NASA CR- 139153

VOLUME IV

GEODYN SYSTEM

SUPPORT PROGRAMS

Contract Number: NAS 5-11735 - MOD 65

PCN 550-W-72416

Prepared By:

N. Eileen Mullins
Wolf Research and Development Corporation
6801 Kenilworth Avenue
Riverdale, Maryland

Prepared For:

Goddard Space Flight Center
Greenbelt, Maryland

30 September 1974

(NASA-CR-139153) GEODYN SYSTEM SUPPORT PROGRAM, VOLUME 4 (Wolf Research and Development Corp.) 362 p HC $10.00 CSCL 08E

N75-11418

Unclas

G3/43 03465

TABLE OF CONTENTS

TABLE OF CONTENTS (Cont.)

TABLE OF CONTENTS (Cont.)

INTRODUCTION

The GEODYN Orbit Determination and Geodetic Parameter Estimation System consists of a set of computer programs designed to determine and analyze definitive satellite orbits and their associated geodetic and measurement parameters. This manual describes the Support Programs used by the GEODYN System. The mathematics and programming descriptions are detailed in the first section. The second section contains the operational procedures of each program.

GEODYN ancillary analysis programs may be grouped into three different categories:

1. Orbit Comparison - DELTA

2. Data Analysis using Reference Orbits - GEORGE

3. Pass Geometry Computations - GROUNDTRACK

All of the above three programs use one or more tapes written by the GEODYN program in either a data reduction or orbit generator run. Although it is not necessary, these programs are generally run immediately following the associated GEODYN run, thus minimizing tape handling problems. In addition all three programs use the WRDC PLOT PACKAGE and can produce a graphical depiction of their results both on printer plots and on SC4020 microfilm or hardcopy plots.

In addition to the above analysis programs, the GEODYN System contains five data management routines:

1. Sort-merge program in DODS format - DODS SORT-MERGE

2. Sort-merge program in GEOS format - GEOS SORT-MERGE

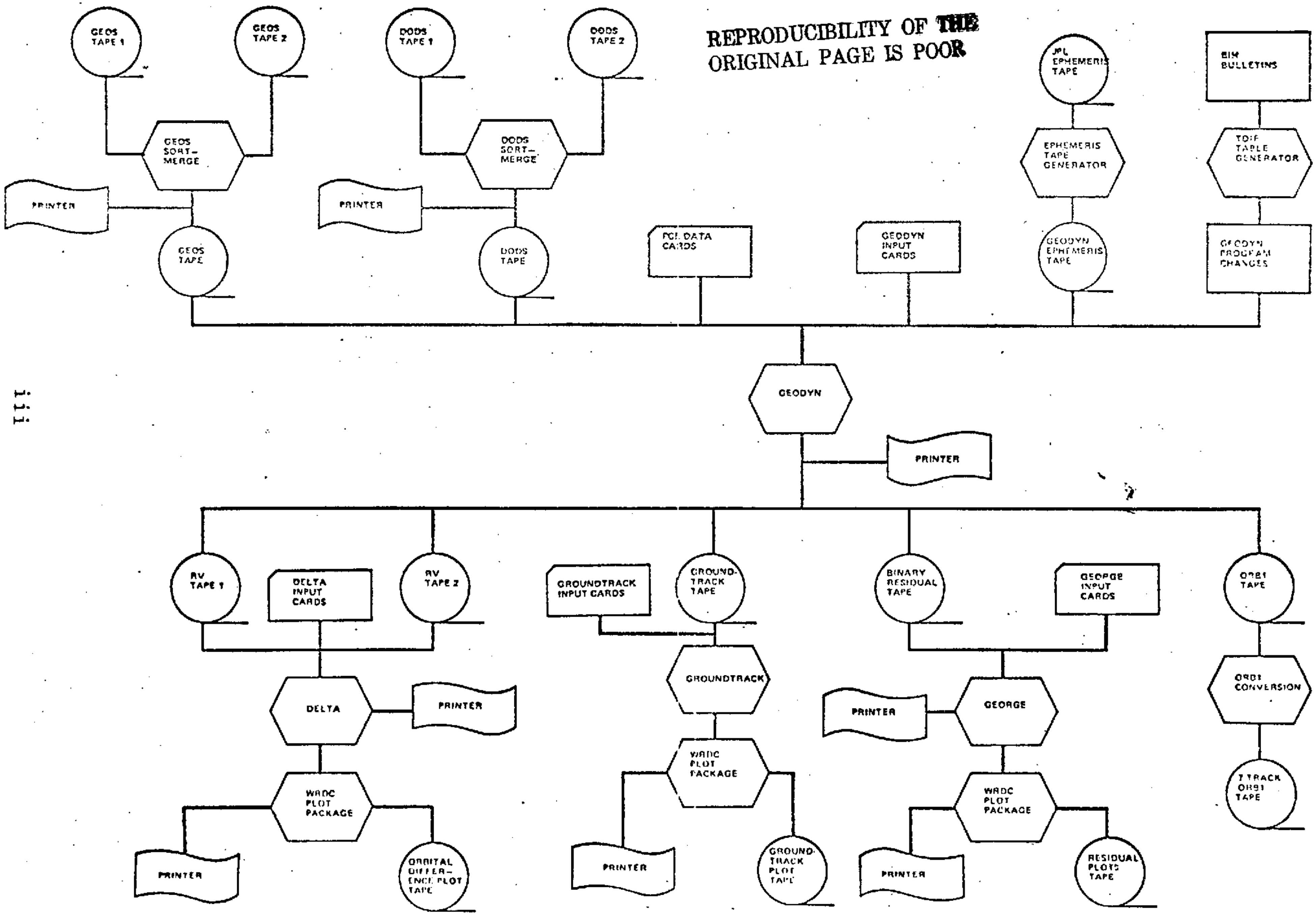
3. EPHEMERIS TAPE GENERATOR

4. 9-7 Track conversion - ORB1 CONVERSION

5. TDIF TABLE GENERATOR

The flowchart on the following page depicts the structure of the entire GEODYN System.

GEODYN SYSTEM FLOWCHART



SECTION 1.0

MATHEMATICS AND PROGRAMMING DESCRIPTIONS OF THE GEODYN SUPPORT PROGRAMS

1.1 GEODYN ANALYSES AND GRAPHICS SUPPORT PROGRAMS

There exist three ancillary programs, DELTA, GEORGE, and GROUNDTRACK, which are used with the GEODYN program in the analysis of GEODYN determined trajectories and residuals. These programs are entirely independent of the GEODYN program. All three use as input GEODYN generated data files, thus, usually they are run as a second job step after a GEODYN run.

DELTA is used to print and/or plot along-track, cross-track and radial differences between two trajectories. It differences orbits of the same satellite for the same time period but generated with different values for certain parameters or reduced over different data spans.

GEORGE performs a regression analysis of the residuals for each pass of data about a trajectory to determine trends in possible timing and measurement biases.

GROUNDTRACK simply plots the groundtrack of the satellite over a particular tracking station or stations to provide geometric insights into data trends.

All three programs will optionally produce printer and/or SC4020 plots to illustrate the computed results. Hence the WOLF PLOT PACKAGE must be included when using these programs.

1.1.1 DELTA

INTRODUCTION

The graphic support program DELTA prints and/or plots trajectory differences. The two trajectories enter the program from two magnetic tapes in either an R-V tape format or ORB1 tape format. If the tapes are in the ORB1 format the subroutine RDORB1 is called to obtain each trajectory point; DELTA itself can read the R-V tapes. The subroutine READER is the driver for the sequence of calls to the Plot Package, which provide the plots of the trajectory differences.

DELTA uses the DSQRT, MOD, and FLOAT system routines and approximately 250K bytes of core. The program will difference 1400 time points of two orbits in less than three minutes of CPU time.

Subroutine and common block cross reference charts appear in this section. The calling routines are at the top of the subroutine chart and the common blocks are listed down the side of the common block chart.

The routines in the Plot Package are all in G and H level FORTRAN with the exception of TIMING which is in IBM 360 Assembly Language. These routines were designed to be efficient on the IBM 360 series machines; no attempt whatever has been made to pursue the myth of compatibility.

The trajectory tapes input to DELTA consist of the satellite positions (X,Y,Z) and velocities $(\dot{X},\dot{Y},\dot{Z})$ in the Cartesian system at given time intervals.

If X_1, Y_1, Z_1 are the Cartesian coordinates of satellite position from tape 1 and X_2, Y_2, Z_2 are the coordinates from tape 2 then the position difference vector is

$$\Delta\overline{P} = (\Delta X = X_2 - X_1, \quad \Delta Y = Y_2 - Y_1, \text{ and } \Delta Z = Z_2 - Z_1).$$

The velocity difference vector $\Delta\overline{V} = (\Delta\dot{X}, \Delta\dot{Y}, \Delta\dot{Z})$ is computed similarly.

These vectors are then resolved into a radial vector, $\underline{H}$, a cross-track vector $\underline{C}$, and an approximation to an along-track vector, $\underline{L}$ (for nearly circular orbits).

First the distance from the geocenter to the satellite, R, is computed where

$$R = \sqrt{X^2 + Y^2 + Z^2}$$

and the square of the magnitude of the velocity vector $(\overline{V})$,

$$V^2 = \dot{X}^2 + \dot{Y}^2 + \dot{Z}^2.$$

Thus the unit vector, $\hat{U}$, in the radial direction is

$$\hat{U} = \left(\frac{X}{R}, \frac{Y}{R}, \frac{Z}{R} \right)$$

DELTA

Then to calculate the magnitude of the vector in our along-track direction (normal to $\hat{U}$ in the orbit plane), A, we must compute $\hat{U} \cdot \bar{V}$ because

$$A = \sqrt{V^2 - (\hat{U} \cdot \bar{V})^2}$$

Now we compute the unit vectors in our along-track direction $\bar{A} = (a_1, a_2, a_2)$ where

$$a_1 = \left(\dot{X}_2 - (\hat{U} \cdot \bar{V}) \left(\frac{X}{R} \right) \right) \Big/ A$$

$$a_2 = \left(\dot{Y}_2 - (\hat{U} \cdot \bar{V}) \left(\frac{Y}{R} \right) \right) \Big/ A$$

$$a_3 = \left(Z_2 - (\hat{U} \quad \bar{V}) \left(\frac{Z}{R} \right) \right) \Big/ A$$

and the cross-track direction $\bar{C} = (C_1, C_2, C_2)$ where

$$\bar{C} = \bar{A} \times \hat{U}$$

or

$$C_1 = (a_2)\left(\frac{Z}{R}\right) - \left(\frac{Y}{R}\right)(a_3)$$

$$C_2 = (a_3)\left(\frac{X}{R}\right) - \left(\frac{Z}{R}\right)(a_1)$$

$$C_3 = (a_1)\left(\frac{Y}{R}\right) - \left(\frac{X}{R}\right)(a_2)$$

Finally we compute the position differences in radial, H_p, cross-track C_p, and approximation to along-track, L_p;

$$H_p = \hat{U} \cdot \Delta\overline{p}$$

$$C_p = \overline{C} \cdot \Delta\overline{p}$$

$$L_p = \overline{A} \cdot \Delta\overline{p}$$

and the velocity differences in the radial, H_v, cross-track, C_v, and approximation to along-track, L_p:

$$H_v = \hat{U} \cdot \Delta\overline{V}$$

$$C_v = \overline{C} \cdot \Delta\overline{V}$$

$$L_v = \overline{A} \cdot \Delta\overline{V}$$

SUBROUTINE CROSS REFERENCE CHART

CALLING ROUTINES

CALLED ROUTINES	MAIN	ADTIME	RDORB1
ADDYMD		●	
ADTIME	●		●
RDORB1	●		
READER	●		

COMMON BLOCK CROSS REFERENCE CHART

ROUTINES

COMMON BLOCK	MAIN	READER
PLOTTP	●	●

MAIN-DELTA

DESCRIPTION

The main routine DELTA reads data from two RV tapes or receives data from the routine RDORB1, calculates and prints radial, cross-track, and along-track differences, and calls READER to make plots if requested.

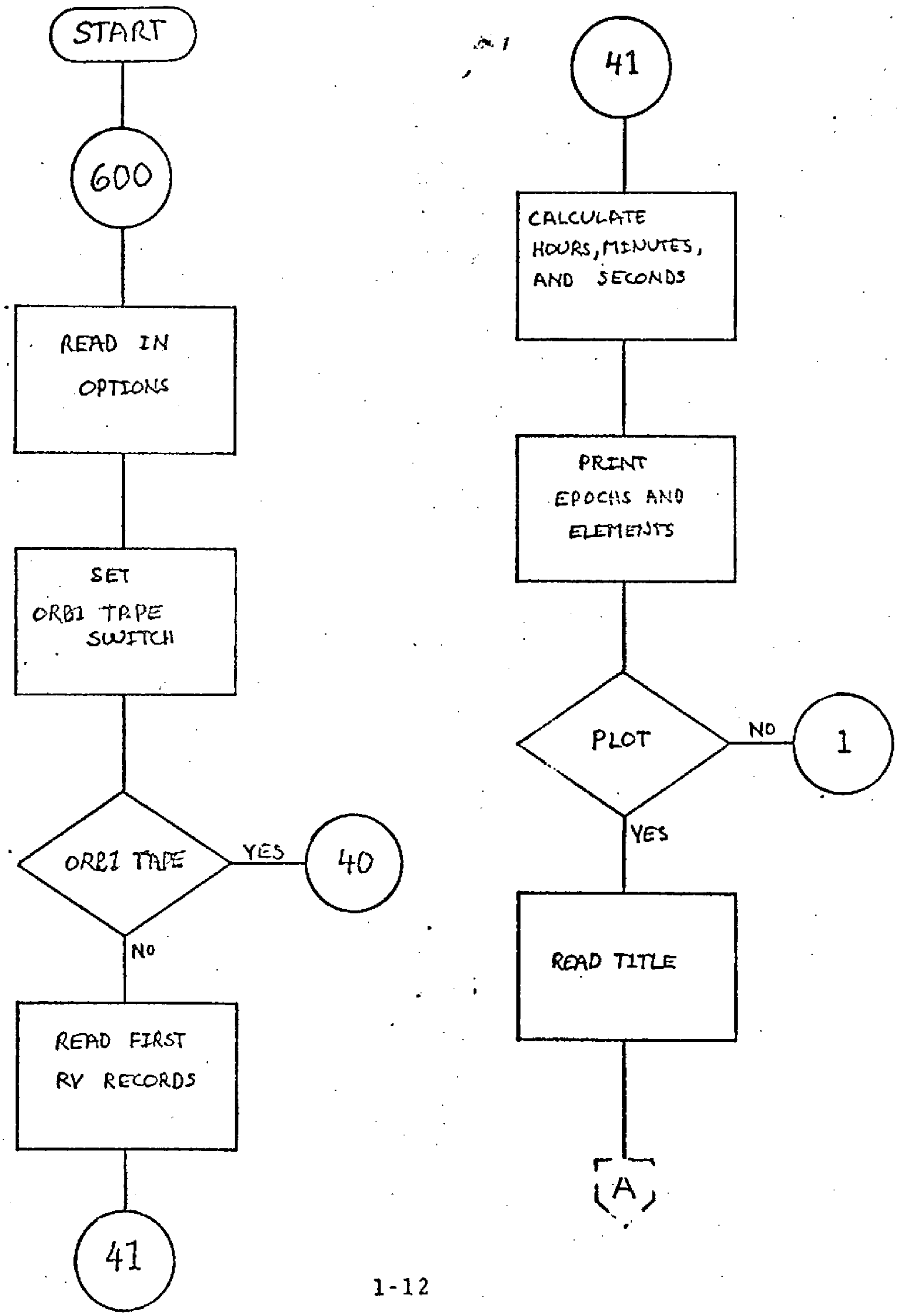
NAME	MAIN - DELTA
PURPOSE	DIFFERENCES GEODYN GENERATED RV OR ORBI TAPES
SUBROUTINES USED	RDORBI READER
COMMON BLOCK	PLOTTP
INPUT FILES	INPT - DELTA INPUT CARDS RVTAP1 - RV TAPE1 RVTAP2 - RV TAPE2
OUTPUT FILE	OUTP - PRINTER
RESTRICTIONS	NONE
REFERENCES	NONE

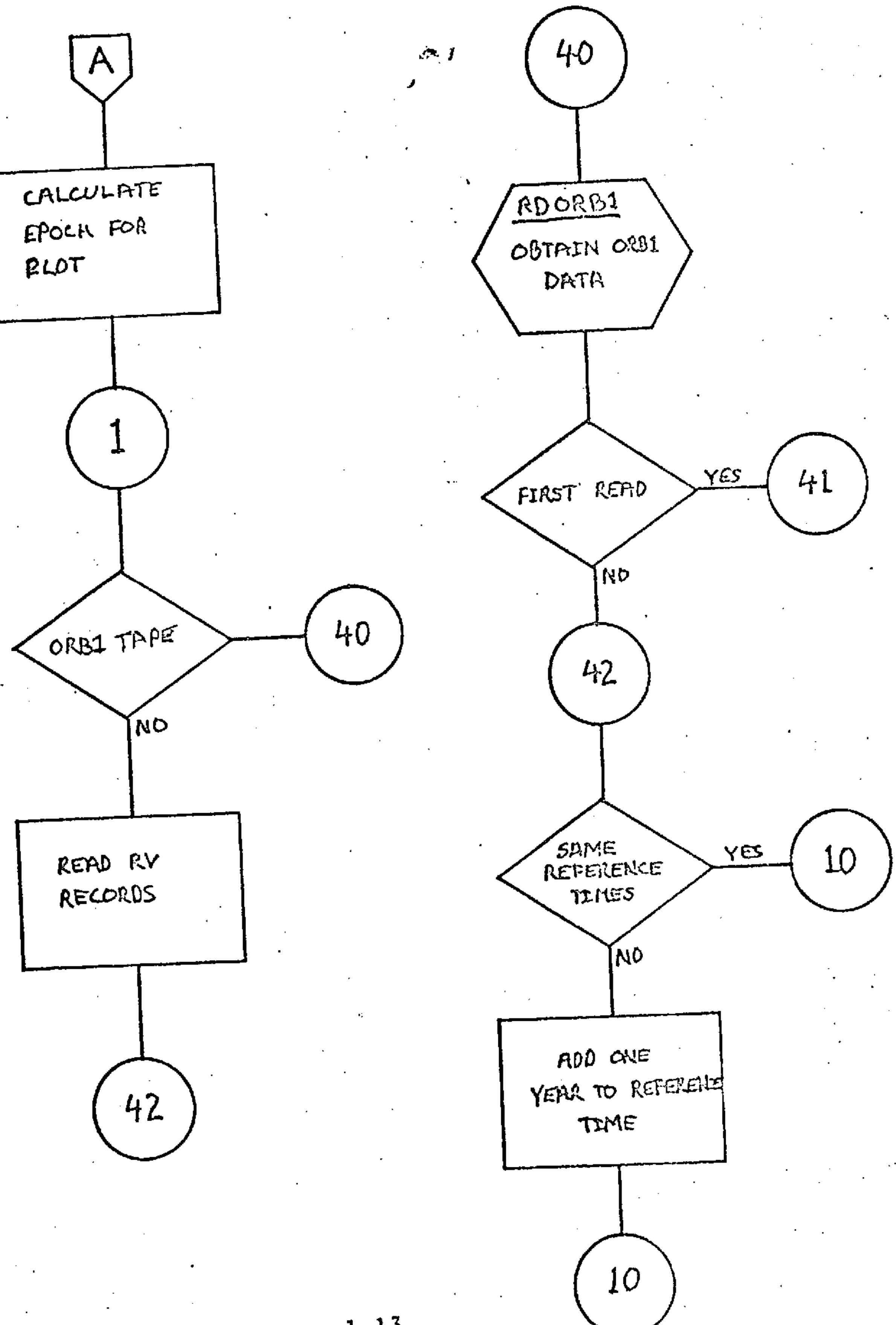
```
      DOUBLE PRECISION XYZEN1(6),XYZEN2(6),DSQRT,DX(3),DXDOT(3),U(3),
     *                R2,R,V2,UDOTV,VDOTV2,AT(3),C(3),DR(3),DV(3),VSQ
      REAL*8 DAYS1,DAYS2,YMD1,YMD2,HMS1,HMS2,EOF,SUMPOS,SUMVEL,DAYR(2),
     .       SDR(3),SDV(3),XIND,TITLE
      REAL NTRVL
      DIMENSION DDR(3)
      LOGICAL IPLOT,ISW1,ISW2
      DOUBLE PRECISION DELTAT
      LOGICAL ORBI,LASTSW
      INTEGER RVTAP1,RVTAP2,INTF,OUTF
      DATA INTF,OUTP,RVTAP1,RVTAP2/5,6,21,22/
      DATA EOF,ISW1,ISW2/0.999D3,2*.FALSE./
      DATA ORBI/.FALSE./
      COMMON/PLOTTP/DAYS(4000),RADL(4000),CTRK(4000),ATRK(4000),
     .              TITLE(21),IEPOCH(2),INDEX,NOPT,SCALE(2),NTRVL
      DATA DAYR,SDR,SDV,XIND/3.66D2,3.65D2,7*0.0D0/
      DATA NUM/0/
  600 INDEX=0
C DETERMINE INPUT TAPE UNIT NUMBERS,PLOTTING OPTIONS AND SCALES,TYPE OF
      READ(INTF,1000) IRV1,IRV2,IPLOT,NOPT,NORBI,NUM1,ILAST,SCALE,NTRVL
      NUM1=MAX0(1,NUM1)
      IF(NOPT.LE.0.OR.NOPT.GE.7) NOPT=7
C SET ORBI SWITCH
      ORBI=NORBI.GT.0
      LASTSW=ILAST.EQ.0
C RESET RV TAPE UNITS IF REQUESTED
      IF(IRV1.GT.0) RVTAP1=IRV1
      IF(IRV2.GT.0) RVTAP2=IRV2
      REWIND RVTAP1
      REWIND RVTAP2
      NORBI=1
      IF(ORBI) GO TO 40
C READ FIRST DATA RECORD
      READ(RVTAP1) DAYS1,IYMD1,IHM1,SEC1,XYZEN1
```

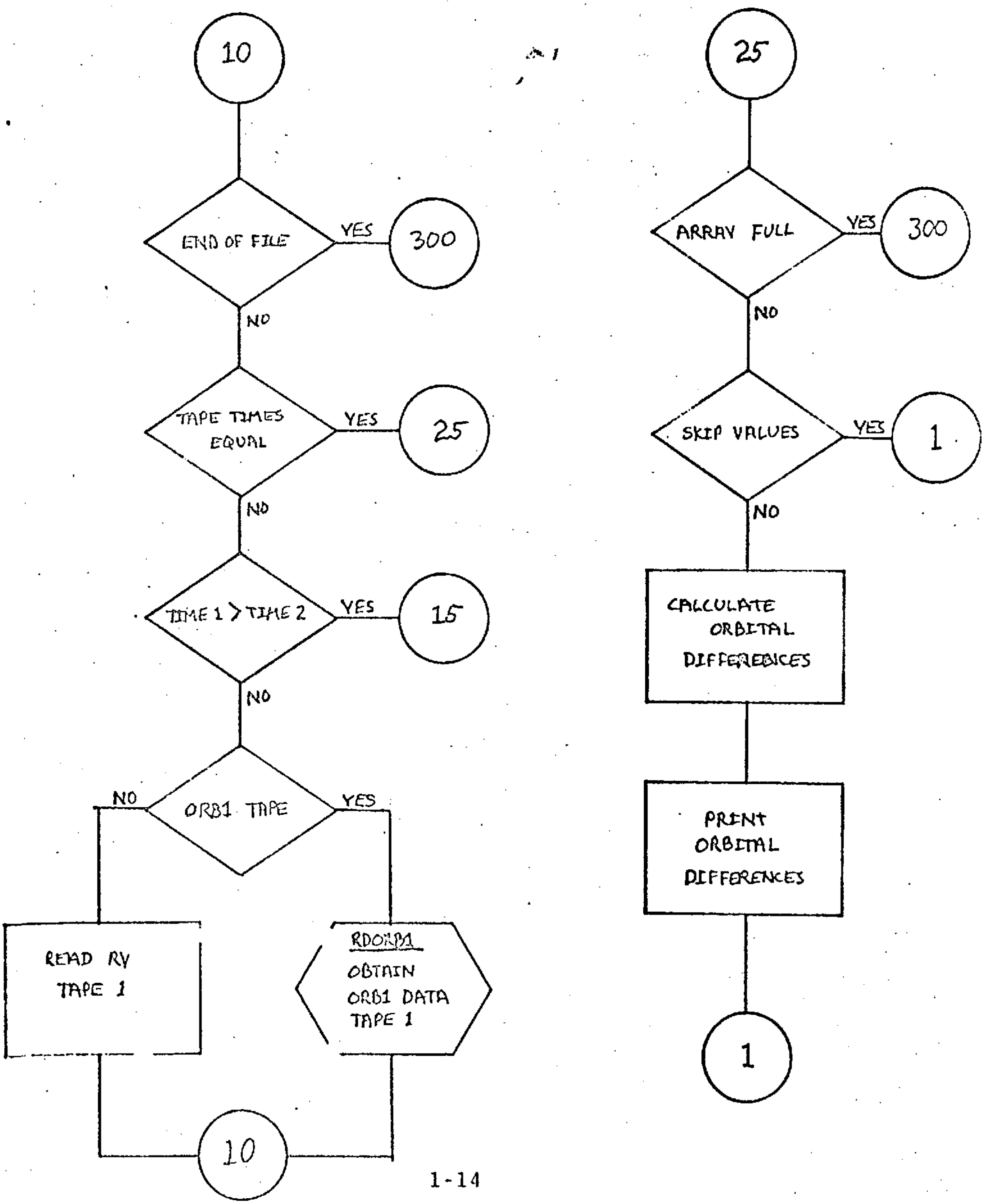
```
      READ(RVTAP2) DAYS2,IYMD2,IHM2,SEC2,XYZEN2
   41 CONTINUE
      IHMS1=(IHM1*100)+IFIX(SEC1)
      IHMS2=(IHM2*100)+IFIX(SEC2)
      N=1
C WRITE EPOCH AND ELEMENTS OF RV TAPE 1
      WRITE(OUTF,30000) N,IYMD1,IHMS1,XYZEN1
      N=2
C WRITE EPOCH AND ELEMENTS OF RV TAPE 2
      WRITE(OUTF,30000) N,IYMD2,IHMS2,XYZEN2
      IF(.NOT.IPLOT) GO TO 2
C READ TITLE IF PLOT IS REQUESTED
      READ(INTF,999) TITLE
C CALCULATE EPOCH IN YEAR, MONTH, DAY, HOUR, MINUTE, SECOND FOR PLOT
      IEPOCH(1)=IYMD2
      IEPOCH(2)=IHMS2
    2 N=0
      WRITE(OUTF,30001)
      IY1=IYMD1/10000
      IY2=IYMD2/10000
      IF(IY1.GE.IY2) GO TO 3
      L1=MINO(MOD(IY1,4),1)+1
      ISW1=.TRUE.
      GO TO 1
C READ HEADER RECORDS ON ORB1 TAPES
   40 READ(RVTAP1) DAYS1
      READ(RVTAP2) DAYS2
C READ ORB1 DATA RECORDS
   50 CALL RDORB1(DAYS1,XYZEN1,RVTAP1,1,IYMD1,IHM1,SEC1)
      CALL RDORB1(DAYS2,XYZEN2,RVTAP2,2,IYMD2,IHM2,SEC2)
      GO TO (41,42),NORB1
    3 IF(IY1.EQ.IY2) GO TO 1
      L1=MINO(MOD(IY2,4),1)+1
      ISW2=.TRUE.
    1 NORB1=2
      IF(ORB1) GO TO 50
C READ RV DATA RECORD
      READ(RVTAP1) DAYS1,IYMD1,IHM1,SEC1,XYZEN1
      READ(RVTAP2) DAYS2,IYMD2,IHM2,SEC2,XYZEN2
   42 CONTINUE
      IF(ISW1) DAYS1=DAYS1+DAYR(L1)
      IF(ISW2) DAYS2=DAYS2+DAYR(L1)
   10 IF(DAYS1.EQ.EOF   .OR.DAYS2.EQ.EOF   ) GO TO 300
      IF(DABS(DAYS2-DAYS1).LT..5D-06) GO TO 25
      IF(DAYS2.LT.DAYS1) GO TO 15
C READ DATA
      IF(.NOT.ORB1) READ(RVTAP1) DAYS1,IYMD1,IHM1,SEC1,XYZEN1
      IF(ORB1) CALL RDORB1(DAYS1,XYZEN1,RVTAP1,1,IYMD1,IHM1,SEC1)
      IF(ISW1) DAYS1=DAYS1+DAYR(L1)
      GO TO 10
C READ DATA
   15 IF(.NOT.ORB1) READ(RVTAP2) DAYS2,IYMD2,IHM2,SEC2,XYZEN2
      IF(ORB1) CALL RDORB1(DAYS2,XYZEN2,RVTAP2,2,IYMD2,IHM2,SEC2)
      IF(ISW2) DAYS2=DAYS2+DAYR(L1)
      GO TO 10
   25 IF(INDEX.GE.4000) GO TO 300
```

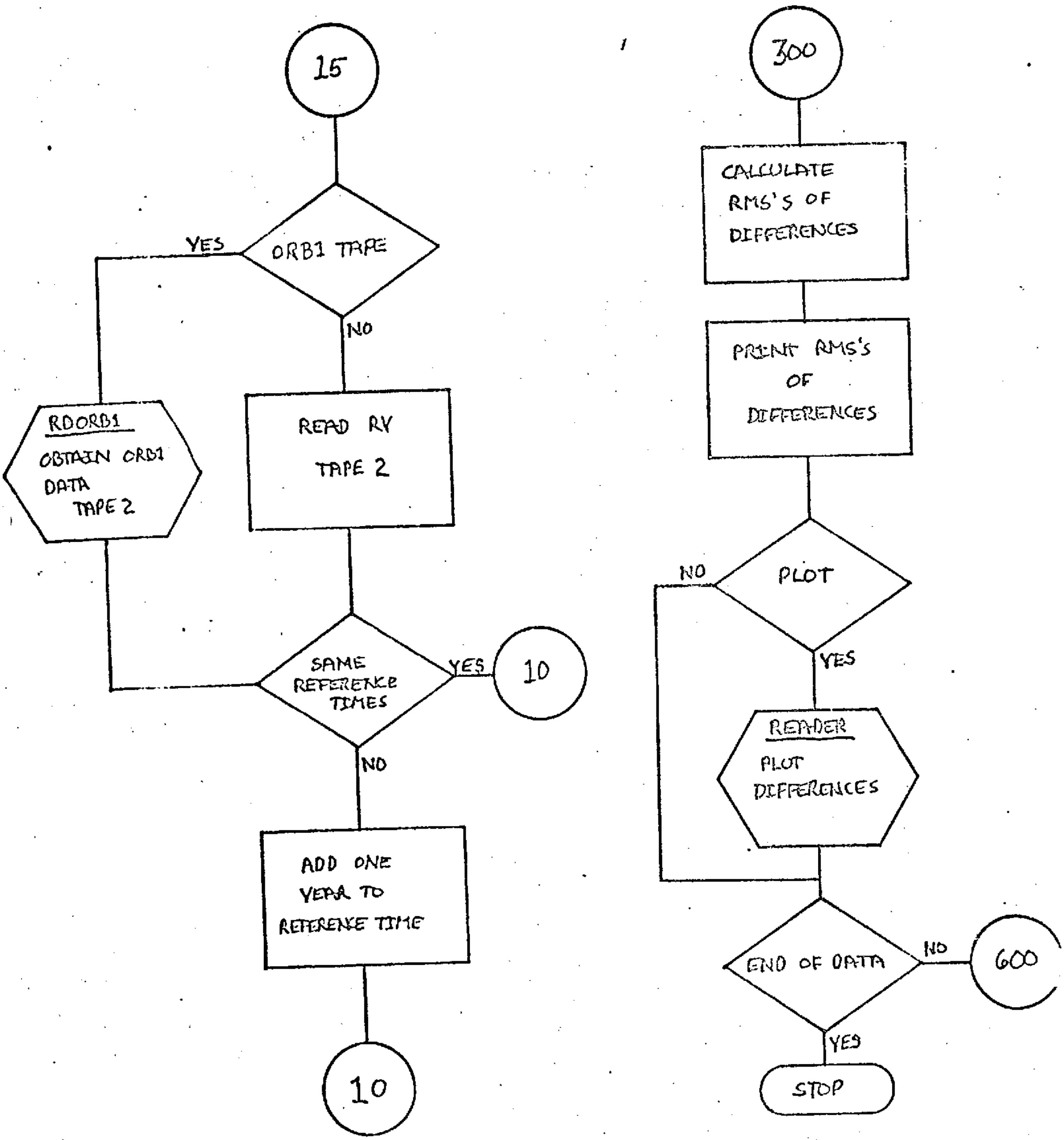
```
      NUM=MOD(NUM,NUM1)
      NUM=NUM+1
      IF(NUM.NE.1) GO TO 1
      INDEX=INDEX+1
C CALCULATE RADIAL,CROSS TRACK,ALONG TRACK DIFFERENCES
      R2=XYZEN2(1)**2+XYZEN2(2)**2+XYZEN2(3)**2
      R=DSQRT(R2)
      V2=XYZEN2(4)**2+XYZEN2(5)**2+XYZEN2(6)**2
      IHMS2=(IHM2*100)+IFIX(SEC2+0.5)
      DO 100 I=1,3
      DX(I)=XYZEN2 (I)-XYZEN1(I)
      DXDOT(I)=XYZEN2(I+3)-XYZEN1(I+3)
  100 U(I)=XYZEN2(I)/R
      UDOTV=XYZEN2(4)*U(1)+XYZEN2(5)*U(2)+XYZEN2(6)*U(3)
      UDOTV2=UDOTV**2
      VSQ=DSQRT(DABS(V2-UDOTV2))
      DO 150 I=1,3
  150 AT(I)=(XYZEN2(I+3)-UDOTV*U(I))/VSQ
      C(1)=AT(2)*U(3)-U(2)*AT(3)
      C(2)=AT(3)*U(1)-U(3)*AT(1)
      C(3)=AT(1)*U(2)-U(1)*AT(2)
      DR(1)=U(1)*DX(1)+U(2)*DX(2)+U(3)*DX(3)
      DR(2)=C(1)*DX(1)+C(2)*DX(2)+C(3)*DX(3)
      DR(3)=AT(1)*DX(1)+AT(2)*DX(2)+AT(3)*DX(3)
      DV(1)=U(1)*DXDOT(1)+U(2)*DXDOT(2)+U(3)*DXDOT(3)
      DV(2)=C(1)*DXDOT(1)+C(2)*DXDOT(2)+C(3)*DXDOT(3)
      DV(3)=AT(1)*DXDOT(1)+AT(2)*DXDOT(2)+AT(3)*DXDOT(3)
      XIND=XIND+1.D0
      DO 175 I=1,3
      SDR(I)=SDR(I)+DR(I)**2
  175 SDV(I)=SDV(I)+DV(I)**2
      DO 225 I=1,3
      DV(I)=DV(I)*100.D0
  225 DXDOT(I)=DXDOT(I)*100.D0
      WRITE(OUTF,3002) IYMD2,IHMS2,DX,DXDOT,DR,DV
      IF(.NOT.IPLOT) GO TO 250
      RADL(INDEX)=DR(1)
      CTRK(INDEX)=DR(2)
      ATRK(INDEX)=DR(3)
      DAYS(INDEX)=DAYS2
      IF(INDEX.EQ.1) DELTAT=DAYS2
      IF(INDEX.EQ.2) DELTAT=DAYS2-DELTAT
  250 N=N+1
      IF(N.LT.50) GO TO 1
      N=0
      WRITE(OUTF,3001)
      GO TO 1
C CALCULATE RMS OF RADIAL,CROSS TRACK,ALONG TRACK DIFFERENCES
  300 SUMPOS=0.D0
      SUMVEL=0.D0
      DO 325 J=1,3
      SUMPOS=SUMPOS+SDR(J)
      SUMVEL=SUMVEL+SDV(J)
      SDR(J)=DSQRT(SDR(J)/XIND)
  325 SDV(J)=DSQRT(SDV(J)/XIND)
      SUMPOS=DSQRT(SUMPOS/XIND)
```

```
      SUMVEL=DSQRT(SUMVEL/XIND)
      WRITE(OUTP,3003) SDR,SUMPOS,SDV,SUMVEL
      REWIND IRV1
      REWIND IRV2
      IF(.NOT.IPLOT) GO TO 500
C MAKE OPTIONAL PLOT AND/OR PLOT TAPE
      CALL READER(DELTAT,LASTSW)
C IF NOT ENDFILE,REREAD DATA
  500 IF(LASTSW) STOP
      NUM=0
      ISW1=.FALSE.
      ISW2=ISW1
      XIND=0.0D0
      DO 510 I=1,2
      SDR(I)=0.0D0
  510 SDV(I)=0.0D0
      CALL RDORB1(DAYS1,XYZEN1,RVTAP1,-1,IYMD1,IHM1,SEC1)
      GO TO 600
  999 FORMAT(7A8)
 1000 FORMAT(2I2,L1,2I1,I2,I1,3F12.5)
 1001 FORMAT(3D10.5/6D10.5)
 3002 FORMAT(1H ,I6,I7,11F10.2,F9.2)
30000 FORMAT(1H1,'EPOCH AND ELEMENTS --- SET',I2/1H0,5X,'YEAR,MONTH'
     *       ',DAY ',I6,4X,'HOUR,MINUTE,SECOND ',I6/
     *         1H0,11X1HX 12X1HY 12X1HZ 10X4HXDOT 9X4HYDOT 9X4HZDOT/
     *         1X,3(10X3H(M)),1X3(8X5H(M/S))//4X,3F13.1,3F13.4)
30001 FORMAT(1H1,5H DATE,3X,2(10X,20HPOSITION DIFFERENCES,10X,
     1         20HVELOCITY DIFFERENCES)/1H ,2X,2HOF,2(20X,8H(METERS),
     2         22X,9H(CM/SEC))/1H ,5H DATA,3X,51(2H- )/
     3         1H ,36X,5HCROSS,5X,5HALONG,15X,
     4         5HCROSS,5X,5HALONG/14H YYMMDD HHMMSS,6X,2HDX,8X,2HDY,8X,
     5         2HDZ,6X,5HDXDOT,5X,5HDYDOT,5X,5HDZDOT,4X,6HRADIAL,4X,
     6        5HTRACK,5X,5HTRACK,5X,6HRADIAL,4X,5HTRACK,5X,5HTRACK/)
 3003 FORMAT(////45X,'RMS OF POSITION AND VELOCITY DIFFERENCES'//15X,
     1   'POSITION DIFFERENCES (METERS)',42X,'VELOCITY DIFFERENCES'
     2   1X,'(CM/SEC)'//4X,'RADIAL',7X,'CROSS TRACK',4X,'ALONG TRACK',
     3   7X,'TOTAL',19X,'RADIAL',7X,'CROSS TRACK',4X,'ALONG TRACK',7X,
     4   'TOTAL'//F10.2,3F15.2,15X,2PF10.2,3F15.2)
      END
```


START
600
READ IN OPTIONS
SET ORB1 TAPE SWITCH
ORB1 TAPE
YES
40
NO
READ FIRST RV RECORDS
41
41
CALCULATE HOURS, MINUTES, AND SECONDS
PRINT EPOCHS AND ELEMENTS
PLOT
NO
1
YES
READ TITLE
A


A
CALCULATE EPOCH FOR PLOT
1
ORB1 TAPE
40
NO
READ RV RECORDS
42
40
RDORB1
OBTAIN ORB1 DATA
FIRST READ
YES
41
NO
42
SAME REFERENCE TIMES
YES
10
NO
ADD ONE YEAR TO REFERENCE TIME
10


10
END OF FILE
YES
300
NO
TAPE TIMES EQUAL
YES
25
NO
TIME 1 > TIME 2
YES
15
NO
ORB1 TAPE
NO
YES
READ RV TAPE 1
RDORB1
OBTAIN ORB1 DATA TAPE 1
10
25
ARRAY FULL
YES
300
NO
SKIP VALUES
YES
1
NO
CALCULATE ORBITAL DIFFERENCES
PRINT ORBITAL DIFFERENCES
1


15
ORB1 TAPE
YES
NO
RDORB1
OBTAIN ORB1 DATA TAPE 2
READ RV TAPE 2
SAME REFERENCE TIMES
YES
10
NO
ADD ONE YEAR TO REFERENCE TIME
10
300
CALCULATE RMS'S OF DIFFERENCES
PRINT RMS'S OF DIFFERENCES
PLOT
NO
YES
READER
PLOT DIFFERENCES
END OF DATA
NO
600
YES
STOP

ADTIME

DESCRIPTION

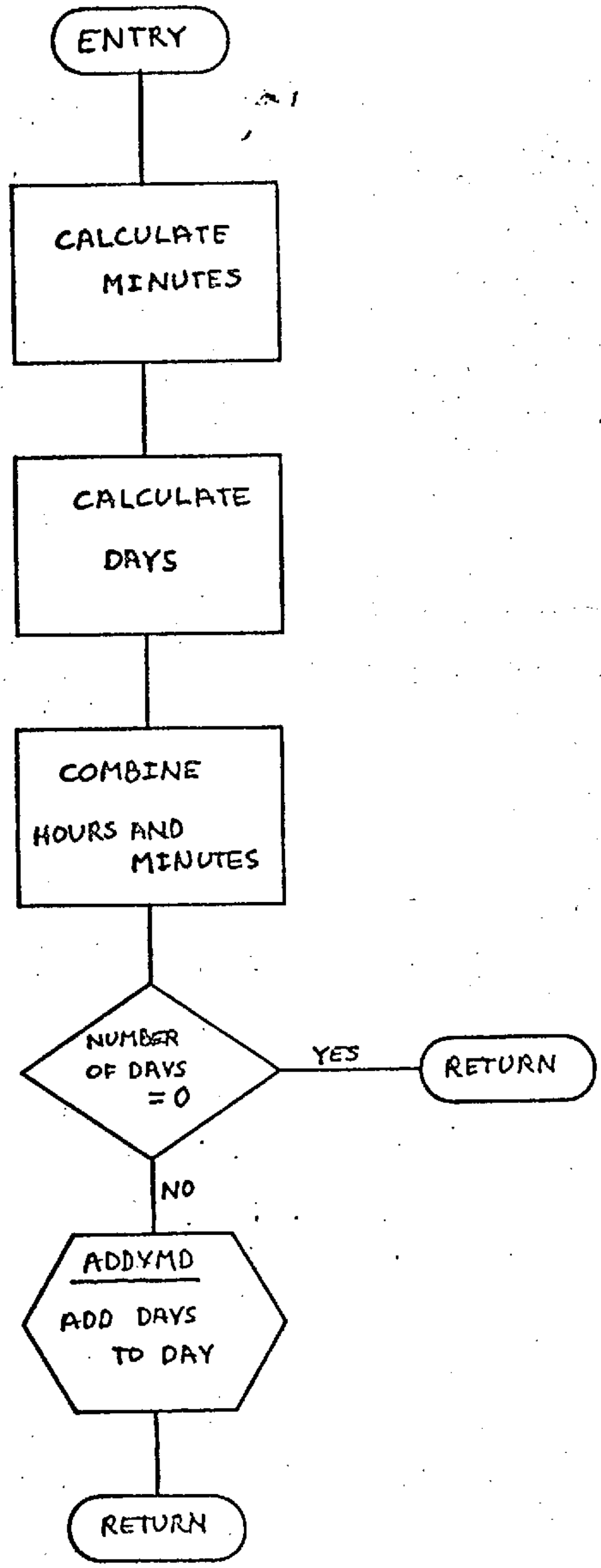
The subroutine ADTIME updates the time of the measurement by the number of seconds between each data point, calling ADDYMD to recompute the date when necessary.

NAME	ADTIME
PURPOSE	CONVERTS HOURS, MINUTES, SECONDS TO DAYS
CALLING SEQUENCE	CALL ADTIME(IYMD,IHM,SEC)

SYMBOL	TYPE	DESCRIPTION
IYMD	I	INPUT - YEAR,MONTH,DAY IN FORM YYMMDD
IHM	I	INPUT - HOUR,MINUTE IN FORM HHMM
SEC	R	INPUT - SECOND

SUBROUTINE USED	ADDYMD
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE ADTIME(IYMD,IHM,SEC)
C CALCULATE MINUTES
 20   IM=SEC/60.
      IF(SEC.LT.0.)IM=IM-1
      SEC=SEC-60.*FLOAT(IM)
      IM=IHM-40*(IHM/100)+IM
C CALCULATE DAYS
      ID=IM/1440
      IF(IM.LT.0)ID=ID-1
C CALCULATE HOURS,MINUTES
      IM=IM-ID*1440
      IHM=IM+40*(IM/60)
C ADD DAYS TO DATE
      IF(ID.NE.0) CALL ADDYMD(IYMD,ID)
      RETURN
      END
```


ENTRY
CALCULATE MINUTES
CALCULATE DAYS
COMBINE HOURS AND MINUTES
NUMBER OF DAYS = 0
YES
RETURN
NO
ADDYMD
ADD DAYS TO DAY
RETURN

RDORB1

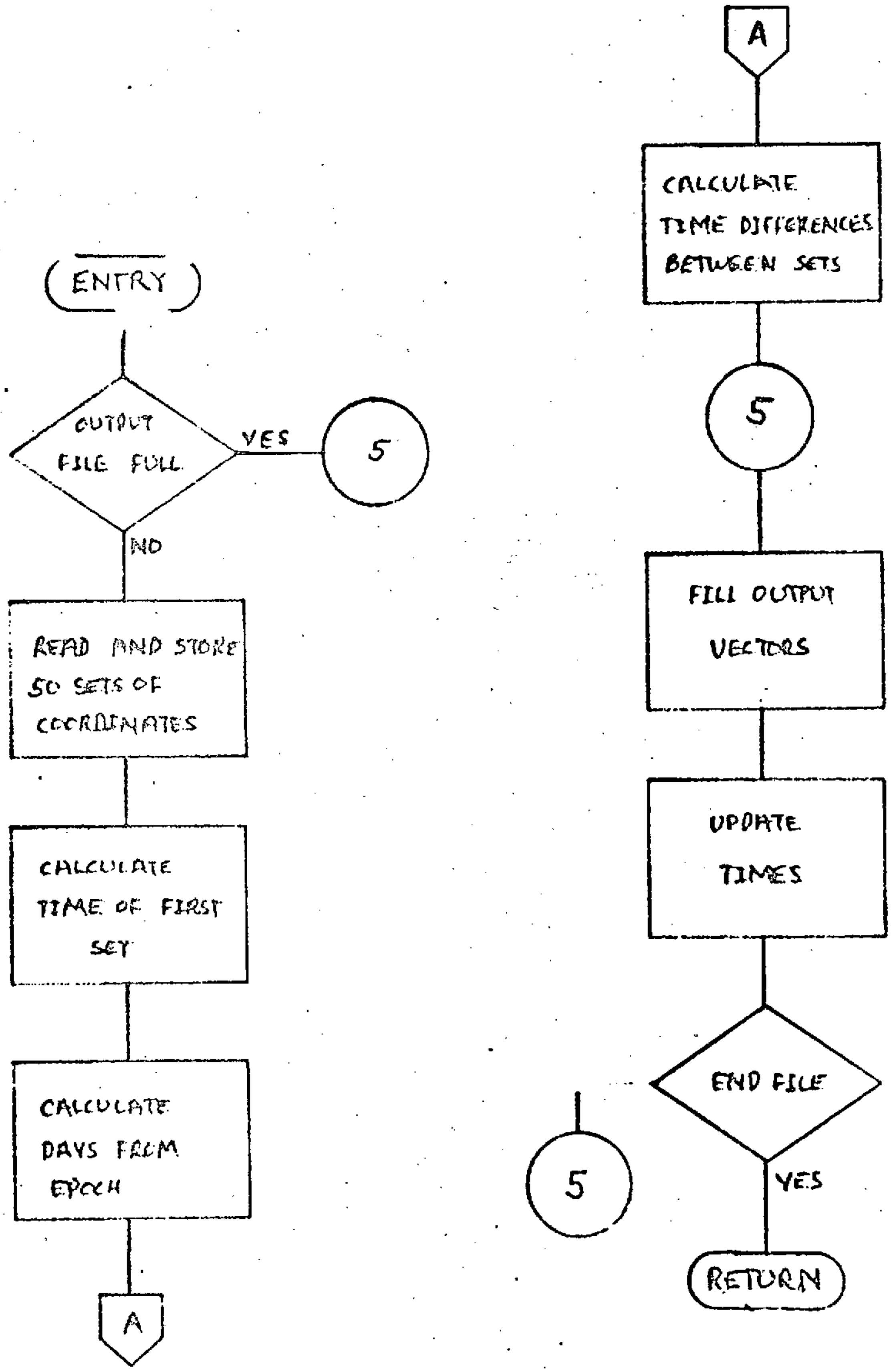
DESCRIPTION

RDORB1 reads a record of 50 data points from one of two ORB1 tapes and stores them, returning one point to the calling program. One point is returned for each subsequent call to RDORB1 for a specific tape until it is necessary to read another record.

NAME		RDORB1
PURPOSE		READS ORB1 TAPES
CALLING SEQUENCE		CALL RDORB1(TIME,XYZ,ORB1,N,IYMDA,IHMA,ASEC)
SYMBOL	TYPE	DESCRIPTION
TIME	DP	OUTPUT - NUMBER OF DAYS FROM EPOCH
XYZ (6)	DP	OUTPUT - COORDINATES OF POSITION AND VELOCITY
ORB1	I	INPUT - UNIT NUMBER OF RV TAPE
N	I	INPUT - RV TAPE INDICATOR (1 OR 2)
IYMDA	I	OUTPUT - YEAR,MONTH,DAY OF COORDINATES
IHMA	I	OUTPUT - HOUR,MINUTE OF COORDINATES
ASEC	R	OUTPUT - SECONDS OF COORDINATES
SUBROUTINE USED		ADTIME
COMMON BLOCKS		NONE
INPUT FILE		ORB1 - ORB1 TAPE
OUTPUT FILES		NONE
RESTRICTIONS		NONE
REFERENCES		NONE

```
      SUBROUTINE RDORB1(TIME,XYZ,ORB1,N,IYMDA,IHMA,ASEC)
      REAL*8 BUF1(5),BUF2(5),BUF(5,2),ELEMS1(6,50),ELEMS2(6,50),
     1       ELEMS(6,50,2),DELTAT(2),DAYS(2),SEC,TIME,XYZ(6),EOF
      INTEGER ORB1
      DIMENSION M(2),NOTIST(2),IYMD(2),IHM(2),SEC(2)
      LOGICAL NOTIST
      EQUIVALENCE (BUF1,BUF),(BUF2,BUF(1,2)),(ELEMS1,ELEMS),
     1            (ELEMS2,ELEMS(1,1,2))
      DATA M/2*50/
      DATA NOTIST/2*.FALSE./
      DATA EOF/99999999.0D3/
      IF(N.LT.0) GO TO 25
C TEST IF ARRAY IS EMPTY
      IF(M(N).LT.50) GO TO 5
C READ COORDINATES INTO AN ARRAY
      IF(N.EQ.1) READ(ORB1,END=20) BUF1,ELEMS1
      IF(N.EQ.2) READ(ORB1,END=20) BUF2,ELEMS2
```

```
      M(N)=0
C CALCULATE TIME OF FIRST SET OF COORDINATES
      IF(BUF(1,N).EQ.EOF*1.0D-3) GO TO 20
      IYMD(N)=BUF(1,N)
      SEC(N)=BUF(3,N)-BUF(4,N)
      IH=IDINT(SEC(N)/3.6D3)*100
      IM=IDINT((SEC(N)-DFLOAT(IH/100)*3.6D3)/6.0D1)
      SEC(N)=SEC(N)-DFLOAT(IM)*6.0D1-DFLOAT(IH/100)*3.6D3
      IHM(N)=IH+IM
      IF(NOT1ST(N)) GO TO 5
C CALCULATE DAYS FROM EPOCH
      DAYS(N)=BUF(2,N)+BUF(3,N)/8.64D4
C CALCULATE TIME DIFFERENCE BETWEEN COORDINATE SETS
      DELTAT(N)=BUF(4,N)/8.64D4
      NOT1ST(N)=.TRUE.
    5 M(N)=M(N)+1
      J=M(N)
C FILL OUTPUT VECTORS
      DO 10 I=1,6
   10 XYZ(I)=ELEMS(I,J,N)*1.0D3
      TIME=DAYS(N)
C UPDATE TIMES
      DAYS(N)=DAYS(N)+DELTAT(N)
      SEC(N)=SEC(N)+BUF(4,N)
      CALL ACTIME(IYMD(N),IHM(N),SEC(N))
      IYMDA=IYMD(N)
      IHMA=IHM(N)
      ASEC=SEC(N)
C TEST FOR ENDFILE
      IF(XYZ(1).EQ.EOF) TIME=999.0D0
      RETURN
C RETURN END OF FILE
   20 TIME=999.0D0
      RETURN
   25 DO 30 I=1,2
      M(I)=50
   30 NOT1ST(I)=.FALSE.
      RETURN
      END
```


ENTRY
OUTPUT FILE FULL
YES
NO
5
READ AND STORE 50 SETS OF COORDINATES
CALCULATE TIME OF FIRST SET
CALCULATE DAYS FROM EPOCH
A
CALCULATE TIME DIFFERENCES BETWEEN SETS
FILL OUTPUT VECTORS
UPDATE TIMES
END FILE
RETURN

READER

DESCRIPTION

The subroutine READER controls the calls to the WRDC Plot Package routines to generate the printer plots and/or plot tape for the orbital differences computed in DELTA.

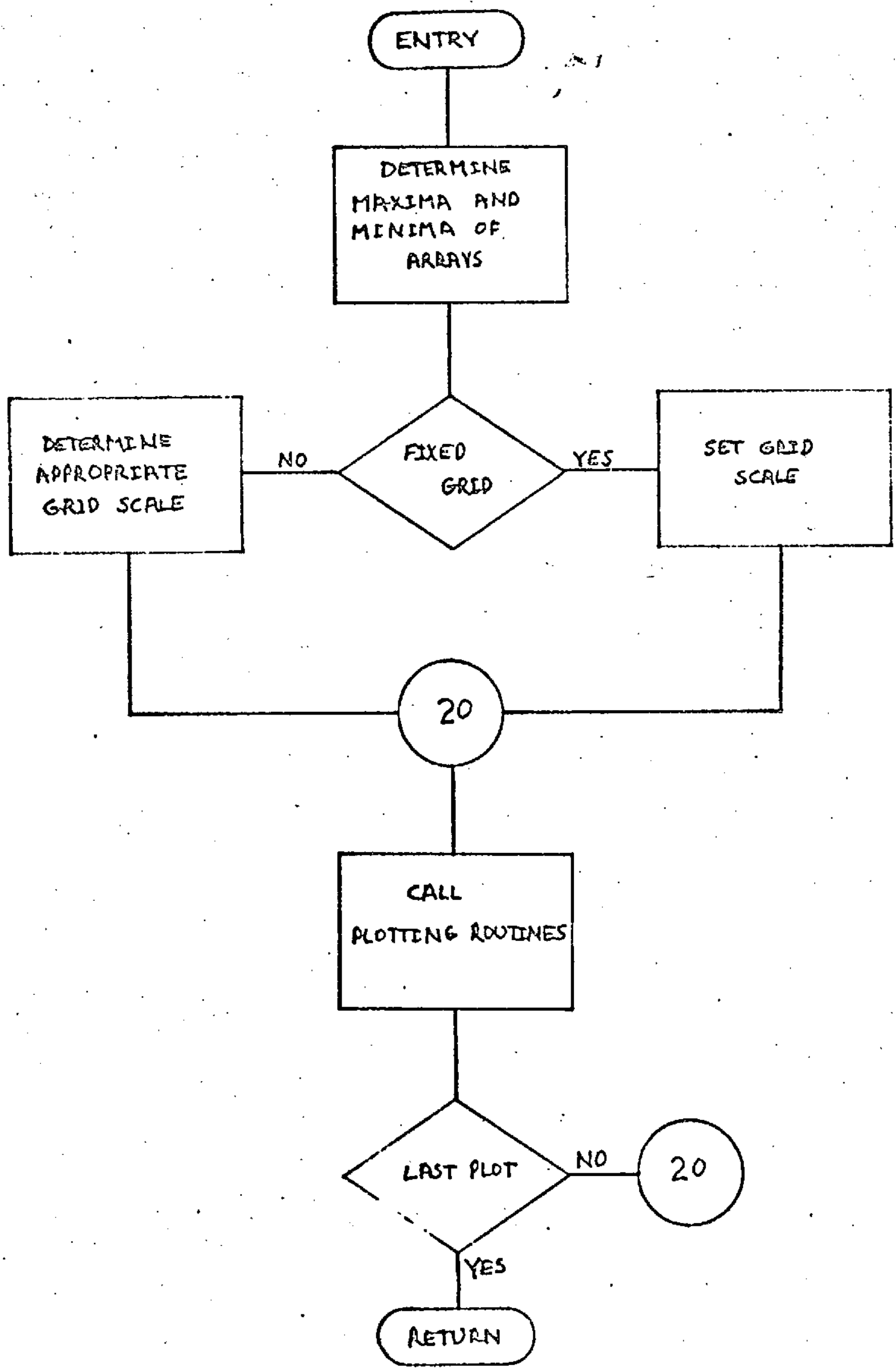
NAME	READER
PURPOSE	PLOTS ORBITAL DIFFERENCES
CALLING SEQUENCE	CALL READER(DT,LASTSW)

SYMBOL	TYPE	DESCRIPTION
DT	DP	INPUT - TIME FROM EPOCH IN DAYS
LASTSW	L	INPUT - LAST PLOT SWITCH

SUBROUTINES USED	AMOD	EDIT	FRMADV	HORLIN	MAXMIN
	OGRID	PLOT	PLOTST	PTYNUM	VERLIN
COMMON BLOCKS	CPLOT$	PLOTTP			
INPUT FILES	NONE				
OUTPUT FILES	NONE				
RESTRICTIONS	NONE				
REFERENCES	NONE				

```
      SUBROUTINE READER(DT,LASTSW)
      LOGICAL LASTSW
      REAL NTRVL
      DOUBLE PRECISION DT
      COMMON/CPLOT$/G1(2),LOGX,LOGY,XLOLIM,YLOLIM,XHILIM,YHILIM,
     *   XSCAL,YSCAL,FXLD,FYLD,G2(5)
      COMMON/PLCTTP/DAYS(4000),RADIAL(4000),CRSTRK(4000),ALGTRK(4000),
     *        ARRAY(21),IYMD2,IHMS2,INDEX,NOPT,SCALE1,SCALE2,NTRVL
      LOGICAL PLOTS,TAPE
      DOUBLE PRECISION TITLE(3),ARRAY
      DATA TITLE/8HEPOCH    ,8H        ,8H        /
      DATA NUM1/143/
      DATA TAPE/.FALSE./
C INITIALIZE
      IF(NOPT.NE.4) TAPE=.TRUE.
      DBASE=DAYS(1)-DT
      ANUM=3456.0D0*DT
      DO 10 I=1,INDEX
   10 DAYS(I)=(DAYS(I)-DBASE)*24.
C DETERMINE MAXIMA AND MINIMA OF ARRAYS
   15 CALL MAXMIN(RADIAL,INDEX,RMIN,RMAX)
      CALL MAXMIN(ALGTRK,INDEX,ALMIN,ALMAX)
      CALL MAXMIN(CRSTRK,INDEX,CMIN,CMAX)
      IF(NTRVL.GT.0) GO TO 17
      REALMX = AMAX1 (RMAX,CMAX,ALMAX)
      REALMN = AMIN1 (RMIN,CMIN,ALMIN)
C DETERMINE MAXIMUM AND MINIMUM OF PLOTTING GRID
```

