   -0.151,   -0.158,   -0.175,   -0.169,   -0.109,   SATC 202
     *     -0.165,   -0.148,   -0.137,   -0.067,   -0.075,   -0.047,   SATC 203
     *     -0.080,   -0.098,   -0.034,   -0.062,   -0.045,   -0.095,   SATC 204
     *     -0.031,    0.018,   -0.007,   -0.009,   -0.047,   -0.012,   SATC 205
     *     -0.030,    0.023,   -0.004,    0.108,    0.052,    0.061,   SATC 206
     *      0.065,    0.059,    0.056,    0.099,    0.085,    0.131,   SATC 207
     *      0.014,    0.019,   -0.172,   -0.195,   -0.238,   -0.200,   SATC 208
     *     -0.146,   -0.140,   -0.137,   -0.476,   -0.472,   -0.491,   SATC 209
     *      0.052,    0.078,    0.041,    0.051,    0.057,    0.078,   SATC 210
     *      0.131,    0.110,    0.109,    0.158,    0.180,    0.163,   SATC 211
     *      0.183,   -0.420,   -0.396,   -0.382,   -0.417,   -0.353,   SATC 212
     *     -0.354,   -0.343,   -0.313,   -0.291,   -0.285,   -0.251,   SATC 213
     *     -0.251,   -0.237,   -0.196,   -0.183,   -0.180,   -0.115,   SATC 214
     *     -0.108,   -0.076,   -0.064,   -0.045,   -0.069,   -0.044,   SATC 215
     *     -0.038,   -0.002,   -0.052,    0.021,    0.041,    0.040,   SATC 216
     *      0.065,    0.099,    0.096,    0.119,    0.134,    0.125/   SATC 217
     REAL*8 LAUNCH,CDAY,JDAY,DAY,YMDAY                                 SATC 218
     LOGICAL FIRST                                                     SATC 219
     DATA NCL/24/                                                      SATC 220
C DEFINE REFERENCE TIME                                                SATC 221
     IF(FIRST) LAUNCH=YMDAY(690101,0,0.D0)                             SATC 222
     FIRST=.FALSE.                                                     SATC 223
```

```
      ODAY=DAY*8.6404                                                   SATC 224
      CDAY=(DINT(ODAY+2.D0)/4)*4                                        SATC 225
C SET TIME TO MULTIPLE OF 4 SECONDS                                     SATC 226
      SATCL2=CDAY-ODAY+.5D-3                                            SATC 227
      REFTIM=DAY-LAUNCH                                                 SATC 228
      IF(REFTIM.GT.DAYS(3)) RETURN                                      SATC 229
      N=1                                                               SATC 230
      IF(REFTIM.GE.DAYS(N).AND.REFTIM.LT.DAYS(N+1)) GO TO 330           SATC 231
      N=2                                                               SATC 232
      IF(REFTIM.GE.DAYS(N).AND.REFTIM.LT.DAYS(N+1)) GO TO 430           SATC 233
      DO 10 I=2,NCLOCK                                                  SATC 234
      IF(REFTIM.GT.DAYNO(I)) GO TO 10                                   SATC 235
C INTERPOLATE                                                           SATC 236
      RATE=(CLC(I)-CLC(I-1))/(DAYNO(I)-DAYNO(I-1))                      SATC 237
      SATCL2=SATCL2+(RATE*(REFTIM-DAYNO(I-1))+CLC(I-1))*1.E-3           SATC 238
      RETURN                                                            SATC 239
   10 CONTINUE                                                          SATC 240
      RETURN                                                            SATC 241
  330 SATCL2=SATCL2+SATC21(REFTIM)                                      SATC 242
      RETURN                                                            SATC 243
  430 SATCL2=SATCL2+SATC22(REFTIM)                                      SATC 244
      RETURN                                                            SATC 245
      END                                                               SATC 246
```

NAME — SATC21

PURPOSE — PART 2 OF SATELLITE CLOCK CORRECTION FOR GEOS 2 (SATCL2)

CALLING SEQUENCE — X=SATC21(REFTIM)

SYMBOL	TYPE	DESCRIPTION
REFTIM	R	INPUT - REFERENCE TIME
SATC21	R	OUTPUT - CLOCK CORRECTION IN SECONDS

SUBROUTINES USED — NONE

COMMON BLOCKS — NONE

INPUT FILES — NONE

OUTPUT FILES — NONE

```
      FUNCTION SATC21(REFTIM)                                           SATC 272
      DIMENSION DAYN0(456),CLC(456),DAYN05(114),DAYN07(111),            SATC 273
     *   DAYN09(114),DAYN0B(114),CLC5(114),CLC7(111),CLC9(114),         SATC 274
     *   CLC0(114)                                                      SATC 275
      EQUIVALENCE (DAYS1,DAYN0(1)), (DAYN0(1),DAYN0(2)),                SATC 276
     *   (DAYN07(1),DAYN0(113)), (DAYN0B(1),DAYN0(227)),                SATC 277
     *   (DAYN09(1),DAYN0(340)), (CLS1,CLC(1)), (CLC5(1),CLC(2)),       SATC 278
     *   (CLC7(1),CLC(113)), (CLC9(1),CLC(227)), (CLC0(1),CLC(340))     SATC 279
      DATA DAYS1,CLS1/232.2979,0.125/                                   SATC 280
      DATA DAYN06/                                                      SATC 281
     *    232.7917, 233.2354, 233.6849, 233.8912, 234.2485, 234.8181,   SATC 282
     *    234.8359, 235.1931, 235.2613, 235.4313, 235.8354, 235.9083,   SATC 283
     *    236.2750, 236.8451, 237.2132, 237.2492, 237.8543, 237.8629,   SATC 284
     *    238.2267, 238.3221, 238.7951, 239.2380, 239.7153, 240.6915,   SATC 285
     *    241.1917, 241.2553, 241.8354, 242.2742, 242.2735, 242.8486,   SATC 286
     *243.2164,243.2167,243.2917,243.8617,243.8619,244.2739,244.3055,   SATC 287
     *244.8750,245.3187,245.3119,245.8583,246.2582,246.3326,246.8265,   SATC 288
     *    246.9292, 247.2867, 247.7445, 247.8783, 248.2083, 248.3526,   SATC 289
     *    248.4521, 248.8555, 249.2218, 249.3958, 249.8653, 250.2333,   SATC 290
     *    250.3090, 250.8785, 251.2465, 251.7229, 251.8140, 252.2567,   SATC 291
     *    252.3731, 252.8392, 253.2729, 253.3500, 253.8433, 254.2118,   SATC 292
     *    254.3561, 254.7798, 254.8555, 254.8547, 255.2243, 255.2963,   SATC 293
     *    255.7930, 256.2587, 256.8729, 256.9373, 256.8818, 257.2505,   SATC 294
     *    257.8193, 257.8953, 258.3612, 258.8325, 258.9033, 259.2764,   SATC 295
     *    259.3457, 260.2158, 261.2896, 260.8543, 261.2243, 261.3027,   SATC 296
     *    261.8721, 262.2410, 262.7147, 262.8453, 263.2512, 263.3264,   SATC 297
     *    263.8233, 264.2262, 264.2673, 264.7435, 264.8712, 265.2505,   SATC 298
     *    265.3574, 265.8491, 266.2142, 266.3337, 266.8643, 267.2334,   SATC 299
     *    267.3069, 267.8755, 268.3201, 268.8130, 268.8575, 269.3333/   SATC 300
```

```
      DATA DAYN07/                                                                  SATC 701
     .    269.8319,  270.2708,  270.7472,  270.8389,  271.2839,  271.3611,        SATC 702
     .    271.8521,  272.2971,  272.3750,  272.8660,  273.2361,  273.3103,        SATC 703
     .    273.8792,  274.0485,  274.7835,  274.8927,  275.2617,  275.3374,        SATC 704
     .    275.8565,  276.2749,  276.3576,  276.8423,  277.2874,  277.3645,        SATC 705
     .    277.8565,  278.2005,  278.3784,  278.8687,  279.2402,  279.3137,        SATC 706
     .    279.8826,  280.2867,  280.3527,  280.8240,  281.2958,  281.8096,        SATC 707
     .    281.8458,  281.9325,  281.9839,  282.3124,  282.8540,  282.8457,        SATC 708
     .    282.8221,  283.2915,  283.3543,  283.8757,  284.3040,  284.8721,        SATC 709
     .    284.8492,  285.3171,  285.8853,  285.8424,  286.2561,  286.3311,        SATC 710
     .    286.8992,  287.3442,  287.4359,  287.9124,  288.3574,  288.8491,        SATC 711
     .    288.8255,  289.3713,  289.8623,  289.8787,  290.3041,  290.8755,        SATC 712
     .    290.8526,  291.3313,  291.8697,  292.3747,  292.8240,  292.9019,        SATC 713
     .    293.3479,  293.8394,  293.9158,  294.2850,  294.3611,  294.9200,        SATC 714
     .    295.3760,  295.8663,  296.3118,  296.8792,  296.9553,  297.3250,        SATC 715
     .    297.8923,  297.9692,  298.3341,  298.8303,  298.9055,  299.3513,        SATC 716
     .    299.3630,  299.9187,  300.3665,  300.8962,  300.9336,  301.3020,        SATC 717
     .    301.3784,  301.8694,  302.3152,  302.8823,  302.9826,  303.3284,        SATC 718
     .    303.8958,  303.9936,  304.2573,  304.7416,  304.9349,  305.2805,        SATC 719
     .    305.3547,  305.9221,  305.9297/                                         SATC 720
                                                                                  SATC 721
      DATA DAYN08/                                                                SATC 722
     .    323.8567,  323.9324,  324.1040,  324.3784,  324.9456,  325.3164,        SATC 723
     .    325.3923,  325.9587,  326.3296,  326.4135,  326.8965,  327.3620,        SATC 724
     .    327.4192,  327.9097,  328.3555,  328.3334,  328.9030,  329.3687,        SATC 725
     .    329.8601,  329.9358,  330.3819,  330.8733,  330.9490,  331.3198,        SATC 726
     .    331.3953,  331.9434,  332.4089,  332.8999,  332.9932,  333.3657,        SATC 727
     .    333.3226,  334.0260,  334.0027,  334.3721,  334.9334,  335.4270,        SATC 728
     .    335.4439,  335.9541,  336.3916,  336.8756,  336.9165,  337.4231,        SATC 729
     .    337.8694,  338.0526,  338.3950,  338.4675,  338.9207,  339.4339,        SATC 730
     .    339.8605,  340.4509,  340.8347,  340.8998,  340.9580,  341.4124,        SATC 731
     .    341.4895,  341.9554,  342.4255,  342.4834,  342.9951,  343.3645,        SATC 732
     .    343.4797,  343.9326,  344.0093,  344.4525,  344.9458,  345.0215,        SATC 733
     .    345.4658,  345.9590,  346.0347,  346.4733,  346.9730,  347.0722,        SATC 734
     .    347.4158,  347.9029,  347.9854,  348.3915,  348.9230,  349.0068,        SATC 735
     .    349.3985,  349.8750,  349.9471,  350.3869,  350.9117,  351.0453,        SATC 736
     .    350.3492,  350.8249,  350.9076,  351.3524,  351.9231,  352.1824,        SATC 737
     .    352.3756,  352.8505,  352.9194,  353.3498,  353.9652,  354.3598,        SATC 738
     .    354.4325,  354.9080,  354.9858,  355.4284,  355.9459,  356.3645,        SATC 739
     .    356.4534,  356.9283,  357.0055,  357.4720,  358.0492,  358.4104,        SATC 740
     .    358.4797,  358.9554,  359.0000,  359.4922,  360.0686,  360.4360/        SATC 741
      DATA DAYN09/                                                                SATC 742
     .    360.5151,  361.0054,  361.0825,  361.5269,  362.0193,  362.0957,        SATC 743
     .    362.5624,  363.0325,  363.1014,  363.5700,  364.0457,  364.1139,        SATC 744
     .    364.5589,  365.0270,  365.1027,  365.5669,  366.0738,  366.1402,        SATC 745
     .    366.6125,  367.0864,  367.1640,  367.6459,  368.1470,  368.1165,        SATC 746
     .    368.6797,  369.0491,  369.1178,  369.6029,  370.0670,  370.1304,        SATC 747
     .    370.6561,  371.0762,  371.1434,  371.6203,  372.0132,  372.1566,        SATC 748
     .    372.6339,  373.0264,  373.1068,  373.5703,  374.0326,  374.1082,        SATC 749
     .    374.6132,  375.0532,  375.1214,  375.6063,  376.0647,  376.1239,        SATC 750
     .    376.6195,  377.0042,  377.0758,  377.6312,  378.0196,  378.0832,        SATC 751
     .    378.6374,  379.0027,  379.1049,  379.6749,  380.0477,  380.1855,        SATC 752
     .    380.6316,  381.0569,  381.1243,  381.6000,  382.0742,  382.1373,        SATC 753
     .    382.6137,  383.0374,  383.1057,  383.6269,  384.0779,  384.1630,        SATC 754
     .    384.6113,  385.0339,  385.1771,  385.6471,  386.0134,  386.1502,        SATC 755
     .    386.6343,  387.0603,  387.1242,  387.7034,  388.0715,  388.1417,        SATC 756
     .    388.6166,  389.0874,  389.1541,  389.7205,  390.0216,  390.1673,        SATC 757
```

```
     433.5444,  434.0374,  434.4805,  434.5576,  435.0505,  435.1277,   SATC 357
     435.4937,  436.0627,  436.4436,  436.5069,  437.0776,  437.1451,   SATC 358
     437.5208,  438.0908,  438.4563,  438.5239,  439.1074,  439.1714,   SATC 359
     439.5479,  440.1013,  440.4692,  440.5449,  441.0388,  441.0810/   SATC 360
 DATA CL(5)/                                                            SATC 361
       -0.182,    0.163,    0.246,    0.261,    0.255,    0.276,      SATC 362
        0.293,   -7.546,  -20.321,  -20.727,    0.119,    0.157,      SATC 363
        0.144,    0.209,  -40.382,  -40.430,  -56.356,    0.027,      SATC 364
      -37.220,  -40.344,    0.314,   -2.475,   -7.760,    0.012,      SATC 365
       -0.016,   -0.010,   -0.005,    0.0  ,   -0.030,   -0.028,      SATC 366
   -0.024,  5.226,  6.286,  0.294,  0.324,  0.337,  0.341,  0.345,     SATC 367
        0.387,    0.361,  -31.096,    0.382,    0.407,    0.358,      SATC 368
        0.393,    0.129,    0.148,    0.129,   -1.394,   -1.389,      SATC 369
       -1.442,    0.159,    0.026,    0.192,    0.187,    0.221,      SATC 370
        0.239,    0.196,    0.271,    0.273,    0.275,    0.315,      SATC 371
        0.315,    0.332,    0.352,    0.347,    0.394,    0.393,      SATC 372
        0.409,    0.418,    0.438,    0.268,    0.253,    0.257,      SATC 373
        0.229,    0.218,   -0.119,   -0.106,   -0.178,   -0.190,      SATC 374
       -0.162,   -0.200,   -0.193,   -0.196,   -0.142,   -0.192,      SATC 375
       -0.197,   -0.252,   -0.209,   -0.211,   -0.251,   -0.234,      SATC 376
       -0.261,   -0.257,   -0.267,   -0.364,   -0.390,   -0.390,      SATC 377
       -0.409,    0.384,    0.359,    0.362,    0.291,    0.248,      SATC 378
        0.279,    0.212,    0.115,    0.106,    0.058,    0.083,      SATC 379
        0.119,    0.069,    0.035,    0.026,    0.023,   -0.011/      SATC 380
 DATA CL(6)/                                                            SATC 381
       -0.049,   -0.072,   -0.090,   -0.105,   -0.126,   -0.128,      SATC 382
       -0.129,   -0.149,   -0.137,   -0.157,   -0.161,   -0.157,      SATC 383
       -0.139,   -0.147,   -0.148,   -0.178,   -0.152,   -0.169,      SATC 384
       -0.205,   -0.184,   -0.188,   -0.224,   -0.237,   -0.253,      SATC 385
       -0.276,   -0.302,   -0.276,   -0.351,   -0.327,   -0.310,      SATC 386
       -0.382,    0.229,    0.266,    0.285,    0.224,   -1.233,      SATC 387
       -1.115,   -1.104,   -1.122,    0.259,    0.230,    0.215,      SATC 388
        0.217,    0.272,    0.293,    0.279,    0.259,    0.209,      SATC 389
        0.219,    0.165,    0.199,    0.213,    0.137,    0.135,      SATC 390
        0.153,    0.094,    0.140,    0.145,    0.111,    0.139,      SATC 391
        0.119,    0.088,    0.152,    0.137,    0.093,    0.105,      SATC 392
        0.117,    0.046,    0.065,    0.054,    0.017,    0.010,      SATC 393
       -0.010,   -0.054,   -0.064,   -0.107,    0.025,   -0.012,      SATC 394
       -0.005,   -0.032,   -0.011,   -0.076,   -0.016,   -0.023,      SATC 395
       -0.016,   -0.064,   -0.062,   -0.126,   -0.074,   -0.073,      SATC 396
       -0.123,   -0.111,   -0.163,   -0.160,   -0.157,   -0.125,      SATC 397
       -0.122,   -0.176,   -0.168,   -0.147,   -0.154,   -0.190,      SATC 398
       -0.175,   -1.041,   -1.066,   -1.040,   -1.170,   -1.105,      SATC 399
       -1.132,   -1.168,    0.335/                                    SATC 400
 DATA CL(4)/                                                            SATC 401
        0.162,    0.168,    0.022,    0.153,    0.137,    0.147,      SATC 402
        0.123,    0.175,    0.193,    0.169,    0.222,    0.195,      SATC 403
        0.235,    0.175,    0.232,    0.188,    0.196,    0.213,      SATC 404
        0.252,    0.192,    0.238,    0.227,    0.231,    0.273,      SATC 405
        0.226,    0.209,    0.256,    0.203,    0.305,    0.288,      SATC 406
        0.355,    0.174,    0.190,    0.143,    0.179,    0.040,      SATC 407
        0.098,    0.115,    0.159,    0.206,    0.238,    0.270,      SATC 408
        0.211,    0.147,    0.144,    0.113,    0.175,    0.021,      SATC 409
        0.211,    0.052,    0.210,    0.104,    0.014,    0.049,      SATC 410
        0.091,    0.056,    0.093,    0.090,    0.132,    0.112,      SATC 411
        0.126,    0.133,    0.150,    0.179,    0.136,    0.206,      SATC 412
```

```
     *       0.195,    0.192,    0.200,    0.199,    0.215,    0.212,   SATC 413
     *       0.228,    0.233,    0.249,    0.040,    0.061,    0.059,   SATC 414
     *       0.085,    0.091,    0.106,    0.108,    0.127,    0.152,   SATC 415
     *       0.163,    0.170,    0.181,    0.205,    0.221,    0.224,   SATC 416
     *       0.256,    0.264,    0.235,    0.246,    0.309,    0.253,   SATC 417
     *       0.260,    0.310,    0.112,    0.044,    0.139,    0.159,   SATC 418
     *       0.164,    0.198,    0.190,    0.204,    0.223,    0.241,   SATC 419
     *       0.267,    0.277,    0.285,    0.273,    0.297,    0.291/   SATC 420
      DATA CLC3/                                                        SATC 421
     *       0.297,    0.006,    0.019,    0.027,    0.023,    0.034,   SATC 422
     *       0.043,    0.043,    0.040,    0.040,    0.040,    0.064,   SATC 423
     *       0.078,    0.092,    0.102,    0.091,    0.117,    0.114,   SATC 424
     *       0.119,    0.104,    0.110,    0.108,    0.132,    0.095,   SATC 425
     *       0.089,    0.090,    0.090,    0.075,    0.074,    0.075,   SATC 426
     *       0.059,    0.069,    0.055,    0.052,    0.058,    0.053,   SATC 427
     *       0.060,    0.076,    0.078,    0.071,    0.091,    0.091,   SATC 428
     *       0.090,    0.094,    0.090,    0.081,    0.093,    0.093,   SATC 429
     *       0.088,    0.100,    0.097,    0.101,    0.103,    0.094,   SATC 430
     *       0.076,    0.063,    0.055,    0.087,    0.112,    0.126,   SATC 431
     *       0.165,    0.161,    0.175,    0.173,    0.214,    0.223,   SATC 432
     *       0.222,    0.238,    0.254,    0.263,    0.275,    0.292,   SATC 433
     *       0.287,    0.305,    0.329,    0.332,    0.344,    0.346,   SATC 434
     *       0.148,    0.113,    0.110,    0.105,    0.090,    0.090,   SATC 435
     *       0.077,    0.062,    0.047,    0.045,    0.029,    0.023,   SATC 436
     *       0.017,    0.003,    0.005,    0.0  ,    0.012,    0.005,   SATC 437
     *       0.018,    0.049,    0.081,    0.077,    0.102,    0.141,   SATC 438
     *       0.135,    0.170,    0.200,    0.202,    0.228,    0.250,   SATC 439
     *       0.243,   -0.358,   -0.389,   -0.382,   -0.414,   -0.481/   SATC 440
      DATA NCLOCK/45*/                                                  SATC 441
      DO 10 I=2,NCLOCK                                                  SATC 442
      IF(REFTIM.GT.DAYNO(I)) GO TO 10                                   SATC 443
C INTERPOLATE                                                           SATC 444
      RATE=(CLC(I)-CLC(I-1))/(DAYNO(I)-DAYNO(I-1))                      SATC 445
      SATC21=(RATE*(REFTIM-DAYNO(I-1))+CLC(I-1))*1.E-3                  SATC 446
      RETURN                                                            SATC 447
   10 CONTINUE                                                          SATC 448
      RETURN                                                            SATC 449
      END                                                               SATC 450
```

NAME SATC22

PURPOSE PART 3 OF SATELLITE CLOCK CORRECTION FOR GEOS 2 (SATCL3)

CALLING SEQUENCE X=SATC22(REFTIM)

SYMBOL	TYPE	DESCRIPTION
REFTIM	R	INPUT - REFERENCE TIME
SATC22	R	OUTPUT - CLOCK CORRECTION IN SECONDS

SUBROUTINES USED NONE

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
      FUNCTION SATC22(REFTIM)                                          SATC 475
      DIMENSION DAYN0(459),CLC(459),DAYN10(114),DAYN11(114),           SATC 476
     *   DAYN12(114),DAYN13(114),DAYN14(32),CL10(114),CL11(114),       SATC 477
     *   CL12(114),CL13(114),CL14(32)                                  SATC 478
      EQUIVALENCE (DAYS2,DAYN0(1)), (DAYN10(1),DAYN7(2)),              SATC 479
     *   (DAYN11(1),DAYN0(115)), (DAYN12(1),DAYN0(230)),               SATC 480
     *   (DAYN13(1),DAYN0(344)), (DAYN14(1),DAYN0(459)),               SATC 481
     *   (CLS2,CLC(1)), (CL10(1),CLC(2)), (CL11(1),CLC(115)),          SATC 482
     *   (CL12(1),CLC(230)), (CL13(1),CLC(344)), (CL14(1),CLC(459))    SATC 483
      DATA DAYS2,CLS2/457.4819,-0.481/                                 SATC 484
      DATA DAYN10/                                                     SATC 485
     *    457.4873, 457.5581, 458.0520, 458.4956, 456.5720, 459.1416,  SATC 486
     *    459.5083, 459.5859, 460.0784, 460.4478, 460.5915, 460.5269,  SATC 487
     *    461.0216, 461.4662, 461.5354, 462.0231, 462.4734, 462.5486,  SATC 488
     *    463.1179, 463.4861, 463.5519, 464.1319, 464.4903, 464.5757,  SATC 489
     *    465.0686, 465.5125, 465.5901, 466.0318, 466.4592, 466.5256,  SATC 490
     *    467.1721, 467.4644, 467.5389, 468.1082, 468.4771, 468.5520,  SATC 491
     *    469.0576, 469.4457, 469.5159, 470.1353, 470.5027, 470.5791,  SATC 492
     *    471.0720, 471.1484, 471.5159, 472.0452, 472.5241, 472.6069,  SATC 493
     *    473.0984, 473.4709, 473.5422, 474.1116, 474.4812, 474.5554,  SATC 494
     *    475.1250, 475.4937, 475.5603, 476.1287, 476.5058, 476.5825,  SATC 495
     *    477.0757, 477.1519, 477.5210, 477.5342, 478.0349, 478.5325,  SATC 496
     *    479.0134, 479.1021, 479.4732, 479.5457, 480.1152, 480.4546,  SATC 497
     *    480.5406, 481.1254, 481.4776, 482.1424, 482.5123, 482.5151,  SATC 498
     *    483.0459, 483.3791, 483.4575, 484.0234, 484.0335, 484.1687,  SATC 499
     *    484.3366, 484.4139, 485.1005, 485.4766, 485.5438, 485.6211,  SATC 500
     *    486.1137, 486.4888, 486.5636, 487.1319, 487.5012, 487.5762,  SATC 501
     *    488.1450, 488.5144, 488.5901, 489.1540, 489.5240, 489.6033,  SATC 502
     *    490.1957, 490.1721, 491.5470, 491.1340, 491.5178, 491.5532,  SATC 503
```

```
     492.1221, 492.1992, 492.5667, 493.1353, 493.5046, 493.5798/   SATC F04
 DATA DAYN11/                                                      SATC F05
     494.0728, 494.1514, 494.5935, 495.0859, 495.1638, 495.5313,   SATC F06
     496.1755, 496.5437, 496.6206, 497.1123, 497.1947, 497.5569,   SATC F07
     497.5095, 498.1255, 498.2019, 498.4963, 498.5771, 499.1387,   SATC F08
     499.5098, 499.5940, 500.1519, 500.5215, 500.5972, 501.0894,   SATC F09
     501.1650, 501.5347, 502.1025, 502.5479, 502.6243, 503.1160,   SATC F10
     503.1943, 503.5610, 504.1292, 504.2175, 504.5742, 505.1423,   SATC F11
     505.5125, 505.5874, 506.0708, 506.1555, 506.6016, 507.0930,   SATC F12
     507.1687, 507.6145, 508.1810, 508.5513, 508.6277, 509.1187,   SATC F13
     509.1958, 509.5652, 510.1318, 510.2090, 510.5734, 511.1458,   SATC F14
     511.5166, 511.5908, 512.1532, 512.1589, 512.5313, 512.5040,   SATC F15
     512.6006, 513.1721, 513.6422, 513.6179, 513.6145, 514.1553,   SATC F16
     514.5547, 514.6311, 515.5679, 516.1353, 516.2124, 516.5796,   SATC F17
     516.5762, 670.3574, 671.7214, 672.7969, 671.3609, 671.7346,   SATC F18
     671.8101, 672.3833, 672.7471, 672.8240, 673.3965, 673.7603,   SATC F19
     673.8379, 674.3303, 674.6997, 674.7742, 675.3435, 675.7124,   SATC F20
     675.7874, 676.2817, 676.7249, 676.8018, 677.2642, 677.7400,   SATC F21
     678.1074, 678.7512, 678.8274, 679.3077, 679.7465, 679.8442,   SATC F22
     680.3337, 680.7061, 681.2708, 681.3469, 681.7185, 681.7908,   SATC F23
     682.2854, 682.3606, 682.8040, 683.3777, 683.7449, 683.8170/   SATC F24
 DATA DAYN12/                                                      SATC F25
     684.2108, 684.7546, 684.8311, 685.3240, 685.4014, 685.7678,   SATC F26
     686.3374, 686.7073, 686.7813, 687.3506, 687.7240, 687.7942,   SATC F27
     688.2645, 688.7324, 688.8081, 689.2456, 689.8213, 690.3909,   SATC F28
     690.7588, 691.2715, 691.9464, 692.3409, 692.7109, 692.7847,   SATC F29
     693.3540, 693.7241, 693.7983, 694.1679, 694.7335, 694.8115,   SATC F30
     695.2811, 695.7400, 695.8247, 696.2179, 696.7625, 696.8384,   SATC F31
     697.3311, 697.7756, 698.3442, 698.7383, 698.8657, 699.3574,   SATC F32
     699.8018, 700.2713, 700.7400, 700.8149, 701.3142, 701.7534,   SATC F33
     701.8281, 702.4006, 702.7659, 702.8423, 703.3345, 703.4109,   SATC F34
     703.7791, 704.3477, 704.7939, 704.8691, 705.3608, 705.4380,   SATC F35
     705.8066, 706.3740, 706.7471, 706.8134, 707.3123, 707.7588,   SATC F36
     707.8303, 708.3247, 708.4014, 708.7748, 709.3379, 709.4146,   SATC F37
     709.7854, 710.4275, 711.7993, 711.8645, 711.4414, 711.8101,   SATC F38
     712.2777, 712.7476, 712.8228, 713.3904, 713.7632, 713.8356,   SATC F39
     714.3281, 714.4048, 714.8491, 715.3416, 715.4177, 715.7866,   SATC F40
     716.2547, 716.4309, 716.7998, 717.3673, 717.4456, 717.8130,   SATC F41
     718.3611, 718.4583, 718.8262, 719.4043, 719.7636, 719.8394,   SATC F42
     720.2811, 720.4075, 720.6525, 721.3450, 721.4211, 721.7900,   SATC F43
     722.3582, 722.8025, 723.3713, 723.4478, 723.8179, 724.3845/   SATC F44
 DATA DAYN13/                                                      SATC F45
     724.4617, 724.8296, 725.2226, 725.7703, 725.8428, 726.4100,   SATC F46
     726.7813, 726.8550, 727.3494, 727.4253, 727.7935, 728.4414,   SATC F47
     728.8066, 728.8838, 729.3198, 730.1882, 730.4992, 730.8333,   SATC F48
     731.4014, 731.7722, 731.8464, 732.3790, 732.4145, 733.3521,   SATC F49
     733.4277, 737.8035, 734.2652, 734.8123, 735.3734, 735.4548,   SATC F50
     735.4250, 736.2916, 736.4640, 736.8581, 737.4045, 737.4812,   SATC F51
     737.8409, 738.2423, 738.4140, 738.8639, 739.4312, 739.8027,   SATC F52
     740.2770, 740.8579, 741.4443, 741.8145, 741.8811, 741.4579,   SATC F53
     742.2345, 742.4714, 742.8471, 743.4067, 743.4075, 743.8533,   SATC F54
     744.1896, 744.4214, 744.8972, 745.4768, 745.8811, 746.4478,   SATC F55
     746.4174, 747.0146, 747.3[illegible], 748.0177, 748.4473, 749.0110,   SATC F56
     749.3567, 750.4241, 750.6714, 751.0416, 751.3878, 752.4512,   SATC F57
     752.8213, 753.4080, 753.8347, 754.0314, 754.8484, 755.4146,   SATC F58
     755.8834, 756.4977, 756.8742, 757.4434, 757.8842, 758.4575,   SATC F59
```

```
     752.9019,  759.4707,  759.8331,  761.4044,  761.9506,  761.4180,    SATC F60
     761.9638,  762.4312,  762.9770,  763.4443,  763.8979,  764.4576,    SATC F61
     764.9048,  765.4714,  765.8430,  766.3543,  767.8472,  767.9470,    SATC F62
     768.4746,  769.8804,  769.4478,  769.8943,  770.3853,  770.9075,    SATC F63
     771.3984,  771.8464,  772.4116,  772.4592,  773.4248,  773.8706/    SATC F64
 DATA DAYN14/                                                            SATC F65
     774.4409,  774.8838,  775.4512,  775.4984,  776.4644,  776.9116,    SATC F66
     777.3805,  777.9241,  778.4915,  778.8611,  779.4262,  779.8755,    SATC F67
     780.4443,  780.8875,  781.4542,  781.9036,  782.4707,  783.9150,    SATC F68
     787.4039,  783.9277,  784.4978,  784.8660,  785.4458,  785.8792,    SATC F69
     786.4478,  786.8929,  787.4617,  787.9041,  788.4714,  788.9172,    SATC F70
     789.4495,  789.9319/                                                SATC F71
 DATA CL10/                                                              SATC F72
     -0.132,    -0.124,    -0.178,    -0.218,    -0.216,    -0.257,     SATC F73
     -0.305,    -0.300,     0.349,    -0.473,    -0.375,     0.050,     SATC F74
      0.015,    -0.024,    -0.027,    -0.074,    -0.104,    -0.100,     SATC F75
     -0.139,    -0.190,    -0.184,    -0.228,    -0.264,    -0.258,     SATC F76
     -0.316,    -0.352,    -0.346,    -0.391,    -0.396,    -0.420,     SATC F77
     -0.459,    -0.485,    -0.490,    -0.494,    -0.557,    -0.555,     SATC F78
      0.165,     0.122,     0.099,     0.060,     0.035,     0.071,     SATC F79
     -0.012,    -0.015,    -0.048,    -0.096,    -0.134,    -0.138,     SATC F80
     -0.176,    -0.210,    -0.211,    -0.248,    -0.277,    -0.277,     SATC F81
     -0.213,    -0.343,    -0.346,    -0.377,    -0.431,    -0.415,     SATC F82
     -0.459,    -0.486,    -0.473,     0.306,     0.291,     0.276,     SATC F83
      0.271,     0.254,     0.236,     0.233,     0.211,     0.193,     SATC F84
      0.196,     0.182,    -0.270,    -0.203,    -0.209,     0.473,     SATC F85
      0.477,     0.468,     0.466,     0.464,     0.468,     0.460,     SATC F86
      0.454,     0.461,     0.477,     0.422,     0.423,     0.434,     SATC F87
      0.422,     0.411,     0.360,     0.221,     0.210,     0.211,     SATC F88
      0.190,     0.176,     0.179,     0.178,     0.169,     0.176,     SATC F89
      0.167,     0.168,     0.164,     0.169,     0.172,     0.172,     SATC F90
      0.178,     0.180,     0.184,     0.183,     0.183,     0.187/     SATC F91
 DATA CL11/                                                              SATC F92
      0.194,     0.199,     0.203,     0.204,     0.215,     0.216,     SATC F93
      0.232,     0.205,     0.212,     0.220,     0.227,     0.230,     SATC F94
      0.237,     0.245,     0.240,     0.247,     0.256,     0.252,     SATC F95
      0.244,     0.250,     0.226,     0.218,     0.224,     0.220,     SATC F96
      0.229,     0.221,     0.229,     0.239,     0.242,     0.240,     SATC F97
      0.240,     0.249,     0.260,     0.257,     0.256,     0.280,     SATC F98
      0.279,     0.280,     0.323,     0.314,     0.324,     0.251,     SATC F99
      0.261,     0.254,     0.249,     0.246,     0.249,     0.238,     SATC F00
      0.237,     0.234,     0.233,     0.234,     0.226,     0.230,     SATC F01
      0.272,     0.234,     0.234,    -8.900,    -8.900,    -8.920,     SATC F02
      0.752,     0.334,     0.319,     0.322,    -0.040,    -0.040,     SATC F03
     -0.043,     0.0   ,   -0.324,    -0.328,    -0.329,    -0.334,     SATC F04
     -0.321,    -0.139,    -0.150,    -0.141,    -0.135,    -0.146,     SATC F05
     -0.139,    -0.153,    -0.144,    -0.138,    -0.131,    -0.137,     SATC F06
     -0.135,    -0.151,    -0.150,    -0.147,    -0.142,     0.0   ,    SATC F07
     -0.155,    -0.163,    -0.178,    -0.204,    -0.204,    -0.203,     SATC F08
     -0.197,    -0.203,    -0.189,    -0.188,    -0.199,    -0.185,     SATC F09
     -0.180,    -0.187,    -0.184,    -0.181,    -0.193,    -0.195,     SATC F10
     -0.194,    -0.187,    -0.203,    -0.215,    -0.211,    -0.215/     SATC F11
 DATA CL12/                                                              SATC F12
     -0.324,     0.0   ,   -0.231,    -0.230,    -0.247,    -0.240,     SATC F13
     -0.229,    -0.227,    -0.246,    -0.241,     0.193,     0.193,     SATC F14
      0.188,     0.160,     0.176,     0.164,     0.140,     0.166,     SATC F15
```

```
     *       0.162,    0.152,    0.142,    0.145,    0.154,    0.157,   SATC 416
     *       0.163,    0.135,    0.145,    0.184,    0.141,    0.140,   SATC 417
     *       0.145,    0.124,    0.126,    0.102,    0.077,    0.079,   SATC 418
     *       0.075,    0.083,    0.095,    0.095,    0.099,    0.107,   SATC 419
     *       0.106,    0.114,    0.113,    0.118,    0.0  ,    0.0  ,   SATC 420
     *      -0.045,   -0.050,   -0.055,   -0.060,   -0.065,   -0.070,   SATC 421
     *      -0.070,   -0.075,   -0.080,   -0.085,   -0.090,   -0.090,   SATC 422
     *      -0.095,   -0.100,   -0.110,   -0.110,   -0.115,   -0.116,   SATC 423
     *      -0.107,   -0.107,   -0.119,   -0.105,   -0.125,   -0.132,   SATC 424
     *      -0.147,   -0.149,   -0.164,   -0.175,   -0.174,   -0.184,   SATC 425
     *      -0.196,   -0.194,   -0.211,   -0.210,   -0.226,   -0.226,   SATC 426
     *      -0.232,   -0.225,   -0.040,   -0.025,   -0.035,   -0.036,   SATC 427
     *      -0.064,   -0.081,   -0.097,   -0.097,   -0.110,   -0.111,   SATC 428
     *      -0.121,   -0.132,   -0.133,   -0.144,   -0.160,   -0.173,   SATC 429
     *      -0.186,   -0.176,    0.240,    0.229,    0.236,    0.226,   SATC 430
     *       0.210,    0.194,    0.192,    0.196,    0.175,    0.179/   SATC 431
      DATA CL13/                                                        SATC 432
     *       0.171,    0.165,    0.154,    0.144,    0.149,    0.144,   SATC 433
     *       0.133,    0.134,    0.128,    0.135,    0.126,    0.116,   SATC 434
     *       0.117,    0.110,    0.092,    0.089,    0.094,    0.090,   SATC 435
     *       0.091,    0.065,    0.065,    0.070,    0.083,    0.053,   SATC 436
     *       0.057,    0.034,    0.045,    0.024,    0.031,    0.029,   SATC 437
     *       0.031,    0.050,    0.038,    0.032,    0.015,    0.002,   SATC 438
     *       0.027,    0.010,    0.017,    0.017,    0.009,    0.010,   SATC 439
     *       0.028,    0.018,    0.013,    0.002,    0.016,    0.015,   SATC 440
     *       0.023,    0.015,    0.009,    0.020,    4.730,    4.740,   SATC 441
     *       0.120,    0.124,    0.115,    0.116,    0.125,    0.127,   SATC 442
     *       0.116,    0.125,    0.115,    0.119,    0.109,    0.122,   SATC 443
     *       0.122,    0.124,    0.118,    0.124,    0.112,    0.109,   SATC 444
     *       0.106,    0.097,    0.095,    0.103,    0.095,    0.090,   SATC 445
     *       0.0  ,    0.095,    0.089,    0.097,    0.081,    0.066,   SATC 446
     *       0.055,    0.056,    0.043,    0.071,    0.027,    0.037,   SATC 447
     *       0.029,    0.037,    0.028,    0.030,    0.029,    0.028,   SATC 448
     *       0.024,    0.023,    0.029,    0.033,    0.235,    0.254,   SATC 449
     *       0.350,    0.350,    0.305,    0.300,    0.353,    0.325,   SATC 450
     *       0.324,    0.307,    0.316,   -0.093,   -0.092,   -0.094/   SATC 451
      DATA CL14/                                                        SATC 452
     *      -0.092,   -0.100,   -0.091,   -0.093,   -0.094,   -0.092,   SATC 453
     *      -0.089,   -0.090,   -0.079,   -0.091,   -0.087,   -0.091,   SATC 454
     *      -0.091,   -0.053,   -0.101,   -0.104,   -0.106,   -0.115,   SATC 455
     *      -0.120,   -0.120,   -0.096,   -0.130,   -0.130,   -0.143,   SATC 456
     *      -0.132,   -0.125,   -0.117,   -0.123,   -0.118,   -0.118,   SATC 457
     *      -0.126,   -0.113/                                          SATC 458
      DATA NCLOCK/489/                                                  SATC 459
      DO 10 I=2,NCLOCK                                                  SATC 460
      IF(REFTIM.GT.DAYNO(I)) GO TO 10                                   SATC 461
C INTERPOLATE                                                           SATC 462
      RATE=(CLC(I)-CLC(I-1))/(DAYNO(I)-DAYNO(I-1))                      SATC 463
      SATC22=(RATE*(REFTIM-DAYNO(I-1))+CLC(I-1))*1.E-3                  SATC 464
      RETURN                                                            SATC 465
   10 CONTINUE                                                          SATC 466
      RETURN                                                            SATC 467
      END                                                               SATC 468
```

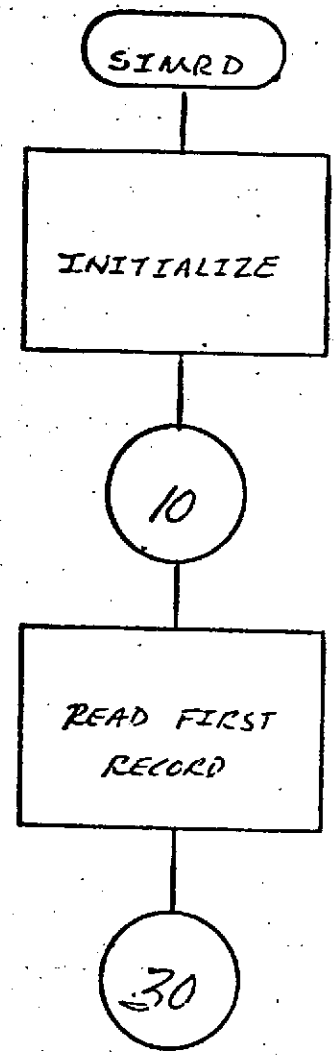
NAME	SIMRD				
PURPOSE	TO READ SIMULATED DATA TAPES				
CALLING SEQUENCE	CALL SIMRD(NSTARD)				
	SYMBOL TYPE	DESCRIPTION			
	NSTARD I	INPUT-- NUMBER OF STATIONS THAT WERE READ			
SUBROUTINES USED	RANDWR	TDIF	YMDAY	NUMBR2	BIAS
COMMON BLOCKS	APARAM	CEPHEM	CGEOS	CONSTS	CPARAM
	CSTINF	CTIME	INTBLK	PREBLK	SIGBLK
	STANUM				
INPUT FILES	IOBS - DATA TAPE				
OUTPUT FILES	PRINTER				
REFERENCE	'GEODYN PROGRAM OPERATIONS DESCRIPTION' - APPENDIX C VOLUME 3 - GEODYN DOCUMENTATION				

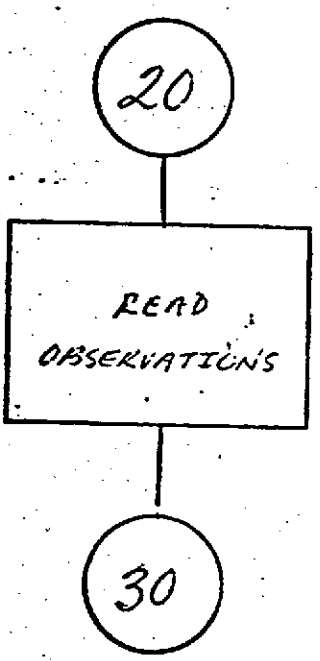
```
      SUBROUTINE SIMRD(NSTARD)                                          SIMR  27
      IMPLICIT REAL*8 (A-H,O-Z)                                         SIMR  28
      LOGICAL*1 OKSAT,VHECHN,PREPEC                                     SIMR  29
      LOGICAL NDATE,SATSAT                                              SIMR  30
      INTEGER*2 CELL,CHANEL,NVEAS,MTYPE,PRETYP,IMTYPE,ISTNO,ISATNO,     SIMR  31
     *   ITYPE,ESTAND,ISTAPE,ISTAND,STANCS,JBASE,KBASE                  SIMR  32
      INTEGER RECNO                                                     SIMR  33
      REAL TDIF,SIGSTD,DAYINT,SIGCHG,SGPRAT                             SIMR  34
      DOUBLE PRECISION JNAME,NAME                                       SIMR  35
      DIMENSION SIG(2)                                                  SIMR  36
      COMMON/APARAM/INPAR,INPARI,NSIAS,NSTSTA,NSAT,NGPREC(5)            SIMR  37
      COMMON/CEPHEM/JNAME(361),ISTAPE(361),ESTAND(361),ISTNO(366)       SIMR  38
      COMMON/CGEOS/ISAT2(2),IPREPR(250),NPEC,NSIG,NCULL,SIGCHG(50),     SIMR  39
     *   IMTYPE(50),ISTNO(50),CELL(2,100)                               SIMR  40
      COMMON/CONSTS/PI,DTWOPI,D2R,S2R                                   SIMR  41
      COMMON/CPARAM/NSTA,NMAST(19)                                      SIMR  42
      COMMON/CSTINF/JBASE(283),KBASE(283),LBASE                         SIMR  43
      COMMON/CTIME/DATAEP,DAYREF,DAYE,DAYSTP,DAYINT(15)                 SIMR  44
      COMMON/INTBLK/INTBK1(53),NDATE,INTBK2(78)                         SIMR  45
      COMMON/PREBLK/DAY,OBS1,OBS2,SIG1,SIG2,SPRNDX,ISTA,MTYPE,NMEAS,    SIMR  46
     *   ISATNO,PRETYP,CHANEL,VHECHN,PREPEC,PRCNO                       SIMR  47
      COMMON/SIGBLK/SIGSTD(30),SGPRAT(30),IOBS,IOTAPE(3)                SIMR  48
      COMMON/STANUM/NAME(280),STANCS(280),NOSTOP                        SIMR  49
      EQUIVALENCE (SIG(1),SIG1),(KKSAT,SIG2)                           SIMR  50
      ATIME(DAY)=TDIF(4,3,DAY)/8.64E4                                   SIMR  51
C INITIALIZE                                                            SIMR  52
      IF(IOBS.LE.0) IOBS=21                                             SIMR  53
```