```
      CALL PTYNUM(REALMN,REALMX,REALMN,REALMX,NY)
      GO TO 15
   17 CONTINUE
      REALMX = SCALE1
      REALMN = SCALE2
      NY=NTRVL+.5
   16 CONTINUE
      ISTART=1
      XK=0.
      XMIN=0.
C START PLOT
      PLOTS=.TRUE.
      CALL PLOTST(NCPT,PLOTS)
      CALL HORLIN(22HTRAJECTORY DIFFERENCES,22,512,500)
      CALL EDIT(IYMD2,'(I6)',TITLE(2),P)
      CALL EDIT(IHMS2,'(I6)',TITLE(3),P)
      CALL HORLIN(ARRAY( 1),56,512,470)
      CALL HORLIN(ARRAY( 8),56,512,450)
      CALL HORLIN(ARRAY(15),56,512,430)
      CALL HORLIN(TITLE,24,512,400)
C INITIALIZE GRID ON NEXT PAGE
   20 CALL FRMADV
      CALL HORLIN(22HTRAJECTORY DIFFERENCES,22,512,1000)
      CALL HORLIN(16HHOURS FROM EPOCH,16,512,0)
      CALL HORLIN(25H----- RADIAL DIFFERENCES ,25,512,503)
      CALL HORLIN(29H***** CROSS TRACK DIFFERENCES,29,512,487)
      CALL HORLIN(29H..... ALONG TRACK DIFFERENCES,29,512,471)
   25 XK=AMOD(XK+1.,2.)
      YLOLIM=32.+500.*XK
      YHILIM=432.+500.*XK
      NUM=MIN0(INDEX-ISTART,NUM1)+1
      XMAX=XMIN+ANUM
      CALL DGRID(XMIN,XMAX,12,'F5.1)',1,REALMN,REALMX,NY,'F7.1)',1,0)
      XMIN=XMAX
      CALL VERLIN(6HMETERS,6,0,INT(YHILIM+YLOLIM)/2)
C PLOT DATA POINTS
      CALL PLOT(DAYS(ISTART),RADIAL(ISTART),NUM,4H    )
      CALL PLOT(DAYS(ISTART),CRSTRK(ISTART),NUM,4H   *)
      CALL PLOT(DAYS(ISTART),ALGTRK(ISTART),NUM,4H   .)
      ISTART=ISTART+NUM
      IF(INDEX.LT.ISTART) GO TO 50
      IF(XK.GT.0.) GO TO 25
      GO TO 20
   50 CALL FRMADV
      IF(TAPE.AND.LASTSW) CALL PLOTST(7,.FALSE.)
C END OF PLOT
      IF(LASTSW) CALL ENDPLT
      RETURN
      END
```


ENTRY
DETERMINE MAXIMA AND MINIMA OF ARRAYS
DETERMINE APPROPRIATE GRID SCALE
NO
FIXED GRID
YES
SET GRID SCALE
20
CALL PLOTTING ROUTINES
LAST PLOT
NO
20
YES
RETURN

ADDYMD

DESCRIPTION

(See GEODYN)

1.1.2 GEORGE

INTRODUCTION

The support program GEORGE analyzes GEODYN measurement residuals. The residuals enter GEORGE from a tape generated by GEODYN and are analyzed on a pass by pass basis for either the station and/or measurement type specified by card input to GEORGE.

The main routine GEORGE selects the residuals to be analyzed and breaks them up into individual passes. GEORGE also controls which types of plots are to be made, if any.

REGANL performs the regression analysis and can edit data points on the basis of their standard deviations from the mean.

The subroutines HISTO and PLOTER provide visual aids in analyzing the residuals. HISTO plots a histogram of either the residuals or the ratios to sigma for each pass and a grand summation histogram for all the passes analyzed. PLOTER plots either residuals versus time or measurement rate versus residuals for each pass of data. Both subroutines are driver routines for the Plot Package.

The subroutine DIFF computes the difference in days between any two dates, and the subroutine RYMDI resolves a date in one word into three words: the year, the month, and the day.

GEORGE requires approximately 525K bytes of core and the IBM 360 system routines DSQRT and MOD. GEORGE will analyze about 1000 residuals in less than three minutes.

The subroutine REGANL determines measurement biases (or zero-set errors) and timing errors in each pass of data and then performs a regression and analysis of the residuals.

The zero-set error, A, and timing error, B, are determined by using a least squares method of solving the following equation: REGANL

$$Y = A + BX \qquad (1)$$

where

Y is the residual and

X is the measurement rate.

Taking the partials of (1) with respect to B and then with respect to A and setting them to zero, we get

$$\sum_{i=1}^{N} X_i Y_i - B \sum_{i=1}^{N} X_i^2 - A \sum_{i=1}^{N} X_i = 0 \qquad (2)$$

$$\sum_{i=1}^{N} Y_i - B \sum_{i=1}^{N} X_i - NA = 0 \qquad (3)$$

where N is the number of points in the pass. REGANL

The two equations are solved simultaneously for A and B.

First REGANL computes the sums of the rates,

$$\sum_{i=1}^{N} X_i,$$

and residuals,

$$\sum_{i=1}^{N} Y_i,$$

the products of X_i and Y_i,

$$\sum_{i=1}^{N} X_i Y_i,$$

the squares of the rates,

$$\sum_{i=1}^{N} X_i^2$$

and finally, the squares of the residuals,

$$\sum_{i=1}^{N} Y_i^{\,2}.$$

Then the corrected sum of the products, CSXY, and the corrected sums of the squares, CSX^2 and CSY^2, are computed as follows:

$$CSXY = \sum_{i=1}^{N} X_i Y_i - \sum_{i=1}^{N} X_i \sum_{i=1}^{N} Y_i \Big/ N$$

$$CSX^2 = \sum_{i=1}^{N} X_i^{\,2} - \left(\sum_{i=1}^{N} X_i \right)^2 \Big/ N$$

$$CSY^2 = \sum_{i=1}^{N} Y_i^{\,2} - \left(\sum_{i=1}^{N} Y_i \right)^2 \Big/ N$$

Now, solving for B we get

$$B = CSXY/CSX^2,$$

and solving for A using B we get

$$A = \left(\sum_{i=1}^{N} Y_i - B \sum_{i=1}^{N} X_i \right) \Big/ N.$$

The regression analysis is performed next. (See Anderson, R.L., and Bancroft, J.A., Statistical Theory in Research, 1952, McGraw-Hill Book Co., Inc., New York, pp. 156-157.)

The regression sum of squares, RSS, is REGANL

$$RSS = CSXY^2 / CSX^2$$

and the regression mean, RM, is

$$RM = (CSY^2 - RSS) / (N - 1),$$

which is nothing more than the square of the standard deviation of the residuals about the trajectory.

The standard deviations of the zero-set error, SDZ, and timing error, SDT, are

$$SDZ = \sqrt{RM \sum_{i=1}^{N} X_i^2 / NCSX^2}$$

and

$$SDT = \sqrt{RM / (N-1)}$$

The noise about the fitted line, D, is

$$D = \sqrt{RM} \ .$$

The residual mean square, RMSQ, is computed as

$$RMSQ = \frac{CSY^2 - RSS}{N - 1} \ .$$

To test the randomness of the result, we compute the residuals corrected for zero-set and timing error biases, CR_i, as

$$CR_i = RESID_i - A_i - B_i X_i$$

where $RESID_i$ is the residual.

Then we compute difference sum of squares between subsequent residuals, DSQ, as

$$DSQ = \sum_{i=1}^{N} \left(CR_{i+1} - CR_i \right)^2 \ .$$

The random normal deviate, RND, is then

$$RND = \frac{\left(\frac{DSQ}{2RM}\right) - 1}{\sqrt{(N-2)/(N^2-1)}}$$

The noise is random if

$$|RND| < 2.58$$

and non-random if

$$|RND| > 2.58.$$

SUBROUTINE CROSS REFERENCE CHART

CALLING ROUTINES

CALLED ROUTINES	MAIN	DIFF	HISTO	NEWMM	PLOTER
AMAX1				O	
AMIN1				O	
DIFF	O				
EDIT			O		O
ENDPLT	O				O
FRMADV	O		O		O
HISTO	O				
HORLIN			O		O
MAXMIN			O		O
MINT			O		O
NEWMM			O		O
OGRID			O		O
PLOT			O		
PLOTER	O				
PLOTST	O				O
PTYNUM			O		O
REGANL	O				
RYMDI		O			
VERLIN			O		O

COMMON BLOCK CROSS REFERENCE CHART

ROUTINES

COMMON BLOCKS	MAIN	HISTO	NEWMM	PLOTER	REGANL
ARRAY	●	●	●	●	●
COONST	●				●
LOGIC	●	●	●		

MAIN-GEORGE

DESCRIPTION

The main routine GEORGE reads the GEORGE INPUT CARDS and sets the switches for the type, station number and network of the data to be analyzed. It also sets the switches for the type of analysis (residual or ratio) and type or types of plots desired. GEORGE then reads the residual tape and separates the data into passes. Once a pass is established, GEORGE calls REGANL to compute the zero set and timing errors and perform the regression analysis. If plots of the residuals or ratios are desired, PLOTER is called. If histograms are desired, HISTO is called. This procedure is followed until all the data specified is analyzed.

The tracking networks acceptable to GEORGE and the code abbreviations are given below:

Network	Code Name
STADAN	STADAN
DOPPLER	DOPLER
U.S.A.F.	USAF
C-BAND	C BAND
SECOR	SECOR
U.S.C.&G.S.	USC+GS
SPEOPT	SPEOPT
INTERNATIONAL	INTERL
SAO	SAO

The types of measurements and the code names acceptable to GEORGE are listed below:

Measurement Type	Code Name
right ascension	RT ASC
declination	DECLIN
range	RANGE
range rate	R RATE
alpha	ALPHA
beta	BETA
x angle	X ANGL
y angle	Y ANGL
azimuth	AZMUTH
elevation	ELEV

NAME	MAIN - GEORGE				
PURPOSE	PERFORMS A REGRESSION ANALYSIS OF RESIDUALS OR RATIOS AND PLOTS THE RESULTS				
SUBROUTINES USED	DATE	DIFF	HISTO	REGANL	RYMDI
	EDIT	ENDPLT	FRMADV	HORLIN	MAXMIN
	MINT	NEWMM	OGRID	PLOTST	PIYNUM
	VERLIN	PLOTER			
COMMON BLOCKS	ARRAY	LOGIC	COONST		
INPUT FILES	GEORGE INPUT CARDS NONAME RESIDUAL TAPE				
OUTPUT FILE	PRINTER				
RESTRICTIONS	A MAXIMUM OF 4000 POINTS PER PASS WILL BE ANALYZED				
REFERENCES	NONE				

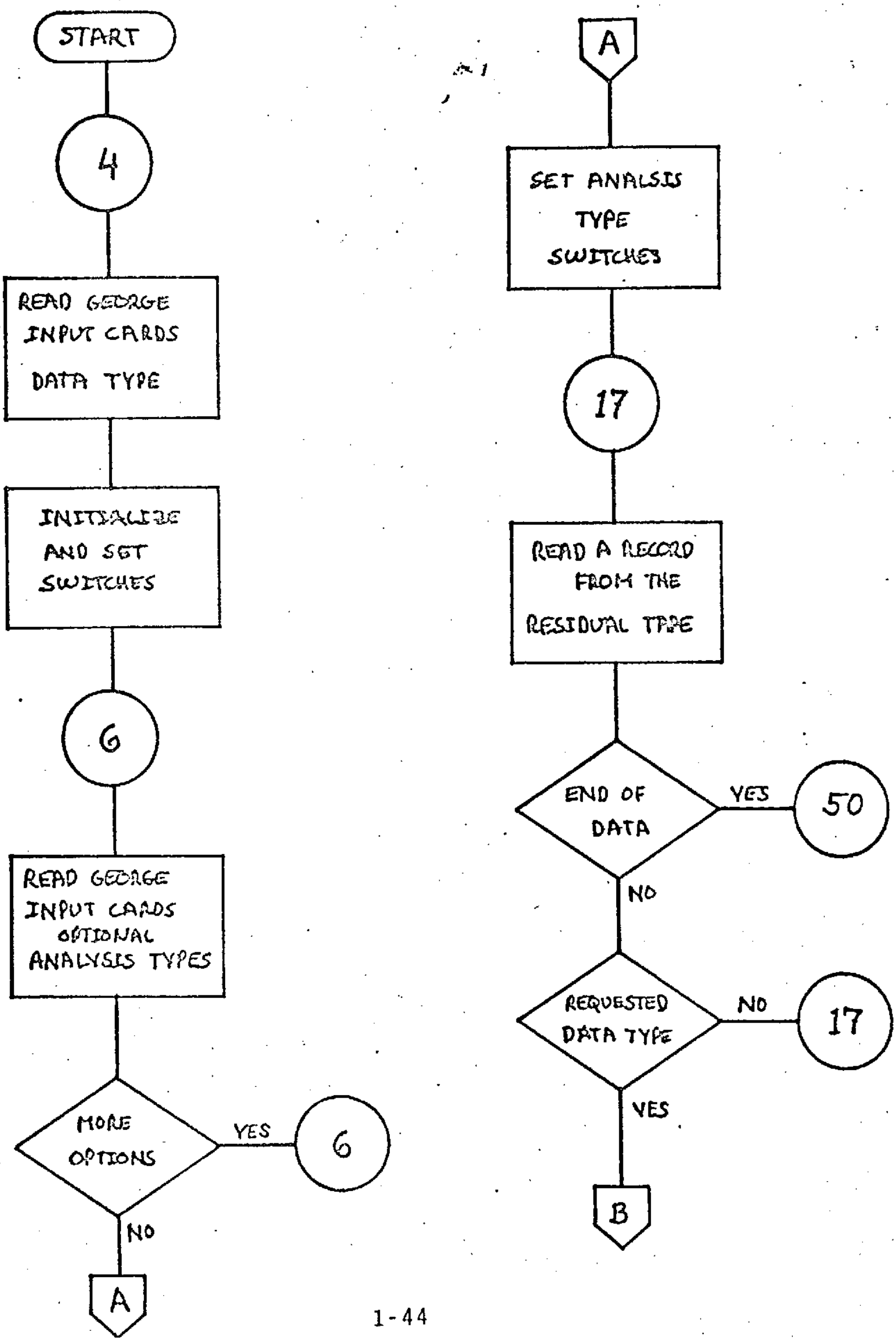
```
10001 FORMAT(3(A6,4X))                                                     GEOR
10002 FORMAT(A6,4X,5(F10.0))                                               GEOR
20001 FORMAT(1H1,10X,'*** ANALYSIS OF ',A6,' RESIDUALS ***')               GEOR
20002 FORMAT(1H ,/,14X,'NETWORK - - ',A6)                                  GEOR
20003 FORMAT(1H ,14X,'STATION - - ',A6)                                    GEOR
20004 FORMAT(1H ,50H ILLEGAL MEASUREMENT TYPE - SKIPPING TO NEXT CASE ) GEOR
20005 FORMAT(1H ,50H ILLEGAL NETWORK NAME     - SKIPPING TO NEXT CASE ) GEOR
20006 FORMAT(1H ,28H ILLEGAL OPTION CARD      - ,A6,46H REMAINING OPTIONGEOR
     *S IGNORED  - SKIPPING TO DATA)                                       GEOR
20020 FORMAT(1H0,62H NO DATA OF THE TYPE SPECIFIED FOUND  -- SKIPPING TOGEOR
     * NEXT CASE)                                                          GEOR
20021 FORMAT(1H0,20H OBSERVATIONS BELOW ,F5.1,29H DEGREES WILL NOT BE ANGEOR
     *ALYZED)                                                              GEOR
20022 FORMAT(1H0,21H RESIDUALS DEVIATING ,F4.1,                            GEOR
     ,53H UNITS OR MORE FROM THE FITTED LINE WILL BE REJECTED)             GEOR
20023 FORMAT(1H0,43H TOO MANY OBSERVATIONS -- REMAINDER IGNORED)           GEOR
      DOUBLE PRECISION  ACHAN(3,50) , ATYPE , CHAN  , CCHAN(3,50) , DATAGEOR
      DOUBLE PRECISION  EL    , ELEV  , FNET(9)    , FTYPE                 GEOR
      DOUBLE PRECISION  FTYPE1(14)  , GRARR , IBLANK, ISTA  , KSTA         GEOR
      DOUBLE PRECISION  LASER , LAST  , MTYPE , NAME  , NAMEST, NET        GEOR
      DOUBLE PRECISION  OBSO1 , OBSO2 , OPT   , OPTION(10)  , SAOLAS       GEOR
      DOUBLE PRECISION  STNAME(100) , TEST  , TYPE                         GEOR
      DIMENSION IEND(100) , ISTART(100), VALUE(5)                          GEOR
      COMMON /ARRAY / IYMD(4000),IHM(4000),SEC(4000),ELEV(4000),           GEOR
     .                RESID(4000),OBOOT(4000),ICOUNT(4000),NAMEST(4000),GEOR
     .                FTYPE(14),RATIO(4000)                                GEOR
      COMMON /LOGIC / LASTIM,SWITCH                                        GEOR
      COMMON /COONST/ NET,TYPE,KSTA,A,B,ISAVE,REJECT,REJSW                 GEOR
      LOGICAL ANGSW,HISTSW,HSTST,LASTIM,PLOTSW,PLTST,PRCSW,REJSW,SWITCH GEOR
      LOGICAL GRDSW                                                        GEOR
      DATA HSTST,PLTST / 2*.FALSE. /                                       GEOR
```

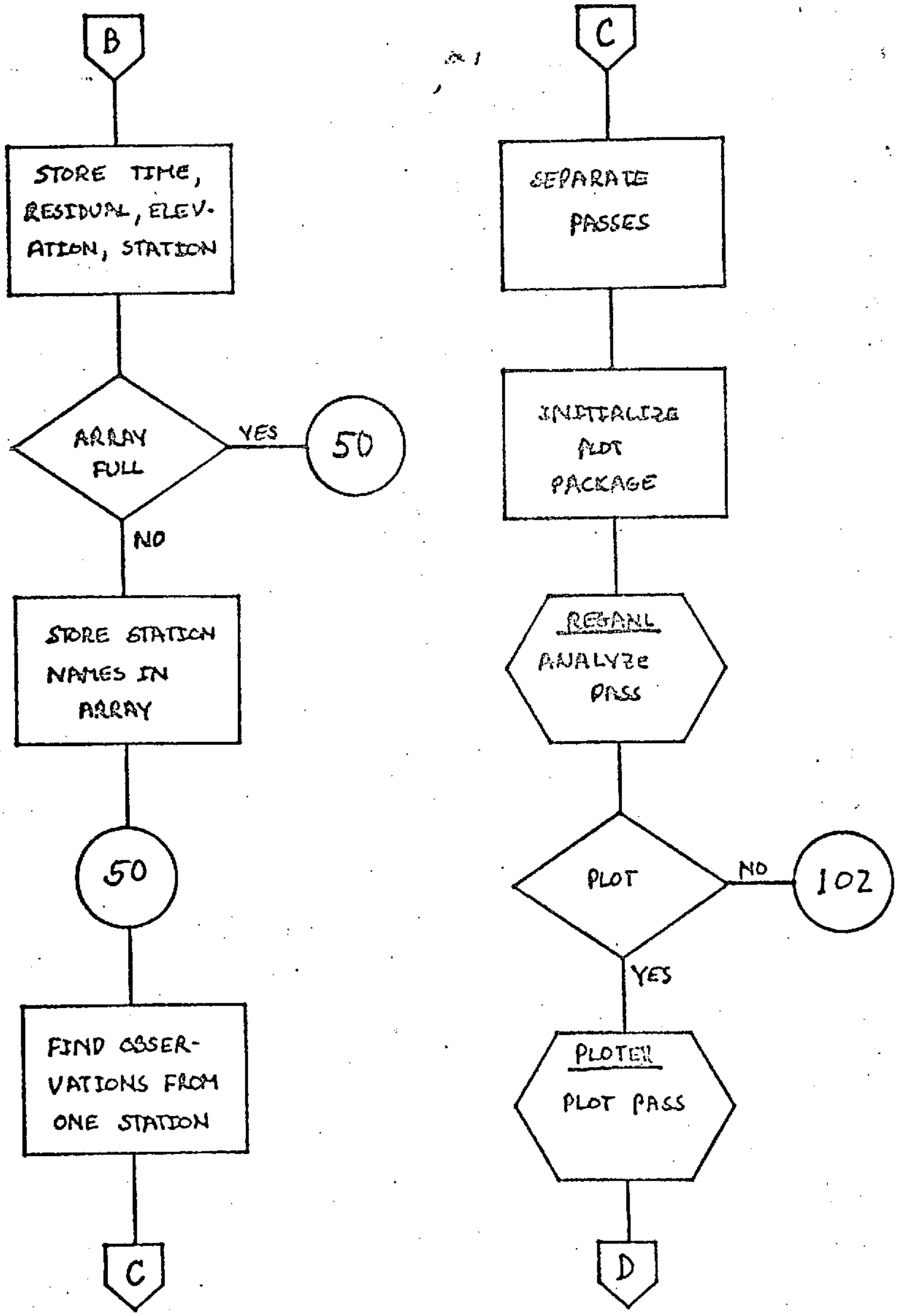
```
      REJECT=1.0E9
      DATA JACHAN,JCCHAN / 2*0 /
      DATA GRARR,IBLANK,LASER,LAST,SAOLAS/6HGRARR ,6H      ,6HLASER ,
     .     6HLAST  ,6HSAOLAS/
      DATA FTYPE1/6HRT ASC,6HRANGE ,6HR RATE,6HFREQ  ,6HALPHA ,6HX ANGL,
     .            6HAZMUTH,6HDECLIN,6H      ,6H      ,6H      ,6HBETA  ,
     .            6HY ANGL,6HELEV  /
      DATA FNET/6HSTADAN,6HDOPLER,6HUSAF  ,6HC BAND,6HSECOR ,6HUSC+GS,
     .          6HSPEOPT,6HINTCPL,6HSAO   /
      DATA OPTION/6HCHAN A,6HEL CUT,6HCHAN C,6HAMBIG ,6HPROCES,6HHISTGM,
     .            6HPLOT  ,6H      ,6HREJECT,6HDATA  /
      DO 105 I=1,14
  105 FTYPE(I)=FTYPE1(I)
C READ GEORGE INPUT CARDS AND SET SWITCHES FOR NETWORK AND STATION
    4 READ 10001,TYPE,NET,KSTA
      IF (NET.EQ.GRARR)FNET(1) = GRARR
      IF (NET.EQ.LASER)FNET(7) = LASER
      IF (NET.EQ.SAOLAS)FNET(7)= SAOLAS
      IF(NET.EQ.CBAND)FNET(9)=CBAND
      PRINT 20001,TYPE
      GRDSUM=.FALSE.
      LASTIM=.FALSE.
      SWITCH=.FALSE.
      ITOT=0
      NSAVE=0
      NMEAS=0
      ISAVE=0
      DO 3 I=1,7
      IF (TYPE.EQ.FTYPE(I))ISAVE = I
    3 CONTINUE
      IF(ISAVE.NE.0) GO TO 99
      PRINT 20004
      GO TO 27
   99 CONTINUE
      IF (NET.EQ.IBLANK) GO TO 1
      PRINT 20002,NET
      DO 5 I=1,9
      IF (NET.EQ.FNET(I))NSAVE=I
    5 CONTINUE
      IF (NSAVE.NE.0) GO TO 1
      PRINT 20005
      GO TO 27
    1 CONTINUE
      IF (KSTA.NE.IBLANK)PRINT 20003,KSTA
C INITIALIZE CONSTANTS AND SWITCHES
      DO 20 I=1,100
   20 STNAME(I)= IBLANK
      CUT = 0.0
      I4020=4
      AMBSW=.FALSE.
      HISTSW=.FALSE.
      PLOTSW=.FALSE.
      PROSW=.FALSE.
      REJSW=.FALSE.
C READ GEORGE INPUT  CARDS AND SET ANALYSIS AND PLOTTING OPTIONS
    6 READ 10002,OPT,VALUE
```

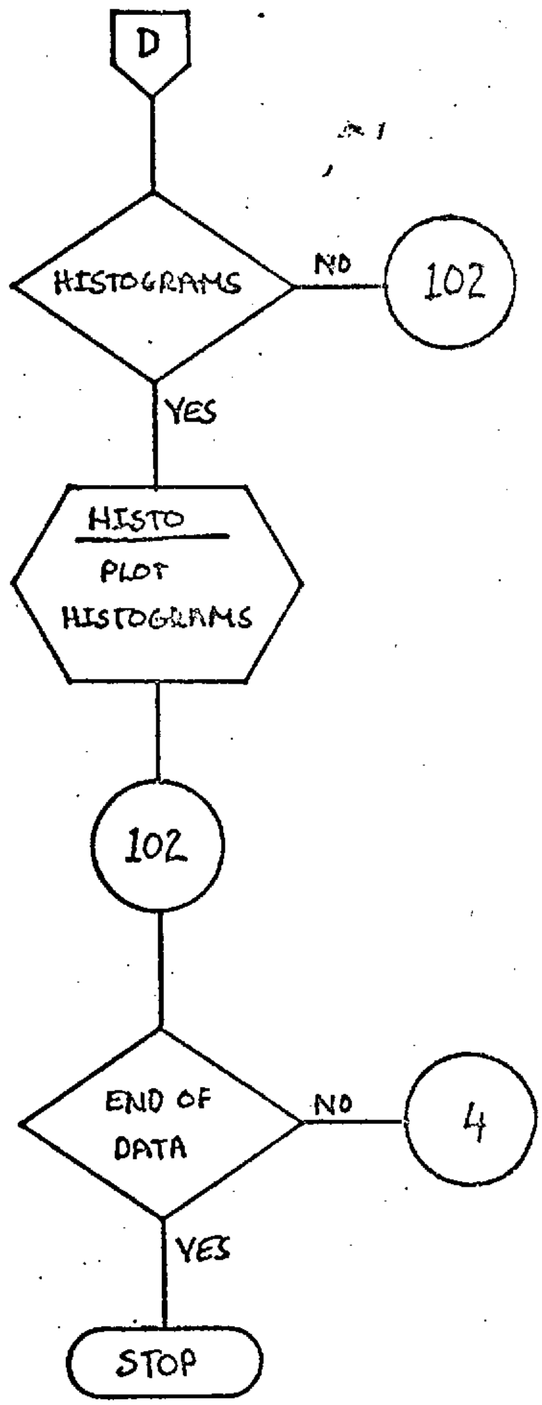
```
      DO 7 I=1,10
      IF (OPT.NE.OPTION(I))GO TO 7
      GO TO (8,9,10,11,12,13,14,15,16,17),I
    7 CONTINUE
      PRINT 20006,OPT
      GO TO 17
    8 JACHAN = JACHAN + 1
      ACHAN(1,JACHAN) = VALUE(1)
      ACHAN(2,JACHAN) = VALUE(2)
      ACHAN(3,JACHAN) = VALUE(4)+1
      GO TO 6
    9 CUT = VALUE(1)
      PRINT 20021,CUT
      GO TO 6
   10 JCCHAN = JCCHAN + 1
      CCHAN(1,JCCHAN)= VALUE(1)
      CCHAN(2,JCCHAN)= VALUE(2)
      CCHAN(3,JCCHAN)= VALUE(4) + 1
      GO TO 6
   11 AMBSW = .TRUE.
      GO TO 6
   12 PROSW = .TRUE.
      GO TO 6
   13 HISTSW = .TRUE.
      SWITCH=VALUE(1).EQ.1..OR.VALUE(1).EQ.3.
      GROSUM=VALUE(1).EQ.2..OR.VALUE(1).EQ.3.
      GO TO 6
   14 PLOTSW = .TRUE.
      IF(VALUE(1).EQ.1.) I4020=6
      GO TO 6
   15 CONTINUE
      GO TO 6
   16 CONTINUE
      REJSW =.TRUE.
       REJECT = VALUE(1)
      PRINT 20022,REJECT
      GO TO 6
C READ NONAME RESIDUAL TAPE
   17 READ(15,END=18,ERR=18)  IYMD1,IHM1,SEC1,ISTA,MTYPE,OBS01,RESID1,
     .      RATIO1,OBDOT1,OBS02,RESID2,RATIO2,OBDOT2,EL,INET
C TEST FOR END OF DATA
      IF (ISTA.EQ.IBLANK)GO TO 18
      IF (INET.EQ.0)INET = 2
C TEST FOR REQUESTED DATA
      IF(EL.LT.CUT) GO TO 17
      IF (ISTA.NE.KSTA.AND.KSTA.NE.IBLANK)GO TO 17
      IF (NET.NE.FNET(INET).AND.NET.NE.IBLANK)GO TO 17
      IF (MTYPE.NE.TYPE)GO TO 17
C STORE RESIDUALS, TIME, ELEVATION, AND STATION NAME
      NMEAS = NMEAS + 1
      IF (NMEAS.LT.4000)GO TO 104
      PRINT 20023
      GO TO 18
  104 IYMD(NMEAS)= IYMD1
      IHM(NMEAS) = IHM1
      SEC(NMEAS) = SEC1
```

```
      ELEV(NMEAS) = EL
      NAMEST(NMEAS) = ISTA
      IF(KSTA.EQ.IBLANK) GO TO 101
      IST=1
      STNAME(1)=KSTA
      GO TO 22
  101 CONTINUE
      DO 21 I=1,100
      IF (STNAME(I).EQ.ISTA)GO TO 22
      IF(STNAME(I).NE.IBLANK) GO TO 21
      IST=I
      STNAME(I)=ISTA
      GO TO 22
   21 CONTINUE
   22 CONTINUE
      RESID(NMEAS) = RESID1
      RATIO(NMEAS) = RATIO1
      OBDOT(NMEAS) = OBDOT1
C STORE PAIRED RESIDUALS IN UPPER HALF OF ARRAY
      IF(ISAVE.NE.1.AND.ISAVE.NE.6.AND.ISAVE.NE.7) GO TO 17
      MEASN=NMEAS+2000
      RESID(MEASN) = RESID2
      RATIO(MEASN) = RATIO2
      OBDOT(MEASN) = OBDOT2
      IYMD(MEASN)=IYMD1
      IHM(MEASN)=IHM1
      SEC(MEASN)=SEC1
      ELEV(MEASN)=EL
      NAMEST(MEASN)=ISTA
      GO TO 17
   18 REWIND 15
      IF (NMEAS.NE.0)GO TO 50
      PRINT 20020
      GO TO 102
   50 CONTINUE
C MATCH MEASUREMENTS WITH STATION NAME
      DO 23 J =1,IST
      JJ=0
      DO 19 I=1,NMEAS
      IF(NAMEST(I).NE.STNAME(J))GO TO 19
      JJ = JJ + 1
       ICOUNT(JJ)=I
   19 CONTINUE
      ISTART(1) = ICOUNT(1)
      NPASS = 1
      KK = JJ - 1
      DO 24 I=1,KK
      N = ICOUNT(I)
      M = ICOUNT(I+1)
C TEST FOR  A NEW PASS
      IHMS1 = IHM(N)*100
      IHMS2 = IHM(M)*100
      CALL DIFF(IYMD(N),IHMS1,IYMD(M),IHMS2,IDAY,ISEC)
      IHR=IDAY*24+ISEC/3600
      IF (IHR.LT.1)GO TO 24
      NPASS = NPASS + 1
```

```
      ISTART(NPASS) = M
      IEND(NPASS-1) = N
   24 CONTINUE
      IF((.NOT.PLOTSW.OR.PLTST).AND.(.NOT.HISTSW.OR.HSTST)) GO TO 106
C INITIALIZE PLOT PACKAGE
      CALL PLOTST(IAC20,.TRUE.)
      CALL FRMADV
      PLTST=.TRUE.
      HSTST=.TRUE.
  106 IEND(NPASS)=M
      DO 25 I=1, NPASS
      M = ISTART(I)
C PERFORM THE REGRESSION ANALYSIS
      CALL REGANL(ISTART(I),IEND(I),STNAME(J),MM)
      IF(.NOT.PLOTSW) GO TO 107
C MAKE THE PLOTS
      CALL PLOTER(ISTART(I),IEND(I),MM,STNAME(J),ISAVE)
      IF(ISAVE.EQ.1.OR.ISAVE.EQ.6.OR.ISAVE.EQ.7)
     .CALL PLOTER(ISTART(I)+2000,IEND(I)+2000,MM,STNAME(J),ISAVE+7)
  107 IF (GRDSUM) GO TO 25
      IF (.NOT.HISTSW) GO TO 25
C MAKE THE HISTOGRAMS
      CALL HISTO(ISTART(I),IEND(I),MM,STNAME(J),ISAVE)
      IF(ISAVE.NE.1.AND.ISAVE.NE.6.AND.ISAVE.NE.7) GO TO 25
      CALL  HISTO(ISTART(I)+2000,IEND(I)+2000,MM,STNAME(J),ISAVE+7)
   25 CONTINUE
   23 CONTINUE
      LASTIM=.TRUE.
      IF (.NOT.HISTSW) GO TO 102
      CALL HISTO(1,NMEAS,MM,STNAME(J),ISAVE)
      IF (ISAVE.EQ.1.OR.ISAVE.EQ.6.OR.ISAVE.EQ.7)
     .                CALL HISTO(2001,MEASN,MM,STNAME(4),ISAVE+7)
C TEST FOR LAST DATA CARD
  102 READ 10002,TEST
      IF (TEST.EQ.IBLANK) GO TO 4
      IF (TEST.EQ.LAST) GO TO 27
      GO TO 102
   27 CONTINUE
      IF(PLTST.OR.HSTST) CALL ENDPLT
      STOP
      END
```


START
4
READ GEORGE INPUT CARDS DATA TYPE
INITIALIZE AND SET SWITCHES
6
READ GEORGE INPUT CARDS OPTIONAL ANALYSIS TYPES
MORE OPTIONS
YES
6
NO
A
A
SET ANALSIS TYPE SWITCHES
17
READ A RECORD FROM THE RESIDUAL TAPE
END OF DATA
YES
50
NO
REQUESTED DATA TYPE
NO
17
YES
B


B
STORE TIME, RESIDUAL, ELEV-ATION, STATION
ARRAY FULL
YES
50
NO
STORE STATION NAMES IN ARRAY
50
FIND OBSER-VATIONS FROM ONE STATION
C
C
SEPARATE PASSES
INITIALIZE PLOT PACKAGE
REGANL
ANALYZE PASS
PLOT
NO
102
YES
PLOTER
PLOT PASS
D


D
HISTOGRAMS
NO
102
YES
HISTO
PLOT HISTOGRAMS
102
END OF DATA
NO
4
YES
STOP

DIFF

DESCRIPTION

(See EPHEMERIS TAPE GENERATOR)

HISTO

DESCRIPTION

HISTO determines the grid size and labels to produce histograms of each pass and a grand summation histogram if requested. It is basically a drive program for the WOLF PLOT PACKAGE; hence all routines called by HISTO are members of the PLOT PACKAGE.

NAME	HISTO
PURPOSE	PLOTS HISTOGRAMS OF ANALYZED DATA
CALLING SEQUENCE	CALL HISTO(ISTART,IEND,MM,NAME,ISAVE)

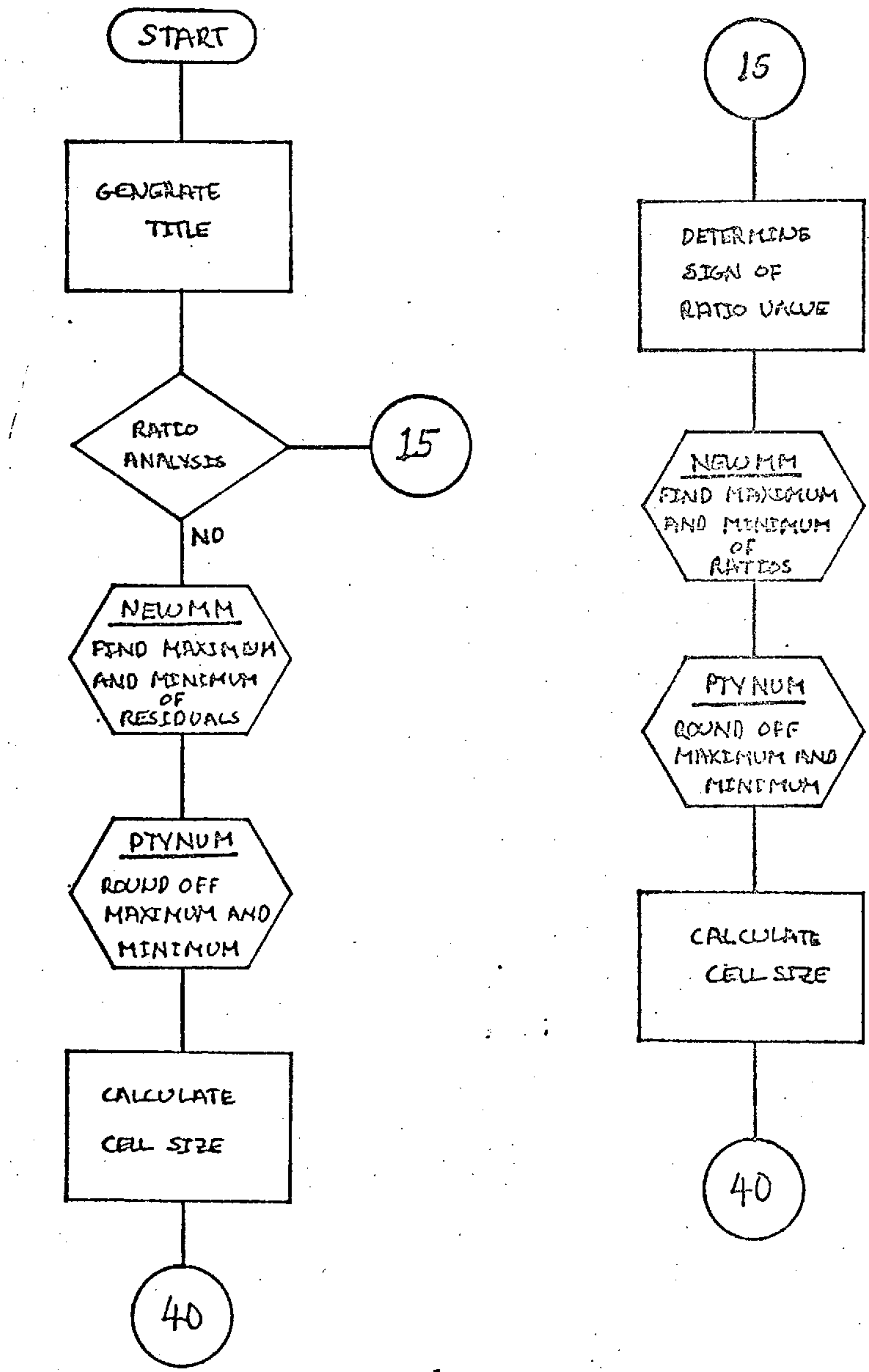
SYMBOL	TYPE	DESCRIPTION
ISTART	I	INPUT - INDEX OF START OF PASS IN ARRAY
IEND	I	INPUT - INDEX OF END OF PASS IN ARRAY
MM	I	INPUT - NUMBER OF POINTS IN PASS
ISAVE	I	INPUT - INDEX OF TYPE OF PLOT REQUESTED

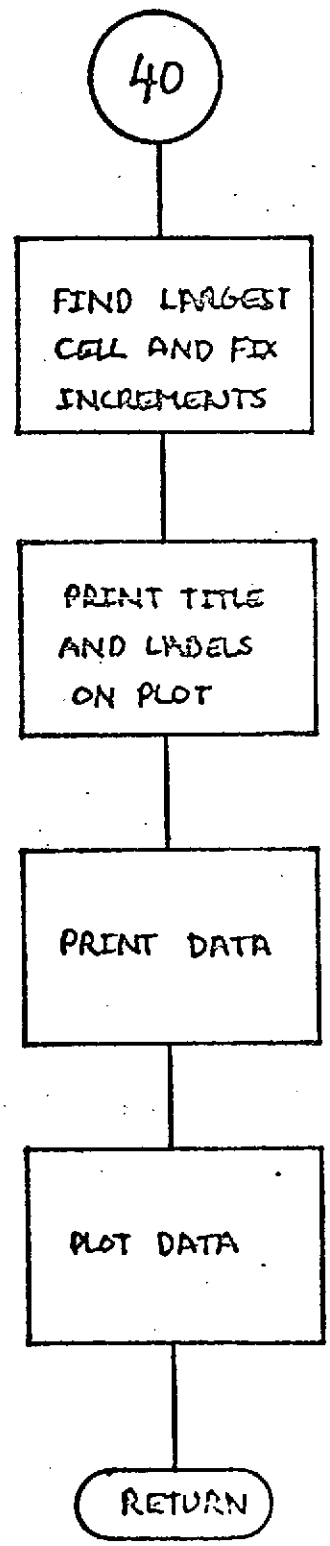
SUBROUTINES USED	EDIT	FRMADV	HORLIN	MAXMIN	MINT
	NEWMM	OGRID	PLOT	PTYNUM	VERLIN
COMMON BLOCKS	ARRAY	LOGIC			
INPUT FILES	NONE				
OUTPUT FILE	PRINTER				
RESTRICTIONS	NONE				
REFERENCES	NONE				

```
      SUBROUTINE HISTO(ISTART,IEND,MM,NAME,ISAVE)
      DOUBLE PRECISION  ELEV  , FTYPE , IBLANK, NAME   , NAMEST, TITLE(6)
      DOUBLE PRECISION  TYPE  , XTITL(6)    , YTITL(6)
      COMMON /ARRAY / IYMD(4000),IHM(4000),SEC(4000),ELEV(4000),
     .              RESID(4000),OBDOT(4000),ICOUNT(4000),NAMEST(4000),
     .              FTYPE(14),RATIO(4000)
      COMMON /LOGIC / LASTIM,SWITCH
      LOGICAL LASTIM,SWITCH
      DIMENSION SIZE(20),X(2),Y(2)
      DATA XTITL/8HHISTOGRA,8HM  OF  R,8HATIO  FR,3HFQUENCIE,8HS       ,
     .         8H        /
      DATA YTITL/8HHISTOGRA,8HM  OF  R,8HESIDUAL ,
     .         8H FREQUEN,8HCIES    ,8H        /
C GENERATE TITLE FOR TYPE OF HISTOGRAM
      IF (MM.LT.5) GO TO 70
      TYPE=FTYPE(ISAVE)
      TITLE(6)=TYPE
      DO 5 I=1,20
    5 SIZE(I)=0.
      IF(SWITCH) GO TO 15
C SET UP HISTOGRAM GRID SIZE FOR RESIDUAL ANALYSIS
      CALL NEWMM(NAME,RESID,ISTART,IEND,RMAX,RMIN)
      CALL PTYNUM(RMIN,RMAX, RMIN,RMAX,NX)
```

```
      CSIZE=(ABS(RMAX-RMIN)/NX)
      DO 10 I=ISTART,IEND
      IF(.NOT.LASTIM.AND.NAME.NE.NAMEST(I)) GO TO 10
      J=((RESID(I)-RMIN)/CSIZE)+1.
      SIZE(J)=SIZE(J)+1.
   10 CONTINUE
      GO TO 40
   15 DO 20 I=ISTART,IEND
      IF(.NOT.LASTIM.AND.NAME.NE.NAMEST(I)) GO TO 20
      RATIO(I)=SIGN(RATIO(I),RESID(I))
   20 CONTINUE
C SET UP HISTOGRAM GRID SIZE FOR RATIO ANALYSIS
      CALL NEWMM(NAME,RATIO,ISTART,IEND,RMAX,RMIN)
      IF (RMIN.EQ.RMAX) GO TO 65
      CALL PTYNUM(RMIN,RMAX, RMIN,RMAX,NX)
      CSIZE=(ABS(RMAX-RMIN)/NX)
      DO 30 I=ISTART,IEND
      IF(.NOT.LASTIM.AND.NAME.NE.NAMEST(I)) GO TO 30
      IF((RATIO(I)*1000.).EQ.0.) GO TO 30
      J=((RATIO(I)-RMIN)/CSIZE)+1.
      SIZE(J)=SIZE(J)+1.
   30 CONTINUE
C FIND THE HIGHEST COUNT AND DETERMINE INCREMENTS
   40 CALL MAXMIN(SIZE,NX,VMIN,VMAX)
      VMAX=VMAX+1.
      CALL PTYNUM(0.,VMAX,VMIN,VMAX,NY)
      IF(VMAX/FLOAT(NY).LT.1.) NY=VMAX
C SET GRID ENDPOINT VALUES
      CALL UGRID(RMIN,RMAX,NX,5HF6.1),1,VMIN,VMAX,NY,5HF0.1),1,0)
      IF(SWITCH) GO TO 45
      DO 41 I=1,5
   41 TITLE(I)=YTITL(I)
      CALL HORLIN(15HRESIDUAL VALUES,15,512,0)
      GO TO 50
   45 DO 46 I=1,5
   46 TITLE(I)=XTITL(I)
      CALL HORLIN(21HRATIO TO SIGMA VALUES,21,512,0)
   50 CALL HORLIN(TITLE,48,512,1000)
      IF(LASTIM)CALL HORLIN(38HGRAND SUMMATION OF ALL PASSES ANALYZED,
     .38,512,982)
      CALL VERLIN(15HFREQUENCY COUNT,15,0,512)
C PRINT DATA
      IF (.NOT.LASTIM) GO TO 56
      ITOTAL=0
      WRITE(6,1000)
      DO 55 J=1,NX
           VALUE1= RMIN + (J-1)*CSIZE
           VALUE2= RMIN +  J*CSIZE
           ITOTAL= ITOTAL + SIZE(J)
           WRITE(6,1001) VALUE1,VALUE2,SIZE(J)
  55  CONTINUE
      WRITE(6,1002) ITOTAL
  56  CONTINUE
C PLOT DATA
      X(1)=RMIN
      SIZE(NX+1)=0.
```

```
      DO 60 I=1,NX
      X(2)=X(1)+CSIZE
      Y(1)=SIZE(I)
      Y(2)=SIZE(I)
      IF(SIZE(I).GT.0.)CALL PLOT(X,Y,2,4H     )
      Y(1)=AMAX1(SIZE(I),SIZE(I+1))
      Y(2)=0.
      X(1)=X(2)
   60 IF (Y(1).GT.0.) CALL PLOT(X,Y,2,4H     )
      CALL FRMADV
      RETURN
65    WRITE(6,1003)
      RETURN
 70   PRINT 100
      RETURN
100   FORMAT(1H ,'    INSUFFICIENT DATA FOR A MEANINGFUL HISTOGRAM')
 101  FORMAT(1H ,30X,A6,11X,F10.4,10X,I3)
 102  FORMAT(1H ,30X,A6,11X,F10.4,10X,I3)
 103  FORMAT(1H1,27X,'STATION NAME',6X,'RESIDUAL VALUE',6X,'COUNT',//)
  104 FORMAT(1H1,27X,'STATION NAME',6X,'RESIDUAL VALUE',5X,'COUNT',6X,
     .'CONTINUED',//)
 105  FORMAT(1H1,27X,'STATION NAME',8X,'RATIO VALUE',6X,'COUNT',//)
  106 FORMAT(1H1,27X,'STATION NAME',8X,'RATIO VALUE',5X,'COUNT',6X,
     .'CONTINUED',//)
1000  FORMAT(1H1,25X,'HISTOGRAM DATA',//,20X,'INTERVAL',8X,'FREQUENCY',/
     ./)
1001  FORMAT(1H ,15X,F5.1,2X,'TO',2X,F5.1,10X,F5.1)
1002  FORMAT(1H ,//,14X,'TOTAL NO. OF WTD. POINTS = ',I6)
1003  FORMAT(1H ,///,5X,'THE RATIO TO SIGMA VALUES ARE ALL ZERO.  NO PLO
     .T CAN BE MADE.')
      END
```


START
GENERATE TITLE
RATIO ANALYSIS
15
NO
NEWMM
FIND MAXIMUM AND MINIMUM OF RESIDUALS
PTYNUM
ROUND OFF MAXIMUM AND MINIMUM
CALCULATE CELL SIZE
40
15
DETERMINE SIGN OF RATIO VALUE
NEWMM
FIND MAXIMUM AND MINIMUM OF RATIOS
PTYNUM
ROUND OFF MAXIMUM AND MINIMUM
CALCULATE CELL SIZE
40


40
FIND LARGEST CELL AND FIX INCREMENTS
PRINT TITLE AND LABELS ON PLOT
PRINT DATA
PLOT DATA
RETURN

NEWMM

DESCRIPTION

NEWMM is a simple program utilizing WOLF PLOT PACKAGE routines to determine maximum and minimum values for either all or part of a specified array.

NAME NEWMM

PURPOSE FIND THE MAXIMUM AND MINIMUM VALUES IN AN ARRAY FOR SPECIFIED STATIONS

CALLING SEQUENCE CALL NEWMM(NAME,ARRAI,ISTART,IEND,RMAX,RMIN)

SYMBOL	TYPE	DESCRIPTION
NAME	DP	INPUT - NAME OF STATION
ARRAI (4000)	R	INPUT - ARRAY TO BE SEARCHED
ISTART	I	INPUT - INDEX OF STARTING VALUE IN ARRAY
IEND	I	INPUT - INDEX OF ENDING VALUE IN ARRAY
RMAX	R	OUTPUT - MAXIMUM VALUE
RMIN	R	OUTPUT - MINIMUM VALUE

SUBROUTINES USED NONE

COMMON BLOCKS ARRAY LOGIC

INPUT FILES NONE

OUTPUT FILES NONE

RESTRICTIONS NONE

REFERENCES NONE

```
      SUBROUTINE NEWMM(NAME,ARRAI,ISTART,IEND,RMAX,RMIN)
      DOUBLE PRECISION        ELEV    ,FTYPE   ,NAME    ,NAMEST
      COMMON /ARRAY / IYMD(4000),IHM(4000),SEC(4000),ELEV(4000),
     .                RESID(4000),OBDOT(4000),ICOUNT(4000),NAMEST(4000),
     .                FTYPE(14),RATIO(4000)
      COMMON /LOGIC / LASTIM,SWITCH
      LOGICAL LASTIM
      DIMENSION ARRAI(4000)
      RMAX=ARRAI(ISTART)
      RMIN=RMAX
C TEST IF BEGINNING AND ENDING INDICES ARE DIFFERENT
      IF(ISTART.EQ.IEND)RETURN
      DO 10 I=ISTART,IEND
C SEARCH ARRAY FOR STATION NAME
      IF(.NOT.LASTIM.AND.NAMEST(I).NE.NAME) GO TO 10
C FIND MAXIMUM
      RMAX=AMAX1(RMAX,ARRAI(I))
C FIND MINIMUM
```

```
      RMIN=AMIN1(RMIN,ARRAY(I))
   10 CONTINUE
      RETURN
      END
```

PLOTER

DESCRIPTION

PLOTER is the drive program for the WOLF PLOT PACKAGE which produces the plots of residuals vs. time or measurement rate vs. residuals if either are requested.

NAME	PLOTER
PURPOSE	PLOTS RESIDUALS VS. TIME AND MEASUREMENT RATE VS. RESIDUALS FOR SPECIFIED PASSES
CALLING SEQUENCE	CALL PLOTER(IPASS,ISTOP,MM,NAME,ISAVE)

SYMBOL	TYPE	DESCRIPTION
IPASS	I	INPUT - INDEX OF STARTING POINT
ISTOP	I	INPUT - INDEX OF ENDING POINT
MM	I	INPUT - NUMBER OF POINTS IN PASS
ISAVE	I	INPUT - INDEX OF TYPE OF DATA

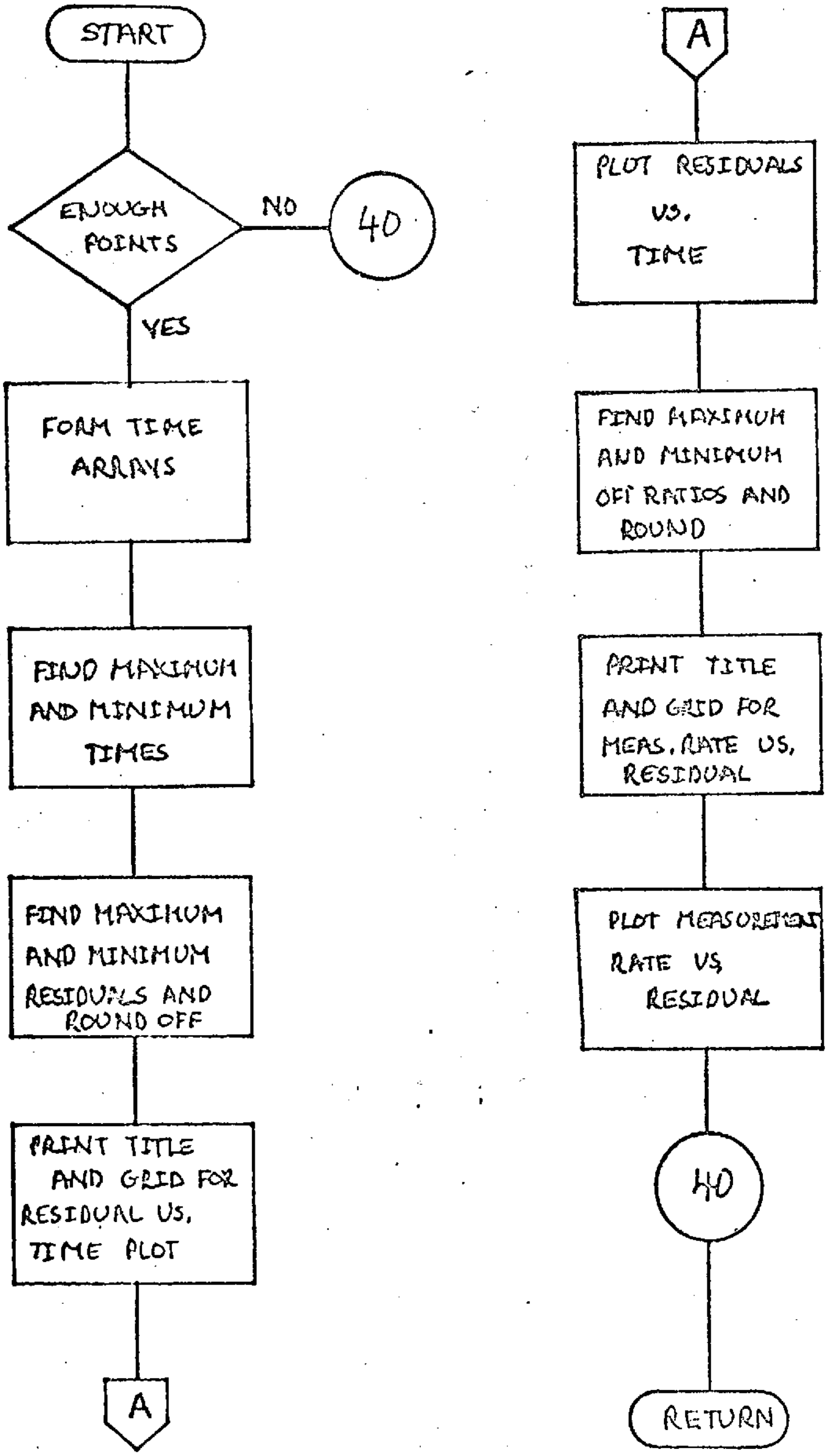
SUBROUTINES USED	EDIT	ENDPLT	FRMADV	HORLIN	MAXMIN
	MINT	OGRID	PLOTST	PTYNUM	VERLIN
	NEWMM				

COMMON BLOCK	ARRAY
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE PLOTER(IPASS,ISTOP,MM,NAME,ISAVE)
      DOUBLE PRECISION  ELEV  , FTYPE , NAMEST, TITLE(8)    , TYPE
      REAL*8 NAME
      COMMON /ARRAY / IYMD(4000),IHM(4000),SEC(4000),ELEV(4000),
     .               RESID(4000),OBDOT(4000),ICOUNT(4000),NAMEST(4000),
     .               FTYPE( 14),RATIO(4000)
      DIMENSION FMIN(4000)
      REAL MRMAX, MRMIN
      DATA TITLE/8H         ,8H         ,8H RESIDUA,8HLS       ,
     .          8H DATE    ,8H         ,8H TIME    ,3H      /
   50 FORMAT(53H INSUFFICIENT DATA IN THIS PASS FOR A MEANINGFUL PLOT)
C TEST IF ENOUGH DATA
      IF(MM.LT.5) GO TO 40
      TYPE=FTYPE(ISAVE)
C FORM TIME ARRAY AND FIND MAXIMUM ANDMINIMUM TIMES
      IYMD1=IYMD(IPASS)
      IHM1=IHM(IPASS)
      BMIN=IHM(IPASS)-IHM(IPASS)/100*40
      DO 10  I=IPASS,ISTOP
      FMIN(I)=FLOAT(IHM(I)-IHM(I)/100*40)-BMIN+SEC(I)/60.
   10 IF(FMIN(I).LT.0.) FMIN(I)=FMIN(I)+1440
```

```
      J=ISTOP-IPASS+1
      CALL MAXMIN(FMIN(IPASS),J,SMIN,SMAX)
      CALL PTYNUM(SMIN,SMAX,SMIN,SMAX,NX)
      KSAVE=IPASS
C FIND MAXIMUM AND MINIMUM VALUES AND ROUND
      CALL NEWMM(NAME,RESID,IPASS,ISTOP,RMAX,RMIN)
      CALL PTYNUM(RMIN,RMAX,RMIN,RMAX,NY)
C PRINT TITLE AND GRID FOR RESIDUAL VS. TIME PLOTS
      TITLE(1)=NAME
      TITLE(2)=TYPE
      CALL UGRID(SMIN,SMAX,NX,5HF4.1),1,RMIN,RMAX,NY,5HF6.1),1,0)
      CALL EDIT(IYMD1,3HI6),TITLE(5),P)
      CALL EDIT(IHM1,3HI6),TITLE(6),P)
      CALL HORLIN(TITLE,64,512,1000)
      CALL HORLIN(15HTIME IN MINUTES,15,512,0)
      IF(MOD(ISAVE,7).LE.1)
     .CALL VERLIN(24HRESIDUALS IN ARC SECONDS,24,0,512)
      IF(ISAVE.EQ.2)    CALL VERLIN(19HRESIDUALS IN METERS,19,0,512)
      IF(ISAVE.EQ.3)     CALL VERLIN(31HRESIDUALS IN CENTIMETERS/SECOND,
     .31,0,512)
C PLOT RESIDUAL VS. TIME
      DO 5 I=IPASS,ISTOP
      IF(NAME.NE.NAMEST(I)) GO TO 5
      CALL PLOT(FMIN(I),RESID(I),1,4H   *)
    5 CONTINUE
      CALL FRMADV
C FIND MAXIMUM AND MINIMUM VALUES AND ROUND
      CALL NEWMM(NAME,OBDOT,IPASS,ISTOP,MRMAX,MRMIN)
      CALL PTYNUM(MRMIN,MRMAX,MRMIN,MRMAX,NX)
C PRINT TITLE AND GRID FOR MEASUREMENT RATE VS. RESIDUAL
      CALL UGRID(MRMIN,MRMAX,NX,5HF7.0),1,RMIN,RMAX,NY,5HF6.1),1,0)
      CALL HORLIN(TITLE, 64,512,1000)
      GO TO (11,12,13,14,15,16,17,18,18,18,18,22,23,24),ISAVE
   11 CALL HORLIN(11HRT ASC RATE,11,512,0)
      GO TO 25
   12 CALL HORLIN(10HRANGE RATE,10,512,0)
      GO TO 25
   13 CALL HORLIN(11HR RATE RATE,11,512,0)
      GO TO 25
   14 CALL HORLIN(14HFREQUENCY RATE,14,512,0)
      GO TO 25
   15 CALL HORLIN(10HALPHA RATE,10,512,0)
      GO TO 25
   16 CALL HORLIN(12HX ANGLE RATE,12,512,0)
      GO TO 25
   17 CALL HORLIN(11HAZMUTH RATE,11,512,0)
      GO TO 25
   18 CALL HORLIN(16HDECLINATION RATE,16,512,0)
      GO TO 25
   22 CALL HORLIN(9HBETA RATE,9,512,0)
      GO TO 25
   23 CALL HORLIN(12HY ANGLE RATE,12,512,0)
      GO TO 25
   24 CALL HORLIN(14HELEVATION RATE,14,512,0)
   25 IF(MOD(ISAVE,7).LE.1)
     .CALL VERLIN(24HRESIDUALS IN ARC SECONDS,24,0,512)
```

```
      IF(ISAVE.EQ.2)    CALL VERLIN(19HRESIDUALS IN METERS,19,0,512)
      IF(ISAVE.EQ.3)     CALL VERLIN(31HRESIDUALS IN CENTIMETERS/SECOND,
     .31,0,512)
      DO 30 I=IPASS,ISTOP
C PLOT RESIDUALS VS. MEASUREMENT  RATE
      IF(NAME.NE.NAMEST(I)) GO TO 30
      CALL PLOT(DDDOT(I),RESID(I),1,4H   *)
   30 CONTINUE
      CALL FRMADV
      GO TO 60
   40 PRINT 50
   60 RETURN
      END
```


START
ENOUGH POINTS
NO
40
YES
FORM TIME ARRAYS
FIND MAXIMUM AND MINIMUM TIMES
FIND MAXIMUM AND MINIMUM RESIDUALS AND ROUND OFF
PRINT TITLE AND GRID FOR RESIDUAL VS. TIME PLOT
A
A
PLOT RESIDUALS VS. TIME
FIND MAXIMUM AND MINIMUM OF RATIOS AND ROUND
PRINT TITLE AND GRID FOR MEAS. RATE VS. RESIDUAL
PLOT MEASUREMENT RATE VS RESIDUAL
40
RETURN

REGANL

DESCRIPTION

REGANL is the analysis subroutine of the GEORGE PROGRAM. It uses a least squares method to determine zero set measurement biases and timing errors for each pass of data. REGANL also computes standard deviations of the errors and the noise about the fitted line. Finally it performs a randomness test of the results.

If data is to be edited, REGANL uses the results of its computations to eliminate points above a specified rejection criterion and re-computes all of the results.

NAME	REGANL
PURPOSE	PERFORMS THE REGRESSION ANALYSIS MAKES A RANDOMNESS TEST AND COMPUTES ZERO SET AND TIMING ERRORS IN PASSES OF DATA
CALLING SEQUENCE	CALL REGANL(IPASS,IPASS2,NAME,MM)

SYMBOL	TYPE	DESCRIPTION
IPASS	I	INPUT - INDEX OF BEGINNING POINT IN ARRAY
IPASS2	I	INPUT - INDEX OF END POINT IN ARRAY
NAME	DP	INPUT - STATION NAME
MM	I	INPUT - NUMBER OF POINTS IN PASS

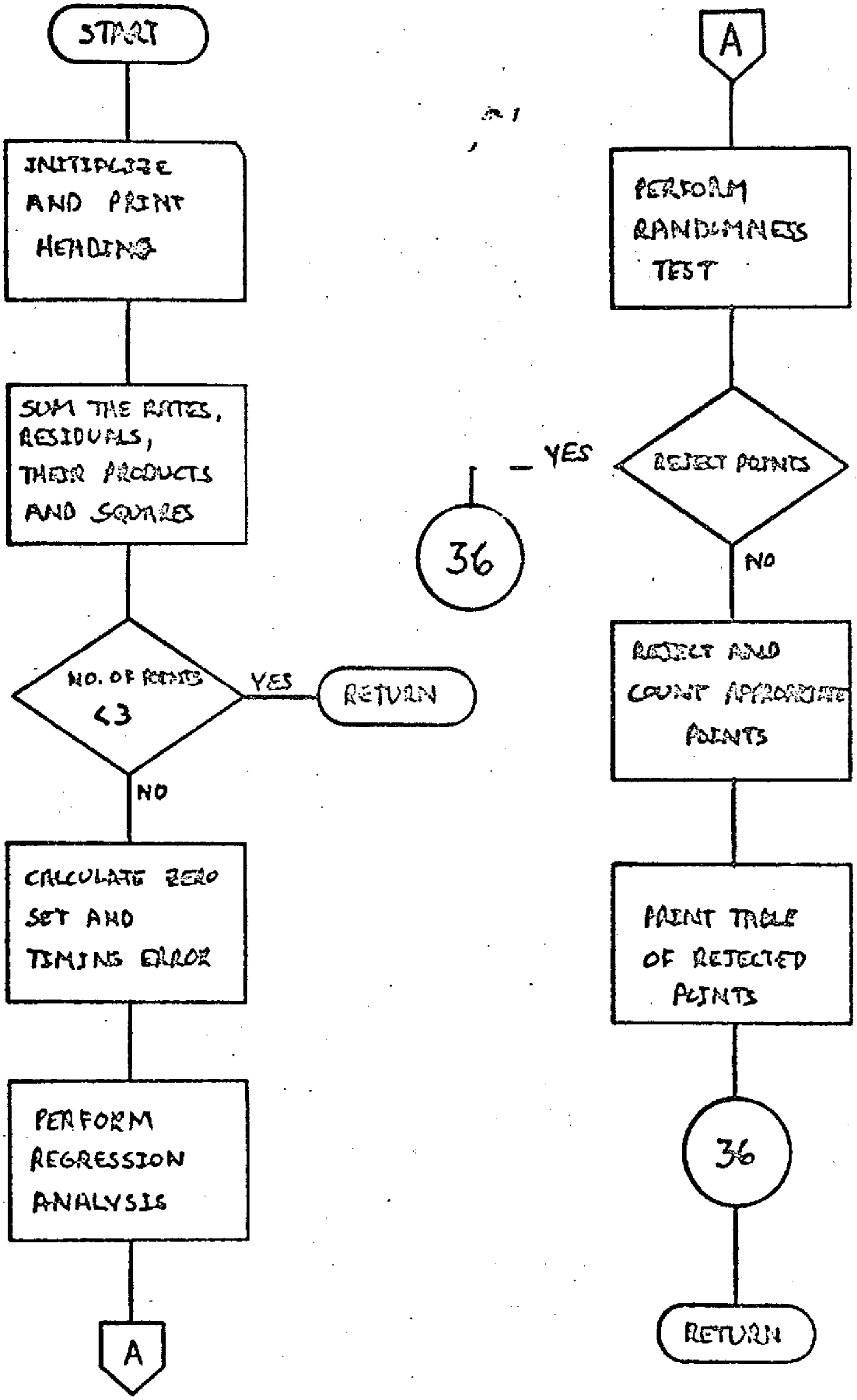
SUBROUTINES USED	NONE
COMMON BLOCKS	ARRAY CCONST
INPUT FILES	NONE
OUTPUT FILE	PRINTER
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE REGANL (IPASS,IPASS2,NAME,MM)
20007 FORMAT(1H1,16H  STATION NAME   ,A6//
     *              16H  DATE OF PASS   ,I6/
     *              16H  TIME OF PASS   ,I6/)
20008 FORMAT(1H0,51H   TIME OF DATA      MEAS.RATE    RESIDUAL  ELEVATION)
20009 FORMAT(1H ,1X,I6,1X,I4,1X,F4.1,F14.4,F10.1,F10.1)
20010 FORMAT(1H ,51H YYMMDD HHMM SS.S    (MTR/SEC)    (METERS)   (DEGREES))
20011 FORMAT(1H ,51H YYMMDD HHMM SS.S  (CM/SEC/SEC) (CM/SEC)    (DEGREES))
20012 FORMAT(1H ,51HYYMMDD HHMM SS.S (ARC SEC/SEC) (ARC SECS) (DEGREES))
20013 FORMAT(1H0,30H ZERO SET ERROR ESTIMATE        =,F10.1,39H   STANDARD
     *DEVIATION OF THE ESTIMATE =,F10.1/
     *              31H  TIMING ERROR ESTIMATE(SECS) =,F10.4,39H   STANDARD
     * DEVIATION OF THE ESTIMATE =,F10.6/
     *              31H  NOISE ABOUT THE FITTED LINE =,F10.2//
     * 30H           ANALYSIS OF VARIANCE/
     * 63H          SOURCE                 SUM OF SQUARES            DF          MEAN
     *SQUARE//
     * 20H        REGRESSION       ,F18.2,I12,F18.2/
     * 20H        RESIDUAL         ,F18.2,I12,F18.2//
     * 20H        TOTAL            ,F18.2,I12)
20014 FORMAT(1H0,30H INSUFFICIENT DATA IN PASS     )
20015 FORMAT(1H ,13H ANALYSIS OF ,A6,10H RESIDUALS)
```

```
20016 FORMAT(1H0,31H NOISE ABOUT THE LINE IS RANDOM/
     *               25H  RANDOM NORMAL DEVIATE =,G13.3)
20017 FORMAT(1H0,49H NOISE ABOUT THE LINE IS SIGNIFICANTLY NON RANDOM/
     *               25H  RANDOM NORMAL DEVIATE =,F5.1)
20018 FORMAT(1H1,31H                    REJECTED POINTS           )
      DOUBLE PRECISION   ELEV   ,FTYPE  ,NAME   ,NAMEST ,TYPE
      DOUBLE PRECISION KSTA,NET
      COMMON /ARRAY / IYMD(4000),IHM(4000),SEC(4000),ELEV(4000),
     *                RESID(4000),OBDOT(4000),ICOUNT(4000),NAMEST(4000),
     *                FTYPE(14),RATIO(4000)
      COMMON/CCONST/NET,TYPE,KSTA,A,B,ISAVE,REJECT,REJSW
      DIMENSION ICHECK(50)
      LOGICAL REJSW
C INITIALIZE VARIBLES
      IFLAG=0
   31 SIGX=0.0
      DO 103 I=1,50
  103 ICHECK(I) = 0
      SIGY = 0.0
      SIGXY = 0.0
      SIGXSQ = 0.0
      SIGYSQ = 0.0
      MM = 0
      MM1=0
      SIGX1=0.0
      SIGY1=0.0
      SIGXY1=0.0
      SIGXS1=0.0
      SIGYS1=0.0
      K=ISAVE+IFLAG*7
C PRINT HEADING
      PRINT 20007,NAME,IYMD(IPASS),IHM(IPASS)
      PRINT 20015,FTYPE(K)
      PRINT 20008
      GO TO (1,2,3,4,4,1,1),ISAVE
    1 PRINT 20012
      GO TO 4
    2 PRINT 20010
      GO TO 4
    3 PRINT 20011
    4 CONTINUE
      DO 26  L=IPASS,IPASS2
      IF(NAMEST(L ).NE.NAME) GO TO 26
      LL=L
      IF(IFLAG.EQ.1) LL=LL+2000
      IF (ISAVE.EQ.3) OBDOT(LL) = OBDOT(LL)*100.0
      IF (ISAVE.EQ.1.OR.ISAVE.EQ.7)OBDOT(LL)=OBDOT(LL)*0.1296E7/6.2831853
      MM = MM + 1
C SUM THE RATES, RESIDUALS, THEIR PRODUCTS AND THEIR SQUARES
      SIGX = SIGX + OBDOT(LL)
      SIGY = SIGY + RESID(LL)
      SIGXY = SIGXY + OBDOT(LL)*RESID(LL)
      SIGXSQ = SIGXSQ + OBDOT(LL)**2
      SIGYSQ = SIGYSQ + RESID(LL)**2
       PRINT 20019,IYMD(L),IHM(L),SEC(L),OBDOT(LL),RESID(LL),ELEV(L)
   26 CONTINUE
```

```
   34 C=FLOAT(MM+MM1)
      IFLAG2 = 0
C TEST FOR MORE THAN TWO POINTS
      IF((MM+MM1).LT.3) PRINT 20014
      IF ((MM+MM1).LT.3) RETURN
C COMPUTE TIMING ERROR
      VXY=SIGXY+SIGXY1-(SIGX+SIGX1)*(SIGY+SIGY1)/C
      VX=SIGXSQ+SIGXS1-(SIGX+SIGX1)**2/C
      VY=SIGYSQ+SIGYS1-(SIGY+SIGY1)**2/C
      B=VXY/VX
C COMPUTE ZERO SET
      A=((SIGY+SIGY1)-B*(SIGX+SIGX1))/C
      REGSS=VXY*VXY/VX
C COMPUTE THE REGRESSION MEAN
      RES=VY-REGSS
      RVAR=RES/(C-1.0)
C COMPUTE STANDARD DEVIATIONS
      BIASSD=SQRT(RVAR*(SIGXSQ+SIGXS1)/(C*VX))
      ERTIM = SQRT(RVAR/VX)
      II = 1
       JJ = MM+ MM1 -2
       NN = MM + MM1 - 1
C COMPUTE THE NOISE ABOUT THE FITTED LINE
      D = SQRT(RVAR)
      RESMS = RES  /(C-1.0)
C PRINT SUMMARY TABLE AT END OF EACH ANALYSIS
      PRINT 20013,A,BIASSD,B,ERTIM,D,REGSS,II,REGSS,RES  ,JJ,RESMS,VY,NN
      IF((MM+MM1).LT.10)GO TO 35
      DSQ=0.0
      K=IPASS2-1
      DO 32 L=IPASS,K
      IF(NAMEST(L).NE.NAME) GO TO 32
      LL=L
      IF(IFLAG.EQ.1) LL=LL+2000
      D1=RESID(LL)-A-B*OBDOT(LL)
      D2=RESID(LL+1)-A-B*OBDOT(LL)
      DSQ=DSQ+(D1-D2)**2
   32 CONTINUE
      DSQ=DSQ/((C-1.0)
C PERFORM RANDOMNESS TEST
      RND =  (DSQ/(2.0*RVAR)-1.0)/(SQRT((C-2.0)/(C*C-1.0)))
      IF (ABS(RND).LT.2.58)PRINT 20016,RND
      IF (ABS(RND).GT.2.58)PRINT 20017,RND
   35 CONTINUE
C TEST REJECTION VALUE
      IF(.NOT.REJSW)GO TO 36
      SIGX1=0.0
      SIGY1=0.0
      SIGXY1=0.0
      SIGXS1=0.0
      SIGYS1=0.0
      MM1=0.0
      K=0
      DO 33 L=IPASS,IPASS2
      IF(NAMEST(L).NE.NAME) GO TO 33
      LL=L
```

```
      IF(IFLAG.EQ.1) LL=LL+2000
      DIFF=RESID(LL)-A-B*OBDOT(LL)
C TEST ACCEPTABILITY OF POINT
      IF(ABS(DIFF).LT.REJECT) GO TO 33
      K=K+1
      IF  (K.GT.1)GO TO 14
C PRINT HEADINGS
      PRINT 20019
      PRINT 20008
      GO TO(11,12,13,14,14,11,11),ISAVE
   11 PRINT 20012
      GO TO 14
   12 PRINT 20010
      GO TO 14
   13 PRINT 20011
   14 CONTINUE
       IF (ICHECK(K).NE.LL) IFLAG2 =1
       ICHECK(K) = LL
C PRINT TABLE OF REJECTED VALUES
       PRINT 20009,IYMD(L),IHM(L),SEC(L),OBDOT(LL),RESID(LL),ELEV(L)
      SIGX1=SIGX1-OBDOT(LL)
      SIGY1=SIGY1-RESID(LL)
      SIGXY1=SIGXY1-OBDOT(LL)*RESID(LL)
      SIGXS1=SIGXS1-OBDOT(LL)**2
      SIGYS1=SIGYS1-RESID(LL)**2
      MM1=MM1-1
   33 CONTINUE
   36 CONTINUE
       IF (K.GT.0.AND.IFLAG2.EQ.1) GO TO 34
      IF(ISAVE.NE.1.AND.ISAVE.NE.7) RETURN
      IF(IFLAG.EQ.1) RETURN
      IFLAG=1
      GO TO 31
      END
```


START
INITIALIZE AND PRINT HEADING
SUM THE RATES, RESIDUALS, THEIR PRODUCTS AND SQUARES
NO. OF POINTS <3
YES
RETURN
NO
CALCULATE ZERO SET AND TIMING ERROR
PERFORM REGRESSION ANALYSIS
A
A
PERFORM RANDOMNESS TEST
REJECT POINTS
YES
36
NO
REJECT AND COUNT APPROPRIATE POINTS
PRINT TABLE OF REJECTED POINTS
36
RETURN

RYMDI

DESCRIPTION

(See EPHEMERIS TAPE GENERATOR)

1.1.3 GROUNDTRACK

INTRODUCTION

GROUNDTRACK provides geometric insights into GEODYN results by plotting the satellite groundtrack for each pass over a particular station.

The main routine GROUNDTRACK controls the type of plot (groundtrack only or groundtrack with land plots), fixes the size of the grid, reads the data required for the groundtrack requested, and makes the required calls to the Plot Package.

The subroutine CENTER centers the station position on the plotting grid. The subroutine LAND finds the required data in the WRLMAP block data to plot the land masses on the grid. WRLMAP is part of the Plot Package.

The subroutine DATIME converts minutes into days, hours and minutes. The subroutine ADDYMD is a member of GEODYN; DIFTIM is the same as subroutine DIFF in GEODYN; RYMDI is in GEORGE.

This program requires a minimum of 500K bytes of memory and uses as input one 9-track tape.

SUBROUTINE CROSS REFERENCE CHART

CALLING ROUTINES

CALLED ROUTINES	MAIN	CENTER	LAND
ADDYMD	●		
CENTER	●		
COORD	●		
DATIME	●		
DIFTIM	●		
EDIT	●		
ENDPLT	●		
GRID	●		
HORLIN	●		
LAND	●		
MAXMIN		●	
PLOT	●		●
PTYNUM	●		
RYMDI	●		
VERLIN	●		

MAIN-GROUNDTRACK

DESCRIPTION

The main program GROUNDTRACK reads and separates satellite ephemeris data into passes by station and determines from the GROUNDTRACK INPUT CARDS which data is to be plotted.

GROUNDTRACK calls CENTER to center the station position on the grid. If requested it calls LAND to determine the land masses on the grid. Finally it calls the PLOT PACKAGE routines to make the plots.

NAME	MAIN - GROUNDTRACK
PURPOSE	PLOTS SATELLITE GROUNDTRACKS FOR A SPECIFIED STATION
SUBROUTINES USED	ADDYMD CENTER DATIME DIFTIM LAND RYMDI
COMMON BLOCKS	NONE
INPUT FILES	GROUNDTRACK INPUT CARDS GROUNDTRACK TAPE
OUTPUT FILE	5 - PRINTER
RESTRICTIONS	MAXIMUM OF ONE STATION PER PLOT
REFERENCES	NONE

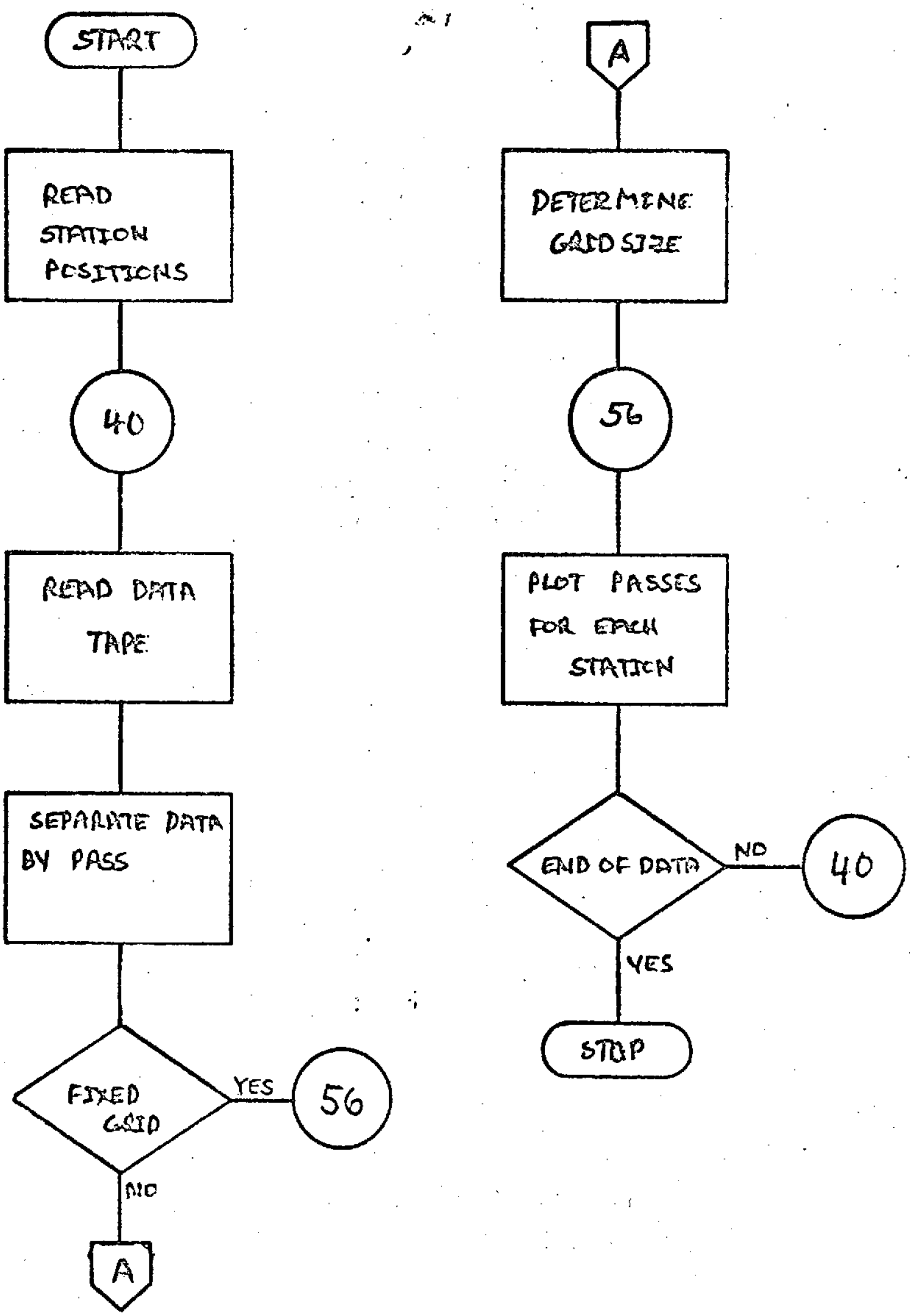
```
      DIMENSION ISTANO(10),IHM(50),ITIME(50),IPLDAY(100)
      DIMENSION IMIN(7000),LYMD(7000),LHM(7000)
      DIMENSION STALAT(10),STALON(10),VALUE(6),SATLAT(7000),
     *SATLON(7000),SATH(7000)
      REAL*8 OPT,OPTION(5),LAST,END,BLANK,STANAM(10),NAME(7000),EXTRAS
      DATA OPTION/'PLOTS ','TIME  ','GRDSET','LNDPLT','DATA  '/
      DATA LAST,BLANK/'LAST  ','      '/
      DATA END/6HEND   /
      LOGICAL*1 FT(21)/'FROM          TO       '/
      LOGICAL*1 FIXGRD,PLOTIN,TIMEIN,LANDPT,PRIME
    5 CONTINUE
      FIXGRD=.FALSE.
      LANDPT=.FALSE.
      PLOTIN=.FALSE.
      TIMEIN=.FALSE.
      IDATA=0
      NSTA=0
C READ IN STATION POSITION CARDS
   10 READ(5,1000) STANAM(NSTA+1),ISTANO(NSTA+1),STALAT(NSTA+1),
     *             STALON(NSTA+1)
      IF(STANAM(NSTA+1).EQ.END) GO TO 40
      NSTA=NSTA+1
      GO TO 10
C READ OPTIONAL GROUNDTRACK INPUT CARDS
   40 READ(5,1005) OPT,VALUE
      DO 32 I=1,5
      IF(OPT.NE.OPTION(I)) GO TO 32
      GO TO (33,34,36,37,46),I
   32 CONTINUE
C ERRONEOUS INPUT CARD
      WRITE(6,1003) OPT
      GO TO 40
C SET SWITCH FOR SC4020 TAPE
```

```
   33 CONTINUE
      PLOTIN=.TRUE.
      I4020=4
      IF(VALUE(1).GT.0.) I4020=6
      GO TO 40
C SET TIME PARAMETERS FOR A DATA PERIOD REQUESTED
   34 CONTINUE
      TIMEIN=.TRUE.
      IYMDST=VALUE(1)+.5
      NHMST=VALUE(2)+.5
      IYMDEN=VALUE(3)+.5
      NHMEN=VALUE(4)+.5
      GO TO 40
C SET GRID PARAMETERS FOR A FIXED GRID
   36 FIXGRD=.TRUE.
      SATLN1=VALUE(1)
      SATLN2=VALUE(2)
      INTY=VALUE(3)
      SATLT1=VALUE(4)
      SATLT2=VALUE(5)
      INTX=VALUE(6)
      GO TO 40
C SET WORLD MAP OPTION
   37 CONTINUE
      LANDPT=.TRUE.
         CALL WFLMAP
      GO TO 40
   40 CONTINUE
      CALL PLOTST(I4020,.TRUE.)
      CALL FRMADV
      WRITE(6,1007)  STANAM(NSTA)
      IF(LANDPT) WRITE(6,1011)
      IF(FIXGRD) WRITE(6,1012) SATLT1,SATLT2,INTX,SATLN1,SATLN2,INTY
      IF(TIMEIN) WRITE(6,1013) IYMDST,NHMST,IYMDEN,NHMEN
      IF(PLOTIN.AND.I4020.EQ.4) WRITE(6,1014)
      IF(PLOTIN.AND.I4020.EQ.6) WRITE(6,1015)
C READ IN DATA TAPE
   30 READ(11,1001,END=35)LYMD(IDATA+1),LHM(IDATA+1),NAME(IDATA+1),
     *            SATLAT(IDATA+1),SATLON(IDATA+1),SATH(IDATA+1)
      IF(NAME(IDATA+1).NE.STANAM(NSTA)) GO TO 30
      IDATA=IDATA+1
      CALL DIFTIM(LYMD(1),0,LYMD(IDATA),LHM(IDATA),LDAY,LMIN)
      IMIN(IDATA)=LDAY*1440+LMIN
      GO TO 30
C STORE STATION DATA AND TEST FOR GREENWICH MERIDIAN(PRIME)
   35 REWIND 11
      CALL DIFTIM(LYMD(1),0,IYMDST,NHMST,JDAY,JMIN)
      ISTART=JDAY*1440+JMIN
      CALL DIFTIM(LYMD(1),0,IYMDEN,NHMEN,MDAY,MMIN)
      IEND=MDAY*1440+MMIN
      DO 70 ISTA=1,NSTA
      PRIME=STALON(ISTA).LT.45..OR.STALON(ISTA).GT.315.
      IF(PRIME.AND.STALON(ISTA).GT.180.)STALON(ISTA)=360.-STALON(ISTA)
      NPOINT=0
      NPASS=0
      JHM=-60
```

```
C SEPARATE STATION DATA
      DO 50 IP=1,IDATA
      IF(IMIN(IP).LT.ISTART) GO TO 50
      IF(IMIN(IP).GT.IEND) GO TO 55
      NPOINT=NPOINT+1
      IF(PRIME.AND.SATLON(NPOINT).GT.180.)
     .   SATLON(NPOINT)=360.-SATLON(NPOINT)
C TEST FOR NEW PASS
      IF(IMIN(IP)-JHM.LT.50) GO TO 45
      NPASS=NPASS+1
C SAVE LAST DATA POINT
      IHM(NPASS)=LHM(IP)
      ITIME(NPASS)=NPOINT
      NYMD=LYMD(IP)
      NYRM=NYMD/100
      IDAYNO=NYMD-NYRM*100
      IPLDAY(NPASS)=IDAYNO
      IF(NPASS.EQ.1)  WRITE(6,1002) STANAM(ISTA),IYMDST,NHMST,IYMDEN,
     .NHMEN
      JHM=IMIN(IP)
      CALL DATIME(IMIN(IP),IHRMIN,IDAY)
      JJHM=IHRMIN
      IYMD=LYMD(1)
      CALL ADDYMD(IYMD,IDAY)
      LINES=0
      GO TO 43
   45 IF(MOD(LINES,5).EQ.0) WRITE(6,1010)
      WRITE(6,1004) LHM(IP),SATLAT(IP),SATLON(IP),SATH(IP)
      IF(LINES.EQ.40) GO TO 42
      LINES=LINES+1
      GO TO 50
   42 WRITE(6,1006) STANAM(ISTA)
   43 LINES=0
      WRITE(6,1003) NPASS,IYMD,JJHM
   50 CONTINUE
C PLACE STATION IN MIDDLE OF GRID
   55 CONTINUE
      IF(NPOINT.EQ.0) GO TO 70
      ITIME(NPASS+1)=NPOINT+1
      IF(FIXGRD) GO TO 56
      CALL CENTER(STALON(ISTA),SATLON,NPOINT,SATMIN,SATMAX)
      CALL PTYNLM(SATMIN,SATMAX,SMIN,SMAX,NY)
      CALL CENTER(STALAT(ISTA),SATLAT,NPOINT,SATLMN,SATLMX)
      CALL PTYNLM(SATLMN,SATLMX,PMIN,PMAX,NX)
      CALL GRID(SMIN,SMAX,NY,'I3)',1,PMIN,PMAX,NX,'I3)',1,0)
      IF(LANDPT) CALL LAND(SMIN,SMAX,PMIN,PMAX)
      WRITE(6,1009) SMIN,SMAX,NY,PMIN,PMAX,NX
      GO TO 57
C USING FIXED GRID METHOD
   56 CONTINUE
      CALL GRID(SATLN1,SATLN2,INTY,'I3)',1,SATLT1,SATLT2,INTX,'I3)',1,0)
      IF(LANDPT) CALL LAND(SATLN1,SATLN2,SATLT1,SATLT2)
   57 CONTINUE
C CENTER AND LABEL GRID
      CALL EDIT(IYMDST,3HI6),IDATE,P)
      CALL HORLIN(IDATE,6,512,1016)
```

```
      CALL HORLIN(STANAM(ISTA),6,512,1000)
      CALL HORLIN(17HPLOTTED PERIOD IS,17,300,1016)
      CALL HORLIN(26H+ DENOTES STATION POSITION,26,300,1000)
      CALL EDIT(IYMDST,3HI6),FT(6),P)
      CALL EDIT(IYMDEN,3HI6),FT(16),F)
      CALL HORLIN(FT,21,900,1000)
      CALL HORLIN(9HLONGITUDE,9,512,0)
      CALL VERLIN(8HLATITUDE,8,0,512)
      CALL PLOT(STALON(ISTA),STALAT(ISTA),1,4H   +)
C PLOT PASS
      DO 60 IP=1,NPASS
      IFPT=ITIME(IP)
      CALL EDIT(IHM(IP),3HI4),JJHM,GARB)
      CALL EDIT(IPLDAY(IP),3HI2),NDAY,NOTE)
      CALL PLOT(SATLON(IFPT),SATLAT(IFPT),ITIME(IP+1)-IFPT,4H   S)
      CALL PLOT(SATLON(IFPT),SATLAT(IFPT),1,4H   *)
      CALL CLORD(SATLON(IFPT),SATLAT(IFPT),IX,IY)
      CALL CLORD(SATLON(IFPT+1),SATLAT(IFPT+1),IX1,IY1)
      DX=IX-IX1
      DY=IY-IY1
      R=SQRT(DX**2+DY**2)
      IX=24.*DX/R+IX
      IY=24.*DY/R+IY
      CALL HORLIN(JJHM,4,IX,IY-8)
      CALL HORLIN(NDAY,2,IX,IY+6)
 60   CONTINUE
      CALL FRMADV
 70   CONTINUE
C TEST FOR END OF DATA
 72   READ(5,1005) EXTRAS
      IF(EXTRAS.EQ.BLANK) GO TO 5
      IF(EXTRAS.EQ.LAST) GO TO 75
      GO TO 72
 75   CONTINUE
      CALL ENDPLT
      STOP
 1000 FORMAT(A6,I4,F3.0,12X,F3.0)
 1001 FORMAT(I6,2X,I4,11X,A6,2X,F15.8,2X,F15.8,2X,F15.5)
 1002 FORMAT(1H1,19X,'STATION NAME ',A6,/,19X,'START DATE    ',I6,/,
     .19X,'START TIME    ',I4,/,19X,'END DATE      ',I6,/,19X,
     .'END TIME      ',I4,/)
 1003 FORMAT(1H1,19X,'PASS NUMBER  ',I3,/,19X,'DATE OF PASS ',I6,/,19X,
     .'TIME OF PASS ',I4,//,6X,'TIME',17X,'SATELLITE',/,6X,'HOUR',5X,
     .'LATITUDE',4X,'LONGITUDE',5X,'HEIGHT',/,5X,'MINUTE',4X,'(DEGREES)'
     .,3X,'(DEGREES)',4X,'(METERS)',/)
 1004 FORMAT(1H ,5X,I4,5X,F9.3,3X,F9.3,3X,F10.2)
 1005 FORMAT(A6,4X,6(F10.0))
 1006 FORMAT(1H0,19X,A6,3X,6HCONT'D,///)
 1007 FORMAT(1H1,15X,'OPTIONS REQUESTED ARE AS FOLLOWS',//,
     .15X,' STATION NAME ',A6,//)
 1008 FORMAT(1H1,1X,'ILLEGAL OPTION CARD',A6,3X,
     .'IGNORED REMAINING OPTIONS , EXECUTION CONTINUING')
 1009 FORMAT(1H1,///,13X,'BASIC  GRID  SIZE',/,6X,'LONGITDE VALUE',3X,
     .'LATITUDE  VALUE',/,6X,'HIGH   LOW  INT   HIGH   LOW  INT',/,
     .5X,F5.1,2X,F5.1,2X,I2,3X,F5.1,2X,F5.1,3X,I2)
 1010 FORMAT(1X)
```

```
1011 FORMAT(1H ,10X,'WORLD-MAP OVERLAY PLOT',/)
1012 FORMAT(1H ,15X,'GRID SET WITH THE VALUES',/,10X,'MINIMUM LATITUDE
    .VALUE   ',F5.1,/,10X,'MAXIMUM LATITUDE VALUE   ',F5.1,/,10X,'NUMBER
    .OF LATITUDE INTERVALS   ',I3,/,10X,'MINIMUM LONGITUDE VALUE   ',F5.1
    .,/,10X,'MAXIMUM LONGITUDE VALUE   ',F5.1,/,10X,'NUMBER OF LONGITUDE
    . INTERVALS ',I3,/)
1013 FORMAT(1H ,15X,'TIME INTERVAL TO BE PLOTTED',/,
    .10X,'START DATE   ',I6,10X,'START TIME   ',I4,/,
    .10X,'  END DATE   ',I6,10X,'  END TIME   ',I4,/)
1014 FORMAT(1H ,10X,'PRINTER PLOT ONLY',/)
1015 FORMAT(1H ,10X,'PRINTER AND SC4020 PLOT',/)
     END.
```


START
READ STATION POSITIONS
40
READ DATA TAPE
SEPARATE DATA BY PASS
FIXED GRID
YES
56
NO
A
A
DETERMINE GRID SIZE
56
PLOT PASSES FOR EACH STATION
END OF DATA
NO
40
YES
STOP

CENTER

DESCRIPTION

CENTER calls MAXMIN to determine the center of the grid at which point it places the station.

NAME	CENTER
PURPOSE	PLACES A GIVEN POINT IN THE CENTER OF A GRID
CALLING SEQUENCE	CALL CENTER(CENPT,ARRAY,N,CMIN,CMAX)
SYMBOL TYPE	DESCRIPTION
CENPT	INPUT - POINT TO BE CENTERED
ARRAY	INPUT - ARRAY OF POINTS TO BE PLOTTED
N	INPUT - NUMBER OF ENTRIES IN THE ARRAY
CMIN	OUTPUT - MINIMUM VALUE OF THE PLOTTING SCALE
CMAX	OUTPUT - MAXIMUM VALUE OF THE PLOTTING SCALE
SUBROUTINE USED	MAXMIN
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE CENTER(CENPT,ARRAY,N,CMIN,CMAX)                CENT  34
      DIMENSION ARRAY(N)                                        CENT  35
C COMPUTE MAXIMA AND MINIMA                                     CENT  36
      CALL MAXMIN(ARRAY,N,CMIN,CMAX)                            CENT  37
      FLENTH=AMAX1(CMAX-CENPT,CENPT-CMIN)                       CENT  38
C CENTER THE POINT                                              CENT  39
      CMIN=CENPT-FLENTH                                         CENT  40
      CMAX=CENPT+FLENTH                                         CENT  41
      RETURN                                                    CENT  42
      END                                                       CENT  43
```

DATIME

DESCRIPTION

Subroutine DATIME converts a given number of minutes to days, hours and minutes.

NAME		DATIME
PURPOSE		CONVERTS MINUTES TO DAYS AND HOURS AND MINUTES
CALLING SEQUENCE		CALL DATIME(MIN,IHRMIN,IDAY)
SYMBOL	TYPE	DESCRIPTION
MIN	R	INPUT - MINUTES TO BE CONVERTED
IHRMIN	R	OUTPUT - NUMBER OF HOURS AND MINUTES AFTER CONVERSION
IDAY	R	OUTPUT - NUMBER OF DAYS AFTER CONVERSION
SUBROUTINES USED		NONE
COMMON BLOCKS		NONE
INPUT FILES		NONE
OUTPUT FILES		NONE
RESTRICTIONS		NONE
REFERENCES		NONE

```
      SUBROUTINE DATIME(MIN,IHRMIN,IDAY)
C CONVERT MINUTES TO DAYS
      IDAY=MIN/1440
      IF(IDAY.GE.1) GO TO 20
      MIN1=MIN
      GO TO 10
   20 MIN1=MIN-IDAY*1440
   10 CONTINUE
C EXTRACT HOURS
      IHR=MIN1/60
      LHR=MIN1-IHR*60
C COMBINE HOURS AND MINUTES
      IHRMIN=IHR*100+LHR
      RETURN
      END
```

LAND

DESCRIPTION

LAND determines the points which make up the land masses on the grid. It references the block data routine WRLMAP through the entry to EARTH to obtain the data.

NAME		LAND
ENTRY POINT		PURPOSE
LAND		PLOTS LAND MASSES ON GRID
EARTH		INITIALIZATION OF ARGUMENTS IN CALLING SEQUENCE
CALLING SEQUENCE		CALL LAND(LONG1,LONG2,LAT1,LAT2)
SYMBOL	TYPE	DESCRIPTION
LONG1	R	INPUT - STARTING LONGITUDINAL BOUNDARY OF THE REGION
LONG2	R	INPUT - STOPPING LONGITUDINAL BOUNDARY OF THE REGION
LAT1	R	INPUT - STARTING LATITUDINAL BOUNDARY OF THE REGION
LAT2	R	INPUT - STOPPING LATITUDINAL BOUNDARY OF THE REGION
CALLING SEQUENCE		CALL EARTH(NE,NBOD,A,B)
SYMBOL	TYPE	DESCRIPTION
NE	I	INPUT-OUTPUT - NUMBER OF BODIES OF LAND
NBOD	I	INPUT-OUTPUT - NUMBER OF VECTOR POINTS IN EACH BODY OF LAND
A	R	INPUT-OUTPUT - LONGITUDE OF THE VECTOR POINTS
B	R	INPUT-OUTPUT - LATITUDE OF THE VECTOR POINTS
SUBROUTINE USED		PLOT
COMMON BLOCKS		NONE
INPUT FILES		NONE
OUTPUT FILES		NONE
RESTRICTIONS		NONE
REFERENCES		NONE

```
      SUBROUTINE LAND(LONG1,LONG2,LAT1,LAT2)
      REAL LONG1,LONG2,LAT1,LAT2,A(1),B(1)
      LOGICAL NEG
      INTEGER NBOD(1)
      NEG=LONG1.LT.0.
      IP=0
C FIND FIRST POINT IN DATA TO BE ON THE GRID
```

```
      DO 30 I=1,NE
      N=0
      NP=NBOD(I)
      DO 20 J=1,NP
      IP=IP+1
C IF NEGATIVE INTERVAL, SUBTRACT FROM 360
      IF(NEG.AND.A(IP).GT.180.) A(IP)=360.-A(IP)
C TEST IF POINTS ARE ON GRID
      IF(A(IP).GT.LONG1.AND.A(IP).LT.LONG2.AND.B(IP).GT.LAT1.AND.B(IP).
     .LT.LAT2) GO TO 10
C PLOT POINTS
      IF(N.GT.0) CALL PLOT(A(IP-N),B(IP-N),N,'     ')
      N=0
      GO TO 20
   10 N=N+1
   20 CONTINUE
C IF PLOT POINTS EXTEND BEYOND INTERVAL, PLOT TO EXTREMITIES
      IF(N.GT.0) CALL PLOT(A(IP-N+1),B(IP-N+1),N,'     ')
   30 CONTINUE
      IF(.NOT.NEG) RETURN
C REVERSE PLOTTING
      IP=0
      DO 40 I=1,NE
      K=NBOD(I)
      DO 40 J=1,K
      IP=IP+1
      IF(A(IP).LT.0.) A(IP)=360.-A(IP)
   40 CONTINUE
      ENTRY EARTH(NE,NBOD,A,B)
      RETURN
      END
```

WRLMAP

DESCRIPTION

WRLMAP contains all the data needed to plot land masses anywhere on the earth.

NAME	WRLMAP
PURPOSE	TO CALL EARTH WITH THE WORLD MAP DATA
CALLING SEQUENCE	CALL WRLMAP
SUBROUTINE USED	EARTH
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE WRLMAP
      REAL*4 A 1(4908),A 2(139),A 3(139),A 4(139),A 5(139),A 6(139),
     *       A 7(139),A 8(139),A 9(139),A10(139),A11(139),A12(139),
     *       A13(139),A14(139),A15(139),A16(139),A17(139),A18(139),
     *       A19(139),A20(139),A21(139),A22(139),A23(139),A24(139),
     *       A25(139),A26(139),A27(139),A28(139),A29(139),A30(139),
     *       A31(139),A32(139),A33(139),A34(139),A35(139),A36( 43),
     *       B 1(4908),B 2(139),B 3(139),B 4(139),B 5(139),B 6(139),
     *       B 7(139),B 8(139),B 9(139),B10(139),B11(139),B12(139),
     *       B13(139),B14(139),B15(139),B16(139),B17(139),B18(139),
     *       B19(139),B20(139),B21(139),B22(139),B23(139),B24(139),
     *       B25(139),B26(139),B27(139),B28(139),B29(139),B30(139),
     *       B31(139),B32(139),B33(139),B34(139),B35(139),B36( 43)
      INTEGER C(110)
      EQUIVALENCE (A 2(1),A1(140)),(A 3(1),A1(279)),(A 4(1),A1(418)),
     *   (A 5(1),A1(557)),(A 6(1),A1(696)),(A 7(1),A1(835)),
     *   (A 8(1),A1(974)),(A 9(1),A1(1113)),(A10(1),A1(1252)),
     *   (A11(1),A1(1391)),(A12(1),A1(1530)),(A13(1),A1(1669)),
     *   (A14(1),A1(1808)),(A15(1),A1(1947)),(A16(1),A1(2086)),
     *   (A17(1),A1(2225)),(A18(1),A1(2364)),(A19(1),A1(2503)),
     *   (A20(1),A1(2642)),(A21(1),A1(2781)),(A22(1),A1(2920)),
     *   (A23(1),A1(3059)),(A24(1),A1(3198)),(A25(1),A1(3337)),
     *   (A26(1),A1(3476)),(A27(1),A1(3615)),(A28(1),A1(3754)),
     *   (A29(1),A1(3893)),(A30(1),A1(4032)),(A31(1),A1(4171)),
     *   (A32(1),A1(4310)),(A33(1),A1(4449)),(A34(1),A1(4588)),
     *   (A35(1),A1(4727)),(A36(1),A1(4866))
      EQUIVALENCE (B 2(1),B1(140)),(B 3(1),B1(279)),(B 4(1),B1(418)),
     *   (B 5(1),B1(557)),(B 6(1),B1(696)),(B 7(1),B1(835)),
     *   (B 8(1),B1(974)),(B 9(1),B1(1113)),(B10(1),B1(1252)),
     *   (B11(1),B1(1391)),(B12(1),B1(1530)),(B13(1),B1(1669)),
     *   (B14(1),B1(1808)),(B15(1),B1(1947)),(B16(1),B1(2086)),
     *   (B17(1),B1(2225)),(B18(1),B1(2364)),(B19(1),B1(2503)),
     *   (B20(1),B1(2642)),(B21(1),B1(2781)),(B22(1),B1(2920)),
     *   (B23(1),B1(3059)),(B24(1),B1(3198)),(B25(1),B1(3337)),
```

```
     *    (B26(1),B1(3476)),(B27(1),B1(3615)),(B28(1),B1(3754)),
     *    (B29(1),B1(3893)),(B30(1),B1(4032)),(B31(1),B1(4171)),
     *    (B32(1),B1(4310)),(B33(1),B1(4449)),(B34(1),B1(4588)),
     *    (B35(1),B1(4727)),(B36(1),B1(4866))
C NUMBER OF DISTINCT BODIES (CLOSED CONTOURS)
      DATA N/110/
C THE DATA IN EACH "A" ARRAY ARE THE LONGITUDE COORDINATES FOR THE
C VECTOR SET DEFINING A BODY. THE DATA IN EACH "B" ARRAY ARE THE
C CORRESPONDING LATITUDE COORDINATES IN THE VECTOR SET.
      DATA A 1/  278.41,  278.71,  278.34,  277.97,  278.34,  277.23,
     *  276.71,  275.53,  274.85,  274.79,  275.16,  275.23,  276.41,
     *  276.71,  277.67,  278.12,  279.23,  279.45,  278.71,  277.82,
     *  276.78,  276.04,  274.86,  274.05,  273.01,  273.90,  273.97,
     *  274.19,  273.97,  274.56,  273.60,  272.94,  273.31,  273.01,
     *  272.27,  271.31,  270.57,  269.97,  268.94,  268.13,  268.87,
     *  269.46,  269.90,  270.42,  271.09,  271.97,  272.34,  271.75,
     *  270.80,  270.12,  269.09,  268.27,  267.75,  267.46,  266.87,
     *  266.50,  266.94,  266.72,  266.50,  265.76,  265.61,  264.94,
     *  264.87,  264.28,  264.28,  264.94,  265.24,  266.27,  267.24,
     *  267.16,  268.20,  269.38,  270.49,  271.31,  272.71,  274.27,
     *  275.08,  276.56,  277.30,  276.93,  276.86,  277.75,  277.67,
     *  278.63,  279.08,  278.86,  279.00,  279.75,  280.12,  280.71,
     *  280.12,  280.26,  279.67,  279.89,  280.78,  282.41,  282.48,
     *  281.67,  281.89,  280.41,  281.00,  281.45,  281.45,  281.23,
     *  281.60,  281.82,  283.00,  283.59,  284.70,  285.37,  286.18,
     *  287.52,  287.59,  288.33,  289.66,  289.07,  289.51,  289.59,
     *  290.28,  290.63,  291.29,  292.77,  293.73,  293.65,  293.73,
     *  294.47,  295.36,  295.29,  296.18,  295.14,  295.58,  296.40,
     *  296.92,  296.62,  297.58,  297.21,  297.66,  298.47,  298.77/
      DATA B 1/   68.98,   68.59,   68.20,   67.57,   66.79,   66.35,
     *   66.01,   66.54,   66.35,   65.81,   65.71,   65.18,   65.13,
     *   64.74,   64.64,   64.01,   63.81,   63.18,   63.18,   63.23,
     *   63.71,   63.32,   62.79,   63.13,   63.28,   63.52,   64.15,
     *   64.49,   64.04,   65.42,   66.10,   65.96,   65.76,   65.27,
     *   65.67,   65.67,   65.71,   65.57,   65.96,   65.32,   65.47,
     *   65.37,   65.13,   65.32,   65.08,   65.08,   64.40,   63.57,
     *   63.86,   63.57,   63.76,   62.93,   62.84,   63.28,   63.57,
     *   63.13,   62.69,   62.30,   61.71,   61.32,   60.93,   60.93,
     *   60.15,   59.18,   58.64,   58.79,   58.45,   58.74,   59.01,
     *   56.59,   56.74,   56.06,   55.47,   55.33,   54.69,   54.79,
     *   54.99,   54.45,   53.62,   52.99,   52.30,   51.67,   51.23,
     *   50.89,   50.99,   51.13,   51.52,   51.23,   51.52,   51.86,
     *   52.69,   53.38,   53.96,   54.45,   54.74,   55.86,   56.30,
     *   56.98,   57.72,   58.55,   59.13,   59.47,   60.11,   61.19,
     *   62.15,   62.54,   62.50,   62.20,   62.35,   62.01,   61.96,
     *   61.96,   61.03,   60.84,   60.45,   60.15,   59.42,   58.55,
     *   58.35,   57.42,   57.76,   57.67,   57.67,   58.06,   58.84,
     *   59.81,   59.72,   59.37,   58.55,   57.67,   57.47,   57.72,
     *   57.18,   56.90,   56.35,   56.30,   55.86,   55.62,   55.23/
      DATA A 2/  299.58,  300.76,  301.50,  299.88,  300.32,  301.28,
     *  302.17,  303.13,  303.65,  303.13,  303.06,  301.21,  300.02,
     *  299.51,  298.91,  298.54,  297.51,  296.69,  295.14,  296.10,
     *  297.51,  296.55,  295.58,  294.55,  294.99,  294.55,  293.95,
     *  293.07,  294.33,  294.10,  295.14,  295.21,  296.25,  296.40,
     *  295.21,  295.56,  296.77,  297.80,  298.99,  295.14,  294.25,
     *  292.77,  293.88,  294.70,  293.36,  292.10,  292.11,  290.92,
```

```
     .  289.81,  289.07,  287.96,  289.26,  287.74,  288.95,  286.92,   WRLM 112
     .  286.18,  286.04,  285.37,  284.22,  284.63,  285.30,  284.85,   WRLM 113
     .  284.26,  283.30,  283.59,  284.09,  284.19,  283.45,  283.37,   WRLM 114
     .  283.06,  283.98,  282.41,  281.89,  282.19,  281.89,  282.63,   WRLM 115
     .  283.15,  282.63,  282.15,  282.34,  283.00,  282.34,  280.71,   WRLM 116
     .  280.04,  279.15,  277.59,  278.04,  277.99,  278.49,  278.31,   WRLM 117
     .  278.71,  279.38,  278.41,  277.33,  276.64,  277.75,  276.41,   WRLM 118
     .  276.78,  276.04,  275.23,  274.56,  273.31,  273.01,  272.27,   WRLM 119
     .  271.09,  270.64,  269.83,  269.31,  270.05,  270.64,  270.27,   WRLM 120
     .  269.53,  268.42,  266.72,  266.05,  265.16,  264.05,  263.75,   WRLM 121
     .  262.43,  262.43,  261.61,  261.76,  261.54,  261.91,  261.17,   WRLM 122
     .  261.54,  260.95,  261.76,  261.98,  262.87,  263.31,  263.91,   WRLM 123
     .  265.39,  265.16,  265.43,  266.79,  267.96,  267.83,  268.42/   WRLM 124
      DATA B 2/   54.74,   54.50,   54.16,   53.52,   53.28,   53.52,   WRLM 125
     .   53.18,   53.28,   52.25,   51.96,   51.13,   51.38,   50.99,   WRLM 126
     .   49.82,   50.06,   49.72,   50.16,   50.21,   50.30,   49.62,   WRLM 127
     .   49.13,   48.99,   49.28,   49.18,   48.65,   48.26,   43.35,   WRLM 128
     .   47.91,   47.62,   46.74,   46.74,   46.01,   46.50,   45.82,   WRLM 129
     .   45.67,   45.33,   45.62,   45.28,   44.74,   44.40,   43.38,   WRLM 130
     .   43.77,   44.60,   45.20,   45.23,   45.18,   44.31,   44.45,   WRLM 131
     .   43.33,   43.62,   43.28,   42.79,   42.31,   41.28,   41.38,   WRLM 132
     .   41.18,   40.35,   40.99,   41.53,   40.40,   39.77,   39.04,   WRLM 133
     .   39.33,   39.82,   38.99,   38.11,   37.48,   37.23,   38.01,   WRLM 134
     .   38.50,   39.48,   38.84,   38.50,   37.37,   37.28,   36.89,   WRLM 135
     .   36.21,   35.97,   35.56,   35.23,   34.70,   34.50,   33.43,   WRLM 136
     .   32.89,   32.41,   31.43,   30.41,   29.33,   28.95,   28.11,   WRLM 137
     .   27.33,   26.51,   25.68,   25.97,   27.04,   27.24,   27.92,   WRLM 138
     .   28.65,   29.09,   29.82,   29.63,   29.77,   30.36,   30.21,   WRLM 139
     .   30.65,   30.46,   30.60,   30.16,   29.48,   28.85,   28.65,   WRLM 140
     .   28.89,   29.24,   29.33,   29.25,   29.58,   29.63,   28.85,   WRLM 141
     .   28.65,   28.25,   27.92,   27.04,   26.46,   25.87,   24.95,   WRLM 142
     .   24.02,   23.04,   21.92,   20.80,   19.85,   19.09,   18.46,   WRLM 143
     .   18.41,   17.97,   17.55,   17.78,   18.26,   18.90,   19.19/   WRLM 144
      DATA A 3/  268.42,  268.94,  270.05,  270.49,  271.23,  272.05,   WRLM 145
     .  272.71,  271.97,  272.27,  271.63,  271.90,  271.83,  270.94,   WRLM 146
     .  270.86,  271.09,  270.57,  272.12,  272.57,  273.90,  275.23,   WRLM 147
     .  276.19,  275.60,  275.97,  275.23,  275.90,  275.01,  275.75,   WRLM 148
     .  277.01,  277.45,  278.26,  278.93,  280.41,  281.82,  283.08,   WRLM 149
     .  283.15,  283.96,  283.96,  284.93,  285.89,  286.55,  287.59,   WRLM 150
     .  286.92,  287.00,  286.78,  287.22,  286.92,  287.37,  288.40,   WRLM 151
     .  287.81,  288.48,  289.44,  289.51,  289.14,  289.29,  290.77,   WRLM 152
     .  289.96,  289.81,  290.85,  291.96,  292.99,  294.18,  294.62,   WRLM 153
     .  295.21,  296.10,  296.99,  296.69,  297.29,  296.99,  297.88,   WRLM 154
     .  299.21,  298.84,  300.25,  300.02,  300.99,  302.25,  301.02,   WRLM 155
     .  304.02,  304.47,  306.24,  306.54,  307.72,  307.94,  308.46,   WRLM 156
     .  307.80,  308.68,  308.54,  309.42,  310.61,  310.98,  312.01,   WRLM 157
     .  312.98,  314.68,  314.83,  314.53,  314.53,  315.64,  315.79,   WRLM 158
     .  317.94,  319.93,  320.67,  321.71,  322.08,  323.34,  323.63,   WRLM 159
     .  324.00,  324.30,  323.71,  323.63,  322.83,  321.93,  321.64,   WRLM 160
     .  319.93,  319.56,  319.42,  319.79,  319.42,  319.86,  319.35,   WRLM 161
     .  319.34,  318.60,  318.38,  317.64,  317.42,  316.75,  316.46,   WRLM 162
     .  315.05,  314.16,  312.31,  311.42,  310.24,  310.16,  309.79,   WRLM 163
     .  310.24,  309.50,  308.98,  308.61,  307.57,  306.46,  306.32/   WRLM 164
      DATA B 3/   19.95,   21.14,   21.14,   21.38,   21.29,   21.25,   WRLM 165
     .   20.80,   20.56,   20.12,   19.63,   18.65,   18.17,   18.17,   WRLM 166
     .   17.14,   16.51,   15.65,   15.92,   15.73,   15.53,   15.68,   WRLM 167
```

```
     .   14.55,   13.78,   12.95,   12.46,   11.87,   11.15,   10.66,      WRLM 168
     .    9.44,    9.68,    9.29,    9.65,    9.49,    8.85,    9.29,      WRLM 169
     .   10.17,   10.45,   10.95,   11.29,   11.29,   12.47,   11.75,      WRLM 170
     .   11.24,   10.90,   10.36,   10.17,    9.55,    9.39,    9.73,      WRLM 171
     .   10.61,   11.29,   11.78,   11.48,   11.92,   12.51,   12.12,      WRLM 172
     .   11.55,   11.05,   10.27,   10.41,   10.02,    9.55,   10.61,      WRLM 173
     .   10.31,   10.45,   10.41,    9.65,    9.58,    8.61,    9.05,      WRLM 174
     .    6.41,    6.15,    7.29,    6.61,    6.27,    5.63,    5.35,      WRLM 175
     .    5.83,    5.34,    5.34,    4.41,    3.95,    3.39,    3.55,      WRLM 176
     .    2.37,    1.88,    1.25,    0.41,    0.37,   -0.66,   -0.61,      WRLM 177
     .   -0.95,   -1.29,   -1.73,   -1.88,   -2.32,   -1.95,   -2.22,      WRLM 178
     .   -2.37,   -3.39,   -3.97,   -4.75,   -5.24,   -5.24,   -5.97,      WRLM 179
     .   -7.00,   -7.95,   -8.65,   -9.53,  -10.22,  -10.90,  -12.15,      WRLM 180
     .  -12.22,  -13.00,  -14.17,  -15.14,  -12.65,  -17.04,  -17.82,      WRLM 181
     .  -18.85,  -19.48,  -20.05,  -20.46,  -21.34,  -21.48,  -22.21,      WRLM 182
     .  -22.35,  -22.31,  -23.48,  -24.55,  -24.65,  -25.75,  -26.41,      WRLM 183
     .  -27.38,  -27.92,  -28.60,  -29.75,  -30.60,  -31.09,  -31.67/      WRLM 184
      DATA A 4/ 305.35,  304.76,  303.73,  302.54,  301.83,  300.84,      WRLM 185
     .  300.76,  300.17,  301.06,  300.99,  302.54,  301.65,  301.50,      WRLM 186
     .  300.39,  299.43,  297.95,  296.77,  296.03,  296.52,  296.25,      WRLM 187
     .  294.92,  293.96,  293.66,  295.14,  293.96,  293.73,  293.29,      WRLM 188
     .  293.51,  293.36,  292.18,  292.03,  291.07,  291.59,  292.85,      WRLM 189
     .  292.70,  291.37,  291.66,  290.55,  289.81,  289.74,  289.96,      WRLM 190
     .  288.55,  287.96,  287.15,  285.81,  286.18,  285.30,  284.78,      WRLM 191
     .  284.48,  285.30,  285.15,  284.85,  285.37,  285.15,  285.59,      WRLM 192
     .  285.37,  286.11,  285.44,  285.81,  285.37,  285.81,  285.22,      WRLM 193
     .  285.37,  286.11,  285.37,  285.52,  286.04,  285.74,  286.41,      WRLM 194
     .  286.26,  286.85,  286.70,  287.00,  286.85,  287.44,  286.85,      WRLM 195
     .  287.89,  287.59,  287.66,  288.18,  288.11,  287.96,  288.48,      WRLM 196
     .  286.26,  287.66,  288.33,  288.03,  288.40,  287.89,  287.07,      WRLM 197
     .  285.52,  284.26,  283.67,  282.63,  282.11,  282.04,  281.00,      WRLM 198
     .  281.15,  280.19,  280.71,  279.67,  279.38,  278.49,  277.82,      WRLM 199
     .  278.19,  279.60,  277.97,  278.04,  278.04,  278.49,  278.41,      WRLM 200
     .  279.45,  279.67,  280.71,  280.63,  281.30,  281.15,  281.15,      WRLM 201
     .  280.71,  280.41,  280.26,  279.08,  278.26,  279.15,  277.82,      WRLM 202
     .  277.30,  276.41,  274.93,  274.93,  273.75,  273.31,  272.86,      WRLM 203
     .  271.38,  272.05,  270.94,  270.49,  269.83,  269.61,  268.94/      WRLM 204
      DATA B 4/ -32.75,  -33.75,  -34.26,  -34.36,  -33.92,  -32.94,      WRLM 205
     .  -33.25,  -33.97,  -34.16,  -34.80,  -36.01,  -36.35,  -37.23,      WRLM 206
     .  -37.53,  -38.11,  -38.45,  -38.25,  -38.89,  -39.77,  -40.31,      WRLM 207
     .  -39.96,  -39.85,  -40.94,  -41.53,  -41.77,  -42.79,  -43.55,      WRLM 208
     .  -44.01,  -44.99,  -44.01,  -44.65,  -45.15,  -45.77,  -46.26,      WRLM 209
     .  -47.38,  -47.82,  -48.65,  -49.18,  -49.33,  -50.30,  -51.20,      WRLM 210
     .  -51.45,  -51.06,  -51.72,  -51.82,  -50.94,  -50.85,  -50.45,      WRLM 211
     .  -49.38,  -48.45,  -48.36,  -46.65,  -46.11,  -45.23,  -45.09,      WRLM 212
     .  -44.70,  -44.25,  -43.13,  -42.65,  -42.01,  -41.23,  -40.05,      WRLM 213
     .  -39.09,  -38.45,  -37.43,  -36.50,  -35.95,  -35.19,  -34.41,      WRLM 214
     .  -33.97,  -32.84,  -31.75,  -30.75,  -29.77,  -28.89,  -28.16,      WRLM 215
     .  -27.15,  -26.41,  -25.48,  -24.46,  -23.58,  -22.95,  -22.31,      WRLM 216
     .  -21.48,  -20.85,  -20.07,  -19.09,  -18.31,  -17.43,  -17.00,      WRLM 217
     .  -16.17,  -15.39,  -14.31,  -13.78,  -13.14,  -12.12,  -11.24,      WRLM 218
     .  -10.22,   -9.44,   -8.66,   -7.49,   -6.71,   -5.83,   -5.14,      WRLM 219
     .   -4.02,   -2.76,   -2.51,   -1.58,   -0.85,    0.22,    1.39,      WRLM 220
     .    1.68,    2.76,    3.58,    4.56,    4.90,    5.78,    6.66,      WRLM 221
     .    7.34,    8.17,    8.75,    8.51,    7.83,    7.44,    7.19,      WRLM 222
     .    7.88,    8.46,    8.56,    9.39,    9.83,   10.66,   11.48,      WRLM 223
```