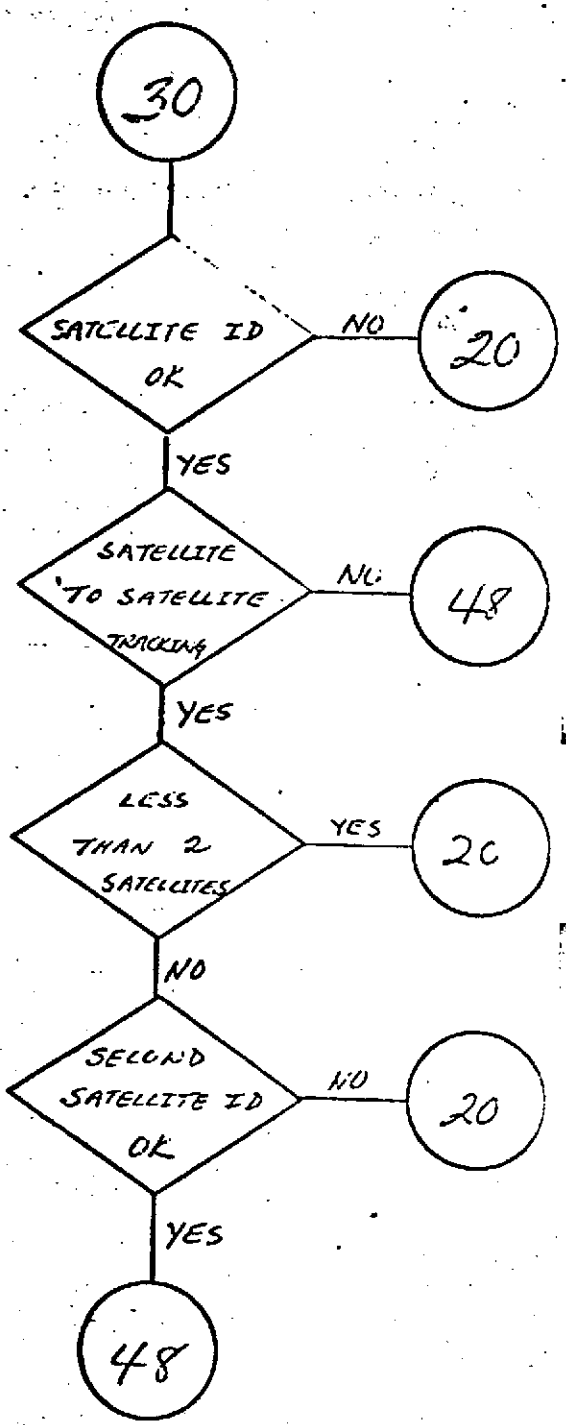
```
      [illegible]                                                       SIMR  93
      NOSAT =.FALSE.                                                    SIMR  94
      [illegible]                                                       SIMR  95
      WHICH=.FALSE.                                                     SIMR  97
      DO 100=.FALSE.                                                    SIMR  98
C [illegible]                                                           SIMR  99
   10 READ(IOPS,[illegible]) IYMD,IHM,SEC,C1,C2,SIG1,SIG2,ISN,         SIMR  01
     *    MTYPE,[illegible],ISTID,SATX,CHANL                           SIMR  01
      IF(ISN.EQ.0) ISATID(I)=ISATID                                     SIMR  02
      GO TO 30                                                          SIMR  03
C READ OBSERVATIONS                                                     SIMR  04
   20 READ(IOPS,[illegible]) IYMD,IHM,SEC,C1,C2,SIG1,SIG2,ISN,         SIMR  05
     *    MTYPE,[illegible],ISTID,SATX,CHANL                           SIMR  06
C CHECK SATELLITE ID                                                    SIMR  07
   30 OKSAT=.FALSE.                                                     SIMR  08
      DO 40 I=1,NSAT                                                    SIMR  09
      IF(OKSAT) GO TO 40                                                SIMR  70
      ISATNO=I                                                          SIMR  71
      OKSAT=ISATID.EQ.ISAT2(I)                                          SIMR  72
   40 CONTINUE                                                          SIMR  73
      IF(.NOT.OKSAT) GO TO 20                                           SIMR  74
C CHECK SECOND SATELLITE ID IF SAT-SAT TRACKING                         SIMR  75
      IF(MTYPE.NE.2.AND.MTYPE.NE.3) GO TO 48                            SIMR  76
      SATSAT=KKSAT.GT.0                                                 SIMR  77
      IF(.NOT.SATSAT) GO TO 48                                          SIMR  78
      IF(NSAT.LT.2) GO TO 20                                            SIMR  79
      OKSAT=.FALSE.                                                     SIMR  80
      DO 45 I=1,NSAT                                                    SIMR  81
      OKSAT=KKSAT.EQ.ISAT2(I)                                           SIMR  82
      IF(.NOT.OKSAT) GO TO 45                                           SIMR  83
      KKSAT=I                                                           SIMR  84
      GO TO 48                                                          SIMR  85
   45 CONTINUE                                                          SIMR  86
      GO TO 20                                                          SIMR  87
   48 NN1=0                                                             SIMR  88
      NN2=0                                                             SIMR  89
      IF(NSIG.LE.0) GO TO 60                                            SIMR  90
C CHECK FOR SIGMA CHANGE                                                SIMR  91
      DO 50 I=1,NSIG                                                    SIMR  92
      IF(ISN.NE.ISTNO(I).AND.ISTNO(I).NE.0) GO TO 50                    SIMR  93
      IF(MTYPE.EQ.IMTYP(I).OR.IMTYP(I).EQ.0) NN1=I                      SIMR  94
      IF(MTYPE.GT.7) GO TO 50                                           SIMR  95
      IF(MTYPE+7.EQ.IMTYPE(I).OR.IMTYPE(I).EQ.0) NN2=I                  SIMR  96
   50 CONTINUE                                                          SIMR  97
   60 ISTA=ISN                                                          SIMR  98
C COMPUTE TIME IN A.1 DAYS FROM JAN 0.0                                 SIMR  99
      DAY=YMDAY(IYMD,IHM,SEC)                                           SIMR 100
      DAY=DAY+ATIME(DAY)                                                SIMR 101
      IF(DAY.LT.DATAEP) GO TO 20                                        SIMR 102
      IF(DAY.GT.DAYSTP) GO TO 200                                       SIMR 103
C CHECK FOR STATION PRESENT                                             SIMR 104
      IF(ISN.EQ.0) GO TO 100                                            SIMR 105
      ISN=NUMBR2(ISTA,ISTAND,NSTA)                                      SIMR 106
      IF(ISN.GT.0) GO TO 90                                             SIMR 107
      ISN=NUMBR2(ISTA,ISTAND,NSTAND)                                    SIMR 108
      IF(ISN.GT.0) GO TO 80                                             SIMR 109
```

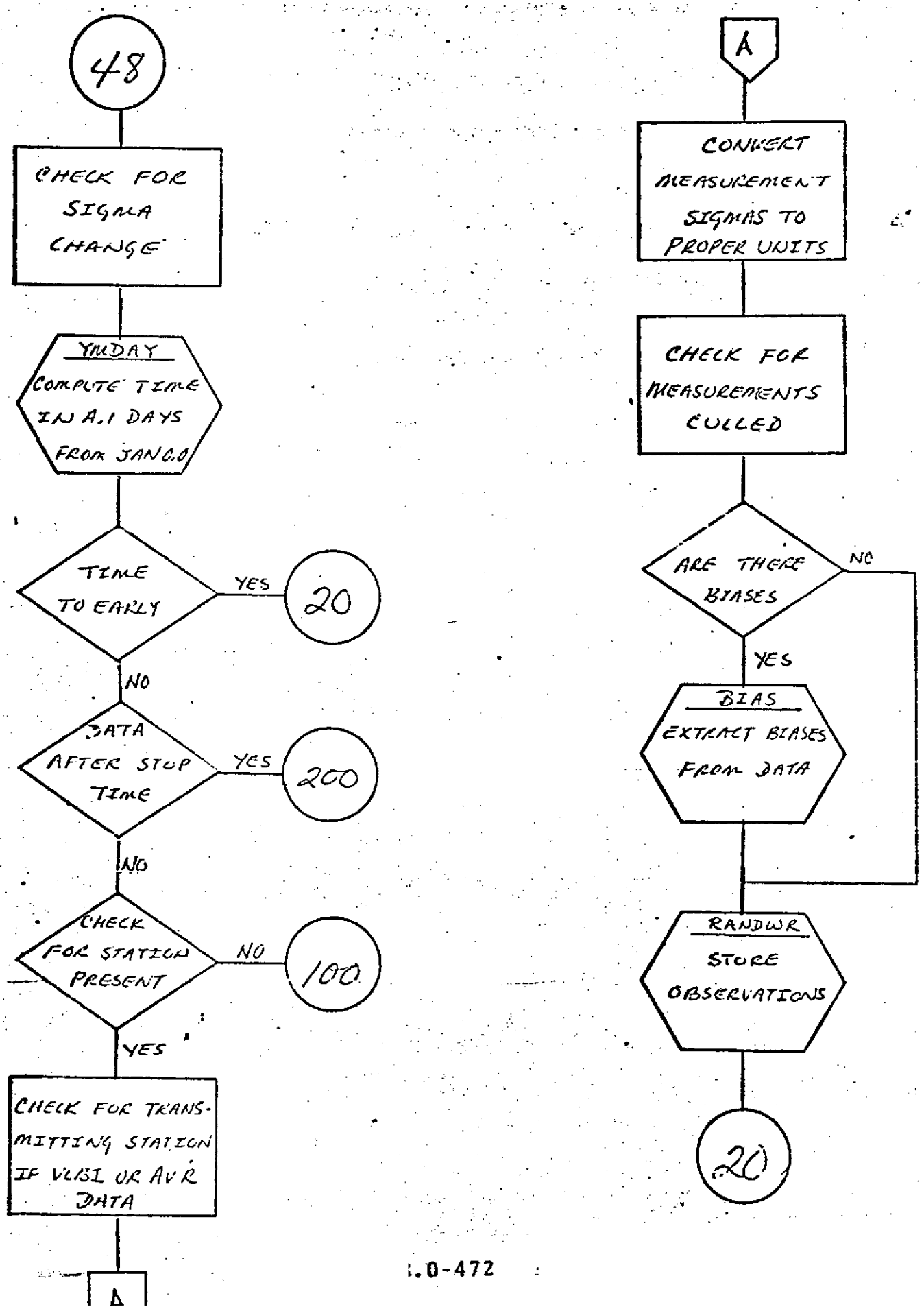
```
      ISN=NLOCAT(ISTA,ISTANO,NOSTNS)                                    SIMR 110
      IF(ISN.GT.0) GO TO 92                                             SIMR 111
      IF(ISTA.EQ.ISTAV.AND.MTYP.EQ.2) GO TO 92                          SIMR 112
      PRINT 10,ISTA                                                     SIMR 113
      GO TO 91                                                          SIMR 114
   90 NSTA=NSTA+1                                                       SIMR 115
      ISTAN(NSTA)=ISTA                                                  SIMR 116
      ISN=NSTA                                                          SIMR 117
   92 ISTA=ISTAV                                                        SIMR 118
      IF(MTYP.LT.27) GO TO 100                                          SIMR 119
C CHECK FOR TRANSMITTING STATION IF VLBI OR SVT DATA                    SIMR 120
      ISTA=ISTAT                                                        SIMR 121
      ISN2=NLOCAT(ISTAV,ISTAN,NSTA)                                     SIMR 122
      IF(ISN2.GT.0) GO TO 94                                            SIMR 123
      ISN2=NLOCAT(ISTAT,ISTAN,NSTA)                                     SIMR 124
      IF(ISN2.GT.0) GO TO 93                                            SIMR 125
      ISN2=NLOCAT(ISTA2,ISTAN,NOSTNS)                                   SIMR 126
      IF(ISN2.GT.0) GO TO 93                                            SIMR 127
      PRINT 10,ISTA2                                                    SIMR 128
      GO TO 20                                                          SIMR 129
   93 NSTA=NSTA+1                                                       SIMR 130
      ISN2=NSTA                                                         SIMR 131
      ISTAN(ISN2)=ISTA2                                                 SIMR 132
   94 CHANL=ISN2                                                        SIMR 133
      IF(LBAS.LE.0) GO TO 95                                            SIMR 134
      DO 96 I=1,LBAS                                                    SIMR 135
      IF(ISN.NE.JBAS(I)) GO TO 96                                       SIMR 136
      IF(ISN2.NE.KBAS(I)) GO TO 96                                      SIMR 137
      GO TO 100                                                         SIMR 138
   96 CONTINUE                                                          SIMR 139
   98 LBAS=LBAS+1                                                       SIMR 140
      JBAS(LBAS)=ISN                                                    SIMR 141
      KBAS(LBAS)=ISN2                                                   SIMR 142
  100 IF(NN1.LT.0) GO TO 150                                            SIMR 143
C CONVERT MEASUREMENT SIGMAS TO PROPER UNITS                            SIMR 144
      IF(MTYP.GT.14) GO TO 300                                          SIMR 145
      GO TO (11),115,120,115,130,135,110),MTYP                          SIMR 146
  110 SIG1=SIGCHG(NN1)*S2R/DCOS(OBS2)                                   SIMR 147
      GO TO 150                                                         SIMR 148
  115 SIG1=SIGCHG(NN1)                                                  SIMR 149
      GO TO 150                                                         SIMR 150
  120 SIG1=0.0150*SIGCHG(NN1)                                           SIMR 151
      GO TO 150                                                         SIMR 152
  130 SIG1=OBS1**2                                                      SIMR 153
      IF(SIG1.LT.1.00) SIG1=DSQRT(1.000-SIG1)                           SIMR 154
      SIG1=SIGCHG(NN1)*1.0D-3/SIG1                                      SIMR 155
      GO TO 150                                                         SIMR 156
  135 SIG1=SIGCHG(NN1)*S2R                                              SIMR 157
  150 IF(NN2.LE.0) GO TO 170                                            SIMR 158
      GO TO (155,170,170,160,165,155,155),MTYP                          SIMR 159
  155 SIG2=SIGCHG(NN2)*S2R                                              SIMR 160
      GO TO 170                                                         SIMR 161
  160 SIG2=SIGCHG(NN2)*0.0150                                           SIMR 162
      GO TO 170                                                         SIMR 163
  165 SIG2=OBS2**2                                                      SIMR 164
      IF(SIG2.LT.1.000) SIG2=DSQRT(1.000-SIG2)                          SIMR 165
```

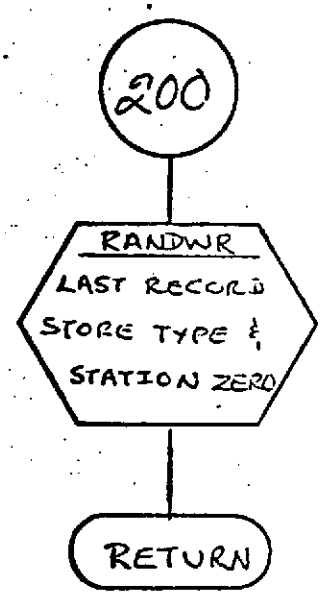
```
      SIG2=SIGCH3(NN1)*1.E-3/SIG2                                       SIMR  146
      GO TO 171                                                         SIMR  147
  300 MTP=MTYP-10                                                       SIMR  148
      IF(MTP.GT.15) GO TO 22                                            SIMR  149
      GO TO (171,17,17,311,18,312,313,321,322,                          SIMR  150
     *   323,324,171,171,175),MTP                                       SIMR  151
  311 SIG1=SIGCH3(NN1)*1.E-3                                            SIMR  152
      GO TO 17                                                          SIMR  153
  312 SIG1=SIGCH3(NN1)*1.E-3                                            SIMR  154
      GO TO 17                                                          SIMR  155
  313 SIG1=SIGCH3(NN1)*1.E-3                                            SIMR  156
      GO TO 17                                                          SIMR  157
  321 SIG1=SIGCH3(NN1)*1.E-3                                            SIMR  158
  322 IF(NCULL.EQ.0) GO TO 185                                          SIMR  159
C CHECK FOR MEASUREMENTS CULLED                                         SIMR  160
      DO 185 I=1,NCULL                                                  SIMR  161
      DO 185 J=1,NMEAS                                                  SIMR  162
      IF(NUMBER+J-CULL(1,I)) 185,181,175                                SIMR  163
  175 IF(NUMBER+J.GT.CULL(2,I)) GO TO 185                               SIMR  164
  181 SIG(J)=0.0                                                        SIMR  165
  185 CONTINUE                                                          SIMR  166
  185 NUMBER=NUMBER+NMEAS                                               SIMR  167
      RECNO=RECNO+1                                                     SIMR  168
C CHECK FOR BIASES                                                      SIMR  169
      IF(NBIAS.GT.0) CALL BIAS                                          SIMR  170
C STORE OBSERVATIONS                                                    SIMR  171
      CALL RANDW                                                        SIMR  172
      GO TO 20                                                          SIMR  173
  200 MTYPE=0                                                           SIMR  174
      IST=0                                                             SIMR  175
C LAST RECORD STORE TYPE AND STATION ZERO                               SIMR  176
      RECNO=RECNO+1                                                     SIMR  177
      CALL RANDW                                                        SIMR  178
      PRINT 2000,NUMSEP,IOBS                                            SIMR  179
      DAYSTP=DAY                                                        SIMR  180
      RETURN                                                            SIMR  181
 1000 FORMAT(1X,'STATION ',I4,' NOT FOUND IN FILE')                     SIMR  182
 2000 FORMAT(1H0//25X,I6,' OBSERVATIONS SELECTED FROM MASTER SIMULATED',SIMR  183
     *   ' DATA TAPE NUMBER',I3)                                        SIMR  184
      END                                                               SIMR  185
```


SIMRD
INITIALIZE
10
READ FIRST RECORD
30


20
READ OBSERVATIONS
30


30
SATELLITE ID OK
NO
20
YES
SATELLITE TO SATELLITE TRACKING
NO
48
YES
LESS THAN 2 SATELLITES
YES
20
NO
SECOND SATELLITE ID OK
NO
20
YES
48


48
CHECK FOR SIGMA CHANGE
YMDAY
COMPUTE TIME IN A.1 DAYS FROM JAN 0.0
TIME TO EARLY
YES
20
NO
DATA AFTER STOP TIME
YES
200
NO
CHECK FOR STATION PRESENT
NO
100
YES
CHECK FOR TRANSMITTING STATION IF VLBI OR AVR DATA
A
A
CONVERT MEASUREMENT SIGMAS TO PROPER UNITS
CHECK FOR MEASUREMENTS CULLED
ARE THERE BIASES
NO
YES
BIAS
EXTRACT BIASES FROM DATA
RANDWR
STORE OBSERVATIONS
20


200
RANDWR
LAST RECORD
STORE TYPE &
STATION ZERO
RETURN

SQUANT

DESCRIPTION

SQUANT is a subroutine which computes quantities related to the Earth-fixed station positions. The first call has a different processing than the subsequent or normal processing.

The first call processing for each station consists of

- convert ϕ, the geodetic latitude, and λ, the east longitude, to radians.
- compute the Cartesian coordinates.
- If the station is to be adjusted, compute the matrix of partial derivatives of the rectangular coordinates with respect to the geodetic spherical coordinates.
- compute the $\hat{N}$, $\hat{Z}$, and $\hat{E}$ with vectors describing the horizontal topocentric coordinate system.

Subsequent processing consists

- converting the adjusted rectangular station positions to spherical geodetic coordinates, and their covariances as well (using subroutine PLHOUT), and
- recompute the $\hat{N}$, $\hat{Z}$, and $\hat{E}$ unit vectors on the basis of the new (adjusted) positions.

NAME	SQUANT
ENTRY POINT	PURPOSE
SQANT1	INITIALIZATION
SQUANT	TO CONVERT STATION POSITIONS TO X,Y,Z COORDINATE SYSTEM ON FIRST CALL AND TO PHI,LAMBDA,H COORDINATE SYSTEM ON SUBSEQUENT CALLS AND TO COMPUTE PARTIALS OF X,Y,Z WITH RESPECT TO LAMBDA, PHI,H

CALLING SEQUENCE CALL SQANT1(H,PARTL,PLHSIG,STAXYZ,EHAT,NHAT,ZHAT, THPRIM,SLAT,SLON)

SYMBOL	TYPE	DESCRIPTION
H (1)	DP	INPUT & OUTPUT - STATION HEIGHT
PARTL (3,3,1)	R	OUTPUT - PARTIALS OF X,Y,Z WITH RESPECT TO PHI, LAMBDA, AND H
PLHSIG (3,3,1)	R	INPUT - STATION SPHERICAL COORDINATE SIGMAS
STAXYZ (3,1)	DP	INPUT - TRACKING STATION EARTH FIXED CARTESIAN COORDINATES
EHAT (3,1)	DP	OUTPUT - STATION EAST UNIT VECTOR
NHAT (3,1)	DP	OUTPUT - STATION NORTH UNIT VECTOR
ZHAT (3,1)	DP	OUTPUT - STATION LOCAL VERTICAL UNIT VECTOR
THPRIM (2,1)	DP	OUTPUT - SINE AND COSINE OF THE DIFFERENCE BETWEEN GEODETIC AND GEOCENTRIC LATITUDES
SLAT (1)	DP	INPUT & OUTPUT - TRACKING STATION LATITUDE
SLON (1)	DP	INPUT & OUTPUT - TRACKING STATION LONGITUDE

CALLING SEQUENCE CALL SQUANT(NSTA,NSTEST,FRSTIM)

SYMBOL	TYPE	DESCRIPTION
NSTA	I	INPUT - INTERNAL NUMBER OF STATIONS
NSTEST	I	INPUT - NUMBER OF ADJUSTED STATIONS

FRSTIN - INPUT - SWITCH IS TRUE ON FIRST CALL TO SQUANT FOR A CASE

SUBROUTINES USED CLEAR PLHOUT

COMMON BLOCKS INTBLK

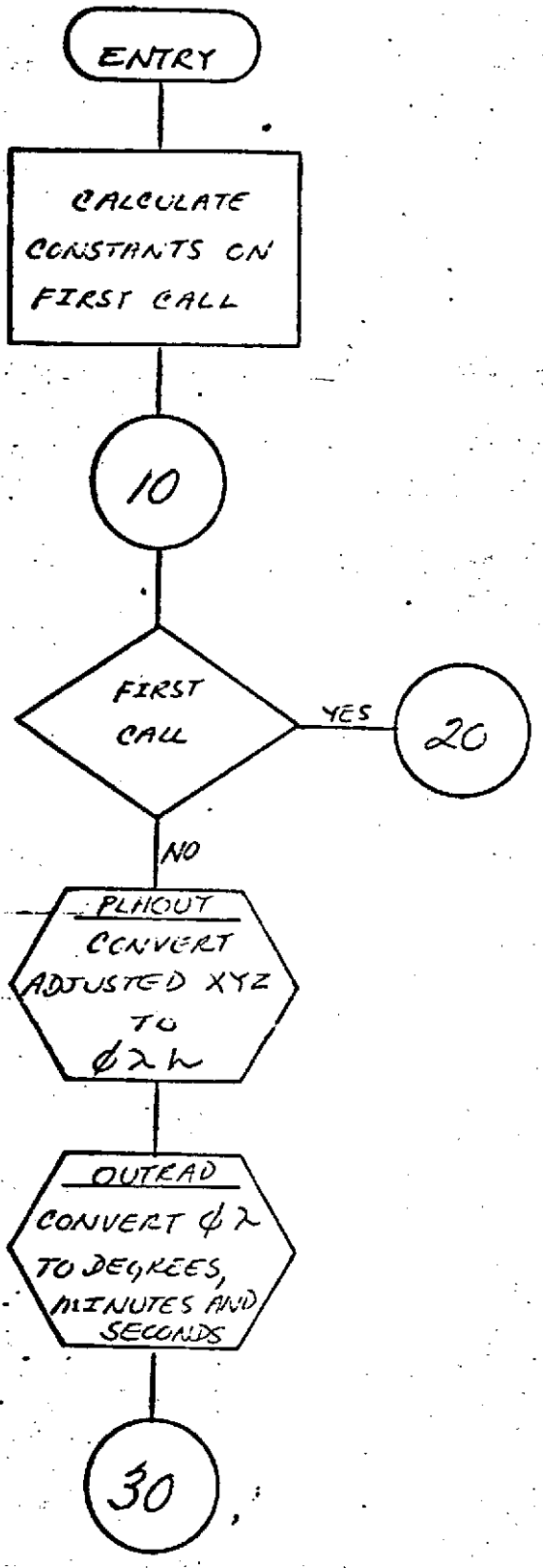
INPUT FILES NONE

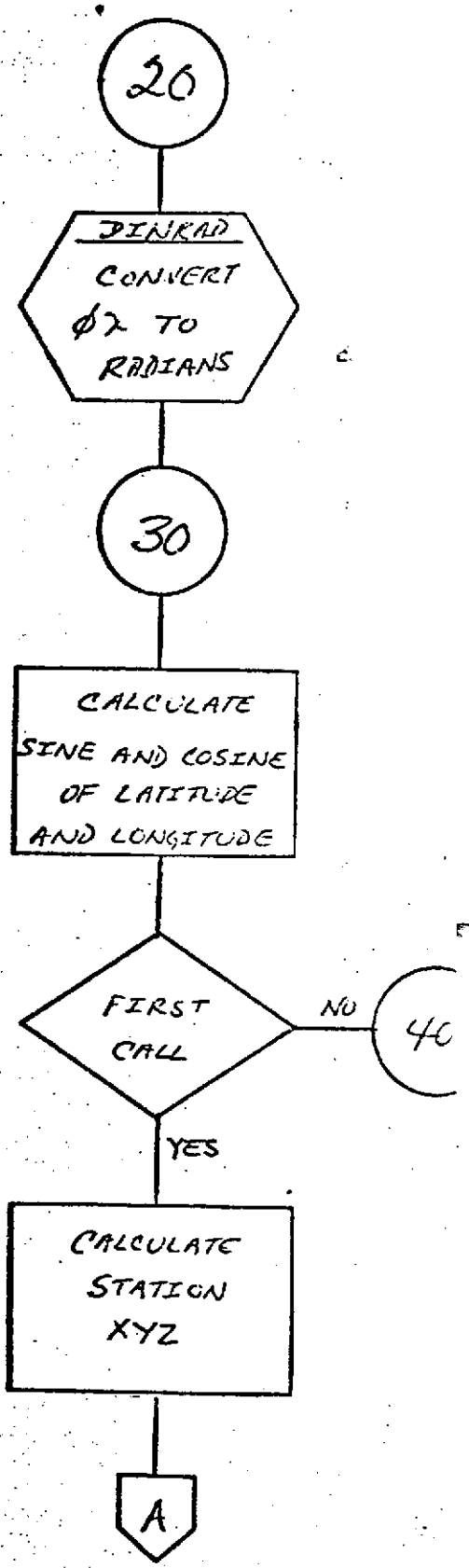
OUTPUT FILES NONE

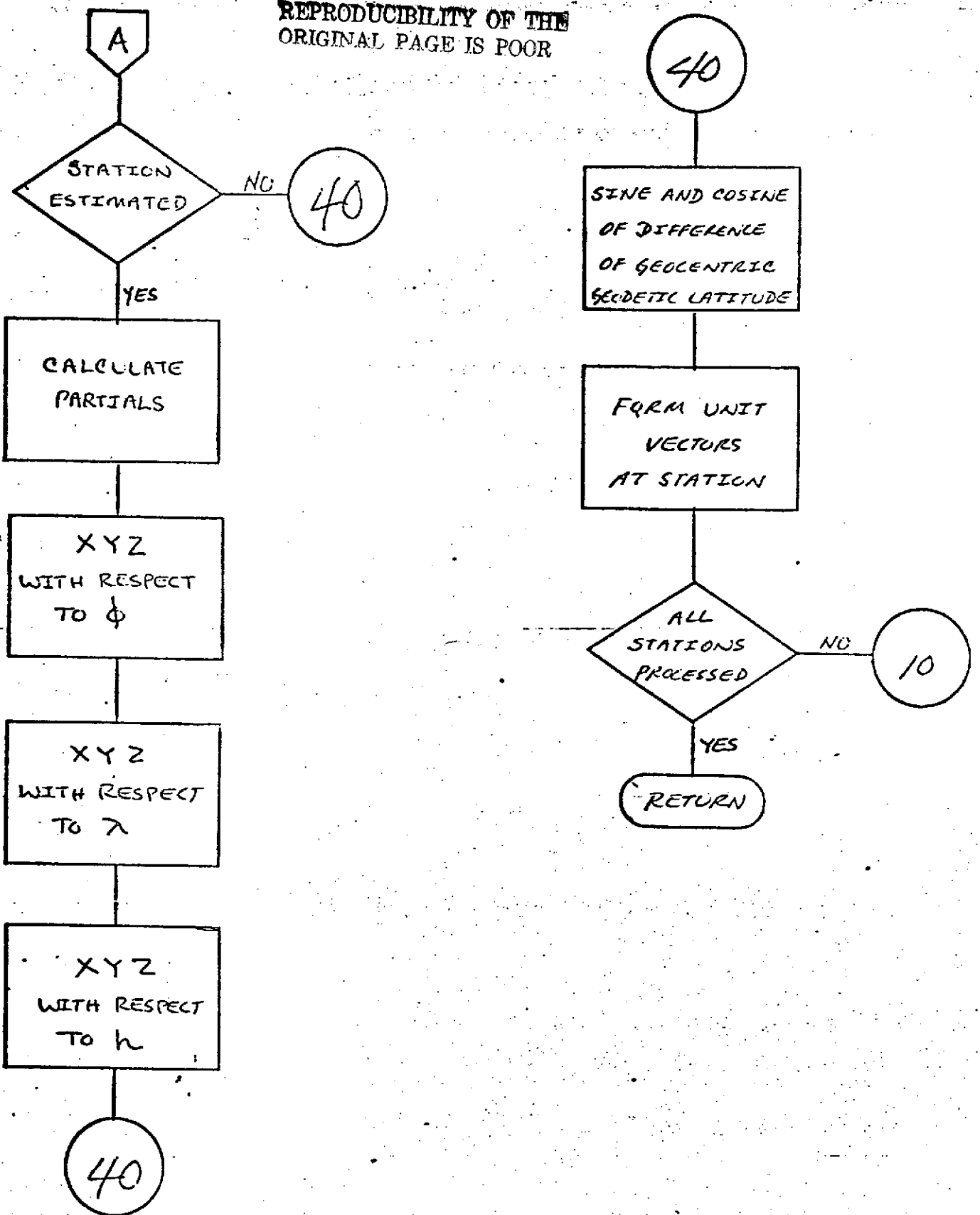
```
      SUBROUTINE SQUANT(H,PARTL,PLHSIG,STAXYZ,EHAT,NHAT,ZHAT,THPRIM,     SQUA  70
     *   RLAT,RLON,ISTAND)                                                SQUA  71
      IMPLICIT REAL*8 (A-H,O-Z)                                           SQUA  72
      DIMENSION H(1),PARTL(3,3,1),PLHSIG(3,3,1),STAXYZ(3,1),EHAT(3,1),    SQUA  73
     *  NHAT(3,1),ZHAT(3,1),THPRIM(3,1),RLAT(1),RLON(1)                   SQUA  74
      INTEGER*2 ISTAND                                                    SQUA  75
      DIMENSION ISTAND(1)                                                 SQUA  76
      LOGICAL FRSTIM                                                      SQUA  77
      REAL PARTL,PLHSIG                                                   SQUA  78
      DOUBLE PRECISION NHAT                                               SQUA  79
      COMMON/INTBLK/IGS(8),AE,AESQ,FLAT,FESQ(52)                          SQUA  80
      RETURN                                                              SQUA  81
      ENTRY SQUANT(NSTA,NSTEST,FRSTIM)                                    SQUA  82
C CALCULATE CONSTANTS ON FIRST CALL                                       SQUA  83
      IF(.NOT.FRSTIM) GO TO 10                                            SQUA  84
      ESQ=FLAT                                                            SQUA  85
      ESQ1=(1.0D0-ESQ)**2                                                 SQUA  86
      ESQ=1.0D0-ESQ1                                                      SQUA  87
      CALL CLEAR(STAXYZ,6,NSTA)                                           SQUA  88
      CALL CLEAR(EHAT,6,NSTA)                                             SQUA  89
      CALL CLEAR(NHAT,6,NSTA)                                             SQUA  90
      CALL CLEAR(ZHAT,6,NSTA)                                             SQUA  91
      CALL CLEAR(THPRIM,4,NSTA)                                           SQUA  92
   10 DO 100 I=1,NSTA                                                     SQUA  93
      IF(ISTAND(I).EQ.-4) GO TO 100                                       SQUA  94
      IF(FRSTIM) GO TO 20                                                 SQUA  95
C ON ALL CALLS AFTER FIRST, CONVERT ADJUSTED X,Y,Z TO PHI,LAMBDA,H        SQUA  96
      CALL PLHOUT(STAXYZ(1,I),PARTL(1,1,I),PLHSIG(1,1,I),RLAT(I),         SQUA  97
     *   RLON(I),H(I))                                                    SQUA  98
C CALCULATE SINE AND COSINE OF GEODETIC LATITUDE AND EAST LONGITUDE       SQUA  99
   20 SLATG=DSIN(RLAT(I))                                                 SQUA 100
      SLTGSQ=SLATG**2                                                     SQUA 101
      CLATG=DSQRT(1.0D0-SLTGSQ)                                           SQUA 102
      SINLON=DSIN(RLON(I))                                                SQUA 103
      COSLON=DCOS(RLON(I))                                                SQUA 104
      R=AE/DSQRT(1.0D0-ESQ*SLTGSQ)                                        SQUA 105
      RCL=CLATG*(H(I)+R)                                                  SQUA 106
C ON FIRST CALL, CALCULATE STATION X,Y,Z                                  SQUA 107
      IF(.NOT.FRSTIM) GO TO 30                                            SQUA 108
      STAXYZ(1,I)=RCL*COSLON                                              SQUA 109
      STAXYZ(2,I)=RCL*SINLON                                              SQUA 110
      STAXYZ(3,I)=CLATG*(H(I)+ESQ1*R)                                     SQUA 111
```

```
C IF FIRST CALL AND STATION TO BE ADJUSTED, CALCULATE PARTIALS
      IF(I.GT.NSTEST) GO TO 40
C INTERMEDIATE CONSTANTS
      ESQPHI=F2Q*SLATG**2
      E1PHI=1.0D0-ESQPHI
      CPCL=CLATG*COSLON
      CPSL=CLATG*SINLON
      ESQ1P=E*ESQ1
      HR=H(I)+R
C PARTIALS OF X,Y,Z WITH RESPECT TO PHI
      PARTL(1,1,I)=-SLATG*COSLON*(HR-(ESQ*R*CLATG**2)/E1PHI)
      PARTL(1,2,I)=SINLON*PARTL(1,1,I)/COSLON
      PARTL(1,3,I)=CLATG*(H(I)+ESQ1R*(1.0D0+ESQPHI/E1PHI))
C PARTIALS OF X,Y,Z WITH RESPECT TO LAMBDA
      PARTL(2,1,I)=-STAXYZ(2,I)
      PARTL(2,2,I)=STAXYZ(1,I)
      PARTL(2,3,I)=0.
C PARTIALS OF X,Y,Z WITH RESPECT TO HEIGHT
      PARTL(3,1,I)=CPCL
      PARTL(3,2,I)=CPSL
      PARTL(3,3,I)=SLATG
C SINE AND COSINE OF DIFFERENCE OF GEOCENTRIC & GEODETIC LATITUDES
   40 THETPR=PLAT(I)-DATAN(STAXYZ(3,I)/RCL)
      THPRIM(1,I)=DSIN(THETPR)
      THPRIM(2,I)=DCOS(THETPR)
C FORM UNIT VECTORS AT STATION
C ...NORTH VECTOR
      NHAT(1,I)=-SLATG*COSLON
      NHAT(2,I)=-SLATG*SINLON
      NHAT(3,I)= CLATG
C ...LOCAL VERTICAL
      ZHAT(1,I)=CLATG*COSLON
      ZHAT(2,I)=CLATG*SINLON
      ZHAT(3,I)=SLATG
C ...EAST VECTOR
      EHAT(1,I)=-SINLON
      EHAT(2,I)= COSLON
      EHAT(3,I)=0.0D0
  100 CONTINUE
      RETURN
      END
```

SQUA 112
SQUA 113
SQUA 114
SQUA 115
SQUA 116
SQUA 117
SQUA 118
SQUA 119
SQUA 120
SQUA 121
SQUA 122
SQUA 123
SQUA 124
SQUA 125
SQUA 126
SQUA 127
SQUA 128
SQUA 129
SQUA 130
SQUA 131
SQUA 132
SQUA 133
SQUA 134
SQUA 135
SQUA 136
SQUA 137
SQUA 138
SQUA 139
SQUA 140
SQUA 141
SQUA 142
SQUA 143
SQUA 144
SQUA 145
SQUA 146
SQUA 147
SQUA 148
SQUA 149
SQUA 150
SQUA 151
SQUA 152


ENTRY
CALCULATE CONSTANTS ON FIRST CALL
10
FIRST CALL
YES
20
NO
PLHOUT
CONVERT ADJUSTED XYZ TO ϕλh
OUTRAD
CONVERT ϕλ TO DEGREES, MINUTES AND SECONDS
30


20
DINRAD
CONVERT ϕλ TO RADIANS
30
CALCULATE SINE AND COSINE OF LATITUDE AND LONGITUDE
FIRST CALL
NO
40
YES
CALCULATE STATION XYZ
A



STAINF

DESCRIPTION

Subroutine STAINF computes the statistical information derived from the measurement residuals. These statistics are primarily for printout purposes, being composed of those statistics in the residual summary printout. In addition, STAINF computes the weighted RMS considering the degrees of freedom removed due to the regression.

There are four entries; their function in the order of computation is to

1) Initialize (zero all storage areas),

2) Sum weighted and unweighted measurements for each type of statistic,

3) Compute statistics for each station by measurement type, and

4) Compute statistics for all weighted measurements either in the arc or in the entire run.

NAME STAINF

ENTRY POINT	PURPOSE
STAIF1	INITIALIZATION
STAIF2	TO CORRECT STATISTICAL INFORMATION FOR ELECTRONIC BIAS COMPENSATION
STAINF	TO COMPUTE STATISTICAL INFORMATION

CALLING SEQUENCE CALL STAIF1(ASUM,NSUM,NSTA,NSAT,CSUM,MSUM,NBASE, LSUM)

SYMBOL	TYPE	DESCRIPTION
ASUM (8,4,NSTA,NSAT)	R	INPUT - SUMMING ARRAYS FOR PARTICULAR STATIONS, SATELLITES, AND MEASUREMENT TYPES
NSUM (3,4,NSTA,NSAT)	I	INPUT - NUMBER OF RESIDUALS AND WEIGHTED RESIDUAL RATIOS FOR PARTICULAR STATIONS, SATELLITES, AND MEASUREMENT TYPES
NSTA	I	INPUT - NUMBER OF TRACKING STATIONS
NSAT	I	INPUT - NUMBER OF SATELLITES
CSUM (8,2,NBASE,1)		INPUT - SCRATCH SPACE FOR SUMMING STATISTICAL INFORMATION
MSUM (3,2,NBASE,1)	I	INPUT - SCRATCH SPACE FOR SUMMING STATISTICAL INFORMATION
NBASE	I	INPUT - NUMBER OF BASE LINES ASSOCIATED WITH MEASUREMENTS INVOLVING TWO STATIONS
LSUM (NBASE,1)	I	INPUT - SCRATCH SPACE FOR SUMMING STATISTICAL INFORMATION

CALLING SEQUENCE CALL STAIF2(ISTAE,MTPE,SIGE,EBIAS,NEPASS,NOSAT)

SYMBOL	TYPE	DESCRIPTION
ISTAE	I	INPUT - STATION NUMBER USED IN COMPENSATING FOR ELECTRONIC BIAS
MTPE	I	INPUT - MEASUREMENT TYPE FOR ELECTRONIC BIAS COMPENSATION
SIGE	R	INPUT - SIGMA FOR ELECTRONIC BIAS COMPENSATION
EBIAS	R	INPUT - BIAS WHICH IS COMPENSATED FOR
NEPASS	I	INPUT - NUMBER OF MEASUREMENTS IN THE PASS FOR ELECTRONIC BIAS COMPENSATION

NOSAT	I	INPUT - SATELLITE NUMBER FOR ELECTRONIC BIAS COMPENSATION

CALLING SEQUENCE CALL STAINF(ITYPE,ISTA,MTYPE,SIG,RESID,LINNER, ISAT,ISTA2)

SYMBOL	TYPE	DESCRIPTION
ITYPE	I	INPUT - DETERMINES IF INITIALIZATION, ADDITION, STATION SUMMARY, OR TOTAL SUMMARY IS REQUESTED
ISTA	I	INPUT - INTERNAL STATION NUMBER
MTYPE	I	INPUT - MEASUREMENT TYPE
SIG	R	INPUT - SIGMA
RESID	R	INPUT - RESIDUAL
LINNER	L	INPUT - .TRUE. - LAST INNER ITERATION
ISAT	I	INPUT - SATELLITE NUMBER
IST42	I	INPUT - SECOND STATION WHEN MEASUREMENT INVOLVES TWO STATIONS

SUBROUTINES USED SMSTAT CLEAR RMSCMP

COMMON BLOCKS CSTINF CSTAT

INPUT FILES NONE

OUTPUT FILES NONE

REFERENCES "GEODYN SYSTEMS DESCRIPTION" VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE STAINF(ASUM,NSUM,NSTA,NSAT,CSUM,WSUM,NBASE,LSUM)          STAI  80
      *INTEGER*2 ISAT                                                      STAI  90
      DIMENSION ASUM(6,4,NSTA,NSAT),NSUM(3,4,NSTA,NSAT),                   STAI 100
     *   CSUM(6,2,NBASE,1),WSUM(3,2,NBASE,1),LSUM(NBASE,1)                 STAI 101
      COMMON/CSTINF/MEASNO(4),NOBS(4),RMSMEAN(4),BIAS(4),RMS(4),           STAI 102
     *   MEAST(4),WTMEAN(4),EMSNTC(4),WTRMS(4),TYPRMS(3),NOTYPE(2,30),     STAI 103
     *   RSUM(3,12),RMSALL(3),NOALL(3),NOWTRS,JBASE                        STAI 104
      COMMON/CSTAT/RESID1,SIG1,NMTOT,WTSUMT                                STAI 105
      N=NSTA*NSAT                                                          STAI 106
      LBASE=N+NSE                                                          STAI 107
      LST=N+NSTA                                                           STAI 108
      RETURN                                                               STAI 109
C COMPENSATE FOR ELECTRONIC BIAS REMOVAL                                   STAI 110
      ENTRY STAIF2(IST42,MTYP,SIGE,EBIAS,NBPASS,NTSAT)                     STAI 111
```

```
      ISAT=0                                                            STAI 112
  500 ISAT=ISAT+1                                                       STAI 113
      DO 600 I=1,4                                                      STAI 114
      MNO=NSUM(3,I,ISTAE,ISAT)                                          STAI 115
      IF(MNO-MTYPE) 600,525,600                                         STAI 116
  525 ASUM(1,I,ISTAE,ISAT)=ASUM(1,I,ISTAE,ISAT)                         STAI 117
     *                    -EBIAS*FLOAT(NEPASS)                         STAI 118
      ASUM(3,I,ISTAE,ISAT)=ASUM(3,I,ISTAE,ISAT)                         STAI 119
     *                     -FLOAT(NEPASS)*EBIAS**2                      STAI 120
      ASUM(7,I,ISTAE,ISAT)=0.                                           STAI 121
      RATIO=EBIAS/SIGE                                                  STAI 122
      R2N=FLOAT(NEPASS)*RATIO                                           STAI 123
      ASUM(2,I,ISTAE,ISAT)=ASUM(2,I,ISTAE,ISAT)-R2N                     STAI 124
      R2N=R2N*RATIO                                                     STAI 125
      ASUM(4,I,ISTAE,ISAT)=ASUM(4,I,ISTAE,ISAT)-R2N                     STAI 126
      WTSUMT=WTSUMT-R2N                                                 STAI 127
      ASUM(3,I,ISTAE,ISAT)=0.                                           STAI 128
  600 CONTINUE                                                          STAI 129
      IF(ISAT.LT.NCSAT) GO TO 500                                       STAI 130
      RETURN                                                            STAI 131
      ENTRY STAINF(ITYPE,ISTA,MTYPE,SIG,RESID,LINNER,ISAT,ISTA2)        STAI 132
      LOGICAL LINNER                                                    STAI 133
      SIG1=SIG                                                          STAI 134
      RESID1=RESID                                                      STAI 135
C DETERMINE IF INITIALIZATION, ADDITION, STATION SUMMARY, OR TOTAL      STAI 136
C SUMMARY                                                               STAI 137
      GO TO (1,101,201,401),ITYPE                                       STAI 138
C ZERO STORAGE AREAS                                                    STAI 139
    1 CALL CLEAR(ASUM,32,N)                                             STAI 140
      CALL CLEAR(NSUM,32,N)                                             STAI 141
      CALL CLEAR(TYPRMS,146,1)                                          STAI 142
      IF(LBASE.LE.0) GO TO 10                                           STAI 143
      CALL CLEAR(CSUM,16,LBASE*N)                                       STAI 144
      CALL CLEAR(MSUM,6,LBASE*N)                                        STAI 145
      CALL CLEAR(LSUM,LBASE,N)                                          STAI 146
   10 NMTOT=0                                                           STAI 147
      WTSUMT=0.                                                         STAI 148
      IF(ISTA.NE.0) RETURN                                              STAI 149
      NMALL=0                                                           STAI 150
      WTSUMA=0.                                                         STAI 151
      NDEGF=1                                                           STAI 152
      CALL CLEAR(RMSALL,30,2)                                           STAI 153
      RETURN                                                            STAI 154
C SUM STATISTICS                                                        STAI 155
  101 IF(MTYPE.GT.14) GO TO 201                                         STAI 156
C TYPES 1 - 14                                                          STAI 157
      DO 200 I=1,4                                                      STAI 158
      MNO=NSUM(3,I,ISTA,ISAT)                                           STAI 159
      IF(MNO.GT.0) IF(MNO-MTYPE) 200,125,200                            STAI 160
      NSUM(3,I,ISTA,ISAT)=MTYPE                                         STAI 161
  125 CALL SMSTAT(ASUM(1,I,ISTA,ISAT),NSUM(1,I,ISTA,ISAT))              STAI 162
      RETURN                                                            STAI 163
  200 CONTINUE                                                          STAI 164
      RETURN                                                            STAI 165
  201 IF(MTYPE.GT.26) GO TO 210                                         STAI 166
C TYPES 15 - 26                                                         STAI 167
```