```
     . 12.41,   13.24,   13.87,   13.51,   13.78,   13.53,   14.12/    WRLM 224
   DATA A 5/   267.09,  266.20,  265.16,  264.79,  263.61,  262.72,    WRLM 225
     . 260.43,  259.73,  258.60,  257.84,  257.70,  256.23,  255.39,   WRLM 226
     . 253.69,  253.77,  253.95,  252.95,  252.38,  251.92,  250.07,   WRLM 227
     . 250.14,  248.73,  248.66,  247.03,  247.25,  246.22,  246.37,   WRLM 228
     . 244.44,  244.74,  244.07,  244.37,  245.49,  245.55,  246.51,   WRLM 229
     . 246.37,  247.62,  247.32,  248.36,  248.36,  249.79,  249.47,   WRLM 230
     . 248.91,  247.59,  246.51,  246.23,  244.36,  244.44,  243.55,   WRLM 231
     . 243.55,  242.66,  242.96,  242.07,  242.15,  241.92,  240.22,   WRLM 232
     . 238.74,  238.67,  237.48,  236.82,  237.48,  236.15,  235.49,   WRLM 233
     . 235.56,  234.89,  235.71,  235.63,  235.34,  235.49,  235.72,   WRLM 234
     . 235.04,  235.64,  235.70,  236.45,  237.54,  237.63,  237.26,   WRLM 235
     . 237.71,  237.54,  235.49,  233.93,  232.45,  231.49,  231.41,   WRLM 236
     . 230.62,  230.82,  230.16,  229.27,  229.05,  228.82,  227.94,   WRLM 237
     . 228.38,  227.79,  226.53,  226.31,  226.46,  226.23,  225.27,   WRLM 238
     . 225.35,  224.90,  224.16,  223.20,  224.01,  223.50,  222.09,   WRLM 239
     . 221.35,  219.79,  218.98,  218.24,  216.54,  215.72,  214.47,   WRLM 240
     . 212.64,  213.36,  211.95,  210.99,  211.65,  210.62,  209.28,   WRLM 241
     . 208.25,  206.99,  207.21,  208.40,  209.36,  209.35,  207.73,   WRLM 242
     . 206.32,  206.10,  205.07,  204.99,  205.73,  204.18,  202.48,   WRLM 243
     . 202.48,  201.37,  201.37,  200.40,  199.44,  198.92,  196.85/   WRLM 244
   DATA B 5/    14.55,   15.09,   15.24,   16.02,   16.22,   15.73,   WRLM 245
     .  16.46,   16.51,   16.85,   17.09,   17.82,   18.17,   18.85,   WRLM 246
     .  19.63,   20.80,   21.77,   22.85,   23.53,   24.21,   25.43,   WRLM 247
     .  26.31,   27.29,   28.07,   28.89,   29.28,   30.07,   30.75,   WRLM 248
     .  31.12,   30.99,   30.41,   29.97,   29.58,   28.99,   28.50,   WRLM 249
     .  27.56,   26.75,   26.18,   25.19,   24.70,   23.82,   23.19,   WRLM 250
     .  23.58,   24.21,   24.85,   25.82,   27.04,   27.97,   29.04,   WRLM 251
     .  29.43,   30.11,   30.80,   31.48,   32.41,   33.48,   33.37,   WRLM 252
     .  34.55,   35.28,   36.31,   36.94,   37.53,   38.45,   38.89,   WRLM 253
     .  39.87,   40.45,   41.18,   42.55,   43.67,   44.60,   45.67,   WRLM 254
     .  46.69,   47.57,   47.23,   47.52,   47.98,   47.62,   48.21,   WRLM 255
     .  48.65,   49.08,   49.82,   50.60,   50.09,   51.23,   51.86,   WRLM 256
     .  52.74,   53.77,   54.20,   53.33,   53.91,   54.69,   55.77,   WRLM 257
     .  56.25,   56.79,   56.69,   57.03,   57.47,   57.86,   57.33,   WRLM 258
     .  57.72,   57.91,   57.23,   57.62,   58.35,   58.94,   58.25,   WRLM 259
     .  58.74,   59.08,   59.03,   59.42,   59.52,   59.42,   59.86,   WRLM 260
     .  59.57,   60.25,   60.59,   60.11,   59.52,   59.23,   59.37,   WRLM 261
     .  59.08,   58.94,   59.81,   60.35,   60.20,   60.69,   60.79,   WRLM 262
     .  60.15,   59.47,   59.08,   58.59,   58.45,   57.76,   57.23,   WRLM 263
     .  56.89,   56.79,   56.11,   55.77,   55.72,   55.33,   54.99/   WRLM 264
   DATA A 6/   197.22,  198.33,  199.29,  200.63,  200.92,  201.66,   WRLM 265
     . 199.66,  198.85,  197.61,  196.92,  197.29,  197.29,  196.33,   WRLM 266
     . 195.15,  195.74,  194.56,  194.11,  193.74,  194.56,  195.22,   WRLM 267
     . 195.59,  197.22,  198.63,  198.25,  198.77,  197.96,  196.63,   WRLM 268
     . 195.52,  194.63,  193.15,  192.71,  191.60,  192.93,  194.41,   WRLM 269
     . 194.78,  195.59,  196.26,  197.37,  198.70,  197.74,  197.44,   WRLM 270
     . 196.18,  195.74,  194.85,  194.26,  192.35,  194.19,  196.11,   WRLM 271
     . 197.15,  196.48,  198.55,  198.92,  200.92,  201.29,  202.77,   WRLM 272
     . 202.92,  203.81,  204.33,  205.07,  206.77,  207.43,  208.03,   WRLM 273
     . 209.14,  209.88,  209.95,  211.83,  213.13,  214.17,  215.28,   WRLM 274
     . 216.17,  217.87,  219.87,  220.63,  222.98,  223.94,  226.09,   WRLM 275
     . 227.05,  228.61,  228.90,  230.01,  230.30,  232.03,  233.41,   WRLM 276
     . 234.75,  235.93,  236.67,  237.26,  239.52,  239.65,  240.96,   WRLM 277
     . 241.70,  242.89,  243.92,  245.25,  245.18,  246.22,  247.62,   WRLM 278
     . 249.25,  250.67,  250.95,  251.77,  251.69,  251.57,  252.73,   WRLM 279
```

• 251.77,	253.77,	254.08,	255.02,	256.73,	257.30,	258.80,	WRLM 280
• 259.84,	260.72,	262.35,	263.54,	263.68,	265.09,	265.24,	WRLM 281
• 257.31,	260.72,	264.72,	263.68,	262.94,	264.35,	264.72,	WRLM 282
• 265.90,	266.42,	268.13,	269.64,	267.75,	267.63,	268.94,	WRLM 283
• 259.24,	269.90,	270.79,	271.31,	271.66,	272.85,	273.38/	WRLM 284
DATA B 6/	55.77,	56.01,	56.89,	57.23,	57.91,	58.57,	WRLM 285
• 58.11,	58.84,	58.25,	58.25,	59.18,	59.41,	59.62,	WRLM 286
• 59.47,	60.54,	60.93,	61.57,	62.30,	62.45,	63.37,	WRLM 287
• 62.64,	62.79,	62.93,	63.75,	64.30,	64.79,	64.20,	WRLM 288
• 64.54,	64.10,	64.45,	65.03,	65.52,	66.15,	66.35,	WRLM 289
• 65.81,	66.10,	65.71,	65.81,	66.30,	66.35,	67.13,	WRLM 290
• 66.88,	67.42,	67.62,	68.01,	67.71,	68.93,	68.98,	WRLM 291
• 69.37,	69.76,	70.01,	70.44,	70.20,	70.74,	71.22,	WRLM 292
• 70.44,	70.10,	70.74,	70.20,	70.83,	70.25,	70.93,	WRLM 293
• 70.10,	70.83,	70.05,	70.01,	70.25,	69.66,	70.30,	WRLM 294
• 69.81,	69.81,	69.96,	68.98,	69.18,	68.59,	69.66,	WRLM 295
• 69.66,	70.15,	69.91,	70.64,	69.76,	69.96,	70.25,	WRLM 296
• 69.27,	70.10,	69.23,	69.76,	69.76,	69.03,	69.23,	WRLM 297
• 68.74,	68.79,	68.54,	68.74,	67.91,	67.86,	67.91,	WRLM 298
• 67.66,	66.88,	66.74,	65.86,	66.59,	67.42,	67.91,	WRLM 299
• 68.69,	68.59,	68.93,	68.10,	68.25,	68.44,	67.91,	WRLM 300
• 68.15,	67.57,	68.25,	67.76,	67.23,	66.64,	67.62,	WRLM 301
• 67.81,	68.98,	69.32,	69.57,	69.96,	70.69,	71.57,	WRLM 302
• 71.76,	70.79,	70.69,	70.01,	69.65,	69.18,	68.88,	WRLM 303
• 68.10,	68.05,	68.49,	68.35,	67.52,	66.98,	67.71/	WRLM 304
DATA A 7/	273.82,	274.42,	275.16,	59.85,	58.67,	57.78,	WRLM 305
• 57.41,	56.97,	56.15,	55.41,	54.38,	53.19,	52.60,	WRLM 306
• 52.38,	51.34,	50.67,	50.01,	49.27,	47.79,	47.27,	WRLM 307
• 46.97,	46.16,	45.35,	44.61,	44.09,	43.57,	43.05,	WRLM 308
• 42.83,	41.57,	40.61,	40.31,	41.13,	41.72,	42.24,	WRLM 309
• 41.57,	40.76,	40.09,	38.61,	38.91,	38.39,	37.87,	WRLM 310
• 38.31,	37.94,	37.94,	38.02,	39.13,	39.57,	40.24,	WRLM 311
• 40.76,	41.05,	42.16,	43.05,	43.50,	43.87,	44.09,	WRLM 312
• 45.72,	46.68,	46.97,	46.97,	46.83,	47.27,	47.49,	WRLM 313
• 48.31,	48.90,	49.34,	49.19,	49.71,	49.79,	50.01,	WRLM 314
• 51.04,	51.12,	51.71,	52.67,	53.63,	54.15,	54.60,	WRLM 315
• 55.19,	55.63,	55.86,	55.65,	56.30,	56.00,	56.60,	WRLM 316
• 56.82,	57.48,	58.22,	58.82,	59.70,	59.41,	58.59,	WRLM 317
• 57.85,	57.63,	57.34,	56.52,	56.00,	55.93,	55.41,	WRLM 318
• 55.12,	53.63,	52.30,	51.56,	50.38,	49.64,	49.19,	WRLM 319
• 48.08,	47.57,	46.31,	44.98,	44.01,	43.05,	42.90,	WRLM 320
• 42.90,	42.31,	42.46,	42.30,	41.94,	41.50,	41.20,	WRLM 321
• 40.90,	39.94,	39.13,	38.83,	38.91,	38.83,	38.24,	WRLM 322
• 38.17,	37.50,	36.83,	36.17,	36.09,	35.80,	35.58,	WRLM 323
• 35.06,	35.13,	34.47,	34.24,	33.56,	32.99,	32.84/	WRLM 324
DATA B 7/	67.62,	68.25,	68.93,	25.14,	25.09,	25.38,	WRLM 325
• 26.11,	26.65,	26.46,	26.36,	26.31,	26.85,	27.33,	WRLM 326
• 27.33,	28.11,	28.75,	29.09,	29.48,	29.72,	30.26,	WRLM 327
• 31.24,	31.82,	32.36,	22.99,	33.33,	34.36,	35.43,	WRLM 328
• 36.50,	37.04,	37.53,	37.97,	38.31,	38.36,	39.14,	WRLM 329
• 39.04,	38.89,	38.75,	38.65,	37.97,	37.62,	36.89,	WRLM 330
• 36.59,	36.36,	36.01,	35.72,	35.67,	34.94,	34.79,	WRLM 331
• 34.41,	34.06,	33.53,	33.19,	32.55,	31.43,	30.65,	WRLM 332
• 30.39,	30.25,	29.77,	29.35,	28.69,	20.31,	27.82,	WRLM 333
• 27.14,	26.75,	26.36,	26.35,	26.07,	25.43,	24.80,	WRLM 334
• 25.09,	24.41,	23.77,	23.62,	23.92,	23.82,	24.02,	WRLM 335

```
     .   24.51,   25.04,   25.72,   25.97,   25.56,   25.09,   24.70,   WRLM 336
     .   24.12,   23.52,   23.66,   23.14,   22.60,   22.07,   21.48,   WRLM 337
     .   20.70,   20.20,   19.76,   19.19,   18.56,   17.97,   17.34,   WRLM 338
     .   16.80,   16.36,   16.12,   15.14,   15.00,   14.17,   14.26,   WRLM 339
     .   14.26,   14.02,   13.78,   13.24,   12.75,   12.95,   13.68,   WRLM 340
     .   14.26,   14.85,   15.73,   16.73,   17.14,   17.78,   18.56,   WRLM 341
     .   18.85,   19.82,   21.04,   21.63,   22.07,   22.90,   23.43,   WRLM 342
     .   24.02,   24.41,   24.80,   25.53,   25.87,   26.94,   27.68,   WRLM 343
     .   28.07,   28.65,   29.19,   28.50,   28.50,   28.55,   29.04/   WRLM 344
      DATA A 8/   32.39,   32.25,   32.47,   33.28,   33.65,   33.95,   WRLM 345
     .   34.61,   35.35,   35.06,   36.17,   37.28,   36.76,   36.76,   WRLM 346
     .   37.65,   38.46,   38.93,   39.05,   38.68,   39.42,   40.58,   WRLM 347
     .   41.87,   42.01,   42.31,   42.75,   44.01,   45.42,   46.23,   WRLM 348
     .   47.27,   48.01,   48.75,   50.23,   50.75,   50.53,   50.53,   WRLM 349
     .   50.90,   50.53,   49.93,   49.55,   48.60,   48.01,   46.60,   WRLM 350
     .   46.09,   45.12,   44.53,   44.24,   43.20,   42.75,   42.38,   WRLM 351
     .   42.31,   41.20,   41.13,   39.94,   39.42,   38.91,   38.39,   WRLM 352
     .   37.43,   36.98,   37.57,   37.65,   37.94,   38.61,   38.46,   WRLM 353
     .   38.54,   39.50,   39.72,   39.05,   39.35,   39.65,   39.05,   WRLM 354
     .   39.57,   33.17,   37.72,   36.30,   35.95,   35.21,   34.39,   WRLM 355
     .   33.58,   32.99,   33.28,   34.17,   33.87,   34.69,   34.32,   WRLM 356
     .   34.47,   34.32,   33.72,   32.10,   31.80,   31.55,   31.65,   WRLM 357
     .   30.62,   29.65,   29.14,   28.84,   28.25,   27.43,   26.62,   WRLM 358
     .   26.62,   25.95,   24.25,   23.07,   21.88,   21.00,   20.48,   WRLM 359
     .   20.11,   19.81,   19.44,   19.37,   18.92,   18.48,   17.44,   WRLM 360
     .   16.78,   16.18,   16.92,   17.15,   16.33,   16.48,   15.52,   WRLM 361
     .   15.67,   14.70,   13.67,   14.25,   14.04,   13.30,   13.15,   WRLM 362
     .   12.56,   12.85,   12.34,   11.82,   12.19,   11.74,   11.08,   WRLM 363
     .   10.71,   10.78,   10.49,   11.00,   10.78,   10.63,   11.15/   WRLM 364
      DATA B 8/   29.77,   28.80,   27.63,   26.80,   26.07,   24.99,   WRLM 365
     .   24.51,   24.02,   23.04,   22.26,   21.34,   20.41,   19.68,   WRLM 366
     .   19.04,   18.17,   17.19,   16.12,   15.24,   14.95,   14.07,   WRLM 367
     .   13.29,   12.70,   11.78,   11.29,   10.70,   10.56,   10.80,   WRLM 368
     .   10.95,   11.09,   11.39,   11.87,   11.83,   11.39,   10.90,   WRLM 369
     .   10.51,    9.92,    8.65,    7.83,    6.95,    5.97,    4.66,   WRLM 370
     .    3.34,    3.34,    2.71,    1.73,    1.49,    0.95,    0.61,   WRLM 371
     .    0.17,    0.37,   -1.05,   -1.98,   -2.71,   -3.63,   -4.46,   WRLM 372
     .   -4.75,   -5.34,   -5.83,   -6.80,   -7.78,   -7.92,   -8.80,   WRLM 373
     .   -9.44,   -9.68,  -11.09,  -11.53,  -11.92,  -12.90,  -13.97,   WRLM 374
     .  -14.90,  -16.07,  -16.46,  -16.55,  -17.63,  -18.07,  -18.56,   WRLM 375
     .  -18.85,  -19.14,  -20.12,  -20.90,  -21.34,  -22.12,  -22.90,   WRLM 376
     .  -23.77,  -24.07,  -24.55,  -24.55,  -25.14,  -26.36,  -27.38,   WRLM 377
     .  -28.36,  -28.89,  -29.67,  -30.16,  -30.75,  -31.04,  -31.67,   WRLM 378
     .  -32.31,  -32.55,  -33.04,  -33.53,  -33.14,  -33.63,  -33.82,   WRLM 379
     .  -33.67,  -33.92,  -33.77,  -34.21,  -34.36,  -33.87,  -33.53,   WRLM 380
     .  -32.75,  -32.02,  -31.82,  -31.09,  -30.26,  -29.77,  -28.94,   WRLM 381
     .  -28.50,  -27.68,  -26.75,  -25.97,  -25.53,  -24.75,  -23.82,   WRLM 382
     .  -22.85,  -22.26,  -21.48,  -20.60,  -19.68,  -18.99,  -18.46,   WRLM 383
     .  -17.09,  -16.61,  -15.19,  -14.31,  -13.39,  -12.61,  -12.07/   WRLM 384
      DATA A 9/   12.11,   12.19,   12.26,   11.97,   11.97,   11.82,   WRLM 385
     .   11.00,   11.37,   10.71,   10.26,    8.78,    8.56,    8.41,   WRLM 386
     .    7.82,    7.67,    7.75,    7.60,    7.97,    8.49,    8.63,   WRLM 387
     .    8.26,    7.67,    7.75,    7.30,    7.30,    7.23,    6.12,   WRLM 388
     .    4.71,    4.64,    4.05,    3.01,    2.42,    1.53,    1.01,   WRLM 389
     .  359.90,    0.05,  359.75,  359.94,  358.42,  357.46,  356.50,   WRLM 390
     .  355.76,  354.94,  353.17,  351.54,  350.95,  350.06,  349.32,   WRLM 391
```

```
     .  348.95,  348.13,  346.21,  345.54,  345.62,  343.24,  343.56,      WRLM 392
     .  343.03,  342.68,  342.29,  342.35,  341.77,  341.54,  341.92,      WRLM 393
     .  342.29,  341.34,  342.43,  342.29,  342.73,  342.58,  342.95,      WRLM 394
     .  341.62,  341.99,  342.95,  343.17,  344.35,  344.91,  344.95,      WRLM 395
     .  345.70,  347.99,  348.36,  349.17,  349.17,  348.35,  348.87,      WRLM 396
     .  350.00,  350.00,  351.39,  352.20,  353.31,  354.28,  355.76,      WRLM 397
     .  357.24,  358.72,    0.42,    1.69,    2.71,    3.23,    4.19,      WRLM 398
     .    4.55,    5.30,    6.04,    7.82,    9.30,    9.97,   10.26,      WRLM 399
     .   10.49,    9.82,    9.60,    8.71,   10.12,   10.04,   11.08,      WRLM 400
     .   12.71,   13.45,   14.63,   14.70,   15.96,   17.59,   19.51,      WRLM 401
     .   19.51,   20.48,   21.44,   22.95,   23.73,   24.55,   26.10,      WRLM 402
     .   27.21,   28.40,   29.65,   31.50,   31.50,   31.73,   32.69,      WRLM 403
     .   31.95,   32.17,   33.21,   34.54,   34.91,   34.76,   35.21/      WRLM 404
      DATA B 9/ -11.83,  -11.39,  -10.36,   -9.39,   -8.22,   -7.39,      WRLM 405
     .   -6.02,   -5.39,   -5.14,   -4.07,   -3.10,   -2.12,   -1.54,      WRLM 406
     .   -1.05,   -3.61,    0.17,    1.00,    1.78,    2.41,    3.05,      WRLM 407
     .    3.34,    3.78,    4.27,    4.51,    3.63,    3.49,    3.97,      WRLM 408
     .    4.36,    4.90,    5.19,    5.39,    5.93,    6.17,    5.83,      WRLM 409
     .    5.68,    5.54,    5.29,    5.39,    5.19,    4.56,    5.00,      WRLM 410
     .    5.24,    5.05,    4.95,    4.27,    4.46,    4.80,    5.10,      WRLM 411
     .    5.68,    6.17,    7.24,    8.41,    9.39,   10.61,   10.80,      WRLM 412
     .   11.24,   12.22,   12.56,   13.19,   13.58,   14.22,   14.85,      WRLM 413
     .   15.43,   16.07,   17.49,   18.51,   18.95,   19.68,   19.87,      WRLM 414
     .   20.80,   22.07,   22.60,   23.58,   24.99,   25.53,   26.02,      WRLM 415
     .   27.24,   28.21,   28.75,   29.48,   30.02,   31.14,   31.77,      WRLM 416
     .   32.45,   32.89,   33.97,   34.45,   35.38,   35.38,   35.28,      WRLM 417
     .   35.23,   35.58,   36.01,   36.31,   36.36,   36.70,   36.70,      WRLM 418
     .   36.94,   36.84,   36.94,   36.94,   37.14,   36.70,   35.72,      WRLM 419
     .   35.09,   35.04,   34.60,   34.41,   33.82,   33.14,   32.89,      WRLM 420
     .   33.14,   32.36,   32.65,   31.63,   31.43,   30.26,   31.14,      WRLM 421
     .   32.16,   32.45,   32.89,   32.06,   32.11,   31.53,   31.53,      WRLM 422
     .   31.09,   30.94,   30.94,   30.89,   30.36,   29.63,   28.65,      WRLM 423
     .   30.07,   30.65,   31.09,   31.82,   32.45,   33.48,   34.36/      WRLM 424
      DATA A10/   35.28,   36.09,   35.35,   33.67,   32.39,   30.99,      WRLM 425
     .   29.95,   28.69,   27.80,   25.51,   25.81,   26.18,   26.77,      WRLM 426
     .   26.69,   25.73,   25.07,   24.84,   23.51,   23.07,   22.11,      WRLM 427
     .   22.62,   22.55,   22.99,   22.62,   22.03,   21.59,   21.00,      WRLM 428
     .   20.92,   19.96,   20.77,   19.44,   18.92,   18.11,   18.63,      WRLM 429
     .   18.85,   18.33,   18.11,   16.63,   15.44,   15.52,   15.37,      WRLM 430
     .   14.11,   14.04,   14.48,   13.08,   12.85,   11.74,   11.89,      WRLM 431
     .   12.71,   12.85,   13.96,   14.78,   15.15,   16.18,   16.92,      WRLM 432
     .   17.44,   17.29,   16.18,   16.04,   16.41,   15.67,   15.15,      WRLM 433
     .   14.63,   15.00,   13.82,   13.45,   13.30,   12.26,   11.97,      WRLM 434
     .   10.86,    9.89,    9.60,   10.19,    9.00,    7.23,    5.53,      WRLM 435
     .    4.49,    2.71,    2.86,    2.57,    2.05,    1.16,    0.94,      WRLM 436
     .  359.68,  359.24,  359.38,  356.87,  358.79,  357.98,  356.57,      WRLM 437
     .  354.72,  353.61,  353.24,  352.20,  351.98,  351.39,  350.21,      WRLM 438
     .  350.35,  349.84,  349.84,  350.07,  350.95,  350.80,  351.32,      WRLM 439
     .  352.50,  353.46,  354.42,  355.90,  356.64,  358.13,  358.20,      WRLM 440
     .  357.75,  358.57,  357.68,  356.79,  356.57,  354.79,  355.68,      WRLM 441
     .  357.31,  357.68,  359.01,  359.24,    0.12,    1.53,    2.05,      WRLM 442
     .    2.42,    4.12,    3.31,    3.97,    4.34,    4.79,    5.67,      WRLM 443
     .    6.78,    7.01,    7.75,    8.26,    8.93,    8.49,    7.67/      WRLM 444
      DATA B10/   35.04,   35.82,   35.97,   36.31,   35.87,   36.21,      WRLM 445
     .   35.77,   35.67,   36.36,   38.11,   38.31,   38.84,   39.67,      WRLM 446
     .   40.31,   40.13,   40.35,   40.60,   40.40,   40.50,   39.90,      WRLM 447
```

```
     .    39.62,   39.39,   38.45,   37.77,   36.79,   36.21,   36.45,   WRLM 448
     .    37.23,   37.72,   38.69,   38.97,   39.07,   39.92,   40.50,   WRLM 449
     .    41.53,   42.21,   42.35,   42.70,   43.04,   43.52,   43.96,   WRLM 450
     .    44.26,   44.70,   45.13,   45.13,   45.28,   45.28,   44.84,   WRLM 451
     .    44.01,   43.13,   42.94,   42.11,   41.33,   41.43,   41.14,   WRLM 452
     .    40.70,   40.40,   40.11,   39.62,   39.04,   39.21,   37.92,   WRLM 453
     .    38.31,   39.33,   40.11,   40.50,   41.14,   41.38,   42.26,   WRLM 454
     .    42.65,   42.89,   43.18,   44.06,   44.60,   43.77,   43.48,   WRLM 455
     .    43.77,   43.33,   42.65,   41.92,   41.62,   41.28,   40.40,   WRLM 456
     .    40.21,   39.57,   38.94,   37.67,   37.38,   36.84,   36.55,   WRLM 457
     .    36.55,   35.82,   35.97,   36.31,   36.99,   37.04,   37.04,   WRLM 458
     .    37.62,   38.40,   39.62,   40.65,   41.33,   42.65,   43.04,   WRLM 459
     .    42.79,   42.89,   42.74,   42.99,   42.94,   43.04,   43.77,   WRLM 460
     .    44.26,   45.52,   45.35,   46.89,   47.47,   47.77,   48.21,   WRLM 461
     .    48.30,   49.28,   49.08,   49.62,   49.18,   50.06,   49.72,   WRLM 462
     .    50.35,   50.89,   51.77,   51.91,   52.40,   52.16,   52.79,   WRLM 463
     .    52.69,   53.23,   52.99,   53.13,   53.33,   53.86,   54.55/   WRLM 464
      DATA A11/     7.67,    7.01,    7.97,    9.15,    9.75,   10.71,   WRLM 465
     .     9.23,    9.75,   10.19,   10.78,   12.48,   13.82,   14.04,   WRLM 466
     .    14.85,   15.52,   16.18,   17.52,   18.26,   19.66,   19.81,   WRLM 467
     .    20.92,   20.85,   20.55,   20.92,   20.85,   22.55,   22.77,   WRLM 468
     .    24.03,   24.03,   23.81,   24.92,   26.40,   27.29,   28.54,   WRLM 469
     .    27.88,   28.17,   27.43,   26.40,   24.84,   24.03,   22.92,   WRLM 470
     .    22.03,   21.14,   19.66,   17.96,   17.29,   16.85,   16.18,   WRLM 471
     .    15.96,   17.00,   18.18,   17.89,   17.00,   16.70,   16.92,   WRLM 472
     .    17.00,   16.41,   16.26,   15.30,   14.48,   14.19,   13.37,   WRLM 473
     .    13.00,   13.15,   12.04,   11.60,   10.78,    9.15,    8.49,   WRLM 474
     .     8.04,    7.23,    5.97,    5.38,    5.97,    5.60,    6.64,   WRLM 475
     .     6.49,    5.16,    4.86,    5.67,    6.19,    6.55,    7.30,   WRLM 476
     .     7.97,    9.60,    9.89,   10.26,   10.93,   11.15,   12.11,   WRLM 477
     .    12.56,   13.45,   13.59,   14.56,   14.70,   15.67,   14.78,   WRLM 478
     .    15.37,   16.41,   17.29,   17.29,   17.44,   18.70,   19.00,   WRLM 479
     .    19.37,   19.66,   20.63,   21.59,   22.99,   24.33,   25.66,   WRLM 480
     .    26.10,   26.84,   27.51,   28.17,   28.77,   29.36,   30.54,   WRLM 481
     .    30.54,   29.51,   29.43,   29.29,   28.62,   27.58,   28.69,   WRLM 482
     .    29.58,   30.25,   31.73,   32.47,   33.43,   34.76,   35.65,   WRLM 483
     .    39.17,   37.35,   39.35,   40.39,   40.31,   40.09,   38.91/   WRLM 484
      DATA B11/    55.18,   55.96,   56.55,   56.45,   56.45,   55.86,   WRLM 485
     .    55.33,   54.84,   54.06,   53.57,   53.86,   53.28,   53.72,   WRLM 486
     .    53.86,   54.25,   54.11,   54.40,   54.11,   54.11,   54.94,   WRLM 487
     .    54.84,   55.33,   56.01,   56.69,   57.23,   57.86,   57.13,   WRLM 488
     .    57.08,   58.06,   58.94,   59.03,   59.08,   58.25,   57.62,   WRLM 489
     .    58.94,   59.72,   60.50,   60.64,   60.20,   59.81,   59.81,   WRLM 490
     .    60.01,   59.57,   59.81,   59.62,   59.37,   59.08,   59.47,   WRLM 491
     .    59.03,   58.64,   58.89,   58.45,   58.15,   57.76,   57.47,   WRLM 492
     .    56.79,   56.59,   55.91,   56.01,   55.42,   55.62,   55.52,   WRLM 493
     .    56.30,   56.59,   57.13,   57.65,   58.84,   58.50,   57.91,   WRLM 494
     .    57.86,   57.67,   57.96,   58.94,   59.47,   59.76,   60.59,   WRLM 495
     .    60.93,   61.23,   61.76,   62.25,   62.15,   62.45,   62.45,   WRLM 496
     .    63.18,   62.93,   63.37,   64.10,   64.35,   64.54,   64.54,   WRLM 497
     .    65.13,   65.32,   66.15,   66.49,   67.03,   67.62,   67.96,   WRLM 498
     .    68.25,   67.91,   67.60,   68.25,   68.84,   68.40,   68.40,   WRLM 499
     .    69.18,   69.66,   69.27,   69.85,   69.57,   70.15,   69.76,   WRLM 500
     .    70.40,   70.64,   69.80,   70.25,   70.35,   70.35,   70.10,   WRLM 501
     .    69.66,   69.66,   69.08,   68.79,   68.28,   68.44,   68.40,   WRLM 502
     .    68.59,   69.23,   69.57,   68.68,   68.54,   68.54,   68.15,   WRLM 503
```

```
     .    64.49,    58.20,    67.71,    67.23,    66.48,    66.15,    66.05/    WRLM 504
      DATA A12/     38.58,    37.26,    35.21,    33.87,    32.69,    32.17,    WRLM 505
     .    32.54,    33.43,    33.95,    34.60,    35.13,    35.95,    36.39,    WRLM 506
     .    37.72,    37.37,    38.17,    37.06,    37.50,    38.17,    38.91,    WRLM 507
     .    39.35,    40.24,    40.53,    39.87,    39.94,    41.20,    42.15,    WRLM 508
     .    43.64,    44.24,    44.31,    43.05,    43.42,    44.24,    44.68,    WRLM 509
     .    44.98,    45.27,    46.31,    45.35,    45.42,    46.33,    47.57,    WRLM 510
     .    48.23,    49.05,    49.79,    50.03,    50.90,    52.15,    52.75,    WRLM 511
     .    54.23,    55.26,    56.37,    57.04,    58.30,    59.56,    60.15,    WRLM 512
     .    61.18,    62.22,    62.59,    63.70,    64.29,    65.63,    66.51,    WRLM 513
     .    67.25,    67.70,    68.88,    69.47,    68.35,    69.10,    70.73,    WRLM 514
     .    71.77,    71.84,    72.66,    72.36,    72.88,    72.51,    73.10,    WRLM 515
     .    72.51,    73.40,    73.10,    73.40,    73.17,    73.91,    73.54,    WRLM 516
     .    74.43,    74.36,    74.21,    74.65,    75.17,    75.17,    74.73,    WRLM 517
     .    75.39,    75.91,    76.13,    75.97,    76.50,    76.28,    76.95,    WRLM 518
     .    77.76,    78.36,    78.58,    79.47,    79.84,    79.84,    79.54,    WRLM 519
     .    79.64,    79.91,    79.84,    79.93,    81.32,    82.87,    83.91,    WRLM 520
     .    84.35,    84.72,    85.98,    86.13,    86.37,    86.50,    87.09,    WRLM 521
     .    88.13,    89.46,    90.27,    90.64,    91.01,    91.83,    92.12,    WRLM 522
     .    92.12,    92.12,    92.64,    93.63,    93.38,    94.05,    93.68,    WRLM 523
     .    94.19,    93.82,    93.38,    93.90,    93.75,    94.19,    95.01/    WRLM 524
      DATA B12/     65.57,    65.67,    65.96,    66.06,    66.79,    66.93,    WRLM 525
     .    66.15,    65.96,    65.62,    64.74,    64.06,    63.67,    63.23,    WRLM 526
     .    63.03,    62.54,    62.98,    63.57,    64.54,    64.30,    63.71,    WRLM 527
     .    63.86,    63.47,    64.15,    64.88,    65.32,    65.08,    65.57,    WRLM 528
     .    65.37,    65.67,    66.15,    67.42,    68.20,    68.15,    68.40,    WRLM 529
     .    68.15,    68.15,    67.81,    66.93,    66.54,    66.30,    65.96,    WRLM 530
     .    66.59,    67.08,    67.52,    67.13,    67.42,    67.71,    67.96,    WRLM 531
     .    67.23,    67.76,    67.96,    67.57,    68.30,    67.71,    25.04,    WRLM 532
     .    25.58,    25.43,    25.72,    25.43,    25.72,    25.19,    25.04,    WRLM 533
     .    24.16,    23.63,    22.95,    22.51,    21.97,    21.14,    20.99,    WRLM 534
     .    21.43,    22.26,    21.87,    21.34,    20.99,    20.56,    19.92,    WRLM 535
     .    19.48,    18.95,    17.92,    16.90,    16.22,    15.43,    15.24,    WRLM 536
     .    14.61,    13.78,    13.34,    13.09,    12.70,    12.12,    11.92,    WRLM 537
     .    11.58,    11.14,    10.75,    10.17,     9.73,     9.00,     8.70,    WRLM 538
     .     8.31,     8.41,     9.44,     9.92,    10.51,    11.39,    12.12,    WRLM 539
     .    12.41,    13.48,    14.51,    15.53,    16.26,    17.63,    18.17,    WRLM 540
     .    18.80,    19.48,    19.58,    20.17,    20.51,    21.48,    21.92,    WRLM 541
     .    21.92,    21.77,    22.75,    23.19,    22.95,    22.80,    21.97,    WRLM 542
     .    20.99,    20.46,    20.17,    19.97,    19.53,    19.04,    18.41,    WRLM 543
     .    17.97,    17.29,    17.00,    16.70,    16.22,    15.82,    15.63/    WRLM 544
      DATA A13/     95.97,    97.01,    96.78,    97.08,    97.23,    97.01,    WRLM 545
     .    97.67,    97.38,    97.67,    98.04,    97.89,    97.67,    97.38,    WRLM 546
     .    97.67,    97.52,    97.82,    97.45,    97.60,    98.04,    98.63,    WRLM 547
     .    99.45,    99.52,    99.23,    99.39,    99.97,   100.26,   100.12,    WRLM 548
     .   100.56,   102.11,   104.04,   104.25,   103.52,   103.89,   103.59,    WRLM 549
     .   103.69,   103.02,   102.71,   102.19,   101.52,   100.63,    99.67,    WRLM 550
     .   100.12,    99.45,    99.82,    99.75,   100.26,    99.75,    99.32,    WRLM 551
     .   100.56,   102.19,   102.26,   103.59,   104.70,   104.25,   103.74,    WRLM 552
     .   104.85,   105.37,   106.33,   106.55,   107.81,   108.70,   109.00,    WRLM 553
     .   108.92,   108.48,   108.55,   107.59,   106.48,   106.70,   106.70,    WRLM 554
     .   107.00,   108.26,   108.77,   109.14,   109.74,   110.40,   110.55,    WRLM 555
     .   111.51,   112.45,   112.70,   113.44,   114.40,   116.19,   117.51,    WRLM 556
     .   118.91,   119.51,   119.65,   120.76,   120.69,   120.34,   121.65,    WRLM 557
     .   121.06,   120.54,   120.91,   121.59,   122.17,   121.50,   121.95,    WRLM 558
     .   121.50,   120.62,   119.60,   119.68,   119.73,   120.47,   120.54,    WRLM 559
```

```
     .   120.92,  121.53,  122.32,  121.36,  120.16,  119.73,  119.80,    WRLM 560
     .   121.13,  121.65,  121.43,  122.39,  123.23,  123.45,  124.61,    WRLM 561
     .   125.28,  124.24,  124.54,  125.65,  126.32,  126.63,  126.17,    WRLM 562
     .   125.32,  125.17,  125.75,  126.17,  126.93,  125.09,  126.76,    WRLM 563
     .   129.28,  128.91,  125.46,  129.63,  127.50,  127.72,  128.66/    WRLM 564
      DATA B13/   16.55,   16.51,   16.07,   15.78,   14.85,   14.61,    WRLM 565
     .    14.02,   13.63,   12.96,   12.61,   11.73,   10.90,   10.61,    WRLM 566
     .     9.78,    9.63,    9.10,    8.56,    7.83,    7.92,    7.39,    WRLM 567
     .     7.05,    6.61,    7.29,    6.55,    5.54,    4.75,    4.02,    WRLM 568
     .     3.29,    2.56,    1.83,    2.24,    2.85,    3.29,    3.63,    WRLM 569
     .     3.83,    4.17,    5.00,    5.63,    6.46,    6.95,    7.34,    WRLM 570
     .     8.61,    9.34,   10.41,   11.09,   11.73,   12.61,   12.80,    WRLM 571
     .    12.56,   12.22,   11.67,   10.61,   10.02,    9.53,    8.90,    WRLM 572
     .     8.46,    8.80,    9.29,    9.88,   10.56,   11.39,   12.61,    WRLM 573
     .    13.58,   14.80,   15.48,   16.70,   17.63,   18.36,   19.63,    WRLM 574
     .    20.21,   20.60,   21.14,   20.65,   20.26,   20.99,   21.58,    WRLM 575
     .    21.97,   21.97,   22.51,   22.65,   22.51,   22.90,   24.12,    WRLM 576
     .    25.04,   25.16,   26.65,   27.63,   28.07,   28.75,   29.67,    WRLM 577
     .    29.97,   30.11,   30.50,   30.65,   31.14,   31.43,   31.72,    WRLM 578
     .    32.50,   32.84,   33.92,   34.45,   35.33,   35.72,   35.72,    WRLM 579
     .    36.26,   36.65,   37.48,   37.62,   37.92,   38.45,   39.53,    WRLM 580
     .    40.35,   39.72,   38.89,   39.43,   39.67,   39.43,   39.28,    WRLM 581
     .    38.84,   38.31,   37.58,   37.72,   37.97,   37.33,   36.70,    WRLM 582
     .    35.87,   35.19,   34.41,   34.02,   34.26,   34.55,   35.48,    WRLM 583
     .    36.45,   37.33,   38.26,   38.84,   39.28,   40.35,   40.09/    WRLM 584
      DATA A14/  129.28,  129.42,  130.24,  131.94,  133.13,  134.68,    WRLM 585
     .   134.90,  136.53,  136.68,  137.85,  138.16,  139.64,  139.49,    WRLM 586
     .   140.23,  140.23,  140.97,  140.82,  141.04,  140.60,  140.08,    WRLM 587
     .   138.90,  137.64,  136.60,  138.16,  137.34,  136.16,  134.90,    WRLM 588
     .   136.09,  137.64,  138.01,  139.34,  141.12,  142.08,  142.75,    WRLM 589
     .   143.34,  144.15,  145.86,  146.15,  147.19,  148.59,  149.55,    WRLM 590
     .   149.19,  150.59,  150.74,  150.95,  151.85,  153.26,  154.59,    WRLM 591
     .   154.59,  155.55,  156.74,  158.07,  158.81,  159.62,  159.77,    WRLM 592
     .   160.07,  161.32,  162.66,  163.47,  163.77,  164.14,  165.91,    WRLM 593
     .   164.60,  163.77,  164.21,  163.40,  161.47,  161.18,  159.99,    WRLM 594
     .   159.55,  158.81,  157.92,  156.96,  155.92,  155.40,  159.22,    WRLM 595
     .   155.85,  156.59,  156.44,  156.81,  158.51,  158.51,  158.81,    WRLM 596
     .   159.10,  160.07,  160.73,  161.55,  162.51,  162.14,  162.51,    WRLM 597
     .   163.17,  163.69,  163.40,  162.51,  163.25,  162.66,  163.10,    WRLM 598
     .   163.91,  165.62,  166.13,  167.69,  169.32,  170.28,  171.02,    WRLM 599
     .   171.83,  172.43,  173.83,  174.72,  175.76,  177.09,  178.35,    WRLM 600
     .   179.75,  179.09,  178.72,  177.16,  176.20,  176.57,  177.09,    WRLM 601
     .   178.64,  179.61,  179.66,  180.05,  180.86,  181.46,  181.46,    WRLM 602
     .   182.49,  183.31,  185.01,  184.79,  185.67,  186.04,  186.56,    WRLM 603
     .   185.97,  187.01,  187.60,  189.52,  189.52,  188.41,  187.09/    WRLM 604
      DATA B14/   41.53,   42.11,   42.16,   43.09,   42.84,   43.43,    WRLM 605
     .    44.06,   44.79,   45.48,   45.82,   46.65,   47.82,   48.79,    WRLM 606
     .    49.47,   50.79,   51.28,   52.05,   52.55,   53.28,   53.96,    WRLM 607
     .    53.81,   53.52,   53.91,   54.35,   54.74,   54.50,   54.64,    WRLM 608
     .    54.99,   55.03,   55.81,   56.69,   57.96,   58.74,   58.74,    WRLM 609
     .    59.23,   58.84,   58.64,   58.98,   59.13,   58.40,   58.74,    WRLM 610
     .    59.57,   59.33,   58.69,   58.20,   58.30,   58.79,   58.94,    WRLM 611
     .    59.76,   60.30,   60.79,   61.47,   61.32,   60.79,   60.40,    WRLM 612
     .    60.01,   60.30,   60.79,   61.23,   61.85,   62.30,   62.11,    WRLM 613
     .    61.47,   61.06,   60.59,   60.25,   59.47,   59.13,   59.03,    WRLM 614
     .    59.03,   57.96,   57.72,   57.13,   56.06,   55.08,   54.25,    WRLM 615
```

```
.     53.72,   52.79,   51.91,   51.33,   51.52,   52.30,   52.99,   WRLM 616
.     53.42,   54.05,   54.30,   53.95,   54.30,   54.94,   55.52,   WRLM 617
.     55.61,   55.95,   56.50,   57.03,   57.23,   58.01,   58.55,   WRLM 618
.     59.42,   59.57,   59.28,   60.11,   60.11,   59.52,   60.01,   WRLM 619
.     59.72,   60.25,   61.28,   61.03,   61.62,   62.06,   61.86,   WRLM 620
.     61.81,   62.45,   63.32,   63.47,   64.10,   64.79,   64.20,   WRLM 621
.     63.91,   64.40,   65.13,   65.65,   66.10,   65.37,   64.74,   WRLM 622
.     65.27,   64.35,   64.49,   64.05,   63.91,   63.67,   64.15,   WRLM 623
.     65.10,   65.27,   65.57,   65.37,   65.96,   66.35,   66.79/   WRLM 624
 DATA A15/    186.34,  185.08,  184.79,  184.34,  182.27,  180.79,  WRLM 625
.    180.72,  179.16,  177.83,  176.79,  176.27,  177.68,  177.16,   WRLM 626
.    175.83,  175.39,  173.83,  172.57,  171.39,  170.50,  171.51,   WRLM 627
.    171.32,  169.84,  168.36,  166.21,  166.13,  165.17,  163.84,   WRLM 628
.    162.66,  161.77,  160.88,  159.99,  159.62,  158.44,  157.33,   WRLM 629
.    155.25,  153.48,  153.77,  153.03,  152.74,  151.04,  150.59,   WRLM 630
.    149.26,  148.30,  147.56,  146.00,  144.75,  145.63,  144.82,   WRLM 631
.    142.67,  141.56,  139.79,  138.82,  139.05,  138.23,  137.42,   WRLM 632
.    136.83,  135.05,  133.94,  132.63,  131.57,  130.76,  129.65,   WRLM 633
.    128.91,  129.65,  129.65,  128.63,  128.02,  126.98,  125.35,   WRLM 634
.    124.76,  124.61,  123.87,  122.91,  121.50,  120.02,  118.10,   WRLM 635
.    116.55,  115.07,  113.44,  112.85,  111.07,  109.81,  108.92,   WRLM 636
.    107.37,  106.41,  106.35,  103.26,  109.59,  111.22,  111.74,   WRLM 637
.    112.40,  111.59,  110.33,  109.81,  108.92,  108.48,  107.74,   WRLM 638
.    107.07,  106.33,  105.81,  104.19,  102.41,  101.82,  101.30,   WRLM 639
.    100.41,   99.38,   97.52,   96.04,   95.90,   94.79,   93.08,   WRLM 640
.     92.94,   91.46,   89.53,   88.13,   86.79,   85.83,   84.65,   WRLM 641
.     85.53,   85.26,   83.46,   83.31,   82.65,   80.90,   81.26,   WRLM 642
.     81.76,   82.06,   82.50,   83.09,   82.65,   83.30,   84.42,   WRLM 643
.     84.79,   85.53,   86.05,   85.76,   86.50,   87.16,   86.50/   WRLM 644
 DATA B15/     66.74,   66.93,   66.54,   67.57,   67.81,   67.81,   WRLM 645
.     68.59,   68.54,   68.93,   68.93,   69.27,   69.66,   69.71,   WRLM 646
.     69.62,   69.42,   69.62,   69.52,   70.25,   69.47,   68.54,   WRLM 647
.     68.20,   68.30,   69.23,   69.23,   69.91,   69.86,   69.57,   WRLM 648
.     69.27,   69.42,   68.98,   69.71,   70.54,   70.54,   70.44,   WRLM 649
.     70.40,   69.86,   70.54,   70.35,   70.74,   70.59,   71.22,   WRLM 650
.     71.32,   72.00,   71.61,   71.03,   70.93,   71.66,   72.05,   WRLM 651
.     72.15,   72.40,   71.86,   71.66,   70.59,   70.88,   70.59,   WRLM 652
.     71.13,   70.79,   71.27,   70.83,   70.40,   70.40,   71.42,   WRLM 653
.     71.96,   72.54,   73.13,   73.03,   72.98,   73.47,   73.52,   WRLM 654
.     73.27,   73.71,   73.32,   72.79,   72.54,   73.52,   73.47,   WRLM 655
.     73.61,   73.37,   73.13,   73.76,   73.65,   73.32,   73.03,   WRLM 656
.     72.93,   72.98,   73.55,   74.15,   74.00,   74.54,   75.03,   WRLM 657
.     76.05,   76.35,   76.20,   77.03,   76.30,   76.55,   76.20,   WRLM 658
.     75.96,   76.69,   77.27,   77.25,   76.83,   76.00,   75.86,   WRLM 659
.     76.05,   76.05,   75.52,   75.66,   75.61,   75.86,   75.95,   WRLM 660
.     75.56,   75.22,   75.17,   74.74,   74.75,   74.78,   74.44,   WRLM 661
.     73.76,   72.74,   72.85,   73.42,   73.37,   73.18,   72.49,   WRLM 662
.     72.15,   71.86,   71.22,   70.69,   70.15,   69.65,   69.65,   WRLM 663
.     69.52,   68.64,   68.05,   67.32,   66.84,   65.96,   66.64/   WRLM 664
 DATA A16/     85.24,   85.24,   84.42,   83.31,   82.28,   81.91,   WRLM 665
.     80.65,   80.21,   79.02,   77.91,   77.02,   77.25,   76.65,   WRLM 666
.     75.99,   75.69,   74.51,   75.02,   74.06,   73.84,   75.69,   WRLM 667
.     75.91,   77.02,   77.32,   78.35,   79.32,   80.06,   80.06,   WRLM 668
.     80.95,   80.72,   79.61,   78.87,   77.47,   76.50,   75.99,   WRLM 669
.     75.91,   74.58,   74.21,   75.02,   74.66,   73.17,   72.29,   WRLM 670
.     71.40,   70.21,   70.51,   71.19,   71.47,   71.84,   72.21,   WRLM 671
```

```
     .    71.64,    72.21,    72.21,    71.59,    72.43,    72.35,    71.10,    WRLM 672
     .    70.14,    69.47,    67.85,    66.37,    66.07,    66.68,    66.00,    WRLM 673
     .    66.96,    67.99,    67.49,    66.29,    64.96,    63.92,    63.40,    WRLM 674
     .    62.74,    62.07,    61.04,    60.76,    60.50,    59.85,    58.59,    WRLM 675
     .    56.69,    56.00,    55.63,    54.67,    53.93,    52.97,    52.01,    WRLM 676
     .    51.50,    51.19,    51.04,    50.21,    49.49,    48.45,    48.16,    WRLM 677
     .    46.68,    46.31,    45.35,    44.91,    43.95,    41.13,    40.39,    WRLM 678
     .    38.93,    38.31,    37.06,    35.51,    35.06,    33.95,    32.94,    WRLM 679
     .    32.10,    31.28,    30.69,    30.39,    29.95,    28.99,    27.29,    WRLM 680
     .    25.66,    24.03,    22.92,    22.25,    21.00,    19.59,    18.53,    WRLM 681
     .    17.96,    17.59,    16.70,    16.48,    15.15,    14.11,    13.59,    WRLM 682
     .    12.63,    12.26,    11.67,    10.86,    10.04,     8.93,     8.04,    WRLM 683
     .     7.23,     5.30,     4.34,     2.49,     1.38,     1.09,     0.49/    WRLM 684
      DATA B16/     67.42,    68.44,    69.08,    68.98,    69.13,    70.30,    WRLM 685
     .    71.32,    72.15,    72.10,    72.93,    72.49,    71.22,    71.61,    WRLM 686
     .    72.49,    72.64,    71.96,    71.27,    70.49,    69.47,    69.27,    WRLM 687
     .    68.79,    68.54,    67.62,    67.37,    67.62,    67.27,    66.49,    WRLM 688
     .    65.27,    65.37,    66.01,    66.84,    67.03,    66.38,    67.42,    WRLM 689
     .    68.20,    68.98,    68.30,    67.71,    67.27,    66.35,    66.25,    WRLM 690
     .    66.20,    66.20,    66.15,    66.35,    66.74,    67.13,    68.05,    WRLM 691
     .    69.32,    70.30,    71.37,    71.85,    72.30,    72.98,    72.93,    WRLM 692
     .    73.13,    72.54,    71.71,    71.57,    70.93,    70.25,    70.05,    WRLM 693
     .    69.18,    68.54,    68.10,    68.54,    68.74,    68.79,    69.13,    WRLM 694
     .    68.88,    69.23,    69.37,    68.69,    68.49,   -66.45,   -66.54,    WRLM 695
     .   -66.45,   -65.71,   -65.81,   -65.81,   -65.23,   -65.47,   -65.13,    WRLM 696
     .   -65.03,   -65.32,   -65.42,   -65.27,   -65.62,   -66.01,   -66.69,    WRLM 697
     .   -66.20,   -66.54,   -66.25,   -66.69,   -66.49,   -66.64,   -66.98,    WRLM 698
     .   -67.08,   -67.66,   -68.20,   -68.79,   -68.74,   -68.84,   -68.15,    WRLM 699
     .   -67.47,   -67.47,   -67.86,   -68.23,   -68.59,   -68.64,   -69.27,    WRLM 700
     .   -69.37,   -69.47,   -69.27,   -69.71,   -69.52,   -69.27,   -69.52,    WRLM 701
     .   -69.23,   -69.81,   -69.71,   -69.91,   -68.84,   -68.74,   -68.30,    WRLM 702
     .   -68.79,   -69.62,   -69.86,   -69.62,   -69.47,   -69.71,   -69.37,    WRLM 703
     .   -69.52,   -69.03,   -69.66,   -69.52,   -69.62,   -68.93,   -69.18/    WRLM 704
      DATA A17/    359.53,   358.42,   357.15,   356.94,   356.42,   355.90,    WRLM 705
     .   355.02,   354.72,   355.02,   354.13,   352.87,   351.76,   351.98,    WRLM 706
     .   351.39,   350.87,   350.21,   349.17,   347.99,   347.54,   346.80,    WRLM 707
     .   345.76,   345.10,   343.77,   341.77,   340.73,   340.21,   339.18,    WRLM 708
     .   338.66,   337.55,   337.16,   335.75,   334.74,   335.03,   333.35,    WRLM 709
     .   333.11,   333.26,   333.26,   332.15,   331.63,   331.10,   329.43,    WRLM 710
     .   329.19,   327.93,   327.41,   325.49,   324.82,   322.89,   322.01,    WRLM 711
     .   321.19,   321.12,   320.30,   319.85,   319.19,   318.45,   317.71,    WRLM 712
     .   317.12,   315.05,   314.09,   313.42,   312.01,   310.31,   309.42,    WRLM 713
     .   308.24,   307.43,   305.50,   304.24,   302.69,   301.24,   300.39,    WRLM 714
     .   298.71,   298.84,   297.88,   299.28,   298.99,   299.36,   298.91,    WRLM 715
     .   299.43,   299.36,   299.95,   299.43,   298.32,   297.88,   296.69,    WRLM 716
     .   296.55,   295.51,   295.51,   295.21,   294.62,   293.88,   294.84,    WRLM 717
     .   294.92,   296.40,   296.55,   297.29,   297.43,   298.10,   298.84,    WRLM 718
     .   299.65,   300.17,   300.76,   301.65,   300.91,   299.65,   298.77,    WRLM 719
     .   298.03,   297.66,   295.86,   295.21,   293.59,   293.14,   292.55,    WRLM 720
     .   292.03,   291.29,   290.63,   290.59,   291.51,   292.40,   291.88,    WRLM 721
     .   290.55,   289.81,   290.40,   291.37,   292.25,   291.37,   291.29,    WRLM 722
     .   290.77,   290.03,   288.48,   286.63,   286.68,   285.74,   284.63,    WRLM 723
     .   283.15,   281.89,   280.86,   280.86,   280.34,   279.15,   277.82/    WRLM 724
      DATA B17/    -68.93,   -68.40,   -68.69,   -69.37,   -69.27,   -69.37,    WRLM 725
     .   -69.12,   -69.52,   -69.61,   -69.32,   -69.08,   -69.42,   -70.20,    WRLM 726
     .   -70.49,   -70.30,   -70.69,   -70.01,   -69.91,   -70.44,   -70.64,    WRLM 727
```

```
.  -71.03,  -71.32,  -71.47,  -71.32,  -71.96,  -71.86,  -71.85,   WRLM 728
.  -72.25,  -72.15,  -72.74,  -73.01,  -72.79,  -72.25,  -72.00,   WRLM 729
.  -72.35,  -73.08,  -73.71,  -73.71,  -74.30,  -74.49,  -74.49,   WRLM 730
.  -74.76,  -75.03,  -75.32,  -75.42,  -75.96,  -76.54,  -76.98,   WRLM 731
.  -77.81,  -78.25,  -78.49,  -78.74,  -79.03,  -78.53,  -78.83,   WRLM 732
.  -79.03,  -78.83,  -78.78,  -79.17,  -79.22,  -79.47,  -80.00,   WRLM 733
.  -80.59,  -80.54,  -80.10,  -79.76,  -79.66,  -79.03,  -78.25,   WRLM 734
.  -77.47,  -76.93,  -76.20,  -75.03,  -74.05,  -73.22,  -72.83,   WRLM 735
.  -72.15,  -71.57,  -70.98,  -70.44,  -70.10,  -69.47,  -69.32,   WRLM 736
.  -68.64,  -67.96,  -67.52,  -67.13,  -67.13,  -66.64,  -66.45,   WRLM 737
.  -66.42,  -64.98,  -64.49,  -64.49,  -64.49,  -63.57,  -63.71,   WRLM 738
.  -63.23,  -63.42,  -62.65,  -62.30,  -61.62,  -62.54,  -62.01,   WRLM 739
.  -62.45,  -63.13,  -63.42,  -64.69,  -64.84,  -65.27,  -66.40,   WRLM 740
.  -66.45,  -65.71,  -65.86,  -66.15,  -67.13,  -67.62,  -67.76,   WRLM 741
.  -67.76,  -68.49,  -68.79,  -69.66,  -70.49,  -70.64,  -71.96,   WRLM 742
.  -72.35,  -72.54,  -72.74,  -72.69,  -71.91,  -71.22,  -71.52,   WRLM 743
.  -71.61,  -70.98,  -71.22,  -72.00,  -72.35,  -72.05,  -71.63/   WRLM 744
 DATA A18/  276.34,  274.93,  272.71,  271.53,  269.90,  268.50,   WRLM 745
.   266.94,  265.39,  263.83,  262.59,  261.32,  258.80,  257.25,   WRLM 746
.   255.62,  254.14,  253.95,  252.65,  251.77,  251.10,  250.58,   WRLM 747
.   249.47,  248.14,  246.44,  244.96,  243.40,  242.07,  240.00,   WRLM 748
.   238.59,  237.04,  236.08,  234.35,  234.00,  232.82,  230.82,   WRLM 749
.   229.05,  227.34,  226.83,  225.94,  224.38,  223.64,  222.90,   WRLM 750
.   222.01,  220.02,  219.06,  218.17,  217.43,  215.13,  213.80,   WRLM 751
.   213.58,  213.95,  213.43,  212.62,  211.98,  211.80,  210.76,   WRLM 752
.   211.28,  210.69,  210.19,  208.62,  207.58,  205.95,  206.84,   WRLM 753
.   208.32,  209.80,  211.13,  209.73,  210.99,  213.28,  214.98,   WRLM 754
.   216.39,  218.02,  218.83,  217.94,  216.83,  215.13,  213.58,   WRLM 755
.   212.32,  210.91,  210.76,  211.28,  211.88,  213.06,  214.54,   WRLM 756
.   214.32,  215.20,  214.32,  215.13,  214.39,  214.61,  214.61,   WRLM 757
.   213.43,  211.95,  210.99,  210.62,  210.84,  209.43,  209.95,   WRLM 758
.   209.14,  207.58,  205.95,  205.95,  206.40,  205.88,  205.14,   WRLM 759
.   204.18,  204.10,  202.92,  201.44,  200.77,  199.74,  200.63,   WRLM 760
.   200.33,  199.59,  199.74,  198.77,  198.70,  197.96,  196.11,   WRLM 761
.   195.00,  194.11,  193.52,  192.11,  191.37,  189.97,  186.71,   WRLM 762
.   187.30,  185.53,  184.05,  182.57,  181.16,  179.98,  178.72,   WRLM 763
.   177.61,  176.05,  174.28,  173.02,  171.91,  171.54,  170.87/   WRLM 764
 DATA B18/  -71.76,  -71.47,  -71.37,  -71.22,  -71.66,  -71.76,   WRLM 765
.  -71.52,  -71.27,  -71.13,  -70.86,  -70.05,  -70.83,  -70.88,   WRLM 766
.  -71.03,  -71.96,  -72.74,  -72.83,  -72.89,  -73.52,  -73.91,   WRLM 767
.  -73.03,  -72.89,  -72.88,  -73.08,  -72.74,  -72.49,  -72.44,   WRLM 768
.  -72.40,  -72.54,  -72.15,  -72.79,  -73.81,  -74.69,  -74.86,   WRLM 769
.  -74.49,  -73.96,  -73.61,  -73.61,  -74.15,  -74.69,  -74.39,   WRLM 770
.  -74.25,  -74.49,  -74.98,  -74.93,  -74.54,  -74.35,  -75.08,   WRLM 771
.  -75.37,  -76.30,  -76.20,  -75.96,  -76.54,  -76.83,  -76.64,   WRLM 772
.  -76.25,  -75.91,  -76.15,  -75.96,  -76.00,  -76.15,  -77.05,   WRLM 773
.  -77.27,  -77.17,  -77.61,  -77.95,  -78.54,  -78.93,  -79.47,   WRLM 774
.  -79.27,  -79.08,  -80.10,  -80.15,  -80.39,  -80.25,  -80.44,   WRLM 775
.  -79.91,  -80.25,  -81.23,  -82.20,  -82.73,  -83.42,  -84.25,   WRLM 776
.  -84.64,  -85.17,  -85.71,  -86.13,  -86.20,  -86.68,  -87.22,   WRLM 777
.  -87.27,  -87.07,  -86.54,  -86.10,  -85.27,  -85.22,  -84.80,   WRLM 778
.  -84.73,  -84.95,  -84.73,  -85.27,  -85.55,  -85.95,  -85.37,   WRLM 779
.  -85.27,  -85.03,  -84.73,  -85.37,  -85.90,  -86.00,  -85.32,   WRLM 780
.  -84.93,  -85.12,  -84.66,  -84.59,  -84.34,  -84.00,  -83.86,   WRLM 781
.  -83.90,  -83.90,  -83.76,  -83.90,  -83.61,  -83.86,  -83.65,   WRLM 782
.  -83.47,  -83.42,  -83.56,  -83.27,  -83.47,  -83.32,  -83.22,   WRLM 783
```

```
     . -83.12,  -83.37,  -83.17,  -83.22,  -83.27,  -82.83,  -83.08/    WRLM 784
      DATA A19/ 170.80,  170.59,  170.67,  169.24,  169.39,  166.73,    WRLM 785
     .  167.70, 166.95,  165.69,  164.65,  163.84,  164.06,  164.21,    WRLM 786
     .  165.91, 167.54,  168.43,  169.61,  169.69,  170.21,  168.95,    WRLM 787
     .  169.10, 168.13,  167.39,  166.53,  165.25,  163.84,  164.06,    WRLM 788
     .  163.25, 161.92,  160.36,  160.51,  159.77,  159.33,  159.82,    WRLM 789
     .  159.03, 160.21,  161.47,  162.95,  163.69,  164.73,  164.35,    WRLM 790
     .  164.73, 164.21,  162.73,  163.15,  161.92,  162.58,  162.14,    WRLM 791
     .  162.73, 164.43,  164.14,  165.17,  165.17,  165.91,  166.43,    WRLM 792
     .  167.76, 168.73,  170.28,  169.91,  170.65,  169.76,  168.13,    WRLM 793
     .  166.95, 165.76,  165.54,  163.54,  162.36,  160.29,  158.81,    WRLM 794
     .  157.62, 155.48,  153.77,  153.25,  152.89,  152.00,  151.04,    WRLM 795
     .  150.00, 148.52,  146.23,  146.23,  145.86,  144.75,  143.26,    WRLM 796
     .  142.15, 141.19,  140.16,  139.42,  138.45,  136.83,  135.05,    WRLM 797
     .  133.94, 132.46,  131.13,  130.39,  130.16,  128.96,  128.02,    WRLM 798
     .  126.09, 124.91,  123.36,  121.13,  120.39,  118.25,  117.21,    WRLM 799
     .  115.51, 113.66,  112.70,  111.46,  110.77,  110.18,  109.00,    WRLM 800
     .  107.37, 106.26,  104.48,  104.11,  104.04,  102.78,  101.45,    WRLM 801
     .  100.34,  99.23,   97.52,   96.41,   95.85,   94.71,   93.60,    WRLM 802
     .   92.57,  91.75,   90.12,   88.72,   86.87,   85.90,   83.98,    WRLM 803
     .   82.65,  81.39,   80.72,   79.32,   77.54,   76.36,   76.06/    WRLM 804
      DATA B19/ -83.66,  -83.90,  -84.44,  -84.59,  -84.98,  -85.12,    WRLM 805
     .  -85.17, -84.88,  -84.49,  -84.39,  -84.78,  -84.44,  -84.05,    WRLM 806
     .  -84.44, -83.71,  -83.81,  -83.71,  -83.27,  -83.12,  -82.88,    WRLM 807
     .  -82.64, -82.64,  -82.83,  -82.49,  -82.20,  -82.00,  -81.76,    WRLM 808
     .  -81.32, -81.61,  -81.51,  -80.93,  -80.49,  -80.34,  -79.55,    WRLM 809
     .  -79.08, -77.95,  -77.76,  -77.65,  -77.81,  -77.61,  -77.17,    WRLM 810
     .  -76.93, -76.54,  -76.20,  -75.76,  -75.37,  -74.39,  -74.35,    WRLM 811
     .  -74.10, -74.10,  -73.76,  -73.42,  -73.27,  -72.83,  -72.40,    WRLM 812
     .  -72.30, -72.49,  -72.25,  -71.71,  -71.42,  -70.30,  -70.20,    WRLM 813
     .  -69.86, -69.62,  -69.27,  -69.62,  -69.23,  -69.42,  -68.84,    WRLM 814
     .  -68.35, -68.25,  -67.81,  -67.96,  -67.86,  -67.76,  -67.52,    WRLM 815
     .  -67.81, -67.52,  -67.57,  -66.93,  -66.69,  -66.35,  -66.20,    WRLM 816
     .  -66.35, -65.86,  -65.81,  -65.23,  -65.81,  -65.62,  -65.32,    WRLM 817
     .  -65.32, -65.13,  -65.13,  -65.32,  -65.57,  -65.67,  -65.37,    WRLM 818
     .  -65.37, -65.86,  -66.40,  -66.40,  -66.01,  -66.10,  -65.27,    WRLM 819
     .  -65.27, -65.13,  -64.88,  -64.93,  -64.54,  -64.88,  -65.42,    WRLM 820
     .  -65.76, -65.42,  -65.96,  -65.76,  -64.98,  -64.69,  -64.84,    WRLM 821
     .  -65.52, -65.52,  -65.42,  -65.42,  -65.13,  -65.47,  -65.23,    WRLM 822
     .  -65.37, -65.42,  -65.57,  -65.86,  -65.71,  -66.01,  -66.01,    WRLM 823
     .  -66.40, -66.45,  -67.13,  -67.03,  -67.96,  -68.84,  -69.23/    WRLM 824
      DATA A20/  75.23,   74.58,   73.17,   72.43,   70.95,   69.77,    WRLM 825
     .   68.44,  67.03,   65.18,   63.48,   61.78,   60.37,  117.95,    WRLM 826
     .  117.14, 116.32,  115.07,  115.81,  116.18,  115.51,  115.81,    WRLM 827
     .  115.21, 114.92,  114.47,  114.33,  113.81,  114.18,  113.44,    WRLM 828
     .  114.33, 114.33,  113.88,  113.88,  113.22,  113.59,  113.44,    WRLM 829
     .  113.36, 114.18,  114.47,  115.29,  116.18,  117.58,  118.40,    WRLM 830
     .  118.84, 120.25,  121.58,  121.50,  122.54,  122.76,  122.69,    WRLM 831
     .  122.99, 123.73,  124.32,  124.61,  124.32,  125.35,  125.50,    WRLM 832
     .  126.54, 127.05,  128.31,  128.98,  130.02,  130.16,  130.53,    WRLM 833
     .  130.98, 130.83,  131.35,  132.16,  133.05,  133.13,  132.66,    WRLM 834
     .  133.57, 134.61,  135.72,  136.38,  137.27,  137.86,  137.71,    WRLM 835
     .  136.90, 136.16,  136.46,  135.64,  136.97,  136.90,  138.01,    WRLM 836
     .  138.62, 139.56,  141.04,  141.34,  141.78,  141.64,  142.15,    WRLM 837
     .  142.08, 141.78,  142.52,  142.69,  143.04,  143.12,  144.09,    WRLM 838
     .  143.86, 144.23,  142.93,  144.36,  145.04,  145.78,  146.60,    WRLM 839
```

```
     .  146.23,  146.32,  146.37,  147.25,  146.90,  146.45,  148.37,     WRLM 840
     .  149.96,  150.00,  149.48,  150.81,  150.89,  151.48,  151.46,     WRLM 841
     .  152.15,  152.96,  153.40,  153.11,  153.18,  154.11,  153.45,     WRLM 842
     .  152.89,  153.63,  154.44,  153.70,  154.14,  153.55,  152.69,     WRLM 843
     .  153.03,  153.11,  152.44,  152.52,  152.29,  152.44,  151.63/     WRLM 844
      DATA B20/  -69.37,  -69.37,  -69.13,  -68.79,  -68.15,  -67.37,     WRLM 845
     .  -67.47,  -66.98,  -67.08,  -65.84,  -66.98,  -66.69,  -34.41,     WRLM 846
     .  -34.63,  -34.20,  -33.77,  -32.94,  -32.31,  -31.77,  -30.59,     WRLM 847
     .  -30.36,  -29.82,  -28.70,  -27.72,  -27.33,  -26.99,  -25.31,     WRLM 848
     .  -25.68,  -24.65,  -24.26,  -23.73,  -23.34,  -22.70,  -21.87,     WRLM 849
     .  -20.46,  -21.68,  -20.90,  -20.93,  -20.26,  -20.26,  -20.41,     WRLM 850
     .  -19.68,  -19.58,  -19.29,  -18.41,  -18.02,  -17.29,  -16.61,     WRLM 851
     .  -16.17,  -16.90,  -16.22,  -15.87,  -15.34,  -14.60,  -13.97,     WRLM 852
     .  -13.97,  -13.53,  -14.41,  -14.22,  -14.65,  -13.73,  -13.24,     WRLM 853
     .  -12.90,  -12.36,  -11.97,  -12.61,  -12.22,  -11.87,  -11.05,     WRLM 854
     .  -11.44,  -11.92,  -11.97,  -11.73,  -12.07,  -11.67,  -12.65,     WRLM 855
     .  -12.95,  -13.44,  -13.78,  -14.22,  -14.95,  -15.39,  -15.78,     WRLM 856
     .  -16.17,  -17.24,  -16.95,  -16.26,  -15.24,  -14.22,  -13.14,     WRLM 857
     .  -12.75,  -11.92,  -11.53,  -10.75,  -10.31,  -11.24,  -11.48,     WRLM 858
     .  -12.02,  -12.85,  -13.58,  -14.07,  -14.12,  -14.80,  -15.14,     WRLM 859
     .  -16.07,  -17.00,  -18.21,  -18.70,  -18.70,  -19.34,  -19.97,     WRLM 860
     .  -20.21,  -20.26,  -21.24,  -21.39,  -22.31,  -22.70,  -23.49,     WRLM 861
     .  -24.07,  -24.51,  -25.33,  -25.33,  -25.68,  -26.02,  -26.31,     WRLM 862
     .  -26.55,  -26.99,  -27.38,  -28.41,  -29.19,  -29.63,  -30.16,     WRLM 863
     .  -30.55,  -31.04,  -31.63,  -32.05,  -32.55,  -32.00,  -33.67/     WRLM 864
      DATA A21/  151.04,  150.44,  150.30,  149.56,  148.74,  147.56,     WRLM 865
     .  146.89,  145.86,  144.75,  143.86,  142.97,  142.08,  141.34,     WRLM 866
     .  139.86,  139.56,  139.46,  139.12,  139.56,  139.71,  139.34,     WRLM 867
     .  138.97,  138.16,  137.49,  137.49,  138.08,  138.01,  138.75,     WRLM 868
     .  137.71,  137.49,  136.75,  135.79,  136.09,  135.05,  135.12,     WRLM 869
     .  134.09,  133.27,  132.16,  131.50,  130.61,  129.20,  127.43,     WRLM 870
     .  127.43,  125.80,  124.61,  124.17,  123.28,  122.76,  121.65,     WRLM 871
     .  121.13,  119.43,  118.25,  117.29,  116.32,  295.44,  296.03,     WRLM 972
     .  297.58,  298.47,  296.47,  296.18,  297.06,  297.73,  298.40,     WRLM 873
     .  299.14,  300.91,  300.32,  300.76,  302.39,  303.21,  304.10,     WRLM 874
     .  305.28,  305.72,  305.87,  305.65,  305.65,  306.69,  307.43,     WRLM 875
     .  308.31,  309.20,  309.94,  311.72,  313.05,  313.05,  313.87,     WRLM 876
     .  314.31,  314.90,  315.86,  316.46,  315.72,  315.27,  314.68,     WRLM 877
     .  314.24,  313.94,  314.68,  315.64,  316.60,  317.27,  318.16,     WRLM 878
     .  318.23,  319.19,  319.64,  320.45,  320.90,  320.45,  320.16,     WRLM 879
     .  320.75,  321.19,  322.82,  324.45,  325.26,  325.86,  326.52,     WRLM 880
     .  327.41,  328.22,  329.33,  329.70,  330.44,  331.70,  332.81,     WRLM 881
     .  332.29,  332.07,  333.33,  334.44,  335.25,  336.74,  338.22,     WRLM 882
     .  339.18,  338.59,  337.70,  336.81,  335.92,  334.74,  335.25,     WRLM 883
     .  336.07,  336.88,  337.92,  337.70,  336.59,  336.00,  337.19/     WRLM 884
      DATA B21/  -34.60,  -35.53,  -36.26,  -37.19,  -37.33,  -37.87,     WRLM 885
     .  -38.36,  -38.36,  -37.92,  -38.55,  -38.26,  -37.87,  -37.53,     WRLM 886
     .  -37.23,  -36.55,  -35.87,  -35.33,  -34.89,  -33.63,  -34.65,     WRLM 887
     .  -33.97,  -34.21,  -34.60,  -33.82,  -33.43,  -32.59,  -32.50,     WRLM 888
     .  -32.26,  -32.84,  -32.99,  -33.19,  -34.21,  -33.87,  -33.09,     WRLM 889
     .  -31.87,  -31.38,  -31.24,  -30.60,  -30.99,  -31.58,  -31.38,     WRLM 890
     .  -31.97,  -31.82,  -32.41,  -33.04,  -33.43,  -33.77,  -33.87,     WRLM 891
     .  -34.11,  -34.02,  -34.45,  -34.70,  -34.41,   82.59,   82.78,     WRLM 892
     .   82.93,   82.34,   81.86,   80.98,   80.93,   80.59,   81.17,     WRLM 893
     .   81.47,   81.08,   81.42,   81.76,   81.90,   82.44,   82.10,     WRLM 894
     .   82.25,   82.59,   82.15,   81.71,   81.42,   81.71,   81.37,     WRLM 895
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
     .   81.66,   82.54,   81.95,   82.10,   81.37,   81.95,   81.61,    WRLM 896
     .   81.37,   81.08,   81.17,   81.42,   81.61,   81.55,   81.51,    WRLM 897
     .   81.01,   82.29,   82.73,   82.83,   83.08,   83.27,   82.93,    WRLM 898
     .   82.59,   82.30,   82.64,   82.34,   82.49,   82.64,   82.93,    WRLM 899
     .   83.22,   83.17,   83.37,   83.51,   83.32,   83.08,   83.22,    WRLM 900
     .   83.51,   83.51,   82.22,   83.61,   83.32,   83.61,   83.42,    WRLM 901
     .   83.32,   82.73,   82.88,   83.03,   82.59,   82.83,   83.03,    WRLM 902
     .   82.64,   82.39,   82.25,   82.00,   81.66,   81.42,   81.12,    WRLM 903
     .   81.32,   81.76,   81.66,   81.27,   81.12,   80.54,   80.73/    WRLM 904
      DATA A22/  337.48,  340.07,  340.51,  340.95,  341.18,  341.47,    WRLM 905
     .  342.43,  343.25,  344.60,  345.75,  346.36,  347.91,  346.50,    WRLM 906
     .  346.43,  345.02,  344.28,  344.43,  343.47,  343.03,  343.10,    WRLM 907
     .  342.73,  342.51,  342.14,  340.65,  340.58,  339.62,  340.36,    WRLM 908
     .  340.07,  338.66,  339.55,  340.07,  338.59,  338.66,  339.99,    WRLM 909
     .  338.96,  337.62,  337.70,  338.36,  339.33,  339.62,  338.88,    WRLM 910
     .  337.85,  337.25,  335.00,  334.44,  333.26,  332.44,  332.22,    WRLM 911
     .  333.48,  334.89,  336.51,  337.40,  337.70,  338.07,  337.85,    WRLM 912
     .  337.18,  336.29,  335.18,  334.22,  333.33,  332.52,  332.07,    WRLM 913
     .  331.63,  331.26,  332.00,  332.89,  334.00,  335.03,  336.22,    WRLM 914
     .  335.63,  335.33,  333.70,  332.66,  331.92,  331.11,  330.00,    WRLM 915
     .  329.11,  328.30,  327.41,  326.60,  326.08,  324.82,  324.45,    WRLM 916
     .  323.34,  322.38,  321.64,  320.45,  320.16,  319.34,  319.19,    WRLM 917
     .  318.16,  318.08,  317.71,  317.34,  316.46,  315.86,  315.05,    WRLM 918
     .  313.50,  313.13,  312.53,  312.01,  310.90,  310.31,  309.65,    WRLM 919
     .  310.24,  309.65,  309.42,  309.35,  307.80,  307.72,  306.91,    WRLM 920
     .  307.57,  307.57,  308.09,  308.02,  307.20,  306.32,  306.83,    WRLM 921
     .  306.54,  307.50,  308.46,  309.24,  309.20,  308.91,  308.00,    WRLM 922
     .  307.23,  306.60,  305.95,  305.65,  305.28,  305.95,  306.32,    WRLM 923
     .  307.13,  307.65,  306.91,  306.76,  307.20,  308.31,  309.35/    WRLM 924
      DATA B22/   81.12,   81.66,   82.15,   81.56,   81.27,   81.47,    WRLM 925
     .   81.47,   81.81,   82.20,   82.29,   81.90,   81.91,   81.51,    WRLM 926
     .   80.88,   80.78,   80.73,   80.10,   80.30,   80.30,   79.71,    WRLM 927
     .   79.61,   79.81,   78.93,   78.83,   77.95,   77.47,   76.54,    WRLM 928
     .   76.15,   76.30,   75.91,   75.17,   75.47,   74.74,   73.66,    WRLM 929
     .   73.71,   74.25,   73.66,   73.57,   73.37,   72.74,   72.69,    WRLM 930
     .   73.27,   73.22,   73.57,   72.98,   73.37,   73.27,   72.59,    WRLM 931
     .   72.20,   72.69,   72.10,   71.91,   71.52,   71.22,   70.88,    WRLM 932
     .   70.98,   70.69,   71.32,   71.57,   71.47,   71.08,   71.22,    WRLM 933
     .   70.74,   70.35,   70.15,   70.40,   70.20,   70.25,   69.91,    WRLM 934
     .   69.62,   68.93,   68.93,   68.79,   68.49,   68.54,   68.25,    WRLM 935
     .   68.05,   67.47,   67.66,   67.47,   66.88,   66.55,   65.86,    WRLM 936
     .   65.62,   65.42,   65.18,   64.98,   64.59,   64.10,   63.67,    WRLM 937
     .   62.79,   62.11,   61.76,   61.32,   60.45,   59.67,   59.52,    WRLM 938
     .   59.57,   60.06,   60.35,   60.15,   60.11,   59.57,   59.81,    WRLM 939
     .   60.84,   61.23,   62.15,   62.69,   62.80,   63.25,   63.71,    WRLM 940
     .   64.10,   65.13,   65.42,   65.91,   65.86,   66.15,   66.74,    WRLM 941
     .   67.86,   67.57,   67.52,   68.30,   68.54,   69.13,   69.42,    WRLM 942
     .   69.18,   68.69,   68.74,   69.32,   70.15,   70.15,   69.71,    WRLM 943
     .   69.47,   69.57,   70.44,   70.74,   70.54,   70.15,   69.96/    WRLM 944
      DATA A23/  309.35,  308.61,  308.24,  307.13,  306.46,  305.72,    WRLM 945
     .  305.72,  304.54,  304.24,  304.61,  304.32,  304.17,  303.13,    WRLM 946
     .  303.21,  302.47,  302.39,  301.65,  301.06,  300.02,  299.05,    WRLM 947
     .  298.64,  296.32,  296.69,  294.62,  294.85,  293.51,  292.25,    WRLM 948
     .  292.03,  290.25,  290.40,  289.66,  289.81,  291.29,  292.40,    WRLM 949
     .  291.96,  290.33,  288.77,  288.70,  287.59,  286.63,  286.48,    WRLM 950
     .  287.00,  287.59,  286.33,  288.92,  289.44,  289.81,  291.81,    WRLM 951
```

```
     .   292.77,  293.73,  292.55,  291.23,  292.25,  293.66,   24.03,   WRLM 952
     .    22.40,   21.59,   21.14,   19.65,   19.37,   19.14,   19.74,   WRLM 953
     .    19.59,   20.46,   21.00,   20.19,   20.11,   19.37,   16.70,   WRLM 954
     .    16.48,   17.56,   16.85,   16.41,   16.25,   15.74,   16.63,   WRLM 955
     .    15.44,   14.63,   13.52,   13.67,   12.19,   12.11,   11.15,   WRLM 956
     .    10.63,   11.23,   12.71,   14.25,   16.11,   17.52,   16.48,   WRLM 957
     .    17.37,   19.00,   20.63,   21.14,   21.74,   22.43,   23.59,   WRLM 958
     .    24.77,   25.21,   24.18,  344.89,  345.84,  345.17,  343.84,   WRLM 959
     .   343.10,  342.14,  342.66,  341.62,  340.58,  339.92,  339.25,   WRLM 960
     .   339.77,  338.88,  336.74,  337.62,  336.31,  335.92,  335.25,   WRLM 961
     .   336.51,  337.25,  335.70,  335.03,  336.29,  336.96,  338.29,   WRLM 962
     .   338.96,  340.07,  340.21,  341.40,  342.58,  343.54,  344.28,   WRLM 963
     .   344.14,  344.55,  345.47,  292.99,  291.60,  290.11,  288.63/   WRLM 964
      DATA B23/   70.40,   70.59,   70.83,   71.08,   71.37,   71.96,   WRLM 965
     .    71.61,   71.03,   71.57,   71.81,   72.30,   72.98,   73.42,   WRLM 966
     .    74.20,   74.30,   74.64,   74.98,   75.47,   75.47,   75.71,   WRLM 967
     .    75.76,   75.96,   75.81,   75.71,   75.22,   75.32,   75.52,   WRLM 968
     .    75.32,   75.81,   75.96,   76.44,   76.93,   76.74,   76.93,   WRLM 969
     .    77.27,   77.13,   77.56,   77.13,   77.47,   77.76,   77.95,   WRLM 970
     .    78.05,   78.05,   78.20,   78.15,   78.69,   79.27,   79.12,   WRLM 971
     .    79.22,   79.32,   79.52,   79.66,   80.20,   80.39,   78.54,   WRLM 972
     .    78.69,   79.03,   79.03,   79.22,   78.93,   78.69,   78.64,   WRLM 973
     .    78.44,   78.34,   77.91,   77.65,   76.93,   77.52,   77.55,   WRLM 974
     .    77.47,   77.32,   76.15,   76.00,   76.69,   77.27,   78.00,   WRLM 975
     .    77.86,   77.27,   77.32,   78.13,   78.25,   78.96,   79.22,   WRLM 976
     .    79.95,   80.15,   79.61,   79.03,   78.78,   79.03,   79.47,   WRLM 977
     .    79.95,   79.95,   79.71,   79.95,   80.00,   80.30,   79.81,   WRLM 978
     .    79.61,   79.08,   78.64,   66.25,   64.59,   64.20,   63.62,   WRLM 979
     .    63.86,   63.62,   63.18,   62.79,   62.64,   62.40,   62.45,   WRLM 980
     .    62.98,   63.08,   63.23,   63.71,   64.15,   64.25,   64.59,   WRLM 981
     .    64.69,   64.79,   65.18,   65.76,   66.40,   65.57,   65.19,   WRLM 982
     .    65.96,   65.71,   65.96,   65.76,   65.96,   66.10,   65.91,   WRLM 983
     .    65.37,   65.32,   64.69,   61.47,   61.71,   62.11,   62.79/   WRLM 984
      DATA A24/  287.89,  286.65,  285.59,  283.96,  282.11,  281.00,   WRLM 985
     .   281.23,  282.34,  283.45,  284.26,  285.15,  285.59,  285.59,   WRLM 986
     .   285.37,  285.81,  285.52,  286.04,  284.78,  283.96,  283.74,   WRLM 987
     .   282.34,  281.00,  279.67,  280.12,  279.82,  279.08,  277.82,   WRLM 988
     .   277.01,  276.04,  275.23,  274.27,  273.97,  272.71,  272.05,   WRLM 989
     .   272.20,  271.16,  270.42,  270.20,  270.42,  270.57,  271.38,   WRLM 990
     .   271.75,  272.42,  273.97,  275.16,  275.01,  274.56,  274.49,   WRLM 991
     .   274.86,  275.53,  275.53,  275.82,  275.01,  275.38,  275.38,   WRLM 992
     .   276.86,  278.34,  278.12,  277.97,  278.41,  279.00,  279.67,   WRLM 993
     .   279.00,  279.63,  279.23,  280.12,  280.71,  281.52,  282.34,   WRLM 994
     .   283.15,  282.71,  283.08,  283.45,  284.41,  284.93,  285.67,   WRLM 995
     .   286.18,  287.52,  286.92,  287.81,  288.40,  289.44,  290.48,   WRLM 996
     .   291.29,  292.33,  291.51,  291.66,  293.07,  294.33,  295.51,   WRLM 997
     .   295.67,  295.95,  296.32,  297.06,  296.99,  298.10,  298.69,   WRLM 998
     .   298.32,  296.10,  294.70,  293.22,  292.25,  290.85,  291.51,   WRLM 999
     .   292.65,  294.10,  295.44,  295.14,  295.44,  294.62,  292.99,   WRLM1000
     .   291.95,  290.55,  291.22,  292.70,  292.99,  293.83,  293.29,   WRLM1001
     .   280.41,  279.08,  278.64,  276.12,  274.42,  273.01,  272.05,   WRLM1002
     .   271.09,  270.05,  269.90,  270.42,  270.86,  272.05,  273.38,   WRLM1003
     .   274.49,  275.01,  276.64,  278.30,  275.67,  276.71,  278.19/   WRLM1004
      DATA B24/   63.23,   63.91,   64.35,   63.91,   64.01,   64.01,   WRLM1005
     .    64.35,   65.08,   64.84,   65.08,   64.88,   65.03,   65.32,   WRLM1006
     .    66.35,   66.74,   68.15,   66.83,   68.59,   69.37,   69.85,   WRLM1007
```

```
     .    69.91,    69.37,    69.27,    69.65,    69.91/,   69.57,    69.62,     WRLM1008
     .    69.47,    69.57,    70.05,    69.81,    70.05,    70.20,    70.64,     WRLM1009
     .    70.63,    71.18,    71.52,    72.25,    72.35,    72.59,    73.12,     WRLM1010
     .    73.60,    73.81,    73.66,    74.10,    73.61,    72.79,    71.81,     WRLM1011
     .    71.42,    71.22,    70.74,    71.13,    71.52,    71.91,    72.40,     WRLM1012
     .    73.52,    73.13,    72.54,    72.05,    72.20,    72.44,    72.40,     WRLM1013
     .    72.83,    73.08,    73.47,    73.42,    73.52,    73.08,    73.03,     WRLM1014
     .    72.74,    72.25,    71.91,    72.00,    71.71,    71.61,    71.13,     WRLM1015
     .    71.08,    71.61,    70.93,    70.64,    70.59,    70.59,    69.95,     WRLM1016
     .    70.05,    69.32,    68.98,    67.65,    67.23,    67.32,    67.13,     WRLM1017
     .    66.69,    66.79,    67.23,    67.13,    66.45,    66.59,    66.06,     WRLM1018
     .    65.80,    65.52,    64.98,    65.71,    66.20,    65.91,    65.18,     WRLM1019
     .    64.30,    63.67,    63.57,    63.08,    62.20,    62.35,    62.59,     WRLM1020
     .    63.47,    63.76,    62.74,    62.59,    62.11,    51.96,    61.81,     WRLM1021
     .    76.20,    76.10,    76.39,    76.30,    75.91,    75.56,    76.20,     WRLM1022
     .    75.91,    75.56,    76.15,    76.69,    76.93,    77.17,    76.83,     WRLM1023
     .    76.88,    76.69,    77.17,    77.86,    78.05,    77.91,    78.25/     WRLM1024
      DATA A25/   277.15,   275.75,   275.03,   274.56,   273.75,   274.19,    WRLM1025
     .   273.97,   274.19,   275.97,   277.15,   278.71,   279.97,   281.97,    WRLM1026
     .   281.23,   280.63,   279.52,   279.00,   277.60,   276.55,   275.60,    WRLM1027
     .   275.01,   273.60,   272.49,   271.60,   269.90,   269.75,   270.94,    WRLM1028
     .   272.49,   273.23,   274.27,   275.60,   276.56,   278.49,   279.57,    WRLM1029
     .   280.71,   281.97,   283.52,   285.07,   286.18,   287.44,   289.59,    WRLM1030
     .   291.29,   292.03,   290.55,   291.37,   293.22,   291.59,   289.88,    WRLM1031
     .   288.77,   288.40,   286.41,   285.22,   285.15,   283.45,   282.85,    WRLM1032
     .   283.22,   284.41,   283.89,   282.93,   282.41,   281.15,   280.41,    WRLM1033
     .   279.82,   278.93,   278.12,   276.78,   276.19,   274.64,   273.45,    WRLM1034
     .   272.20,   270.64,   269.75,   269.38,   269.90,   270.12,   270.12,    WRLM1035
     .   270.49,   271.09,   271.68,   272.34,   272.20,   272.49,   271.75,    WRLM1036
     .   271.38,   270.49,   269.46,   268.27,   267.31,   266.87,   266.35,    WRLM1037
     .   266.27,   265.39,   265.16,   265.90,   266.35,   267.83,   269.01,    WRLM1038
     .   269.61,   269.61,   269.46,   268.35,   267.09,   267.63,   268.42,    WRLM1039
     .   268.50,   269.30,   270.57,   271.31,   273.01,   274.19,   275.97,    WRLM1040
     .   277.23,   278.26,   278.78,   279.45,   278.75,   279.97,   257.10,    WRLM1041
     .   255.84,   254.43,   252.73,   251.47,   251.62,   250.58,   249.62,    WRLM1042
     .   248.59,   247.92,   246.88,   245.63,   244.66,   242.89,   242.74,    WRLM1043
     .   244.29,   246.44,   247.55,   248.22,   247.85,   246.37,   244.59/    WRLM1044
      DATA B25/    78.25,    78.39,    78.25,    78.73,    78.78,    78.98,    WRLM1045
     .    79.32,    79.42,    79.66,    78.98,    79.61,    79.91,    80.20,     WRLM1046
     .    80.80,    81.47,    80.90,    80.44,    80.39,    80.39,    80.98,     WRLM1047
     .    80.73,    80.44,    80.54,    80.69,    80.93,    81.27,    81.61,     WRLM1048
     .    81.51,    81.47,    82.29,    82.10,    82.64,    82.39,    82.73,     WRLM1049
     .    82.64,    82.78,    82.64,    82.29,    82.98,    82.44,    82.59,     WRLM1050
     .    82.73,    82.15,    81.22,    81.32,    81.22,    80.54,    80.30,     WRLM1051
     .    79.61,    78.98,    79.27,    78.73,    78.39,    78.88,    78.69,     WRLM1052
     .    78.10,    77.91,    77.56,    77.66,    77.13,    77.08,    76.59,     WRLM1053
     .    75.96,    75.91,    76.25,    76.00,    76.05,    75.76,    75.52,     WRLM1054
     .    75.81,    75.47,    75.52,    76.10,    76.69,    77.17,    77.56,     WRLM1055
     .    78.00,    77.86,    77.91,    78.49,    79.17,    79.32,    79.71,     WRLM1056
     .    79.61,    80.25,    80.34,    80.49,    80.83,    81.08,    80.98,     WRLM1057
     .    80.76,    79.95,    79.52,    78.98,    78.05,    77.91,    77.76,     WRLM1058
     .    77.37,    76.78,    76.20,    76.25,    76.30,    75.71,    74.93,     WRLM1059
     .    73.81,    73.96,    74.10,    74.83,    74.74,    74.88,    75.08,     WRLM1060
     .    74.69,    74.93,    74.74,    74.83,    74.59,    75.51,    68.40,     WRLM1061
     .    68.98,    69.18,    68.81,    68.98,    68.15,    68.59,    68.79,     WRLM1062
     .    68.69,    68.84,    68.79,    69.47,    69.37,    69.52,    70.10,     WRLM1063
```

```
     . 70.20,   70.15,   70.10,   70.49,   70.03,   70.83,   70.83/  WRLM1064
 DATA A26/     243.40,  244.37,  244.65,  243.70,  242.96,  242.44,  WRLM1065
     . 243.18, 244.22,  245.40,  246.44,  247.35,  248.59,  249.25,  WRLM1066
     . 251.03, 251.55,  251.62,  252.06,  252.36,  253.47,  254.95,  WRLM1067
     . 255.17, 256.36,  256.36,  257.02,  258.06,  256.55,  257.10,  WRLM1068
     . 244.22, 243.33,  241.85,  240.22,  239.33,  238.67,  237.04,  WRLM1069
     . 235.86, 234.75,  235.63,  235.55,  235.93,  236.60,  237.85,  WRLM1070
     . 238.89, 240.15,  240.59,  241.33,  242.59,  244.00,  253.99,  WRLM1071
     . 252.58, 251.47,  250.58,  249.84,  249.33,  247.92,  245.77,  WRLM1072
     . 246.22, 247.70,  246.44,  244.74,  242.31,  243.18,  244.29,  WRLM1073
     . 244.81, 245.63,  247.03,  247.85,  249.03,  251.10,  252.14,  WRLM1074
     . 254.14, 254.58,  254.06,  263.31,  261.98,  261.02,  260.59,  WRLM1075
     . 259.32, 259.02,  257.32,  257.17,  258.21,  259.21,  258.87,  WRLM1076
     . 258.21, 259.69,  259.84,  260.13,  261.51,  262.57,  262.13,  WRLM1077
     . 263.39, 263.17,  269.01,  268.20,  266.37,  265.24,  263.98,  WRLM1078
     . 264.05, 264.57,  266.13,  266.79,  267.46,  268.57,  269.09,  WRLM1079
     . 269.09, 255.62,  257.02,  257.99,  259.24,  259.39,  261.24,  WRLM1080
     . 261.54, 260.21,  258.50,  257.84,  257.10,  255.99,  255.69,  WRLM1081
     . 235.93, 235.86,  236.45,  237.41,  238.08,  239.78,  240.59,  WRLM1082
     . 241.78, 242.52,  242.37,  241.26,  240.15,  239.93,  238.82,  WRLM1083
     . 238.00, 237.04,  236.45,  235.78,  260.58,  259.76,  259.76/  WRLM1084
 DATA B26/      70.93,   71.32,   71.52,   71.66,   71.47,   71.52,  WRLM1085
     .  72.15,  71.96,   72.59,   72.15,   72.49,   72.98,   72.35,  WRLM1086
     .  72.59,  72.05,   71.76,   71.75,   72.74,   72.64,   71.57,  WRLM1087
     .  71.13,  70.93,   70.64,   70.59,   69.31,   69.42,   60.64,  WRLM1088
     .  72.59,  72.25,   72.30,   71.42,   70.93,   70.98,   70.83,  WRLM1089
     .  71.22,  71.91,   72.50,   73.13,   73.76,   73.91,   73.81,  WRLM1090
     .  73.27,  73.57,   73.13,   73.52,   72.98,   72.88,   74.59,  WRLM1091
     .  74.44,  74.25,   74.39,   73.86,   74.10,   73.66,   73.71,  WRLM1092
     .  74.10,  74.44,   74.69,   74.73,   75.03,   75.17,   75.01,  WRLM1093
     .  75.71,  76.05,   75.42,   75.47,   75.13,   75.91,   75.52,  WRLM1094
     .  75.66,  75.32,   74.74,   71.03,   71.18,   70.63,   71.52,  WRLM1095
     .  71.32,  71.96,   72.10,   72.83,   72.59,   72.64,   72.79,  WRLM1096
     .  73.47,  73.71,   74.15,   73.37,   73.32,   73.08,   72.59,  WRLM1097
     .  72.40,  71.52,   72.44,   72.35,   72.30,   72.10,   72.35,  WRLM1098
     .  73.08,  73.47,   73.18,   73.57,   73.37,   73.47,   73.32,  WRLM1099
     .  72.93,  75.61,   75.47,   75.86,   75.56,   76.00,   76.00,  WRLM1100
     .  74.49,  74.54,   74.69,   74.39,   74.74,   74.93,   75.66,  WRLM1101
     .  75.56,  75.86,   76.25,   76.30,   76.54,   77.03,   76.93,  WRLM1102
     .  77.13,  76.49,   76.05,   76.05,   75.91,   75.47,   75.42,  WRLM1103
     .  75.27,  75.47,   75.42,   75.71,   76.88,   76.69,   77.47/  WRLM1104
 DATA A27/     258.65,  257.39,  256.29,  256.50,  256.88,  257.17,  WRLM1105
     . 258.21, 258.95,  259.54,  260.13,  260.87,  260.43,  260.58,  WRLM1106
     . 250.81, 249.25,  249.84,  249.92,  248.59,  247.48,  247.70,  WRLM1107
     . 249.10, 250.44,  251.25,  251.03,  262.80,  261.83,  260.73,  WRLM1108
     . 260.87, 261.32,  261.76,  262.65,  266.57,  265.90,  265.24,  WRLM1109
     . 263.83, 263.98,  263.09,  263.98,  264.79,  265.39,  266.50,  WRLM1110
     . 266.64, 265.61,  264.57,  263.46,  263.31,  263.98,  264.79,  WRLM1111
     . 265.61, 266.72,  265.39,  263.83,  264.72,  265.31,  265.83,  WRLM1112
     . 266.42, 264.13,  263.31,  262.72,  262.13,  261.46,  261.02,  WRLM1113
     . 261.39, 262.13,  263.24,  263.76,  263.98,  264.28,  150.09,  WRLM1114
     . 149.93, 149.26,  148.30,  147.71,  147.15,  146.97,  146.45,  WRLM1115
     . 146.15, 145.26,  144.60,  143.93,  144.00,  143.19,  142.23,  WRLM1116
     . 141.64, 141.41,  140.67,  139.79,  139.42,  139.42,  138.38,  WRLM1117
     . 138.45, 138.82,  138.07,  138.82,  138.35,  138.31,  138.31,  WRLM1118
     . 137.64, 137.27,  136.90,  135.45,  135.72,  136.46,  135.42,  WRLM1119
```

. 135.12,	134.61,	133.94,	132.93,	133.25,	132.68,	132.01,	WRLM1120
. 131.72,	121.32,	130.90,	131.27,	131.94,	131.72,	132.46,	WRLM1121
. 133.27,	132.16,	133.13,	134.16,	134.34,	134.90,	135.56,	WRLM1122
. 136.53,	138.01,	139.56,	140.30,	141.19,	142.75,	143.63,	WRLM1123
. 144.52,	145.49,	146.23,	146.45,	146.89,	147.63,	148.45/	WRLM1124
DATA B27/	77.42,	77.47,	77.61,	78.15,	78.10,	78.30,	WRLM1125
. 76.25,	78.64,	78.34,	78.39,	77.91,	77.66,	76.84,	WRLM1126
. 77.03,	76.93,	77.52,	77.91,	77.95,	78.10,	78.34,	WRLM1127
. 76.34,	78.54,	77.36,	77.17,	78.33,	78.73,	79.03,	WRLM1128
. 79.71,	79.52,	79.37,	78.93,	73.81,	73.96,	73.76,	WRLM1129
. 73.86,	74.15,	74.69,	74.74,	74.93,	74.59,	74.50,	WRLM1130
. 73.96,	77.17,	76.76,	76.88,	77.27,	77.47,	77.61,	WRLM1131
. 77.27,	76.20,	75.81,	75.91,	76.35,	76.20,	76.56,	WRLM1132
. 76.30,	68.40,	68.49,	68.40,	68.74,	68.69,	68.98,	WRLM1133
. 69.37,	69.66,	69.47,	69.18,	68.83,	68.40,	-10.27,	WRLM1134
. -10.12,	-9.92,	-9.92,	-9.34,	-9.14,	-8.66,	-8.27,	WRLM1135
. -8.02,	-7.58,	-8.02,	-8.02,	-8.85,	-9.29,	-9.24,	WRLM1136
. -8.95,	-8.17,	-8.07,	-8.12,	-7.97,	-7.39,	-7.49,	WRLM1137
. -7.19,	-7.10,	-6.71,	-6.27,	-5.97,	-5.39,	-4.95,	WRLM1138
. -4.56,	-4.27,	-4.22,	-3.68,	-4.12,	-4.46,	-4.02,	WRLM1139
. -3.49,	-3.78,	-3.63,	-3.29,	-3.19,	-2.41,	-2.37,	WRLM1140
. -2.12,	-2.32,	-2.02,	-1.54,	-1.54,	-2.02,	-2.12,	WRLM1141
. -2.02,	-1.24,	-0.27,	-0.71,	-1.54,	-2.41,	-3.05,	WRLM1142
. -3.15,	-1.73,	-2.07,	-2.56,	-3.00,	-3.05,	-3.58,	WRLM1143
. -3.78,	-4.17,	-4.71,	-5.19,	-5.58,	-5.63,	-6.32/	WRLM1144
DATA A28/	148.00,	148.30,	148.67,	149.04,	149.63,	150.67,	WRLM1145
. 151.26,	151.78,	151.04,	120.47,	119.43,	119.43,	119.28,	WRLM1146
. 119.65,	119.21,	118.91,	119.51,	119.51,	119.65,	119.80,	WRLM1147
. 119.43,	119.95,	120.91,	121.65,	122.47,	123.28,	123.58,	WRLM1148
. 124.47,	124.69,	124.69,	125.43,	125.65,	125.58,	124.84,	WRLM1149
. 124.24,	122.84,	122.02,	121.28,	122.10,	122.99,	123.50,	WRLM1150
. 123.80,	124.76,	124.69,	123.87,	123.58,	123.28,	122.39,	WRLM1151
. 122.69,	122.62,	122.91,	123.50,	123.06,	123.73,	123.43,	WRLM1152
. 122.69,	122.47,	121.95,	121.83,	121.65,	121.88,	121.28,	WRLM1153
. 120.84,	121.28,	120.99,	120.99,	120.99,	121.06,	120.84,	WRLM1154
. 120.91,	120.62,	114.92,	113.83,	112.99,	112.70,	111.88,	WRLM1155
. 111.37,	110.55,	110.48,	110.55,	110.40,	109.14,	109.22,	WRLM1156
. 109.37,	109.07,	110.33,	110.70,	111.22,	112.48,	112.48,	WRLM1157
. 113.36,	113.51,	113.59,	114.92,	115.36,	115.51,	115.36,	WRLM1158
. 116.18,	116.77,	117.43,	118.40,	118.32,	118.77,	118.25,	WRLM1159
. 118.32,	118.17,	117.73,	118.17,	118.17,	118.69,	119.06,	WRLM1160
. 118.69,	118.17,	118.03,	118.03,	116.99,	117.14,	116.47,	WRLM1161
. 116.25,	116.25,	115.95,	114.92,	95.60,	95.60,	95.38,	WRLM1162
. 95.90,	97.38,	97.45,	97.67,	98.26,	99.63,	98.56,	WRLM1163
. 98.56,	98.86,	100.34,	100.19,	100.34,	100.34,	101.82/	WRLM1164
DATA B28/	-6.95,	-7.44,	-7.97,	-8.61,	-9.24,	-9.14,	WRLM1165
. -9.10,	-10.02,	-10.22,	-5.49,	-5.14,	-4.51,	-4.12,	WRLM1166
. -3.39,	-2.76,	-2.22,	-1.53,	-1.19,	-0.37,	0.22,	WRLM1167
. 0.51,	0.90,	1.10,	1.34,	1.19,	0.85,	0.76,	WRLM1168
. 0.71,	1.58,	1.73,	1.73,	1.15,	0.76,	0.32,	WRLM1169
. 0.32,	0.32,	0.37,	-0.22,	-0.32,	-0.17,	-0.56,	WRLM1170
. -0.41,	-1.24,	-1.19,	-1.10,	-1.10,	-1.24,	-1.34,	WRLM1171
. -1.98,	-2.32,	-3.19,	-3.44,	-4.12,	-4.51,	-5.10,	WRLM1172
. -5.00,	-4.36,	-4.12,	-3.68,	-3.34,	-3.10,	-2.32,	WRLM1173
. -2.61,	-3.10,	-3.05,	-3.49,	-4.17,	-4.22,	-4.56,	WRLM1174
. -5.00,	-5.63,	-2.97,	-3.49,	-3.58,	-3.10,	-3.19,	WRLM1175

```
     .   -2.76,   -2.41,   -1.88,   -1.76,   -1.15,   -0.66,   -0.12,    WRLM1176
     .    0.61,    1.39,    1.93,    1.51,    2.37,    2.37,    3.05,    WRLM1177
     .    3.24,    3.63,    4.36,    4.46,    4.46,    5.19,    5.63,    WRLM1178
     .    5.93,    6.41,    5.63,    5.44,    5.14,    4.61,    4.27,    WRLM1179
     .    3.73,    3.39,    2.76,    2.46,    2.07,    1.83,    1.37,    WRLM1180
     .    1.29,    0.71,    0.61,    0.32,   -0.41,   -1.15,   -1.73,    WRLM1181
     .   -2.27,   -2.83,   -3.19,   -3.83,    5.24,    5.00,    4.61,    WRLM1182
     .    4.41,    3.49,    3.10,    2.22,    1.78,    1.54,    1.24,    WRLM1183
     .    0.66,    0.27,   -0.22,   -0.66,   -1.63,   -2.32,   -3.10/    WRLM1184
      DATA A29/  102.56,  103.59,  105.07,  105.44,  106.33,  106.55,    WRLM1185
     .  106.04,  106.41,  106.14,  106.04,  105.37,  104.11,  104.11,    WRLM1186
     .  104.04,  103.45,  103.30,  102.26,  101.60,  100.63,   99.89,    WRLM1187
     .   99.67,   98.86,   98.04,   97.97,   97.01,   95.90,  123.65,    WRLM1188
     .  123.65,  123.65,  123.21,  122.69,  122.76,  121.95,  121.28,    WRLM1189
     .  121.36,  122.47,  123.13,  123.36,  123.50,  123.73,  123.65,    WRLM1190
     .  123.43,  123.87,  124.54,  124.10,  123.21,  123.06,  122.99,    WRLM1191
     .  124.39,  124.54,  125.35,  124.91,  124.76,  125.29,  125.43,    WRLM1192
     .  125.13,  124.91,  124.84,  125.13,  125.95,  125.06,  124.54,    WRLM1193
     .  124.47,  123.73,  123.65,  123.28,  122.91,  123.21,  122.76,    WRLM1194
     .  122.17,  130.98,  131.79,  132.31,  133.20,  134.75,  135.64,    WRLM1195
     .  135.64,  135.72,  136.90,  138.16,  138.16,  138.16,  139.56,    WRLM1196
     .  139.79,  139.03,  139.56,  139.64,  140.30,  140.82,  141.12,    WRLM1197
     .  141.86,  141.49,  141.34,  140.60,  140.75,  140.53,  140.82,    WRLM1198
     .  140.60,  139.49,  138.97,  137.57,  137.12,  136.97,  136.46,    WRLM1199
     .  135.27,  134.16,  133.20,  132.61,  131.13,  146.23,  146.15,    WRLM1200
     .  145.49,  145.41,  144.38,  144.00,  143.19,  141.93,  140.38,    WRLM1201
     .  139.56,  140.08,  140.38,  141.41,  141.78,  141.71,  142.38,    WRLM1202
     .  142.75,  143.26,  143.78,  144.60,  145.49,  146.45,  143.86,    WRLM1203
     .  142.97,  142.23,  141.71,  142.30,  142.08,  141.86,  142.08/    WRLM1204
      DATA B29/   -3.83,   -4.80,   -5.63,   -5.24,   -5.14,   -5.05,    WRLM1205
     .   -4.12,   -3.34,   -3.34,   -2.66,   -2.12,   -1.59,   -0.85,    WRLM1206
     .   -0.46,   -0.12,    0.37,    0.80,    1.24,    1.88,    2.17,    WRLM1207
     .    2.71,    3.39,    3.68,    4.55,    4.66,    5.34,    9.50,    WRLM1208
     .   10.12,   10.51,   10.56,   11.19,   11.44,   12.12,   12.56,    WRLM1209
     .   12.90,   12.51,   12.22,   11.58,   11.05,   11.34,   11.87,    WRLM1210
     .   12.36,   12.61,   12.41,   12.80,   12.95,   13.24,   13.78,    WRLM1211
     .   13.24,   13.39,   12.95,   12.56,   12.02,   12.46,   11.44,    WRLM1212
     .   11.58,   11.44,   10.51,   10.46,   10.36,   10.46,   10.12,    WRLM1213
     .    9.83,    9.68,    9.19,    9.05,    9.63,   10.41,   10.51,    WRLM1214
     .   11.44,   34.20,   34.65,   34.89,   35.53,   35.43,   35.23,    WRLM1215
     .   36.21,   37.33,   36.50,   36.99,   37.48,   37.82,   38.31,    WRLM1216
     .   38.84,   39.38,   40.26,   40.63,   40.70,   41.04,   40.45,    WRLM1217
     .   39.62,   38.65,   37.87,   37.62,   36.94,   36.36,   35.28,    WRLM1218
     .   34.80,   34.89,   34.70,   34.89,   34.55,   34.06,   33.43,    WRLM1219
     .   34.36,   34.06,   34.21,   33.97,   33.97,   44.06,   43.38,    WRLM1220
     .   42.89,   42.21,   42.55,   42.06,   42.01,   42.26,   41.38,    WRLM1221
     .   41.72,   42.26,   43.23,   43.57,   44.06,   44.79,   44.65,    WRLM1222
     .   44.16,   44.16,   43.62,   43.67,   43.67,   44.01,   45.72,    WRLM1223
     .   46.16,   45.67,   46.45,   47.13,   47.62,   48.69,   48.99/    WRLM1224
      DATA A30/  142.08,  142.23,  141.93,  142.15,  141.86,  141.78,    WRLM1225
     .  141.64,  142.38,  142.52,  142.45,  142.89,  143.04,  143.34,    WRLM1226
     .  143.34,  143.26,  143.93,  144.00,  144.38,  144.75,  143.49,    WRLM1227
     .  143.34,  143.04,  143.49,  144.00,  144.00,  125.72,  125.06,    WRLM1228
     .  124.37,  123.58,  123.65,  123.80,  124.32,  124.32,  123.36,    WRLM1229
     .  122.99,  123.29,  123.21,  122.91,  122.76,  123.35,  123.58,    WRLM1230
     .  123.58,  124.10,  124.47,  124.61,  125.13,  125.13,  125.50,    WRLM1231
```

```
     . 125.50,  126.17,  126.91,  127.20,  126.39,  126.76,  126.32,   WRLM1232
     . 125.95,  125.95,  125.95,  125.72,  122.25,  121.21,  121.43,   WRLM1233
     . 120.64,  120.62,  120.02,  120.25,  119.95,  119.65,  120.47,   WRLM1234
     . 120.64,  120.69,  120.84,  120.25,  120.47,  120.91,  121.21,   WRLM1235
     . 122.32,  122.17,  122.76,  122.91,  122.25,  122.25,  122.32,   WRLM1236
     . 121.88,  121.73,  122.47,  122.32,   80.21,   79.84,   79.61,   WRLM1237
     .  80.13,   80.35,   79.47,   79.91,   80.43,   80.87,   81.61,   WRLM1238
     .  81.70,   82.20,   82.13,   81.54,   80.95,   80.55,   80.58,   WRLM1239
     . 127.20,  125.95,  124.76,  124.10,  124.10,  124.32,  125.13,   WRLM1240
     . 125.28,  125.72,  126.39,  127.20,  127.20,  120.99,  121.06,   WRLM1241
     . 120.10,  119.36,  120.32,  121.05,  122.28,  122.32,  121.13,   WRLM1242
     . 121.06,  121.50,  122.47,  123.21,  119.36,  118.77,  118.03,   WRLM1243
     . 116.92,  116.55,  117.88,  118.32,  119.21,  119.36,  105.89/   WRLM1244
      DATA B30/  49.62,   50.01,   50.55,   51.13,   51.82,   52.11,   WRLM1245
     .  52.60,   53.18,   53.47,   53.85,   53.47,   52.79,   52.35,   WRLM1246
     .  51.86,   51.28,   50.55,   50.25,   49.43,   48.55,   49.08,   WRLM1247
     .  48.21,   47.62,   47.33,   46.65,   45.67,    5.53,    6.32,   WRLM1248
     .   6.22,    6.66,    7.14,    7.24,    7.53,    8.02,    7.59,   WRLM1249
     .   7.97,    8.61,    9.00,    9.34,   10.07,   10.55,   10.07,   WRLM1250
     .   9.49,    8.66,    8.85,    8.61,    8.61,    9.14,    9.49,   WRLM1251
     .   9.19,    8.61,    8.61,    7.68,    7.73,    6.61,    7.14,   WRLM1252
     .   6.66,    6.36,    6.02,    5.63,   13.48,   13.73,   14.02,   WRLM1253
     .  14.46,   14.12,   14.36,   14.85,   15.39,   15.97,   16.07,   WRLM1254
     .  16.26,   16.36,   17.04,   17.39,   17.82,   18.41,   18.95,   WRLM1255
     .  18.75,   18.36,   17.82,   17.43,   17.04,   16.46,   16.02,   WRLM1256
     .  15.39,   15.14,   14.46,   13.83,    9.34,    8.95,    8.36,   WRLM1257
     .   7.97,    7.68,    7.53,    6.95,    6.95,    6.27,    6.17,   WRLM1258
     .   6.71,    6.95,    7.34,    7.68,    8.41,    8.90,    9.34,   WRLM1259
     .  -8.27,   -8.51,   -8.56,   -9.00,   -9.83,  -10.07,  -10.41,   WRLM1260
     .  -9.83,   -9.39,   -9.19,   -8.95,   -8.22,  -10.31,   -9.92,   WRLM1261
     .  -9.00,   -9.39,   -9.83,  -10.31,   -8.02,   -7.88,   -7.68,   WRLM1262
     .  -8.12,   -8.56,   -8.46,   -8.02,   -7.73,   -8.17,   -7.92,   WRLM1263
     .  -7.88,   -8.41,   -8.31,   -8.66,   -8.51,   -7.97,   -6.46/   WRLM1264
      DATA A31/ 106.85,  108.55,  109.83,  110.77,  111.51,  112.99,   WRLM1265
     . 114.33,  113.59,  114.70,  115.36,  114.47,  113.89,  112.85,   WRLM1266
     . 111.37,  110.25,  109.59,  108.92,  107.66,  107.15,  106.04,   WRLM1267
     . 129.13,  129.79,  130.16,  131.27,  131.42,  131.35,  130.68,   WRLM1268
     . 129.57,  129.05,  128.76,  129.42,  128.02,  127.35,  126.98,   WRLM1269
     . 127.35,  128.24,  128.24,  128.02,  128.46,  128.31,  128.68,   WRLM1270
     . 129.28,  129.23,  129.79,  129.13,  128.46,  128.17,  127.50,   WRLM1271
     . 127.28,  127.43,  127.43,  127.80,  117.35,  118.17,  118.62,   WRLM1272
     . 119.36,  119.43,  119.51,  119.21,  118.77,  118.40,  118.62,   WRLM1273
     . 118.32,  117.95,  117.58,  117.43,  130.46,  130.68,  130.24,   WRLM1274
     . 129.79,  129.05,  129.28,  130.02,  131.72,  131.42,  131.79,   WRLM1275
     . 131.64,  131.42,  130.83,  121.13,  120.54,  119.80,  120.25,   WRLM1276
     . 121.28,  122.17,  121.73,  121.65,  121.21,  121.28,  147.04,   WRLM1277
     . 146.30,  146.52,  147.63,  148.37,  148.67,  148.67,  148.00,   WRLM1278
     . 147.34,  146.60,  155.25,  154.74,  154.07,  154.81,  155.11,   WRLM1279
     . 156.07,  155.70,  149.41,  149.93,  150.74,  151.26,  151.55,   WRLM1280
     . 152.22,  152.15,  152.29,  150.89,  150.22,  149.41,  150.96,   WRLM1281
     . 152.29,  153.48,  153.70,  153.26,  153.18,  152.95,  152.47,   WRLM1282
     . 152.62,  150.96,  162.14,  162.43,  162.43,  161.77,  161.19,   WRLM1283
     . 161.47,  161.99,  159.99,  160.95,  161.40,  160.88,  160.44/   WRLM1284
      DATA B31/  -5.78,   -5.97,   -6.46,   -6.41,   -7.05,   -6.61,   WRLM1285
     .  -7.00,   -7.22,   -8.07,   -8.07,   -8.36,   -6.17,   -8.07,   WRLM1286
     .  -6.07,   -7.86,   -8.02,   -7.49,   -7.39,   -6.71,   -6.46,   WRLM1287
```

```
     .   -3.54,   -3.39,   -3.65,   -3.97,   -3.54,   -3.00,   -3.15,   WRLM1288
     .   -3.55,   -3.10,   -3.34,   -3.77,   -3.10,   -3.10,   -3.39,   WRLM1289
     .   -3.55,   -3.60,   -3.16,    2.32,    1.66,    1.78,    1.29,   WRLM1290
     .    1.34,    0.76,    9.27,    0.37,    0.65,    0.22,    0.32,   WRLM1291
     .    1.19,    1.19,    1.78,    2.27,    8.31,    8.55,    8.70,   WRLM1292
     .    9.34,   10.17,   10.66,   11.09,   10.61,   10.27,    9.88,   WRLM1293
     .    9.19,    8.95,    8.90,    8.41,   31.04,   31.67,   32.00,   WRLM1294
     .   32.40,   32.84,   33.48,   33.63,   32.94,   32.36,   31.72,   WRLM1295
     .   30.99,   30.94,   30.94,   22.15,   22.90,   23.48,   24.21,   WRLM1296
     .   24.80,   24.75,   24.07,   23.34,   22.90,   22.26,   44.11,   WRLM1297
     .   44.50,   45.57,   45.67,   45.67,   45.28,   44.54,   44.65,   WRLM1298
     .   44.26,   43.96,   49.96,   49.82,   50.16,   50.72,   50.84,   WRLM1299
     .   50.64,   50.11,   -5.58,   -5.34,   -5.19,   -4.32,   -4.12,   WRLM1300
     .   -3.97,   -4.75,   -5.14,   -5.73,   -5.93,   -5.63,   -2.56,   WRLM1301
     .   -2.56,   -3.15,   -3.29,   -3.58,   -4.02,   -3.63,   -3.19,   WRLM1302
     .   -3.10,   -2.66,   -9.49,   -9.10,   -9.05,   -8.07,   -8.36,   WRLM1303
     .   -9.00,   -9.39,   -9.19,   -9.14,   -9.63,   -9.92,   -9.49/   WRLM1304
      DATA A32/  160.44,  158.07,  158.83,  159.70,  159.92,  159.40,   WRLM1305
     .  159.18,  165.62,  164.80,  165.02,  165.76,  165.76,  166.73,   WRLM1306
     .  167.69,  167.47,  166.73,  171.54,  172.13,  173.17,  172.80,   WRLM1307
     .  171.69,  176.64,  177.24,  177.68,  177.16,  177.46,  177.98,   WRLM1308
     .  178.13,  178.13,  177.75,  183.97,  183.82,  184.34,  185.01,   WRLM1309
     .  185.75,  186.64,  185.82,  185.60,  184.93,  184.55,  180.79,   WRLM1310
     .  181.23,  182.05,  182.34,  181.75,  180.72,  180.72,  180.57,   WRLM1311
     .  185.45,  185.75,  186.69,  186.73,  185.97,  185.30,  204.03,   WRLM1312
     .  204.70,  204.77,  205.29,  204.40,  203.96,  203.73,   47.94,   WRLM1313
     .   47.34,   47.49,   46.53,   46.31,   45.64,   44.31,   42.38,   WRLM1314
     .   42.31,   42.61,   42.61,   42.46,   42.61,   41.72,   41.57,   WRLM1315
     .   41.87,   41.79,   42.31,   42.31,   43.13,   44.24,   44.90,   WRLM1316
     .   45.35,   45.35,   46.23,   46.09,   46.68,   47.12,   46.83,   WRLM1317
     .   47.49,   47.49,   48.16,   48.01,   48.01,   48.90,   48.60,   WRLM1318
     .   48.60,   48.90,   48.75,   47.85,  355.09,  354.42,  355.90,   WRLM1319
     .  354.57,  354.35,  355.68,  356.50,  356.64,  356.05,  355.61,   WRLM1320
     .  354.94,  354.42,  353.54,  353.31,  353.54,  353.91,  353.61,   WRLM1321
     .  353.24,  353.83,  354.20,  354.79,  355.83,  356.94,  356.42,   WRLM1322
     .  357.46,  357.98,  357.46,  356.72,  356.87,  358.13,  358.42,   WRLM1323
     .  358.64,  359.09,  359.09,    0.27,    0.05,    1.09,    0.35/   WRLM1324
      DATA B32/   -9.53,   -8.07,   -6.95,   -7.10,   -7.92,   -8.07,   WRLM1325
     .   -7.05,   54.55,   54.94,   55.81,   55.03,   54.40,   54.40,   WRLM1326
     .   54.30,   53.96,   54.35,   52.79,   53.57,   53.08,   52.84,   WRLM1327
     .   52.89,   51.52,   51.91,   51.47,   51.13,   50.84,   51.18,   WRLM1328
     .   50.94,   50.45,   50.60,   52.45,   51.91,   52.06,   51.52,   WRLM1329
     .   51.77,   52.30,   52.11,   52.21,   52.06,   52.06,   52.06,   WRLM1330
     .   51.52,   51.91,   51.13,   51.23,   51.18,   51.57,   51.96,   WRLM1331
     .   60.20,   60.59,   60.20,   59.95,   59.76,   60.11,   20.31,   WRLM1332
     .   20.41,   20.12,   19.58,   19.24,   19.82,   20.02,  -11.34,   WRLM1333
     .  -11.53,  -12.31,  -13.05,  -14.12,  -14.90,  -15.34,  -15.87,   WRLM1334
     .  -16.65,  -17.14,  -18.12,  -19.19,  -20.12,  -20.99,  -22.26,   WRLM1335
     .  -22.65,  -23.58,  -24.36,  -25.14,  -25.33,  -24.80,  -24.85,   WRLM1336
     .  -24.30,  -23.73,  -23.43,  -22.55,  -21.48,  -20.46,  -19.38,   WRLM1337
     .  -18.41,  -17.73,  -16.95,  -16.31,  -15.39,  -15.04,  -14.12,   WRLM1338
     .  -13.53,  -13.00,  -12.17,  -11.40,   49.63,   50.01,   50.34,   WRLM1339
     .   51.23,   51.96,   52.55,   52.89,   53.72,   54.16,   54.72,   WRLM1340
     .   55.42,   55.77,   55.96,   56.50,   56.79,   56.57,   57.08,   WRLM1341
     .   57.76,   57.76,   58.20,   58.01,   58.35,   58.35,   57.57,   WRLM1342
     .   57.33,   56.79,   56.45,   56.01,   55.47,   55.13,   54.45,   WRLM1343
```