```
      NTYPE=MTYPE-14                                                    STAI 168
      CALL SMSTAT(PSUM(1,NTYPE),NOTYPE(1,MTYPE))                        STAI 169
      RETURN                                                            STAI 170
C TYPES 27 - 31                                                         STAI 171
  210 IND=(ISAT-1)*LSTA+ISTA                                            STAI 172
      DO 275 J=1,LPASE                                                  STAI 173
      KSTA=LSUM(J,IND)                                                  STAI 174
      IF(KSTA.GT.0) IF(KSTA-ISTA2) 275,225,275                          STAI 175
      LSUM(J,IND)=ISTA2                                                 STAI 176
  225 DO 270 I=1,2                                                      STAI 177
      MNO=MSUM(3,I,J,IND)                                               STAI 178
      IF(MNO.GT.0) IF(MNO-MTYPE) 270,250,270                            STAI 179
      MSUM(3,I,J,IND)=MTYPE                                             STAI 180
  250 CALL SMSTAT(CSUM(1,I,J,IND),MSUM(1,I,J,IND))                      STAI 181
      RETURN                                                            STAI 182
  270 CONTINUE                                                          STAI 183
  275 CONTINUE                                                          STAI 184
      RETURN                                                            STAI 185
C STATION SUMMARY                                                       STAI 186
  301 IF(ISTA.EQ.0) GO TO 351                                           STAI 187
      IF(ISTA2.GT.0) GO TO 325                                          STAI 188
C SINGLE STATION MEASUREMENTS                                           STAI 189
      CALL RNSCMP(NSUM(1,1,ISTA,ISAT),ASUM(1,1,ISTA,ISAT),4)            STAI 190
      RETURN                                                            STAI 191
C TWO STATION MEASUREMENTS                                              STAI 192
  325 IND=(ISAT-1)*LSTA+ISTA                                            STAI 193
      DO 350 I=1,LPASE                                                  STAI 194
      KSTA=LSUM(I,IND)                                                  STAI 195
      IF(KSTA.NE.ISTA2) GO TO 350                                       STAI 196
      CALL RNSCMP(MSUM(1,1,I,IND),CSUM(1,1,I,IND),2)                    STAI 197
      NOBS(3)=0                                                         STAI 198
      RETURN                                                            STAI 199
  350 CONTINUE                                                          STAI 200
      NOBS(1)=0                                                         STAI 201
      RETURN                                                            STAI 202
C MEASUREMENTS NOT INVOLVING STATIONS                                   STAI 203
  351 DO 400 I=1,12                                                     STAI 204
      J=I+14                                                            STAI 205
      K=NOTYPE(1,J)                                                     STAI 206
      IF(K.EQ.0) GO TO 400                                              STAI 207
      XN=K                                                              STAI 208
      IF(K.LT.10) GO TO 395                                             STAI 209
      RSUM(5,I)=(RSUM(3,I)-RSUM(1,I)**2/XN)/RSUM(5,I)                   STAI 210
      HSUM(5,I)=(2.*RSUM(5,I)-1.)/SQRT((XN-2.)/(XN**2-1.))             STAI 211
      RSUM(3,I)=SQRT(RSUM(3,I)/(XN-1.))                                 STAI 212
  395 RSUM(1,I)=RSUM(1,I)/XN                                            STAI 213
      K=NOTYPE(2,J)                                                     STAI 214
      IF(K.EQ.0) GO TO 400                                              STAI 215
      XN=K                                                              STAI 216
      TYPRMS(I)=RSUM(4,I)                                               STAI 217
      IF(K.LT.10) GO TO 395                                             STAI 218
      HSUM(6,I)=(RSUM(6,I)-RSUM(2,I)**2/XN)/RSUM(6,I)                   STAI 219
      RSUM(4,I)=(2.*RSUM(6,I)-1.)/SQRT((XN-2.)/(XN**2-1.))             STAI 220
      HSUM(4,I)=SQRT(RSUM(4,I)/(XN-1.))                                 STAI 221
  395 RSUM(2,I)=RSUM(2,I)/XN                                            STAI 222
  400 CONTINUE                                                          STAI 223
```

```
      RETURN                                                            STAI 224
C TOTAL SUMMARY                                                         STAI 225
  401 IF(ISTA.EQ.0) GO TO 467                                           STAI 226
      MTYPE=NMTOT                                                       STAI 227
      IF(.NOT.LINNER) GO TO 410                                         STAI 228
      NMALL=NMALL+NMTOT                                                 STAI 229
      NODEGF=NODEGF+ISTA                                                STAI 230
      WTSUMA=WTSUMA+WTSUMT                                              STAI 231
  410 SIG=SQRT(WTSUMT/FLOAT(NMTOT-ISTA-1))                              STAI 232
      DO 450 I=1,30                                                     STAI 233
      IF(LINNER) NOALL(I)=NOALL(I)+NOTYPE(2,I)                          STAI 234
      IF(LINNER) RMSALL(I)=RMSALL(I)+TYPRMS(I)                          STAI 235
  450 IF(NOTYPE(2,I).NE.0) TYPRMS(I)=SQRT(TYPRMS(I)/FLOAT(NOTYPE(2,I))) STAI 236
      RETURN                                                            STAI 237
  460 NODEGF=NODEGF+MTYPE                                               STAI 238
      MTYPE=NMALL                                                       STAI 239
      SIG=SQRT(WTSUMA/FLOAT(NMALL-NODEGF))                              STAI 240
      DO 470 I=1,30                                                     STAI 241
  470 IF(NOALL(I).NE.0) TYPRMS(I)=SQRT(RMSALL(I)/FLOAT(NOALL(I)))       STAI 242
      RETURN                                                            STAI 243
      END                                                               STAI 244
      DIMENSION NOTYPE(2),SUM(8)                                        STAI 245
      COMMON/CSTAT/RESID,SIG,NMTOT,WTSUMT                               STAI 246
      IF(NOTYPE(1).EQ.0) SUM(7)=RESID                                   STAI 247
      NOTYPE(1)=NOTYPE(1)+1                                             STAI 248
      SUM(1)=SUM(1)+RESID                                               STAI 249
      SUM(3)=SUM(3)+RESID**2                                            STAI 250
      SUM(5)=SUM(5)+(RESID-SUM(7))**2                                   STAI 251
      SUM(7)=RESID                                                      STAI 252
      IF(SIG.NE.0.) RETURN                                              STAI 253
      R=RESID/SIG                                                       STAI 254
      R2=R**2                                                           STAI 255
      NMTOT=NMTOT+1                                                     STAI 256
      WTSUMT=WTSUMT+R2                                                  STAI 257
      IF(NOTYPE(2).EQ.0) SUM(8)=R                                       STAI 258
      NOTYPE(2)=NOTYPE(2)+1                                             STAI 259
      SUM(2)=SUM(2)+R                                                   STAI 260
      SUM(4)=SUM(4)+R2                                                  STAI 261
      SUM(6)=SUM(6)+(R-SUM(8))**2                                       STAI 262
      SUM(8)=R                                                          STAI 263
      RETURN                                                            STAI 264
      END                                                               STAI 265
```

STAINP

DESCRIPTION

STAINP reads tracking station positions from the GEODYN Input Cards. STAINP can read tracking stations in geodetic (latitude, longitude, height) or Cartesian coordinates.

STAINP assumes the station position to be in Cartesian coordinates. If, however, the station vector is considerably smaller in magnitude than one Earth radius STAINP considers the station position to be given in geodetic coordinates.

STAINP will not accept duplicate stations and prints an error message upon any such encounters.

This subroutine also loads the variable storage station arrays selecting station coordinates from the input values supplimented by station coordinate values stored in the block data subroutine STAPOS.

NAME STAINP

PURPOSE TO READ GEODYN TRACKING STATIONS AND CONVERT FROM CARTESIAN COORDINATES WHENEVER NECESSARY

CALLING SEQUENCE CALL STAINP(NSTA,KSTAND,STANAM,STALAT,STALON, HEIGHT,NSTARD,NSTEST,ESTAND)

SYMBOL	TYPE	DESCRIPTION
NSTA	I	INPUT - NUMBER OF STATIONS USED
KSTAND (1)	I*2	OUTPUT - STATION NUMBERS
STANAM (1)	DP	OUTPUT - STATION NAMES
STALAT (1)	DP	OUTPUT - STATION LATITUDES
STALON (1)	DP	OUTPUT - STATION LONGITUDES
HEIGHT (1)	DP	OUTPUT - STATION HEIGHTS
NSTARD	I	INPUT - NUMBER OF STATIONS THAT WERE READ FROM CARDS
NSTEST	I	INPUT - NUMBER OF STATIONS REQUESTING ADJUSTMENT
ESTAND	I*2	OUTPUT - MASTER STATION NUMBERS FOR ESTIMATED STATIONS

SUBROUTINES USED NUMBR2 ERROR DARCTN

COMMON BLOCKS CEPHEM STANUM STAPOS CONSTS INTRLK TPEBLK

INPUT FILES INTP - GEODYN INPUT CARDS

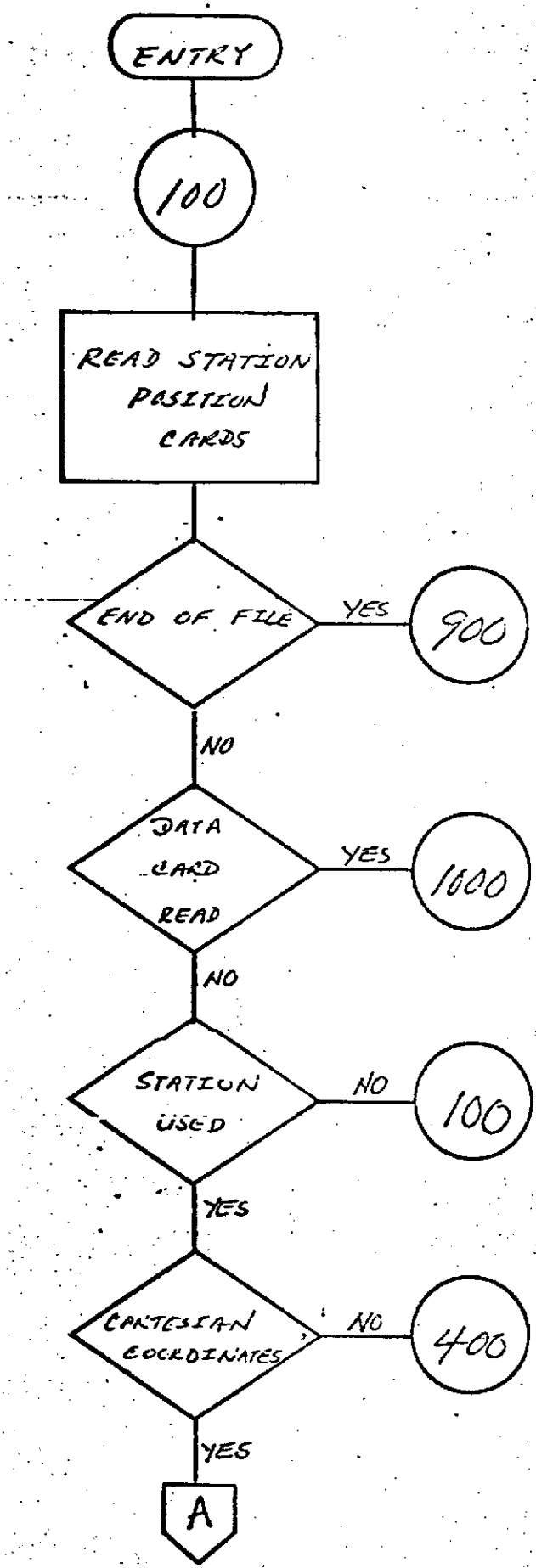
OUTPUT FILES NONE

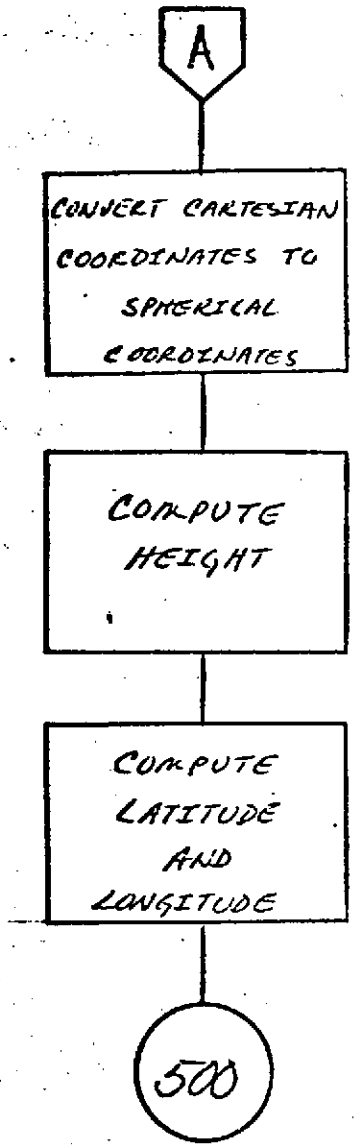
REFERENCES 'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION

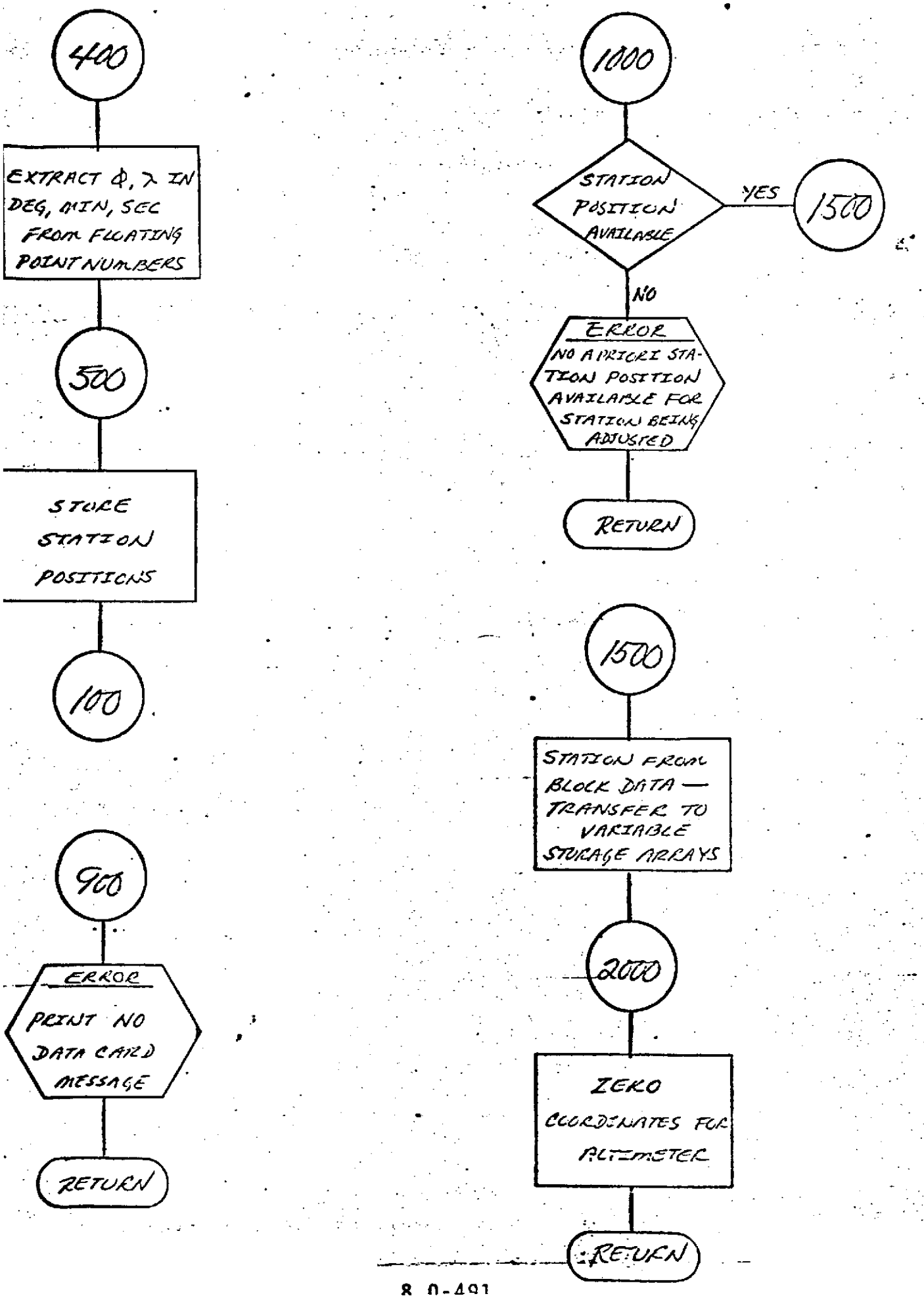
```
      SUBROUTINE STAINP(NSTA,KSTAND,STANAM,STALAT,STALON,HEIGHT,
     *   NSTARD,NSTEST,ESTAND)
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION KSTAND(1),STANAM(1),STALAT(1),STALON(1),HEIGHT(1),
     *   ESTAND(1),NAMES(8)
```

```
      LOGICAL*1 NAME8,NAME7                                                  STAI  56
      INTEGER*2 KSTAND,ISTAND,STANOS,ESTANO,ISTARD,ESTAND                    STAI  57
      INTEGER OUTP                                                           STAI  58
      DOUBLE PRECISION NAME,L                                                STAI  59
      COMMON/CEPHEM/DNAME(381),ISTARD(381),ESTANO(381),ISTAND(356)           STAI  60
      COMMON/STANUM/SNAME(280),STANOS(280),NOSTOR                            STAI  61
      COMMON/STAPOS/FLAT(280),RLON(280),H(280)                               STAI  62
      COMMON/CONSTS/PI,TWOPI,DRAD,RSEC                                       STAI  63
      COMMON/INTBLK/THDOT(4),AE,AESQ,FLAT,FS032(59)                          STAI  64
      COMMON/IOBLK/INTP,OUTP,IDUM(10)                                        STAI  65
      DATA DATA/6HDATA  /,STAPOS/6HSTAPOS/                                   STAI  66
      DATA ALTMTR/6HALTMTR/                                                  STAI  67
      DATA DELTA/1.00-3/                                                     STAI  68
      EQUIVALENCE (JSTAND,DSTAND),(NAME,NAME8(1)),(NAME7,NAME8(7))           STAI  69
      IF(NSTA.LE.0) RETURN                                                   STAI  70
      DO 10 I=1,NSTA                                                         STAI  71
   10 KSTAND(I)=0                                                            STAI  72
      IF(NSTARD.LE.0) GO TO 1000                                             STAI  73
      REWIND INTP                                                            STAI  74
      READ(INTP,5000)                                                        STAI  75
      ESQ=(1.0D0-FLAT)**2                                                    STAI  76
      ESQ=1.0D0-ESQ                                                          STAI  77
   20 READ(INTP,10005) NAME                                                  STAI  78
      IF(NAME.NE.STAPOS) GO TO 20                                            STAI  79
C READ STATION POSITION CARDS                                                STAI  80
  100 READ(INTP,10005,END=900) NAME,JSTAND,P,L,HT,NAME7                      STAI  81
      IF(NAME.EQ.DATA) GO TO 1000                                            STAI  82
      ISN=NUMBR2(JSTAND,ISTAND,NSTA)                                         STAI  83
C SKIP CARD IF STATION NOT USED                                              STAI  84
      IF(ISN.LE.0) GO TO 100                                                 STAI  85
      IF(KSTAND(ISN).LE.0) GO TO 150                                         STAI  86
      CALL ERROR(3,DSTAND)                                                   STAI  87
      GO TO 100                                                              STAI  88
C CHECK COORDINATE SYSTEMS                                                   STAI  89
  150 XYSQ=P*P+L*L                                                           STAI  90
      RSQ=XYSQ+HT*HT                                                         STAI  91
      IF(RSQ.LT.3.5D13) GO TO 400                                            STAI  92
C CONVERT CARTESIAN TO SPHERICAL COORDINATES                                 STAI  93
      T=ESQ*HT                                                               STAI  94
C ...HEIGHT                                                                  STAI  95
      DO 200 J=1,25                                                          STAI  96
      ZT=HT+T                                                                STAI  97
      H1=DSQRT(XYSQ+ZT**2)                                                   STAI  98
      SINPHI=ZT/H1                                                           STAI  99
      ESQSP=ESQ*SINPHI                                                       STAI 100
      H2=AE/DSQRT(1.0D0-ESQSP*SINPHI)                                        STAI 101
      T1=H2*ESQSP                                                            STAI 102
      IF(DABS(T1-T).LT.DELTA) GO TO 300                                      STAI 103
  200 T=T1                                                                   STAI 104
  300 HT=H1-H2                                                               STAI 105
      RTXYSQ=DSQRT(XYSQ)                                                     STAI 106
      RSQ=DATAN2(ZT,RTXYSQ)                                                  STAI 107
C ...LONGITUDE                                                               STAI 108
      L=DARCTN(L,P)                                                          STAI 109
C ...LATITUDE                                                                STAI 110
      P=RSQ                                                                  STAI 111
```

```
      GO TO 500                                                          STAI 112
  400 JLATD=P*1.0D-4                                                     STAI 113
      P=P-JLATD*1.0D+4                                                   STAI 114
      JLATM=P*1.0D-2                                                     STAI 115
      TSLAT=P-JLATM*1.0D+2                                               STAI 116
      JLOND=L*1.0D-4                                                     STAI 117
      L=L-JLOND*1.0D+4                                                   STAI 118
      JLONM=L*1.0D-2                                                     STAI 119
      TSLON=L-JLONM*1.0D+2                                               STAI 120
      P=(DFLOAT(JLATD)+DFLOAT(JLATM)/6.0D1+TSLAT/3.6D3)*DRAD             STAI 121
      L=(DFLOAT(JLOND)+DFLOAT(JLONM)/6.0D1+TSLON/3.6D3)*DRAD             STAI 122
C STORE STATION POSITION IN VARIABLE STORAGE ARRAYS                      STAI 123
  500 KSTANO(ISN)=JSTANO                                                 STAI 124
      STANAM(ISN)=NAME                                                   STAI 125
      STALAT(ISN)=P                                                      STAI 126
      STALON(ISN)=L                                                      STAI 127
      HEIGHT(ISN)=HT                                                     STAI 128
      IF(ISN.LE.NSTEST) ESTANO(ISN)=FSTANO(ISN)                          STAI 129
      GO TO 100                                                          STAI 130
  900 CALL ERROR(6,DSTANO)                                               STAI 131
      RETURN                                                             STAI 132
 1000 DO 2000 I=1,NSTA                                                   STAI 133
      IF(KSTANO(I).NE.0) GO TO 2000                                      STAI 134
      JSTANO=ISTANO(I)                                                   STAI 135
      ISN=NUMBR2(JSTANO,STANOS,NOSTOR)                                   STAI 136
      IF(ISN.GT.0) GO TO 1500                                            STAI 137
      IF(JSTANO.NE.-4) GO TO 1200                                        STAI 138
      KSTANO(I)=-4                                                       STAI 139
      STANAM(I)=ALTMTR                                                   STAI 140
      STALAT(I)=0.0D0                                                    STAI 141
      STALON(I)=0.0D0                                                    STAI 142
      HEIGHT(I)=0.0D0                                                    STAI 143
      GO TO 2000                                                         STAI 144
 1200 CALL ERROR(9,DSTANO)                                               STAI 145
      RETURN                                                             STAI 146
C IF STATION FROM BLOCK DATA THEN TRANSFER TO VARIABLE STORAGE ARRAYS    STAI 147
 1500 KSTANO(I)=JSTANO                                                   STAI 148
      STANAM(I)=SNAME(ISN)                                               STAI 149
      STALAT(I)=RLAT(ISN)                                                STAI 150
      STALON(I)=RLON(ISN)                                                STAI 151
      HEIGHT(I)=H(ISN)                                                   STAI 152
      IF(I.LE.NSTEST) ESTANO(I)=FSTANO(I)                                STAI 153
 2000 CONTINUE                                                           STAI 154
C ZERO COORDINATES FOR ALTIMETER                                         STAI 155
      RETURN                                                             STAI 156
 5000 FORMAT(1X/1X/1X)                                                   STAI 157
10005 FORMAT(A5,I4,3D15.8,A1)                                            STAI 158
      END                                                                STAI 159
```


ENTRY
100
READ STATION POSITION CARDS
END OF FILE
YES
900
NO
DATA CARD READ
YES
1000
NO
STATION USED
NO
100
YES
CARTESIAN COORDINATES
NO
400
YES
A


A
CONVERT CARTESIAN COORDINATES TO SPHERICAL COORDINATES
COMPUTE HEIGHT
COMPUTE LATITUDE AND LONGITUDE
500


400
EXTRACT φ, λ IN DEG, MIN, SEC FROM FLOATING POINT NUMBERS
500
STORE STATION POSITIONS
100
900
ERROR
PRINT NO DATA CARD MESSAGE
RETURN
1000
STATION POSITION AVAILABLE
YES
1500
NO
ERROR
NO A PRIORI STATION POSITION AVAILABLE FOR STATION BEING ADJUSTED
RETURN
1500
STATION FROM BLOCK DATA — TRANSFER TO VARIABLE STORAGE ARRAYS
2000
ZERO COORDINATES FOR ALTIMETER
RETURN

NAME SMSTAT

PURPOSE TO SUM STATISTICS

CALLING SEQUENCE CALL SMSTAT(SUM,NOTYPE)

SYMBOL	TYPE	DESCRIPTION
SUM (8)	R	INPUT & OUTPUT - ARRAY THAT INFORMATION GETS SUMMED INTO
NOTYPE (2)	I	INPUT & OUTPUT - ARRAY CONTAINING NUMBER OF MEASUREMENTS

SUBROUTINES USED NONE

COMMON BLOCKS CSTAT

INPUT FILES NONE

OUTPUT FILES NONE

REFERENCES 'GEODYN SYSTEMS DESCRIPTION'
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE SMSTAT(SUM,NOTYPE)                                     STAT 273
      DIMENSION NOTYPE(2),SUM(8)                                        STAT 274
      COMMON/CSTAT /RESID,SIG,NMTOT,WTSUMT                              STAT 275
      IF(NOTYPE(1).EQ.0) SUM(7)=RESID                                   STAT 276
      NOTYPE(1)=NOTYPE(1)+1                                             STAT 277
      SUM(1)=SUM(1)+RESID                                               STAT 278
      SUM(3)=SUM(3)+RESID**2                                            STAT 279
      SUM(5)=SUM(5)+(RESID-SUM(7))**2                                   STAT 280
      SUM(7)=RESID                                                      STAT 281
      IF(SIG.LT.0.) RETURN                                              STAT 282
      R=RESID/SIG                                                       STAT 283
      R2=R**2                                                           STAT 284
      NMTOT=NMTOT+1                                                     STAT 285
      WTSUMT=WTSUMT+R2                                                  STAT 286
      IF(NOTYPE(2).EQ.0) SUM(8)=R                                       STAT 287
      NOTYPE(2)=NOTYPE(2)+1                                             STAT 288
      SUM(2)=SUM(2)+R                                                   STAT 289
      SUM(4)=SUM(4)+R2                                                  STAT 290
      SUM(6)=SUM(6)+(R-SUM(8))**2                                       STAT 291
      SUM(8)=R                                                          STAT 292
      RETURN                                                            STAT 293
      END                                                               STAT 294
```

```
  395 BSUM(2,I)=BSUM62,I)/XN                                              STAT  [illegible]
  400 CONTINUE                                                            STAT  [illegible]
      RETURN                                                              STAT  [illegible]
C TOTAL SUMMARY                                                           STAT  [illegible]
  401 IF(ISTA.EQ.0) GO TO 460                                             STAT  [illegible]
      MTYPE=NMTOT                                                         STAT  [illegible]
      IF(.NOT.LINNER) GO TO 410                                           STAT  [illegible]
      NMALL=NMALL+NMTOT                                                   STAT  [illegible]
      NODEGF=NODEGF+ISTA                                                  STAT  [illegible]
      WTSUMA=WTSUMA+WTSUMT                                                STAT  [illegible]
  410 SIG=SQRT(WTSUMT/FLOAT(NMTOT-ISTA-1))                                STAT  [illegible]
      DO 450 I=1,30                                                       STAT  [illegible]
      IF(LINNER) NOALL(I)=NOALL(I)+NOTYPE(2,I)                            STAT  [illegible]
      IF(LINNER) RMSALL(I)=RMSALL(I)+TYPRMS(I)                            STAT  [illegible]
  450 IF(NOTYPE(2,I).NE.0) TYPRMS(I)=SQRT(TYPRMS(I)/FLOAT(NOTYPE(2,I)))   STAT  [illegible]
      RETURN                                                              STAT  [illegible]
  460 NODEGF=NODEGF+MTYPE                                                 STAT  [illegible]
      MTYPE=NMALL                                                         STAT  [illegible]
      SIG=SQRT(WTSUMA/FLOAT(NMALL-NODEGF))                                STAT  [illegible]
      DO 470 I=1,30                                                       STAT  [illegible]
  470 IF(NOALL(I).NE.0) TYPRMS(I)=SQRT(RMSALL(I)/FLOAT(NOALL(I)))         STAT  [illegible]
      RETURN                                                              STAT  [illegible]
      END                                                                 STAT  [illegible]
```

NAME STAPOS

PURPOSE BLOCK DATA STORAGE OF STATION POSITIONS
UNITS FOR COORDINATES
LAT - RADIANS
LON - RADIANS
H - METERS

COMMON BLOCKS STAPOS STANUM

```
      BLOCK DATA                                                        STAP  14
      DOUBLE PRECISION LAT,LON,HT,NAME                                  STAP  15
      INTEGER*2 NUMBER                                                  STAP  16
      COMMON/STAPOS/LAT(280),LON(280),HT(280)                           STAP  17
      COMMON/STANUM/NAME(280),NUMBER(280),NSTOR                         STAP  18
      DATA NSTOR/254/                                                   STAP  19
      DOUBLE PRECISION NAME 1                                           STAP  20
      DIMENSION NAME 1(114)                                             STAP  21
      EQUIVALENCE (NAME  (  1),NAME 1(1))                               STAP  22
      DATA NAME 1/                                                      STAP  23
     * 7HMILL3   ,7HGBM3   ,7HRDA3   ,7HANG3   ,7HCY13   ,7HACN3   ,   STAP  24
     * 7HMAD3   ,7HCRO3   ,7HGWM3   ,7HHSKB   ,7HHAW3   ,7HGRSB   ,   STAP  25
     * 7HGYM3   ,7HTEX3   ,7HGTC3   ,7HPIDD   ,7HECHO   ,7HVEND   ,   STAP  26
     * 7HMAPO   ,7HWOMD   ,7HTIDD   ,7HJOBD   ,7HRDBD   ,7HCCRD   ,   STAP  27
     * 7HMRTD   ,7HKOUROU ,7HRPOINT ,7HETMYR6 ,7HQUITO6 ,7HLIMAP6 ,   STAP  28
     * 7HSNTAG6 ,7HNEWFL6 ,7HCOLEGE ,7HGFORKS ,7HWNKFL6 ,7HJOBUR6 ,   STAP  29
     * 7HMOJAVE ,7HOOMERA ,7HULASK6 ,7H1HPOIN ,7H1ETMYR ,7HMADGA6 ,   STAP  30
     * 7H1OOMER ,7H1QUITO ,7H1LIMAP ,7H1SATAG ,7H1MOJAV ,7H1JOBUP ,   STAP  31
     * 7H1NEWFL ,7H1COLEG ,7H1GFORK ,7H1WNKFL ,7H1ULASK ,7H1ROSMN ,   STAP  32
     * 7H1OPORL ,7H1ROSMA ,7H1TANAN ,7HORORA6 ,7HMADGARS,7HROSPANS,   STAP  33
     * 7HSNTAGRS,7HALASKRS,7HCARVONS,7HMADGARV,7HROSRANV,7HSNTAGRV,   STAP  34
     * 7HALASKRV,7HCARVONV,7HANTIGA ,7HGRNVLE ,7HGRVILL ,7HUSAFAC ,   STAP  35
     * 7HREDFRD ,7HSEMMES ,7HSWANIS ,7HGROTRK ,7HCURACO ,7HTRNDAD ,   STAP  36
     * 7HGRANFK ,7HTWINOK ,7HROTHGR ,7HATHNGR ,7HTORRSP ,7HCHOFUJ ,   STAP  37
     * 7HKINDLY ,7HEDWADS ,7HHUNTER ,7HJUPRAF ,7HABERDN ,7HHOMEST ,   STAP  38
     * 7HCHYWYM ,7HETRGR6 ,7HETPCAK ,7HETRPRE ,7HETRPAT ,7HETRANT ,   STAP  39
     * 7HETRGRT ,7HETPMRT ,7HETRGR8 ,7HWSH122 ,7HWSC113 ,7HWSH123 ,   STAP  40
     * 7HMSS127 ,7HWSP124 ,7HWST125 ,7HWTRPRS ,7HWTRTR2 ,7HWTRTR1 ,   STAP  41
     * 7HMTRPRO ,7HWTRVAN ,7HEGLINF ,7HEGLINF ,7HPMRSN5 ,7HPMRSN6 /   STAP  42
      DOUBLE PRECISION NAME 2                                           STAP  43
      DIMENSION NAME 2(114)                                             STAP  44
      EQUIVALENCE (NAME  (115),NAME 2(1))                               STAP  45
      DATA NAME 2/                                                      STAP  46
     * 7HPMRVR1 ,7HPMRMR2 ,7HPMRRM1 ,7HPMRRM2 ,7HPMRSN2 ,7HPMRSN3 ,   STAP  47
     * 7HPMRSN4 ,7HPVSRM3 ,7HPMRRM4 ,7HPMRRK1 ,7HEDAFB3 ,7HNELHAR ,   STAP  48
     * 7HNELYNV ,7HNWIW2A ,7HNWIW2B ,7HNWIE3A ,7HNWIE3B ,7HNBER34 ,   STAP  49
     * 7HNTANAN ,7HWTRKAU ,7HNBERO5 ,7HNCARNV ,7HNWALI8 ,7HNWALI3 ,   STAP  50
     * 7HM0093R ,7HRAFAR1 ,7HRAFAB5 ,7HHOURTN ,7HSANSAL ,7HPTARGO ,   STAP  51
     * 7HHERNDN ,7HQUICAL ,7HLARSON ,7HKRGTON ,7HGREENV ,7HMAUIHA ,   STAP  52
     * 7HMALSEC ,7HFTWART ,7HHNTAFB ,7HHOMEFL ,7HRFLTVL ,7HVOLOGD ,   STAP  53
     * 7HEREVAN ,7HKIEVAA ,7HKRASNO ,7HNOVOSI ,7HRIAZAN ,7HTARTU2 ,   STAP  54
     * 7HTASHK1 ,7HASTRMO ,7HLYNNLK ,7HTIMENS ,7H1UNDAK ,7H1EDINB ,   STAP  55
```

```
     *  7H1COLBA ,7H1BERMD ,7H1PURID ,7H1GSFCO ,7H1GSFCP ,7H1CKVLE ,     STAP  56
     *  7H1DENVR ,7H1JUM24 ,7H1JUM40 ,7H1JUPC1 ,7H1JUPC4 ,7H1SUDBR ,     STAP  57
     *  7H1JAMAC ,7H1GSFCN ,7HWALMOT ,7H1CARVN ,7HGODLAS ,7HRDSLAS ,     STAP  58
     *  7HWALLAS ,7HYORLAS ,7HCRWLAS ,7HHOWLAS ,7HHOWLA2 ,7HMOBLA2 ,     STAP  59
     *  7HSENLAS ,7HSFELLAS ,7HORGLAS ,7HOLILAS ,7HARELAS ,7HHOPLAS ,     STAP  60
     *  7HNATLAS ,7HGRELAS ,7HBRNSCH ,7HDELFTH ,7HZIMWLD ,7HMALVRN ,     STAP  61
     *  7HROYORS ,7HATHENS ,7HHAUTEP ,7HNICEFR ,7HMICLAS ,7HSALLAS ,     STAP  62
     *  7HMUDONI ,7HEDINBH ,7HMUNICH ,7HFRANKF ,7HSANFLR ,7HHAUTLR ,     STAP  63
     *  7H1ORGAN ,7H1OLEAN ,7HWOOMER ,7H1SPAIN ,7H1TOKYO ,7H1NATAL ,     STAP  64
     *  7H1QUIPA ,7H1SIRAZ ,7H1CURAC ,7H1JUPTR ,7H1VILDO ,7H1MAUID /     STAP  65
      DOUBLE PRECISION NAME 3                                          STAP  66
      DIMENSION NAME 3( 26)                                            STAP  67
      EQUIVALENCE (NAME  (229),NAME 3(1))                              STAP  68
      DATA NAME 3/                                                     STAP  69
     *  7HDAKARO ,7HHOPKIN ,7HAUSBAK ,7HDODAIR ,7HDFZEIT ,7HNATALB ,     STAP  70
     *  7HCOMRIV ,7HBRAZIL ,7HJUPGED ,7HAGASSI ,7HZIMMER ,7HRIGALA ,     STAP  71
     *  7HUZHGOR ,7HGRFECE ,7HCALBAK ,7HCOLAKE ,7HOSLONR ,7HJOHNST ,     STAP  72
     *  7HMTJOHN ,7HSANVTO ,7HEDWAFB ,7HPOTDAM ,7HZVENIG ,7HHELSIK ,     STAP  73
     *  7HDAKAR  ,7HSANVIT /                                             STAP  74
      INTEGER*2 NUMB 1                                                 STAP  75
      DIMENSION NUMB 1(247)                                            STAP  76
      EQUIVALENCE (NUMBER(  1),NUMB 1(1))                              STAP  77
      DATA NUMB 1/                                                     STAP  78
     *   1,   2,   3,   4,   5,   6,   7,   8,   9,  10,  11,  12,  13, STAP  79
     *  14,  15, 511, 512, 513, 514, 541, 542, 551, 561, 562, 571,1000, STAP  80
     *1001,1003,1005,1006,1008,1012,1013,1014,1015,1016,1017,1018,1019, STAP  81
     *1021,1022,1023,1024,1026,1026,1028,1030,1031,1032,1033,1034,1035, STAP  82
     *1036,1037,1038,1040,1040,1121,1123,1126,1127,1128,1132,1622,1626, STAP  83
     *1627,1629,1652,3106,3333,3334,3400,3401,3402,3404,3405,3406,3407, STAP  84
     *3451,3452,3453,3463,3464,3465,3471,3472,3648,3649,3657,3861,3902, STAP  85
     *4040,4041,4050,4060,4061,4081,4082,4083,4142,4143,4144,4145,4146, STAP  86
     *4151,4240,4241,4242,4260,4280,4340,4341,4400,4401,4402,4403,4440, STAP  87
     *4441,4442,4443,4444,4445,4446,4450,4540,4610,4690,4732,4733,4734, STAP  88
     *4735,4740,4741,4742,4760,4761,4840,4860,4946,4948,4949,4954,4956, STAP  89
     *4003,5001,5200,5201,5202,5333,5411,5508,5648,5649,5861,6002,6014, STAP  90
     *6018,6023,6027,6035,6042,6051,6052,6100,6107,6113,7034,7036,7037, STAP  91
     *7039,7040,7042,7043,7044,7045,7071,7072,7073,7074,7075,7076,7077, STAP  92
     *7078,7079,7050,7051,7052,7053,7054,7055,7056,7057,7058,7804,7901, STAP  93
     *7902,7907,7921,7929,7930,8004,8009,8010,8011,8013,8014,8015,8019, STAP  94
     *8021,8022,8030,8031,8032,8033,7804,7815,9001,9002,9003,9004,9005, STAP  95
     *9006,9007,9008,9009,9010,9011,9012,9020,9021,9023,9025,9028,9029, STAP  96
     *9031,9039,9049,9050,9056,9074,9077,9091,9113,9114,9115,9117,9119/ STAP  97
      INTEGER*2 NUMB 2                                                 STAP  98
      DIMENSION NUMB 2(  7)                                            STAP  99
      EQUIVALENCE (NUMBER(248),NUMB 2(1))                              STAP 100
      DATA NUMB 2/                                                     STAP 101
     *9120,9425,9429,9430,9435,9020,9120/                              STAP 102
      DOUBLE PRECISION LAT  1                                          STAP 103
      DIMENSION LAT  1( 76)                                            STAP 104
      EQUIVALENCE (LAT   (  1),LAT  1(1))                              STAP 105
      DATA LAT  1/                                                     STAP 106
     * 0.4175801100, 0.4648276700, 0.5646344700, 0.3059950300,         STAP 107
     * 0.4446773600,-0.1384472900, 0.7160812400,-0.4347089500,         STAP 108
     * 0.2333114700,-0.6210474400, 0.3961737700, 0.6168287300,         STAP 109
     * 0.4980460300, 0.4826454000, 0.6806514000, 0.6176630500,         STAP 110
     * 0.6160736300, 0.6151878900, 0.6182981100,-0.5477174200,         STAP 111
```

```
     *-0.6178622100,-0.4518644800, 0.7056151400, 0.7060395900,      STAP 112
     * 0.4970598800, 0.0916487700, 0.6707387000, 0.4633516600,      STAP 113
     *-0.0108712000,-0.2055467400,-0.5785708000, 0.8332457200,      STAP 114
     * 1.1322234700, 0.8381533100, 0.8979059500,-0.4517563700,      STAP 115
     * 0.6166243100,-0.5478679400, 1.1340664100, 0.6707337000,      STAP 116
     * 0.4633516600,-0.3317725300,-0.5478679600,-0.0108712000,      STAP 117
     *-0.2055457400,-0.5785708000, 0.6166243100,-0.4517563700,      STAP 118
     * 0.8332457200, 1.1322234700, 0.8381533100, 0.8979059500,      STAP 119
     * 1.1340664100, 0.6143911900,-0.6217865500, 0.6143912600,      STAP 120
     *-0.3317725300,-0.6217865500,-0.3219736000, 0.6142882800,      STAP 121
     *-0.5785774200, 1.1339744700,-0.4346424900,-0.3319872200,      STAP 122
     * 0.6142560600,-0.5786094200, 1.1339663000,-0.4346616900,      STAP 123
     * 0.2993004800, 0.5843433000, 0.5833881400, 0.6907874800,      STAP 124
     * 0.7409771300, 0.5372224600, 0.3037798800, 0.3740273200/      STAP 125
      DOUBLE PRECISION LAT  2                                       STAP 126
      DIMENSION LAT  2( 76)                                         STAP 127
      EQUIVALENCE (LAT   ( 77),LAT  2(1))                           STAP 128
      DATA LAT  2/                                                  STAP 129
     * 0.2110192200, 0.1875053300, 0.8367821900, 0.6304804800,      STAP 130
     * 0.8973779700, 0.6613157600, 0.7066373700, 0.6225417700,      STAP 131
     * 0.5651906200, 0.6102273900, 0.5585368700, 0.4716063400,      STAP 132
     * 0.6889158000, 0.4451848900, 0.7179078300, 0.4645295500,      STAP 133
     * 0.4970975000,-0.4528068400, 0.4926431000, 0.2992148800,      STAP 134
     * 0.3745930300, 0.4961043100, 0.4649886400, 0.5742537300,      STAP 135
     * 0.5647545900, 0.5742393000, 0.5901640300, 0.5837247300,      STAP 136
     * 0.5776768400, 0.6044553900, 0.6035870500, 0.6035820000,      STAP 137
     * 0.6044586300, 0.6035824300, 0.5309574600, 0.5309574600,      STAP 138
     * 0.5502550900, 0.5502558220, 0.3863136000, 0.3863176400,      STAP 139
     * 0.5955307800, 0.5955426600, 0.5802685800, 0.5802809300,      STAP 140
     * 0.5802932900, 0.5955493700, 0.5955560800, 0.3843570100,      STAP 141
     * 0.6103391100, 0.6860674500, 0.6860646500, 0.6609076500,      STAP 142
     * 0.6609076800, 0.6518311900, 0.6518311900, 0.5645785000,      STAP 143
     *-0.3316439300, 0.3861247800, 0.5645760600,-0.4345251300,      STAP 144
     * 0.6604548000, 0.6607850300,-0.5378853000, 0.9098049200,      STAP 145
     * 0.9093055000, 0.7877440400, 0.4209492400, 0.6035820000,      STAP 146
     * 0.6805709500, 0.5728229200, 0.8235294400, 0.7618259900,      STAP 147
     * 0.5833900500, 0.3634413500, 0.6607702600, 0.5571432000/      STAP 148
      DOUBLE PRECISION LAT  3                                       STAP 149
      DIMENSION LAT  3( 76)                                         STAP 150
      EQUIVALENCE (LAT   (153),LAT  3(1))                           STAP 151
      DATA LAT  3/                                                  STAP 152
     * 0.5595279600, 0.4448769700, 0.6811590400, 1.0336227700,      STAP 153
     * 0.7022041400, 0.8805740200, 0.7858926700, 0.9596402000,      STAP 154
     * 0.9535557900, 1.0181087300, 0.7216936500, 0.6811627700,      STAP 155
     * 0.9924093400, 0.8476269200, 0.8381533100, 0.4604106900,      STAP 156
     * 0.6788179000, 0.5648560800, 0.3186611400, 0.6810291800,      STAP 157
     * 0.6810427400, 0.5696861900, 0.6919665700, 0.4715965000,      STAP 158
     * 0.4715984300, 0.4715981300, 0.4715992300, 0.6108038400,      STAP 159
     * 0.3154898700, 0.6806632100, 0.6608348200,-0.4347004300,      STAP 160
     * 0.6810375600, 0.6142927000, 0.6607818000, 0.6810442400,      STAP 161
     *-0.4346664600, 0.5530146400, 0.5530141900, 0.6810402300,      STAP 162
     * 0.7452701400, 0.6363926200, 0.5658983900,-0.4530926700,      STAP 163
     *-0.2373841000, 0.5520937800,-0.1034601100, 0.6645937700,      STAP 164
     * 0.9177295700, 0.9076039600, 0.8181578000, 0.9100748600,      STAP 165
     * 0.9727393400, 0.6630191500, 0.7667694400, 0.7631600000,      STAP 166
     * 0.7667703300,-0.6061475400, 0.8518304400, 0.9727393400,      STAP 167
```

```
     * 0.8342934600, 0.8765023500, 0.6363942200, 0.7667606100,                STAP 168
     * 0.5558982400,-0.4530906700,-0.5428122900, 0.6364033300,                STAP 169
     * 0.6225104300, 0.5124156300,-0.2873844100, 0.5172661300,                STAP 170
     * 0.2110156900, 0.4715972200,-0.5575122000, 0.3614076200/                STAP 171
      DOUBLE PRECISION LAT  4                                                 STAP 172
      DIMENSION LAT  4( 26)                                                   STAP 173
      EQUIVALENCE (LAT   (229),LAT  4(1))                                     STAP 174
      DATA LAT  4/                                                            STAP 175
     * 0.2573255900, 0.5529928100,-0.5478715800, 0.6284151100,                STAP 176
     * 0.1526712600,-0.1034601100,-0.8008958600,-0.1034610300,                STAP 177
     * 0.4715952900, 0.7418664900, 0.8181579000, 0.9939422200,                STAP 178
     * 0.8488183800, 0.6646039900, 0.6102377300, 0.9554402000,                STAP 179
     * 1.0508795900, 0.2922327700,-0.7677514300, 0.7092830400,                STAP 180
     * 0.6102376900, 0.9142721000, 0.9720339200, 1.0500205700,                STAP 181
     * 0.2573023300, 0.7092800200/                                            STAP 182
      DOUBLE PRECISION LON  1                                                 STAP 183
      DIMENSION LON  1( 76)                                                   STAP 184
      EQUIVALENCE (LON   (  1),LON  1(1))                                     STAP 185
      DATA LON  1/                                                            STAP 186
     * 4.8748166400, 4.9176780500, 5.1546866900, 5.2053985800,                STAP 187
     * 6.9102959000, 6.0331221900, 6.2104237400, 1.9846813600,                STAP 188
     * 2.5261452500, 2.6001518600, 3.4965034700, 4.2433522500,                STAP 189
     * 4.3507345100, 4.5836045800, 4.9420226300, 4.2437751500,                STAP 190
     * 4.2445419400, 4.2447269400, 4.2430749900, 2.3891359000,                STAP 191
     * 2.6002125100, 0.4531983300, 6.2090229700, 6.2069493300,                STAP 192
     * 4.8770393000, 5.3615589600, 4.9377721400, 4.8543625900,                STAP 193
     * 4.9117216900, 4.9366590700, 5.0497885700, 5.3630387100,                STAP 194
     * 3.7028942500, 4.5900196200, 5.2710064700, 0.4835832600,                STAP 195
     * 4.2428901600, 2.3888473700, 3.7083908500, 4.9377721400,                STAP 196
     * 4.8543625900, 0.8255372800, 2.3888473700, 4.9117216900,                STAP 197
     * 4.9366590700, 5.0497885700, 4.2428901600, 0.4835832600,                STAP 198
     * 5.3630387100, 3.7028942500, 4.5900196200, 5.2710064700,                STAP 199
     * 3.7033908500, 4.8367977900, 2.5997399200, 4.8367963400,                STAP 200
     * 0.8255372800, 2.5997399200, 0.8255958600, 4.8367233300,                STAP 201
     * 5.0498218000, 3.7085923200, 1.9847250900, 0.8255959600,                STAP 202
     * 4.9367246800, 5.0498217700, 3.7086198300, 1.9847250600,                STAP 203
     * 5.2047634800, 4.6948782600, 4.6964403800, 4.4526176100,                STAP 204
     * 5.0392327500, 4.7428920500, 4.8181256700, 5.0414559700/                STAP 205
      DOUBLE PRECISION LON  2                                                 STAP 206
      DIMENSION LON  2( 76)                                                   STAP 207
      EQUIVALENCE (LON   ( 77),LON  2(1))                                     STAP 208
      DATA LON  2/                                                            STAP 209
     * 5.0817424500, 5.2078834200, 4.5935696800, 4.5864477200,                STAP 210
     * 0.1658104500, 0.4143549600, 6.2234366000, 2.4353138500,                STAP 211
     * 5.1342564700, 4.2251878600, 4.8667850200, 4.8849533700,                STAP 212
     * 4.9555007800, 4.8801485800, 4.4529161300, 4.9157552300,                STAP 213
     * 4.8768559600, 0.4949284700, 4.8764594200, 5.2046969500,                STAP 214
     * 5.0416948200, 4.8753229900, 4.9171537300, 4.4316026300,                STAP 215
     * 4.4266789100, 4.4314026600, 4.4216314100, 4.4308277700,                STAP 216
     * 4.4303358100, 4.1451770200, 4.1789850900, 4.1789871000,                STAP 217
     * 4.1451601500, 4.1786402400, 4.7682714600, 4.7682683300,                STAP 218
     * 4.1971776900, 4.1971752500, 3.4954071100, 3.4954077600,                STAP 219
     * 4.2033037900, 4.2035738900, 4.1971638300, 4.1971526100,                STAP 220
     * 4.1971812300, 4.2035869800, 4.2035800700, 3.4945182300,                STAP 221
     * 4.2249034400, 4.2745365700, 4.2745407300, 4.9663231600,                STAP 222
     * 4.9663230200, 4.9584228800, 4.9584228800, 5.1547650300,                STAP 223
```

```
     • 0.8257977900, 3.4964966000, 5.1547660600, 1.9847387800,      STAP 224
     • 4.9557190000, 4.9652966100, 2.3882757200, 6.2034523500,      STAP 225
     • 6.2034428000, 6.2628070200, 4.9828412000, 4.1789870000,      STAP 226
     • 4.9335471100, 4.2388596400, 4.2003567600, 4.6150211600,      STAP 227
     • 4.6954376400, 3.5523405400, 4.9652904000, 4.8595799500/      STAP 228
      DOUBLE PRECISION LON  3                                       STAP 229
      DIMENSION LON  3( 76)                                         STAP 230
      EQUIVALENCE (LON   (153),LON  3(1))                           STAP 231
      DATA LON  3/                                                  STAP 232
     • 4.8657695200, 4.9804223100, 4.9423236900, 0.6962166900,      STAP 233
     • 0.7765715200, 0.5323596000, 0.6802002500, 1.4460052900,      STAP 234
     • 0.6933153600, 0.4654211300, 1.2075351300, 4.9423290100,      STAP 235
     • 4.5192343500, 4.8629945900, 4.5000196200, 4.5669830300,      STAP 236
     • 4.6738989900, 5.1547107300, 5.1312820600, 4.9422898000,      STAP 237
     • 4.9422997100, 4.7883901600, 4.4574665200, 4.8849465400,      STAP 238
     • 4.8849473800, 4.8849485300, 4.8849487200, 4.8703917600,      STAP 239
     • 4.9426130800, 4.9420830500, 4.9553047700, 1.9848058200,      STAP 240
     • 4.9422800800, 4.8367247000, 4.9652896900, 4.9422845600,      STAP 241
     • 1.9847200000, 4.3479951300, 4.3479932000, 4.9422822700,      STAP 242
     • 4.9422750100, 6.1748830600, 4.4234915200, 0.4930184300,      STAP 243
     • 5.0352033500, 4.3479950700, 5.6694469900, 0.4176999700,      STAP 244
     • 0.1833445000, 0.0762868400, 0.1302856600, 6.2488280500,      STAP 245
     • 6.2263212000, 0.4142000400, 0.0997006400, 0.1274026500,      STAP 246
     • 0.0996995300, 2.4198500400, 0.0389108600, 6.2268212000,      STAP 247
     • 0.1923818400, 0.1523618300, 6.1748560600, 0.0996951200,      STAP 248
     • 4.4234821100, 0.4930184300, 2.3873331800, 6.1748600300,      STAP 249
     • 2.4353945400, 1.3667963000, 5.0353939500, 0.9166432900,      STAP 250
     • 5.0817419000, 4.8649509200, 5.1468630600, 3.5559829000/      STAP 251
      DOUBLE PRECISION LON  4                                       STAP 252
      DIMENSION LON  4( 26)                                         STAP 253
      EQUIVALENCE (LON   (229),LON  4(1))                           STAP 254
      DATA LON  4/                                                  STAP 255
     • 5.9778732100, 4.3479950700, 2.3889830700, 2.4293596000,      STAP 256
     • 0.6799565500, 5.6694469900, 5.1031259000, 5.6694500700,      STAP 257
     • 4.8849481200, 5.0342568300, 0.1302856800, 0.4199076400,      STAP 258
     • 0.3991835900, 0.4177078500, 4.2251877800, 4.3625577100,      STAP 259
     • 0.1876359500, 3.3246037600, 2.9752487200, 0.3115046200,      STAP 260
     • 4.2231838000, 0.2280415800, 0.6417029200, 0.4354858800,      STAP 261
     • 5.9778732100, 0.3115046100/                                  STAP 262
      DOUBLE PRECISION HT   1                                       STAP 263
      DIMENSION HT   1( 95)                                         STAP 264
      EQUIVALENCE (HT    (  1),HT   1(1))                           STAP 265
      DATA HT   1/                                                  STAP 266
     •  -45.00000,  -46.00000,  -66.00000,  -27.00000,  208.00000,  STAP 267
     •  555.00000,  821.00000,    8.00000,   76.00000, 1149.00000,  STAP 268
     • 1123.00000,  920.00000,  -28.00000,  -41.00000,   -6.00000,  STAP 269
     •  983.00000,  936.00000, 1040.00000,  979.00000,  148.00000,  STAP 270
     •  673.00000, 1410.00000,  825.00000,  775.00000,  -41.00000,  STAP 271
     •  -19.00000,  -53.69300,  -42.00000, 3556.91300,   50.70300,  STAP 272
     •  713.89200,   48.00000,  156.36700,  203.16200,   90.41000,  STAP 273
     • 1540.97700,  876.25400,  130.40300,  283.12500,  -53.69300,  STAP 274
     •  -42.00000, 1359.77700,  130.40300, 3556.91300,   50.70300,  STAP 275
     •  713.89000,  876.25400, 1540.97700,   48.00000,  156.36700,  STAP 276
     •  203.16200,   90.41000,  283.12500,  849.93300,  949.57800,  STAP 277
     •  850.06300, 1359.77700,  949.57800, 1392.00000,  818.18000,  STAP 278
     •  727.19300,  340.39900,    1.15300, 1381.00000,  814.00000,  STAP 279
```

```
     727.00000,  340.00000,    1.00000, -59.37800, -15.90900,   STAP 280
     -16.66400, 2141.20100,   30.06400,  17.43000,   7.44800,   STAP 281
     -61.08200,  -40.73100,  245.51500, 243.17100, 255.67800,   STAP 282
     368.65500,   59.74100,  635.09400,  69.34900, -13.33200,   STAP 283
     752.74400,  -45.05800,  -36.71700, -54.78500, -49.80500,   STAP 284
    1840.19900,  -41.90700,  -40.37800, 1609.20600, -38.82200/  STAP 285
    DOUBLE PRECISION HT   2                                     STAP 286
    DIMENSION HT   2( 95)                                       STAP 287
    EQUIVALENCE (HT    ( 96),HT   2(1))                         STAP 288
    DATA HT   2/                                                STAP 289
     -3.66000,  -27.03600,  -42.79200,  -45.66900, 1217.35100,  STAP 290
    1185.61400, 1217.55100, 1484.31300, 1577.09800, 1225.94100,  STAP 291
       3.27600,  599.26700,  599.23700,   -8.03200,   60.80600,  STAP 292
     -25.14700,  -26.76700,  220.97000,  221.26000,  447.41900,  STAP 293
     442.98700,  -47.35000,  -47.55800,  218.97100,  220.01200,  STAP 294
     219.43300,  -47.30700,  -47.30600,  -15.30900,  765.45500,  STAP 295
    2787.74500, 2776.33600,  -54.06600,  -54.06600,  -60.47400,  STAP 296
     -60.47400,  -36.79500, 1320.10700, 1127.48300,  -35.55200,  STAP 297
      12.30400,  -50.28500,  -47.72400,  124.34800,  182.55700,  STAP 298
     182.55600,   41.87300,  -49.62500,  665.88700,   68.33500,  STAP 299
      76.65300,  322.16000,  426.00100,  -17.16400,   28.27400,  STAP 300
     -49.29300,  -29.38300,  -43.25900,  -43.71600,  -15.25400,  STAP 301
     150.00000,  960.00000,  184.00000,   40.00000,  150.00000,  STAP 302
     114.00000,   75.00000,  440.80000,  -14.65400,  293.50800,  STAP 303
     231.43100,  203.15200,    7.78200,  212.51800,  -27.00000,  STAP 304
     -18.36000,   -6.19300,   -6.09300,  124.72000, 1745.43100,  STAP 305
     -37.67500,  -37.42500,  -38.16600,  -37.51600,  221.00000,  STAP 306
     404.75600,   -8.80000,  -55.11900,  -12.94100,    3.00000,  STAP 307
     819.68900,  -60.00000,   -5.20700,   -5.36100, 2317.83300/  STAP 308
    DOUBLE PRECISION HT   3                                     STAP 309
    DIMENSION HT   3( 64)                                       STAP 310
    EQUIVALENCE (HT    (191),HT   3(1))                         STAP 311
    DATA HT   3/                                                STAP 312
    2317.71100,   -4.85600,  177.60600,   38.30000, 1600.80000,  STAP 313
    1570.30000, 2490.00000, 2339.14100,   25.39000,  490.32800,  STAP 314
      91.55300,   45.54000,  933.22000,  137.02000,  309.62700,  STAP 315
     133.59200,  694.32000,  405.22000,  674.87100,    6.29500,  STAP 316
     190.01000,  309.62700,  961.70800,  193.81900,   55.87500,  STAP 317
     686.09200, 1615.00000, 1570.00000,  158.11500,   55.44000,  STAP 318
      80.00000, 1856.00000, 2490.00000, 1563.80000,  -24.00000,  STAP 319
     -23.00000,  636.54000, 3031.81600,  171.00000, 2339.05000,  STAP 320
     138.40600,  879.00000, 1901.30000,   25.39000,  234.06400,  STAP 321
      44.00000,  -35.78600,  131.45400,  933.00000,  -14.56000,  STAP 322
     204.69000,  490.32000,  743.44400,  654.00000,  595.04000,  STAP 323
      -7.00000, 1011.00000,  144.00000,  729.17400,  122.06400,  STAP 324
     133.92800,   40.00000,  170.67200,  144.01000/              STAP 325
    END                                                         STAP 326
```