```
     .   54.74,   54.55,   53.90,   53.39,   [illegible],   52.11,   51.23/  WRLM1344
      DATA A33/    0.79,  355.61,  355.72,  356.13,  356.54,  355.90,  WRLM1345
     .  355.39,  345.69,  351.21,  351.24,  351.46,  352.57,  353.24,  WRLM1346
     .  353.17,  353.54,  353.24,  354.15,  353.54,  352.[illegible],  351.83,  WRLM1347
     .  351.02,  351.02,  349.47,  349.39,  349.69,  349.24,  349.39,  WRLM1348
     .  349.60,  349.69,  333.40,  333.33,  334.74,  334.51,  333.55,  WRLM1349
     .  343.17,  343.94,  344.14,  343.94,  343.10,  343.03,  341.62,  WRLM1350
     .  341.64,  341.47,  341.10,  341.25,  341.47,  341.55,  341.59,  WRLM1351
     .  341.84,  341.55,  341.32,   11.97,   13.00,   14.63,   14.41,  WRLM1352
     .   14.85,   14.19,   13.52,   13.22,   11.97,   26.32,   25.51,  WRLM1353
     .   24.70,   24.40,   23.14,   22.77,   23.96,   25.29,   25.95,  WRLM1354
     .   30.91,   31.73,   33.06,   33.13,   33.65,   33.21,   33.13,  WRLM1355
     .   32.62,   31.80,   30.91,   31.13,    6.78,    7.15,    7.01,  WRLM1356
     .    7.67,    8.26,    8.26,    8.63,    8.69,    8.34,    7.45,  WRLM1357
     .    6.93,    8.41,    8.93,    9.00,    8.71,    8.19,    8.19,  WRLM1358
     .    6.26,    2.12,    1.53,    2.27,    2.71,    3.45,    3.63,  WRLM1359
     .    3.62,    2.71,    2.42,  274.71,  275.97,  277.33,  278.19,  WRLM1360
     .  279.67,  280.63,  251.23,  282.48,  283.15,  283.67,  285.22,  WRLM1361
     .  284.70,  283.15,  282.19,  281.52,  280.85,  279.97,  279.45,  WRLM1362
     .  278.80,  277.89,  277.01,  275.23,  284.63,  285.74,  285.96,  WRLM1363
     .  286.55,  286.18,  287.59,  288.33,  290.03,  290.40,  289.51/  WRLM1364
      DATA B33/   50.74,   50.21,   50.40,   50.01,   50.35,   49.91,  WRLM1365
     .   49.52,   50.50,   50.94,   50.71,   51.13,   51.23,   51.62,  WRLM1366
     .   52.01,   52.94,   53.23,   53.52,   54.16,   54.35,   54.74,  WRLM1367
     .   54.45,   53.86,   53.86,   53.13,   52.79,   52.06,   51.33,  WRLM1368
     .   50.89,   50.69,   37.53,   37.72,   37.53,   37.23,   37.43,  WRLM1369
     .   28.40,   28.41,   27.87,   27.63,   27.77,   28.21,   27.63,  WRLM1370
     .   26.99,   26.85,   27.14,   27.43,   27.58,   29.28,   28.94,  WRLM1371
     .   28.46,   28.60,   29.19,   37.58,   36.94,   36.55,   36.99,  WRLM1372
     .   37.53,   37.95,   37.72,   38.11,   37.72,   35.04,   35.14,  WRLM1373
     .   35.14,   35.58,   35.43,   35.09,   34.55,   34.65,   34.94,  WRLM1374
     .   34.36,   34.50,   34.50,   34.80,   35.33,   35.33,   35.67,  WRLM1375
     .   35.33,   35.23,   34.94,   34.65,   40.65,   39.87,   39.09,  WRLM1376
     .   38.89,   38.89,   39.43,   39.62,   40.40,   40.99,   41.09,  WRLM1377
     .   40.79,   43.09,   42.74,   42.21,   41.67,   41.92,   42.60,  WRLM1378
     .   43.04,   39.72,   39.04,   38.89,   39.28,   39.23,   39.57,  WRLM1379
     .   40.21,   40.11,   39.72,   22.36,   22.95,   23.14,   22.80,  WRLM1380
     .   22.75,   22.16,   21.82,   21.63,   20.95,   20.41,   19.92,  WRLM1381
     .   19.63,   19.87,   20.02,   20.65,   20.70,   20.90,   21.53,  WRLM1382
     .   21.58,   21.97,   22.46,   22.26,   18.51,   18.46,   19.04,  WRLM1383
     .   19.19,   19.78,   19.63,   19.24,   18.99,   18.75,   18.31/  WRLM1384
      DATA A34/  288.33,  286.92,  285.96,  287.96,  287.15,  288.33,  WRLM1385
     .  287.52,  286.92,  286.11,  286.11,  284.78,  284.41,  284.85,  WRLM1386
     .  285.37,  287.37,  288.03,  289.29,  289.66,  289.88,  289.29,  WRLM1387
     .  289.29,  286.11,  297.58,  297.95,  298.77,  299.65,  300.76,  WRLM1388
     .  301.65,  302.17,  301.65,  300.54,  299.58,  298.77,  297.95,  WRLM1389
     .  302.25,  302.91,  303.73,  303.87,  302.91,  302.02,  302.02,  WRLM1390
     .  301.13,  301.95,  299.43,  299.80,  300.99,  301.95,  301.13,  WRLM1391
     .  300.10,  299.65,  299.14,  298.03,  298.32,  295.35,  293.14,  WRLM1392
     .  292.70,  291.29,  290.85,  289.59,  289.00,  289.00,  289.59,  WRLM1393
     .  289.44,  290.25,  290.85,  292.48,  293.22,  291.07,  290.55,  WRLM1394
     .  290.10,  290.55,  290.65,  286.13,  287.01,  286.63,  285.67,  WRLM1395
     .  286.04,  267.46,  267.09,  267.38,  267.66,  267.16,  267.53,  WRLM1396
     .  267.90,  268.64,  268.27,  268.72,  269.46,  269.53,  270.12,  WRLM1397
     .  270.12,  269.01,  267.46,  303.59,  302.69,  301.65,  301.21,  WRLM1398
     .  300.17,  300.62,  299.73,  300.32,  300.99,  302.32,  303.73,  WRLM1399
```

```
     .  302.99,  304.16,  304.86,  306.34,  306.61,  306.54,  306.99,   WRLM1400
     .  305.65,  305.26,  305.65,  304.93,  303.87,  303.50,  302.47,   WRLM1401
     .  303.13,  303.57,  303.05,  322.10,  321.41,  322.01,  323.12,   WRLM1402
     .  322.89,  322.08,  321.55,  320.15,  314.45,  314.83,  315.79,   WRLM1403
     .  315.20,  315.35,  314.46,  312.31,  313.42,  313.64,  314.01/   WRLM1404
      DATA B34/  18.51,   18.12,   18.36,  -67.76,  -68.59,  -69.65,   WRLM1405
     .  -69.76,  -70.23,  -70.15,  -70.45,  -70.30,  -70.59,  -70.83,   WRLM1406
     .  -71.13,  -71.37,  -70.83,  -70.79,  -70.10,  -69.32,  -68.74,   WRLM1407
     .  -68.05,  -67.76,  -61.47,  -60.89,  -60.59,  -61.03,  -60.54,   WRLM1408
     .  -60.40,  -60.64,  -61.32,  -61.23,  -61.67,  -61.13,  -61.52,   WRLM1409
     .  -63.62,  -62.76,  -62.40,  -62.06,  -61.71,  -61.91,  -62.35,   WRLM1410
     .  -62.93,  -63.52,  -51.28,  -51.77,  -51.43,  -50.84,  -50.60,   WRLM1411
     .  -50.35,  -50.60,  -50.16,  -50.53,  -51.28,  -51.18,  -53.72,   WRLM1412
     .  -53.13,  -52.74,  -52.21,  -51.62,  -51.96,  -53.03,  -53.42,   WRLM1413
     .  -53.62,  -54.01,  -53.67,  -53.91,  -53.72,  -54.50,  -54.11,   WRLM1414
     .  -54.50,  -54.89,  -54.55,  -52.40,  -52.69,  -53.13,  -53.93,   WRLM1415
     .  -52.40,    0.37,   -0.17,   -0.51,   -1.24,   -1.54,   -1.58,   WRLM1416
     .   -1.73,   -1.63,   -1.34,   -1.10,   -1.10,   -0.76,   -0.32,   WRLM1417
     .   -0.07,   -0.22,    0.12,   51.08,   50.55,   50.21,   49.62,   WRLM1418
     .   49.62,   48.94,   48.55,   48.21,   47.67,   47.91,   47.52,   WRLM1419
     .   46.94,   47.18,   47.47,   46.69,   47.18,   47.77,   48.40,   WRLM1420
     .   48.30,   48.45,   48.79,   49.28,   48.94,   49.47,   49.47,   WRLM1421
     .   50.25,   50.50,   51.03,  -53.38,  -53.47,  -53.72,  -53.77,   WRLM1422
     .  -54.20,  -54.35,  -53.81,  -53.52,  -59.62,  -59.28,  -59.72,   WRLM1423
     .  -59.86,  -60.30,  -59.76,  -59.67,  -59.03,  -59.47,  -59.86/   WRLM1424
      DATA A35/ 313.72,  312.61,  303.23,  304.10,  304.84,  305.35,   WRLM1425
     .  305.80,  305.43,  303.80,  284.63,  284.63,  284.56,  283.15,   WRLM1426
     .  283.74,  284.63,  286.85,  286.13,  285.81,  286.41,  286.85,   WRLM1427
     .  230.67,  231.56,  232.75,  233.85,  234.89,  236.00,  236.00,   WRLM1428
     .  235.04,  234.45,  232.89,  231.04,  228.38,  227.79,  228.16,   WRLM1429
     .  227.64,  226.68,  226.90,  228.01,  192.93,  191.82,  192.35,   WRLM1430
     .  192.93,  191.00,  191.89,  193.08,  192.78,  191.97,  191.37,   WRLM1431
     .  206.62,  205.58,  204.47,  204.40,  205.36,  206.32,  172.06,   WRLM1432
     .  172.65,  173.54,  173.54,  174.50,  174.50,  173.91,  173.98,   WRLM1433
     .  173.68,  172.83,  173.61,  174.79,  174.50,  175.09,  176.35,   WRLM1434
     .  176.72,  177.01,  178.13,  178.57,  178.20,  178.35,  177.98,   WRLM1435
     .  177.38,  175.83,  174.94,  174.87,  174.20,  173.83,  172.57,   WRLM1436
     .  172.26,  172.13,  171.32,  171.17,  169.61,  168.21,  167.10,   WRLM1437
     .  167.02,  166.36,  166.43,  167.84,  168.06,  168.73,  170.21,   WRLM1438
     .  170.35,  170.72,  170.80,  172.20,  172.43,  173.76,  173.68,   WRLM1439
     .  173.98,  172.65,  145.12,  144.82,  145.78,  145.56,  146.97,   WRLM1440
     .  147.34,  148.45,  148.45,  149.04,  148.52,  149.04,  148.37,   WRLM1441
     .  147.19,  145.41,  176.94,  177.53,  178.79,  179.16,  178.27,   WRLM1442
     .  177.24,  179.04,  179.31,  180.27,  180.79,  181.16,  180.94,   WRLM1443
     .  179.83,  179.16,  168.43,  167.62,  167.32,  166.80,  165.47/   WRLM1444
      DATA B35/ -60.25,  -59.57,  -59.85,  -59.52,  -59.67,  -59.72,   WRLM1445
     .  -60.15,  -60.35,  -60.06,  -51.67,  -51.33,  -50.69,  -50.89,   WRLM1446
     .  -51.43,  -51.72,  -52.89,  -52.25,  -52.59,  -53.38,  -52.84,   WRLM1447
     .   50.55,   49.57,   49.43,   48.60,   48.40,   48.30,   48.00,   WRLM1448
     .   49.23,   49.96,   50.35,   50.45,   52.21,   52.79,   53.77,   WRLM1449
     .   54.45,   54.01,   53.09,   52.45,   54.30,   53.57,   53.52,   WRLM1450
     .   54.25,   60.01,   60.54,   60.65,   59.47,   59.57,   59.96,   WRLM1451
     .   56.94,   57.33,   57.13,   56.79,   56.45,   56.79,  -34.11,   WRLM1452
     .  -34.65,  -35.23,  -35.62,  -36.21,  -37.04,  -37.04,  -37.72,   WRLM1453
     .  -37.72,  -38.01,  -38.60,  -39.14,  -39.77,  -40.26,  -40.11,   WRLM1454
     .  -39.43,  -38.50,  -38.36,  -37.82,  -37.48,  -37.19,  -36.50,   WRLM1455
```

```
     . -36.79,  -36.40,  -36.26,  -35.48,  -34.59,  -34.99,  -34.36,   WRLM1456
     . -39.77,  -40.50,  -40.87,  -42.01,  -42.44,  -42.53,  -43.37,   WRLM1457
     . -44.26,  -44.64,  -45.33,  -45.33,  -46.26,  -45.62,  -45.72,   WRLM1458
     . -45.13,  -44.11,  -43.62,  -42.99,  -42.26,  -41.43,  -40.34,   WRLM1459
     . -40.40,  -40.36,  -40.11,  -40.29,  -41.33,  -42.35,  -42.79,   WRLM1460
     . -42.11,  -42.35,  -42.01,  -42.01,  -41.33,  -40.69,  -39.87,   WRLM1461
     . -40.35,  -40.16,  -16.90,  -17.23,  -17.58,  -17.04,  -16.80,   WRLM1462
     . -16.75,  -15.97,  -15.53,  -15.24,  -15.09,  -15.39,  -15.73,   WRLM1463
     . -16.07,  -15.87,  -21.82,  -21.53,  -20.85,  -20.46,  -19.32/   WRLM1464
      DATA A36/ 165.69,  166.28,  167.54,  166.88,  167.10,  166.95,   WRLM1465
     .  166.95,  166.50,  166.36,  166.83,  170.80,  170.21,  170.05,  WRLM1466
     .  168.55,  169.61,  170.13,  177.99,  178.13,  178.72,  178.37,  WRLM1467
     .  178.57,  177.98,  178.13,  179.93,  180.20,  178.87,  179.53,  WRLM1468
     .  184.42,  184.34,  184.64,  183.68,  183.53,  184.05,  184.34,  WRLM1469
     .  165.69,  166.13,  166.95,  167.25,  168.36,  168.58,  168.28,  WRLM1470
     .  166.95,  166.13/                                               WRLM1471
      DATA B36/ -20.60,  -20.99,  -21.43,  -50.94,  -50.60,  -50.21,   WRLM1472
     . -49.82,  -49.52,  -49.96,  -50.60,  -52.16,  -51.82,  -51.52,   WRLM1473
     . -51.85,  -52.16,  -52.01,  -49.04,  -48.69,  -48.55,  -48.25,   WRLM1474
     . -47.91,  -48.26,  -48.50,  -47.03,  -46.40,  -46.16,  -46.74,   WRLM1475
     . -44.01,  -43.38,  -43.04,  -42.89,  -43.38,  -43.52,  -43.91,   WRLM1476
     . -76.93,  -76.30,  -76.25,  -75.71,  -76.15,  -76.49,  -76.93,   WRLM1477
     . -76.69,  -76.88/                                                WRLM1478
C THE DATA IN THE "C" ARRAY ARE THE NUMBER OF VECTOR SEGMENTS IN EACH  WRLM1479
C DISTINCT BODY STORED IN ORDER IN THE "A" AND "B" ARRAYS              WRLM1480
      DATA C /   837, 746, 576, 494, 130, 279,  46,  35, 122,  83,  55, WRLM1481
     .  49,  20,  25,  20,  13,  13,  18,  16,  11,   7,  11,   7,   7, WRLM1482
     .  12,  80,  62,  51,  43,  44,  39,  22,  33,  34,  28,  17,  12, WRLM1483
     .   6,   7,   9,  21,  11,   6,  15,  14,  13,  10,  10,   7,  11, WRLM1484
     .  10,   7,   6,   6,   5,   4,   5,   4,   5,  10,   8,   6,   7, WRLM1485
     .  40,  45,  22,   5,   6,   6,   5,   9,   9,  11,  11,   7,   9, WRLM1486
     .  22,  13,  19,  12,   9,  11,  13,   5,   5,  16,  28,   8,   6, WRLM1487
     .   6,   7,   6,   5,  11,   7,   4,   6,   6,  29,  23,  14,   6, WRLM1488
     .   8,   8,   7,   6,   7,   4,   7,   9/                         WRLM1489
      CALL EARTH(N,C,A1,B1)                                            WRLM1490
      RETURN                                                           WRLM1491
      END                                                              WRLM1492
```

1.1.4 WOLF SC4020 PLOT PACKAGE

INTRODUCTION

The WOLF Plot Package is a complete system for producing SC4020 and/or printer plots. The package has been designed to be highly flexible and easy to use. Any plot from a quick simple plot (which requires only one call to the package) to highly sophisticated plots (including motion picture plots) can be easily generated with only a basic knowledge of FORTRAN being necessary.

The SC4020 (Stromberg Carlson 4020) is a cathode ray plotter whose outstanding feature is its plotting speed. As such, any user who is producing series of plots should use this plotter. Film (35 mm and 16 mm) and hardcopy are available and the WOLF Plot Package also allows for printer plots which can be used as a quick look for the SC4020 output.

A typewriter mode is available which conveniently allows plotting of character information on the SC4020. This is especially useful as a printer substitute for large amounts of output.

The routines in the Plot Package are all in G and H level FORTRAN with the exception of TIMING which is in IBM 360 Assembly Language. These routines were designed to be efficient on the IBM 360 series machines; no attempt whatever has been made to pursue the myth of compatibility.

SYSTEM REQUIREMENTS

The system requirements for this package are:

- An IBM 360 which supports G or H level FORTRAN. The 360 Assembler must also be available.

- This IBM 360 must use O.S. (360 Operating System)

- The Plot Package requires 45K bytes of core storage.

- An IBM 2400 series 7 track tape drive must be available for the SC4020 Plotter Driver Tape.

In addition, the WOLF Plot Package requires the FORTRAN library routines ALOG10, SIN, and COS.

PROGRAM DESCRIPTION

The WOLF Plot Package is a system of FORTRAN callable subroutines which are used to create plots. It is structured into four major levels as follows:

1. Basic Level - The basic level routines perform the primary functions of the plot package. Except for a few auxiliary routines, the basic level routines are necessary for all other routines. However, few of the basic routines are user called.

The primary basic routine assembles the instructions for the SC4020 tape. There is a printer simulation (of the SC4020) in this routine. This allows for SC4020 plots, printer plot or both simultaneously. The other major basic level routine is used for initialization and termination of the Plot Package.

2. Intermediate Level - The intermediate level contains the major user called routine. Some of the functions of this level are

a. Grid Overlays (both Cartesian and Polar) with labels

b. Scaling functions

c. Plotting of vectors or characters in any of the following coordinate systems:

Linear
Semi-Log
Log-Log
Polar

3. High Level - This level is for quick plots with a minimum of programming effort. At this level, all of the other levels are called upon. Only one FORTRAN statement is necessary to produce a plot of any array of data complete with a labeled grid overlay.

4. Independent Level - These routines perform functions that are independent of all other levels except the basic level. The following are among the functions of this level:

a. Labels: A string of characters can be plotted horizontally, vertically or diagonally (at any inclination and direction).

b. Graphic Letters: Letters can be output in any size and in any font design (i.e., standard block letters, mathematical symbols or even old English script).

c. Typewriter Mode: The typewriter function in the SC4020 plotter can be used by calling the various typewriter routines. These allow for information to be typed (strings of characters output in page format) on either the SC4020 or printer.

In addition to these four levels, there are also a number of auxiliary routines. These perform such functions as conversion of decimal (binary) numbers to EBCDIC equivalents and dump of the SC4020 plot tape.

The functional structure of the Plot Package is illustrated in Figure 1.

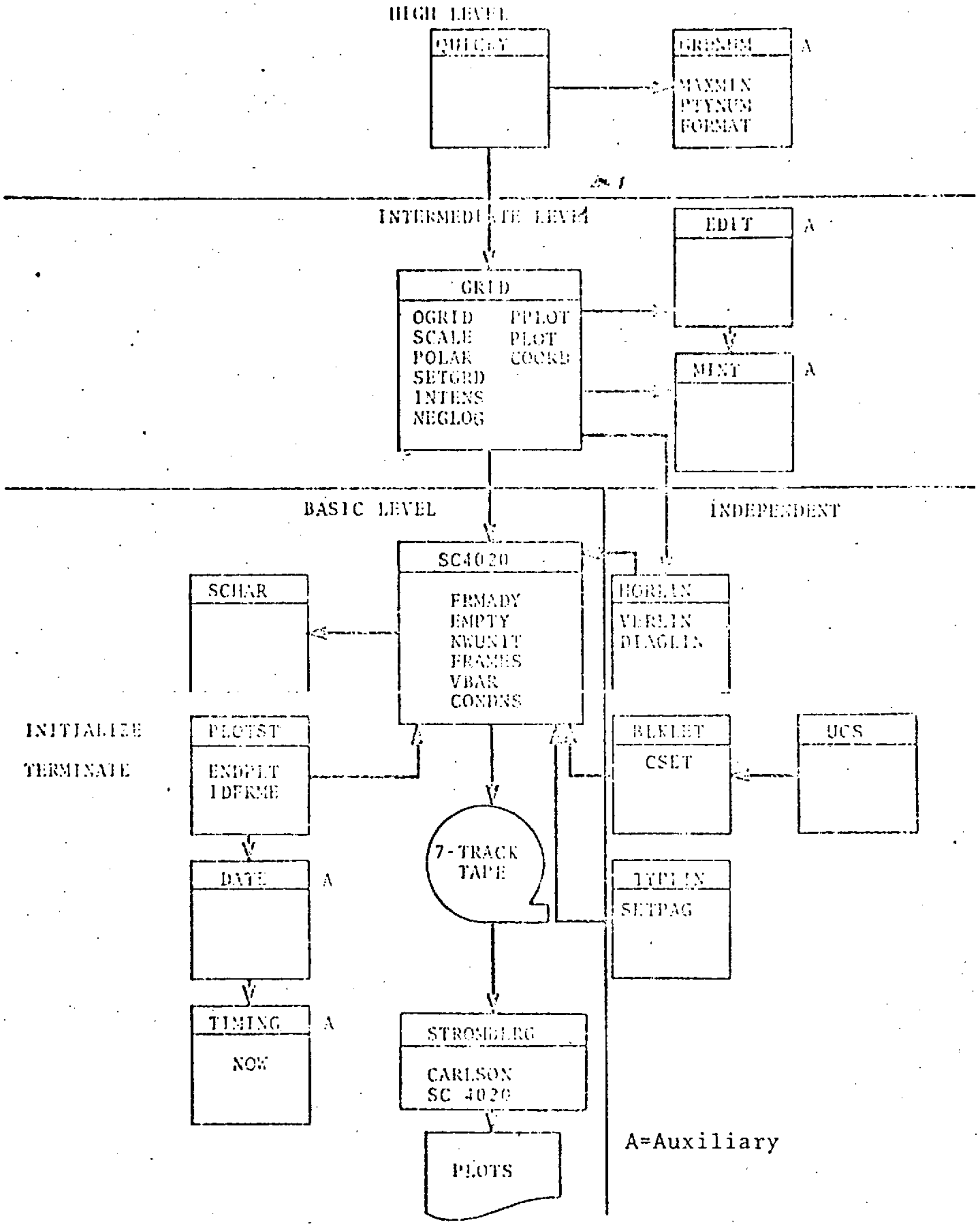
FLOWCHART OF SUBROUTINE STRUCTURE

The flowchart of the subroutine structure is presented below. The entry points associated with each subroutine are presented with their respective control section names ("subroutine names").

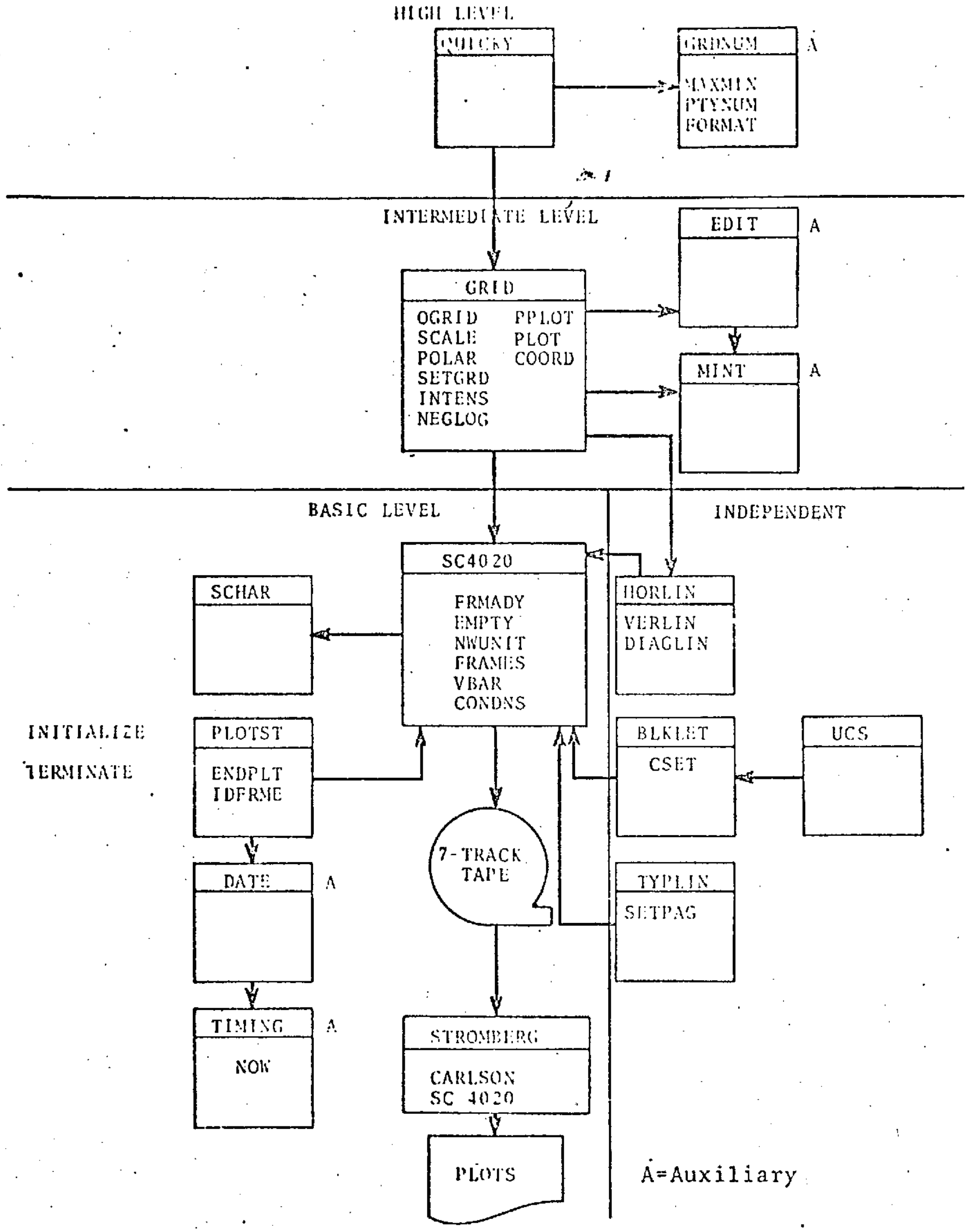
It should be noted that the flow chart is divided according to the four major levels of the Plot Package:

- Basic Level - The basic level routines perform the primary functions of the Plot Package.
- Intermediate Level - The intermediate level contains the major user-called routines.
- High Level - This level is for quick plots with a minimum of programming effort. At this level, all of the other levels are invoked.
- Independent Level - These routines perform functions which are independent of all levels other than the basic level.

WOLF PLOT PACKAGE STRUCTURE



WOLF PLOT PACKAGE STRUCTURE



SUMMARY OF SUBROUTINE ENTRIES
IN THE WOLF PLOT PACKAGE

BLKLET	Draws any set of characters on the SC4020 to any size.
CONDNS	For one page printer plots.
COORD	Recovers the raster coordinates of a point.
CSET	Initializes the character font in BLKLET.
DATE	Returns the current date (in alphameric).
DIAGLN	Generate a diagonal label.
EDIT	Converts and edits binary numbers to EBCDIC.
EMPTY	Terminates the ploter tape output.
ENDPLT	Terminates the Plot Package.
FORMAT	Generates a format code for use with EDIT.
FRAMES	Returns frame count.
FRMADV	Advances the frame.
GRDNUM	Computes arguments for GRID or OGRID.

HORLIN	Generate horizontal label.
IDFRME	Generates the identification frame for the Plot Package.
INTENS	Sets the intensity for PLOT or PPLOT.
MAXMIN	Finds maximum and minimum of an array.
MINT	Truncates to the next algebraically smaller number.
NEGLOG	Enables plotting of negative arguments logarithimically.
NOW	Obtains the current date and time from the system.
NWUNIT	Sets the output unit numbers.
OGRID	Computes the necessary scaling for PLOT; plots and labels an open grid.
PLOT	Plots a set of points or a series of contiguous vectors.
PLOTST	Initializes the Plot Package.
POLAR	Computes the necessary scaling for PPLOT or PLOT; draws and labels a polar grid.

PPLOT	Plots a set of points or a series of contiguous vectors in polar coordinates.
PTYNUM	Computes esthetic plotting limits on data.
QUICKY	Plots X-Y values on an appropriate grid.
SCALE	Computes the scaling for PLOT.
SCHAR	Function value is EBDIC character value corresponding to the input SC4020 character value.
SC4020	Translates plot commands into SC4020 instructions and/or printer plots.
SETGRD	Sets the raster grid limits.
SETPAG	Sets the line count and starting column for TYPLIN.
TYPLIN	Type a line of information on the SC4020.
UCS	Calls CSET with a standard character font.
VBAR	Set use of vertical bar "|" instead of "I" for vertical lines on printer plots.
VERLIN	Generate vertical label.

SUBROUTINE CROSS REFERENCE CHART

The cross reference chart for the WOLF Plot Package is given below. The called routines are listed across the top; the calling routines are listed down the left side. It should be noted that this chart is by subroutine and function entry rather than by control section name. The appropriate control sections have been designated in the flowchart of Subroutine Structure given previously.

CALLED ROUTINES

CALLING ROUTINES	CSET	EDIT	EMPTY	GRDNUM	GRID	HORLIN	MINT	PLOT	SC4020	SCHAR
BLKLET									●	
DIAGLN						●				
EDIT							●			
ENDPLT			●							
FRMADV										●
GRID		●				●	●		●	
HORLIN									●	
IDFRME									●	
OGRID		●				●	●		●	
PLOT									●	
PLOTST									●	
POLAR		●				●			●	
PPLOT									●	
QUICKY				●	●	●		●		
SC4020										●
TYPLIN									●	
UCS	●									
VERLIN									●	

COMMON BLOCK CROSS REFERENCE CHART

ROUTINES

COMMON BLOCK	PLOTST	GRID	HORLIN	SC4020
CPLOTS	O	O	O	O

BLKLET

DESCRIPTION

BLKLET is primarily a user routine which is used to produce block letters (letters that are drawn as vectors instead of being printed).

Before BLKLET can be used, a character set (which consists of the character description arrays) must be input through the entry CSET.

BLKLET first determines the position of the first character and then for each character, using the character description arrays, determines the vectors that make up each character and calls SC4020 to plot these characters.

NAME		BLKLET
ENTRY POINT		PURPOSE
BLKLET		TO DRAW ANY SET OF CHARACTERS ON THE SC4020 TO ANY SIZE
CSET		TO INITIALIZE A CHARACTER FONT IN BLKLET
CALLING SEQUENCE		CALL BLKLET(CHAR,N,IXX,IYY,IDLTX,IDLTY,JSIZE)
SYMBOL	TYPE	DESCRIPTION
CHAR	L*1	INPUT - CHARACTER STRING TO BE PLOTTED
N	I	INPUT - NUMBER OF CHARACTERS
IXX	I	INPUT - X RASTER COUNT OF CENTER OF LINE
IYY	I	INPUT - Y RASTER COUNT OF CENTER OF LINE
IDLTX	I	INPUT - X INCREMENT BETWEEN CHARACTERS
IDLTY	I	INPUT - Y INCREMENT BETWEEN CHARACTERS
JSIZE	I	INPUT - THE ABSOLUTE VALUE OF JSIZE IS THE SIZE FACTOR: =1 NORMAL PRINTED SIZE; .GE.1 NORMAL UPRIGHT CHARACTERS; .LT.1 CHARACTERS ROTATED 90 DEGREES COUNTER CLOCKWISE
CALLING SEQUENCE		CALL CSET(NC,ICHAR,IPOS,IVEC)
SYMBOL	TYPE	DESCRIPTION
NC	I	INPUT - NUMBER OF CHARACTER IN SET
ICHAR	L	INPUT - CHARACTERS
IPOS	I	INPUT - ARRAY OF POINTERS TO IVEC
IVEC	I	INPUT - ARRAY OF CHARACTERS DESCRIPTION
SUBROUTINE USED		SC4020
COMMON BLOCKS		NONE
INPUT FILES		NONE
OUTPUT FILES		NONE
RESTRICTIONS		MASTER CHARACTER SET MUST BE INPUT THROUGH ONE OF THE FOLLOWING METHODS:

1. CALL UCS WILL INPUT A STANDARD FONT

2. SPECIAL CHARACTER FONTS MAY BE INPUT
VIA CSET

IPOS AND IVEC CANNOT BE CHANGED AFTER THE CALL TO CSET BECAUSE THEIR LOCATION AND NOT THEIR VALUES ARE SAVED

REFERENCES NONE

```
      SUBROUTINE BLKLET(CHAR,N,IXX,IYY,IDLTX,IDLTY,JSIZE)
      LOGICAL*1 CHAR(1),ICHAR(1),LL1(4)
      EQUIVALENCE(LL,LL1(1))
      INTEGER IX(4),SHIFT(4)/ Z1000,Z0100,Z0010,Z0001 /
      INTEGER*2 IPOS(1),IVEC(1)
C COMPUTE CENTER OF FIRST CHARACTER
      ISIZE=IABS(JSIZE)
      IXB=IXX-((N-1)*IDLTX-6*ISIZE)/2
      IYB=IYY-((N-1)*IDLTY-9*ISIZE)/2
C LOOP ON ALL CHARACTERS
      DO 50 I=1,N
      LL1(4)=CHAR(I)
      IC=LL
C FIND CHARACTER
      DO 10 J=1,NC
      LL1(4)=ICHAR(J)
      IF (IC.EQ.LL) GO TO 20
 10   CONTINUE
      GO TO 45
C DRAW CHARACTER
C NOTE THAT IVEC IS PACKED X,Y,DX-6,DY
 20   IS=IPOS(J)
      IE=IPOS(J+1)-1
      DO 40 K=IS,IE
      DO 30 L=1,4
 30   IX(L)=MOD(IVEC(K)/SHIFT(L),SHIFT(3))*ISIZE
      IX(3)=IX(3)-6*ISIZE
      IF(JSIZE.GT.0) GO TO 35
      DO 32 L=1,3,2
      IT=IX(L)
      IX(L)=IX(L+1)
 32   IX(L+1)=IT
 35   IX(1)=IX(1)+IXB
      IX(2)=IX(2)+IYB
 40   CALL SC4020 (11,IX(1),IX(2),IX(3),IX(4))
 45   IXB=IXB+IDLTX
      IYB=IYB+IDLTY
 50   CONTINUE
C CSET ENTRY
      ENTRY CSET (NC,ICHAR,IPOS,IVEC)
C INITIALIZE CHARACTER FONT
      RETURN
      END
```

DATE

DESCRIPTION

DATE produces an array describing the current date in alphanumerics. This routine is used by PLOTST to produce the ID frames, but can also be used by the user.

DATA calls NOW in order to determine the current date in the integer form YYDDD where YY is the year and DDD is the number of the day in the year. The year is determined by division by 1000 and the day is the modulo 1000. Then using the array IDAYS (which give the relationship of days versus month), the month and day of the month are determined (leap years are taken into consideration).

Finally the year, month and day of the month are put into character coding of the form year/month/day.

NAME DATE

PURPOSE TO RETURN THE CURRENT DATE (IN ALPHANUMERIC)

CALLING SEQUENCE CALL DATE(CHAR)

SYMBOL	TYPE	DESCRIPTION
CHAR	L*1	OUTPUT - CURRENT DATE IN THE FORM MM/DD/YY (MONTH, DAY, YEAR)

SUBROUTINE USED NOW

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

RESTRICTIONS NONE

REFERENCES NONE

```
      SUBROUTINE DATE (CHAR)
      DIMENSION CHAR(8),IDAYS(12),IDATE(3)
      LOGICAL*1 CHAR,LCHAR1,LCHAR2,LSLASH,DUM(2)
      INTEGER*2 ICHAR
      EQUIVALENCE (ICHAR,DUM(1),LCHAR1),(DUM(2),LCHAR2),(IDATE(2),ID)
C EBCDIC ZERO,SLASH, AND ONE CHARACTER SHIFT CONSTANT
      DATA IZ,LSLASH,ISHIFT /ZF0,1H/,256 /
C ELAPSED DAYS OF YEAR FOR EACH MONTH
      DATA IDAYS /31,59,90,120,151,181,212,243,273,304,334,365 /
C RECOVER DAY OF YEAR IN IBM FORMAT (YYDDD)
      CALL NOW (IYDD,DUMMY)
C GET YEAR,MONTH DAY
      IDATE(3)=IYDD/1000
      ID=MOD(IYDD,1000)
      ICOR=0
      IF (MOD(IDATE(3),4).EQ.0.AND.ID.GT.31) ICOR=1
      DO 10 I=1,12
      IF (ID.LE.IDAYS(I)+ICOR) GO TO 20
 10   CONTINUE
 20   IDATE(1)=I
      IF (I-2) 50,40,30
 30   ID=ID-ICOR
 40   ID=ID-IDAYS(I-1)
C CONVERT DATE TO ALPHANUMERIC - REMEMBER EQUIVALENCES
 50   J=1
      DO 60 I=1,3
      ICHAR=(IDATE(I)/10+IZ)*ISHIFT+MOD(IDATE(I),10)+IZ
      CHAR(J)=LCHAR1
      CHAR(J+1)=LCHAR2
```

```
 60     J=J+3
        CHAR(3)=LSLASH
        CHAR(6)=LSLASH
C ALL DONE
        RETURN
        END
```

EDIT

DESCRIPTION

EDIT is used to convert any single precision number (integer or floating point) to an equivalent character array. EDIT is used for labeling values. Such routines as QUICKY and GRID use EDIT although the user often has the need to call EDIT.

EDIT first determines the characteristics of the format. The type (E, F or I), the width, the decimal point position and the power factor must be determined.

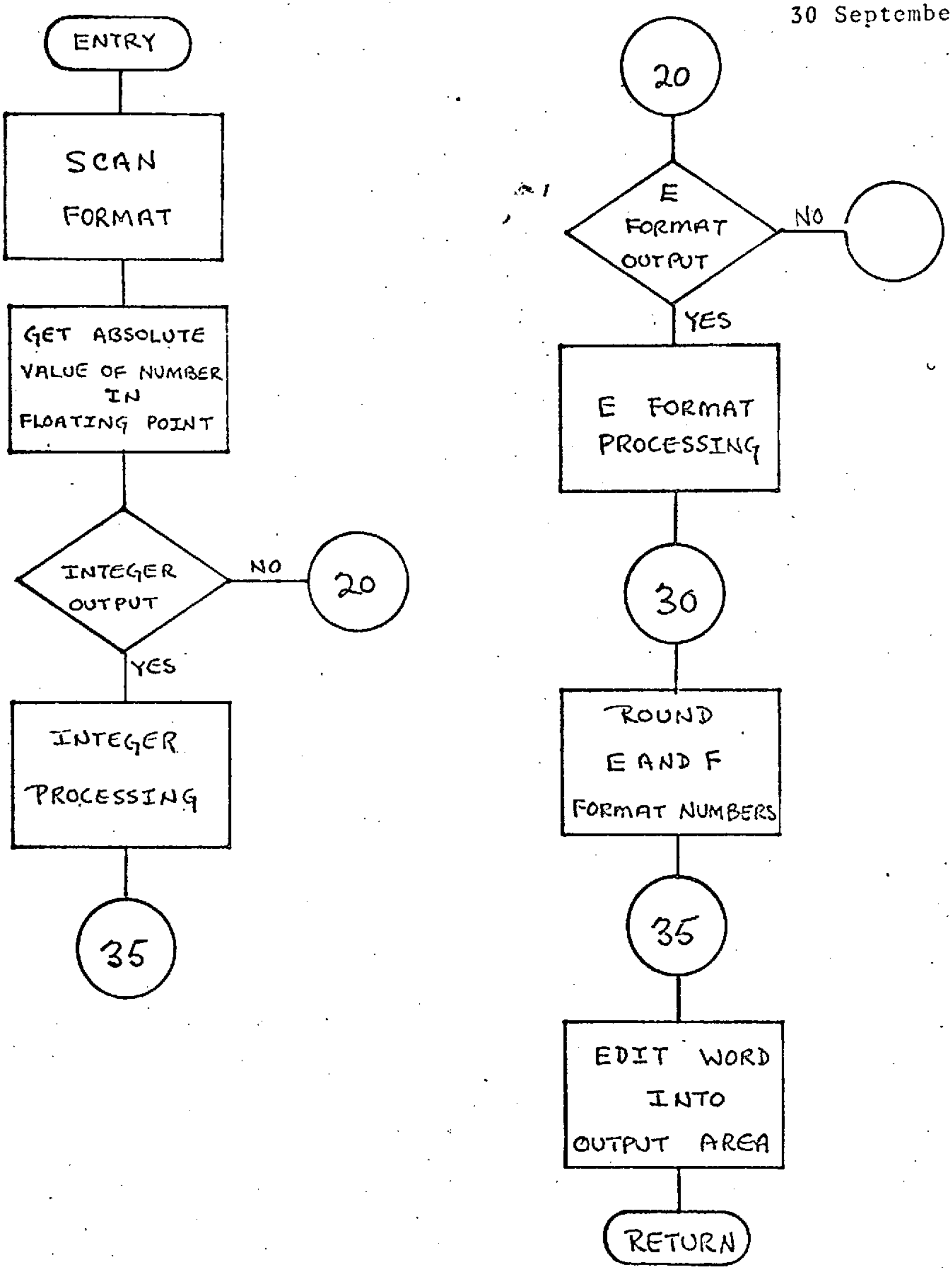
Then by using divisions by 10 to determine the least significant digits and modulo 10 to determine the remaining higher order digits, the number is converted to character codes.

NAME		EDIT
PURPOSE		TO CONVERT AND EDIT BINARY NUMBERS TO EBCDIC
CALLING SEQUENCE		CALL EDIT(A,FORM,OUT,N)
SYMBOL	TYPE	DESCRIPTION
A (1)	R	INPUT - NUMBER TO BE CONVERTED
FORM (K)	A*1	INPUT - EDITING FORMAT (K IS THE NUMBER OF BYTES IN THE FORMAT)
OUT (W)	A*1	OUTPUT - EDITED EBCDIC NUMBER (W IS THE REQUESTED FIELD WIDTH)
N (1)	I	OUTPUT - NUMBER OF PRINTABLE CHARACTERS
SUBROUTINES USED		NONE
COMMON BLOCKS		NONE
INPUT FILES		NONE
OUTPUT FILES		NONE
RESTRICTIONS		NONE
REFERENCES		NONE

```
      SUBROUTINE EDIT (A,FORM,OUT,N)
      LOGICAL*1 SETDIG,LDIG,LMODE,OUT,FORM,NEG,ZSW
      INTEGER SPECS,W,D,EE,PER,BLANK,PLUS,P,ZZ
      DIMENSION SPECS(3),SETDIG(36),OUT(1),FORM(1)
      EQUIVALENCE (SPECS(1),W),(SPECS(2),D),(SPECS(3),P),(M,B1)
      EQUIVALENCE (SETDIG(1),PLUS),(SETDIG(5),MINUS),(SETDIG(9),ZZ),
     . (SETDIG(13),PER),(SETDIG(17),II),(SETDIG(21),EE),
     . (SETDIG(25),BLANK),(SETDIG(29),IDIG),(SETDIG(32),LDIG),
     . (SETDIG(33),MODE),(SETDIG(36),LMODE)
      DATA SETDIG /
     . Z00,Z00,Z00,1H+,Z00,Z00,Z00,1H-,Z00,Z00,Z00,1H0,
     . Z00,Z00,Z00,1H.,Z00,Z00,Z00,1HI,Z00,Z00,Z00,1HE,
     . Z00,Z00,Z00,1H ,Z00,Z00,Z00,Z00,Z00,Z00,Z00,Z00   /
C CLEAR FORMAT SCAN BUFFER
      DO 5 I=1,3
    5 SPECS(I)=0
C SCAN FORMAT
      J=1
      LDIG=FORM(1)
      MODE=IDIG
```

```
      NSIGN=1
      I=1
 7    I=I+1
      LDIG=FCRM(I)
      IF (IDIG.LT.ZZ.OR.IDIG.GT.ZZ+9) GO TO 10
      SPECS(J)=SPECS(J)*10+IDIG-ZZ
      GO TO 7
 10   IF (IDIG.NE.PER) GO TO 15
      SPECS(J)=SPECS(J)*NSIGN
      NSIGN=1
      J=J+1
      GO TO 7
 15   IF (IDIG.EQ.PLUS) GO TO 7
      IF (IDIG.NE.MINUS) GO TO 18
      NSIGN=-1
      GO TO 7
 18   N=W
C GET ABSOLUTE VALUE OF NUMBER IN FLOATING POINT
      NEG=A.LT.0.
      B=ABS(A)
      BI=A
      M=IABS(M)
      IF (M.LT.15728641) B=M
      IF(MODE.NE.II) GO TO 20
C INTEGER PROCESSING
      D=-1
      B=B+.1
      GO TO 35
 20   IF (MODE.NE.EE) GO TO 30
C E FORMAT PROCESSING
      D=MINO(D,N-4)
      W=MAXO(D+1,N-4)
      IF (B.GT.0.) IPOW=-MINT(-ALOGIO(B))-P
      B=B*10.0**(-IPOW)+.5*10.0**(-D)
      IF (B.LT.10.0**P) GO TO 35
      B=B/10.0
      IPOW=IPOW+1
      GO TO 35
C ROUND E AND F FORMAT NUMBERS
 30   B=B+.5*10.**(-D)
C EDIT WORD INTO OUTPUT AREA
 35   IPER=W-D
      POW=10.**(IPER-2)
      IW=1
      IWM=0
      ZSW=.FALSE.
      I=0
 38   I=I+1
      IF(I.NE.IPER) GO TO 40
      IDIG=PER
      GO TO 60
 40   IDIG=MOD(INT(B/POW),10)+ZZ
      POW=POW/(10.)
      IF(ZSW.OR.IDIG.NE.ZZ.OR.I.EQ.IPER-1.OR.I.EQ.W) GO TO 60
 50   IWM=IW
      IDIG=BLANK
```

```
          GO TO 70
   60     ZSW=.TRUE.
   70     IRTN=0
          GO TO 200
   80     IF(I.LT.W) GO TO 38
          IF(MOLE.NE.EE) GO TO 130
          IF(W.GT.N-4) GO TO 90
          IDIG=EE
          GO TO 200
   90     IRTN=IRTN+1
  100     IDIG=PLUS
          IF(IPOW.LT.0) IDIG=MINUS
          IPOW=IABS(IPOW)
          GO TO 200
  110     IDIG=IPOW/10+ZZ
          GO TO 200
  120     IDIG=MOD(IPOW,10)+ZZ
          GO TO 200
  130     IF(IWM.EQ.0.OR..NOT.NEG) RETURN
          IDIG=MINUS
          IW=IWM
          IRTN=5
  200     OUT(IW)=IDIG
          IW=IW+1
          IRTN=IRTN+1
          GO TO (80,100,110,120,130,140),IRTN
  140     RETURN
          END
```


ENTRY
SCAN FORMAT
GET ABSOLUTE VALUE OF NUMBER IN FLOATING POINT
INTEGER OUTPUT
NO
20
YES
INTEGER PROCESSING
35
20
E FORMAT OUTPUT
NO
YES
E FORMAT PROCESSING
30
ROUND E AND F FORMAT NUMBERS
35
EDIT WORD INTO OUTPUT AREA
RETURN

GRDNUM

DESCRIPTION

GRDNUM is a routine which computes certain characteristics of an array of data values (usually either x values or y values). These characters are used in calls to GRID or OGRID. Each characteristic can also be had separately by a separate entry and calls to GRDNUM go through each of these entries.

MAXMIN determines FMIN and FMAX (the minimum and maximum values of the array, respectively).

PTYNUM, given FMIN and FMAX, determine rounded values PMIN and PMAX of FMIN and FMAX, resp., such that [FMIN, FMAX] lies in [PMIN, PMAX] and PMIN and PMAX are esthetically nice boundaries. NINT, the suggested number of intervals in [PMIN, PMAX] is also determined.

Finally, FORMAT determines a good format for numbers in [PMIN, PMAX]. An F format is usually produced unless the values are either too large or too small in which case E9.2.1 is used.

NAME GRDNUM

ENTRY POINT	PURPOSE
GRDNUM	TO COMPUTE ARGUMENTS TO GRID AND OGRID
MAXMIN	TO FIND ARRAY MAXIMUM AND MINIMUM VALUES
PTYNUM	TO COMPUTE ESTHETIC PLOTTING LIMITS ON DATA
FORMAT	TO GENERATE A FORMAT CODE TO LABEL NUMBERS WHOSE VALUES LIE BETWEEN PMIN AND PMAX (FOR USE WITH 'EDIT')

CALLING SEQUENCE CALL GRDNUM(ARRAY,N,PMIN,PMAX,NINT,FMT)

SYMBOL	TYPE	DESCRIPTION
ARRAY	R	INPUT - PLOTTING ARRAY
N	I	INPUT - NUMBER OF ITEMS IN ARRAY
PMIN	R	OUTPUT - SUGGESTED PLOTTING MINIMA
PMAX	R	OUTPUT - SUGGESTED PLOTTING MAXIMA
NINT	I	OUTPUT - SUGGESTED NUMBER OF INTERVALS
FMT	A	OUTPUT - SUGGESTED LABELING FORMAT

CALLING SEQUENCE CALL MAXMIN(ARRAY,N,FMIN,FMAX)

SYMBOL	TYPE	DESCRIPTION
ARRAY	R	INPUT - THE ARRAY
N	I	INPUT - NUMBER OF ITEMS IN THE ARRAY
FMIN	R	OUTPUT - ARRAY MINIMA
FMAX	R	OUTPUT - ARRAY MAXIMA

CALLING SEQUENCE CALL PTYNUM(FMIN,FMAX,PMIN,PMAX,NINT)

SYMBOL	TYPE	DESCRIPTION
FMIN	R	INPUT - ARRAY MINIMA
FMAX	R	INPUT - ARRAY MAXIMA
PMIN	R	OUTPUT - SUGGESTED PLOTTING MINIMA
PMAX	R	OUTPUT - SUGGESTED PLOTTING MAXIMA
NINT	I	OUTPUT - SUGGESTED NUMBER OF INTERVALS

```
CALLING SEQUENCE CALL FORMAT(PMIN,PMAX,FMT)

   SYMBOL  TYPE   DESCRIPTION

   PMIN    R      INPUT - SUGGESTED PLOTTING MINIMA

   PMAX    R      INPUT - SUGGESTED PLOTTING MAXIMA

   FMT     A      OUTPUT - SUGGESTED LABELING FORMAT

 SUBROUTINES USED  NONE

 COMMON BLOCKS     NONE

 INPUT FILES       NONE

 OUTPUT FILES      NONE

 RESTRICTIONS      NONE

 REFERENCES        NONE
```

```
      SUBROUTINE GRDNUM(ARRAY,N,PMIN,PMAX,NINT,FMT)                     GRDN  02
      DIMENSION ARRAY(N)                                                GRDN  03
      LOGICAL*1 NUMS(10)/'0123456789'/,EFMT(7)/'E9.2,1)'/,F/'F'/,       GRDN  04
     *          FMT(1),PAREN,POINT,GRDSW/.TRUE./,ANI/'I'/               GRDN  05
      EQUIVALENCE (PAREN,EFMT(7)),(POINT,EFMT(5))                       GRDN  06
      GRDSW=.FALSE.                                                     GRDN  07
C MAXMIN ENTRY                                                          GRDN  08
      ENTRY MAXMIN(ARRAY,N,FMIN,FMAX)                                   GRDN  09
C FIND ARRAY MAXIMUM AND MINIMUM                                        GRDN  90
      FMIN=ARRAY(1)                                                     GRDN  91
      FMAX=FMIN                                                         GRDN  92
      IF(N.LT.2) GO TO 20                                               GRDN  93
      DO 10 I=2,N                                                       GRDN  94
      IF(ARRAY(I).LT.FMIN)FMIN=ARRAY(I)                                 GRDN  95
 10   IF(ARRAY(I).GT.FMAX)FMAX=ARRAY(I)                                 GRDN  96
 20   IF(GRDSW)RETURN                                                   GRDN  97
C PTYNUM ENTRY                                                          GRDN  98
      ENTRY PTYNUM(FMIN,FMAX,PMIN,PMAX,NINT)                            GRDN  99
      NINT=0                                                            GRDN 100
      IF(FMIN.EQ.FMAX) GO TO 50                                         GRDN 101
C COMPUTE ESTHETIC PLOTTING LIMITS                                      GRDN 102
      DMAG=10.**(-NINT(-ALOG10(ABS(FMAX-FMIN)))-1)                      GRDN 103
      NL=NINT(FMIN/DMAG)                                                GRDN 104
      PMIN=FLOAT(NL)*DMAG                                               GRDN 105
      NH=-NINT(-FMAX/DMAG)                                              GRDN 106
      PMAX=FLOAT(NH)*DMAG                                               GRDN 107
      NINT=NH-NL                                                        GRDN 108
      NINT=11/NINT*NINT                                                 GRDN 109
      IF(NH-NL.EQ.3)NINT=15                                             GRDN 110
      IF(GRDSW)RETURN                                                   GRDN 111
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
C FORMAT ENTRY
      ENTRY FORMAT(PMIN,PMAX,FMT)
C GENERATE FORMAT TO BE USED WITH PMIN AND PMAX
      MAG=INT(1.+ALOG10(AMAX1(ABS(PMAX),ABS(PMIN))))
      IF(MAG.LT.1)MAG=1
      MAGD=-MINT(ALOG10(ABS(PMAX-PMIN)))+1
      IF(MAGD.GT.0) GO TO 25
      IF(MAG.GT.7) GO TO 30
      FMT(1)=ANI
      FMT(2)=NUMS(MAG+2)
      FMT(3)=PAREN
      GO TO 50
   25 IW=MAG+MAGD+2
      IF(IW.GT.8) GO TO 30
      FMT(1)=F
      FMT(2)=NUMS(IW+1)
      FMT(3)=POINT
      FMT(4)=NUMS(MAGD+1)
      FMT(5)=PAREN
      GO TO 50
   30 DO 40 I=1,7
   40 FMT(I)=EFMT(I)
   50 GRDSW=.TRUE.
      RETURN
      END
```

GRID

DESCRIPTION

GRID is the major user routine in the plot package. It is used to set up the scaling factors between subject space and object space, draw grid overlays, to plot vectors.

The object space (the space in the raster world on the SC4020 screen) is defined by SETGRD. If SETGRD is not called, a default object space is used.

The subject space (the space in which the user units exist) is defined by GRID, OGRID, POLAR or SCALE. SCALE defines this space and also sets up the scaling factors and initializations for the routine GRID. GRID and OGRID use the same coding as SCALE (and therefore perform the same function) but also plot a grid overlay. GRID plots a full grid while OGRID plots a partial grid with tick marks. POLAR is essentially the same as GRID except it is used for polar plots.

PLOT is used to plot vectors or characters. Given an array of coordinates in subject space and using the scaling factors from SCALE, PLOT determines corresponding coordinates in the object space and then either prints characters at the points or draws vectors connecting these points. PPLOT is the same as PLOT except it is used for polar data.

COORD is used to recovery the raster coordinate corresponding to any coordinate in the subject space. NEGLOG specifies the error procedure for negative logs and INTENS is used to set character intensity.

NAME	GRID
ENTRY POINT	PURPOSE
GRID	TO COMPUTE THE NECESSARY SCALING FOR SUBROUTINE 'PLOT' AND TO PLOT AND LABEL GRID LINES
OGRID	TO COMPUTE THE NECESSARY SCALING FOR SUBROUTINE 'PLOT' AND LABEL AN OPEN GRID
POLAR	TO COMPUTE THE NECESSARY SCALING FOR 'PLOT' &'PPLOT' AND DRAW AND LABEL A POLAR GRID
SCALE	TO RECOMPUTE THE SCALING FOR SUBROUTINE 'PLOT'
SETGRD	TO SET THE LIMITS FOR THE GRID
INTENS	TO SET THE INTENSITY FOR SUBROUTINE 'PLOT'
NEGLOG	TO ENABLE THE PLOTTING OF NEGATIVE ARGUMENTS LOGARITHMICALLY IN 'PLOT' WITH EITHER A DIFFERENT SYMBOL OR SUPERPOSITION OF SYMBOLS
PLOT	TO PLOT A SET OF POINTS OR A SERIES OF CONTIGUOUS VECTORS
PPLOT	TO PLOT A SET OF POINTS OR A SERIES OF CONTIGUOUS VECTORS IN POLAR COORDINATES
CCORD	TO RECOVER THE RASTER COORDINATES OF A POINT

CALLING SEQUENCE CALL GRID(XLO,XHI,NX,A,NXS,YLO,YHI,NY,B,NYS,LOG)

SYMBOL	TYPE	DESCRIPTION
XLO	R	INPUT - LOWEST VALUE OF ABSCISSA (LOWEST VALUE OF X) AT LEFT SIDE OF GRID
XHI	R	INPUT - HIGHEST VALUE OF ABSCISSA (HIGHEST VALUE OF X) AT RIGHT SIDE OF GRID
NX	I	INPUT - NUMBER OF INTERVALS ON ABSCISSA.
A	A	INPUT - LABELING FORMAT FOR ABSCISSA
NXS	I	INPUT - NUMBER OF INTERVALS AT WHICH TO LABEL THE X-AXIS
YLO	R	INPUT - VALUE OF ORDINATE (LOWEST VALUE OF Y) AT BOTTOM OF GRID
YHI	R	INPUT - VALUE OF ORDINATE (HIGHEST VALUE OF Y) AT TOP OF GRID
NY	I	INPUT - NUMBER OF INTERVALS ON ORDINATE

```
B         A       INPUT - LABELING FORMAT FOR ORDINATE

NYS       I       INPUT - NUMBER OF INTERVALS AT WHICH TO LABEL THE
                          Y-AXIS

LOG       I       INPUT - I=0 LINEAR GRID
                          I=1 ABSCISSA IS LOGARITHMIC
                              ORDINATE IS LINEAR
                          I=2 ABSCISSA IS LINEAR
                              ORDINATE IS LOGARITHMIC
                          I=3 LOGARITHMIC GRID

CALLING SEQUENCE CALL DGRID(XLO,XHI,NX,A,NXS,YLO,YHI,NY,B,NYS,LOG)

 SYMBOL  TYPE   DESCRIPTION

  (SEE CALLING SEQUENCE DESCRIPTION FOR ENTRY POINT GRID)

CALLING SEQUENCE CALL POLAR(RADIUS,NX,A,NXS,IRDH)

 SYMBOL  TYPE   DESCRIPTION

 RADIUS  R      INPUT - VALUE OF OUTER CIRCLE

 NX      I      INPUT - NUMBER OF CONCENTRIC CIRCLES

 A       A      INPUT - LABELING FORMAT FOR CONCENTRIC CIRCLES

 NXS     I      INPUT - NUMBER OF CIRCLES AT WHICH TO LABEL

 IRDH    I      INPUT - INDICATOR FOR LABELING RADIALS:
                        =1 LABEL RADIALS ARE IN DEGREES
                        =2 LABEL RADIALS ARE IN HOURS

CALLING SEQUENCE CALL SCALE(XLO,XHI,YLO,YHI,LOG)

 SYMBOL  TYPE   DESCRIPTION

 (SEE CALLING SEQUENCE DESCRIPTION FOR ENTRY POINT GRID)

CALLING SEQUENCE CALL SETGRD(XLOLIM,YLOLIM,XHILIM,YHILIM)

 SYMBOL  TYPE   DESCRIPTION

 XLOLIM  R      INPUT - LOWEST ABSCISSA POSITION (IN RASTERS)

 YLOLIM  R      INPUT - LOWEST ORDINATE POSITION (IN RASTERS)

 XHILIM  R      INPUT - HIGHEST ABSCISSA POSITION (IN RASTERS)

 YHILIM  R      INPUT - HIGHEST ORDINATE POSITION (IN RASTERS)

CALLING SEQUENCE CALL INTENS(IT)

 SYMBOL  TYPE   DESCRIPTION
```

SYMBOL	TYPE	DESCRIPTION
IT	I	INPUT - NUMBER FROM 0-15 SPECIFYING DESIRED INTENSITY OF CHARACTER PLOTTING

CALLING SEQUENCE CALL NEGLOG(OVERPL,NCHAR)

SYMBOL	TYPE	DESCRIPTION
OVERPL	L	INPUT - OVERPLOT GIVEN SYMBOL WITH NCHAR IF TRUE
NCHAR	I	INPUT - RIGHT ADJUSTED PLOT CHARACTER FOR USE WITH NEGATIVE ARGUMENTS IN POINT PLOT SECTION OF SUBROUTINE 'PLOT' BLANK SUPPRESSES PLOTTING OF NEGATIVE POINTS

CALLING SEQUENCE CALL PLOT(X,Y,N,CHAR)

SYMBOL	TYPE	DESCRIPTION
X	R	INPUT - ARRAY OF 'X' VALUES TO BE PLOTTED
Y	R	INPUT - ARRAY OF 'Y' VALUES TO BE PLOTTED
N	I	INPUT - NUMBERED OF ORDERED PAIRS TO BE PLOTTED
CHAR	I	INPUT - PLOT CHARACTER, RIGHT ADJUSTED

CALLING SEQUENCE CALL PPLOT(X,Y,N,CHAR,IRDH)

SYMBOL	TYPE	DESCRIPTION
X	R	INPUT - ARRAY OF RADIUS VALUES TO BE PLOTTED
Y	R	INPUT - ARRAY OF ANGULAR VALUES TO BE PLOTTED
N	R	INPUT - NUMBER OF ORDERED PAIRS TO BE PLOTTED
CHAR	I	INPUT - PLOT CHARACTER, RIGHT ADJUSTED
IRDH	I	INPUT - INDICATOR OF UNITS OF ANGULAR INPUT =0 ANGLES ARE IN RADIANS =1 ANGLES ARE IN DEGREES =2 ANGLES ARE IN HOURS

CALLING SEQUENCE CALL COORD(X,Y,KX,KY)

SYMBOL	TYPE	DESCRIPTION
X	R	INPUT - X VALUE
Y	R	INPUT - Y VALUE
KX	I	OUTPUT - X RASTER VALUE
KY	I	OUTPUT - Y RASTER VALUE

SUBROUTINES USED	FDIT HORLIN SC4020
COMMON BLOCK	CPLOT$
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	XLO.NE.XHI AND YLO.NE.YHI IF USED, SETGRD MUST BE CALLED BEFORE CORRESPONDING CALLS TO GRID, SCALE, OR OGRID
REFERENCES	NONE

```
      SUBROUTINE GRID (XLO,XHI,NX,A,NXS,YLO,YHI,NY,B,NYS,LOG)
      COMMON /CPLOT$/ G1(2),XLOG(2),XLOLIM(2),XHILIM(2),XSCAL(2),
     . FXLO(2),IT,G2(4)
      LOGICAL LOGX,LOGY,OPEN,SCALSW,XLOG,OVERPL,OVRPLT,NEG,NEGSW,
     . ANGLE
      LOGICAL*1 CHAR(4),IARRAY(4),NCHAR(4),LCHAR
      INTEGER UP
      INTEGER XV,YV,BLANK
      REAL LOLIMX,LOLIMY
      DIMENSION II(2),CP(2),FXHI(2),OUT(5),IX(2),IY(2),A(1),B(1)
      DIMENSION ORG(2),CS(2),DCS(2),KXY(2)
      DIMENSION NLABEL(2)
      DIMENSION X(1),Y(1)
      REAL FDH(3)/1.,57.29578,3.819718/,QUAD(2,2)/0.,1.,3.,2./,
     . PM(2)/1.,-1./,PI12TH/.2617994/
      REAL SIN5/.8715574E-1/,COS5/.9961947/
      EQUIVALENCE (KXY(1),KX1),(KXY(2),KY1)
      EQUIVALENCE (IARRAY(1),ICHAR),(IARRAY(4),LCHAR)
      EQUIVALENCE (XSCAL(1),SCALEX),(XSCAL(2),SCALEY),
     . (FXLO(1),SXLO),(FXLO(2),SYLO)
      EQUIVALENCE (XLOG(1),LOGX),(XLOG(2),LOGY)
      DATA BLANK,ICHAR /4H    ,4H    /
      DATA CP /10,9/
      DATA IZ /0/
      DATA SCALSW,OPEN,OVRPLT,NEGSW,ANGLE/5*.FALSE./
      DATA KXY/2*2/
      GO TO 5
C OGRID ENTRY
      ENTRY OGRID (XLO,XHI,NX,A,NXS,YLO,YHI,NY,B,NYS,LOG)
      OPEN=.TRUE.
      GO TO 5
C SCALE ENTRY
      ENTRY SCALE (XLO,XHI,YLO,YHI,LOG)
      SCALSW=.TRUE.
C INITIALIZE ARRAYS AND ENTRY SWITCHES
 5    FXLO(1)=XLO
      FXLO(2)=YLO
      FXHI(1)=XHI
```

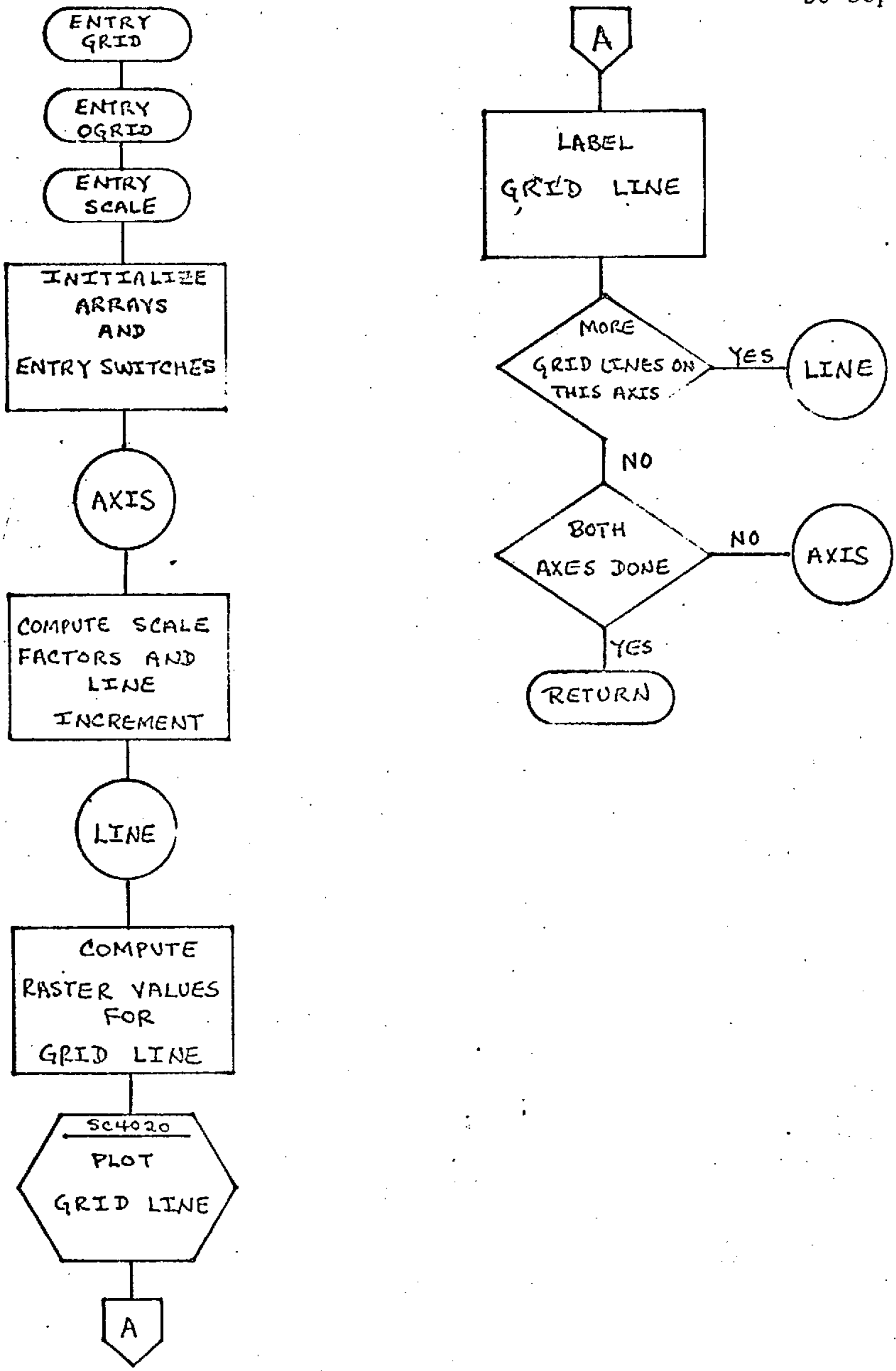
```
      FXHI(2)=YHI
      II(1)=NX
      II(2)=NY
      NLABEL(1)=NXS
      NLABEL(2)=NYS
C LOOP FOR X THEN Y AXIS PROCESSING
      DO 50 IA=1,2
      XLOG(IA)=LOG.EQ.IA.OR.LOG.EQ.3
C COMPUTE SCALE FACTORS AND LINE INCREMENTS
      IF (XLOG(IA)) GO TO 10
      XSCAL(IA)=(XHILIM(IA)-XLOLIM(IA))/(FXHI(IA)-FXLO(IA))
      IF (SCALSW) GO TO 50
      J=II(IA)
      XINC=(FXHI(IA)-FXLO(IA))/FLOAT(J)
      GO TO 15
 10   IF (.NOT.SCALSW) GO TO 12
      FXLO(IA)=ALOG10(FXLO(IA))
      FXHI(IA)=ALOG10(FXHI(IA))
      XSCAL(IA)=(XHILIM(IA)-XLOLIM(IA))/(FXHI(IA)-FXLO(IA))
      GO TO 50
C COMPUTE LOG SPACING INTERVAL ALONG AXIS
 12   NMIN=II(IA)/9
      NINC=II(IA)+NMIN
      XINC=10./FLOAT(NINC)
      NLABEL(IA)=NLABEL(IA)+NLABEL(IA)/9
      XI=FXLO(IA)
      I=MINT(ALOG10(XI))
      NUM=XI/(XINC*10.**I)
      XI=I
      IF (NUM.GT.NMIN) XI=XI+ALOG10(FLOAT(NUM)*XINC)
      IF (NUM.LE.NMIN) NUM=0
      FXLO(IA)=XI
      ISTART=NINC*I+NUM
      XI=FXHI(IA)
      I=MINT(ALOG10(XI))
      NUM=-MINT(-XI/(XINC*10.**I))
      XI=I
      IF (NUM.GT.NMIN) XI=XI+ALOG10(FLOAT(NUM)*XINC)
      IF (NUM.LE.NMIN) NUM=0
      FXHI(IA)=XI
      XSCAL(IA)=(XHILIM(IA)-XLOLIM(IA))/(FXHI(IA)-FXLO(IA))
      J=NINC*I+NUM-ISTART
 15   IB=MOD(IA,2)+1
      IX(IB)=XLOLIM(IB)
C LOOP FOR EACH GRID LINE ON THIS AXIS
      DO 40 I=IZ,J
C COMPUTE RASTER VALUES FOR GRID LINE
      IF (XLOG(IA)) GO TO 20
      XI=XINC*FLOAT(I)
      IX(IA)=XI*XSCAL(IA)+XLOLIM(IA)
      XI=XI+FXLO(IA)
      K=1
      GO TO 30
 20   L=I+ISTART
      K=MOD(L,NINC)
      IF (K.LT.0) K=K+NINC
```

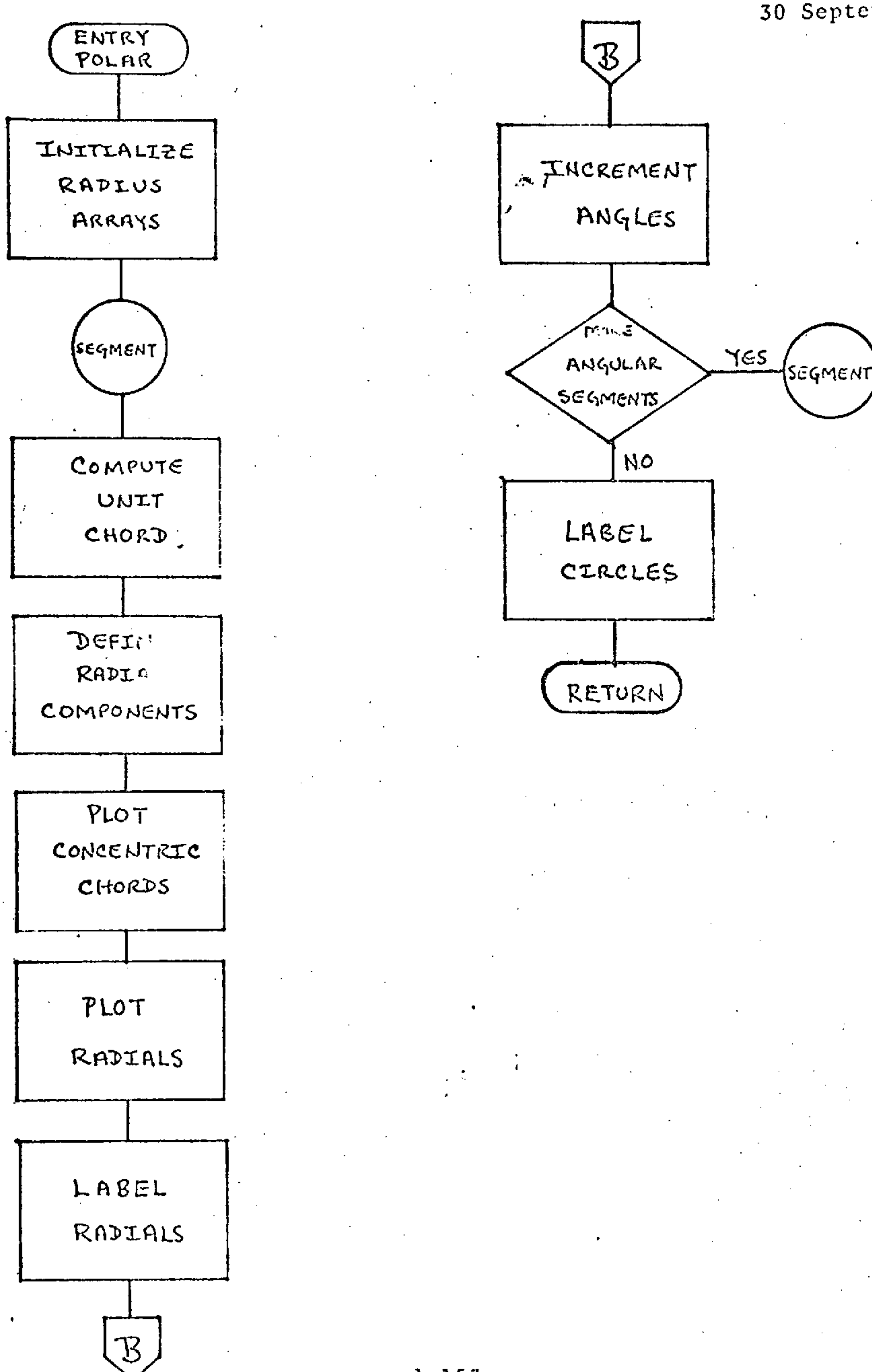
```
      XI=(L-K)/NINC
      IF (K.EQ.0) GO TO 25
      IF (K.LE.KMIN) GO TO 40
      XI=XI+ALOG10(FLOAT(K)*XINC)
 25   IX(IA)=(XI-FXLO(IA))*XSCAL(IA)+XLOLIM(IA)
      XI=10.**XI
C PLOT GRID LINE
 30   ISTOP=XHILIM(IB)
      IF(OPEN.AND..NOT.(I.EQ.IZ.OR.I.EQ.J))ISTOP=XLOLIM(IB)+8.
      IV(IB)=ISTOP-IX(IB)
      IF(IABS(IV(IB)).GT.63) GO TO 33
      IV(IA)=0
      CALL SC4020 (11,IX(1),IX(2),IV(1),IV(2))
      GO TO 34
 33   CALL SC4020(OP(IA),IX(1),IX(2),ISTOP,D)
C LABEL GRID LINE
 34   IF(NLABEL(IA).EQ.0.OR.MOD(K,NLABEL(IA)).NE.0) GO TO 40
      IF (IA.NE.1) GO TO 35
      CALL EDIT(XI,A,OUT,N)
      CALL HORLIN (OUT,N,IX(1),IX(2)-16)
      GO TO 40
 35   CALL EDIT(XI,B,OUT,N)
      CALL HORLIN (OUT,N,IX(1)-4*N,IX(2))
 40   CONTINUE
 50   CONTINUE
      OPEN=.FALSE.
      SCALSW=.FALSE.
      RETURN
C POLAR ENTRY
      ENTRY POLAR(RADIUS,NX,A,NXS,IRDH)
      RAD=AMIN1((XHILIM(1)-XLOLIM(1))/FLOAT(KXY(1)),
     *          (XHILIM(2)-XLOLIM(2))/FLOAT(KXY(2)))
      RAD=AMIN1(RAD,63./SINS)
      RINC=RAD/FLOAT(NX)
C INITIALIZE RADIUS ARRAYS
      DO 510 I=1,2
      ORG(I)=XLOLIM(I)+RAD*FLOAT(KXY(I)-1)
      FXLO(I)=-RADIUS*FLOAT(KXY(I)-1)
      XSCAL(I)=RAD/RADIUS
 510  XLOG(I)=.FALSE.
      CS(1)=1.
      CS(2)=0.
C DRAW ARCS IN 5 DEGREE SEGMENTS
      DO 550 I=1,18
C COMPUTE UNIT CHORD
      DCS(1)=CS(1)*COS5-CS(2)*SIN5-CS(1)
      DCS(2)=CS(2)*COS5+CS(1)*SIN5-CS(2)
C DEFINE RADIAL COMPONENTS
      OFFC=0.
      IF(MOD(I,9).NE.1)OFFC=16.
      IF(MOD(I,3).NE.1)OFFC=32.
      NVEC=(AMAX1(CS(1),CS(2))*(RAD-OFFC)-1.)/63.
      XYV=(RAD-OFFC)/FLOAT(NVEC+1)
C LOOP ON QUADRANTS
      DO 540 K1=1,KX1
      DO 540 K2=1,KY1
```

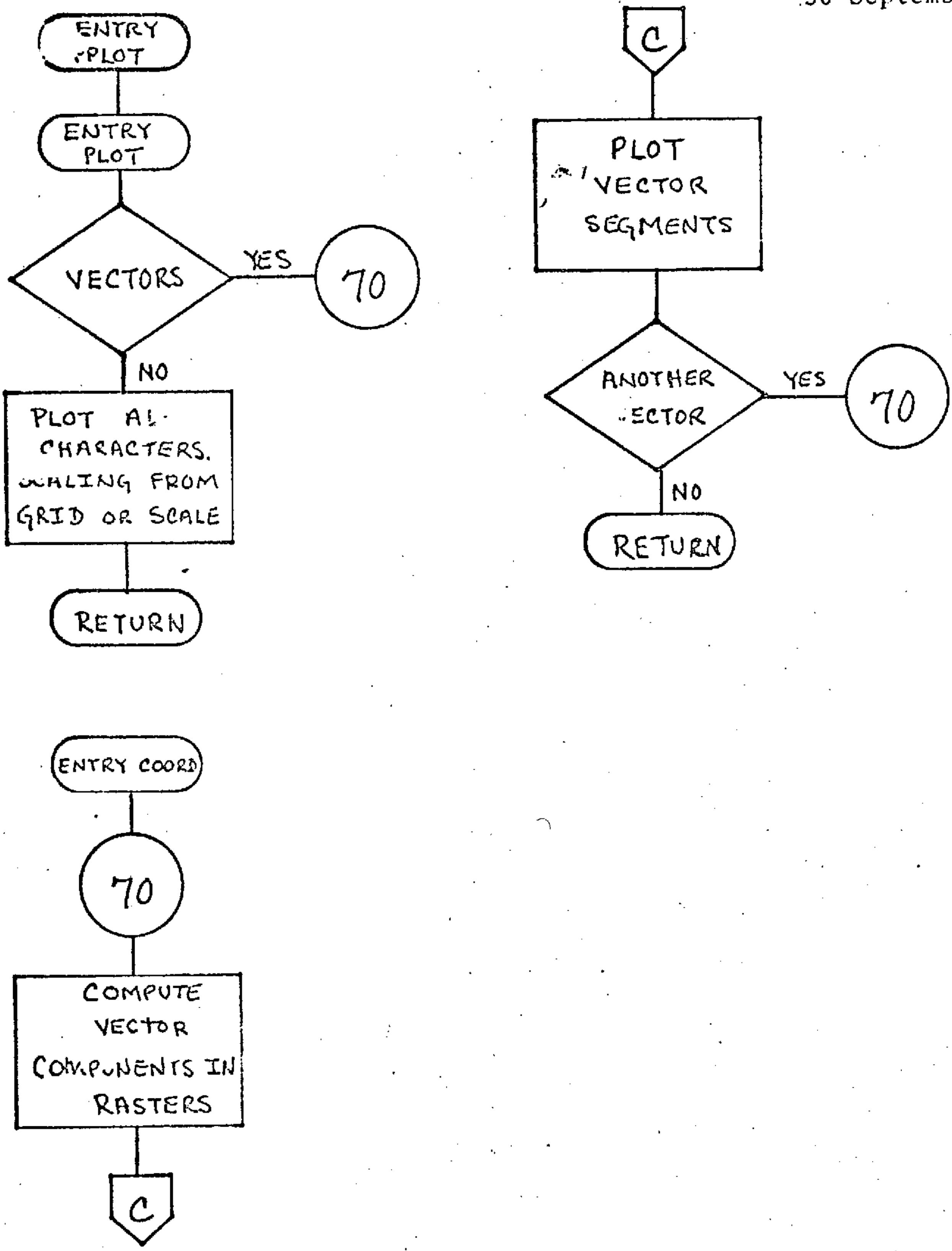
```
      K=IABS(K1-K2)+1
      KT=MOD(K,2)+1
      RX=CS(K)*PM(K1)
      RY=CS(KT)*PM(K2)
      RXV=DCS(K)*PM(K1)
      RYV=DCS(KT)*PM(K2)
C PLOT CONCENTRIC CHORDS
      DO 520 J=1,NX
      RJ=RINC*FLOAT(J)
      IX(1)=RJ*RX+ORG(1)
      IX(2)=RJ*RY+ORG(2)
      IV(1)=RXV*RJ
      IV(2)=RYV*RJ
 520  CALL SC4020 (11,IX(1),IX(2),IV(1),IV(2))
C PLOT RADIALS
      IV(1)=RX*XYV
      IV(2)=RY*XYV
      DO 530 J=IZ,NVEC
      RJ=XYV*FLOAT(J)+OFFC
      IX(1)=RX*RJ+ORG(1)
      IX(2)=RY*RJ+ORG(2)
 530  CALL SC4020 (11,IX(1),IX(2),IV(1),IV(2))
      IF(MOD(I,3).NE.1) GO TO 540
C LABEL RADIALS
      IX(1)=(RAD+16.)*RX+ORG(1)
      IX(2)=(RAD+16.)*RY+ORG(2)
      FNUM=(FLOAT((I-1)/3)+6.*QUAD(K1,K2))*PI12TH+RDH(IRDH+1)
      CALL EDIT(FNUM,'I3)',OUT,N)
      CALL FCRLIN(OUT,N,IX(1),IX(2))
 540  CONTINUE
C INCREMENT ANGLES
      CS(1)=CS(1)+DCS(1)
 550  CS(2)=CS(2)+DCS(2)
      IF (NXS.EQ.0) RETURN
C LABEL CIRCLES
      IX(1)=ORG(1)
      DO 560 J=NXS,NX,NXS
      FNUM=RADIUS*FLOAT(J)/FLOAT(NX)
      CALL EDIT(FNUM,A,OUT,N)
      DO 560 K2=1,KY1
      IX(2)=ORG(2)+PM(K2)*FLOAT(J)*RINC
 560  CALL FCRLIN(OUT,N,IX(1),IX(2))
      RETURN
C SETGRD ENTRY
      ENTRY SETGRD (LOLIMX,LOLIMY,HILIMX,HILIMY)
      IF (LOLIMX.LT.HILIMX.AND.LOLIMY.LT.HILIMY.AND.
     . LOLIMX.GE.0..AND.LOLIMX.LT.1024..AND.
     . HILIMX.GE.0..AND.HILIMX.LT.1024..AND.
     . LOLIMY.GE.0..AND.LOLIMY.LT.1024..AND.
     . HILIMY.GE.0..AND.HILIMY.LT.1024.) GO TO 570
      CALL FCRLIN('SETGRD ARGUMENTS OUT OF RANGE -- LIMITS NOT RESET'
     .,49,512,512)
      CALL SC4020 (17,0,0,0,0)
      RETURN
 570  XLOLIM(1)=LOLIMX
      XLOLIM(2)=LOLIMY
```

```
      XHILIM(1)=HILIMX
      XHILIM(2)=HILIMY
      RETURN
C INTENS ENTRY
      ENTRY INTENS(IT1)
      IT=IT1
      RETURN
C NEGLOG ENTRY
      ENTRY NEGLOG (OVERPL,NCHAR)
      OVRPLT=OVERPL
      LCHAR=NCHAR(4)
      NEGSW=ICHAR.NE.BLANK
      RETURN
C PPLOT ENTRY
      ENTRY PPLOT(X,Y,N,CHAR,IRDH)
      ANGLE=.TRUE.
      RAD=1./RDH(IRDH+1)
C PLOT ENTRY
      ENTRY PLOT (X,Y,N,CHAR)
      LCHAR=CHAR(4)
      IF (ICHAR.EQ.BLANK) GO TO 70
C PLOT ALL CHARACTERS - SCALING FROM "GRID" ROUTINE
      ITX=5
      IF (IT.NE.0) ITX=6
      DO 60 I=1,N
      NEG=.FALSE.
      XX=X(I)
      IF (ANGLE) XX=X(I)*COS(Y(I)*RAD)
      IF (.NOT.LOGX) GO TO 54
      IF (XX) 51,54,53
 51   IF (.NOT.NEGSW) GO TO 54
      XX=-XX
      NEG=.TRUE.
 53   XX=ALOG10(XX)
 54   KX=(XX-SXLO)*SCALEX+XLOLIM(1)
      YY=Y(I)
      IF (ANGLE) YY=X(I)*SIN(Y(I)*RAD)
      IF (.NOT.LOGY) GO TO 58
      IF (YY) 55,58,57
 55   IF (.NOT.NEGSW) GO TO 58
      YY=-YY
      NEG=.TRUE.
 57   YY=ALOG10(YY)
 58   KY=(YY-SYLO)*SCALEY+XLOLIM(2)
C PLOT INDIVIDUAL POINTS
      IF (.NOT.NEG.OR.OVRPLT)
     * CALL SC4020 (ITX,KX,KY,LCHAR,IT)
      IF (NEG)
     * CALL SC4020 (ITX,KX,KY,NCHAR(4),IT)
 60   CONTINUE
      ANGLE=.FALSE.
      RETURN
C COORD ENTRY
      ENTRY COORD (X,Y,KX,KY)
      N=1
C LOOP ON ALL VECTORS
```

```
 70     DO 90 I=1,N
        X1=X2
        Y1=Y2
C COMPUTE VECTOR COORDINATES IN RASTERS
C SCALING TAKEN FROM GRID ROUTINE
        XX=X(I)
        IF (ANGLE) XX=X(I)*COS(Y(I)*RAD)
        IF (LOGX.AND.XX.NE.0.) XX=ALOG10(ABS(XX))
        YY=Y(I)
        IF (ANGLE) YY=X(I)*SIN(Y(I)*RAD)
        IF (LOGY.AND.YY.NE.0.) YY=ALOG10(ABS(YY))
        X2=(XX-SXLO)*SCALEX+XLOLIM(1)
        Y2=(YY-SYLO)*SCALEY+XLOLIM(2)
        IF (I.LE.1) GO TO 90
        RXV=X2-X1
        RYV=Y2-Y1
C LOOP TO PLOT VECTOR IN SEGMENTS NOT GREATER THAN 64 RASTERS
        J=MAX1(ABS(RXV),ABS(RYV))/63
        RXV=RXV/FLOAT(J+1)
        XV=SIGN(AINT(ABS(RXV)-64.)+64.,RXV)
        RYV=RYV/FLOAT(J+1)
        YV=SIGN(AINT(ABS(RYV)-64.)+64.,RYV)
C PLOT VECTOR SEGMENTS
        DO 80 K=IZ,J
        R=K
        KX=X1+R*RXV
        KY=Y1+R*RYV
        CALL SC4020 (11,KX,KY,XV,YV)
 80     CONTINUE
 90     CONTINUE
        KX=X2
        KY=Y2
        ANGLE=.FALSE.
        RETURN
        END
```


ENTRY GRID
ENTRY OGRID
ENTRY SCALE
INITIALIZE ARRAYS AND ENTRY SWITCHES
AXIS
COMPUTE SCALE FACTORS AND LINE INCREMENT
LINE
COMPUTE RASTER VALUES FOR GRID LINE
SC4020
PLOT GRID LINE
A
A
LABEL GRID LINE
MORE GRID LINES ON THIS AXIS
YES
LINE
NO
BOTH AXES DONE
NO
AXIS
YES
RETURN


ENTRY POLAR
INITIALIZE RADIUS ARRAYS
SEGMENT
COMPUTE UNIT CHORD
DEFI
RADI
COMPONENTS
PLOT CONCENTRIC CHORDS
PLOT RADIALS
LABEL RADIALS
B
B
INCREMENT ANGLES
ANGULAR SEGMENTS
YES
SEGMENT
NO
LABEL CIRCLES
RETURN


ENTRY PLOT
ENTRY PLOT
VECTORS
YES
70
NO
PLOT AL CHARACTERS, SCALING FROM GRID OR SCALE
RETURN
ENTRY COORD
70
COMPUTE VECTOR COMPONENTS IN RASTERS
C
C
PLOT VECTOR SEGMENTS
ANOTHER VECTOR
YES
70
NO
RETURN

HORLIN

DESCRIPTION

HORLIN and its two entries VERLIN and DIAGLN are used to plot an array of characters. DIAGLN is used to output a label in which the horizontal and vertical spacing or increments between characters is user specified. HORLIN (horizontal label) assumes there will be no vertical increment and a standard horizontal increment. VERLIN (vertical label) assumes there will be no horizontal increment and a standard vertical increment. Each uses the same coding.

Since the center coordinates are input, the coordinates for the first character must be computed. Then each character is output via a call to SC4020 and after each character the coordinates are incremented for the position of the next character.

```
NAME                 HORLIN

ENTRY POINT          PURPOSE

HORLIN               TO PRINT HORIZONTAL LABELS ON THE SC4020 PLOTTER

VERLIN               TO PRINT VERTICAL LABELS ON THE SC4020 PLOTTER

DIAGLN               TO PRINT DIAGONAL LABELS ON THE SC4020 PLOTTER

CALLING SEQUENCE CALL HORLIN(A,N,X,Y)

  SYMBOL   TYPE      DESCRIPTION

  A         L        INPUT - ALPHANUMERIC INFORMATION TO BE PRINTED

  N         I        INPUT - NUMBER OF CHARACTERS TO BE PRINTED

  X         I        INPUT - RASTER COUNT OF X-COORDINATE OF CENTER OF
                             LINE (RASTER COUNT OF Y-COORDINATE FOR
                             VERTICAL LABELS)

  Y         I        INPUT - RASTER COUNT OF Y-COORDINATE OF CENTER OF
                             LINE ("X" FOR VERTICAL LABELS)

CALLING SEQUENCE CALL VERLIN(A,N,X,Y)

  SYMBOL   TYPE      DESCRIPTION

  A         L        INPUT - ALPHANUMERIC INFORMATION TO BE PRINTED

  N         I        INPUT - NUMBER OF CHARACTERS TO BE PRINTED

  X         I        INPUT - RASTER COUNT OF X-COORDINATE OF CENTER
                             OF LINE
  Y         I        INPUT - RASTER COUNT OF Y-COORDINATE OF CENTER
                             OF LINE

CALLING SEQUENCE CALL DIAGLN(A,N,X,Y,DX,DY)

  SYMBOL   TYPE      DESCRIPTION

  A         L        INPUT - ALPHANUMERIC INFORMATION TO BE PRINTED

  N         I        INPUT - NUMBER OF CHARACTERS TO BE PRINTED

  X         I        INPUT - RASTER COUNT OF X-COORDINATE OF CENTER
                             OF LINE

  Y         I        INPUT - RASTER COUNT OF Y-COORDINATE OF CENTER
                             OF LINE

  DX        I        INPUT - RASTER COUNT BETWEEN CHARACTERS IN X
                             DIRECTION
```

```
DY        I        INPUT - RASTER COUNT BETWEEN CHARACTERS IN Y
                           DIRECTION

SUBROUTINE USED    SC4020

COMMON BLOCK       CPLOT$

INPUT FILES        NONE

OUTPUT FILES       NONE

RESTRICTIONS       NONE

REFERENCES         NONE
```

```
      SUBROUTINE HORLIN (A,N,X,Y)
      IMPLICIT INTEGER*4 (A-Z)
      LOGICAL*1 A(N)
      COMMON /CPLOT$/ G1(12),INTENS,G2(4)
C SET INCREMENTS
      DX=8
      DY=0
      GO TO 10
C VERLIN ENTRY
      ENTRY VERLIN (A,N,X,Y)
C SET INCREMENTS
      DX=0
      DY=-16
C DIAGLN ENTRY
      ENTRY DIAGLN (A,N,X,Y,DX,DY)
C SET INITIAL COORDINATES AND OP CODE
 10   IX=X-(N-1)*DX/2
      IY=Y-(N-1)*DY/2
      OP=5
      IF (INTENS.NE.0) OP=6
C PLOT EACH CHARACTER
      DO 20 I=1,N
      CALL SC4020 (OP,IX,IY,A(I),INTENS)
      IX=IX+DX
 20   IY=IY+DY
      RETURN
      END
```

MINT

DESCRIPTION

MINT is a function routine which determines the value of the largest integer which is less than or equal to the value of a floating point number, X, which has been input. Notice that -MINT (-X) can be used to find the smallest integer greater than or equal to X.

NAME		MINT
PURPOSE		TO TRUNCATE TO THE NEXT ALGEBRAICALLY SMALLER INTEGER
CALLING SEQUENCE		MINT(X)
SYMBOL	TYPE	DESCRIPTION
X	R	INPUT - VALUE TO BE TRUNCATED
MINT	I	OUTPUT - LARGEST INTEGER LESS THAN X (-MINT(-X) TRUNCATES TO THE NEXT ALGEBRAICALLY GREATER INTEGER)
SUBROUTINES USED		NONE
COMMON BLOCKS		NONE
INPUT FILES		NONE
OUTPUT FILES		NONE
RESTRICTIONS		NONE
REFERENCES		NONE

```
      FUNCTION MINT(X)
      MINT=X
      IF (FLOAT(MINT).GT.X) MINT=MINT-1
      RETURN
      END
```

PLOTST

DESCRIPTION

PLOTST is used to initialize the plot package (and produce a leading ID frame) or to terminate the plot package (and produce a trailing ID frame).

PLOTST sets the object space to default values and then calls SC4020 to initialize. If an ID frame is desired, ENTRY IDFRME is used.

IDFRME uses descriptions in data statements to produce an ID frame. DATE is called to put the date on the ID frame.

ENDPLT terminates the plot package by calling SC4020 to terminate (emptying its plot buffers.) Then IDFRME is used to produce the trailing ID frame.

NAME PLOTST

ENTRY POINT	PURPOSE
PLOTST	TO INITIALIZE THE PLOT PACKAGE AND TO SELECT OUTPUT DEVICES
IDFRME	TO GENERATE THE IDENTIFICATION FRAME FOR THE PLOT PACKAGE
ENDPLT	TO TERMINATE THE PLOT PACKAGE

CALLING SEQUENCE CALL PLOTST(N,ID)

SYMBOL	TYPE	DESCRIPTION
N	I	INPUT - SUM OF DEVICE NUMBERS DESIRED SUCH THAT =1 DESIGNATES THE 35 MM CAMERA =2 DESIGNATES THE 9 INCH CAMERA =3 DESIGNATES THE PRINTER
ID	L	INPUT - TRUE FOR ID FRAME DESIRED

CALLING SEQUENCE CALL IDFRME

CALLING SEQUENCE CALL ENDPLT

SUBROUTINES USED	SC4020	HORLIN	DATE	EMPTY
COMMON BLOCK	CPLOT$			
INPUT FILES	NONE			
OUTPUT FILES	NONE			
RESTRICTIONS	NONE			
REFERENCES	NONE			

```
      SUBROUTINE PLOTST (N,ID)
      LOGICAL SWITCH,ID
      LOGICAL PRINT,PLOTER,LOGX,LOGY
      INTEGER PLOT1$
      COMMON /CPLOT$/ PRINT,PLOTER,LOGX,LOGY,XLOLIM,YLOLIM,
     • XHILIM,YHILIM,SCALEX,SCALEY,SXLO,SYLO,IT,IPRNT,PLOT1$,LININC,
     • LINECT
      DIMENSION DAT(2)
      DIMENSION A(8),B(2)
      DATA A/32HAEDC PLOT PACKAGE FOR IBM 360    /
      DATA B /8HRUN ON     /
      DATA IZ /0/
C DEFAULT GRID LIMITS AND SCALE FACTORS
```

```
      LOGX=.FALSE.
      LOGY=.FALSE.
      XLOLIM=32.
      YLOLIM=32.
      XHILIM=992.
      YHILIM=992.
      SCALEX=1.
      SCALEY=1.
      SXLO=0.
      SYLO=0.
      IT=0
      IPRNT=6
      PLOTIS=20
      LININC=16
      LINECT=0
C SELECT DEVICES
      M=4-N
      IF(M.LT.-3)M=-3
      CALL SC4020(M,D,D,D,D)
 30   IF (.NOT.ID) RETURN
C IDFRME ENTRY
      ENTRY IDFRME
      SWITCH=.TRUE.
      GO TO 40
C ENDPLT ENTRY
      ENTRY ENDPLT
      CALL SC4020 (17,D,D,D,D)
      SWITCH=.FALSE.
C DRAW LARGE SQUARE
 40   CALL SC4020 (10,0,0,1023,D)
      CALL SC4020 (10,1023,0,1023,D)
      CALL SC4020 (9,0,0,1023,D)
      CALL SC4020 (9,0,1023,1023,D)
C DRAW SMALLER SQUARE INSIDE
      CALL SC4020 (10,255,256,767,D)
      CALL SC4020 (10,768,256,767,D)
      CALL SC4020 (9,255,256,768,D)
      CALL SC4020 (9,255,767,768,D)
C DRAW RHOMBUS
      DO 50 I=12,511,64
      CALL SC4020 (11,I,512+I,63,63)
      CALL SC4020 (11,I,511-I,63,-63)
      CALL SC4020 (11,1023-I,512+I,-63, 63)
      CALL SC4020 (11,1023-I,511-I,-63,-63)
 50   CONTINUE
C INSERT TITLE AND DATE
      CALL HERLIN (A,31,512,750)
      CALL HERLIN (B,6,512,520)
      CALL DATE (DAT)
      CALL HERLIN (DAT,8,512,508)
      IF (SWITCH) RETURN
C EMPTY BUFFERS AND TERMINATE PLOTTER OUTPUT
      CALL SC4020 (17,D,D,D,D)
      IF (PLOTER) CALL EMPTY
      RETURN
      END
```

QUICKY

DESCRIPTION

QUICKY is a quick plot routine. The user inputs an array of coordinates and QUICKY outputs a plot of his data complete with grid overlay.

QUICKY first calls GRDNUM for the x array to determine x characteristics for the grid overlay. Then GRDNUM is called for the y array. Then GRID is called to output the grid overlay and finally PLOT is called to plot his arrays.

NAME — QUICKY

PURPOSE — TO PLOT X-Y VALUES ON AN APPROPRIATE GRID

CALLING SEQUENCE — CALL QUICKY(X,Y,N,CHAR)

SYMBOL	TYPE	DESCRIPTION
X	R	INPUT - ARRAY OF ABSCISSA VALUES TO BE PLOTTED
Y	R	INPUT - ARRAY OF ORDINATE VALUES TO BE PLOTTED
N	I	INPUT - NUMBER OF COORDINATES IN THE X-Y ARRAYS
CHAR	A	INPUT - RIGHT JUSTIFIED CHARACTER TO BE PLOTTED. IF CHARACTER IS BLANK, VECTORS WILL BE PLOTTED BETWEEN POINTS

SUBROUTINES USED — GRDNUM GRID PLOT HORLIN

COMMON BLOCKS — NONE

INPUT FILES — NONE

OUTPUT FILES — NONE

RESTRICTIONS — N MUST BE GREATER THAN 1 AND NEITHER THE X NOR THE Y ARRAYS MAY HAVE ALL ELEMENTS EQUAL

REFERENCES — NONE

```
      SUBROUTINE QUICKY(X,Y,N,CHAR)
      DIMENSION X(N),Y(N)
      LOGICAL*1 XFMT(7),YFMT(7)
C GET ESTHETIC GRID LIMITS AND FORMATS
      CALL GRDNUM(X,N,XMIN,XMAX,NX,XFMT)
      CALL GRDNUM(Y,N,YMIN,YMAX,NY,YFMT)
      IF(NX.EQ.0.OR.NY.EQ.0) GO TO 10
C DRAW GRID
      CALL GRID(XMIN,XMAX,NX,XFMT,1,YMIN,YMAX,NY,YFMT,1,0)
C PLOT POINTS
      CALL PLOT(X,Y,N,CHAR)
      RETURN
 10   CALL HORLIN('EMPTY ARRAY OR ALL ITEMS EQUAL IN QUICKY',40,512,512)
      RETURN
      END
```

SC4020

DESCRIPTION

SC4020 is the basic routine which formats the SC4020 instruction. It also simulates the SC4020 by outputing printer plots.

There are two major sections: the printer and the SC4020. The first parameter to the SC4020 call is a operation indicator. If the printer has been selected, then, through a computed GO TO, the operation indicator causes the operation to be done. The same happens for the SC4020 if it has been selected.

The printer section consists of the coding which puts characters into the print buffer. The SC4020 section consists of set up the SC4020 instructions and storing these into a buffer which is output when it is filled.

Special entries in SC4020 are equivalent to calling SC4020 with certain operation indicators. These include FRMADV (frame advance) and EMPTY (empty the buffers). Other entries include NWUNIT (to specify the output units), FRAMES (to return the number of frames produced) and VBAR (to substitute a vertical bar instead of an "I" for vertical plotting on the printer.)

NAME	SC4020
ENTRY POINT	PURPOSE
SC4020	TO TRANSLATE PLOT COMMANDS INTO SC4020 INSTRUCTIONS AND/OR PRINTER PLOTS
FRMADV	TO ADVANCE THE FRAME
NWUNIT	TO SET THE OUTPUT UNIT NUMBERS
FRAMES	TO RETURN A COUNT OF THE NUMBER OF FRAMES PRODUCED
EMPTY	TO TERMINATE THE PLOTTER TAPE OUTPUT
VBAR	TO USE THE VERTICAL BAR CHARACTER "|" INSTEAD OF "I" FOR VERTICAL LINES OF THE PRINTER PLOTS
CONDNS	TO SET A FRAME OF PRINTER AS ONE COMPUTER PAGE INSTEAD OF THE NORMAL TWO

CALLING SEQUENCES   BECAUSE EACH OF THE 15 PLOT COMMANDS USES THE ARGUMENT LIST DIFFERENTLY, EACH CALLING SEQUENCE IS LISTED. IN EACH CASE THE ARGUMENT ''D'' IS A DUMMY ARGUMENT, AND THE FIRST ARGUMENT IS THE FUNCTION CODE, ''OP''.

	SYMBOL	TYPE	DESCRIPTION
CALL SC4020(-3,D,D,D,D)	-3	I	COMMAND TO SELECT BOTH CAMERAS AND PRINTER
CALL SC4020(-2,D,D,D,D)	-2	I	COMMAND TO SELECT CAMERA 2 AND PRINTER
CALL SC4020(-1,D,D,D,D)	-1	I	COMMAND TO SELECT CAMERA 1 AND PRINTER
CALL SC4020(0,D,D,D,D)	0	I	COMMAND TO SELECT PRINTER
CALL SC4020(1,D,D,D,D)	1	I	COMMAND TO SELECT CAMERA 1
CALL SC4020(2,D,D,D,D)	2	I	COMMAND TO SELECT CAMERA 2

CALL SC4020(3,D,D,D,D)	3	I	COMMAND TO SELECT BOTH CAMERAS
CALL SC4020(4,D,D,D,D)	4	I	COMMAND TO ADVANCE FILM
CALL SC4020(5,X,Y,CHAR,D)	5	I	COMMAND TO PLOT SINGLE CHARACTER AT COORDINATES X, Y AT CURRENT LIGHT INTENSITY
	X	I	RASTER COUNT OF X COORDINATE
	Y	I	RASTER COUNT OF Y COORDINATE
	CHAR(1)	L*1	CHARACTER TO BE PLOTTED
CALL SC4020(6,X,Y,CHAR,B)	6	I	COMMAND TO PLOT SINGLE CHARACTER AT COORDINATES X,Y WITH B LIGHT INTENSITY
	X	I	RASTER COUNT OF X COORDINATE
	Y	I	RASTER COUNT OF Y COORDINATE
	CHAR(1)	L*1	CHARACTER TO BE PLOTTED
	B	I	LIGHT INTENSITY (0-15)
CALL SC4020(7,X,Y,CHAR,D)	7	I	COMMAND TO PLOT SINGLE CHARACTER AT COORDINATES X, Y AND SET LIGHT INTENSITY TO BRIGHT
	X	I	RASTER COUNT OF X COORDINATE
	Y	I	RASTER COUNT OF Y COORDINATE
	CHAR(1)	L*1	CHARACTER TO BE PLOTTED
CALL SC4020(8,X,Y,CHAR,D)	8	I	COMMAND TO PLOT

			SINGLE CHARACTER AT COORDINATES X, Y AND SET LIGHT INTENSITY TO DIM
	X	I	RASTER COUNT OF X COORDINATE
	Y	I	RASTER COUNT OF Y COORDINATE
	CHAR(1)	L*1	CHARACTER TO BE PLOTTED
CALL SC4020(9,X,Y,STOP,D)	9	I	COMMAND TO GENERATE LINE FROM COORDINATES (X,Y) TO (STOP,Y)
	X	I	RASTER COUNT OF X COORDINATE OF STARTING POINT
	Y	I	RASTER COUNT OF Y COORDINATE OF LINE
	STOP	I	RASTER COUNT OF X COORDINATE OF END POINT
CALL SC4020(10,X,Y,STOP,D)	10	I	COMMAND TO GENERATE LINE FROM COORDINATES (X,Y) TO (X,STOP)
	X	I	RASTER COUNT OF X COORDINATE OF LINE
	Y	I	RASTER COUNT OF Y COORDINATE OF STARTING POINT
	STOP	I	RASTER COUNT OF Y COORDINATE OF END POINT
CALL SC4020(11,X,Y,XV,YV)	11	I	COMMAND TO GENERATE LINE FROM COORDINATES (X,Y) TO (X+XV, Y+YV)
	X	I	RASTER COUNT OF X COORDINATE

	Y	I	RASTER COUNT OF Y COORDINATE
	XV	I	SIGNED RASTER COUNT OF X COMPONENT OF THE VECTOR TO BE PLOTTED
	YV	I	SIGNED RASTER COUNT OF Y COMPONENT O F THE VECTOR TO BE PLOTTED
CALL SC4020(12,D,D,D,D)	12	I	COMMAND TO FORCE PLOTTER SCREEN SQUARE FOR NORMAL PLOTTING (REDUCE IMAGE)
CALL SC4020(13,D,D,D,D)	13	I	COMMAND TO FORCE PLOTTER SCREEN RECTANGULAR FOR CONTINUING PLOT ON NEXT FRAME (EXPAND IMAGE)
CALL SC4020(14,D,D,D,D)	14	I	COMMAND TO PROJECT PREPARED SLIDE ONTO CAMERA (REQUIRES SPEC-IALLY PREAPRED SLIDE)
CALL SC4020(15,X,Y,CHAR,N)	15	I	COMMAND TO BEGIN TYPEWRITER MODE AND TYPE CHARACTERS STARTING FROM (X,Y)
	X	I	RASTER COUNT OF X COORDINATE OF STARTING POINT
	Y	I	RASTER COUNT OF Y COORDINATE OF STARTING POINT
	CHAR(N)	L*1	CHARACTERS TO BE PLOTTED
	N	I	NUMBER OF CHARACTERS TO

			BE PLOTTED
CALL SC4020(16,D,D,CHAR,N)	16	I	COMMAND TO BEGIN TYPEWRITER MODE AND TYPE CHARACTERS STARTING AT THE BEGINNING OF THE LAST VECTOR PLOTTED OR LAST POINT PLOTTED
	CHAR(N)	L*1	CHARACTERS TO BE PLOTTED
	N	I	NUMBER OF CHARACTERS TO BE PLOTTED
CALL SC4020(17,D,D,D,D)	17	I	COMMAND TO ADVANCE FILM,SET LIGHT INTENSITY TO BRIGHT,AND END TYPEWRITER MODE
CALL SC4020(18,D,D,CHAR,N)	18	I	COMMAND FOR CARRIAGE RETURN AND TYPE CHARACTERS STARTING ON NEXT LINE
	CHAR(N)	L*1	CHARACTERS TO BE PLOTTED
	N	I	NUMBER OF CHARACTERS TO BE PLOTTED
CALL SC4020(19,D,D,CHAR,N)	19	I	COMMAND TO CONTINUE TYPEWRITER MODE ADDING CHARACTERS AFTER LAST CHARACTER TYPED
	CHAR(N)	L*1	CHARACTERS TO BE PLOTTED
	N	I	NUMBER OF CHARACTERS TO BE PLOTTED
CALL SC4020(20,D,D,D,D)	20	I	COMMAND TO STOP TYPEWRITER

MODE AND RETURN
TO NORMAL MODE

CALLING SEQUENCE CALL FRMADV

CALLING SEQUENCE CALL NWUNIT(IPRNT,IPLOTR)

SYMBOL	TYPE	DESCRIPTION
IPRNT	I	INPUT - FORTRAN LOGICAL UNIT NUMBER FOR PRINTER PLOTS
IPLOTR	I	INPUT - FORTRAN LOGICAL UNIT NUMBER FOR PLOTTER DRIVE TAPE

CALLING SEQUENCE CALL FRAMES(FRMCNT)

SYMBOL	TYPE	DESCRIPTION
FRMCNT	I	OUTPUT - NUMBER OF FRAMES PRODUCED

CALLING SEQUENCE CALL EMPTY

CALLING SEQUENCE CALL VBAR

CALLING SEQUENCE CALL CONDNS

SUBROUTINE USED SCHAR

COMMON BLOCK CPLOT$

INPUT FILES NONE

OUTPUT FILES PRNT - FORTRAN LOGICAL UNIT NUMBER FOR PRINTER PLOTS
PLOT$ - FORTRAN LOGICAL UNIT FOR SC4020 PLOTS

RESTRICTIONS NONE

REFERENCES NONE

```
      SUBROUTINE SC4020 (OP,X,Y,INCHAR,YV)                              SC40 320
      COMMON /CPLOT$/PRINT,PLOTER,G1(11),PRNT,PLOT1$,LININC,LINECT      SC40 321
      INTEGER SCHAR                                                     SC40 322
      INTEGER OP,OP1,X,Y,XV,YV,PRNT,PLOT1$,FRMCNT,                      SC40 323
     *SHIFT2,SHIFT4,SHIFT5,SHIFT6,SHIFT8                                SC40 324
      INTEGER CARETN,STOPTP,RESET                                       SC40 325
      INTEGER*2 SETPOS,IZAP                                             SC40 326
      LOGICAL PRINT,PLOTER,INITAL,TERR,PERR,TYPING,TYPMOD               SC40 327
      LOGICAL*1 OPCODE(25),ERROR(12),OUT(4092),PBUF(128,128)            SC40 328
      LOGICAL*1 II,MINUS,DOT,BLANK,SLASH,PT,BAR                         SC40 329
      LOGICAL*1 FORMAT(30)/'(2H1 ,64(128A1/2H  ,128A1/2X))'/,           SC40 330
     *PLUS/'+'/                                                         SC40 331
```

```
      LOGICAL*1 LX,DUM(4),CHAR(1),XCHAR(256),INCHAR(YV)
C LX IS EQUIVALENCED TO THE LOW ORDER BYTE IW.
C THIS IS FOR BYTE MANIPULATION PURPOSES.
      EQUIVALENCE (IW,DUM(1)),(DUM(4),LX)
      EQUIVALENCE (XCHAR(2),CHAR(1))
      EQUIVALENCE(SETPOS,IS)
C INITIALIZE SWITCHES
      DATA INITAL,PERR,TERR,TYPING /4*F/
C SC-4020 OP CODES
      DATA CFCODE/ Z21,Z22,Z23,Z26,Z00,Z01,Z02,Z04,
     * Z18,Z1A,Z30,Z25,Z24,Z28,Z10,Z12,Z2E,Z2A,Z00,Z0A  /
      DATA CARETN,STOPTP,RESET /Z0000002A,Z000000CA,Z0000002E /
C INSERTS ERROR SLASHES IN UPPER RIGHT CORNER OF FRAME
      DATA ERROR / Z10,Z0F,Z0F,Z31,Z00,Z08,Z31,Z31,Z31,Z31,Z31,Z0A /
C PLOT CHARACTERS FOR PRINTER PLOTS
      DATA II,MINUS,DOT,BLANK,SLASH,BAR/1HI,1H-,1H.,1H ,1H/,1H|/
C CONSTANTS FOR BIT MANIPULATION
      DATA SHIFT2,SHIFT4,SHIFT5,SHIFT6,SHIFT8   /
     * Z00000004, Z00000010, Z00000020, Z00000040, Z00000100   /
C MISC. CONSTANTS
      DATA IZ,IZAP /0,Z0000 /
      DATA IBLANK /Z00000040 /
C NORMAL ENTRY
      OP1=IABS(OP)
C TEST FOR DEVICE SELECTION OP CODE
      IF (OP1.GT.3) GO TO 1
C SET DEVICE SWITCHES
      PRINT=OP.LE.0
      PLOTER=OP1.GT.0
C RETURN IF PRINTER SELECT ONLY
      IF(OP1.EQ.0) RETURN
      GO TO 1
C FRMADV ENTRY
      ENTRY FRMADV
C SET OP CODE FOR RESET
      OP1=17
C TEST FOR INITIALIZATION
 1    IF (INITAL) GO TO 20
      INITAL=.TRUE.
C ZERO FRAME COUNT AND COMMAND BUFFER INDEX
      IFRM=0
      ICOUNT=0
C SET UP CHARACTER TRANSLATION MATRIX
      IW=0
      DO 5 I=IZ,255
 5    CHAR(I)=LX
      DO 10 I=IZ,63
      IW=I
      KH=SCHAR(I)
 10   CHAR(KH)=LX
C SET PRINT BUFFER TO BLANKS
      DO 15 I=1,128
      DO 15 J=1,128
 15   PBUF(I,J)=BLANK
C ZERO CURRENT POINT REGISTER VALUES
      RX=0.
```