START

DESCRIPTION

START returns the array of back values of accelerations plus the values of the first and second sums needed for the integration routine. It iterates using interpolation until the sums converge. The arguments used for iteration are the epoch position and velocity arrays and initial values assigned to the variational partials. Initial predictions are made with a Taylor series approximation.

NAME	START
PURPOSE	TO START INTEGRATION PROCESS USING INTERPOLATOR FORMULAS AND ITERATING UNTIL DESIRED ACCURACY IS ACHIEVED
CALLING SEQUENCE	CALL START(IORDER,H,FCT,SUM,Y,NN,M1,M2,TIM)

SYMBOL	TYPE	DESCRIPTION
IORDER	I	INPUT - ORDER
H	DP	INPUT - STEPSIZE
FCT (3,1)	DP	OUTPUT - ARRAY OF BACK VALUES OF ACCELERATION
SUM (2,3,1)	DP	OUTPUT - SUM ARRAY USED BY INTEGRATOR AND INTERPOLATOR
Y (6,1)	DP	INPUT - ARRAY OF STATE AND PARTIALS
NN	I	INPUT - NUMBER OF EQUATIONS
M1	I	INPUT - DISPLACEMENT USED BY COWELL
M2	I	INPUT - DISPLACEMENT USED BY COWELL
TIM	DP	INPUT - EPOCH TIME OUTPUT - INTEGRATOR TIME

SUBROUTINES USED	CLEAR F VEVAL COEF DOTPRD ERROR
COMMON BLOCKS	INTBLK
INPUT FILES	NONE
OUTPUT FILES	PRINTER
REFERENCES	'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION

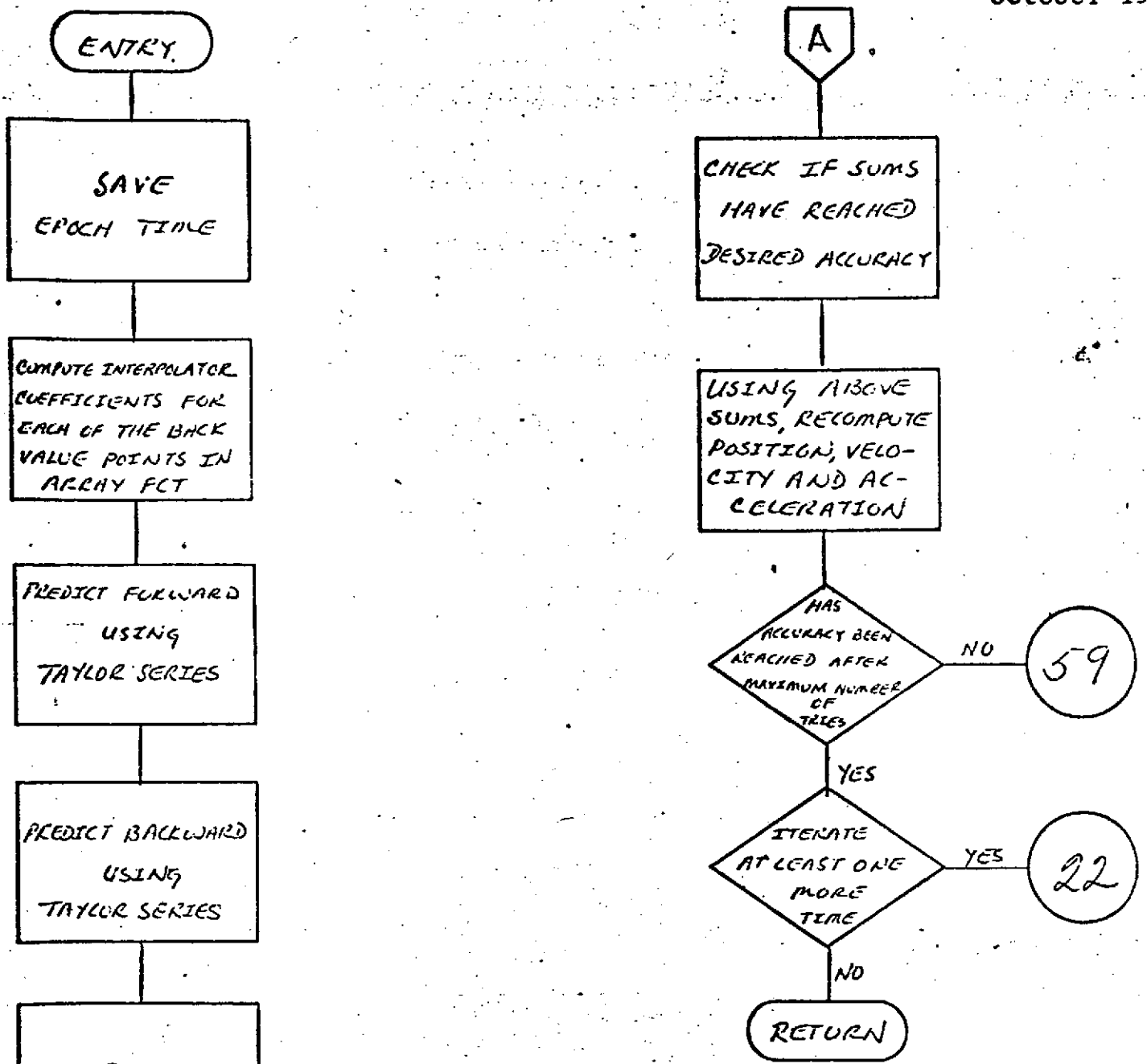
```
      SUBROUTINE START(IORDER,H,FCT,SUM,Y,NN,M1,M2,TIM)                 STAR  48
      IMPLICIT REAL*8(A-H,O-Z)                                          STAR  49
      DIMENSION Y(6,1),AUX(6)                                           STAR  50
      DIMENSION FCT(3,1),SUM(2,3,1),C(15,15,2)                          STAR  51
      COMMON/INTBLK/THOUT(3),GM,AE(62)                                  STAR  52
      DIMENSION A1(15,15),AS1(15,15)                                    STAR  53
      EQUIVALENCE(C(1,1,1),AS1(1,1)),(C(1,1,2),A1(1,1))                 STAR  54
      DATA EPS/1.0D-13/,MAXK/20/                                        STAR  55
```

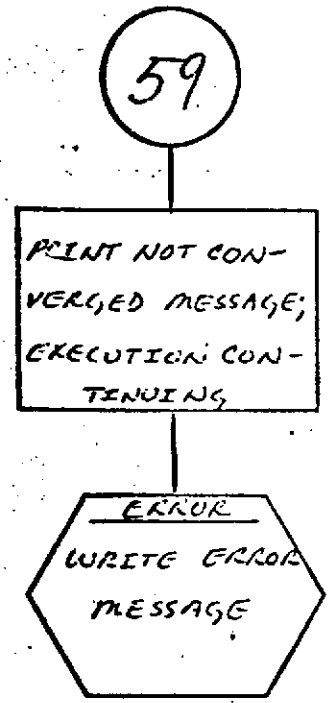
```
      LOGICAL SWITCH,NEQSW                                              STAR  56
      IKNT=0                                                            STAR  57
C SAVE EPOCH TIME                                                       STAR  58
      T=TIM                                                             STAR  59
      NEQSW=NN.GT.1                                                     STAR  60
      IOL2=IORDER-2                                                     STAR  61
      IOL1=IORDER-1                                                     STAR  62
      IEVEN=((IORDER+2)/2)*2-IORDER-1                                   STAR  63
      MI=IOL2/2-IEVEN                                                   STAR  64
      MID=MI+1+IEVEN                                                    STAR  65
      MIDP2=MID+2                                                       STAR  66
      IST=M1-IOL2                                                       STAR  67
      ISTV=M2-IOL2                                                      STAR  68
      CALL CLEAR(SUM,12,NN)                                             STAR  69
      M=IST+MID                                                         STAR  70
      MV=ISTV+MID                                                       STAR  71
      DO 5 J=1,6                                                        STAR  72
    5 AUX(J)=Y(J,1)                                                     STAR  73
      IF(.NOT.NEQSW)GOTO7                                               STAR  74
      DO 6 J=1,6                                                        STAR  75
      JP1=J+1                                                           STAR  76
      DO 4 N=2,NN                                                       STAR  77
    4 Y(J,N)=0.D0                                                       STAR  78
      Y(J,JP1)=1.D0                                                     STAR  79
    6 CONTINUE                                                          STAR  80
C ONE TIME CALL AT EPOCH                                                STAR  81
    7 CALL F(T,Y,FCT(1,M),.FALSE.)                                      STAR  82
    1 FORMAT(1H0,6G20.10)                                               STAR  83
      IF(NEQSW  )CALL VEVAL(Y,FCT(1,M1+MV),6,.TRUE.,M2)                 STAR  84
C COMPUTE INTERPOLATOR COEFFICIENTS FOR EACH OF THE BACK VALUE POINTS    STAR  85
C IN ARRAY FCT                                                          STAR  86
      DO 10 I=1,IOL2                                                    STAR  87
      K=I+2                                                             STAR  88
      S=DFLOAT(I-IOL2)                                                  STAR  89
      CALL COEF(S,IORDER,A1(I,K),AS1(1,K))                              STAR  90
   10 CONTINUE                                                          STAR  91
      S=-DFLOAT(IOL2)                                                   STAR  92
      CALL COEF(S,IORDER,A1(1,2),AS1(1,2))                              STAR  93
      TIM=T                                                             STAR  94
C PREDICT FORWARD USING TAYLOR SERIES                                   STAR  95
      DO 200 I=1,MI                                                     STAR  96
      J=M+I                                                             STAR  97
      JV=MV+I                                                           STAR  98
      R0=DSQRT(DOTPRD(Y,Y))                                             STAR  99
      R1=DOTPRD(Y,Y(4,1))/R0                                            STAR 100
      DO 19 K=1,3                                                       STAR 101
      KP3=K+3                                                           STAR 102
      GERK=-GM*(Y(KP3,1)-3.D0*R1*Y(K,1)/R0)/R0**3                       STAR 103
      Y(K,1)=Y(K,1)+H*(Y(KP3,1)+H*(FCT(K,  J-1)*.5D0+H*GERK/6.D0))      STAR 104
   19 Y(KP3,1)=Y(KP3,1)+H*(FCT(K,  J-1)+H*.5D0*GERK)                    STAR 105
      TIM=TIM+H                                                         STAR 106
      CALL F(TIM,Y,FCT(1,J),.FALSE.)                                    STAR 107
      IF(.NOT.NEQSW)GO TO 200                                           STAR 108
      K0=M1+JV-1                                                        STAR 109
      DO 799 N=2,NN                                                     STAR 110
      KK=K0+(N-2)*M2                                                    STAR 111
```

```
      DO 799 K=1,3                                                      STAR 112
      KP3=K+3                                                           STAR 113
      Y(K,N)=Y(K,N)+H*(Y(KP3,N)+H*FCT(K,KK   )*.5D0)                    STAR 114
  799 Y(KP3,N)=Y(KP3,N)+H*FCT(K,KK)                                     STAR 115
      CALL VEVAL(Y,FCT(1,K0+1),6,.TRUE.,M2)                             STAR 116
  200 CONTINUE                                                          STAR 117
C PREDICT BACKWARD USING TAYLOR SERIES                                  STAR 118
      DO 205 K=1,6                                                      STAR 119
      Y(K,1)=AUX(K)                                                     STAR 120
  205 CONTINUE                                                          STAR 121
      IF(.NOT.NEQSW)GO TO 207                                           STAR 122
      DO 209 J=1,6                                                      STAR 123
      DO 208 N=2,NN                                                     STAR 124
  208 Y(J,N)=0.D0                                                       STAR 125
  209 Y(J,J+1)=1.D0                                                     STAR 126
  207 CONTINUE                                                          STAR 127
      TIM=T                                                             STAR 128
      DO 21 I=1,MID                                                     STAR 129
      J=M-I                                                             STAR 130
      JV=MV-I                                                           STAR 131
      R0=DSQRT(DOTPRD(Y,Y))                                             STAR 132
      R1=DOTPRD(Y,Y(4,1))/R0                                            STAR 133
      DO 20 K=1,3                                                       STAR 134
      KP3=K+3                                                           STAR 135
      GERK=-GM*(Y(KP3,1)-3.D0*R1*Y(K,1)/R0)/R0**3                       STAR 136
      Y(K,1)=Y(K,1)-H*(Y(KP3,1)-H*(FCT(K,  J+1)*.5D0-H*GERK/6.D0))      STAR 137
   20 Y(KP3,1)=Y(KP3,1)-H*(FCT(K,  J+1)-H*.5D0*GERK)                    STAR 138
      TIM=TIM-H                                                         STAR 139
      CALL F(TIM,Y,FCT(1,  J),.FALSE.)                                  STAR 140
      IF(.NOT.NEQSW)GO TO 21                                            STAR 141
      K0=M1+JV+1                                                        STAR 142
      DO 800 N=2,NN                                                     STAR 143
      KK=K0+(N-2)*M2                                                    STAR 144
      DO 800 K=1,3                                                      STAR 145
      KP3=K+3                                                           STAR 146
      Y(K,N)=Y(K,N)-H*(Y(KP3,N)-H*FCT(K,KK   )*.5D0)                    STAR 147
  800 Y(KP3,N)=Y(KP3,N)-H*FCT(K,KK)                                     STAR 148
      CALL VEVAL(Y,FCT(1,K0-1),6,.TRUE.,M2)                             STAR 149
   21 CONTINUE                                                          STAR 150
C RESET EPOCH VALUES                                                    STAR 151
      DO 206 K=1,6                                                      STAR 152
      Y(K,1)=AUX(K)                                                     STAR 153
  206 CONTINUE                                                          STAR 154
      IF(.NOT.NEQSW)GO TO 22                                            STAR 155
      DO 2206 J=1,6                                                     STAR 156
      DO 2207 N=2,NN                                                    STAR 157
 2207 Y(J,N)=0.D0                                                       STAR 158
 2206 Y(J,J+1)=1.D0                                                     STAR 159
   22 CONTINUE                                                          STAR 160
      SWITCH=.FALSE.                                                    STAR 161
      KOUNT=0                                                           STAR 162
C COMPUTE SUMS                                                          STAR 163
   23 DO 30 N=1,NN                                                      STAR 164
      K3=0                                                              STAR 165
      IF(N.GT.1)K3=1                                                    STAR 166
      K4=K3+1                                                           STAR 167
```

```
      ISTP=K3*(ISTV-IST)+IST                                            STAR 168
      K0=K3*M1+(N-K4)*M2+ISTP+IOL1                                      STAR 169
      DO 30 J=1,3                                                       STAR 170
      A=C(1,1)                                                          STAR 171
      B=C(1,2)                                                          STAR 172
      DO 29 I=1,IOL2                                                    STAR 173
      KK=K0-I                                                           STAR 174
      A=A-C(I,MID+2,1)*FCT(J,KK)                                        STAR 175
      B=B-C(I,MID+2,2)*FCT(J,KK)                                        STAR 176
   29 CONTINUE                                                          STAR 177
      A=A-C(IOL1,MID+2,1)*FCT(J,K0-IOL1)                                STAR 178
      A=Y(J+3,N)/H+A                                                    STAR 179
      B=Y(J,N)/H**2+FLOAT(I+MI)*A+B                                     STAR 180
C CHECK IF SUMS HAVE REACHED DESIRED ACCURACY                           STAR 181
      DIFF1=DABS(A-SUM(1,J,N))                                          STAR 182
      SUM(1,J,N)=A                                                      STAR 183
      DIFF2=DABS(B-SUM(2,J,N))                                          STAR 184
      SUM(2,J,N)=B                                                      STAR 185
      IF(DABS(A).GT.1.D-50)DIFF1=DIFF1/DABS(A)                          STAR 186
      IF(DABS(B).GT.1.D-50)DIFF2=DIFF2/DABS(B)                          STAR 187
      IF(DIFF1.GT.EPS)SWITCH=.TRUE.                                     STAR 188
      IF(DIFF2.GT.EPS)SWITCH=.TRUE.                                     STAR 189
   30 CONTINUE                                                          STAR 190
C USING ABOVE SUMS, RECOMPUTE POSITION & VELOCITY & ACCELERATION        STAR 191
      KOUNT=KOUNT+1                                                     STAR 192
      II=IORDER+1-KOUNT                                                 STAR 193
      IF(II.EQ.MIDP2)GOTO 50                                            STAR 194
      I=IST+IOL1-KOUNT                                                  STAR 195
      I2=-KOUNT                                                         STAR 196
      DO 45 N=1,NN                                                      STAR 197
      K3=0                                                              STAR 198
      IF(N.GT.1)K3=1                                                    STAR 199
      K4=K3+1                                                           STAR 200
      ISTP=K3*(ISTV-IST)+IST                                            STAR 201
      K0=K3*M1+(N-K4)*M2+ISTP+IOL1                                      STAR 202
      DO 45 J=1,3                                                       STAR 203
      A=0.D0                                                            STAR 204
      B=0.D0                                                            STAR 205
      DO 44 K=1,IOL2                                                    STAR 206
      KK=K0-K                                                           STAR 207
      A=A+C(K,II,1)*FCT(J,KK)                                           STAR 208
      B=B+C(K,II,2)*FCT(J,KK)                                           STAR 209
   44 CONTINUE                                                          STAR 210
      A=A+C(IOL1,II,1)*FCT(J,K0-IOL1)                                   STAR 211
      A=A+SUM(1,J,N)                                                    STAR 212
      B=B+SUM(1,J,N)*DFLOAT(I2  )+SUM(2,J,N)                            STAR 213
      Y(J,N)    =B*H**2                                                 STAR 214
      Y(J+3,N)=    A*H                                                  STAR 215
   45 CONTINUE                                                          STAR 216
      TIM=T+DFLOAT(I -M  )*H                                            STAR 217
      CALL F(TIM,Y,FCT(1,I),.FALSE.)                                    STAR 218
      IF(NEQSW  )CALL VEVAL(Y,FCT(1,N1+ISTV+IOL1-KOUNT),6,.TRUE.,M2)    STAR 219
      DO 52 J=1,6                                                       STAR 220
   52 Y(J,1)=AUX(J)                                                     STAR 221
      IF(.NOT.NEQSW)GOTO50                                              STAR 222
      DO 53 J=1,6                                                       STAR 223
```

```
      DO 55 K=2,NN                                                        STAR 224
   55 Y(J,K)=0.D0                                                         STAR 225
   53 Y(J,J+1)=1.D0                                                       STAR 226
   59 IF(KOUNT.LT.IOL1)GO TO 23                                           STAR 227
      IKNT=IKNT+1                                                         STAR 228
C EXIT LOOP IF ACCURACY HAS NOT BEEN REACHED AFTER MAXK TRIES             STAR 229
      IF(IKNT.GT.MAXK) GO TO 59                                           STAR 230
C RECOMPUTE SUMS                                                          STAR 231
      IF(SWITCH) GO TO 22                                                 STAR 232
      GO TO 60                                                            STAR 233
   59 PRINT 61,IKNT,EPS,DIFF1,DIFF2                                       STAR 234
      CALL ERROR(10,DIFF1)                                                STAR 235
   61 FORMAT(1H1,20X,'INTEGRATION STARTING SUMS NOT CONVERGED AFTER',     STAR 236
     *   I3,' ITERATIONS'/1H0,15X,'EPS   =',D22.16/16X,'DIFF1 =',         STAR 237
     *   D22.16/16X,'DIFF2 =',D22.16/1H0,20X,'EXECUTION CONTINUING'//)    STAR 238
                                                                          STAR 239
   60 TIM=T+DFLOAT(MI)*H                                                  STAR 240
      RETURN                                                              STAR 241
      END
```


ENTRY
SAVE EPOCH TIME
COMPUTE INTERPOLATOR COEFFICIENTS FOR EACH OF THE BACK VALUE POINTS IN ARRAY FCT
PREDICT FORWARD USING TAYLOR SERIES
PREDICT BACKWARD USING TAYLOR SERIES
RESET EPOCH VALUES
22
COMPUTE SUMS
A
A
CHECK IF SUMS HAVE REACHED DESIRED ACCURACY
USING ABOVE SUMS, RECOMPUTE POSITION, VELOCITY AND ACCELERATION
HAS ACCURACY BEEN REACHED AFTER MAXIMUM NUMBER OF TRIES
NO
59
YES
ITERATE AT LEAST ONE MORE TIME
YES
22
NO
RETURN


59
PRINT NOT CONVERGED MESSAGE; EXECUTION CONTINUING
ERROR
WRITE ERROR MESSAGE

NAME STORE

PURPOSE TO STORE COMMON AND ARC INFORMATION ON DISK

CALLING SEQUENCE CALL STORE(RECALL,COMPAR)

SYMBOL	TYPE	DESCRIPTION
RECALL	L	INPUT - .FALSE. - STORE INFORMATION .TRUE. - RETRIEVE INFORMATION
COMPAR	L	INPUT - .FALSE. = ARC PARAMETER INFORMATION .TRUE. = COMMON PARAMETER INFORMATION

SUBROUTINES USED NONE

COMMON BLOCKS APARAM CELEM CTIME FLXBLK FMODEL
INITBK INTBLK PREBLK PRIORI TPEBLK
VRBLOK

INPUT FILES SCRC - SCRATCH

OUTPUT FILES SCRC - SCRATCH

```
      SUBROUTINE STORE(RECALL,COMPAR)                                   STOR  28
      IMPLICIT REAL*8 (A-H,O-Z)                                         STOR  29
      LOGICAL RECALL,COMPAR,MISLOG                                      STOR  30
      INTEGER XYZTP,PLOTP,SCRC,FLTP,THETGO,ELEMST                       STOR  31
      REAL DAYREF                                                       STOR  32
      DOUBLE PRECISION MODEL                                            STOR  33
      COMMON/APARAM/INPAR(10)                                           STOR  34
      COMMON/CELEM/ELEMST(53)                                           STOR  35
      COMMON/CTIME/DAYREF(23)                                           STOR  36
      COMMON/FLXBLK/ISTORE(450,9)                                       STOR  37
      COMMON/FMODEL/INDEX(4),CS(30,33),MODEL(8)                         STOR  38
      COMMON/INITBK/ISPYMD(43),MISLOG(9)                                STOR  39
      COMMON/INTBLK/THOOT1(27),THETGO(78)                               STOR  40
      COMMON/PREBLK/DAYSTA,NOPS(15)                                     STOR  41
      COMMON/PRIORI/ELEMIN(96)                                          STOR  42
      COMMON/TPEBLK/INTP(3),XYZTP(3),PLOTP(3),SCRC,FLTP(2)              STOR  43
      COMMON/VRBLOK/JSTORE(450,5)                                       STOR  44
      IF(RECALL) GO TO 500                                              STOR  45
      DO 100 I=1,9                                                      STOR  46
  100 WRITE(SCRC) (ISTORE(J,I),J=1,450)                                 STOR  47
      DO 200 I=1,5                                                      STOR  48
  200 WRITE(SCRC),(JSTORE(J,I),J=1,450)                                 STOR  49
      IF(COMPAR) RETURN                                                 STOR  50
      WRITE(SCRC) DAYREF,INDEX,DAYSTA,XYZTP,THETGO,ELEMIN,INPAR,ISPYMD, STOR  51
     *   ELEMST                                                         STOR  52
      RETURN                                                            STOR  53
  500 DO 700 I=1,9                                                      STOR  54
  700 READ(SCRC) (ISTORE(J,I),J=1,450)                                  STOR  55
```

```
      DO 800 I=1,5                                                          STOR  56
  800 READ(SCRC) (JSTORE(J,I),J=1,450)                                      STOR  57
      IF(COMPAR) RETURN                                                     STOR  58
      READ(SCRC) DAYREF,INDEX,DAYSTA,XYZTP,THETGO,ELEMIN,INPAR,IEPYMD,      STOR  59
     *   ELEMST                                                             STOR  60
      RETURN                                                                STOR  61
      END                                                                   STOR  62
```

NAME	SUMMRY
PURPOSE	TO PRINT ARC STATISTICAL SUMMARY
CALLING SEQUENCE	CALL SUMMRY(ARCNO,INNER,OUTER,LINNER,EDIT,NAME)

SYMBOL	TYPE	DESCRIPTION
ARCNO	I	INPUT - ARC NUMBER
INNER	I	INPUT - INNER ITERATION NUMBER
OUTER	I	INPUT - OUTER ITERATION NUMBER
LINNER	L	INPUT - .TRUE. - LAST INNER ITERATION
EDIT	R	INPUT - EDITING LEVEL
NAME (1)	I	INPUT - STATION NAMES

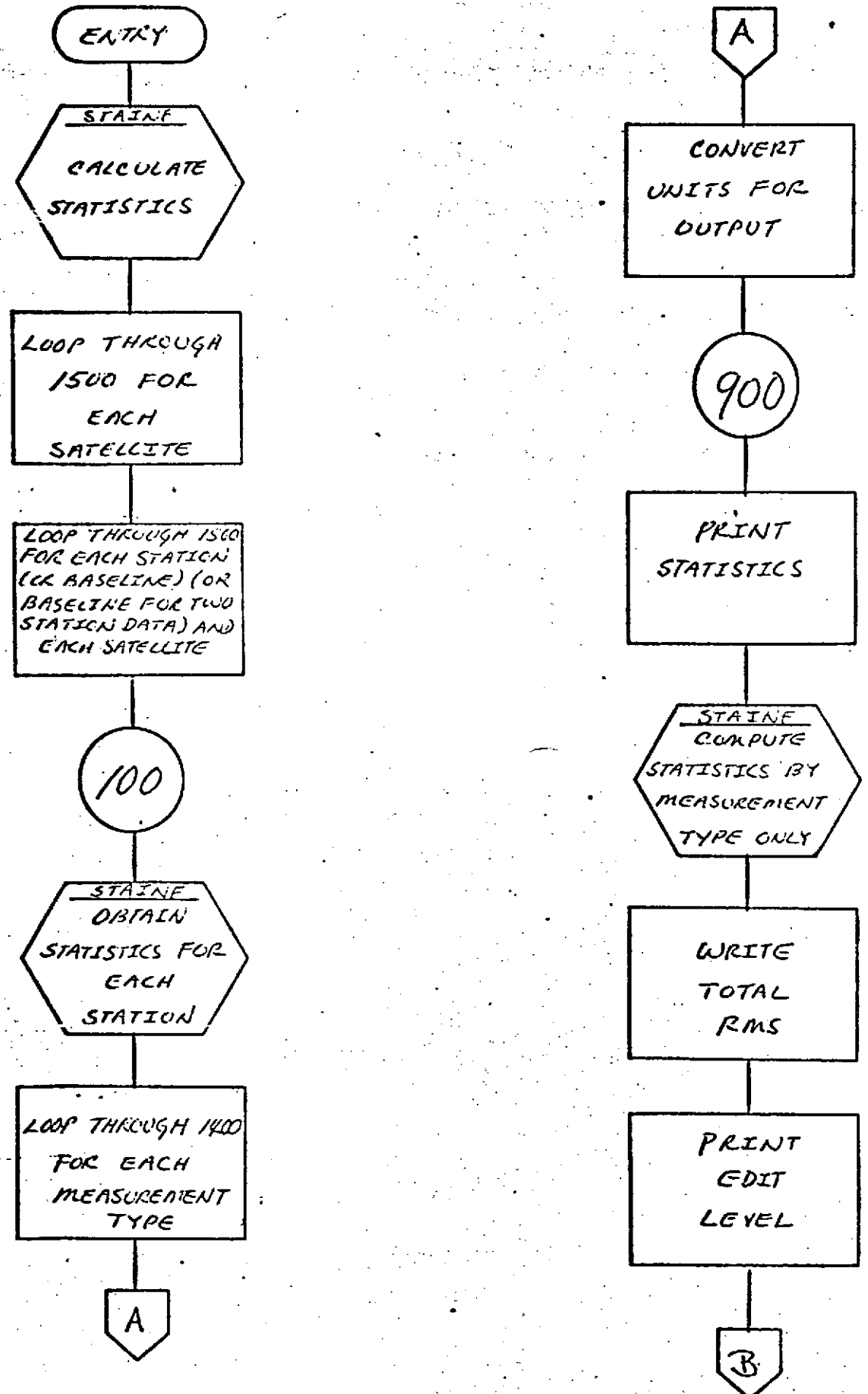
SUBROUTINES USED	STAINF EPROR
COMMON BLOCKS	ALPHRC APARAM CELEM CONSTS CPARAM CSTINF INITBK TPERLK
INPUT FILES	NONE
OUTPUT FILES	OUTP - PRINTER

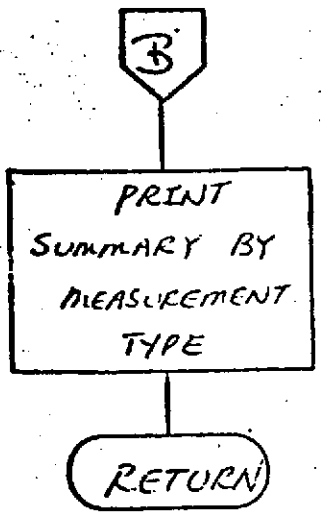
```
      SUBROUTINE SUMMRY(ARCNO,INNER,OUTER,LINNER,EDIT,NAME)                SUMM  34
      LOGICAL*1 BYTE,SLASH                                                 SUMM  35
      LOGICAL CMPGPR,SWITCH,LINNER,HYPER                                   SUMM  36
      INTEGER*2 ISAT                                                       SUMM  37
      INTEGER RECNO1,OUTP,DATP,ARCNO,OUTER                                 SUMM  38
      DOUBLE PRECISION ITNMS,TIMING,BLANK,ATYPE,UNITS,ELEMST,ORBELA,       SUMM  39
     *    EPSEC,ORBEL,CONFIG,NAME,ELCUT                                    SUMM  40
      DIMENSION BYTE(8),NAME(1)                                            SUMM  41
      COMMON/ALPHRC/ITNMS(5),TIMING,BLANK,ATYPE(31),UNITS(15),ELCUT,       SUMM  42
     *    HYPER                                                            SUMM  43
      COMMON/APARAM/INPAR,INPARI,NBIAS,ESTSTA,NSAT,NGPARC,RECNO1,NPARAM,   SUMM  44
     *    NBIAS,MAXPAR                                                     SUMM  45
      COMMON/CELEM/ELEMST(6,2),ORBELA(6,2),IGIS(4),RMSTOT                  SUMM  46
      COMMON/CONSTS/PI,PI1,TWOPI,TWOPI1,DRAD,DRAD1,RSEC,RSEC1              SUMM  47
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,NBIAS,NGPC1,NGPC2,NGPCOM,       SUMM  48
     *    NCSEST,CMPGPR,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,INNRSW,        SUMM  49
     *    NCONST,NDCONS                                                    SUMM  50
      COMMON/CSTINF/MEASNO(4),NOTS(4),RMSMEAN(4),RMSS(4),RMD(4),           SUMM  51
     *    MEASWT(4),WTMEAN(4),RMSWTG(4),WTRMD(4),TYPRMS(30),NUTYPE(2,30),  SUMM  52
     *    RSUM(4,12),RMSNOI(30),NOALL(30),NOWTOU,LBASE                     SUMM  53
      COMMON/INITBK/IEPYMD,IEPHM,EPSEC,IYREF,INNMAX,INNMIN,CONVRG,         SUMM  54
     *    ORBEL(6,2),EDITN,INSUPR,ICSAT(2),SWITCH(21)                      SUMM  55
```

```
      COMMON/TREELK/INTP,OUTP,DATP(10)                                  SUMM  56
      EQUIVALENCE (CONFIG,BYTE(1))                                      SUMM  57
      DATA ASTRSK/1H*/,SLASH/1H//                                       SUMM  58
      L1=LBASE+1                                                        SUMM  59
      ISAT=1                                                            SUMM  60
C CALCULATE STATISTICS                                                  SUMM  61
      CALL STAINF(3,0,P,P,P,P,ISAT,P)                                   SUMM  62
C LOOP THROUGH 1500 FOR EACH SATELLITE                                  SUMM  63
      DO 1500 ISAT=1,NSAT                                               SUMM  64
      WRITE(OUTP,5000) ISAT,ARCNO,INNER,OUTER                           SUMM  65
      LINE=1                                                            SUMM  66
      NOBS(1)=0                                                         SUMM  67
      IF(NSTA.LE.0) GO TO 1600                                          SUMM  68
C LOOP THROUGH 1500 FOR EACH STATION (OR BASELINE)                      SUMM  69
      DO 1500 J=1,NSTA                                                  SUMM  70
      DO 1500 LP1=1,L1                                                  SUMM  71
      L=LP1-1                                                           SUMM  72
      IF(NOBS(1).EQ.0) GO TO 100                                        SUMM  73
      WRITE(OUTP,5100)                                                  SUMM  74
      LINE=LINE+1                                                       SUMM  75
C OBTAIN STATISTICS FOR EACH STATION (OR BASE LINE FOR TWO STATION      SUMM  76
C DATA) AND EACH SATELLITE                                              SUMM  77
  100 CALL STAINF(3,J,P,P,P,P,ISAT,L)                                   SUMM  78
C LOOP THROUGH 1400 FOR EACH MEASUREMENT TYPE                           SUMM  79
      DO 1400 K=1,4                                                     SUMM  80
      IF(NOBS(K).EQ.0) GO TO 1500                                       SUMM  81
      IF(LINE.LT.40) GO TO 200                                          SUMM  82
      WRITE(OUTP,5200)                                                  SUMM  83
      WRITE(OUTP,5000) ISAT,ARCNO,INNER,OUTER                           SUMM  84
      LINE=1                                                            SUMM  85
  200 LINE=LINE+1                                                       SUMM  86
      MTYPE=MEASNO(K)                                                   SUMM  87
C CONVERT UNITS FOR OUTPUT                                              SUMM  88
      IF(MTYPE.LT.14) GO TO 300                                         SUMM  89
      M=MTYPE-26                                                        SUMM  90
      GO TO (400,450,700,700),M                                         SUMM  91
  300 M=MTYPE-(MTYPE/8)*7                                               SUMM  92
      GO TO (500,800,700,700,600,500,500),M                             SUMM  93
  400 RDMEAN(K)=RDMEAN(K)*1.0E+09                                       SUMM  94
      RMSO(K)=RMSO(K)*1.0E+09                                           SUMM  95
      GO TO 800                                                         SUMM  96
  450 RDMEAN(K)=RDMEAN(K)*1.0E+6                                        SUMM  97
      RMSO(K)=RMSO(K)*1.0E+6                                            SUMM  98
      GO TO 800                                                         SUMM  99
  500 RDMEAN(K)=RDMEAN(K)/RSEC                                          SUMM 100
      RMSO(K)=RMSO(K)/RSEC                                              SUMM 101
      GO TO 800                                                         SUMM 102
  600 RDMEAN(K)=RDMEAN(K)*1.0E+03                                       SUMM 103
      RMSO(K)=RMSO(K)*1.0E+03                                           SUMM 104
      GO TO 800                                                         SUMM 105
  700 IF(MTYPE.EQ.4) GO TO 800                                          SUMM 106
      RDMEAN(K)=RDMEAN(K)*1.0E+02                                       SUMM 107
      RMSO(K)=RMSO(K)*1.0E+02                                           SUMM 108
  800 CONFIG=BLANK                                                      SUMM 109
      IF(MTYPE.LT.27) GO TO 900                                         SUMM 110
      CONFIG=NAME(L)                                                    SUMM 111
```

```
      BYTE(8)=SLASH                                                     SUMM 112
C PRINT STATISTICS                                                      SUMM 113
  900 WRITE(OUTP,5300) CONFIG,NAME(J),ATYPE(MTYPE),NOBS(K),RDMEAN(K)    SUMM 114
      IF(NOBS(K).LT.10) GO TO 1000                                      SUMM 115
      A1=BLANK                                                          SUMM 116
      A2=BLANK                                                          SUMM 117
      IF(RND(K).GT.1.98) A1=ASTRSK                                      SUMM 118
      IF(RND(K).GT.2.58) A2=ASTRSK                                      SUMM 119
      WRITE(OUTP,5400) RND(K),A1,A2,RMSO(K)                             SUMM 120
 1000 IF(MEASWT(K).GT.0) WRITE(OUTP,5500) MEASWT(K),WTMEAN(K)           SUMM 121
      IF(MEASWT(K).LT.10) GO TO 1100                                    SUMM 122
      A1=BLANK                                                          SUMM 123
      A2=BLANK                                                          SUMM 124
      IF(WTRND(K).GT.1.98) A1=ASTRSK                                    SUMM 125
      IF(WTRND(K).GT.2.58) A2=ASTRSK                                    SUMM 126
      WRITE(OUTP,5600) WTRND(K),A1,A2,RMSWTO(K)                         SUMM 127
 1100 NOWTOB=NOWTOB+MEASWT(K)                                           SUMM 128
 1400 CONTINUE                                                          SUMM 129
 1500 CONTINUE                                                          SUMM 130
 1600 DO 1700 I=15,26                                                   SUMM 131
 1700 NOWTOB=NOWTOB+NOTYPE(2,I)                                         SUMM 132
      ISAT=1                                                            SUMM 133
      IF(NOWTOB.LT.8) CALL ERROR(5,BLANK)                               SUMM 134
C COMPUTE STATISTICS BY MEASUREMENT TYPE ONLY                           SUMM 135
      CALL STAINF(4,NPARAM,NM,RMSTOT,P,LINNER,P,P)                      SUMM 136
      IF(NSTA.LE.0) GO TO 1800                                          SUMM 137
C WRITE TOTAL RMS                                                       SUMM 138
      WRITE(OUTP,6700) NM,RMSTOT                                        SUMM 139
      WRITE(OUTP,5200)                                                  SUMM 140
C PRINT EDIT LEVEL                                                      SUMM 141
 1800 IF(EDITN.LT.100.) WRITE(OUTP,5800) EDIT                           SUMM 142
C PRINT SUMMARY BY MEASUREMENT TYPE                                     SUMM 143
      WRITE(OUTP,5900) ARCNO,INNER,OUTER                                SUMM 144
      DO 1900 I=1,7                                                     SUMM 145
      IF(NOTYPE(2,I).GT.0) WRITE(OUTP,6000) ATYPE(I),NOTYPE(2,I),       SUMM 146
     .   TYPRMS(I)                                                      SUMM 147
      J=I+7                                                             SUMM 148
      IF(NOTYPE(2,J).GT.0) WRITE(OUTP,6000) ATYPE(J),NOTYPE(2,J),       SUMM 149
     .   TYPRMS(J)                                                      SUMM 150
 1900 IF(NOTYPE(2,I)+NOTYPE(2,J).GT.0) WRITE(OUTP,5100)                 SUMM 151
      DO 2000 I=15,30                                                   SUMM 152
      IF(NOTYPE(2,I).LE.0) GO TO 2000                                   SUMM 153
      WRITE(OUTP,6000) ATYPE(I),NOTYPE(2,I),TYPRMS(I)                   SUMM 154
      WRITE(OUTP,5100)                                                  SUMM 155
 2000 CONTINUE                                                          SUMM 156
      WRITE(OUTP,6100) NM,RMSTOT                                        SUMM 157
      RETURN                                                            SUMM 158
 5000 FORMAT(1H1,22X,9HSATELLITE,I2,4H ARC,I3,                          SUMM 159
     .   32H RESIDUAL SUMMARY BY STATION FOR,                           SUMM 160
     .   16H INNER ITERATION,I3,19H OF OUTER ITERATION,I2//             SUMM 161
     1    14X,13HSTATION  MEAS,13X,31HRESIDUALS FROM ALL OBSERVATIONS,  SUMM 162
     2    8X,9HNO OF WTD,3X,13HMEAN RATIO,2X,9HWEIGHTED,3X,8HWEIGHTED/ SUMM 163
     3    15X,4HNAME,4X,4HTYPE,9X,6HNUMBER,5X,4HMEAN,7X,2HRMS,9X,3HRMS, SUMM 164
     4    7X,9HRESIDUALS,4X,8HTO SIGMA,5X,3HRND,8X,3HRMS/)              SUMM 165
 5100 FORMAT(1X)                                                        SUMM 166
 5200 FORMAT(1HO,12X,71H* - INDICATES RESIDUALS ARE SIGNIFICANTLY NON-RASUMM 167
```

```
     *NDOM - 5 PERCENT LEVEL /12X,72H** - INDICATES RESIDUALS ARE SIGNIFSUMM 168
     *ICANTLY NON-RANDOM - 1 PERCENT LEVEL )                          SUMM 169
  5300 FORMAT(6X,A8,A7,1X,A6,8X,I4,F12.3)                             SUMM 170
  5400 FORMAT(1H+,50X,F9.3,2A1,F9.3)                                  SUMM 171
  5500 FORMAT(1H+,80X,I4,3X,F11.3)                                    SUMM 172
  5600 FORMAT(1H+,100X,F9.3,2A1,F9.3)                                 SUMM 173
  5700 FORMAT(1HO,76X,3HALL,I5,22H WEIGHTED MEASUREMENTS,F14.3)       SUMM 174
  5800 FORMAT(1HO,1H ,16X,50HFOR THIS ITERATION NO MEASUREMENTS WITH RESISUMM 175
     1DUALS ,12HGREATER THAN,F9.2,6H SIGMA/1H ,20X,25HWERE USED IN THE SSUMM 176
     2OLUTION)                                                        SUMM 177
  5900 FORMAT(1HI,23X,3HARC,I3,38H RESIDUAL SUMMARY BY MEASUREMENT TYPE ,SUMM 178
     *    19HFOR INNER ITERATION,I3,19H OF OUTER ITERATION,I2/        SUMM 179
     2         1HO,38X,11HMEASUREMENT,8X,10HNUMBER OF ,               SUMM 180
     3         8HWEIGHTED,8X,8HWEIGHTED/1H ,41X,4HTYPE,16X,9HRESIDUALS,SUMM 181
     4         15X,3HRMS/)                                           SUMM 182
  6000 FORMAT(1H ,40X,A6,15X,I5,15X,F10.3)                            SUMM 183
  6100 FORMAT(1H ,43X,11HRMS FOR ALL,I5,22H WEIGHTED MEASUREMENTS,F10.3) SUMM 184
       END                                                            SUMM 185
```


ENTRY
STAINF
CALCULATE STATISTICS
LOOP THROUGH 1500 FOR EACH SATELLITE
LOOP THROUGH 1560 FOR EACH STATION (OR BASELINE) (OR BASELINE FOR TWO STATION DATA) AND EACH SATELLITE
100
STAINF
OBTAIN STATISTICS FOR EACH STATION
LOOP THROUGH 1400 FOR EACH MEASUREMENT TYPE
A
A
CONVERT UNITS FOR OUTPUT
900
PRINT STATISTICS
STAINF
COMPUTE STATISTICS BY MEASUREMENT TYPE ONLY
WRITE TOTAL RMS
PRINT EDIT LEVEL
B


B
PRINT SUMMARY BY MEASUREMENT TYPE
RETURN

SUNGRV

DESCRIPTION

Subroutine SUNGRV evaluates the accelerations of the satellite due to the gravitational potentials of the Moon, the Sun, and the Planets, Venus, Mars, Jupiter, and Saturn. Intermediate data is stored in COMMON MOONGR for including the effects of these potentials in the variational equation computations in VEVAL.

NAME	SUNGRV
PURPOSE	TO COMPUTE GRAVITATIONAL ACCELERATIONS DUE TO : 1) MOON 2) SUN 3) VENUS 4) MARS 5) JUPITER 6) SATURN
CALLING SEQUENCE	CALL SUNGRV(DX)

SYMBOL	TYPE	DESCRIPTION
DX (3)	DP	INPUT & OUTPUT - ACCELERATION VECTOR

SUBROUTINES USED	DOTPRD
COMMON BLOCKS	CEPHEM MOONGR INTBLK XYZ
INPUT FILES	NONE
OUTPUT FILES	NONE

```
      SUBROUTINE SUNGRV(DX)                                              SUNG  29
      IMPLICIT REAL*8 (A-H,O-Z)                                          SUNG  30
      DIMENSION DX(3)                                                    SUNG  31
      COMMON/CEPHEM/UVBODY(4,6),EQ(644)                                  SUNG  32
      COMMON/MOONGR/DPXUV(6),RHOM(3,6),RHOSQ(6),RHO3(6)                  SUNG  33
      COMMON/INTBLK/THDOT1(9),GM3(6),IB(101),NBODY                       SUNG  34
      COMMON/XYZ/X(6),R,RSQ,ISAT,IFORCE(2)                               SUNG  35
      DO 20 N=1,NBODY                                                    SUNG  36
      DPXUV(N)=DOTPRD(X,UVBODY(1,N))                                     SUNG  37
      RRBODY=UVBODY(4,N)**2                                              SUNG  38
      RHOSQ(N)=RRBODY-2.0D0*DPXUV(N)*UVBODY(4,N)+RSQ                     SUNG  39
      RHO3(N)=RHOSQ(N)*DSQRT(RHOSQ(N))                                   SUNG  40
      DO 20 I=1,3                                                        SUNG  41
      RHOM(I,N)=X(I)-UVBODY(I,N)*UVBODY(4,N)                             SUNG  42
   20 DX(I)=DX(I)-GM3(N)*(RHOM(I,N)/RHO3(N)+UVBODY(I,N)/RRBODY)          SUNG  43
      RETURN                                                             SUNG  44
      END                                                                SUNG  45
```

SURDEN

DESCRIPTION

This subroutine computes the gravitational acceleration due to surface density blocks and the partial derivatives of this acceleration with respect to adjusted surface density parameters. These partials include the constraints affecting the adjustment of surface densities as described in the GEODYN Systems Description, Volume 1.

NAME — SURDEN

ENTRY POINT — PURPOSE

SURDN1 — INITIALIZATION

SURDEN — TO COMPUTE :
1) THE GRADIENT OF THE POTENTIAL DUE TO SURFACE DENSITIES
2) THE PARTIALS OF THE GRADIENTS WITH RESPECT TO THE SURFACE DENSITIES FOR THOSE DENSITIES TO BE ADJUSTED

CALLING SEQUENCE — CALL SURDN1(DENSE,AREA,CENTER,PART,DENCON)

SYMBOL	TYPE	DESCRIPTION
DENSE (1)	DP	INPUT - ARRAY CONTAINING THE SURFACE DENSITIES OF THE BLOCKS
AREA	DP	INPUT - ARRAY CONTAINING THE SURFACE AREAS OF THE SUB-BLOCKS
CENTER (3,1)	DP	INPUT - ARRAY CONTAINING THE EARTH FIXED X,Y,Z COORDINATES OF THE SUB-BLOCK CENTERS
PART (3,1)	DP	OUTPUT - ARRAY CONTAINING THE PARTIALS FOR THE BLOCKS TO BE ADJUSTED
DENCON (NCONST,1)	DP	INPUT - COEFFICIENTS RELATING CONSTRAINED AND UNCONSTRAINED ADJUSTED SURFACE DENSITIES

CALLING SEQUENCE — CALL SURDEN(FCT,THETG)

SYMBOL	TYPE	DESCRIPTION
FCT (3)	DP	INPUT & OUTPUT - ARRAY CONTAINING THE INERTIAL X,Y,Z COORDINATES OF THE GRADIENT TO BE UPDATED
THETG	DP	INPUT - RIGHT ASCENSION OF GREENWICH

SUBROUTINES USED — XEFIX, YEFIX, XINERT, YINERT

COMMON BLOCKS — CPARAM, CSTHET, INTBLK, XYZ

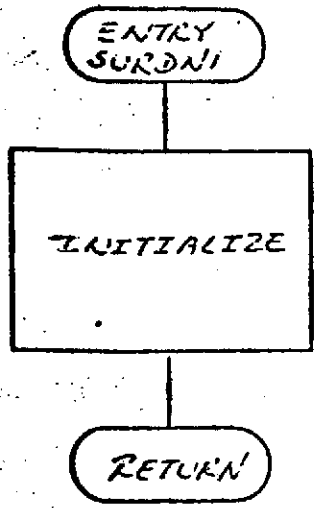
INPUT FILES — NONE

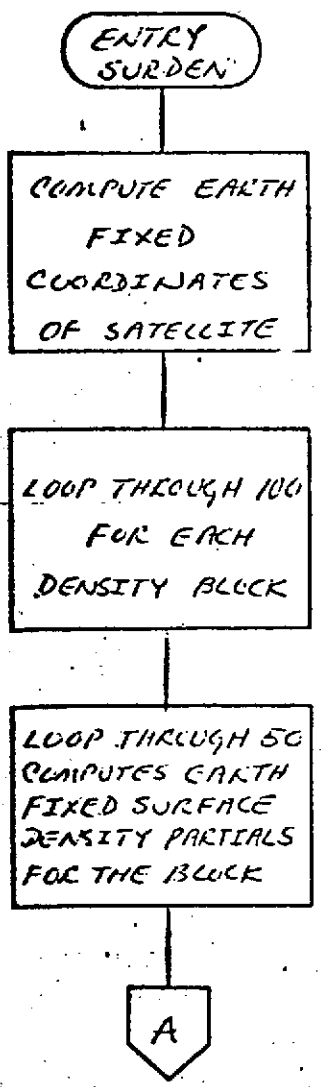
OUTPUT FILES — NONE

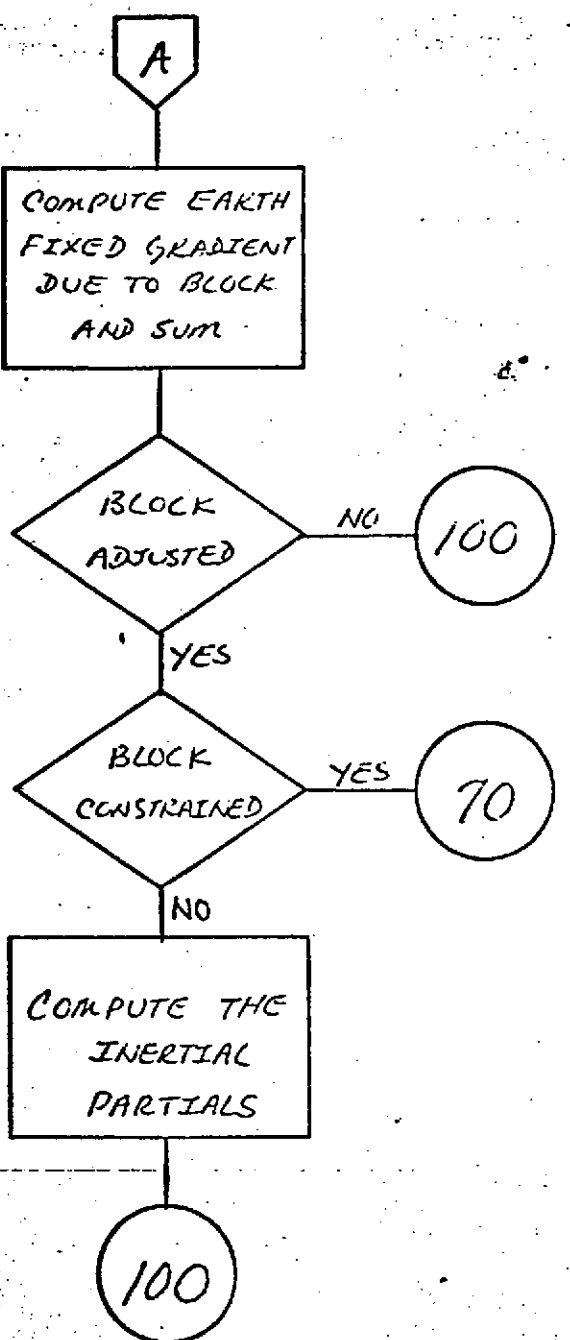
REFERENCES — "GEODYN SYSTEMS DESCRIPTION"
VOLUME 1 - GEODYN DOCUMENTATION

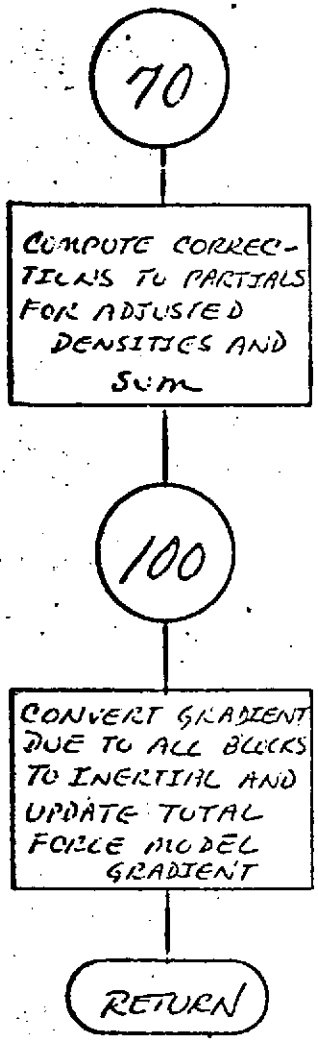
```
C                                                                       SURD  56
      SUBROUTINE SURDEN1(DENSE,AREA,CENTER,PART,DENCON)                 SURD  57
      IMPLICIT REAL*8(A-H,O-Z)                                          SURD  58
      LOGICAL CMPGPR,INNRSW                                             SURD  59
      DIMENSION DENSE(1),AREA(1),CENTER(3,1),GRAD(3),PART(3,1),PAR(3),  SURD  60
     .   PVEC(3),DP(3),FCT(3),DENCON(NCONST,1)                          SURD  61
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,NGPCOM,    SURD  62
     .   NCSEST,CMPGPR,LIM1,LIM2,NBLOCK,NADJ,NTIDST,NTIDEN,INNRSW,      SURD  63
     .   NCONST,NDCONS                                                  SURD  64
      COMMON/CSTHET/CTHETG,STHETG                                       SURD  65
      COMMON/INTBLK/THDOT(58),NEQN(16)                                  SURD  66
      COMMON/XYZ/POINT(8),ISAT,IFORCE(2)                                SURD  67
      DATA NSUB/4/                                                      SURD  68
C INITIALIZE                                                            SURD  69
      NCON2=NADJ-NCONST                                                 SURD  70
      RETURN                                                            SURD  71
      ENTRY SURDEN(FCT,THETG)                                           SURD  72
      IF(NBLOCK.LE.0) RETURN                                            SURD  73
      CTHETG=DCOS(THETG)                                                SURD  74
      STHETG=DSIN(THETG)                                                SURD  75
C COMPUTE THE EARTH FIXED COORDINATES OF SATELLITE                      SURD  76
      PVEC(1)=XEFIX(POINT(1),POINT(2))                                  SURD  77
      PVEC(2)=YEFIX(POINT(1),POINT(2))                                  SURD  78
      PVEC(3)=POINT(3)                                                  SURD  79
      GRAD(1)=0.0D0                                                     SURD  80
      GRAD(2)=0.0D0                                                     SURD  81
      GRAD(3)=0.0D0                                                     SURD  82
      NC=1                                                              SURD  83
      K=NEQN(ISAT)-7-NCON2                                              SURD  84
      K1=K+1                                                            SURD  85
      K2=K+NCON2                                                        SURD  86
      DO 100 KK=1,NBLOCK                                                SURD  87
C LOOP THROUGH 100 FOR EACH DENSITY BLOCK                               SURD  88
      K=K+1                                                             SURD  89
      DO 20 J=1,3                                                       SURD  90
   20 PAR(J)=0.D0                                                       SURD  91
C LOOP THROUGH 50 COMPUTES EARTH FIXED SURFACE DENSITY PARTIALS FOR     SURD  92
C THE BLOCK                                                             SURD  93
      DO 50 I=1,NSUB                                                    SURD  94
      DMAG=0.0D0                                                        SURD  95
      DO 30 J=1,3                                                       SURD  96
      DP(J)=PVEC(J)-CENTER(J,NC)                                        SURD  97
   30 DMAG=DMAG+DP(J)**2                                                SURD  98
      C=AREA(NC)/(DMAG*DSQRT(DMAG))                                     SURD  99
      DO 40 J=1,3                                                       SURD 100
   40 PAR(J)=PAR(J)-C*DP(J)                                             SURD 101
   50 NC=NC+1                                                           SURD 102
      DO 60 J=1,3                                                       SURD 103
C COMPUTE EARTH FIXED GRADIENT DUE TO BLOCK AND SUM                     SURD 104
   60 GRAD(J)=GRAD(J)+DENSE(KK)*PAR(J)                                  SURD 105
      IF(KK.GT.NADJ) GO TO 100                                          SURD 106
      IF(.NOT.INNRSW) GO TO 100                                         SURD 107
      IF(KK.GT.NCON2) GO TO 70                                          SURD 108
C IF BLOCK ADJUSTED COMPUTE THE INERTIAL PARTIALS                       SURD 109
      PART(1,K)=XINERT(PAR(1),PAR(2))                                   SURD 110
      PART(2,K)=YINERT(PAR(1),PAR(2))                                   SURD 111
```

```
      PART(3,K)=PAR(3)                                                  SURD 112
      GO TO 100                                                         SURD 113
C IF BLOCK ADJUSTED THROUGH CONSTRAINTS COMPUTE CORRECTIONS TO          SURD 114
C PARTIALS FOR ADJUSTED DENSITIES AND SUM                               SURD 115
   70 DP(1)=XINERT(PAR(1),PAR(2))                                       SURD 116
      DP(2)=YINERT(PAR(1),PAR(2))                                       SURD 117
      DP(3)=PAR(3)                                                      SURD 118
      DO 80 I=K1,K2                                                     SURD 119
      I1=I-K1+1                                                         SURD 120
      PDCDA=DENCON(KK-NCON2,I1)                                         SURD 121
      PART(1,I)=PART(1,I)+PDCDA*DP(1)                                   SURD 122
      PART(2,I)=PART(2,I)+PDCDA*DP(2)                                   SURD 123
   80 PART(3,I)=PART(3,I)+PDCDA*DP(3)                                   SURD 124
  100 CONTINUE                                                          SURD 125
C CONVERT GRADIENT DUE TO ALL BLOCKS TO INERTIAL AND UPDATE TOTAL       SURD 126
C FORCE MODEL GRADIENT                                                  SURD 127
      FCT(1)=FCT(1)+XINERT(GRAD(1),GRAD(2))                             SURD 128
      FCT(2)=FCT(2)+YINERT(GRAD(1),GRAD(2))                             SURD 129
      FCT(3)=FCT(3)+GRAD(3)                                             SURD 130
      RETURN                                                            SURD 131
      END                                                               SURD 132
```


ENTRY SURDNI
INITIALIZE
RETURN


ENTRY SURDEN
COMPUTE EARTH FIXED COORDINATES OF SATELLITE
LOOP THROUGH 100 FOR EACH DENSITY BLOCK
LOOP THROUGH 50 COMPUTES EARTH FIXED SURFACE DENSITY PARTIALS FOR THE BLOCK
A


A
COMPUTE EARTH FIXED GRADIENT DUE TO BLOCK AND SUM
BLOCK ADJUSTED
NO
100
YES
BLOCK CONSTRAINED
YES
70
NO
COMPUTE THE INERTIAL PARTIALS
100


70
COMPUTE CORRECTIONS TO PARTIALS FOR ADJUSTED DENSITIES AND SUM
100
CONVERT GRADIENT DUE TO ALL BLOCKS TO INERTIAL AND UPDATE TOTAL FORCE MODEL GRADIENT
RETURN

SYMINV

DESCRIPTION

Suborutine SYMINV is a double precision matrix inversion routine designed specifically for inverting a compressed storage symmetric matrix such as is used by ESTIM.

The technique used is the method of partitioning. The initialization consists of inverting a 1 x 1. The routine then constructs successively larger (N x N) inverted partitions of the original matrix until the entire matrix has been inverted.

This routine destroys the input matrix. It also requires a double precision scratch storage vector of length equal to the row dimension of the matrix to be inverted.

NAME		SYMINV
PURPOSE		TO RECURSIVELY FIND INVERSE OF SYMMETRIC MATRIX
CALLING SEQUENCE		CALL SYMINV(SUM1,NDIM,NLIM,DELTA)
SYMBOL	TYPE	DESCRIPTION
SUM1 (1)	DP	INPUT - LOWER RECTANGULAR PART OF MATRIX TO BE INVERTED OUTPUT - LOWER RECTANGULAR PART OF INVERTED MATRIX
NDIM	I	INPUT - DIMENSION OF MATRIX
NLIM	I	INPUT - DIMENSION OF PARTITION TO BE INVERTED
DELTA (1)	DP	SCRATCH
SUBROUTINES USED		NONE
COMMON BLOCKS		NONE
INPUT FILES		NONE
OUTPUT FILES		NONE
REFERENCES		*GEODYN SYSTEMS DESCRIPTION* VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE SYMINV(SUM1,NDIM,NLIM,DELTA)                                  SYMI  34
      DOUBLE PRECISION SUM1(1),DELTA(1)                                        SYMI  35
C INITIALIZE BY FINDING INVERSE OF 1X1                                         SYMI  36
      SUM1(1)=1.000/SUM1(1)                                                    SYMI  37
      IF(NLIM.EQ.1) RETURN                                                     SYMI  38
      N1=NDIM-1                                                                SYMI  39
C RECURSIVELY FIND INVERSE OF NXN KNOWING INVERSE OF (N-1)X(N-1) UNTIL         SYMI  40
C THE INVERSE OF AN NLIM X NLIM SQUARE PARTITION IS FOUND                      SYMI  41
      DO 400 N=2,NLIM                                                          SYMI  42
      NM1=N-1                                                                  SYMI  43
      L1=0                                                                     SYMI  44
      DO 100 L=1,NM1                                                           SYMI  45
      J1=0                                                                     SYMI  46
      DELTA(L)=0.000                                                           SYMI  47
      DO 60 J=1,L                                                              SYMI  48
      JL=J1+L                                                                  SYMI  49
      JN=J1+N                                                                  SYMI  50
      DELTA(L)=DELTA(L)+SUM1(JL)*SUM1(JN)                                      SYMI  51
   60 J1=J1+NDIM-J                                                             SYMI  52
      IF(L.EQ.NM1) GO TO 100                                                   SYMI  53
      LP1=L+1                                                                  SYMI  54
      DO 80 J=LP1,NM1                                                          SYMI  55
```

```
      JN=J1+N                                                          SYMI  56
      JL=L1+J                                                          SYMI  57
      DELTA(L)=DELTA(L)+SUM1(JL)*SUM1(JN)                              SYMI  58
   80 J1=J1+NDIM-J                                                     SYMI  59
  100 L1=L1+NDIM-L                                                     SYMI  60
      J1=N                                                             SYMI  61
      NN=N1+N                                                          SYMI  62
      DO 150 J=1,NM1                                                   SYMI  63
      SUM1(NN)=SUM1(NN)-DELTA(J)*SUM1(J1)                              SYMI  64
  150 J1=J1+NDIM-J                                                     SYMI  65
      SUM1(NN)=1.0D0/SUM1(NN)                                          SYMI  66
      J1=N                                                             SYMI  67
      DO 200 J=1,NM1                                                   SYMI  68
      SUM1(J1)=-DELTA(J)*SUM1(NN)                                      SYMI  69
  200 J1=J1+NDIM-J                                                     SYMI  70
      I1=N                                                             SYMI  71
      DO 300 I=1,NM1                                                   SYMI  72
      J1=I                                                             SYMI  73
      DO 250 J=1,I                                                     SYMI  74
      SUM1(J1)=SUM1(J1)-SUM1(I1)*DELTA(J)                              SYMI  75
  250 J1=J1+NDIM-J                                                     SYMI  76
  300 I1=I1+NDIM-I                                                     SYMI  77
  400 N1=N1+NDIM-N                                                     SYMI  78
      RETURN                                                           SYMI  79
      END                                                              SYMI  80
```

TDIF

DESCRIPTION

TDIF computes the differences in seconds between the time systems

UT1, UT2,UTC, and A.1.

Tabular informaton relating A.1 and UTC is required by TDIF and must periodically be updated. This data is available from the U.S. Naval Observatory.

Tabular information relating A.1 and UT1 is required by TDIF. This data also must be periodically update. A full description of the method used to update this table is available in GEODYN Support Programs, Volume IV.

NAME — TDIF

PURPOSE — TO COMPUTE THE DIFFERENCE IN SECONDS BETWEEN ANY TWO OF THE FOLLOWING TIME SYSTEMS
A.1, UTC, UT1, UT2

CALLING SEQUENCE — X=TDIF(BASE,IN,DAY0)

SYMBOL	TYPE	DESCRIPTION
BASE	I	INPUT - DESIRED TIME SYSTEM : (1=UT1, 2=UT2, 3=UTC, 4=A.1)
IN	I	INPUT - CURRENT TIME SYSTEM : (1=UT1, 2=UT2, 3=UTC, 4=A.1)
DAYC	DP	INPUT - CURRENT TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
TDIF	R	OUTPUT - COMPUTED TIME DIFFERENCE IN SECONDS

COMMON BLOCKS — INITBK

REFERENCES — *GEODYN SYSTEMS DESCRIPTION*
VOLUME 1 - GEODYN DOCUMENTATION

```
      REAL FUNCTION TDIF(BASE,IN,DAYC)                                  TDIF  30
      DOUBLE PRECISION DAY0,REFTIM,YMDAY,DAYS,UTCT,TWOPI                TDIF  31
      INTEGER BASE,ID(4)                                                TDIF  32
      COMMON/INITBK/IG1(55),NOTIST,IG2                                  TDIF  33
      LOGICAL NOTIST                                                    TDIF  34
      DIMENSION UTCT(53),UTCDIF(52),UTCRT(52),A1T1(613),B1T1(152),      TDIF  35
     *   C1T1(152),D1T1(152)                                            TDIF  36
      EQUIVALENCE (B1T1(1),A1T1(158)),(C1T1(1),A1T1(310)),              TDIF  37
     *   (D1T1(1),A1T1(462))                                            TDIF  38
      DATA ID/2,1,4,3/                                                  TDIF  39
      DATA TWOPI/6.28318500/                                            TDIF  40
      DATA LLIMIT/528/                                                  TDIF  41
C TABLE A.1-UT1 -- AT 10 DAY INTERVALS FROM 570917 TO 720303            TDIF  42
      DATA A1T1/                                                        TDIF  43
     *  -.1649,-.1513,-.1361,-.1193,-.1013,-.0837,-.0649,-.0458,-.0270, TDIF  44
     *  -.0096,0.0098,0.0285,0.0457,0.0618,0.0784,0.0948,0.1111,0.1277, TDIF  45
     *  0.1450,0.1629,0.1815,0.1993,0.2176,0.2351,0.2517,0.2665,0.2789, TDIF  46
     *  0.2879,0.2941,0.3063,0.3131,0.3189,0.3249,0.3310,0.3378,0.3458, TDIF  47
     *  0.3554,0.3662,0.3794,0.3942,0.4098,0.4261,0.4431,0.4596,0.4765, TDIF  48
     *  0.4928,0.5093,0.5247,0.5401,0.5548,0.5727,0.5866,0.6000,0.6136, TDIF  49
     *  0.6267,0.6405,0.6548,0.6694,0.6833,0.6981,0.7115,0.7243,0.7364, TDIF  50
     *  0.7471,0.7572,0.7660,0.7734,0.7806,0.7869,0.7937,0.8007,0.8094, TDIF  51
     *  0.8202,0.8319,0.8450,0.8606,0.8770,0.8945,0.9116,0.9287,0.9474, TDIF  52
     *  0.9639,0.9795,0.9942,1.0080,1.0209,1.0335,1.0455,1.0605,1.0753, TDIF  53
     *  1.0909,1.1059,1.1202,1.1340,1.1543,1.1708,1.1854,1.2007,1.2138, TDIF  54
     *  1.2253,1.2343,1.2418,1.2479,1.2526,1.2565,1.2596,1.2634,1.2689, TDIF  55
```

```
     1.2762,1.2858,1.2968,1.3094,1.3226,1.3364,1.3507,1.3652,1.3801,  TDIF  56
     1.3947,1.4094,1.4230,1.4363,1.4488,1.4601,1.4691,1.4778,1.4871,  TDIF  57
     1.4983,1.5139,1.5320,1.5495,1.5638,1.5758,1.5869,1.5978,1.6105,  TDIF  58
     1.6223,1.6331,1.6422,1.6499,1.6563,1.6620,1.6672,1.6725,1.6783,  TDIF  59
     1.6850,1.6930,1.7023,1.7130,1.7249,1.7380,1.7521,1.7665,1.7813,  TDIF  60
     1.7959,1.8103,1.8239,1.8365/                                     TDIF  61
                                                                      TDIF  62
    DATA BIT1/
     1.3497, 1.8630, 1.8758, 1.8891, 1.9015, 1.9164, 1.9314, 1.9468,  TDIF  63
     1.9623, 1.9781, 1.9947, 2.0118, 2.0285, 2.0446, 2.0598, 2.0724,  TDIF  64
     2.0820, 2.0883, 2.0938, 2.0994, 2.1046, 2.1104, 2.1165, 2.1237,  TDIF  65
     2.1320, 2.1425, 2.1553, 2.1696, 2.1843, 2.1997, 2.2169, 2.2346,  TDIF  66
     2.2516, 2.2685, 2.2864, 2.3029, 2.3155, 2.3236, 2.3399, 2.3514,  TDIF  67
     2.3632, 2.3746, 2.3872, 2.4026, 2.4192, 2.4352, 2.4542, 2.4731,  TDIF  68
     2.4915, 2.5091, 2.5263, 2.5423, 2.5567, 2.5695, 2.5806, 2.5887,  TDIF  69
     2.5948, 2.6005, 2.6084, 2.6199, 2.6326, 2.6462, 2.6616, 2.6789,  TDIF  70
     2.6973, 2.7182, 2.7409, 2.7639, 2.7847, 2.8043, 2.8253, 2.8447,  TDIF  71
     2.8581, 2.8893, 2.9103, 2.9309, 2.9509, 2.9710, 2.9914, 3.0128,  TDIF  72
     3.0346, 3.0565, 3.0782, 3.1004, 3.1227, 3.1446, 3.1665, 3.1876,  TDIF  73
     3.2079, 3.2281, 3.2412, 3.2526, 3.2632, 3.2744, 3.2859, 3.2981,  TDIF  74
     3.3103, 3.3253, 3.3417, 3.3604, 3.3810, 3.4038, 3.4268, 3.4487,  TDIF  75
     3.4713, 3.4954, 3.5194, 3.5420, 3.5641, 3.5850, 3.6051, 3.6252,  TDIF  76
     3.6457, 3.6667, 3.6883, 3.7111, 3.7352, 3.7608, 3.7886, 3.8162,  TDIF  77
     3.8411, 3.8643, 3.8870, 3.9097, 3.9325, 3.9541, 3.9744, 3.9928,  TDIF  78
     4.0097, 4.0256, 4.0414, 4.0574, 4.0740, 4.0918, 4.1108, 4.1316,  TDIF  79
     4.1554, 4.1800, 4.2049, 4.2300, 4.2558, 4.2825, 4.3095, 4.3349,  TDIF  80
     4.3592, 4.3817, 4.4059, 4.4291, 4.4536, 4.4781, 4.5021, 4.5259/  TDIF  81
                                                                      TDIF  82
    DATA C1T1/
     4.5497, 4.5744, 4.6015, 4.6295, 4.6579, 4.6864, 4.7150, 4.7428,  TDIF  83
     4.7689, 4.7929, 4.8144, 4.8333, 4.8505, 4.8669, 4.8834, 4.9005,  TDIF  84
     4.9195, 4.9398, 4.9617, 4.9858, 5.0110, 5.0372, 5.0651, 5.0932,  TDIF  85
     5.1211, 5.1487, 5.1765, 5.2031, 5.2281, 5.2524, 5.2762, 5.2997,  TDIF  86
     5.3228, 5.3461, 5.3701, 5.3951, 5.4210, 5.4476, 5.4745, 5.5015,  TDIF  87
     5.5290, 5.5568, 5.5855, 5.6139, 5.6404, 5.6644, 5.6855, 5.7066,  TDIF  88
     5.7252, 5.7422, 5.7583, 5.7742, 5.7909, 5.8089, 5.8280, 5.8471,  TDIF  89
     5.8673, 5.8891, 5.9128, 5.9384, 5.9652, 5.9923, 6.0200, 6.0477,  TDIF  90
     6.0742, 6.0996, 6.1239, 6.1487, 6.1743, 6.2001, 6.2260, 6.2523,  TDIF  91
     6.2793, 6.3064, 6.3324, 6.3567, 6.3806, 6.4119, 6.4421, 6.4684,  TDIF  92
     6.4923, 6.5175, 6.5422, 6.5673, 6.5880, 6.6052, 6.6243, 6.6436,  TDIF  93
     6.6633, 6.6841, 6.7061, 6.7295, 6.7539, 6.7789, 6.8047, 6.8311,  TDIF  94
     6.8574, 6.8837, 6.9108, 6.9392, 6.9673, 6.9949, 7.0212, 7.0464,  TDIF  95
     7.0710, 7.0955, 7.1207, 7.1467, 7.1744, 7.2042, 7.2350, 7.2659,  TDIF  96
     7.2963, 7.3270, 7.3588, 7.3903, 7.4202, 7.4481, 7.4751, 7.5015,  TDIF  97
     7.5261, 7.5486, 7.5687, 7.5879, 7.6074, 7.6268, 7.6470, 7.6688,  TDIF  98
     7.6924, 7.7191, 7.7479, 7.7768, 7.8051, 7.8337, 7.8632, 7.8924,  TDIF  99
     7.9210, 7.9498, 7.9790, 8.0089, 8.0381, 8.0668, 8.0953, 8.1237,  TDIF 100
     8.1527, 8.1828, 8.2143, 8.2464, 8.2786, 8.3104, 8.3421, 8.3738/  TDIF 101
                                                                      TDIF 102
    DATA C1T1/
     8.4050, 8.4350, 8.4673, 8.4893, 8.5136, 8.5365, 8.5583, 8.5789,  TDIF 103
     8.5982, 8.6166, 8.6351, 8.6547, 8.6757, 8.6983, 8.7228, 8.7493,  TDIF 104
     8.7779, 8.8075, 8.8379, 8.8695, 8.8989, 8.9285, 8.9575, 8.9854,  TDIF 105
     9.0122, 9.0380, 9.0635, 9.0896, 9.1151, 9.1403, 9.1663, 9.1935,  TDIF 106
     9.2233, 9.2546, 9.2847, 9.3201, 9.3517, 9.3849, 9.4172, 9.4478,  TDIF 107
     9.4765, 9.5032, 9.5270, 9.5499, 9.5743, 9.5992, 9.6233, 9.6445,  TDIF 108
     9.6704, 9.6975, 9.7243, 9.7560, 9.7873, 9.8211, 9.8564, 9.8928,  TDIF 109
     9.9294, 9.9648, 9.9982,10.0281,10.0556,10.0834,10.1137,10.1449,  TDIF 110
    10.1769,10.2114,10.2451,10.2784/                                  TDIF 111
```

```
C DAYS FROM JAN 1, 1966 FOR UTC CLOCK CORRECTIONS                       TDIF 112
      DATA UTCT/                                                        TDIF 113
     *-3586.208333333330,-3565.208333333330,-3445.208333333330,        TDIF 114
     *-3419.208333333330,-3390.208333333330,-3348.208333333330,        TDIF 115
     *-3334.208333333330,-3264.208333333330,-3215.208333333330,        TDIF 116
     *-3165.208333333330,-3131.208333333330,-3117.208333333330,        TDIF 117
     *-3103.208333333330,-3089.208333333330,-3061.208333333330,        TDIF 118
     *-2998.208333333330,-2977.208333333330,-2942.208333333330,        TDIF 119
     *-2907.208333333330,-2886.208333333330,-2872.208333333330,        TDIF 120
     *-2823.208333333330,-2760.208333333330,-2739.208333333330,        TDIF 121
     *-2725.208333333330,-2627.208333333330,-2592.208333333330,        TDIF 122
     *-2564.208333333330,-2529.208333333330,-2501.208333333330,        TDIF 123
     *-2340.208333333330,-2319.208333333330,-2284.208333333330,        TDIF 124
     *-2249.208333333330,-2235.208333333330,-2207.208333333330,        TDIF 125
     *-1826.D0,-1614.D0,-1460.789311759D0,-792.D0,-730.943661971830,    TDIF 126
     *-640.D0,-487.D0,-457.D0,-365.DC,-306.D0,-184.D0,-122.D0,0.CD0,    TDIF 127
     *761.CD0,2191.0D0,2373.0D0,9999.0D0/                               TDIF 128
C A.1 - UTC AFTER EACH UTC CLOCK CORRECTION                             TDIF 129
      DATA UTCDIF/                                                      TDIF 130
     *-7.61676E-1,-7.59279E-1,-6.61401E-1,-6.32826E-1,-6.04843E-1,      TDIF 131
     *-5.67623E-1,-5.42543E-1,-4.74538E-1,-4.19671E-1,-3.67990E-1,      TDIF 132
     *-3.25988E-1,-2.98414E-1,-2.70676E-1,-2.43070E-1,-2.06172E-1,      TDIF 133
     *-1.26720E-1,-8.76661E-2,-3.49597E-2, 1.84152E-2, 5.68497E-2,      TDIF 134
     * 8.83297E-2, 1.50575E-1, 2.24061E-1, 2.63269E-1, 2.96013E-1,      TDIF 135
     * 4.02225E-1, 4.51382E-1, 4.94829E-1, 5.46547E-1, 5.92719E-1,      TDIF 136
     * 7.59403E-1, 7.97601E-1, 8.48046E-1, 8.99091E-1, 9.31595E-1,      TDIF 137
     * 9.76413E-1, 1.46119E-0, 1.69407E-0, 1.89265E-0, 2.73230E-0,      TDIF 138
     * 2.80068E-0, 3.01863E-0, 3.31660E-0, 3.35650E-0, 3.57559E-0,      TDIF 139
     * 3.75200E-0, 4.01010E-0, 4.19045E-0, 4.35010E-0, 6.21980E-0,      TDIF 140
     * 10.0343817E-0,11.0343817E-0/                                     TDIF 141
      DATA UTCRT/                                                       TDIF 142
C A.1 - UTC RATE OF SEPARATION AFTER EACH UTC CLOCK CORRECTION          TDIF 143
     *1.06651E-3,6.48989E-4,3.17592E-4,2.85086E-4,4.10020E-4,3.62794E-4,TDIF 144
     *6.85787E-4,7.11582E-4,6.46546E-4,6.28629E-4,5.40983E-4,5.52709E-4,TDIF 145
     *5.43303E-4,6.03510E-4,9.43680E-4,9.07322E-4,9.34465E-4,9.53572E-4,TDIF 146
     *8.77939E-4,8.19998E-4,8.62181E-4,8.49298E-4,9.13707E-4,9.10296E-4,TDIF 147
     *8.79719E-4,8.33035E-4,8.37410E-4,9.06216E-4,9.34700E-4,9.11123E-4,TDIF 148
     *8.66292E-4,8.69835E-4,8.97014E-4,8.93154E-4,8.86354E-4,1.25857E-3,TDIF 149
     *1.28717E-3,1.29643E-3,1.12091E-3,1.12000E-3,1.29750E-3,1.29392E-3,TDIF 150
     *1.29664E-3,1.29434E-3,1.29525E-3,1.29590E-3,1.29597E-3,1.29631E-3,TDIF 151
     * 2.56863E-3,2.59200E-3,0.0,0.0,0.0/                               TDIF 152
      DATA NUTC/52/                                                     TDIF 153
      IF(NOT1ST) GO TO 5                                                TDIF 154
      NOT1ST=.TRUE.                                                     TDIF 155
      REFTIM=YMDAY(660101,0,0,0D0)                                      TDIF 156
      LIMIT=LLIMIT-1                                                    TDIF 157
 5    TDIF=0.                                                           TDIF 158
      IF(BASE.EQ.IN) RETURN                                             TDIF 159
      IF(BASE.GT.4.OR.IN.GT.4) RETURN                                   TDIF 160
      IF(BASE.LE.0.OR.IN.LE.0) RETURN                                   TDIF 161
      DAYS=DAYS-REFTIM                                                  TDIF 162
C TIME CORRECTIONS IN ORDER                                             TDIF 163
C  UT2-UT1 TO UT1-A.1 TO A.1-UTC                                        TDIF 164
      I1=ID(IN)                                                         TDIF 165
      I2=ID(BASE)                                                       TDIF 166
      JI=MIN0(I1,I2)                                                    TDIF 167
```

```
      J2=MAXO(I1,I2)                                                    TDIF 168
C START CORRECTION CALCULATION WITH SMALLER TIME BASE                   TDIF 169
      GO TO (10,20,30,50),J1                                            TDIF 170
C COMPUTE UT2-UT1                                                       TDIF 171
10    TWOPID=T*OPI*(DAYS-.201D0)/365.2422D0                             TDIF 172
      FORPID=2.*TWOPID                                                  TDIF 173
      TDIF=TDIF+.022*SIN(TWOPID)-.012*COS(TWOPID)                       TDIF 174
     .          -.006*SIN(FORPID)+.007*COS(FORPID)                     TDIF 175
C TEST FOR OUTPUT TIME SYSTEM                                           TDIF 176
      IF(J2.EQ.2) GO TO 50                                              TDIF 177
C COMPUTE UT1-A.1                                                       TDIF 178
20    DT=(DAYS+3037.0D0)*1.CD-1                                         TDIF 179
      I=MINO(LIMIT,MAX1(1.,DT))                                         TDIF 180
      DT=DT-FLOAT(I)                                                    TDIF 181
      TDIF=TDIF-A1T1(I)-DT*(A1T1(I+1)-A1T1(I))                          TDIF 182
      IF(J2.EQ.3) GO TO 50                                              TDIF 183
C COMPUTE A.1-UTC                                                       TDIF 184
30    DO 40 I=1,NUTC                                                    TDIF 185
      IF(DAYS.LT.UTCT(I+1)) GO TO 45                                    TDIF 186
40    CONTINUE                                                          TDIF 187
      I=NUTC                                                            TDIF 188
45    TDIF=TDIF+UTCDIF(I)+UTCRT(I)*(DAYS-UTCT(I))                       TDIF 189
C SET DIRECTION OF CORRECTION                                           TDIF 190
50    IF(I1.EQ.J1)TDIF=-TDIF                                            TDIF 191
      RETURN                                                            TDIF 192
      END                                                               TDIF 193
```

TIDAL

DESCRIPTION

TIDAL evaluates the acceleration on a satellite caused by the Earth tidal bulge resulting from Lunar and Solar gravitational effects on the Earth.

NAME	TIDAL
PURPOSE	TO COMPUTE ACCELERATION DUE TO SOLID EARTH TIDAL BULGES CAUSED BY LUNAR AND SOLAR GRAVITATIONAL EFFECTS ON EARTH. USES ONLY K2 IN SPHERICAL HARMONIC EXPANSION
CALLING SEQUENCE	CALL TIDAL(DX)

SYMBOL	TYPE	DESCRIPTION
DX (3)	DP	INPUT & OUTPUT - SATELLITE ACCELERATION VECTOR

SUBROUTINES USED	NONE
COMMON BLOCKS	CEPHEM INITBK INTBLK MOONGR XYZ
INPUT FILES	NONE
OUTPUT FILES	NONE
REFERENCES	"GEODYN SYSTEMS DESCRIPTION" VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE TIDAL(DX)                                              TIDA  29
      IMPLICIT REAL*8 (A-H,O-Z)                                         TIDA  30
      LOGICAL NOTIST                                                    TIDA  31
      DOUBLE PRECISION K2,K3,LAMBDA                                     TIDA  32
      DIMENSION CONST(2),ACCEL(3),DX(3)                                 TIDA  33
      COMMON/CEPHEM/UVBODY(24),E0(644)                                  TIDA  34
      COMMON/INITBK/IG1(56),NOTIST                                      TIDA  35
      COMMON/INTBLK/THETS(3),GM,AE,AESQ(4),RATIOM(53),K2,K3,LAMBDA,     TIDA  36
     .   TORFT                                                          TIDA  37
      COMMON/MOONGR/DPXUV(6),RHOM(3,6),RHOSQ(12)                        TIDA  38
      COMMON/XYZ/XYZ(6),R,RSQ,ISAT,IFORCE(2)                            TIDA  39
      IF(NOTIST) GO TO 10                                               TIDA  40
      DO 5 I=1,2                                                        TIDA  41
    5 CONST(I)=K2*AE*RATIOM(I)*0.5D0                                    TIDA  42
      NOTIST=.TRUE.                                                     TIDA  43
   10 K=0                                                               TIDA  44
      DO 100 I1=1,8,4                                                   TIDA  45
      K=K+1                                                             TIDA  46
      DP=DPXUV(K)/R                                                     TIDA  47
      DP3=3.0D0-15.0D0*DP**2                                            TIDA  48
      DO 20 I=1,3                                                       TIDA  49
      J=I1+I-1                                                          TIDA  50
   20 ACCEL(I)=DP*6.0D0*UVBODY(J)+DP3*XYZ(I)/R                          TIDA  51
      J1=I1+3                                                           TIDA  52
      RATIO4=(AE/R)**4/UVBODY(J1)**3                                    TIDA  53
      DO 100 I=1,3                                                      TIDA  54
  100 DX(I)=DX(I)+ACCEL(I)*CONST(K)*RATIO4                              TIDA  55
      RETURN                                                            TIDA  56
      END                                                               TIDA  57
```

NAME	TRUEP
ENTRY POINT	PURPOSE
TRUEP	INITIALIZATION
TRUEP	TO ROTATE TRACKING STATION COORDINATES TO ACCOUNT FOR POLAR WANDERING
CALLING SEQUENCE	CALL TRUEP1(STAXYZ)

SYMBOL	TYPE	DESCRIPTION
STAXYZ (3,1)	DP	INPUT - TRACKING STATION CARTESIAN COORDINATES

CALLING SEQUENCE	CALL TRUEP(DAY,ISTA)

SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT - OBSERVATION TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR FOR THE ARC
ISTA	I	INPUT - STATION INDEX

SUBROUTINES USED	POLE
COMMON BLOCKS	TRUPOL
INPUT FILES	NONE
OUTPUT FILES	NONE
REFERENCES	'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE TRUEP1(STAXYZ)                                          TRUE  40
      IMPLICIT REAL*8 (A-H,O-Z)                                          TRUE  41
      DOUBLE PRECISION MATRX(3,3)                                        TRUE  42
      DIMENSION STAXYZ(3,1)                                              TRUE  43
      COMMON/TRUPOL/TRUE(3)                                              TRUE  44
      DATA MATRX/1.D0,3*0.D0,1.D0,3*0.D0,1.D0/                           TRUE  45
      RETURN                                                             TRUE  46
      ENTRY TRUEP(DAY,ISTA)                                              TRUE  47
C COMPUTE POLE COORDINATES                                               TRUE  48
      CALL POLE (X,Y,DAY)                                                TRUE  49
C ROTATION MATRIX                                                        TRUE  50
      MATRX(1,3) = -X                                                    TRUE  51
      MATRX(2,1) = X*Y                                                   TRUE  52
      MATRX(2,3) = Y                                                     TRUE  53
      MATRX(3,1) = X                                                     TRUE  54
      MATRX(3,2) = -Y                                                    TRUE  55
```

```
C ROTATE COORDINATES                                                    TRUE  56
      DO 100 I=1,3                                                      TRUE  57
      TRUE(I) = 0.D0                                                    TRUE  58
      DO 100 J=1,3                                                      TRUE  59
      TRUE(I)=TRUE(I)+MATR X(I,J)*STAXYZ(J,ISTA)                        TRUE  60
  100 CONTINUE                                                          TRUE  61
      RETURN                                                            TRUE  62
      END                                                               TRUE  63
```

TWOSTA

DESCRIPTION

Subroutine TWOSTA computes the residuals and partial derivatives for observations involving two tracking stations. Subroutine UPDOWN is used by TWOSTA to compute up- and down-link signal transit times necessary in the computation of precise transmission and transponder relay times. The partial derivatives and calculated measurements are sums and/or differences of equivalent range and range rate quantities computed by subroutine PREDCT.

The observation types by program index are:

27) VLBI time dalay, τ_g

28) VLBI fringe rate, ν_F

29) Two-way average range rate, $\dot{\bar{\rho}}_2$, and

30) Three-way average range rate, $\dot{\bar{\rho}}_3$.

NAME	TWOSTA
ENTRY POINT	PURPOSE
TWOST1	INITIALIZATION
TWOSTA	TO COMPUTE MEASUREMENTS & PARTIALS FOR VLBI & AVERAGE RANGE RATE DATA

CALLING SEQUENCE CALL TWOST1(PMPX0,NPARM)

SYMBOL	TYPE	DESCRIPTION
PMPX0 (NPARM,1)	DP	OUTPUT - MEASUREMENT PARTIALS WITH RESPECT TO EPOCH PARAMETERS
NPARM	I	INPUT - MAXIMUM NUMBER OF PARAMETERS PER MEASUREMENT

CALLING SEQUENCE CALL TWOSTA(ISTA,DAY2,RESID,DATASW)

SYMBOL	TYPE	DESCRIPTION
ISTA	I	INPUT - STATION INDEX
DAY2	DP	INPUT - MEASUREMENT TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
RESID	DP	INPUT - MEASUREMENT RESIDUAL (O-C)
DATASW	L	INPUT - .TRUE. WHEN POSITION OF SATELLITE WANTED .FALSE. WHEN MEASUREMENT PARTIALS WANTED

SUBROUTINES USED	GRHRAN	UPDOWN	ORBIT	OBSDOT	PROCES
	PREDCT				
COMMON BLOCKS	CONSTS	CORB1	CUVECT	PREBLK	
INPUT FILES	NONE				
OUTPUT FILES	NONE				

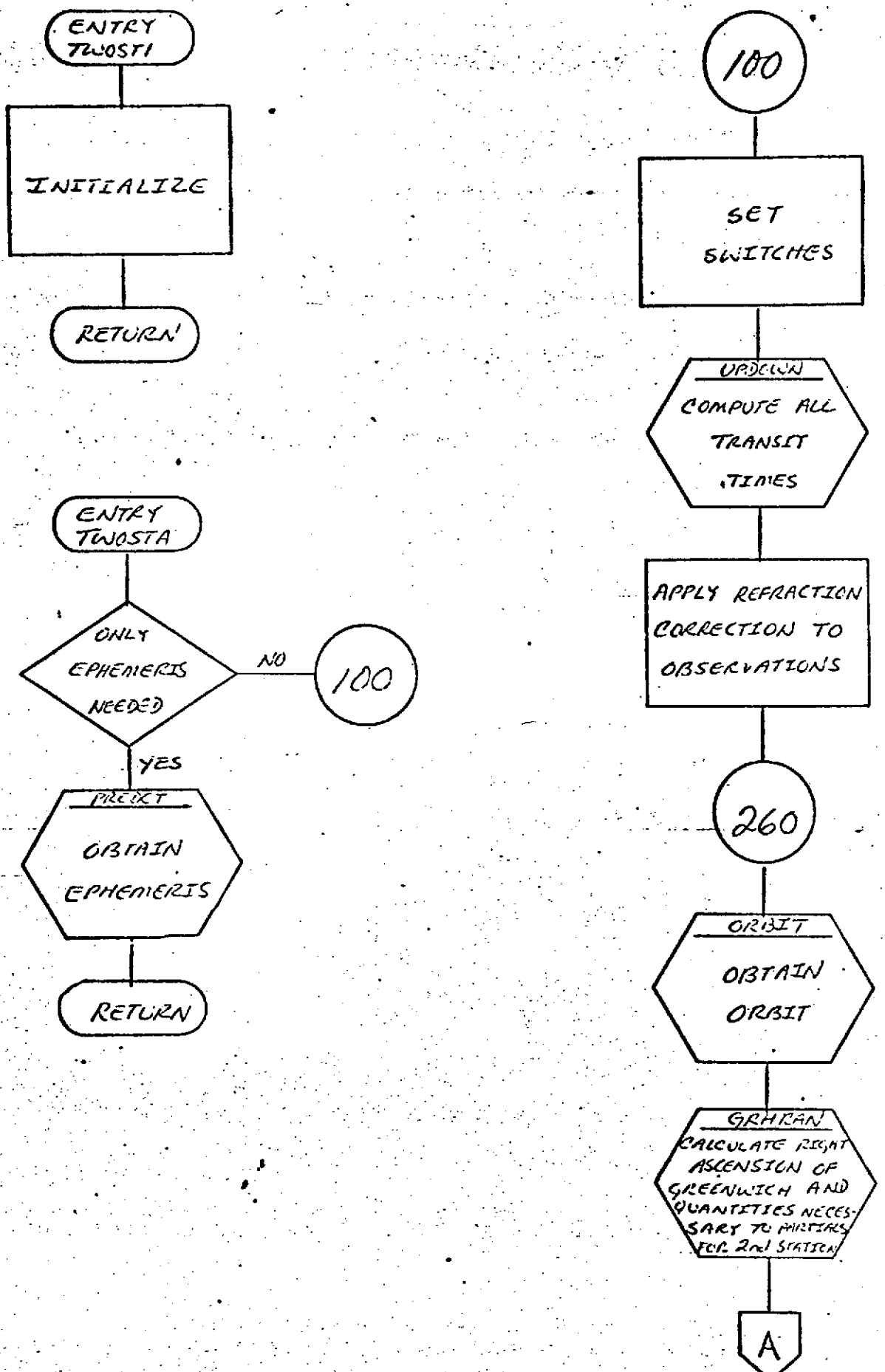
GEODYN SYSTEMS DESCRIPTION
VOLUME 1 - GEODYN DOCUMENTATION

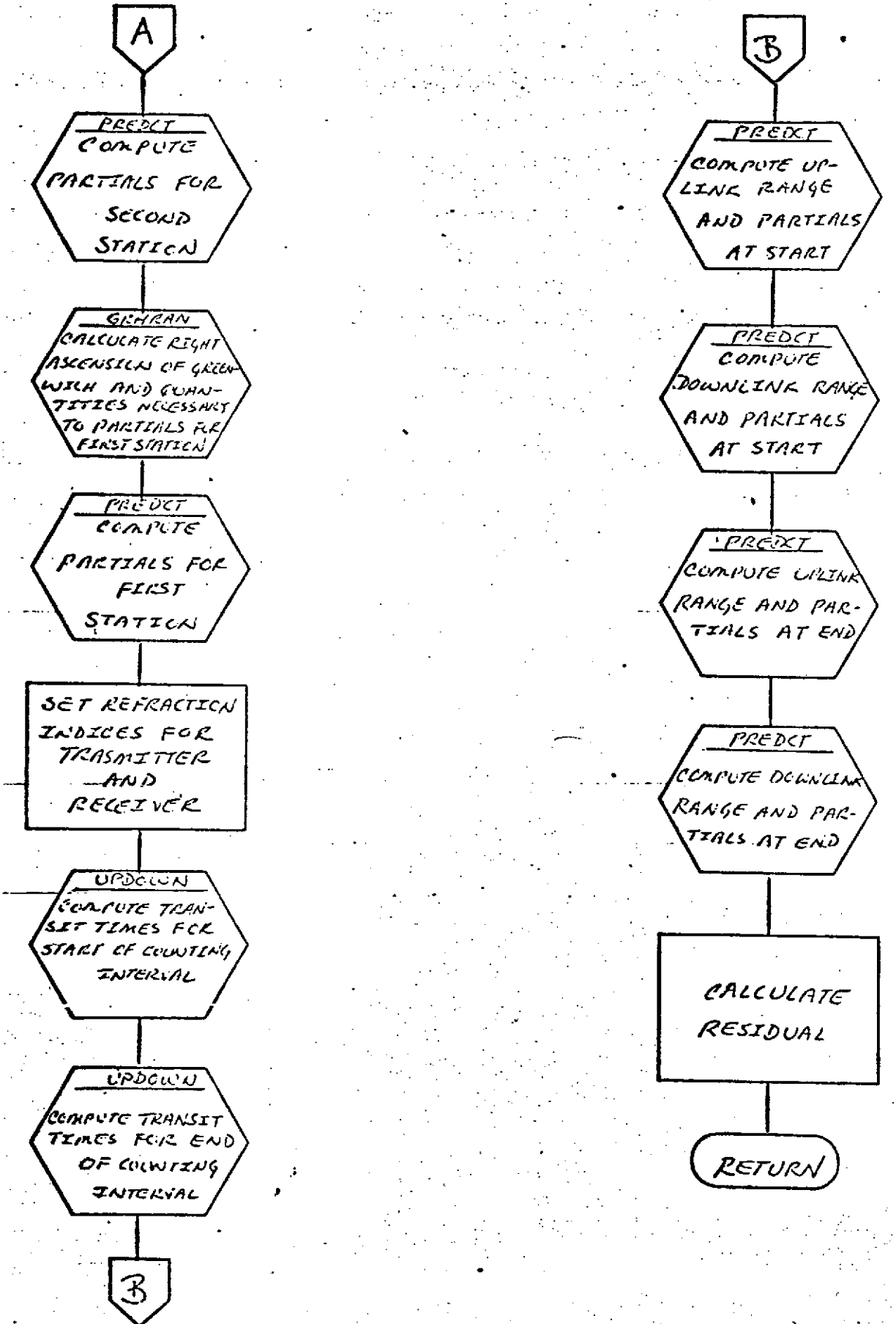
```
      SUBROUTINE TWOST1(PMPX0,NPARM)                                    TWOS  49
      IMPLICIT REAL*8 (A-H,O-Z)                                         TWOS  50
      LOGICAL*1 VHFCHN,PREPRO,SAVEPP                                    TWOS  51
      LOGICAL DATASW,LOGIC                                              TWOS  52
      INTEGER*2 MTYPE,NMEAS,PRETYP,ISN2,ISAT                            TWOS  53
      INTEGER RECNO                                                     TWOS  54
      DIMENSION PMPX0(NPARM,1),RENDX(2)                                 TWOS  55
```

```
      REAL D1D,D1U,D2D,D2U,RFNDX                                        TWOS  56
      COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC                               TWOS  57
      COMMON/CORB1/T,W,U,THETG,PERHT(2),APHT(2),PRD(2)                  TWOS  58
      COMMON/CUVECT/UHAT(3,2),XYZ(3,2),RXYZ(3,2),RFNV(3,2),R(2),        TWOS  59
     *    RSO(2),XYSO(2)                                               TWOS  60
      COMMON/PREBLK/DAYSAT,OBS,DAY1,SIG,D2D,D2U,D1D,D1U,ISN,MTYPE,      TWOS  61
     *    NMEAS,ISAT,PRETYP,ISN2,VHFCHN,PREPRO,RECNO                   TWOS  62
      DATA C/2.99792458/                                                TWOS  63
      EQUIVALENCE (DSTA1,DAY1),(DSTA2,D2D),(FREQ,D1D)                   TWOS  64
      EQUIVALENCE (RFNDX(1),REFRAC)                                     TWOS  65
C INITIALIZE                                                            TWOS  66
      NP3=NPARM-3                                                       TWOS  67
      NP6=NPARM-6                                                       TWOS  68
      RETURN                                                            TWOS  69
      ENTRY TWOSTA(ISTA,DAY2,RESID,DATASW)                              TWOS  70
      IF(.NOT.DATASW) GO TO 100                                         TWOS  71
C OBTAIN EPHEMERIS                                                      TWOS  72
      CALL PREDCT(ISTA,DAY2,RESID,RESID2,DATASW)                        TWOS  73
      RETURN                                                            TWOS  74
C SET SWITCHES                                                          TWOS  75
  100 IF(.NOT.PREPRO) PRETYP=0                                          TWOS  76
      IPRE1=PRETYP/10                                                   TWOS  77
      IPRE2=PRETYP-IPRE1*10                                             TWOS  78
      SAVEPP=PREPRO                                                     TWOS  79
      ISTA2=ISN2                                                        TWOS  80
      KTYPE=MTYPE-26                                                    TWOS  81
      F=1.000                                                           TWOS  82
      IF(PREPRO) GO TO (205,200,300,300),KTYPE                          TWOS  83
      GO TO (250,260,400,400),KTYPE                                     TWOS  84
C VLBI DATA                                                             TWOS  85
  200 F=DSTA2                                                           TWOS  86
  205 IF(IPRE1.LE.0) GO TO 240                                          TWOS  87
      DSTA1=DAY2                                                        TWOS  88
C COMPUTE VLBI TRANSIT TIMES                                            TWOS  89
      CALL UPDOWN(DAY2,DSTA1,DSTA2,1.0DC,ISTA,ISTA2,ISAT,.FALSE.,RCOR,  TWOS  90
     *    RFNDX)                                                       TWOS  91
  240 PREPRO=IPRE2.GT.0                                                 TWOS  92
      IF(.NOT.PREPRO) GO TO 250                                         TWOS  93
C APPLY REFRACTION CORRECTION TO OBSERVATION                            TWOS  94
      PRETYP=IPRE2                                                      TWOS  95
      OBSAVE=OBS                                                        TWOS  96
      OBS=0.000                                                         TWOS  97
      REFRAC=FREQ                                                       TWOS  98
      FREQ=RFNDX(2)                                                     TWOS  99
      ISAVE=MTYPE                                                       TWOS 100
      MTYPE=KTYPE+1                                                     TWOS 101
      CALL ORBIT(DAY2)                                                  TWOS 102
      THETG=GRHRAN(DSTA2,ISTA2)                                         TWOS 103
      CALL PROCES(ISTA2,DSTA2,THETG)                                    TWOS 104
      OBS=-OBS                                                          TWOS 105
      FREQ=RFNDX(1)                                                     TWOS 106
      THETG=GRHRAN(DSTA1,ISTA)                                          TWOS 107
      PREPRO=SAVEPP                                                     TWOS 108
      CALL PROCES(ISTA,DSTA1,THETG)                                     TWOS 109
      OBS=OBSAVE+F*OBS/C                                                TWOS 110
      MTYPE=ISAVE                                                       TWOS 111
```

```
  250 FREQ=F                                                        TWOS 112
C OBTAIN ORBIT                                                      TWOS 113
  260 CALL ORBIT(DAY2)                                              TWOS 114
      JTYPE=KTYPE+1                                                 TWOS 115
      F=FREQ                                                        TWOS 116
      IF(KTYPE.EQ.1) F=1.000                                        TWOS 117
C CALCULATE RIGHT ASCENSION OF GREENWICH AND QUANTITIES NECESSARY TO TWOS 118
C COMPUTE PARTIALS FOR SECOND STATION                               TWOS 119
      THETG=GRHRAN(DSTA2,ISTA2)                                     TWOS 120
      T=DATAN(RENV(3,ISAT)/DSQRT(1.0D0-RENV(3,ISAT)**2))/DRAD       TWOS 121
C COMPUTE PARTIALS FOR SECOND STATION                               TWOS 122
      CALL PREDCT(ISTA2,DAY2,RESID,RESID2,DATASW)                   TWOS 123
      PMPX0(NPARM-8,1)=OBSDOT(JTYPE,ISTA2,P)*F/C                    TWOS 124
      OBSC=-RESID                                                   TWOS 125
      DO 270 I=1,NPARM                                              TWOS 126
  270 PMPX0(I,2)=-PMPX0(I,1)                                        TWOS 127
C CALCULATE RIGHT ASCENSION OF GREENWICH AND QUANTITIES NECESSARY TO TWOS 128
C COMPUTE PARTIALS FOR FIRST STATION                                TWOS 129
      THETG=GRHRAN(DSTA1,ISTA)                                      TWOS 130
C COMPUTE PARTIALS FOR FIRST STATION                                TWOS 131
      CALL PREDCT(ISTA,DAY2,RESID,RESID2,DATASW)                    TWOS 132
      PMPX0(NPARM-6,1)=OBSDOT(JTYPE,ISTA,P)*F/C                     TWOS 133
      OBSC=OBSC+RESID                                               TWOS 134
      DO 280 I=1,NP6                                                TWOS 135
  280 PMPX0(I,1)=PMPX0(I,1)+PMPX0(I,2)                              TWOS 136
      DO 290 I=1,3                                                  TWOS 137
  290 PMPX0(NP6+I+3,1)=PMPX0(NP6+I,2)                               TWOS 138
      PMPX0(NPARM-7,1)=1.0D0                                        TWOS 139
      RESID=OBS-OBSC                                                TWOS 140
      RETURN                                                        TWOS 141
C AVERAGE RANGE RATE DATA                                           TWOS 142
  300 RCOR=0.0D0                                                    TWOS 143
      IF(IPRE1.LE.0) GO TO 400                                      TWOS 144
      REFRAC=0.0D0                                                  TWOS 145
      IF(IPRE2.LE.0) GO TO 350                                      TWOS 146
C SET REFRACTION INDICES FOR TRANSMITTER & RECEIVER                 TWOS 147
      REFRAC=FREQ                                                   TWOS 148
      IF(RFNDX(1).LE.0.) RFNDX(1)=328.5                             TWOS 149
      IF(RFNDX(2).LE.0.) RFNDX(2)=328.5                             TWOS 150
      RFNDX(1)=RFNDX(1)*0.84323360-2                                TWOS 151
      RFNDX(2)=RFNDX(2)*0.84323360-2                                TWOS 152
  350 DAYR=DAY1                                                     TWOS 153
      PREPRO=IPRE2.GT.0                                             TWOS 154
      LOGIC=PREPRO                                                  TWOS 155
C COMPUTE TRANSIT TIMES FOR START OF COUNTING INTERVAL              TWOS 156
      CALL UPDOWN(DAY1,DAYR,DAYT,-1.0D0,ISTA,ISTA2,ISAT,LOGIC,RCOR, TWOS 157
     *   RFNDX)                                                     TWOS 158
      DID=DAYR-DAY1                                                 TWOS 159
      DIU=DAY1-DAYT                                                 TWOS 160
      DAYR=DAY2                                                     TWOS 161
      DAY2=DAYR-DID                                                 TWOS 162
      RCOR=-RCOR                                                    TWOS 163
C COMPUTE TRANSIT TIMES FOR END OF COUNTING INTERVAL                TWOS 164
      CALL UPDOWN(DAY2,DAYR,DAYT,-1.0D0,ISTA,ISTA2,ISAT,LOGIC,RCOR, TWOS 165
     *   RFNDX)                                                     TWOS 166
      D2D=DAYR-DAY2                                                 TWOS 167
```

```
      D2U=DAY2-DAYT                                                    TWOS 168
  400 DAYR=DAY1+D1D                                                    TWOS 169
      DAYT=DAY1-D1U                                                    TWOS 170
      CALL ORBIT(DAY1)                                                 TWOS 171
      THETG=GRHRAN(DAYT,ISTA2)                                         TWOS 172
      T=DATAN(RENV(3,ISAT)/DSQRT(1.0D0-RENV(3,ISAT)**2))/DRAD          TWOS 173
      OBSC=-R(ISAT)                                                    TWOS 174
C COMPUTE UPLINK RANGE & PARTIALS AT START                             TWOS 175
      CALL PREDCT(ISTA2,DAY1,RESID,RESID2,DATASW)                      TWOS 176
      PMPX0(NPARM-8,1)=OBSDOT(2,ISTA2,P)                               TWOS 177
      DO 410 I=1,NP6                                                   TWOS 178
  410 PMPX0(I,2)=-PMPX0(I,1)                                           TWOS 179
      DO 420 I=1,3                                                     TWOS 180
      PMPX0(I+NP6,2)=0.0D0                                             TWOS 181
  420 PMPX0(I+NP3,2)=-PMPX0(I+NP6,1)                                   TWOS 182
      THETG=GRHRAN(DAYR,ISTA)                                          TWOS 183
      OBSC=OBSC-R(ISAT)                                                TWOS 184
C COMPUTE DOWNLINK RANGE & PARTIALS AT START                           TWOS 185
      CALL PREDCT(ISTA,DAY1,RESID,RESID2,DATASW)                       TWOS 186
      PMPX0(NPARM-6,1)=OBSDOT(2,ISTA,P)                                TWOS 187
      DO 430 I=1,NP3                                                   TWOS 188
  430 PMPX0(I,2)=PMPX0(I,2)-PMPX0(I,1)                                 TWOS 189
      CALL ORBIT(DAY2)                                                 TWOS 190
      DAYT=DAY2-D2U                                                    TWOS 191
      DAYR=DAY2+D2D                                                    TWOS 192
      THETG=GRHRAN(DAYT,ISTA2)                                         TWOS 193
      OBSC=OBSC+R(ISAT)                                                TWOS 194
C COMPUTE UPLINK RANGE & PARTIALS AT END                               TWOS 195
      CALL PREDCT(ISTA2,DAY2,RESID,RESID2,DATASW)                      TWOS 196
      PMPX0(NPARM-8,1)=OBSDOT(2,ISTA2,P)                               TWOS 197
      DO 440 I=1,NP6                                                   TWOS 198
  440 PMPX0(I,2)=PMPX0(I,2)+PMPX0(I,1)                                 TWOS 199
      DO 450 I=1,3                                                     TWOS 200
  450 PMPX0(I+NP3,2)=PMPX0(I+NP3,2)+PMPX0(I+NP6,1)                     TWOS 201
      THETG=GRHRAN(DAYR,ISTA)                                          TWOS 202
      OBSC=OBSC+R(ISAT)                                                TWOS 203
C COMPUTE DOWNLINK RANGE & PARTIALS AT END                             TWOS 204
      CALL PREDCT(ISTA,DAY2,RESID,RESID2,DATASW)                       TWOS 205
      PMPX0(NPARM-6,1)=OBSDOT(2,ISTA,P)                                TWOS 206
      DO 460 I=1,NP3                                                   TWOS 207
  460 PMPX0(I,2)=PMPX0(I,2)+PMPX0(I,1)                                 TWOS 208
      DELTAT=(DAY2+D2D-DAY1-D1D)*17.2804                               TWOS 209
      DTINV=1.0D0/DELTAT                                               TWOS 210
      IF(PREPRO) OBS=OBS+RCOR*DTINV                                    TWOS 211
      PREPRO=.FALSE.                                                   TWOS 212
C CALCULATE RESIDUAL                                                   TWOS 213
      RESID=OBS-OBSC*DTINV                                             TWOS 214
      DO 470 I=1,NPARM                                                 TWOS 215
  470 PMPX0(I,1)=PMPX0(I,2)*DTINV                                      TWOS 216
      PMPX0(NPARM-7,1)=1.0D0                                           TWOS 217
      IF(MTYPE.EQ.30) RETURN                                           TWOS 218
      PMPX0(NP6+1,1)=PMPX0(NP6+1,1)+PMPX0(NP3+1,1)                     TWOS 219
      PMPX0(NP6+2,1)=PMPX0(NP6+2,1)+PMPX0(NP3+2,1)                     TWOS 220
      PMPX0(NP6+3,1)=PMPX0(NP6+3,1)+PMPX0(NP3+3,1)                     TWOS 221
      PMPX0(NPARM-6,1)=PMPX0(NPARM-6,1)+PMPX0(NPARM-8,1)               TWOS 222
      PMPX0(NPARM-8,1)=0.0D0                                           TWOS 223
      PMPX0(NP3+1,1)=0.0D0                                             TWOS 224
      PMPX0(NP3+2,1)=0.0D0                                             TWOS 225
      PMPX0(NP3+3,1)=0.0D0                                             TWOS 226
      RETURN                                                           TWOS 227
      END                                                              TWOS 228
```


ENTRY TWOSTI
INITIALIZE
RETURN
ENTRY TWOSTA
ONLY EPHEMERIS NEEDED
NO
100
YES
PREXT
OBTAIN EPHEMERIS
RETURN
100
SET SWITCHES
UPDCOM
COMPUTE ALL TRANSIT TIMES
APPLY REFRACTION CORRECTION TO OBSERVATIONS
260
ORBIT
OBTAIN ORBIT
GRHRAN
CALCULATE RIGHT ASCENSION OF GREENWICH AND QUANTITIES NECESSARY TO MARTINS FOR 2nd STATION
A


A
PREDCT
COMPUTE PARTIALS FOR SECOND STATION
GHRAN
CALCULATE RIGHT ASCENSION OF GREENWICH AND QUANTITIES NECESSARY TO PARTIALS FOR FIRST STATION
PREDCT
COMPUTE PARTIALS FOR FIRST STATION
SET REFRACTION INDICES FOR TRASMITTER AND RECEIVER
UPDOWN
COMPUTE TRANSIT TIMES FOR START OF COUNTING INTERVAL
UPDOWN
COMPUTE TRANSIT TIMES FOR END OF COUNTING INTERVAL
B
B
PREXT
COMPUTE UP-LINK RANGE AND PARTIALS AT START
PREDCT
COMPUTE DOWNLINK RANGE AND PARTIALS AT START
PREXT
COMPUTE UPLINK RANGE AND PARTIALS AT END
PREDCT
COMPUTE DOWNLINK RANGE AND PARTIALS AT END
CALCULATE RESIDUAL
RETURN

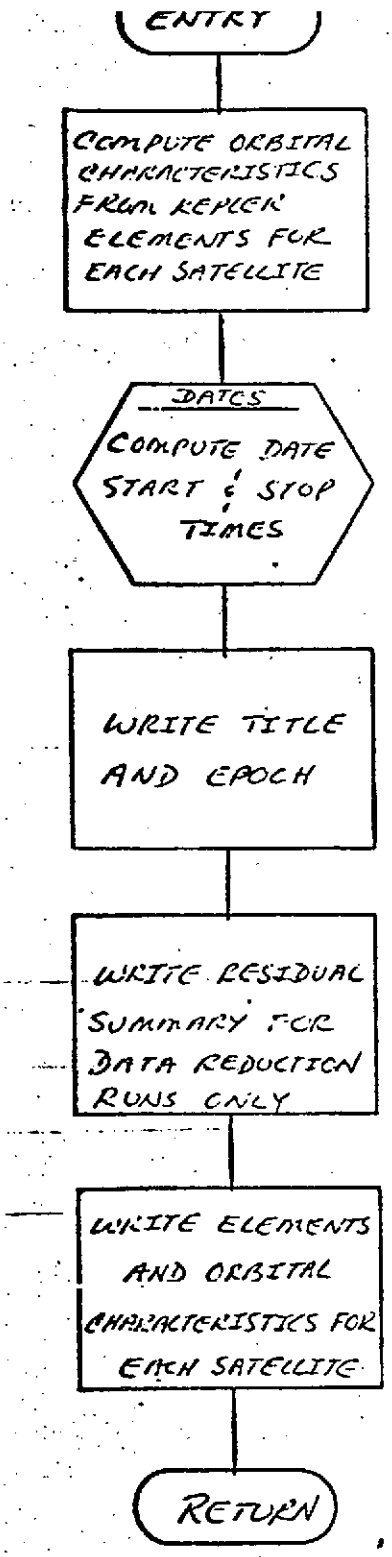
NAME	TYPORB	
PURPOSE	TO PRINT ARC SUMMARY PAGE	
CALLING SEQUENCE	CALL TYPORB(DAY0,MINDEX,ARCNO,OUTER,RMSPOS)	
SYMBOL	TYPE	DESCRIPTION
DAY0	DP	INPUT - TIME OF FINAL SET OF PARTIAL DERIVATIVES FOR DRAG
MINDEX	I	INPUT - TOTAL NUMBER OF MEASUREMENT PLUS 1
ARCNO	I	INPUT - ARC NUMBER
OUTER	I	INPUT - OUTER ITERATION
RMSPOS (2,1)	DP	INPUT - ARRAY CONTAINING RMS OF POSITION AND VELOCITY FOR ADJUSTED ELEMENTS
SUBROUTINES USED	DOTPRD, ELEM, DATES, APPER	
COMMON BLOCKS	APARAM, ALPHRC, CELEM, CGEOS, CORBI, CSTINF, CTIME, GEODYN, INITBK, INTBLK, PRIORI, TPEBLK, XYZOUT, CONSTS	
INPUT FILES	NONE	
OUTPUT FILES	NONE	

"GEODYN SYSTEMS DESCRIPTION"
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE TYPORB(DAY0,MINDEX,ARCNO,OUTER,RMSPOS)                 TYPO  37
      IMPLICIT REAL*8 (A-H,O-Z)                                         TYPO  38
      LOGICAL DRAGSW,ORBTSW,HYPER                                       TYPO  39
      INTEGER ADDR,OUTP,DATP,ARCNO,OUTER                                TYPO  40
      REAL RMSTOT,TYPRMS,BSUM                                           TYPO  41
      DOUBLE PRECISION LHAT,MSAT                                        TYPO  42
      DIMENSION LHAT(3),DL(2),DT(2),RMSPOS(2,1)                         TYPO  43
      COMMON/APARAM/INPAR(4),NSAT,NGPARC(5)                             TYPO  44
      COMMON/ALPHRC/ITNMS(14),ATYPE(47),HYPER                           TYPO  45
      COMMON/CELEM/ELEMST(6,2),ORBELA(6,2),EC,XNU,RMSTOT                TYPO  46
      COMMON/CGEOS/ISATID(2),IPREPR(453)                                TYPO  47
      COMMON/CONSTS/PI,TWOPI,DRAD,PSEC                                  TYPO  48
      COMMON/CORBI/RANODT(2),PERDOT(2),PEPHT(2),APHT(2),PRD(2)          TYPO  49
      COMMON/CSTINF/MEASNO(35),TYPRMS(30),NOTYPE(2,30),BSUM(156),       TYPO  50
     *   NOWTOT,LPASE                                                   TYPO  51
      COMMON/CTIME/DATAEP(2),DSTART,DAYSTP,DAYINT(7),IYREG              TYPO  52
      COMMON/GEODYN/DATE(5)                                             TYPO  53
      COMMON/INITBK/IEPYMD,IEPHM,EPSEC,IDUMI(32),ORBTSW,MISLOG(20)      TYPO  54
      COMMON/INTBLK/THDOT(3),GM,AE,AESQ(47),ASAT(2),MSAT(2),VARSTP(3),  TYPO  55
```

```
     *    ADDR(14)                                                          TYPO  56
          COMMON/PRIORI/ELEMIN(48),TITLE(30),DRAG(12),CD(2),CDD(2),EMISS(2) TYPO  57
          COMMON/TPERLK/INTP,OUTP,DATP(10)                                  TYPO  58
          COMMON/XYZOUT/ELEMS(6,2),DRGPAR(6,2)                              TYPO  59
      C COMPUTE ORBITAL CHARACTERISTICS FROM KEPLER ELEMENTS FOR EACH        TYPO  60
      C SATELLITE                                                           TYPO  61
          DO 100 I=1,NSAT                                                   TYPO  62
          DRAGSW=ADDR(I).GT.0                                               TYPO  63
          CALL ELEM(ELEMST(1,I),ORBELA(1,I),1.,.FALSE.,ORBELA)              TYPO  64
          IF(.NOT.DRAGSW) GO TO 75                                          TYPO  65
          RV=DOTPRD(ELEMS(1,I),ELEMS(4,I))                                  TYPO  66
          VSQ=DOTPRD(ELEMS(4,I),ELEMS(4,I))                                 TYPO  67
          RSQ=DOTPRD(ELEMS(1,I),ELEMS(1,I))                                 TYPO  68
          D=DSQRT(VSQ-RV**2/RSQ)                                            TYPO  69
          DO 50 J=1,3                                                       TYPO  70
       50 LHAT(J)=(-RV*ELEMS(J,I)/RSQ+ELEMS(J+3,I))/D                       TYPO  71
          DL(I)=DOTPRD(LHAT,DRGPAR(1,I))*CD(I)/(DAYC-DSTART)**2             TYPO  72
       75 AE35=(AE/DABS(ORBELA(1,I)))**3.5/(1.000-ORBELA(2,1)**2)**2        TYPO  73
          COSI=DCOS(ORBELA(3,I)*ORAD)                                       TYPO  74
          RANDOT(I)=-9.970D0*AE35*COSI                                      TYPO  75
          PEROOT(I)=4.98D0*AE35*(5.0D0*COSI**2-1.0D0)                       TYPO  76
      100 IF (DRAGSW) DT(I)=6.0D0*PI*DSQRT(DABS(ORBELA(1,I))**3/GM)*        TYPO  77
     *       (DOTPRD(ELEMS(1,I),DRGPAR(1,I))/(RSQ*DSQRT(RSQ))+              TYPO  78
     *       DOTPRD(ELEMS(4,I),DRGPAR(4,I))/GM)*CD(I)/(DAYC-DSTART)         TYPO  79
          NOBS=MINDEX-1                                                     TYPO  80
      C COMPUTE DATA START & STOP TIMES                                     TYPO  81
          CALL DATES(DATAEP(1),IYMD1,IHM1,SEC1)                             TYPO  82
          CALL DATES(DAYSTP,IYMD,IHM,SEC)                                   TYPO  83
          DO 150 K=1,2                                                      TYPO  84
      C WRITE TITLE & EPOCH                                                 TYPO  85
          WRITE(OUTP,1000) DATE,ARCNO,OUTER,TITLE                           TYPO  86
          WRITE(OUTP,4000) IEPYMD,IEPHM,EPSEC,IYMD1,IHM1,SEC1,IYMD,IHM,SEC  TYPO  87
          IF(ORBTSW) GO TO 250                                              TYPO  88
          IF(NOBS.LE.0) GO TO 250                                           TYPO  89
      C WRITE RESIDUAL SUMMARY FOR DATA REDUCTION RUN                       TYPO  90
          WRITE(OUTP,5000)                                                  TYPO  91
          DO 200 I=1,30                                                     TYPO  92
      200 IF(NOTYPE(2,I).GT.0) WRITE(OUTP,6000) ATYPE(I),NOTYPE(2,I),       TYPO  93
     *       TYPRMS(I)                                                      TYPO  94
          WRITE(OUTP,7000) NOWTCB,RMSTOT,NOBS                               TYPO  95
      250 CALL APPER                                                        TYPO  96
      C WRITE ELEMENTS & ORBITAL CHARACTERISTICS FOR EACH SATELLITE         TYPO  97
          DO 150 I=1,NSAT                                                   TYPO  98
          WRITE(OUTP,3000) I,(ELEMST(J,I),J=1,6),(ORBELA(J,I),J=1,6),       TYPO  99
     *       ISATID(I),ASAT(I),MSAT(I),CD(I),CDD(I),EMISS(I)                TYPO 100
          IF(.NOT.ORBTSW) WRITE(OUTP,9000) (RMSPOS(J,I),J=1,2)              TYPO 101
          DRAGSW=ADDR(I).GT.0                                               TYPO 102
          PERMIL=PERHT(I)/1.6093500                                         TYPO 103
          APMIL=APHT(I)/1.6093500                                           TYPO 104
          PRD(I)=TWOPI*DSQRT(DABS(ORBELA(1,I))**3/GM)                       TYPO 105
          PRDMIN=PRD(I)/60.0D0                                              TYPO 106
          WRITE(OUTP,2000) RANDOT(I),PEROOT(I),PERHT(I),PERMIL,APHT(I),     TYPO 107
     *       APMIL,PRDMIN                                                   TYPO 108
      150 IF(DRAGSW) WRITE(OUTP,3000) DL(I),DT(I)                           TYPO 109
          RETURN                                                            TYPO 110
     1000 FORMAT(1H1,14X,5A8/'0ARC',I4,                                     TYPO 111
```

```
     *  SUMMARY FOR LAST INNER ITERATION OF OUTER ',                    TYPO 112
     *  'ITERATION',I2/3(/1X,10A8))                                     TYPO 113
2000 FORMAT(1HC,    9HNODE RATE,3X,12HARG PER RATE,6X,14HPERIGEE HEIGHTTYPO 114
     *  ,12X,13HAPOGEE HEIGHT,10X,6HPERIOD/                            TYPO 115
     *  1X,2(9H(DEG/DAY),4X),4X,4H(KM),                                TYPO 116
     *     6X,7H(MILES),8X,4H(KM),7X,7H(MILES),4X,9H(MINUTES)/         TYPO 117
     *  1X,F9.5,F12.5,3X,2F12.4,2F13.4,F12.3)                          TYPO 118
3000 FORMAT(1HC,34X,4HDRAG,7X,13HPERIOD DEC/34X,10H(M/DAY**2),4X,      TYPO 119
     *  9H(SEC/DAY)/28X,2F14.4)                                        TYPO 120
4000 FORMAT('0EPOCH',I7,I5,F8.4,' BEGIN TIME',I7,I5,F8.4,              TYPO 121
     *  ' END',I7,I5,F8.4)                                             TYPO 122
5000 FORMAT('0MEAS. TYPE',5X,'NO. OF WTD RESIDUALS',5X,'WTD RMS')      TYPO 123
6000 FORMAT(3X,A6,7X,I12,8X,F12.3)                                     TYPO 124
7000 FORMAT('0  TOTAL',8X,I12,8X,F12.3,'  TOTAL NO. OF OBS =',I8)      TYPO 125
8000 FORMAT('0SAT',I2,' CURRENT ELEMENTS.  UNITS: LENGTH-METERS, ',    TYPO 126
     *  'TIME-SECONDS, ANGLES-DEGREES.'/'  X  =',G22.16,'  Y  =',G22.16TYPO 127
     *  ,'  Z  =',G22.16/' XDOT=',G22.16,' YDOT=',G22.16,' ZDOT=',    TYPO 128
     *  G22.16/'  A  =',G22.16,'  E  =',G22.16,' INCL=',G22.16/        TYPO 129
     *  ' NODE=',G22.16,' PERG=',G22.16,' MEAN=',G22.16/' SATID=',I8,  TYPO 130
     *  '  AREA (M**2) =',1PD12.4,'  MASS (KG) =',3PD12.2/' DRAG=',    TYPO 131
     *  1PD12.4,'  DRAG DOT=',D12.4,'  SOLRAD=',D12.4)                 TYPO 132
9000 FORMAT(' RMS POSITION =',F12.3,'  RMS VELOCITY =',F12.6)          TYPO 133
     END                                                               TYPO 134
```


ENTRY
COMPUTE ORBITAL CHARACTERISTICS FROM KEPLER ELEMENTS FOR EACH SATELLITE
DATCS
COMPUTE DATE START & STOP TIMES
WRITE TITLE AND EPOCH
WRITE RESIDUAL SUMMARY FOR DATA REDUCTION RUNS ONLY
WRITE ELEMENTS AND ORBITAL CHARACTERISTICS FOR EACH SATELLITE
RETURN

UPDATE

DESCRIPTION

This subroutine uses the matrix partitioning techniques described in the GEODYN Systems Description, Volume I, to update the estimated values of arc adjusted parameters for corrections resulting from the adjustment of common parameters in the partitioned solution.

NAME — UPDATE

PURPOSE — TO UPDATE ARC ADJUSTED PARAMETERS TO COMPENSATE FOR ADJUSTMENTS TO COMMON PARAMETERS

CALLING SEQUENCE — CALL UPDATE(NARCS,OUTER,NSTART,SUM1,DELTA,DDELTA, TTL,BSNOS,BSVAL)

SYMBOL	TYPE	DESCRIPTION
NARCS	I	INPUT - NUMBER OF ARCS
OUTER	I	INPUT - OUTER ITERATION NUMBER
NSTART	I	INPUT - STARTING LOCATION IN NORMAL MATRIX OF COMMON PARAMETERS
SUM1 (1)	DP	INPUT - NORMAL MATRIX
DELTA (1)	DP	INPUT & OUTPUT - CORRECTION VECTORS FOR ADJUSTED PARAMETERS
DDELTA (1)	DP	SCRATCH
TTL (1)	DP	INPUT - PARAMETER TITLE ARRAY
BSNOS (1)	DP	INPUT - LOCATIONS IN NORMAL MATRIX OF BIASES, DRAG, SOLRAD, AND GEOPOTENTIAL PARAMETERS
BSVAL (1)	DP	INPUT & OUTPUT - VALUES OF BIASES, DRAG, SOLRAD, AND GEOPOTENTIAL PARAMETERS

SUBROUTINES USED — CORREL, DATARD

COMMON BLOCKS — APARAM, CELEM, CONSTS, CPARAM, TPEBLK, PRIORI

INPUT FILES — NONE

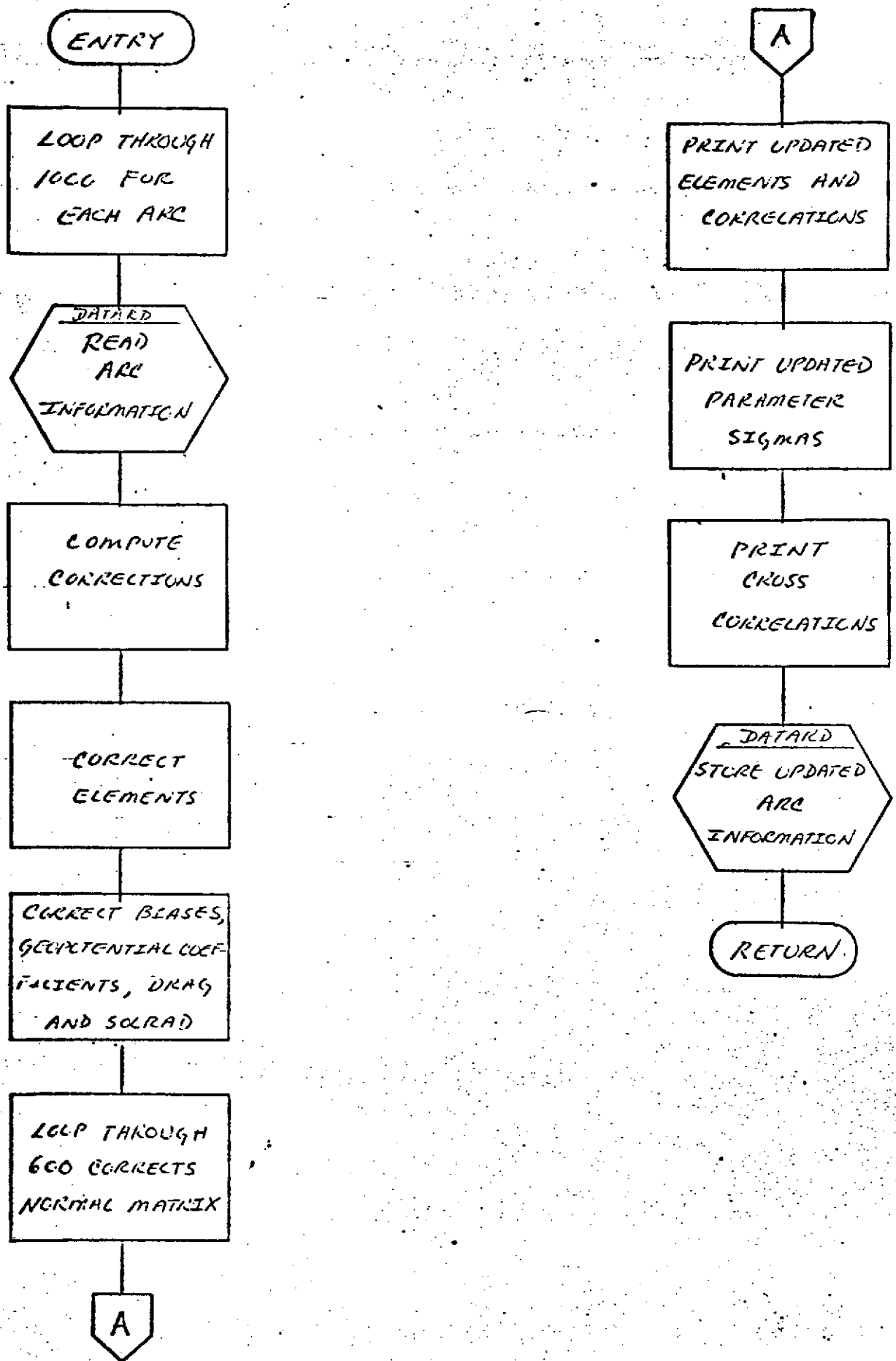
OUTPUT FILES — OUTP - PRINTER

GEODYN SYSTEMS DESCRIPTION
VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE UPDATE(NARCS,OUTER,NSTART,SUM1,DELTA,                  UPDA  51
     *   DDELTA,TTL,BSNOS,BSVAL)                                        UPDA  52
      IMPLICIT REAL*8 (A-H,O-Z)                                         UPDA  53
      LOGICAL CMPGER                                                    UPDA  54
      INTEGER ARCNO,OUTER,OUTP                                          UPDA  55
```

```
      INTEGER*2 BSNOS                                                   UPDA  56
      REAL BSVAL                                                        UPDA  57
      DIMENSION SUM1(1),DELTA(1),DDELTA(1),TTL(1),BSNOS(1),BSVAL(1)     UPDA  58
      COMMON/APARAM/INPAR,INPARI,NBIAS,NSTSTA,NSAT,NGPARC,NORECI,NPARAM,UPDA  59
     *   NEBIAS,MAXPAR                                                  UPDA  60
      COMMON/CELEM/ELEMST(12),ORBELA(12),IDUM(5)                        UPDA  61
      COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC                               UPDA  62
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,NGPCOM,    UPDA  63
     *   NCSEST,CMPGPR,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEN,INNRSW,      UPDA  64
     *   NCONST,NDCONS                                                  UPDA  65
      COMMON/PRIORI/ELEMIN(9),CO(2,3)                                   UPDA  66
      COMMON/TAPEBLK/INTP,OUTP,ITAPES(10)                               UPDA  67
      INDXNO(I)=NDIM*(I-1)-(I*(I-1))/2                                  UPDA  68
C LOOP THROUGH 1000 FOR EACH ARC                                        UPDA  69
      DO 1000 ARCNO=1,NARCS                                             UPDA  70
C READ ARC INFORMATION                                                  UPDA  71
      CALL DATARD(ARCNO,.FALSE.,.FALSE.,.TRUE.)                         UPDA  72
      I1=0                                                              UPDA  73
C COMPUTE CORRECTIONS                                                   UPDA  74
      DO 100 I=1,NPARAM                                                 UPDA  75
      DELTA(I)=0.000                                                    UPDA  76
      DO 50 L=NSTART,NDIM                                               UPDA  77
      L1=I1+L                                                           UPDA  78
   50 DELTA(I)=DELTA(I)-SUM1(L1)*DELTA(L)                               UPDA  79
  100 I1=I1+NDIM-I                                                      UPDA  80
      I1=6*NSAT                                                         UPDA  81
C CORRECT ELEMENTS                                                      UPDA  82
      DO 150 I=1,I1                                                     UPDA  83
  150 ELEMST(I)=ELEMST(I)+DELTA(I)                                      UPDA  84
      NBSGP=MBIAS+NGPARC                                                UPDA  85
C CORRECT BIASES , GEOPOTENTIAL COEFFICIENTS, DRAG & SOLRAD             UPDA  86
      DO 200 I=1,NBSGP                                                  UPDA  87
      IF(BSNOS(I).EQ.0) GO TO 200                                       UPDA  88
      I1=I1+1                                                           UPDA  89
      BSVAL(I)=BSVAL(I)+DELTA(I1)                                       UPDA  90
  200 CONTINUE                                                          UPDA  91
      I1=0                                                              UPDA  92
      DO 250 L=1,NSAT                                                   UPDA  93
      DO 250 I=1,3                                                      UPDA  94
      I1=I1+1                                                           UPDA  95
      IF(BSNOS(I1).LE.0) GO TO 250                                      UPDA  96
      CO(L,I)=BSVAL(I1)                                                 UPDA  97
  250 CONTINUE                                                          UPDA  98
      IST=INDXNO(NSTART)                                                UPDA  99
      L1=0                                                              UPDA 100
C LOOP THROUGH 600 CORRECTS NORMAL MMATRIX                              UPDA 101
      DO 600 L=1,NPARAM                                                 UPDA 102
      MST=IST                                                           UPDA 103
      DO 400 M=NSTART,NDIM                                              UPDA 104
      DDELTA(M)=0.000                                                   UPDA 105
      NM=IST+M                                                          UPDA 106
      DO 300 N=NSTART,M                                                 UPDA 107
      NL=L1+N                                                           UPDA 108
      DDELTA(M)=DDELTA(M)+SUM1(NM)*SUM1(NL)                             UPDA 109
  300 NM=NM+NDIM-N                                                      UPDA 110
      IF(M.EQ.NDIM) GO TO 400                                           UPDA 111
```

```
      MP1=M+1                                                           UPDA 112
      DO 350 N=MP1,NDIM                                                 UPDA 113
      NM=MST+N                                                          UPDA 114
      NL=L1+N                                                           UPDA 115
  350 DDELTA(M)=DDELTA(M)+SUM1(NM)*SUM1(NL)                             UPDA 116
  400 MST=MST+NDIM-M                                                    UPDA 117
      I1=L1                                                             UPDA 118
      DO 500 I=L,NPARAM                                                 UPDA 119
      I2=L1+I                                                           UPDA 120
      DO 450 M=NSTART,NDIM                                              UPDA 121
      IM=I1+M                                                           UPDA 122
  450 SUM1(I2)=SUM1(I2)+SUM1(IM)*DDELTA(M)                              UPDA 123
  500 I1=I1+NDIM-I                                                      UPDA 124
      DO 550 M=NSTART,NDIM                                              UPDA 125
      IM=L1+M                                                           UPDA 126
  550 SUM1(IM)=-DDELTA(M)                                               UPDA 127
  600 L1=L1+NDIM-L                                                      UPDA 128
C PRINT UPDATED ELEMENTS & CORRELATIONS                                 UPDA 129
      WRITE(OUTP,44446)                                                 UPDA 130
      J1=NSAT*6                                                         UPDA 131
      WRITE(OUTP,10304) ARCNO,OUTER,(ELEMST(J),J=1,J1)                  UPDA 132
      CALL CORREL(SUM1,NPARAM,NDIM,ARCNO,TTL)                           UPDA 133
      WRITE(OUTP,44460) ARCNO                                           UPDA 134
C PRINT UPDATED PARAMETER SIGMAS                                        UPDA 135
      DO 650 I=1,NPARAM                                                 UPDA 136
      II=INDXNO(I)+I                                                    UPDA 137
  650 WRITE(OUTP,44461) TTL(I),SUM1(II)                                 UPDA 138
C PRINT CROSS CORRELATIONS                                              UPDA 139
      WRITE(OUTP,44450) ARCNO                                           UPDA 140
      WRITE(OUTP,10214) (TTL(J),J=NSTART,NDIM)                          UPDA 141
      IST=NSTART                                                        UPDA 142
      ISTP=NDIM                                                         UPDA 143
      DO 800 I=1,NPARAM                                                 UPDA 144
      INDEXI=INDXNO(I)                                                  UPDA 145
      II=INDEXI+I                                                       UPDA 146
      DO 700 J=NSTART,NDIM                                              UPDA 147
      JJ=INDXNO(J)+J                                                    UPDA 148
      IJ=INDEXI+J                                                       UPDA 149
  700 SUM1(IJ)=SUM1(IJ)/(SUM1(II)*DSQRT(SUM1(JJ)))                      UPDA 150
      WRITE(OUTP,10215) TTL(I),(SUM1(J),J=IST,ISTP)                     UPDA 151
      IST=IST+NDIM-I                                                    UPDA 152
  800 ISTP=ISTP+NDIM-I                                                  UPDA 153
C STORE UPDATED ARC INFORMATION                                         UPDA 154
 1000 CALL DATARC(ARCNO,.FALSE.,.TRUE.,.FALSE.)                         UPDA 155
      RETURN                                                            UPDA 156
10214 FORMAT(1HC,6X,18A6)                                               UPDA 157
10215 FORMAT(1HC,A6,18F6.3/(7X,18F6.3))                                 UPDA 158
10304 FORMAT(1HC//,40X,24HUPDATED ELEMENTS FOR ARC,I3,22H AFTER OUTER ITUPDA 159
     1ERATION,I2//                                                      UPDA 160
     *    (44X,1HX,25X,1HY,25X,1HZ//30X,3D24.16//42X,4HXDOT,22X,       UPDA 161
     *    4HYDOT,22X,4HZDOT//30X,3D24.16//)                            UPDA 162
44446 FORMAT(1H1)                                                       UPDA 163
44450 FORMAT(1HC//7X,'CROSS CORRELATION COEFFICIENTS BETWEEN STATION ',  UPDA 164
     1       'POSITIONS AND ARC',I3,' ADJUSTED PARAMETERS')             UPDA 165
44460 FORMAT(1HC,10X,'ADJUSTED PARAMETER SIGMAS FOR ARC',I3/)           UPDA 166
44461 FORMAT(13X,A6,G10.3)                                              UPDA 167
      END                                                               UPDA 168
```


ENTRY
LOOP THROUGH 1000 FOR EACH ARC
DATARD
READ ARC INFORMATION
COMPUTE CORRECTIONS
CORRECT ELEMENTS
CORRECT BIASES, GEOPOTENTIAL COEFFICIENTS, DRAG AND SOLRAD
LOOP THROUGH 600 CORRECTS NORMAL MATRIX
A
A
PRINT UPDATED ELEMENTS AND CORRELATIONS
PRINT UPDATED PARAMETER SIGMAS
PRINT CROSS CORRELATIONS
DATARD
STORE UPDATED ARC INFORMATION
RETURN

NAME — UPDOWN

PURPOSE — TO COMPUTE UPLINK AND DOWNLINK TRANSIT TIME FOR AVERAGE RANGE RATE DATA OR THE TWO DOWNLINK TRANSIT TIMES FOR VLBI DATA

CALLING SEQUENCE — CALL UPDOWN(DAY,DSTA1,DSTA2,SIGN,IS,A1,ISTA2,ISAT, PREPRO,OBS,RFINDX)

SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT & OUTPUT - SATELLITE TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR INPUT - NOMINAL ESTIMATE
		OUTPUT - CORRECTED SATELLITE TIME
DSTA1	DP	INPUT - FIRST GROUND TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
DSTA2	DP	OUTPUT - SECOND GROUND TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
SIGN	DP	INPUT - =+1 FOR VLBI DATA =-1 FOR AVERAGING RANGE RATE DATA
ISTA1	I	INPUT - INDEX FOR FIRST STATION
ISTA2	I	INPUT - INDEX FOR SECOND STATION
ISAT	I*2	INPUT - SATELLITE INDEX
PREPRO	L	INPUT - SWITCH REQUESTING TROPOSPHERIC REFRACTION PREPROCESSING
OBS	DP	OUTPUT - VALUE OF MEASUREMENT
RFINDX (2)	R	INPUT - REFRACTION INDICES FOR MEASUREMENT PREPROCESSING

SUBROUTINES USED — ORBIT, GRHRAN

COMMON BLOCKS — CUVECT

INPUT FILES — NONE

OUTPUT FILES — NONE

REFERENCES — 'GEODYN SYSTEMS DESCRIPTION' VOLUME 1 - GEODYN DOCUMENTATION

```
      SUBROUTINE UPDOWN(DAY,DSTA1,DSTA2,SIGN,ISTA1,ISTA2,ISAT,PREPRO,    UPDO  54
     .   OBS,RFINDX)                                                     UPDO  55
```

```
      IMPLICIT REAL*8 (A-H,O-Z)                                          UPDO  56
      LOGICAL NOTIST,PREPRO                                              UPDO  57
      INTEGER*2 ISAT                                                     UPDO  58
      REAL RFINDX                                                        UPDO  59
      DIMENSION RFINDX(2)                                                UPDO  60
      COMMON/CUVECT/UHAT(3,2),XYZ(3,2),RXYZ(3,2),RENV(3,2),R(2),         UPDO  61
     .   RSQ(2),XYSQ(2)                                                  UPDO  62
      DATA C/2.99792D8/,DTOL/1.0D-10/,NOTIST/.FALSE./                    UPDO  63
      IF(NOTIST) GO TO 100                                               UPDO  64
      NOTIST=.TRUE.                                                      UPDO  65
      DAYLIT=1.D0/(C*8.64D4)                                             UPDO  66
  100 N=0                                                                UPDO  67
  200 DPREV=DAY                                                          UPDO  68
      CALL ORBIT(DAY)                                                    UPDO  69
      THETG=GRHRAN(DSTA1,ISTA1)                                          UPDO  70
      DAY=DSTA1-R(ISAT)*DAYLIT                                           UPDO  71
      DT=DABS(DAY-DPREV)                                                 UPDO  72
      IF(DT.LE.DTOL) GO TO 300                                           UPDO  73
      N=N+1                                                              UPDO  74
      IF(N.LE.5) GO TO 200                                               UPDO  75
      DT=DT*8.64D4                                                       UPDO  76
      PRINT 1000,DT,DTOL                                                 UPDO  77
  300 IF(PREPRO) OBS=OBS-RFINDX(1)/(0.0260D0+RENV(3,ISAT))               UPDO  78
      DSTA2=SIGN*(DSTA1-DAY)+DAY                                         UPDO  79
      N=0                                                                UPDO  80
  400 DPREV=DSTA2                                                        UPDO  81
      THETG=GRHRAN(DSTA2,ISTA2)                                          UPDO  82
      DSTA2=DAY+SIGN*R(ISAT)*DAYLIT                                      UPDO  83
      DT=DABS(DSTA2-DPREV)                                               UPDO  84
      IF(DT.LT.DTOL) GO TO 500                                           UPDO  85
      N=N+1                                                              UPDO  86
      IF(N.LE.5) GO TO 400                                               UPDO  87
      DT=DT*8.64D4                                                       UPDO  88
      PRINT 1000,DT,DTOL                                                 UPDO  89
  500 IF(PREPRO) OBS=OBS-RFINDX(2)/(0.0260D0+RENV(3,ISAT))               UPDO  90
      RETURN                                                             UPDO  91
 1000 FORMAT(' $**UPDOWN**$ ACCEPTED TRANSIT TIME ERROR AFTER SIX ',     UPDO  92
     .   'ITERATIONS =',E12.5,' SECONDS. GREATER THAN',E12.5,            UPDO  93
     .   ' DAYS $**UPDOWN**$')                                          UPDO  94
      END                                                                UPDO  95
```

NAME		VCONV
PURPOSE		TO CONVERT VARIANCE-COVARIANCE FROM ONE SYSTEM TO ANOTHER
CALLING SEQUENCE		CALL VCONV(VARIN,VAROUT,PARTL)
SYMBOL	TYPE	DESCRIPTION
VARIN	R	INPUT - INPUT VARIANCE-COVARIANCE MATRIX
VAROUT	R	OUTPUT - OUTPUT VARIANCE-COVARIANCE MATRIX
PARTL	R	INPUT - PARTIALS OF -VAROUT- VARIABLES WITH RESPECT TO -VARIN- VARIABLES
SUBROUTINES USED		NONE
COMMON BLOCKS		NONE
INPUT FILES		NONE
OUTPUT FILES		NONE

```
      SUBROUTINE VCONV(VARIN,VAROUT,PARTL)                              VCON  28
      DIMENSION VARIN(3,3),VAROUT(3,3),PARTL(3,3),T(3)                  VCON  29
C PRE-MULTIPLY INPUT MATRIX BY TRANSPOSE OF PARTIAL MATRIX              VCON  30
      DO 10 I=1,3                                                       VCON  31
      DO 10 J=1,3                                                       VCON  32
      VAROUT(I,J)=0.                                                    VCON  33
      DO 10 K=1,3                                                       VCON  34
   10 VAROUT(I,J)=VAROUT(I,J)+PARTL(K,I)*VARIN(K,J)                     VCON  35
C POST-MULTIPLY ABOVE BY PARTIAL MATRIX                                 VCON  36
      DO 30 I=1,3                                                       VCON  37
      DO 20 J=1,3                                                       VCON  38
      T(J)=0.                                                           VCON  39
      DO 20 K=1,3                                                       VCON  40
   20 T(J)=T(J)+VAROUT(I,K)*PARTL(K,J)                                  VCON  41
      DO 30 J=1,3                                                       VCON  42
   30 VAROUT(I,J)=T(J)                                                  VCON  43
      RETURN                                                            VCON  44
      END                                                               VCON  45
```

VEVAL

DESCRIPTION

VEVAL is a major subroutine in GEODYN and is closely linked with the force model subroutines. Its purpose is to evaluate the variational equations.

Various intermediate data is computed in other routines, especially the direct partial derivatives of the accelerations with respect to the force model coefficients being determined. SUNGRV and EGRAV also supply information for computing the partial derivatives of the gradient of the gravitational potential with respect to the position of the satellite at the current time, i.e.

$$\frac{\partial}{\partial \bar{r}} (VU)$$

Subroutine DENSTY supplies the partial derivative of the atmospheric density with respect to spheroid height.

The order of computation is:

- compute $U_{2c} = \frac{\partial}{\partial \bar{r}} (VU)$;
- compute $D_r = \frac{\partial}{\partial \bar{r}} (D)$, where D is the acceleration due to drag;
- evaluate the variational equations.

NAME	VEVAL
ENTRY POINT	PURPOSE
VEVAL1	INITIALIZATION
VEVAL	TO COMPUTE PARTIALS OF ACCELERATION WITH RESPECT TO INSTANTANEOUS ORBITAL ELEMENTS. TO MULTIPLY THESE PARTIALS BY THE VARIATIONAL PARTIALS TO OBTAIN THE VARIATIONAL ACCELERATION FOR INTEGRATION OF THE VARIATIONAL EQUATIONS

CALLING SEQUENCE CALL VEVAL1(GRPAR)

SYMBOL	TYPE	DESCRIPTION
GRPAR (3,1)	DP	INPUT - PARTIALS OF FORCE MODEL PARAMETERS

CALLING SEQUENCE CALL VEVAL(XI,FCT,MDIM,FEVAL,M2)

SYMBOL	TYPE	DESCRIPTION
XI (MDIM,1)	DP	INPUT - ORBITAL ELEMENTS AND VARIATIONAL PARTIALS
FCT (3,1)	DP	INPUT - ACCELERATIONS AND VARIATIONAL PARTIALS
MDIM	I	INPUT - DIMENSION OF XI IN THE CALLING PROGRAM
FEVAL	L	INPUT - FLAG TO DETERMINE WHICH ACCELERATION PARTIALS ARE REQUIRED: TRUE : WITH RESPECT TO EPOCH ELEMENTS AND PARAMETERS FALSE : WITH RESPECT TO INSTANTANEOUS ELEMENTS
M2	I	INPUT - VARIATIONAL EQUATION DISPLACEMENT

SUBROUTINES USED	RESPAR	CLEAR			
COMMON BLOCKS	CPARAM	DRGBLK	FMODEL	INTBLK	MODNGR
	VMAT	VRBLOK	XYZ		
INPUT FILES	NONE				
OUTPUT FILES	NONE				
REFERENCES	*GEODYN SYSTEMS DESCRIPTION* VOLUME I - GEODYN DOCUMENTATION				

```
      SUBROUTINE VEVAL1(GRPAR)                                          VEVA  57
      IMPLICIT REAL*8 (A-H,O-Z)                                         VEVA  58
      LOGICAL FEVAL,CMPGPR                                              VEVA  59
      DOUBLE PRECISION MODEL                                            VEVA  60
      DIMENSION GRPAR(3,1),XI(MDIM,1),FCT(3,1),R1D(3),P1D(3),AL1D(3),   VEVA  61
     *   R2D(3,3),P2D(3,3),AL2D(3,3)                                    VEVA  62
      COMMON/CPARAM/NSTA,NMAST,NSTEST,NDIM,MBIAS,NGPC1,NGPC2,           VEVA  63
     *   NGPCOM,NCSEST,CMPGPR,LIM1,LIM2,NDEN,NDENST,NTIDST,NTIDEV,      VEVA  64
     *   INNRSW,NCONST,NDCONS                                           VEVA  65
      COMMON/DBGBLK/HT,SPSISO,EXPT(3),C0,SBRHO,BBRHOV,VEL,VELR(3)       VEVA  66
      COMMON/FMODEL/INDEX1,INDEX2,INDEX3,INDEX4,CS(30,33),MODEL(8)      VEVA  67
      COMMON/INTBLK/THDOT1(2),THDT2S,GM,AE,AESQ(2),FSQ32,FFSQ32,        VEVA  68
     *   GM3(6),B(2),BDOT(41),NEQNS(15),NEDDY                           VEVA  69
      COMMON/MODNGR/DPXUV(6),RHOM(3,6),RHOSQ(6),RHO3(6)                 VEVA  70
      COMMON/VMAT/VMATRX(3,6),U2D(3,3),C1DER(3,3),C2DER(3,3,3)          VEVA  71
      COMMON/VRBLCK/A1,CSLM(31),SNLM(31),U1D(3),P(33,30),AORN(30),      VEVA  72
     *   TPM(39)                                                        VEVA  73
      COMMON/XYZ/X(6),R1,R2,ISAT,IFORCE(2)                              VEVA  74
      EQUIVALENCE (P1D(1),C1DER(1,1)),(P1D(1),C1DER(1,2)),              VEVA  75
     *   (AL1D(1),C1DER(1,3)),(SP,P(1,1)),(CP,P(2,1)),(R2D(1,1),        VEVA  76
     *   C2DER(1,1,1)),(P2D(1,1),C2DER(1,1,2)),(AL2D(1,1),C2DER(1,1,3)) VEVA  77
      EQUIVALENCE (PDPHOD,EXPT(1))                                      VEVA  78
      RETURN                                                            VEVA  79
      ENTRY VEVAL(XI,FCT,MDIM,FEVAL,M2)                                 VEVA  80
C OBTAIN COEFFICIENT PARTIALS IF GEOPOTENTIAL IS ADJUSTED               VEVA  81
      IF (CMPGPR) CALL RESPAR                                           VEVA  82
      CALL CLEAR(VMATRX,6,3,2)                                          VEVA  83
      RINV=1.0D0/R1                                                     VEVA  84
      R3=R1*R2                                                          VEVA  85
      AL1D(1)=-X(2)/A1                                                  VEVA  86
      AL1D(2)=X(1)/A1                                                   VEVA  87
      AL2D(1,1)=-2.0D0*AL1D(1)*AL1D(2)                                  VEVA  88
      AL2D(2,2)=-AL2D(1,1)                                              VEVA  89
      AL2D(1,2)=AL1D(1)**2-AL1D(2)**2                                   VEVA  90
      U1D(2)=L1D(2)/CP                                                  VEVA  91
      C7=1.0D0/CP**2                                                    VEVA  92
      NEQN=NEQNS(ISAT)                                                  VEVA  93
C COMPUTE PARTIALS OF GEOPOTENTIAL WITH RESPECT TO R, PHI, LAMBDA       VEVA  94
      DO 110 NC=2,INDEX4                                                VEVA  95
      NS=31-NC                                                          VEVA  96
      FN1=NC+1                                                          VEVA  97
      FM=1.D0                                                           VEVA  98
      C0=P(2,NC)                                                        VEVA  99
      MMX=NC+1                                                          VEVA 100
      DO 110 MC=1,MMX                                                   VEVA 101
      MS=34-MC                                                          VEVA 102
      FM=FM+1.D0                                                        VEVA 103
      P1=P(MC,NC)                                                       VEVA 104
      C1=(CS(NC,MC)*CSLM(MC)+CS(NS,MS)*SNLM(MC))*AORN(NC)               VEVA 105
      C2=(-CS(NC,MC)*SNLM(MC)+CS(NS,MS)*CSLM(MC))*AORN(NC)*FM           VEVA 106
      C3=C0                                                             VEVA 107
      C0=P(MC+2,NC)-TPM(MC+1)*P(MC+1,NC)                                VEVA 108
      U2D(1,1)=U2D(1,1)+FN1*C1*(FN1+1.D0)*P1                            VEVA 109
      U2D(2,1)=U2D(2,1)-FN1*C1*C3                                       VEVA 110
      U2D(3,1)=U2D(3,1)-FN1*C2*P1                                       VEVA 111
```

```
      U2D(3,2)=U2D(3,2)+C2*C3                                          VEVA 112
      U2D(3,3)=U2D(3,3)-FM**2*C1*P1                                    VEVA 113
  110 U2D(2,2)=U2D(2,2)+C1*(C0-TPM(MC)*C3-FM*P1*C7)                    VEVA 114
      U2D(1,1)=2.0D0*GM/R3+U2D(1,1)/R2                                 VEVA 115
      U2D(2,1)=U2D(2,1)/(R1*CP)                                        VEVA 116
      U2D(3,1)=U2D(3,1)/R1                                             VEVA 117
      U2D(2,2)=U2D(2,2)*C7+U1D(2)*TPM(2)/CP                            VEVA 118
      U2D(3,2)=U2D(3,2)/CP                                             VEVA 119
      U2D(1,2)=U2D(2,1)                                                VEVA 120
      U2D(1,3)=U2D(3,1)                                                VEVA 121
      U2D(2,3)=U2D(3,2)                                                VEVA 122
      C=X(3)/R2                                                        VEVA 123
      DO 205 I=1,3                                                     VEVA 124
      R1D(I)=X(I)/R1                                                   VEVA 125
  205 P1D(I)=-C*R1D(I)                                                 VEVA 126
      P1D(3)=P1D(3)+RINV                                               VEVA 127
      DO 210 I=1,3                                                     VEVA 128
      P2D(I,I)=-X(3)/R3                                                VEVA 129
      IF(I.EQ.3) P2D(3,3)=P2D(3,3)+P2D(3,3)                            VEVA 130
      C1=3.D0*X(3)*X(I)/R2                                             VEVA 131
      DO 210 J=I,3                                                     VEVA 132
      IF(J.EQ.3) P2D(I,J)=P2D(I,J)-X(I)/R3                             VEVA 133
  210 P2D(I,J)=P2D(I,J)+C1*X(J)/R3                                     VEVA 134
      DO 215 I=1,3                                                     VEVA 135
      DO 214 J=I,3                                                     VEVA 136
  214 R2D(I,J)=-X(I)*X(J)/R3                                           VEVA 137
  215 R2D(I,I)=R2D(I,I)+RINV                                           VEVA 138
C COMPUTE ACCELERATION PARTIAL COMPONENTS FROM GEOPOTENTIAL,MOON & SUN  VEVA 139
      DO 178 I=1,3                                                     VEVA 140
      DO 178 K=1,3                                                     VEVA 141
      C=0.0D0                                                          VEVA 142
      DO 177 J=1,3                                                     VEVA 143
  177 C=C+C1DER(I,J)*U2D(J,K)                                          VEVA 144
      DO 178 L=I,3                                                     VEVA 145
  178 VMATRX(I,L)=VMATRX(I,L)+C*C1DER(L,K)                             VEVA 146
      DO 180 N=1,NSCOY                                                 VEVA 147
      IF(GM3(N).LE.0.0D0) GO TO 180                                    VEVA 148
      C7=GM3(N)/RHO3(N)                                                VEVA 149
      C8=3.0D0*C7/RHOSQ(N)                                             VEVA 150
      DO 179 I=1,3                                                     VEVA 151
      VMATRX(I,I)=VMATRX(I,I)-C7                                       VEVA 152
      DO 179 J=I,3                                                     VEVA 153
  179 VMATRX(I,J)=VMATRX(I,J)+C8*RHOM(I,N)*RHOM(J,N)                   VEVA 154
  180 CONTINUE                                                         VEVA 155
      DO 190 I=1,3                                                     VEVA 156
      DO 190 J=I,3                                                     VEVA 157
      DO 185 K=1,3                                                     VEVA 158
  185 VMATRX(I,J)=VMATRX(I,J)+U1D(K)*C2DER(I,J,K)                      VEVA 159
  190 IF(I.NE.J) VMATRX(J,I)=VMATRX(I,J)                               VEVA 160
      NN=3                                                             VEVA 161
      IF(.NOT.B(ISAT).GT.0.0D0) GO TO 300                              VEVA 162
C COMPUTE ACCELERATION PARTIAL COMPONENT FROM DRAG                     VEVA 163
      NN=6                                                             VEVA 164
      BBRHO=BBRHOV/VEL                                                 VEVA 165
      BRHOTH=BBRHO*THDT2S                                              VEVA 166
      BRHOV=BBRHO/VEL                                                  VEVA 167
```

```
      BRHTDV=BRHDV*THDT2S                                                VEVA 16
      C1=2.D0*FSQ32*SPSISQ-FFSQ32                                        VEVA 16
      C2=2.D0*C1/R1                                                      VEVA 17
      C3=(SPSISQ-1.D0)*C2                                                VEVA 17
      C2=SPSISQ*C2                                                       VEVA 17
      C0=BBRHDV*PDPHOD/R1                                                VEVA 17
      C1=(1.D0+C2)*C0                                                    VEVA 17
      VELR(3)=X(6)                                                       VEVA 17
      EXPT(1)=X(1)*C1                                                    VEVA 17
      EXPT(2)=X(2)*C1                                                    VEVA 17
      EXPT(3)=X(3)*(C3+1.D0)*C0                                          VEVA 17
      C1=BRHTDV*VELR(1)*VELR(2)                                          VEVA 17
      VMATRX(1,2)=VMATRX(1,2)-BRHDTH*(VEL+VELR(1)**2/VEL)                VEVA 18
      VMATRX(2,1)=VMATRX(2,1)+BRHDTH*(VEL+VELR(2)**2/VEL)                VEVA 18
      VMATRX(1,1)=VMATRX(1,1)+C1                                         VEVA 18
      VMATRX(2,2)=VMATRX(2,2)-C1                                         VEVA 18
      VMATRX(3,1)=VMATRX(3,1)+BRHTDV*VELR(3)*VELR(2)                     VEVA 18
      VMATRX(3,2)=VMATRX(3,2)-BRHTDV*VELR(3)*VELR(1)                     VEVA 18
      DO 200 I=1,3                                                       VEVA 18
      DO 200 J=1,3                                                       VEVA 18
      J1=J+3                                                             VEVA 19
      IF(I.EQ.J) VMATRX(I,J1)=VMATRX(I,J1)-BBRHDV                        VEVA 19
      VMATRX(I,J)=VMATRX(I,J)-VELR(I)*EXPT(J)                            VEVA 19
  200 VMATRX(I,J1)=VMATRX(I,J1)-BRHDV*VELR(I)*VELR(J)                    VEVA 19
  300 DO 301 J=1,6                                                       VEVA 19
      K0=(J-1)*M2+1                                                      VEVA 19
      DO 301 I=1,3                                                       VEVA 19
  301 FCT(I,K0)=0.D0                                                     VEVA 19
      IF(NEQN.LT.8) GO TO 304                                            VEVA 19
C COMPUTE ACCELERATION PARTIALS WITH RESPECT TO EPOCH PARAMETERS          VEVA 19
      DO 302 L1=8,NEQN                                                   VEVA 19
      K0=(L1-2)*M2+1                                                     VEVA 19
      K0C=L1-7                                                           VEVA 20
      DO 302 I=1,3                                                       VEVA 20
  302 FCT(I,K0)=GRPAR(I,K0C)                                             VEVA 20
  304 IF(.NOT.PEVAL) RETURN                                              VEVA 20
      DO 305 L1=2,NEQN                                                   VEVA 20
      K0=(L1-2)*M2+1                                                     VEVA 20
      DO 305 I=1,3                                                       VEVA 20
      SUM=0.D0                                                           VEVA 20
      DO 306 J=1,NN                                                      VEVA 20
  306 SUM=SUM+VMATRX(I,J)*XI(J,L1)                                       VEVA 20
  305 FCT(I,K0)=FCT(I,K0)+SUM                                            VEVA 21
      RETURN                                                             VEVA 21
      END                                                                VEVA 21
```

NAME — YMDAY

PURPOSE — TO COMPUTE FOR A GIVEN DATE THE NUMBER OF DAYS FROM JAN 0.0 OF THE REFERENCE YEAR FOR THE ARC

CALLING SEQUENCE — X=YMDAY(IYMD,IHM,SEC)

SYMBOL	TYPE	DESCRIPTION
IYMD	I	INPUT - DATE IN THE FORM OF YYMMDD
IHM	I	INPUT - TIME IN THE FORM OF HHMM
SEC	R	INPUT - SECONDS
YMDAY	DP	OUTPUT - NUMBER OF DAYS FROM JAN 0.0 OF THE PEFERENCE YEAR FOR THE ARC

SUBROUTINES USED — DIFF

COMMON BLOCKS — CTIME

INPUT FILES — NONE

OUTPUT FILES — NONE

```
      DOUBLE PRECISION FUNCTION YMDAY(IYMD,IHM,SEC)                    YMDA  30
      IMPLICIT REAL*8 (A-H,O-Z)                                        YMDA  31
      COMMON/CTIME/DAYREF(11),IYREG                                    YMDA  32
      IY=IYBEG*10000+101                                               YMDA  33
      IHMS=IHM*100                                                     YMDA  34
      CALL DIFF(IY,0,IYMD,IHMS,ID,IS)                                  YMDA  35
      YMDAY=86400*(ID+1)+IS                                            YMDA  36
      YMDAY=(YMDAY+SEC)/8.64D4                                         YMDA  37
      RETURN                                                           YMDA  38
      END                                                              YMDA  39
```

SECTION 9.0
COMMON BLOCK DESCRIPTIONS

The GEODYN program contains 39 common blocks. Each common block is fully described on the following pages. Some common blocks have more than one version. Each version is described.

/ALPMRC/

```
COMMON/ALPMRC/ITNMS(5),TIMING,BLANK,ATYPE(31),
   UNITS(15),ELCUT,HYPER
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
ITNMS (5)	R*8	000000	Alphanumeric information for printout.	BLOCK DATA	INOUPT
TIMING	R*8	000028	Alphanumeric information for printout.	BLOCK DATA	NONAME INOUPT
BLANK	R*8	000030	Alphanumeric information for printout.	BLOCK DATA	NONAME INOUPT SUMMRY
ATYPE (31)	R*8	000038	Alphanumeric information for printout.	BLOCK DATA	NONAME INOUPT SUMMRY TYPORB
UNITS (15)	R*8	000130	Alphanumeric information for printout.	BLOCK DATA	NONAME INOUPT
ELCUT	R*8	0001A8	Elevation cutoff angle.	INOUPT NEWARC	NONAME DATARD INOUPT
HYPER	L*4	0001B0	Hyperbolic element switch.	MAIN	MAIN DATARD ELEM

/APARAM/

COMMON/APARAM/ INPAR,INPARI,NBIAS,
ESTSTA,NSAT,NGPARC,NOREC1,NPARAM

Variable	Type	Hex Location	Description	Program Where Defined	Program Where Used
INPAR	I*4	000000	Number of force model parameters to be integrated.	NONAME NEWARC	NONAME NEWARC PREDCT STORE
INPARI	I*4	000004	Number of force model parameters in arc.	INOUPT ARCPAR	MAIN NONAME ARCPAR ESTIM INOUPT NEWARC STORE
NBIAS	I*4	000008	Number of biases in arc.	INOUPT BIAS	MAIN NONAME ARCPAR BIAS DODSRD ESTIM GEOSRD INOUPT NEWARC SIMRD STORE
ESTSTA	I*4	00000C	Number of adjusted stations.	NEWARC	NONAME NEWARC STORE
NSAT	I*4	000010	Number of satellites in arc.	MAIN	MAIN NONAME APPER ARCPAR BMTWRT BSCOMP ESTIM GEOSRD GRHRAN INOUPT NEWARC ORBIT PREDCT SIMRD STORE

/APARAM/ (Cont.)

Variable	Type	Hex Location	Description	Program Where Defined	Program Where Used
NGPARC	I*4	000014	Number of adjusted arc geopotential coefficients	INOUPT	MAIN NONAME ARCPAR BIAS DATARD ESTIM INOUPT NEWARC STORE UPDATE
RECNO1	I*4	000018	Number of record of first observation in arc.	INOUPT	NONAME INOUPT NEWARC STORE
NPARAM	I*4	00001C	Number of parameters in arc.	NONAME	NONAME BSCOMP DATARD ESTIM NEWARC STORE SUMMRY UPDATE
NEBIAS	I*4	0003DC	Number of electronic biases in arc.	INOUPT	MAIN NONAME ARCPAR BSCOMP CBROWN DATARD INOUPT NEWARC STORE
MAXPAR	I*4	00000C	Number of parameters per measurement for iteration.	ESTIM	BSCOMP ESTIM NEWARC STORE

/CELEM/

```
COMMON/CELEM/ELEMST(6,2),ORBELA(6,2),XNU,EC,RMSTOT
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
ELEMST (6,2)	R*8	000000	Epoch cartesian elements.	MAIN NONAME DODELM ESTIM ORBIT UPDATE	MAIN NONAME COWELL DATARD DODELM ESTIM INOUPT ORB1 ORBIT STORE TYPORB UPDATE
ORBELA (6,2)	R*8	000060	Epoch Kepler elements.	MAIN NONAME DODELM ORBIT	MAIN NONAME APPER DATARD DODELM INOUPT ORB1 ORBIT STORE TYPORB
XNU	R*8	0000C0	True anomaly.	ELEM	DATARD ELEM ORB1 STORE
EC	R*8	0000C8	Eccentric anomaly.	ELEM	DATARD ELEM ORB1 STORE
RMSTOT	R*4	0000D0	Total RMS	NONAME INOUPT	NONAME DATARD INOUPT NEWARC STORE SUMMRY TYPORB

/CEPHEM/ (Version 1)

COMMON/CEPHEM/AO,PMOON,SUN,ANUT,DUMMY

Variable	Type	Hex Location	Description	Program Where Defined	Program Where Use
AO (25)	R*8	000000	Unit vector and center of mass distance from earth to MOON SUN VENUS MARS JUPITER SATURN plus AO(25) = nutation in right ascension (equation of the equinoxes).	EPHEM	DENSTY EPHEM F GRHRAN PROCES SUNGRV TIDAL
PMOON (306)	R*8	0000C8	Buffer for lunar position and interpolation information	EPHEM	EPHEM
SUN (270)	R*8	000A58	Buffer for solar and planetary position and interpolation information	EPHEM	EPHEM
ANUT (102)	R*4	0012C8	Buffer for nutation information	EPHEM	EPHEM
DUMMY	R*8		Dummy		

/CEPHEM/ (Version 2)

COMMON/CEPEHM/JNAME,ISTARD,ESTANO,ISTANO

Variable	Type	Hex Location	Description	Program Where Defined	Program Where Used
JNAME (381)	R*8	000000	Names of stations read from input	INOUPT	DODSRD INOUPT
ISTARD (381)	I*2	00BE8	Numbers of stations read from input	INOUPT	DODSRD, GEOSRD INOUPT SIMRD
ESTANO (381)	I*2	000EE2	Master station array for station adjustment	INOUPT	INOUPT STAINP
ESTANO (386)	I*2	0011DC	Number of stations to be used	INOUPT DODSRD GEOSRD SIMRD	BIAS DODSRD GEOSRD INOUPT SIMRD STAINP

/CGEOS/ (Version 1)

COMMON/CGEOS/ISATID(2),THETGO(15),IG6(423)

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
ISATID(2)	R*4	000000	---	---	Not Used
THETGO (15)	R*8	000008	Right ascension of Greenwich in degrees on Jan.0.0 from 1958-1975.	JANTHG	JANTHG
IG6(423)	R*4	000080	---	---	Not Used

/CGEOS/ (Version 2)

```
COMMON/CGEOS/ISATID(2),IPREPR(4,50),RFINDX(2,50),
  INDPRE(2,50),NOPRPR,NSIG,NCULL,SIGCHG(50),
  IMTYPE(50),ISTNO(50),CULL(2,100)
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
ISATID (2)	I*4	000000	Satellite ID's.	INOUPT	NONAME DODSRD GEOSRD INOUPT ORB1 PROCES SIMRD TYPORB
IPREPR (4,50)	I*2	000008	Preprocessing indicators.	INOUPT	DODSRD GEOSRD INOUPT PRNTPR
RFINDX (2,50)	R84	000198	Tropospheric refraction indices and constant timing corrections	INOUPT	DODSRD GEOSRD INOUPT PRNTPR
INDPRE (2,50)	I*2	000328	Station numbers and measurement types for preprocessing.	INOUPT	DODSRD GEOSRD INOUPT PRNTPR
NOPRPR	I*4	0003F0	Number of PREPO cards input.	INOUPT	DODSRD GEOSRD INOUPT NEWARC PRNTPR
NSIG	I*4	0003F4	Number os SIGMA cards input.	INOUPT	DODSRD GEOSRD INOUPT NEWARC PCERD SIMRD
NCULL	I*4	0003F8	Number of CULL sets.	INOUPT	DOSDRD GEOSRD INOUPT NEWARC PCERD SIMRD

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
SIGCHG (50)	R*4	0003FC	Input sigma changes.	INOUPT	DODSRD GEOSRD INOUPT NEWARC PCERD SIMRD
IMTYPE (50)	I*2	0004C4	Sigma change types.	INOUPT	DODSRD GEOSRD INOUPT NEWARC PCERD SIMRD
ISTNO (50)	I*2	000528	Sigma change stations.	INOUPT	DODSRD GEOSRD INOUPT NEWARC PCERD SIMRD
CULL (2,100)	I*2	0058C	Cull sets.	INOUPT	DODSRD GEOSRD INOUPT NEWARC PCERD SIMRD

/CONOUT/

```
COMMON/CONOUT/RMSALL,OUTCON,MINOUT,MAXOUT,LITRES,
  MAXSAT,MAX2IN,NSTART,NEQNMX,IVAR,IORDER,NARCS,
  NSTARD,LSTART(6)
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
RMSALL	R*4	000000	RMS for all arcs.	MAIN	MAIN NONAME
OUTCON	R*4	000004	Outer iteration convergence criterion.	INOUPT	NONAME INOUPT
MINOUT	I*4	000008	Minimum number of outer iterations.	INOUPT	NONAME INOUPT
MAXOUT	I*4	00000C	Maximum number of outer iterations.	INOUPT	NONAME INOUPT
LITRES	L*4	000010	Indicates that adjustment not requested on last inner iteration.	INOUPT	NONAME INOUPT
MAXSAT	I*4	000014	Maximum number of satellites per arc.	MAIN	MAIN CBROWN DATARD
MAX2IN	I*4	000018	Maximum number of inner iterations on outer iterations after first outer iteration.	NONAME	NONAME
NSTART	I*4	00001C	Parameter number of first common parameter.	COMPAR ESTIM	NONAME CBROWN COMPAR ESTIM
NEQNMX	I*4	000020	Maximum number of force model equations to be integrated for one arc.	MAIN	MAIN CBROWN ESTIM ORBIT

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
IVAR	I*4	000024	=1 for fixed step integration =2 for variable step integration	MAIN	MAIN
IORDER	I*4	000028	Not Used.		
NARCS	I*4	00002C	Number of arcs.	MAIN	MAIN NONAME ARCPAR
NSTARD	I*4	000030	Number of tracking stations read from input.	INOUPT	COMPAR INOUPT
STARTR	L*4	000034	.TRUE.- Restart tape input.	INOUPT	NONAME INOUPT
STARTW	L*4	000038	.TRUE.- Restart tape output.	INOUPT	NONAME DATARD INOUPT
STARTA	I*4	00003C	Number of arc where restart will begin.	INOUPT	NONAME INOUPT
STARTO	I*4	000040	Number of outer iteration where restart will begin. (not used)	INOUPT	NONAME INOUPT
INSTRT	I*4	000044	Input restart tape number.	INOUPT	NONAME INOUPT
OUTSTR	I*4	000048	Output restart tape number.	INOUPT	NONAME DATARD INOUPT

/CONSTS/

COMMON/CONSTS/DPI,DTWOPI,DRAD,DRSEC

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Defined
DPI	R*8	000000	π	BLOCK DATA	NONAME AVGPOT DENSTY PREDCT PROCES TYPORB
DTWOPI	R*8	000008	2π	BLOCK DATA	NONAME DENSTY DPFCT ELEM GEOSRD GRHRAN NEWARC ORB1 PREDCT PROCES TYPORB
DRAD	R*8	000010	$2\pi/360$ Conversion factor for converting degrees to radians.	BLOCK DATA	NONAME APPER ARCPAR AREAS AVGPOT DELTAZ DODELM ELEM GEOSRD INDENT JANTHG NEWARC ORB1 PCERD POSVEL PREDCT STAINP TWPSTA TYPORB

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
DRSEC	R*8	000018	2π/360/3600	BLOCK DATA	NONAME COMADJ COMPAR DODSRD GEOSRD INOUPT PCERD PROCES SIMRD SUMMRY

/CORB1/

```
COMMON/CORB1/RANDOT(2),PERDOT(2),PERHT(2),APHT(2),
  PRD(2)
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
RANDOT(2)	R*8	000000	Time derivative of the right ascension of the ascending node.	NONAME	NONAME INOUPT ORB1 PREDCT TWOSTA TYPORB
PERDOT(2)	R*8	000010	Time derivative of the argument of perigee.	NONAME	NONAME ORB1 PREDCT TWOSTA TYPORB
PERHT(2)	R*8	000020	Perigee height.	APPER	APPER INOUPT ORB1 TYPORB
APHT(2)	R*8	000030	Apogee height.	APPER	APPER ORB1 TYPORB
PRD(2)	R*8	000040	Orbital period.	NONAME	NONAME INOUPT ORB1 PREDCT TYPORB

/CPARAM/

```
COMMON/CPARAM/ NSTA,NMAST,NSTEST,NDIM,MBIAS
              NGPC1,NGPC2,NGPCOM,NCSEST,CMPGPR,
              LIM1,LIM2,NDEN,NDENST,NTIDST,
              NTIDEN,INNRSW,NCONST,NDCONS
```

Variable	Type	Hex Location	Description	Program Where Defined	Program Where Used
NSTA	I*4	000000	Number of tracking stations.	DODSRD GEOSRD SIMRD INOUPT CBROWN	MAIN ARCPAR CBROWN COMPAR CONSTS DODSRD GEOSRD INOUPT SIMRD SUMMRY
NMAST	I*4	000004	Number of adjusted master stations.	INOUPT	MAIN NONAME BMTWRT CBROWN COMADJ COMPAR CONSTS ESTIM INOUPT PDEN
NSTEST	I*4	000008	Number of estimated stations.	INOUPT	MAIN NONAME BMTWRT CBROWN COMADJ COMPAR CONSTS ESTIM INOUPT
NDIM	I*4	00000C	Maximum dimension of normal matrix.	MAIN	MAIN NONAME BMTWRT BSCOMP CBROWN COMADJ COMPAR CONSTS DATARD ESTIM PDEN UPDATE

Variable	Type	Hex Location	Description	Program Where Defined	Program Where Used
MBIAS	I*4	000010	Maximum number of biases plus drag and solar radiation pressure parameters estimated in any one arc.	MAIN	MAIN ARCPAR CBROWN CONSTS DATARD ESTIM UPDATE
NGPC1	I*4	000014	Relative location of first common adjusted geopotential coefficient.	MAIN	MAIN CONSTS DATARD ESTIM
NGPC2	I*4	000018	Relative location of last common adjusted geopotential coefficient.	MAIN	MAIN CONSTS ESTIM
NGPCOM	I*4	00001C	Number of common adjusted geopotential coefficients.	MAIN	MAIN NONAME ARCPAR BMTWRT BSCOMP COMADJ COMPAR CONSTS DATARD ESTIM
NCSEST	I*4	000020	Number of adjusted geopotential coefficients in this iteration.	INOUPT NONAME	MAIN NONAME ARCPAR BMTWRT CBROWN COMADJ COMPAR CONSTS DATARD INOUPT RESPAR

/CPARAM/ (Cont.)

Variable	Type	Hex Location	Description	Program Where Defined	Program Where Used
CMPGPR	L*4	000024	Logical switch for geopotential partial computations True- partials are computed.	NONAME	MAIN NONAME ARCPAR CONSTS VEVAL
LIM1	I*4	000028	Size of normal arrays to be cleared.	NONAME	NONAME CONSTS ESTIM
LIM2	I*4	00002C	.GT.0 indicates last inner iteration.	NONAME	NONAME CONSTS ESTIM
NDEN	I*4	000030	Number of surface densities.	INOUPT	MAIN NONAME CBROWN COMPAR CONSTS GEOIDH INDENT INOUPT PDEN SURDEN
NDENST	I*4	000034	Number of adjusted surface densities.	INOUPT	MAIN CBROWN COMADJ CONSTS GEOIDH INDENT INOUPT PDEN SURDEN
NTIDST	I*4	000038	Number of adjusted tidal parameters.	CONSTS	MAIN CBROWN CONSTS

Variable	Type	Location	Description	Program Where Defined	Program Where Used
NTIDEN	I*4	00003C	Number of adjusted tidal and density parameters.	MAIN	MAIN NONAME BSCOMP CBROWN COMADJ COMPAR CONSTS ESTIM RESPAR
INNRSW	L*4	000040	Logical switch. True-last inner iteration and surface densities adjusted.	NONAME	NONAME CONSTS SURDEN
NCONST	I*4	000044	Number of surface density constraint equations.	INOUPT	MAIN CBROWN CONSTS GEOIDH INOUPT PDEN SURDEN
NDCONS	I*4	000048	Maximum degree and order of surface density constraint equations.	INOUPT	CONSTS GEOIDH INOUPT

/CSLIM/

COMMON/CSLIM/LLIMIT,ULIMIT

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
LLIMIT	I*4	000000	LLIMIT(N) is the minimum order plus one of terms of degree N which are used in the geopotential expansion.	EGRAV	EGRAV
ULIMIT	I*4	00007C	ULIMIT(N) is the maximum order plus one of terms of degree N which are used in the geopotential expansion.	EGRAV	EGRAV GEOIDH

/CSTAT/

COMMON/CSTAT/RESID,SIG,NMTOT,WTSUMT

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
RESID	R*8	000000	Measurement residual.	STAINF	STAINF SMSTAT
SIG	R*8	000004	Measurement sigma.	STAINF	STAINF SMSTAT
NMTOT	I*4	000008	Total number of weighted measurements in arc.	STAINF SMSTAT	STAINF SMSTAT
WTSUMT	R*8	00000C	Sum of squares of weighted residuals for the arc.	STAINF SMSTAT	BMTWRT STAINF SMSTAT

/CSTHET/

COMMON/CSTHET/CTHETG,STHETG

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
CTHETG	R*8	000000	Cosine of right ascension of Greenwich.	GRHRAN SURDEN	DPFCT GRHRAN SURDEN
STHETG	R*8	000008	Sine of right ascension of Greenwich.	GRHRAN SURDEN	DPFCT GRHRAN SURDEN

/CTIME/

```
COMMON/CTIME/DATAEP,DAYREF,DSTART,DAYSTP,DAYINT,
  DORBIT,DAYEND,DRATE,DORB1,DORB1E,ORBRT,IYBEG
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
DATAEP	R*8	000000	Epoch of data in days from Jan 0.0 of the reference year.	MAIN	MAIN NONAME DATARD DENSTY DODELM DODSRD GEOSRD INOUPT ORBIT PCERD SIMRD STORE TYPORB
DAYREF	R*8	000008	Reference date in days from Jan 0.0 of the reference year.	MAIN	MAIN DATARD ORB1 REFCOR STORE TYPORB
DSTART	R*8	000010	Epoch in days from Jan 0.0 of the reference year.	MAIN DODELM ORBIT	MAIN NONAME COWELL DATARD DODELM F INOUPT ORB1 ORBIT STORE TYPORB

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
DAYSTP	R*8	000018	Data stop time in days from Jan 0.0 of the reference year.	MAIN	MAIN DATARD DODSRD GEOSRD INOUPT NEWARC PCERD SIMRD STORE TYPORB
DAYINT	R*8	000020	Current integration time in days from Jan 0.0 of the reference year.	F	DATARD DENSTY F STORE
DORBIT	R*8	000028	Start time for orbit generator in days from Jan 0.0 of the reference year.	INOUPT NEWARC	NONAME DATARD INOUPT NEWARC STORE
DAYEND	R*8	000030	Stop time for orbit generator in days from Jan 0.0 of the reference year.	INOUPT NEWARC	NONAME DATARD INOUPT NEWARC STORE
DRATE	R*8	000038	Output interval for orbit generator in days.	INOUPT NEWARC	NONAME DATARD INOUPT NEWARC STORE
DORB1	R*8	000040	Start time for ORB1 tape in days from Jan 0.0 of the reference year.	INOUPT NEWARC	NONAME DATARD INOUPT NEWARC ORB1 STORE

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
DORB1E	R*8	000048	Stop time for ORB1 tape in days from Jan 0.0 of the reference year.	INOUPT NEWARC	DATARD INOUPT ORB1 STORE
ORBRT	R*8	000050	Output interval for ORB1 tape in integral seconds.	INOUPT NEWARC	NONAME DATARD INOUPT STORE
IYBEG	I*4	000058	Reference year.	MAIN	MAIN DATARD DATES STORE YMDAY

/CSTINF/ (Version 1)

```
COMMON/CSTINF/MEASO(4),NOBS(4),RDMEAN(4),RMSO(4)
  RND(4),MEASWT(4),WTMEAN(4),RMSWTO(4),WTRND(4),
  TYPRMS(4),NOTYPE(2,30)BSUM(8,12),RMSALL(30),
  NOALL(30),NOWTOB,JBASE
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
MEASNO	R*8	000000	Measurement type numbers.	RMSCMP	RMSCMP SUMMRY
NOBS(4)	I*4	000010	Number of measurements.	RMSCMP	RMSCMP STAINF SUMMRY
RDMEAN(4)	R*4	000020	Residual means.	RMSCMP	RMSCMP SUMMRY
RMSO(4)	R*4	000030	RMS's of residuals.	RMSCMP	RMSCMP SUMMRY
RND(4)	R*4	000040	RND's of residuals.	RMSCMP	RMSCMP SUMMRY
MEASWT(4)	I*4	000050	Number of weighted measurements.	RMSCMP	RMSCMP SUMMRY
WTMEAN(4)	R*4	000060	Weighted residual measurements.	RMSCMP	RMSCMP SUMMRY
RMSWTO(4)	R*4	000070	RMS's of weighted residuals.	RMSCMP	RMSCMP SUMMRY
WTRND(4)	R*4	000080	RND's of weighted measurements.	RMSCMP	RMSCMP SUMMRY
TYPRMS(4)	R*4	000090	Measurement type weighted RMS's.	RMSCMP STAINF	NONAME RMSCMP STAINF SUMMRY TYPORB
NOTYPE (2,30)	I*4	000108	Number of measurements by type.	RMSCMP STAINF	RMSCMP STAINF SUMMRY TYPORB

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
BSUM (8,12)	R*4	0001F8	Summing arrays for PCE measurement types.	STAINF	STAINF
RMSALL (300)	R*4	000378	Measurement type weighted RMS's for all arc.	STAINF	STAINF
NOALL (300)	I*4	0003F0	Number of measurements by type for all arcs.	STAINF	NONAME STAINF
NOWTOB	I*4	000468	Total number of weighted observations.	SUMMRY	NONAME BMTWRT SUMMRY TYPORB
JBASE	I*4	00046C	Number of station measurement base lines	GEOSRD SIMRD CBROWN	CBROWN GEOSRD SIMRD SUMMRY

/CSTINF (Version 2)

COMMON/CSTINF/JBASE(283),KBASE(283),LBASE

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
JBASE (283)	I*2	000000	First station in a measurement baseline.	GEOSRD SIMRD	MAIN GEOSRD SIMRD
KBASE (283)	I*2	000236	Second station in a measurement baseline.	GEOSRD SIMRD	MAIN GEOSRD SIMRD
LBASE	I*4	00046C	Number of station measurement baselines.	GEOSRD SIMRD MAIN	MAIN GEOSRD SIMRD

/CUVECT/

```
COMMON/CUVECT/UHAT(3,2),XYZ(3,2),RXYZ(3,2),
  RENV(3,2),R(2),RSQ(2),XYZSQ(2)
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
UHAT(3,2)	R*8	000000	Earth fixed unit vector from station to satellite.	GRHRAN	GRHRAN OBSDOT PREDCT
XYZ(3,2)	R*8	000030	Earth centered fixed satellite vector.	GRHRAN	GRHRAN OBSDOT PREDCT PROCES
RXYZ(3,2)	R*8	000060	Earth fixed vector from station to satellite.	GRHRAN	GRHRAN PREDCT PROCES
RENV(3,2)	R*8	000090	Station-satellite unit local vector (direction cosines).	GRHRAN	GRHRAN OBSDOT PREDCT PROCES TWOSTA UPDOWN
R(2)	R*8	0000C0	Slant range from station to satellite.	GRHRAN	GRHRAN OBSDOT PREDCT PROCES TWOSTA UPDOWN
RSQ(2)	R*8	0000D0	R*R	GRHRAN	GRHRAN PREDCT
XYZSQ(2)	R*8	0000E0	RXYZ(1,ISAT)**2 + RXYZ(2,ISAT)**2 where ISAT = satellite index.	GRHRAN	GRHRAM PREDCT

/DODDAT/

```
COMMON/DODDAT/TIME1,STNAM1,OBD1,DG(2),OBSCOR,
SATNO,IOBNO1,IWT(6),TCOR,IG2(2),IT,IG1,TTAG,PBIT1,PBIT2,IG
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
TIME1	R*8	000000	Time of DODS observation	DATBSE	DATBSE DODSRD
STNAM1	R*8	000008	Station ID	DATBSE	DATBSE DODSRD
OBD1	R*8	000010	DODS observation	DATBSE	DATBSE DODSRD
DG(2)	R*8	000018	DG(1)-not used DG(2)-DODS Range ambiguity information (bits 56-59)	DATBSE	DATBSE DODSRD
OBSCOR	R*8	000028	DODS observation correction	DATBSE	DATBSE DODSRD
SATNO	I*4	000030	Satellite number	DATBSE	DATBSE DODSRD
IOBNO1	I*4	000034	DODS observation number	DATBSE	DATBSE
IWT(6)	R*4	000038	Not Used	DATBSE	
TCOR	R*4	000050	Time correction	DATBSE	DATBSE DODSRD
IG2 (2)	I*2	000054	Not Used	DATBSE	
IT	I*2	000058	DODS observation type number	DATBSE	DATBSE DODSRD
IG1	I*2	00005A	Not Used	DATBSE	
TTAG	I*2	00005C	Time type and station source indicator	DATBSE	DATBSE DODSRD
PBIT1	I*2	00005E	Preprocessing indicators for corrections added prior to DODS (bits 10-15)	DATBSE	DATBSE DODSRD

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
PBIT2	I*2	000060	Preprocessing indicators for corrections added by DODS (bits 10-15)	DATBSE	DATBSE DODSRD
IG	I*2	000062	Not Used	DATBSE	

/DRGBLK/
COMMON/DRGBLK/HT,SPSISQ, C(4),C3,
C1,VEL,XDOTR,YDOTR,RHO

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
HT.	R*8	000000	DRAG-satellite spheroid height in meters DENSTY-IN- same as in DRAG -OUT- same as in drag multiplied by 10^{-5}	DRAG DENSTY	DENSTY DRAG VEVAL
SPSISQ	R*8	000008	$\sin^2$(geocentric latitude)	DRAG	DRAG VEVAL
C(4)	R*8	000010	C(1)=partial of the density with respect to the spheroid height divided by the density C(I), I=2,4 scratch	DENSTY	DENSTY VEVAL
C3	R*8	000030	The density times the relative air speed	DRAG	DENSTY DRAG VEVAL
C1	R*8	000038	$C_3(1/2)C_D(A/m)$	DRAG	DENSTY DRAG VEVAL
VEL	R*8	000040	Relative air speed of the satellite	DRAG	DENSTY DRAG VEVAL
XDOTR	R*8	000048	x-component of the relative air speed	DRAG	DENSTY DRAG VEVAL
YDOTR	R*8	000050	y-component of the relative air speed	DRAG	DENSTY DRAG VEVAL
RHO	R*8	000058	Atmospheric density in kg/m^3	DRAG	DRAG VEVAL

/FERMSG/
COMMON/FERMSG/IMES(26)

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
IMES (26)	I*4	000000	See IBM SYSTEM/360 GENERAL I/O PACKAGE, Alan Thompson, IBM July 14, 1970 pages 13 and 14.	Defined in system routine DREAD or DWRITE when they encounter I/O error.	ERROR

/FLXBLK/ (Version 1)

COMMON/FLXBLK/AVFLX(675),DFLX(675),KP(675)

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
AVFLX (675)	R*8	000000	Three solar rotation midpoint average 10.7cm flux values. Average provided daily beginning with 12 hours GMT two days prior to day of data epoch.	JANTHG	NONAME ADFLUX BMTWRT DENSTY INDENT JANTHG NEWARC
DFLX (675)	R*8	001518	Daily 10.7cm flux values beginning 12 hours GMT two days prior to day of data epoch.	JANTHG	NONAME ADFLUX BMTWRT DENSTY INDENT JANTHG NEWARC
KP (675)	R*8	002A30	Daily mean of geomagnetic indices K_p beginning 12 hours GMT two days prior to day of data epoch	JANTHG	NONAME ADFLUX BMTWRT DENSTY INDENT JANTHG NEWARC

/FLXBLK/ (Version 2)

COMMON/FLXBLK/INDXCS,(960,3)PLHSIG,PLHSW

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
INDXCS (960,3)	I*2	000000	Indexes for common adjusted geopotential coefficients. INDXCS(I,1)=degree of I^{th} adjusted coefficient. INDXCS(I,2)=Order of I^{th} adjusted coefficient. INDXCS(I,3) = 1 for C's = 2 for S's	INOUPT	COMPAR INOUPT STORE
PLHSIG	R*4	001680	Sigmas and correlations on adjusted station positions.	INOUPT	COMPAR INOUPT STORE
PLHSW	L*1	003A38	Switches telling whether adjusted station sigmas and correlations refer to Cartesian or geodetic coordinates	INOUPT	COMPAR INOUPT STORE

/FLXBLK/ (VERSION 3)

COMMON/FLXBLK/BSTRT(900),BSEND(900),BTYPE(900)

Variable	Type	Hex Location	Description	Program Where Defined	Program Where Used
BSTRT (900)	R*8	000000	Start time for bias adjustment coverage	INOUPT BIAS	ARCPAR BIAS INOUPT STORE
BSEND (900)	R*8	001C20	Stop time for bias adjustment coverage	INOUPT BIAS	ARCPAR BIAS INOUPT STORE
BTYPE (900)	I*2	003840	Types for bias adjustment	INOUPT BIAS	ARCPAR BIAS INOUPT STORE

/FMODEL/

```
COMMON/FMODEL/INDEX1,INDEX2,INDEX3,INDEX4,CS(30,33),
  MODEL(8)
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
INDEX1	I*4	000000	Maximum degree of geopotential plus 1.	BLOCK DATA INOUPT	COEFL DATARD F GEOIDH INOUPT NEWARC STORE
INDEX2	I*4	000004	Maximum degree and order of Legendre polynomials to be computed.	F	AVGPOT DATARD EGRAV F GEOIDH STORE
INDEX3	I*4	000008	Maximum order of geopotential plus 1.	BLOCK DATA INOUPT	COEFL DATARD EGRAV INOUPT NEWARC STORE
INDEX4	I*4	00000C	Maximum degree and order of geopotential for variational equations.	BLOCK DATA INOUPT	DATARD F INOUPT STORE VEVAL
CS (30,33)	R*8	000010	Geopotential coefficients.	BLOCK DATA INOUPT NONAME	NONAME ARCPAR AVGPOT COEFL COMADJ COMPAR EGRAV GEOIDH INOUPT ORB1 VEVAL
MODEL (8)	R*8	001F00	Name of geopotential.	BLOCK DATA	COEFL

/GEODYN/

COMMON/GEODYN/DATE

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
DATE	R*8	000000	GEODYN system date and source tape number.	BLOCK DATA	TYPORB

/GNDTRK/

```
COMMON/GNDTRK/SATLAT(2),SATLON(2),SATH(2),ELEV(2),SATSW
```

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
SATLAT (2)	R*8	000000	Satellite latitude.	PREDCT	NONAME PREDCT
SATLON (2)	R*8	000010	Satellite longitude.	PREDCT	NONAME PREDCT
SATH (2)	R*8	000020	Satellite height.	PREDCT	NONAME PREDCT
ELEV (2)	R*8	000030	Satellite elevation angle.	PREDCT	NONAME PREDCT PROCES
SATSW	L*4	000040	Switch requesting computation of SATLAT,SATLON, SATH.	NONAME	NONAME

/INITBK/

```
COMMON/INITBK,IEPYMD,IEPHM,EPSEC,IYREF,INNMAX,INNMIN,
  CONVRG,ORBEL(6,2),EDITN,INSUPR,IDSAT(2),ORBTSW,
  XYZFSW,SYZLSW,PLTLSW,GRDFSW,KEPLER,SUBSAT,PARTGP,
       PBMAT,BMATNO,SIMDAT,PCESIM,MISLOG(9)
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
IEPYMD	I*4	000000	Epoch YYMMDD	MAIN DODELM ORBIT	MAIN NONAME DATARD DODELM INOUPT ORBIT STORE TYPORB
IEPHM	I*4	000004	Epoch HHMM	MAIN DODELM ORBIT	MAIN NONAME DATARD DODELM INOUPT ORBIT STORE TYPORB
EPSEC	R*8	000008	Epoch seconds	MAIN DODELM ORBIT	MAIN NONAME DATARD DODELM INOUPT ORBIT STORE TYPORB
IYREF	I*4	000010	Reference date	MAIN	MAIN NONAME DATARD INOUPT NEWARC STORE
INNMAX	I*4	000014	Maximum number of inner iterations	INOUPT	MAIN NONAME DATARD INOUPT NEWARC STORE

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
INNMIN	I*4	000018	Minimum number of inner iterations	INOUPT	MAIN NONAME DATARD INOUPT NEWARC STORE
CONVRG	R*4	00001C	Inner iteration convergence criterion	INOUPT	MAIN NONAME DATARD INOUPT NEWARC STORE
ORBEL (6,2)	R*8	000020	Nominal epoch elements	MAIN	MAIN NONAME DATARD INOUPT ORBIT STORE
EDITN	R*4	000080	Editing multiplier	INOUPT	NONAME DATARD INOUPT STORE SUMMRY
INSUPR	I*4	000084	Residual printing indicator	MAIN INOUPT NEWARC	MAIN NONAME DATARD INOUPT NEWARC STORE
IDSAT (2)	I*4	000088	Satellite ID's for ORB1 tapes	INOUPT	NONAME DATARD INOUPT STORE
ORBTSW	L*4	000090	Orbit generator switch	INOUPT NEWARC	MAIN NONAME DATARD INOUPT STORE TYPORB

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
XYZFSW	L*4	000094	Switch for satellite ephemeris on first inner of first outer	INOUPT NEWARC	NONAME DATARD INOUPT STORE
XYZLSW	L*4	000098	Switch for satellite ephemeris on last inner of last outer	INOUPT NEWARC	NONAME DATARD INOUPT STORE
PLTLSW	L*4	00009C	Binary residual tape switch	INOUPT NEWARC	NONAME DATARD INOUPT STORE
GRDFSW	L*4	0000A0	Groundtrack tape switch	INOUPT NEWARC	NONAME DATARD INOUPT STORE
KEPLER	L*4	0000A4	Kepler ephemeris switch	INOUPT NEWARC	NONAME DATARD INOUPT STORE
SUBSAT	L*4	0000A8	Satellite ground-track switch	NONAME NEWARC	NONAME DATARD PREDCT STORE
PARTGP	L*4	0000AC	Partial derivative print switch	INOUPT NEWARC	NONAME DATARD INOUPT STORE
PBMAT	L*4	0000B0	B-matrix print switch	INOUPT NEWARC	BMTWRT DATARD INOUPT STORE
BMATNO	I*4	0000B4	B-matrix number	INOUPT NEWARC	NONAME BMTWRT DATARD ESTIM INOUPT STORE

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
SIMDAT	L*4	0000B8	Simulated data switch	INOUPT NEWARC	NONAME DATARD INOUPT STORE
PCESIM	L*4	0000BC	Simulated element data type switch	INOUPT	NONAME DATARD INOUPT STORE
MISLOG (9)	I*4	0000C0	Initialization switches for several subroutines 1 - DENSTY 2 - DPFCT 3 - EGRAV 4 - EPHEM 5 - NUTATE 6 - PRECES 7 - REFCOR 8 - TDIF 9 - TIDAL	DATARD GEOIDH	NONAME DATARD DENSTY DPFCT EGRAV EPHEM GEOIDH NUTATE PRECES REFCOR TDIF TIDAL

/INTBLK/

```
COMMON/INTBLK/THDOT1,THDOT2,THDT2S,GM,AE,AESQ,FLAT,
  FSQ32,FFSQ32,GM3(6),B(2),BDOT(2),B0(2),APGM(2),APLM(2),
  RPRESS,INITAL,NORRAT,THETG0,MBODY(6),STEPSZ(2,2),
  HLVERB(2),DBLERB(2),CTOL(2),RTOL(2),STPLOW(2),STEPUP(2),
  ORDER(2,2),ASAT(2),MSAT(2),VARSTP(2),HLVDSW(2),NEQN(2),
  ADDR(2),ADDRD(2),SRAD(2),LOVE(3),TOREFT,NBODY
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
THDOT1	R*8	000000	Mean advance in right ascension of Greenwich in radians per mean solar day.	BLOCK DATA NEWARC	F GRHRAN NEWARC ORB1
THDOT2	R*8	000008	Rotation rate of earth in radians per mean solar day.	NEWARC	F GRHRAN NEWARC
THDT2S	R*8	000010	Rotation rate of earth in radians per second.	NEWARC	AVGPOT DRAG EPHEM GEOIDH NEWARC OBSDOT PREDCT VEVAL
GM	R*8	000018	Universal gravitational constant times mass of earth.	BLOCK DATA INOUPT	AVGPOT EGRAV ELEM GEOIDH INOUPT OBSDOT ORB1 ORBIT POSVEL START TYPORB VEVAL

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
AE	R*8	000020	Semimajor axis of earth ellipsoid.	BLOCK DATA INOUPT	APPER AREAS AVGPOT DELTAZ DRAG EGRAV GEOIDH INDENT INOUPT ORBI ORBIT PLHOUT PREDCT SQUANT STAINP TIDAL TYPORB
AESQ	R*8	000028	AE * AE	INOUPT	AREAS F INOUPT ORBIT
FLAT	R*8	000030	Flattening of earth ellipsoid.	BLOCK DATA INOUPT	AREAS AVGPOT DELTAZ INDENT INOUPT PLHOUT PREDCT SQUANT STAINP
FSQ32	R*8	000038	3/2 AE * FLAT**2	INOUPT	APPER DRAG INOUPT PREDCT VEVAL
FFSQ32	R*8	000040	AE * FLAT + FSQ32	INOUPT	APPER DRAG INOUPT PREDCT VEVAL

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
GM3 (6)	R*8	000048	Universal gravitational constant times mass of perturbing celestial bodies ordered as follows: 1 - MOON 2 - SUN 3 - VENUS 4 - MARS 5 - JUPITER 6 - SATURN	ORBIT	ORBIT SUNGRV TIDAL VEVAL
B(2)	R*8	000078	1/2 C_D A/m	ORBIT	COWELL DRAG F ORBIT TIDAL VEVAL
BDOT (2)	R*8	000088	1/2 $\dot{C}_D$ A/m	ORBIT	DRAG F ORBIT TIDAL
B0(2)	R*8	000098	1/2 A/m	ORBIT	NONAME DRAG F ORBIT TIDAL
APGM (2)	R*8	0000B8	P_R A/m (P_R = solar radiation pressure constant)	ORBIT	F ORBIT TIDAL
APLM (2)	R*8	0000A8	C_R P_R A/m	ORBIT	F ORBIT TIDAL

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
RPRESS	R*8	0000C8	Solar radiation pressure constant.	BLOCK DATA	F INOUPT ORBIT TIDAL
INITAL	L*4	0000D0	Integration initialization switch.	NONAME	NONAME COWELL F ORBIT TIDAL
NORRAT	L*4	0000D4	No range rate switch.	GEOSRD DODSRD	NONAME DATARD DODSRD F GEOSRD PCERD SIMRD TIDAL
THETG0	R*8	0000D8	Right ascension of Greenwich on Jan 0.0 of reference year.	JANTHG	NONAME ADFLUX DATARD F GRHRAN ORB1 STORE TIDAL
MBODY (6)	R*8	0000E0	Ratios of masses of perturbing bodies to mass of earth ordered as follows: 1 - MOON 2 - SUN 3 - VENUS 4 - MARS 5 - JUPITER 6 - SATURN	BLOCK DATA INOUPT	DATARD INOUPT NEWARC ORB1 ORBIT STORE TIDAL

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
STEPSZ (2,2)	R*8	000110	Integration step size.	BLOCK DATA INOUPT NEWARC ORBIT	NONAME DATARD INOUPT NEWARC ORBIT STORE TIDAL
HLVERB (2)	R*8	000130	Step size reduction criterion.	BLOCK DATA INOUPT NEWARC	COWELL DATARD INOUPT NEWARC STORE TIDAL
DBLERB (2)	R*8	000140	Step size increase criterion.	BLOCK DATA INOUPT NEWARC	COWELL DATARD INOUPT NEWARC STORE TIDAL
CTOL (2)	R*8	000150	Critical tolerance for determining number of integrator correction iterations.	BLOCK DATA INOUPT NEWARC	COWELL DATARD INOUPT NEWARC STORE TIDAL
RTOL (2)	R*8	000160	Tolerance used to determine new step size in vary-step integrator.	BLOCK DATA INOUPT NEWARC	COWELL DATARD INOUPT NEWARC STORE TIDAL
STPLOW (2)	R*8	000170	Minimum integrator step size.	BLOCK DATA INOUPT NEWARC	COWELL DATARD INOUPT NEWARC STORE TIDAL

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
STEPUP (2)	R*8	000180	Maximum integrator step size.	BLOCK DATA INOUPT NEWARC	COWELL DATARD INOUPT NEWARC STORE TIDAL
ORDER (2,2)	I*4	000190	Integration orders.	BLOCK DATA INOUPT NEWARC	MAIN DATARD INOUPT NEWARC ORBIT STORE TIDAL
ASAT (2)	R*8	0001A0	Satellite areas.	INOUPT NEWARC	NONAME DATARD INOUPT NEWARC ORB1 ORBIT STORE TIDAL TYPORB
MSAT (2)	R*8	0001B0	Satellite masses.	INOUPT NEWARC	NONAME DATARD INOUPT NEWARC ORB1 ORBIT STORE TIDAL TYPORB
VARSTP (2)	L*4	0001C0	Vary-step switch.	INOUPT NEWARC	MAIN DATARD ERROR INOUPT NEWARC ORBIT STORE TIDAL

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
HLVDSW (2)	L*4	0001C8	Halving-doubling switch.	INOUPT NEWARC	COWELL DATARD INOUPT NEWARC STORE TIDAL
NEQN (2)	I*4	0001D0	Number of integrator equations.	NONAME	MAIN NONAME DATARD INOUPT NEWARC ORBIT PREDCT RESPAR STORE SURDEN TIDAL VEVAL
ADDR (2)	I*4	0001D8	Drag parameter numbers.	INOUPT	NONAME ARCPAR BMTWRT DATARD DRAG INOUPT NEWARC ORBIT PREDCT STORE SURDEN TIDAL TYPORB VEVAL

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
ADDRD (2)	I*4	0001E0	Drag rate parameter numbers.	INOUPT	NONAME ARCPAR BMTWRT DATARD DRAG INOUPT NEWARC ORBIT PREDCT STORE SURDEN TIDAL TYPORB VEVAL
SRAD (2)	I*4	0001E8	Solar radiation parameter numbers.	INOUPT	NONAME ARCPAR BMTWRT DATARD F INOUPT NEWARC ORBIT PREDCT STORE SURDEN TIDAL TYPORB VEVAL
LOVE (3)	R*8	0001F0	Tidal parameter values.	BLOCK DATA INOUPT	MAIN ARCPAR DATARD INOUPT STORE SURDEN TIDAL TYPORB VEVAL

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
TOREFT	L*4	000208	Switch for output in true equator and equinox of reference time.	INOUPT NEWARC	MAIN NONAME ARCPAR DATARD INOUPT NEWARC ORBIT STORE SURDEN TYPORB VEVAL
NBODY	I*4	00020C	Number of perturbing celestial bodies.	BLOCK DATA INOUPT NEWARC	MAIN ARCPAR DATARD EPHEM INOUPT NEWARC STORE SUNGRV SURDEN TYPORB VEVAL

/INTERP/

COMMON/INTERP/COMB(21,21),M12(4)

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
COMB (21, 21)	R*8	000000	Binomial coefficient array.	COM	COEF COM
M12 (4)	I*4	000DC8	Displacement array used by integrator and interpolator.	COWELL ORBIT	ORBIT

/MONTHS/

COMMON/MONTHS/MONTH(26)

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
MONTH (26)	I*4	000000	Number of elapsed days at the beginning of each month for leap years and for non-leap years.	BLOCK DATA	ADDYMD DIFF

/MOONGR/

COMMON/MOONGR/DPXUV(6),RHOM(3,6),RHOSQ(6),RHO3(6)

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
DPXUV (6)	R*8	000000	Dot products of satellite position vector with unit vectors to perturbing celestial bodies.	SUNGRV	F SUNGRV TIDAL
RHOM (3,6)	R*8	000030	Satellite position vectors in coordinates centered at perturbing bodies.	SUNGRV	SUNGRV VEVAL
RHOSQ (6)	R*8	0000C0	Square of the difference of the distance of the satellite from the earth and the distance of the satellite from the perturbing body.	SUNGRV	SUNGRV VEVAL
RHO3 (6)	R*8	0000F0	RHOSQ**1.5	SUNGRV	SUNGRV VEVAL

/PREBLK/

```
COMMON/PREBLK/DAYSTA,OBS1,OBS2,SIG1,SIG2,SRFNDX,
   ISTA,MTYPE,NMEAS,ISAT,PRETYP,CHANEL,VHFCHN,
                PREPRO,RECNO
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
DAYSTA	R*8	000000	Time of observation pair	GEOSRD DODSRD PCERD SIMRD NONAME	NONAME BIAS DATARD DODSRD GEOSRD NEWARC PCERD RANDOM SIMRD STORE
OBS1	R*8	000008	First observation	GEOSRD DODSRD PCERD SIMRD NONAME PROCES TWOSTA	NONAME DODSRD GEOSRD PCERD PREDCT PROCES RANDOM SIMRD TWOSTA
OBS2	R*8	000010	Second observation	PROCES TWOSTA	NONAME DODSRD GEOSRD PCERD PREDCT PROCES RANDOM SIMRD TWOSTA
SIG1	R*8	000018	Sigma for first observation	GEOSRD DODSRD PCERD SIMRD NONAME	NONAME DODSRD GEOSRD PCERD PREDCT RANDOM SIMRD

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
SIG2	R*8	000020	Sigma for second observation	PROCES TWOSTA	NONAME DODSRD GEOSRD PCERD PREDCT RANDOM SIMRD TWOSTA
SRFNDX	R*8	000028	Tropospheric refraction index	PROCES TWOSTA	NONAME DODSRD GEOSRD PCERD PREDCT PROCES RANDOM SIMRD TWOSTA
ISTA	I*4	000030	Station index	PROCES TWOSTA	NONAME BIAS DODSRD GEOSRD INOUPT PCERD RANDOM SIMRD
MTYPE	I*2	000034	Measurement type	PROCES TWOSTA	MAIN NONAME BIAS DODSRD GEOSRD INOUPT PCERD PREDCT PROCES RANDOM SIMRD TWOSTA

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
NMEAS	I*2	000036	Number of measurements (1 or 2)	PROCES TWOSTA	NONAME DODSRD GEOSRD PCERD PREDCT RANDOM SIMRD
ISAT	I*2	000038	Satellite index	PROCES TWOSTA	NONAME DODSRD INOUPT OBSDOT PCERD PREDCT PROCES RANDOM SIMRD TWOSTA
PRETYP	I*2	00003A	Preprocessing indicators	PROCES TWOSTA	DODSRD GEOSRD PCERD PREDCT PROCES RANDOM SIMRD TWOSTA
CHANEL	I*2	00003C	Transponder channel	PROCES TWOSTA	NONAME BIAS DODSRD GEOSRD PCERD PROCES RANDOM SIMRD TWOSTA
VHFCHN	L*1	00003E	Switch to indicate VHF transponder	PROCES TWOSTA	DODSRD GEOSRD PCERD PROCES RANDOM SIMRD

/PREBLK/(Cont.)

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
PREPRO	L*1	00003F	Preprocessing switch	PROCES TWOSTA	NONAME DODSRD GEOSRD PCERD PREDCT PROCES RANDOM SIMRD TWOSTA
RECNO	I*4	000040	Measurement record number	PROCES TWOSTA	NONAME DODSRD GEOSRD INOUPT PCERD RANDOM SIMRD

/PRIORI/

```
COMMON/PRIORI/ELEMIN(6,2),VARCOV(6,6,2),TITLE(60),
  DRAGSG(2,3),DRAGO(2,3),CD(2,3)
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
ELEMIN (6,2)	R*8	000000	A priori epoch elements.	MAIN ORBIT	MAIN NONAME BMTWRT DATARD ESTIM INOUPT ORBIT STORE
VARCOV (6,6,2)	R*4	000060	A priori variance/ covariance matrix of the epoch elements.	INOUPT NEWARC	DATARD ESTIM INOUPT NEWARC STORE
TITLE (60)	R*8	000180	Title of arc.	MAIN	MAIN NONAME DATARD INOUPT STORE TYPORB
DRAGSG (2)	R*8	000270	A priori drag sigmas.	INOUPT	NONAME ARCPAR DATARD INOUPT NEWARC STORE
DRGDSG (2)	R*8	000280	A priori drag rate sigmas.	INOUPT	NONAME ARCPAR DATARD INOUPT NEWARC STORE

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
EMISSG (2)	R*8	000290	A priori solar radiation pressure sigmas.	INOUPT	NONAME ARCHAR DATARD INOUPT NEWARC STORE
DRAG0 (2)	R*8	0002A0	A priori drag coefficients, C_D.	INOUPT NEWARC	NONAME DATARD INOUPT NEWARC STORE
DRGD0 (2)	R*8	0002B0	A priori drag rate coefficients, C_D.	INOUPT NEWARC	NONAME DATARD INOUPT NEWARC STORE
EMISS0 (2)	R*8	0002C0	A priori solar radiation pressure coefficients, C_R.	INOUPT NEWARC	NONAME DATARD INOUPT NEWARC STORE
CD (2)	R*8	0002D0	Adjusted drag coefficients C_D.	NONAME INOUPT NEWARC UPDATE	NONAME ARCPAR BMTWRT DATARD INOUPT NEWARC ORB1 ORBIT STORE TYPORB UPDATE

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
CDD (2)	R*8	0002E0	Adjusted drag rate coefficients, C_D.	NONAME INOUPT NEWARC UPDATE	NONAME ARCPAR BMTWRT DATARD INOUPT NEWARC ORB1 ORBIT STORE TYPORB UPDATE
EMISS (2)	R*8	0002F0	Adjusted solar radiation pressure coefficients, C_R.	NONAME INOUPT NEWARC UPDATE	NONAME ARCPAR BMTWRT DATARD INOUPT NEWARC ORB1 ORBIT STORE TYPORB UPDATE

/SIGBLK/

```
COMMON/SIGBLK/SIGSTD(30),SGPRNT(30),IARRAY(4)
```

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
SIGSTD (30)	R*4	000000	Default measurements sigmas.	BLOCK DATA	DODSRD GEOSRD INOUPT PCERD
SGPRNT (30)	R*4	000078	Measurement type sigmas to be printed.	INOUPT	INOUPT DODSRD
IARRAY (4)	I*4	0000F0	Input data tape numbers.	INOUPT	GEOSRD INOUPT PCERD SIMRD

/SRFBLK/ (Version 1)

COMMON/SRFBLK/DUMMY(5400),BESINO(675),BETYPE(675)

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
Single Words 1-5400	R*4	000000	Scratch	-	GEOSRD
BESTNO (675)	I*2	005460	Station number for which electronic biases will be extracted	INOUPT	INOUPT
BETYPE (675)	I*2	0059A6	Measurement types for electronic bias extraction	INOUPT	INOUPT

/SRFBLK/ (Version 2)

COMMON/SRFBLK/PHI(675),XLAM(675),DP(675),SD(675),
SSD(675),NP(675),NL(675),NSD

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
PHI (675)	R*8	000000	Latitudes of centers of master surface density blocks	INOUPT	INOUPT PDEN1
XLAM (675)	R*8	001518	Longitude of centers of master surface density blocks	INOUPT	INOUPT PDEN1
DP (675)	R*4	002A30	Surface density block latitude increments	INOUPT	INOUPT PDEN1
DL (675)	R*4	0034BC	Surface density block longitude increments	INOUPT	INOUPT PDEN1
SD (675)	R*4	003F48	Surface density values	INOUPT	INOUPT PDEN1
SSD (675)	R*4	0049D4	Surface density *a priori* sigmas	INOUPT	INOUPT PDEN1
NP (675)	I*2	005460	Numbers of latitude divisions for surface density master blocks	INOUPT	INOUPT PDEN1
NL (675)	I*2	0059A6	Numbers of longitude divisions for surface density master blocks	INOUPT	INOUPT PDEN1
NSD	I*4	005EEC	Number of surface density master blocks	INOUPT	INOUPT PDEN1

/STANUM/

COMMON/STANUM/NAME,STANOS,NOSTOR

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
NAME	R*8	000000	Station names.	BLOCK DATA	DODSRD STAINP
STANOS	I*2	0008C0	Station numbers.	BLOCK DATA	DODSRD GEOSRD SIMRD STAINP
NOSTOR	I*4	000AF0	Number of stored stations.	BLOCK DATA	DODSRD GEOSRD SIMRD STAINP

/STAPOS/

COMMON/STAPOS/LAT(280),LON(280),HT(280)

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
LAT (280)	R*8	000000	Station geodetic latitude in radians.	BLOCK DATA	STAINP
LON (280)	R*8	0008C0	Station geodetic longitude in radians.	BLOCK DATA	STAINP
HT (280)	R*8	001180	Station height in radians.	BLOCK DATA	STAINP

/TPEBLK/

```
COMMON/TPEBLK/INTP,OUTP,DATP,XYZTP,KEPTAP,RVTP,
  PLOTP,IOBS,SCRA,SCRC,FLTP,GRDTP
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
INTP	I*4	000000	Card input unit-5.	BLOCK DATA	MAIN ADFLUX ALIST GEOSRD INOUPT STAINP
OUTP	I*4	000004	Printer output unit-6.	BLOCK DATA	NONAME ALIST BMTWRT COMADJ COMPAR CORREL INOUPT PDEN PDEN1 SUMMRY TYPORB UPDATE
DATP	I*4	000008	Direct access data storage disk unit-12.	BLOCK DATA	not used
XYZTP	I*4	00000C	Printer output unit for XYZ ephemeris 6, 8 or 9.	BLOCK DATA INOUPT	NONAME DATARD NEWARC STORE
KEPTAP	I*4	000010	Printer output unit for Kepler ephemeris 6,8, or 9.	BLOCK DATA INOUPT	NONAME DATARD INOUPT NEWARC STORE
RVTP	I*4	000014	RV tape unit.	BLOCK DATA INOUPT	NONAME DATARD INOUPT NEWARC STORE

/TPEBLK/ (Cont.)

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
PLOTP	I*4	000018	Binary residual tape unit.	BLOCK DATA	NONAME
IOBS	I*4	00001C	Data selection indicator.	INOUPT	DODSRD GEOSRD INOUPT NEWARC
SCRA	I*4	000020	Scratch disk for a priori information-14.	BLOCK DATA	NONAME ARCPAR DATARD
SCRC	I*4	000024	Scratch disk for normal matrix-16.	BLOCK DATA	NONAME ARCPAR COMPAR DATARD INOUPT STORE
FLTP	I*4	000028	Scratch disk for flux data-13.	BLOCK DATA	NONAME ADFLUX
GRDTP	I*4	00002C	Groundtrack tape unit-18.	BLOCK DATA	NONAME

/TRUPOL/

COMMON/TRUPOL/TRUE

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
TRUE	R*8	000000	Coordinates of tracking station corrected from the position of the true pole.	TRUEP	GRHRAN TRUEP

/XYZ/

COMMON/XYZ/ELEM(6),R,RSQ,ISAT,IFORCE(2)

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
ELEM (6)	R*8	000000	Satellite Cartesian elements.	F	NONAME AVGPOT DENSTY DRAG EGRAV F GEOIDH RESPAR SUNGRV SURDEN TIDAL VEVAL
R	R*8	000030	Satellite earth centered position radius.	EGRAV	NONAME AVGPOT DENSTY DRAG EGRAV GEOIDH RESPAR SURDEN TIDAL VEVAL
RSQ	R*8	000038	R*R	EGRAV	NONAME DRAG EGRAV RESPAR SUNGRV SURDEN VEVAL
ISAT	I*4	000040	Index of satellite orbit being integrated.	ORBIT	DRAG F ORBIT RESPAR SURDEN VEVAL
IFORCE (2)	I*4	000044	Displacements in partial array of drag and solrad partials.	NONAME	NONAME DRAG F

/XYZOUT/

COMMON/XYZOUT/XYZEND(6,2),DRGPAR(6,2)

Variable	Type	Hex Location	Description	Program Where Defined	Programs Where Used
XYZEND (6,2)	R*8	000000	Cartesian satellite coordinates corresponding to time of integration call.	ORBIT	NONAME GRHRAN OBSDOT ORB1 ORBIT PREDCT TYPORB
DRGPAR (6,2)	R*8	000060	Cartesian drag partials corresponding to time of integration call.	ORBIT	GRHRAN OBSDOT ORBIT PREDCT TYPORB

/VMAT/

Common block VMAT is all scratch and used by subroutines COWELL, INOUPT and VEVAL.

/VRBLOK/ (Version 1)

COMMON/VRBLOK/GPSIG(960)

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
GPSIG (960)	R*4	000000	Sigmas on common adjusted geopotential coefficients.	INOUPT	COMPAR INOUPT STORE

/VRBLOK/ (Version 2)

```
COMMON/VRBLOK/BIASSO(900),BIASSG(900),BSTANO(900)
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
BIASSO (900)	R*4	000000	A priori value of adjusted biases.	INOUPT BIAS	ARCPAR BIAS INOUPT STORE
BIASSG (900)	R*4	000E10	A priori sigmas of adjusted biases.	INOUPT BIAS	ARCPAR BIAS INOUPT STORE
BSTANO	I*2	001C20	Station numbers for adjusted biases.	INOUPT BIAS	ARCPAR BIAS INOUPT STORE

/VRBLOK/ (Version 3)

COMMON/VRBLOK/A1,IBUF,AORN

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Defined
A1(66)	R*8	000000	Dummy	---	---
IBUF (1980)	I*4	000210	Scratch	DATARD	DATARD
AORN (69)	R*8	002100	Dummy	---	---

/VRBLOK/ (Version 4)

```
COMMON/VRBLOK/XYSQ,COSLAM(31),SINLAM(31,PR,PPSI,
  PLAMDA,P(33,30),AORN(30),TPSIM(39)
```

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
XYSQ	R*8	000000	X**2+Y**2 where X and Y are the earth centered X&Y coordinate of the satellite.	EGRAV	CBROWN EGRAV INDENT NEWARC RESPAR VEVAL
COSLAM (31)	R*8	000008	The cosines of m*longitude of the satellite for m=0,30.	CBROWN EGRAV	CBROWN EGRAV GEOIDH INDENT NEWARC RESPAR VEVAL
SIMLAM (31)	R*8	000100	The sines of m*longitude of the satellite for m=0,30.	EGRAV	CBROWN EGRAV GEOIDH INDENT NEWARC RESPAR VEVAL
PR	R*8	0001F8	Partial of the earth potential with respect to the radial direction	EGRAV	CBROWN EGRAV INDENT NEWARC VEVAL
PPSI	R*8	000200	Partial of the earth potential with respect to geocentric latitude.	EGRAV	CBROWN EGRAV INDENT NEWARC VEVAL
PLAMDA	R*8	000208	Partial of the earth potential with respect to east longitude.	EGRAV	CBROWN EGRAV INDENT NEWARC VEVAL

Variable	Type	Hex Location	Description	Programs Where Defined	Programs Where Used
P(33,30)	R*8	000210	The Legendre and associated Legendre polynomials of the spherical harmonic expansion. P(m+1,n) is the polynomial associated with degree n and order m.	EGRAV	AVGPOT BMTWRT EGRAV GEOIDH INDENT NEWARC RESPAR VEVAL
AORN(30)	R*8	002100	$AORN(n)=\left(\frac{a_e}{R}\right)^n$ where R is the distance from the earth's center to the point where the potential is being evaluated. A_e is the semi-major axis of the earth, and n is the degree.	EGRAV	AVGPOT EGRAV GEOIDH NEWARC RESPAR VEVAL
TPSIM (30)	R*8	0021F0	The tangents of m times the geocentric latitude for m=0,30.	EGRAV	EGRAV NEWARC RESPAR VEVAL