```
         RY=0.
C TEST FOR INPUT COMMAND IN WRONG MODE
 20      IF ((TYPING.AND.OP1.GE.17).OR.(.NOT.TYPING.AND.OP1.LE.17))
     .   GO TO 25
C SET PRINTER AND PLOTTER ERROR FLAGS
         PERR=.TRUE.
         TERR=.TRUE.
C RETURN IF NOT FRAME ADVANCE
         IF (OP1.NE.4) RETURN
C SET CP CODE FOR RESET
         OP1=17
C SET NEW PROGRAM MODE
 25      TYPMOD=TYPING
C BEGIN PROCESSING
C TRANSFER ON CP CODE
         GO TO (45,45,45,30,45,45,45,45,33,35,35,45,45,45,
     .   43,43,30,45,45,30),OP1
C SET FOR PLOTTING MODE
 30      TYPMOD=.FALSE.
         GO TO 45
C RECOVER X VECTOR COMPONENT OR AXIS END POINT
 35      DO 40 I=1,4
 40      DUM(I)=INCHAR(I)
         XV=IW
         GO TO 45
C SET FOR TYPEWRITER MODE
 43      TYPMOD=.TRUE.
 45      IOP=OP1
C TEST FOR PRINTER PLOTS
         IF (.NOT.PRINT) GO TO 150
C TRANSFER ON CP CODE
         GO TO (150,150,150,125,65,65,65,65,55,60,50,150,150,150,
     .   90,95,125,85,100,145),OP1
C SET PLOT CHARACTER FOR VECTOR
 50      PT=DOT
C CALCULATE MAXIMUM DEFLECTION MAGNITUDE
         V=AMAX0(IABS(XV),IABS(YV))
C TEST FOR DEFLECTION OUT OF RANGE
         IF (V.LT.64.) GO TO 53
         PERR=.TRUE.
         GO TO 150
C COMPUTE PRINTER VECTOR COMPONENTS
 53      N=V/8.
         IF (N.EQ.0) GO TO 70
         RXV=FLOAT(XV)/V
         RYV=FLOAT(YV)/V
         GO TO 70
C SET X AXIS CHARACTER
 55      PT=MINUS
C COMPUTE PRINTER VECTOR COMPONENTS
         N=(XV-X)/8
         RYV=0.
         RXV=1.
         GO TO 70
C SET Y AXIS CHARACTER
 60      PT=II
```

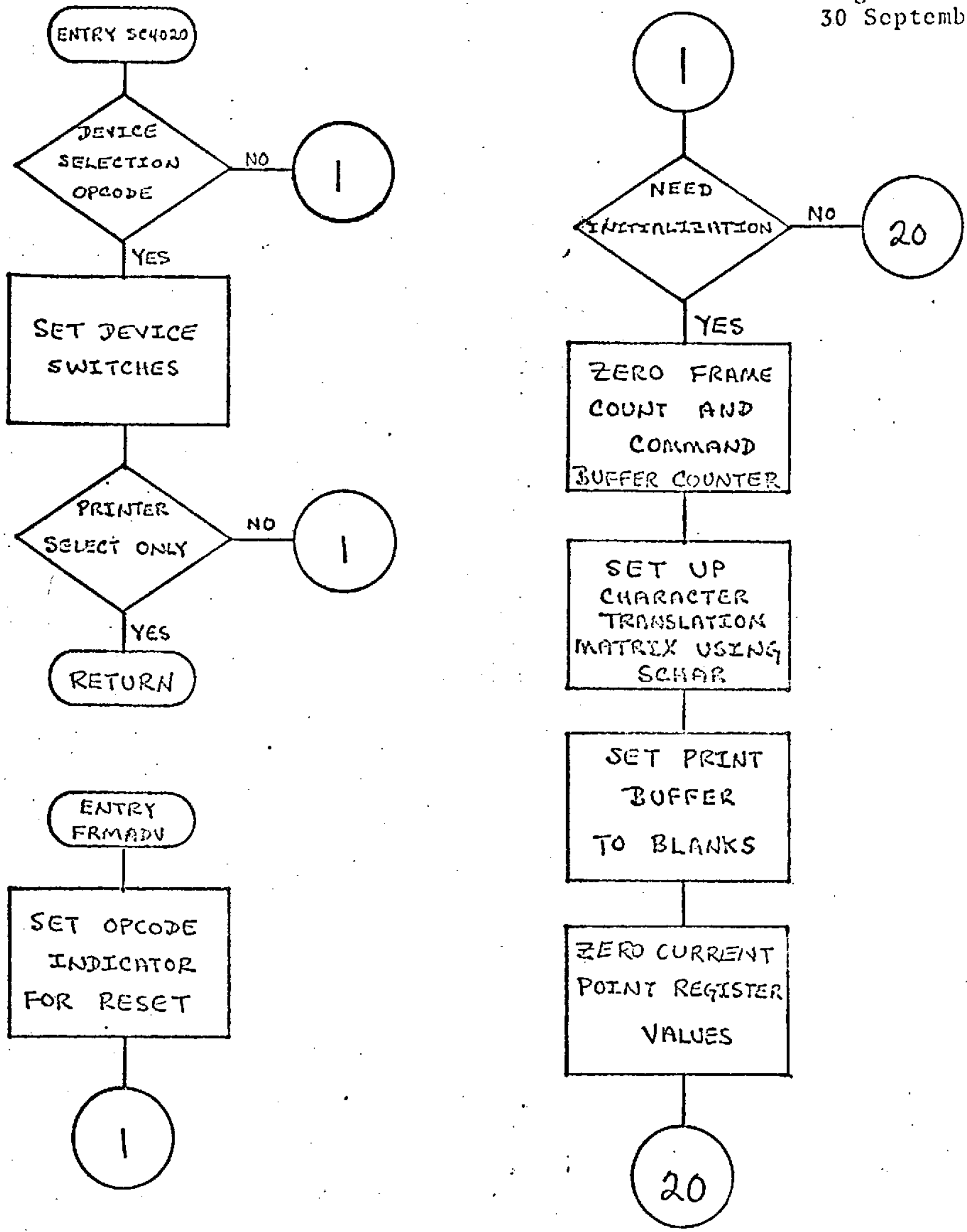
```
C COMPUTE PRINTER VECTOR COMPONENTS
        N=(XV-Y)/8
        RYV=1.
        RXV=0.
        GO TO 70
C SET PLOT CHARACTER TO INPUT
 65     PT=INCHAR(1)
        N=0
C COMPUTE PRINTER DEFLECTIONS FOR ORIGIN
 70     RX=FLOAT(X)/8.
        RY=FLOAT(Y)/8.
C LOOP TO PLOT ALL CHARACTERS IN LINE
        DO 80 I=IZ,N
C COMPUTE PRINTER DEFLECTIONS FOR EACH POINT
        R=I
        IX=RX+R*RXV
        IY=128-INT(RY+R*RYV)
C TEST FOR DEFLECTIONS IN RANGE
        IF (IX.GE.0.AND.IX.LE.127.AND.IY.GE.1.AND.IY.LE.128) GO TO 75
        PERR=.TRUE.
        GO TO 80
C INSERT CHARACTER IN BUFFER
 75     PBUF(IX+1,IY)=PT
 80     CONTINUE
        GO TO 150
C SET CURRENT POINT INDEX TO BEGINNING OF LINE
 85     INDEX=(INDEX+127)/128*128
        GO TO 100
C SET CURRENT POINT INDEX FROM GIVEN POINT
 90     INDEX=128*(127-Y/8)+X/8
        GO TO 105
C SET CURRENT POINT INDEX FROM CURRENT POINT REGISTERS
 95     INDEX=128*(127-INT(RY))+INT(RX)
C TEST FOR CHARACTERS TO PLOT
 100    IF (YV.LT.1) GO TO 150
 105    IW=0
C LOOP TO PROCESS ALL CHARACTERS
        DO 120 I=1,YV
        LX=INCHAR(I)
C TEST FOR CARRIAGE RETURN
        IF (IW.NE.CARETN) GO TO 110
C SET CURRENT POINT INDEX
        INDEX=(INDEX+127)/128*128
        GO TO 120
C TEST FOR STOP TYPE OR RESET COMMAND
 110    IF (IW.NE.STOPTP.AND.IW.NE.RESET) GO TO 115
C MUST BE LAST CHARACTER IN STRING
        IF (I.NE.YV) GO TO 113
C SET MODE SWITCH
        TYPMOD=.FALSE.
C TEST FOR STOP TYPE COMMAND
        IF (IW.EQ.STOPTP) GO TO 145
C SET OP CODE FOR RESET
        OP1=17
        GO TO 125
C SET ERROR INDICATOR
```

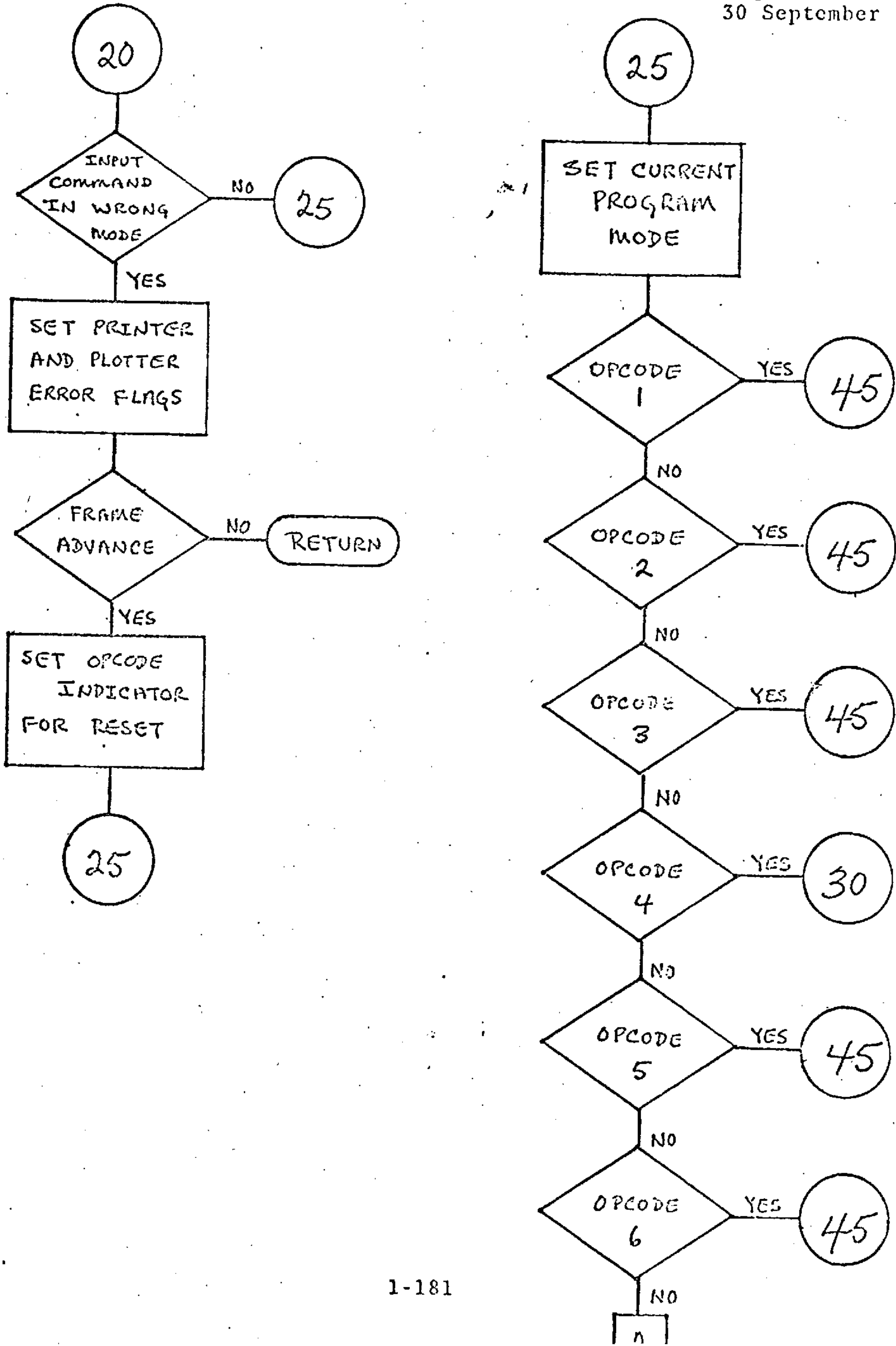
```
 113      PERR=.TRUE.
          GO TO 150
C INCREMENT CURRENT POINT INDEX
 115      INDEX=MOD(INDEX,16384)+1
C INSERT CHARACTER IN BUFFER
          PBUF(INDEX,1)=LX
 120      CONTINUE
          GO TO 150
C TEST FOR ERRORS ON THIS FRAME
 125      IF (.NOT.PERR) GO TO 135
C INSERT ERROR SLASHES
          DO 130 I=121,128
 130      PBUF(I,1)=SLASH
C RESET ERROR SWITCH
          PERR=.FALSE.
C OUTPUT PRINT BUFFER
 135      WRITE(PRNT,FORMAT) PBUF
C INITIALIZE PRINT BUFFER
          DO 140 I=1,128
          DO 140 J=1,128
 140      PBUF(I,J)=BLANK
C TEST FOR RESET COMMAND
          IF (OP1.NE.17) GO TO 145
C ZERO CURRENT POINT REGISTER VALUES
          RX=0.
          RY=0.
          GO TO 150
C SET CURRENT POINT REGISTER VALUES
 145      RX=MOD(INDEX,128)
          RY=128-INDEX/128
C TEST FOR SC4020 PLOTS
 150      IF(.NOT.PLOTER) GO TO 300
C INSERT OP CODE IN COMMAND BUFFER
          OP1=ICP
 155      OUT(ICOUNT+1)=OPCODE(OP1)
          IW=0
C TRANSFER ON OP CODE
          GO TO (285,285,285,255,185,160,185,185,165,170,160,285,285,
     .    285,185,200,250,200,205,275),OP1
C COMPUTE VECTOR COMPONENTS
 160      IXV=IABS(XV)
          IYV=IABS(YV)
C TEST FOR COMPONENTS OUT OF RANGE
          IF (IXV.GT.63.OR.IYV.GT.63) GO TO 350
C INSERT LEADING VECTOR BITS IN COMMAND
          LX=OUT(ICOUNT+1)
          IW=IW+IXV/SHIFT2
          OUT(ICOUNT+1)=LX
C SET VECTOR BIT CONSTANTS
          IW=IYV/SHIFT2
          IF (YV.GT.0) IW=IW+SHIFT4
          IF (XV.GT.0) IW=IW+SHIFT5
          OUT(ICOUNT+4)=LX
          IWY=MOD(IYV,SHIFT2)*SHIFT4
          IWX=MOD(IXV,SHIFT2)*SHIFT4
          GO TO 195
```

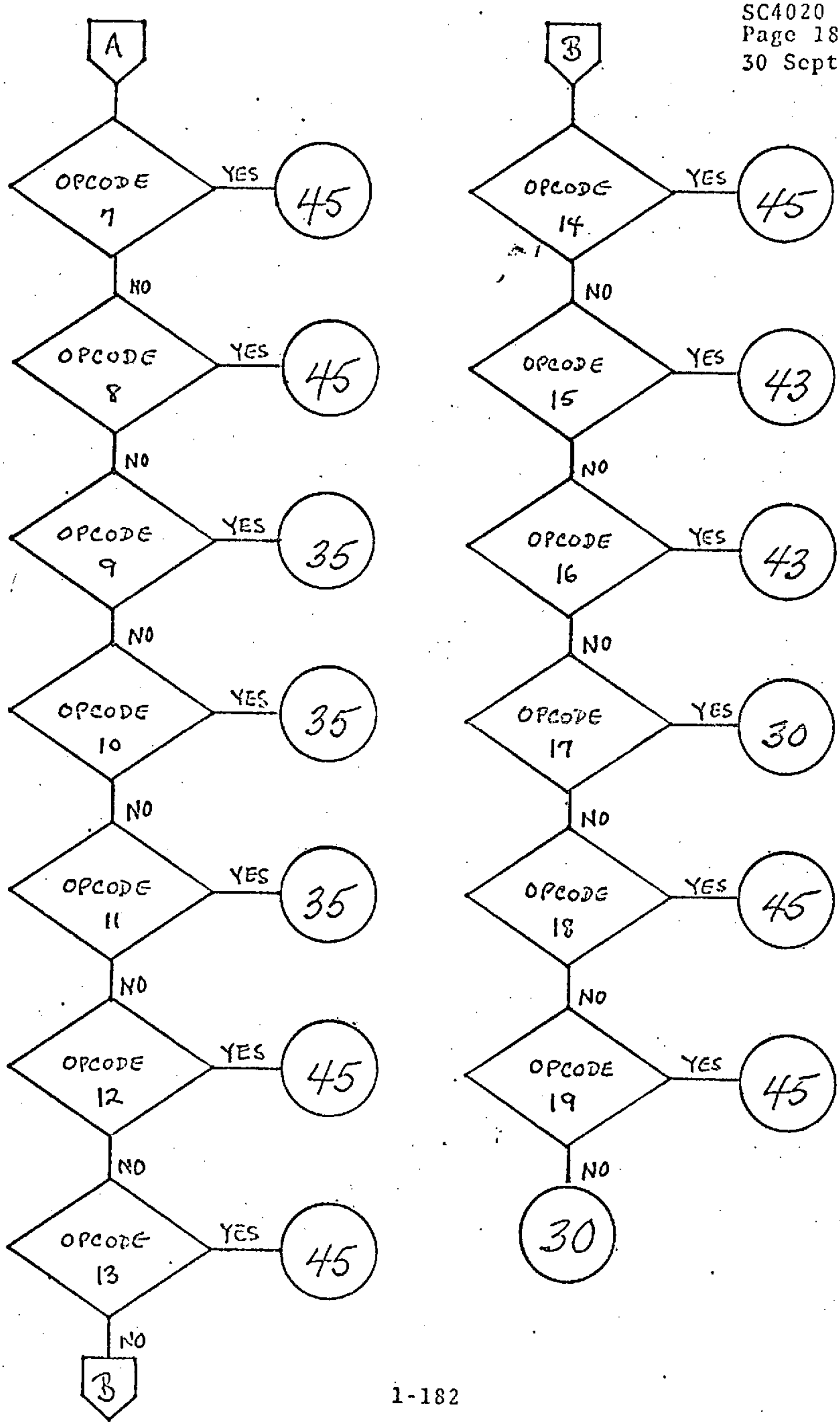
```
C SET X STOP CODE
 165      IS=-1-XV
          SETPOS=IZAP
          GO TO 175
C SET Y STOP CODE
 170      IS=1023-XV
C TEST FOR STOP CODE OUT OF RANGE
 175      IF (XV.GT.1023.OR.XV.LT.0) GO TO 350
C SET DEFLECTION BIT CONSTANTS
          IWY=MOD(IS,SHIFT2)*SHIFT4
          IWX=(IS/SHIFT8)*SHIFT4
          IW=MOD(IS/SHIFT2,SHIFT6)
          OUT(ICOUNT+4)=LX
          GO TO 195
C SET INTENSITY BIT CONSTANTS
 180      IWX=YV/SHIFT2*SHIFT4
          IWY=MOD(YV,SHIFT2)*SHIFT4
          IW=0
C STORE CHARACTER IN COMMAND BUFFER
          LX=INCHAR(1)
          OUT(ICOUNT+4)=CHAR(IW)
          IF (IW.EQ.IBLANK) OUT(ICOUNT+1)=OPCODE(5)
          GO TO 195
C SET BIT CONSTANTS TO ZEROES
 185      IWX=0
          IWY=0
C STORE CHARACTER IN COMMAND BUFFER
 190      IW=0
          LX=INCHAR(1)
          OUT(ICOUNT+4)=CHAR(IW)
C TEST FOR DEFLECTIONS OUT OF RANGE
 195      IF (X.GT.1023.OR.X.LT.0.OR.Y.GT.1023.OR.Y.LT.0) GO TO 350
C INSERT BIT CONSTANTS AND DEFLECTIONS IN COMMAND BUFFER
          IW=IWX+X/SHIFT6
          OUT(ICOUNT+2)=LX
          IW=X
          OUT(ICOUNT+3)=LX
          IS=1023-Y
          IW=IWY+IS/SHIFT6
          OUT(ICOUNT+5)=LX
          IW=IS
          OUT(ICOUNT+6)=LX
C TEST FOR TYPE SPECIFIED POINT OPERATION
          IF (OP1.NE.15) GO TO 285
C INCREMENT COMMAND BUFFER COUNTER
          ICOUNT=ICOUNT+6
C SET TO BEGIN WITH SECOND CHARACTER
          IN=2
          GO TO 210
C CORRECT BUFFER COUNT
 200      ICOUNT=ICOUNT+1
C START ON FIRST CHARACTER
 205      IN=1
C TEST FOR CHARACTERS TO ADD TO BUFFER
 210      IF (IN.GT.YV) GO TO 290
          IW=0
```

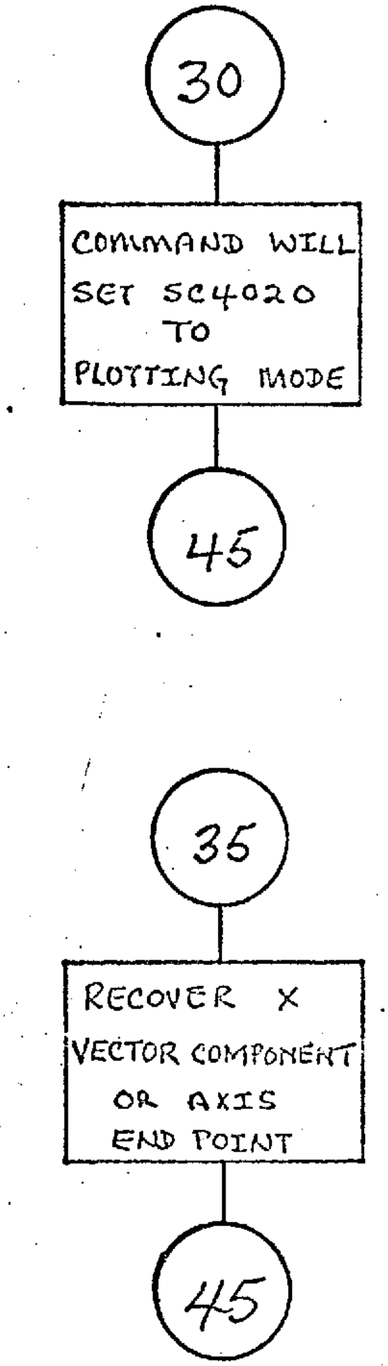
```
C LOOP TO PROCESS ALL CHARACTERS
      DO 230 J=IN,YV
      IF (ICOUNT.LT.4092) GO TO 215
C OUTPUT BUFFER IF NECESSARY
      WRITE (PLOT1$,2000) (OUT(I),I=1,ICOUNT)
      ICOUNT=0
  215 LX=INCHAR(J)
C TEST FOR STOP TYPE OR RESET OPERATION
      IF (IW.NE.STOPTP.AND.IW.NE.RESET) GO TO 220
C MUST BE LAST CHARACTER IN STRING
      IF (J.NE.YV) GO TO 350
C INSERT COMMAND IN BUFFER AND SET MODE SWITCH
      OUT(ICOUNT+1)=LX
      TYPMOD=.FALSE.
      GO TO 240
  220 LX=CHAR(IW)
C INSERT CHARACTER IN BUFFER
      ICOUNT=ICOUNT+1
  230 OUT(ICOUNT)=LX
C TEST TO SEE IF STILL IN TYPEWRITER MODE
  240 IF (TYPMOD) GO TO 290
C TEST TO SEE IF LAST CHARACTER WAS STOP TYPE
      IF (IW.EQ.STOPTP) GO TO 275
C SET OP CODE FOR RESET
      OP1=17
C TEST FOR ERRORS ON THIS FRAME
  250 IF (.NOT.TERR) GO TO 275
C TEST TO SEE IF IN TYPEWRITER MODE
      IF (.NOT.TYPING) GO TO 260
C INSERT STOP CODE COMMAND IN BUFFER AND ADJUST COUNT
      OUT(ICOUNT+1)=OPCODE(20)
      ICOUNT=(ICOUNT+6)/6*6
      GO TO 260
C TEST FOR ERRORS ON THIS FRAME
  255 IF (.NOT.TERR) GO TO 285
C RESET ERROR SWITCH
  260 TERR=.FALSE.
C TEST FOR ROOM IN BUFFER
      IF (ICOUNT.LT.4077) GO TO 265
C OUTPUT BUFFER,ERROR MARK CODES,AND RESET OR FRAME ADVANCE
      WRITE (PLOT1$,2000) (OUT(I),I=1,ICOUNT),ERROR,OPCODE(OP1)
      GO TO 295
C INSERT ERROR CODES IN BUFFER
  265 DO 270 I=1,12
  270 OUT(ICOUNT+I)=ERROR(I)
      OUT(ICOUNT+13)=OPCODE(OP1)
      ICOUNT=ICOUNT+18
      GO TO 290
C ROUND BUFFER COUNT TO AN EVEN COMMAND SIZE
  275 ICOUNT=(ICOUNT+6)/6*6
      GO TO 290
C INCREMENT BUFFER COUNT BY A FULL COMMAND SIZE
  285 ICOUNT=ICOUNT+6
C OUTPUT BUFFER IF FULL
  290 IF (ICOUNT.LT.4092) GO TO 300
      WRITE (PLOT1$,2000) (OUT(I),I=1,ICOUNT)
```

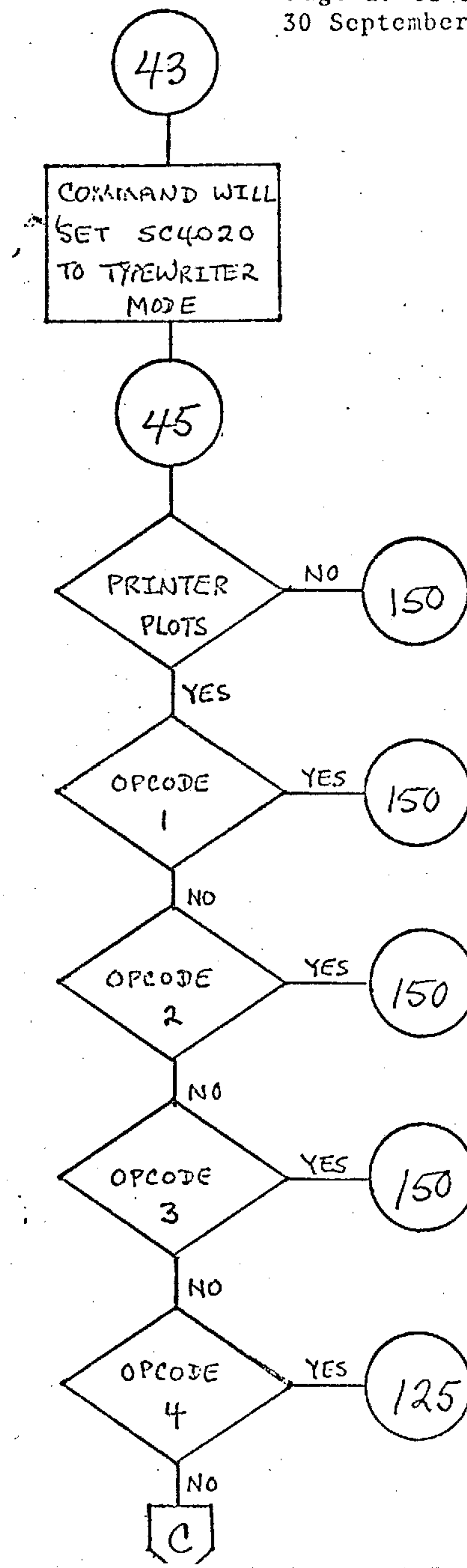
```
C ZERO BUFFER COUNT
 295      ICOUNT=0
C SAVE MODE OF OPERATION
 300      TYPING=TYPMOD
C RETURN IF NOT FRAME ADVANCE OR RESET
          IF (OPI.NE.17.AND.OPI.NE.4) RETURN
C INCREMENT FRAME COUNT AND ZERO LINE COUNT
          IFRM=IFRM+1
          LINECT=0
C RETURN
          RETURN
C SET ERROR INDICATOR AND RETURN
 350      TERR=.TRUE.
          RETURN
C NWUNIT ENTRY
          ENTRY NWUNIT (IPRNT,IPLOTR)
C SET PRINTER AND PLOTTER UNITS
          PRNT=IPRNT
          PLOT1S=IPLOTR
          RETURN
C FRAMES ENTRY
          ENTRY FRAMES (FRMCNT)
C RETURN FRAME COUNT
          FRMCNT=IFRM
          RETURN
C EMPTY ENTRY
          ENTRY EMPTY
C EMPTY PLOT BUFFER AND END FILE OUTPUT UNIT
          IF (ICOUNT.GT.0) WRITE (PLOT1S,2000) (OUT(I),I=1,ICOUNT)
          ICOUNT=0
          END FILE PLOT1S
          RETURN
C VBAR ENTRY
          ENTRY VBAR
C SET Y AXIS CHARACTER TO VERTICAL BAR
          II=BAR
          RETURN
C CONDNS ENTRY
          ENTRY CONDNS
C SET PRINT FORMAT FOR ONE PAGE INSTEAD OF TWO
          FORMAT(18)=PLUS
          RETURN
 2000     FORMAT (6A1)
          END
```

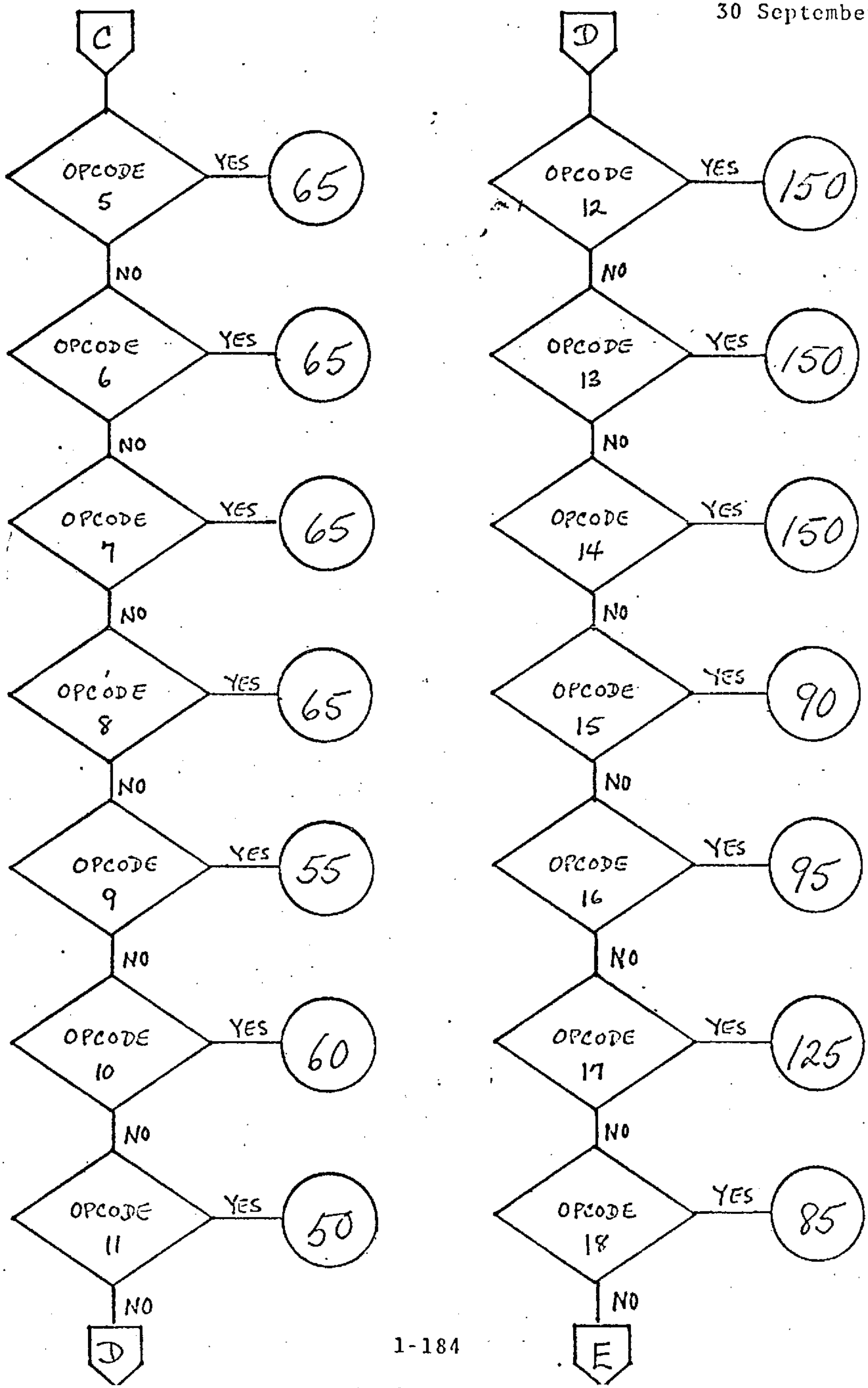

ENTRY SC4020
DEVICE SELECTION OPCODE
NO
1
YES
SET DEVICE SWITCHES
PRINTER SELECT ONLY
NO
1
YES
RETURN
ENTRY FRMADV
SET OPCODE INDICATOR FOR RESET
1
1
NEED INITIALIZATION
NO
20
YES
ZERO FRAME COUNT AND COMMAND BUFFER COUNTER
SET UP CHARACTER TRANSLATION MATRIX USING SCHAR
SET PRINT BUFFER TO BLANKS
ZERO CURRENT POINT REGISTER VALUES
20

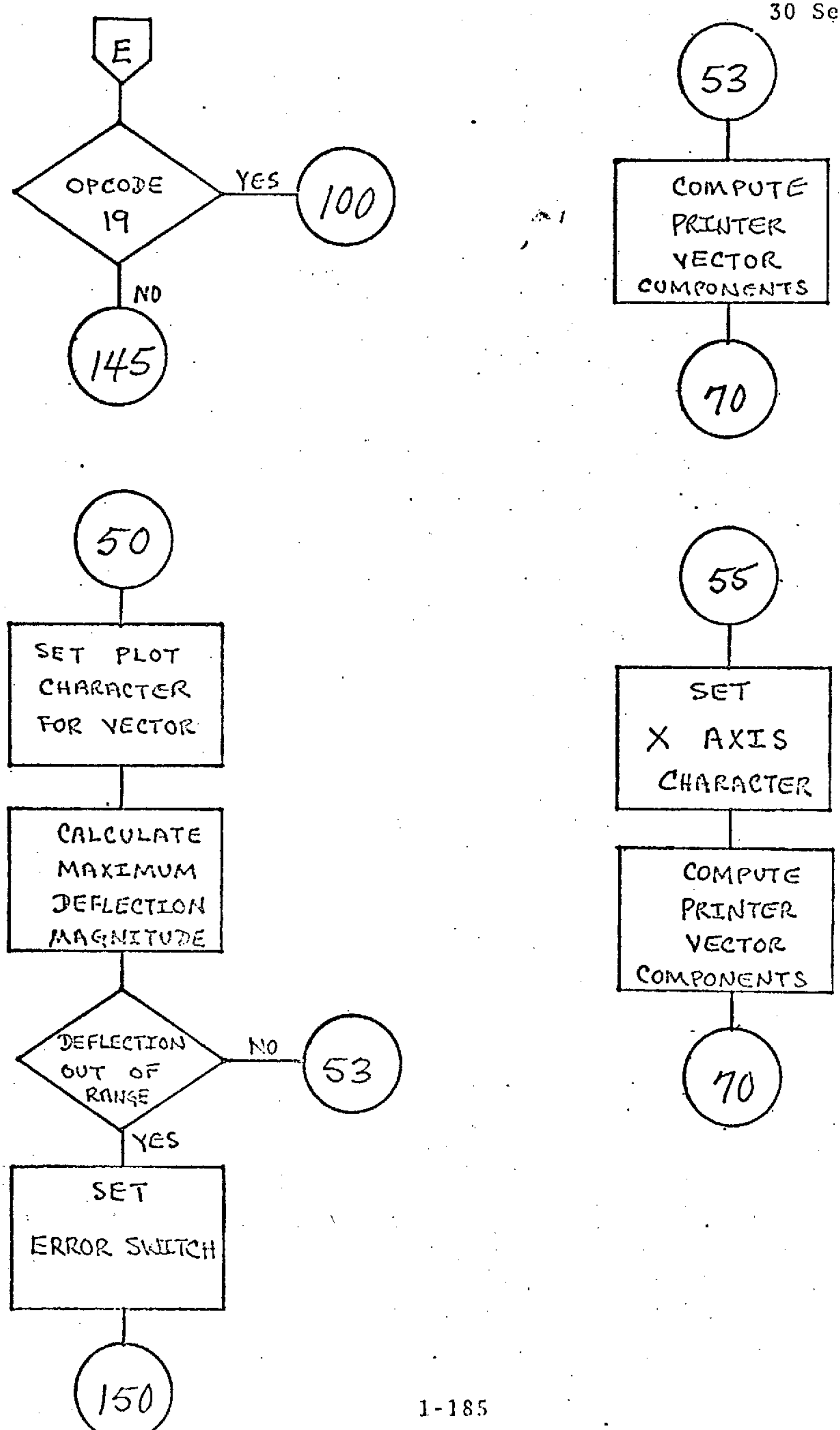

20
INPUT COMMAND IN WRONG MODE
NO
25
YES
SET PRINTER AND PLOTTER ERROR FLAGS
FRAME ADVANCE
NO
RETURN
YES
SET OPCODE INDICATOR FOR RESET
25
25
SET CURRENT PROGRAM MODE
OPCODE 1
YES
45
NO
OPCODE 2
YES
45
NO
OPCODE 3
YES
45
NO
OPCODE 4
YES
30
NO
OPCODE 5
YES
45
NO
OPCODE 6
YES
45
NO
A

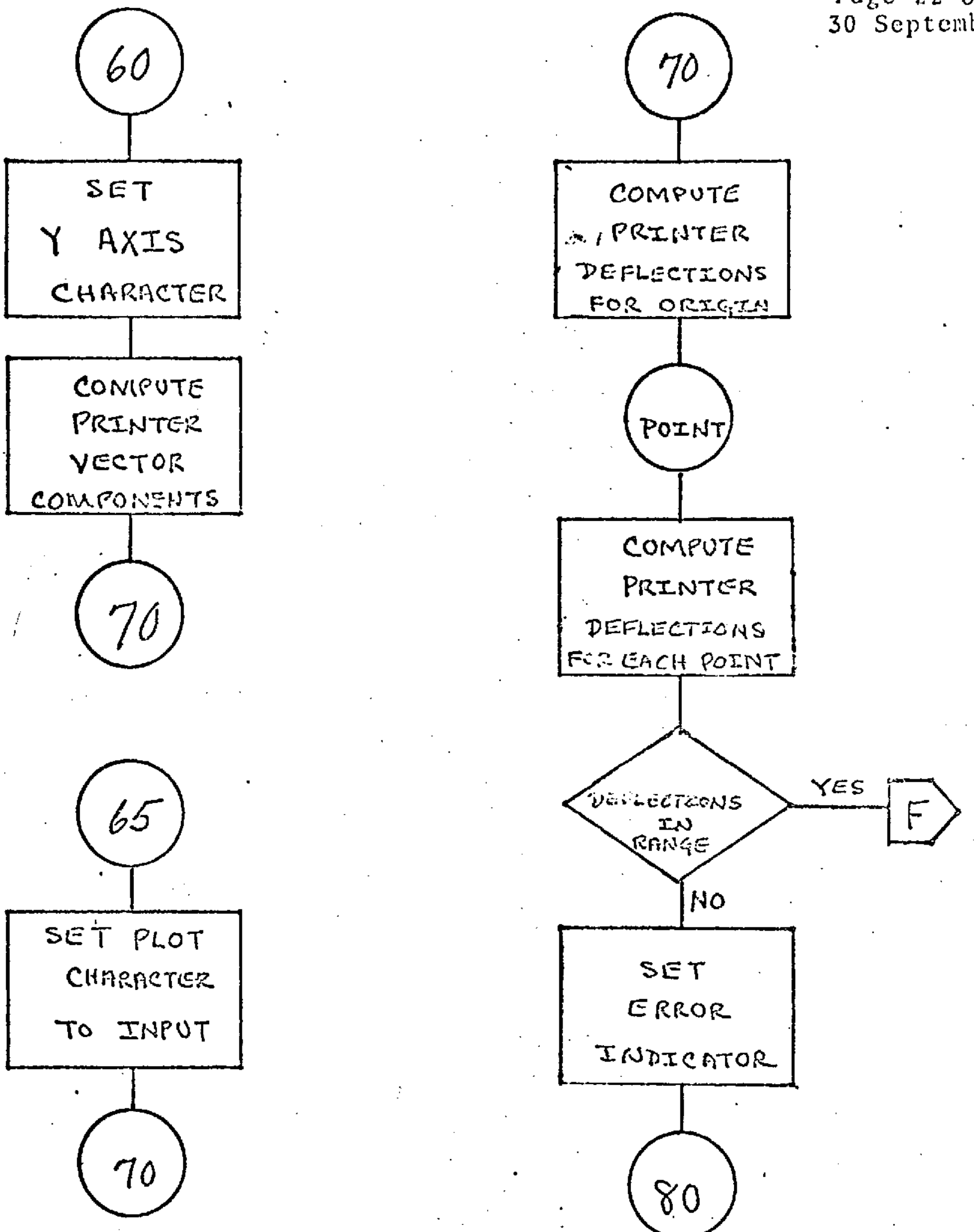

A
OPCODE 7
YES
45
NO
OPCODE 8
YES
45
NO
OPCODE 9
YES
35
NO
OPCODE 10
YES
35
NO
OPCODE 11
YES
35
NO
OPCODE 12
YES
45
NO
OPCODE 13
YES
45
NO
B
B
OPCODE 14
YES
45
NO
OPCODE 15
YES
43
NO
OPCODE 16
YES
43
NO
OPCODE 17
YES
30
NO
OPCODE 18
YES
45
NO
OPCODE 19
YES
45
NO
30

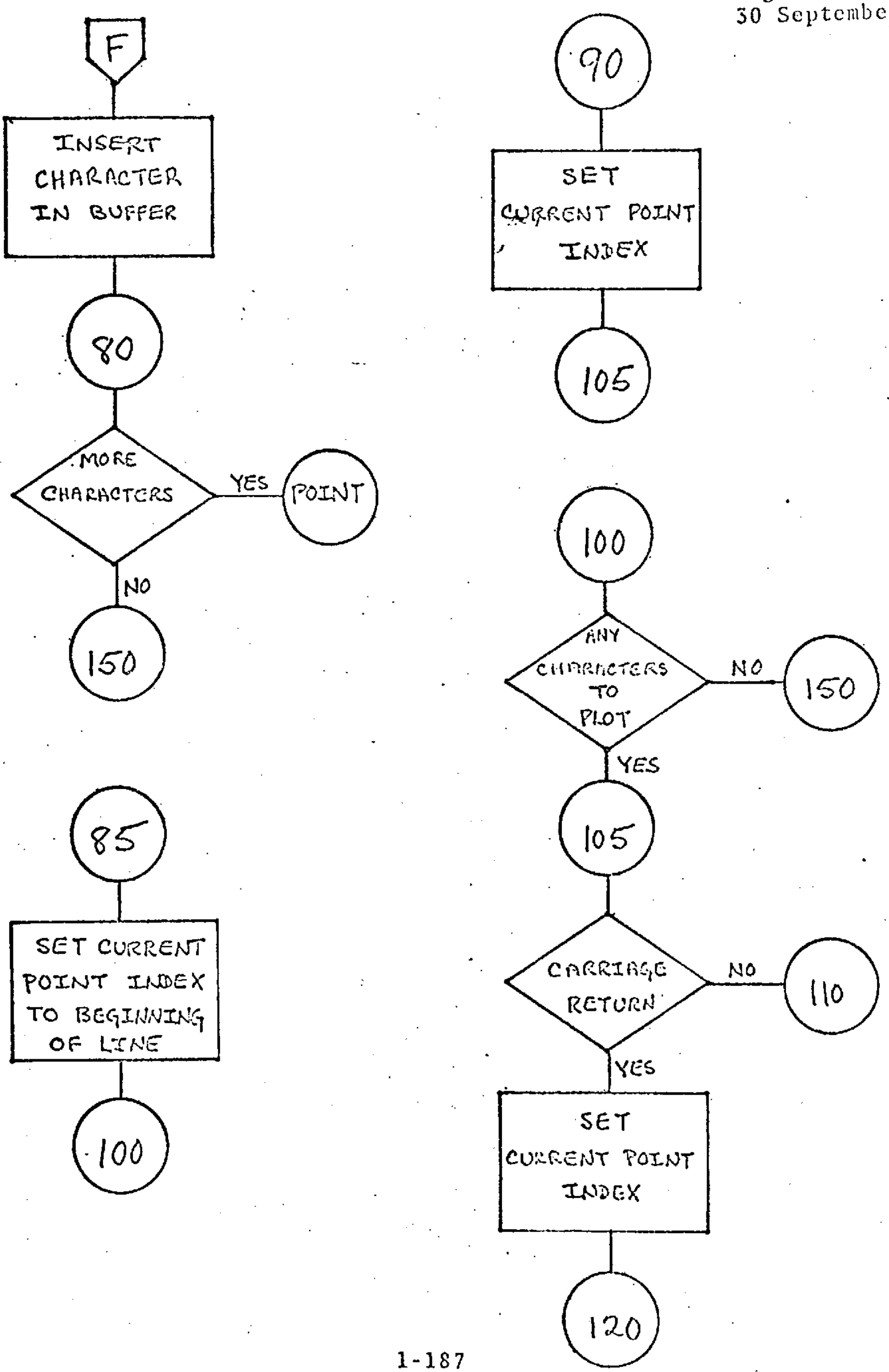

30
COMMAND WILL SET SC4020 TO PLOTTING MODE
45
35
RECOVER X VECTOR COMPONENT OR AXIS END POINT
45

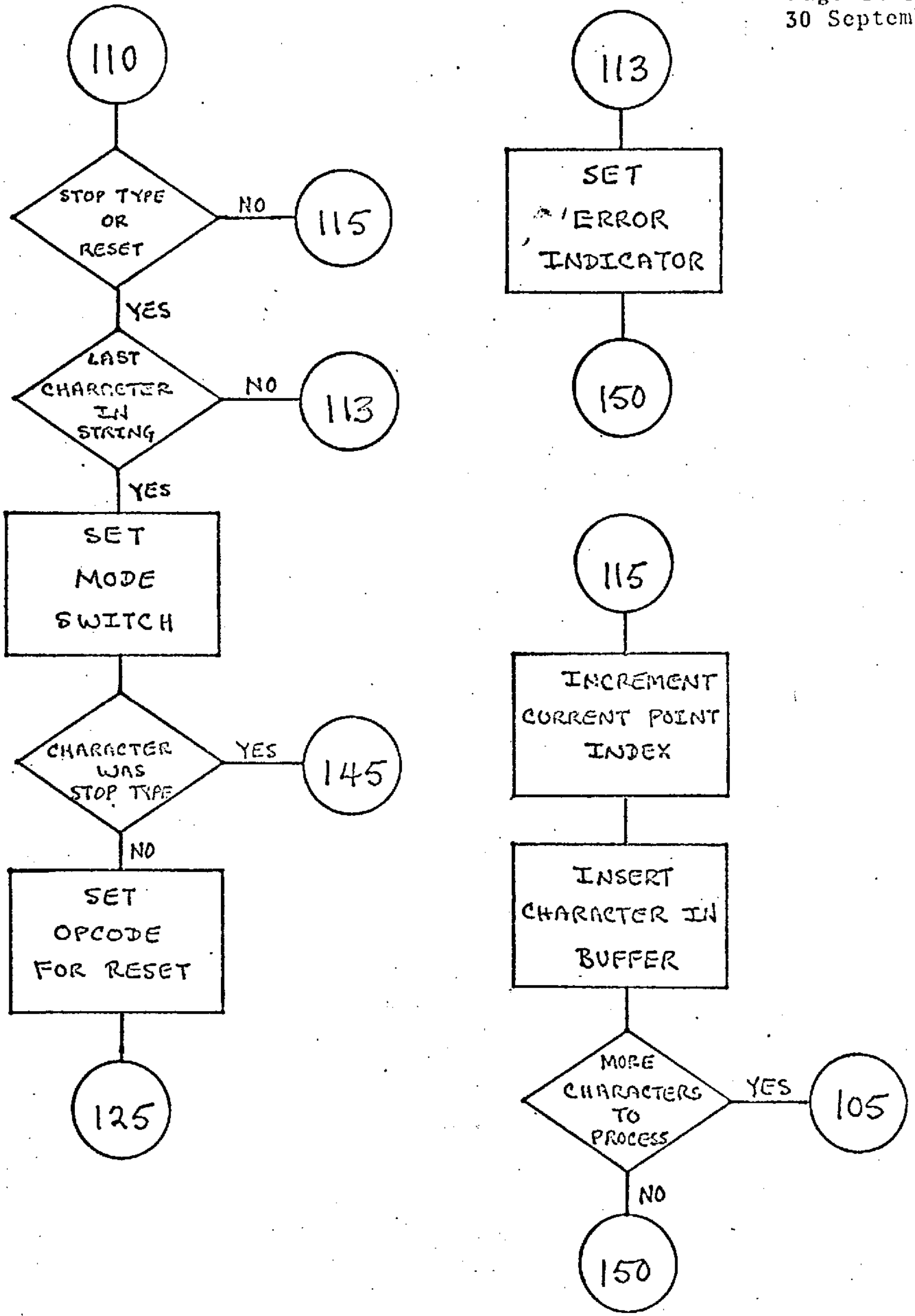

43
COMMAND WILL SET SC4020 TO TYPEWRITER MODE
45
PRINTER PLOTS
NO
150
YES
OPCODE 1
YES
150
NO
OPCODE 2
YES
150
NO
OPCODE 3
YES
150
NO
OPCODE 4
YES
125
NO
C


C
OPCODE 5
YES
65
NO
OPCODE 6
YES
65
NO
OPCODE 7
YES
65
NO
OPCODE 8
YES
65
NO
OPCODE 9
YES
55
NO
OPCODE 10
YES
60
NO
OPCODE 11
YES
50
NO
D
D
OPCODE 12
YES
150
NO
OPCODE 13
YES
150
NO
OPCODE 14
YES
150
NO
OPCODE 15
YES
90
NO
OPCODE 16
YES
95
NO
OPCODE 17
YES
125
NO
OPCODE 18
YES
85
NO
E

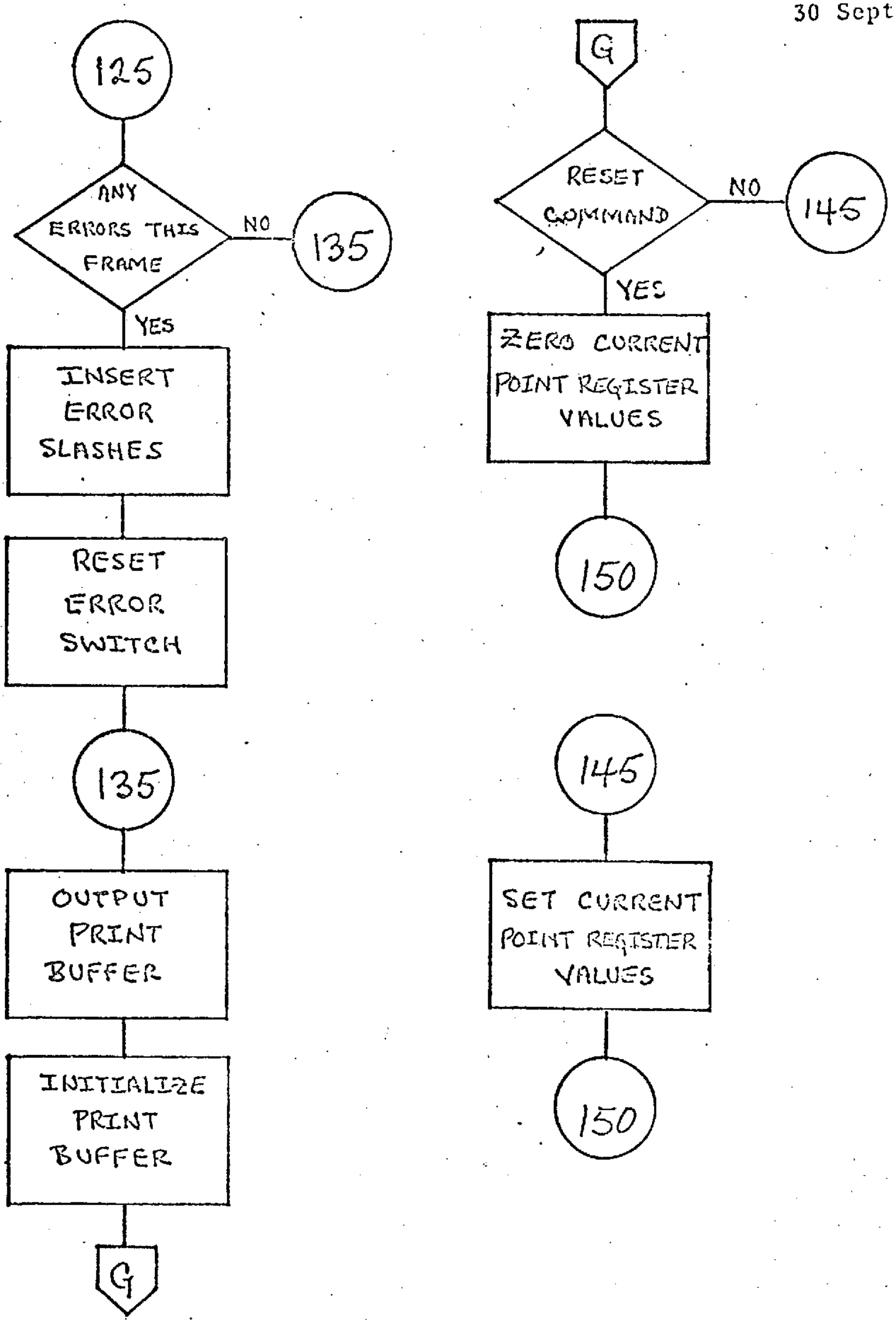

E
OPCODE 19
YES
100
NO
145
53
COMPUTE PRINTER VECTOR COMPONENTS
70
50
SET PLOT CHARACTER FOR VECTOR
CALCULATE MAXIMUM DEFLECTION MAGNITUDE
DEFLECTION OUT OF RANGE
NO
53
YES
SET ERROR SWITCH
150
55
SET X AXIS CHARACTER
COMPUTE PRINTER VECTOR COMPONENTS
70

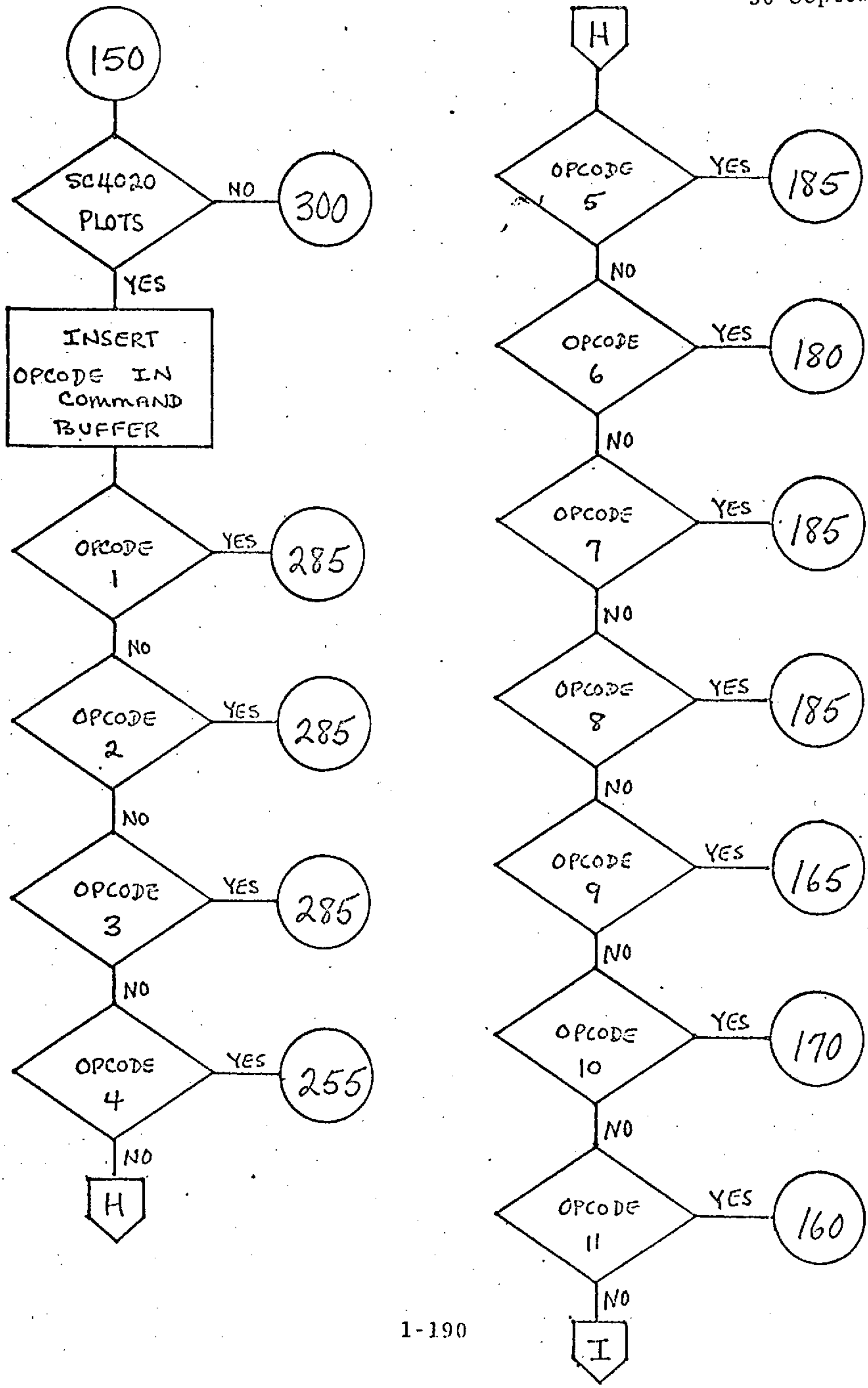

60
SET Y AXIS CHARACTER
COMPUTE PRINTER VECTOR COMPONENTS
70
65
SET PLOT CHARACTER TO INPUT
70
70
COMPUTE PRINTER DEFLECTIONS FOR ORIGIN
POINT
COMPUTE PRINTER DEFLECTIONS FOR EACH POINT
DEFLECTIONS IN RANGE
YES
F
NO
SET ERROR INDICATOR
80

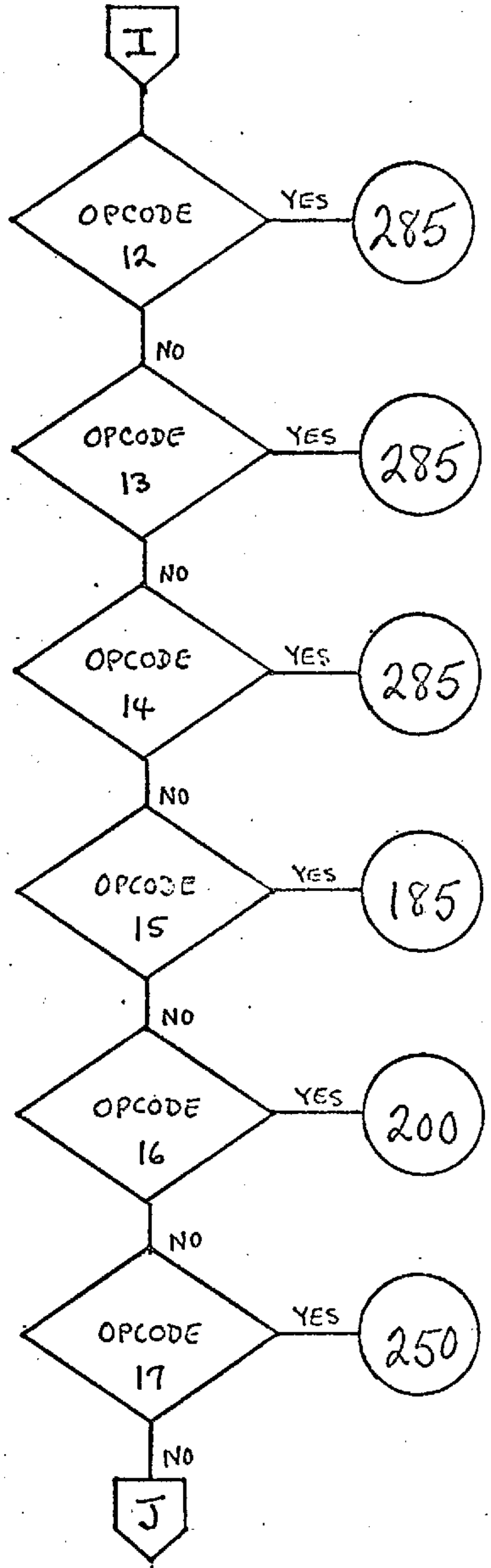

F
INSERT CHARACTER IN BUFFER
80
MORE CHARACTERS
YES
POINT
NO
150
85
SET CURRENT POINT INDEX TO BEGINNING OF LINE
100
90
SET CURRENT POINT INDEX
105
100
ANY CHARACTERS TO PLOT
NO
150
YES
105
CARRIAGE RETURN
NO
110
YES
SET CURRENT POINT INDEX
120

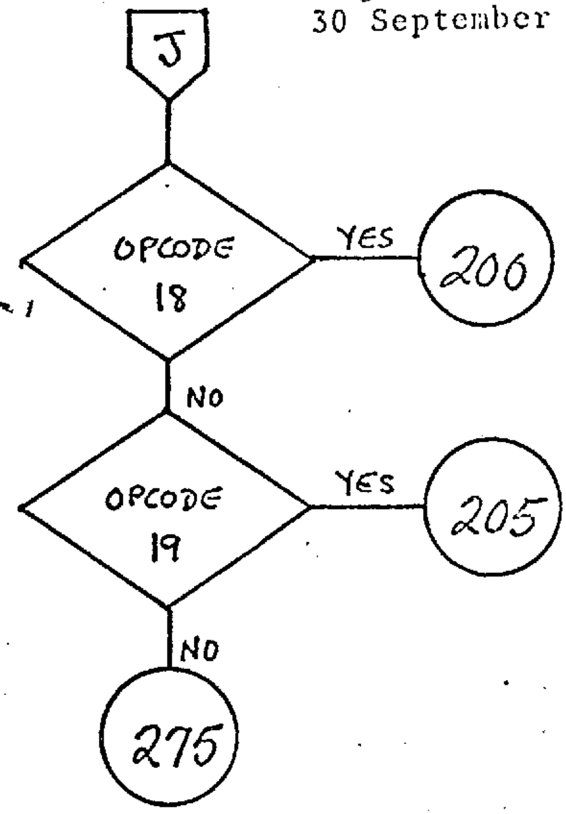

110
STOP TYPE OR RESET
NO
115
YES
LAST CHARACTER IN STRING
NO
113
YES
SET MODE SWITCH
CHARACTER WAS STOP TYPE
YES
145
NO
SET OPCODE FOR RESET
125
113
SET ERROR INDICATOR
150
115
INCREMENT CURRENT POINT INDEX
INSERT CHARACTER IN BUFFER
MORE CHARACTERS TO PROCESS
YES
105
NO
150

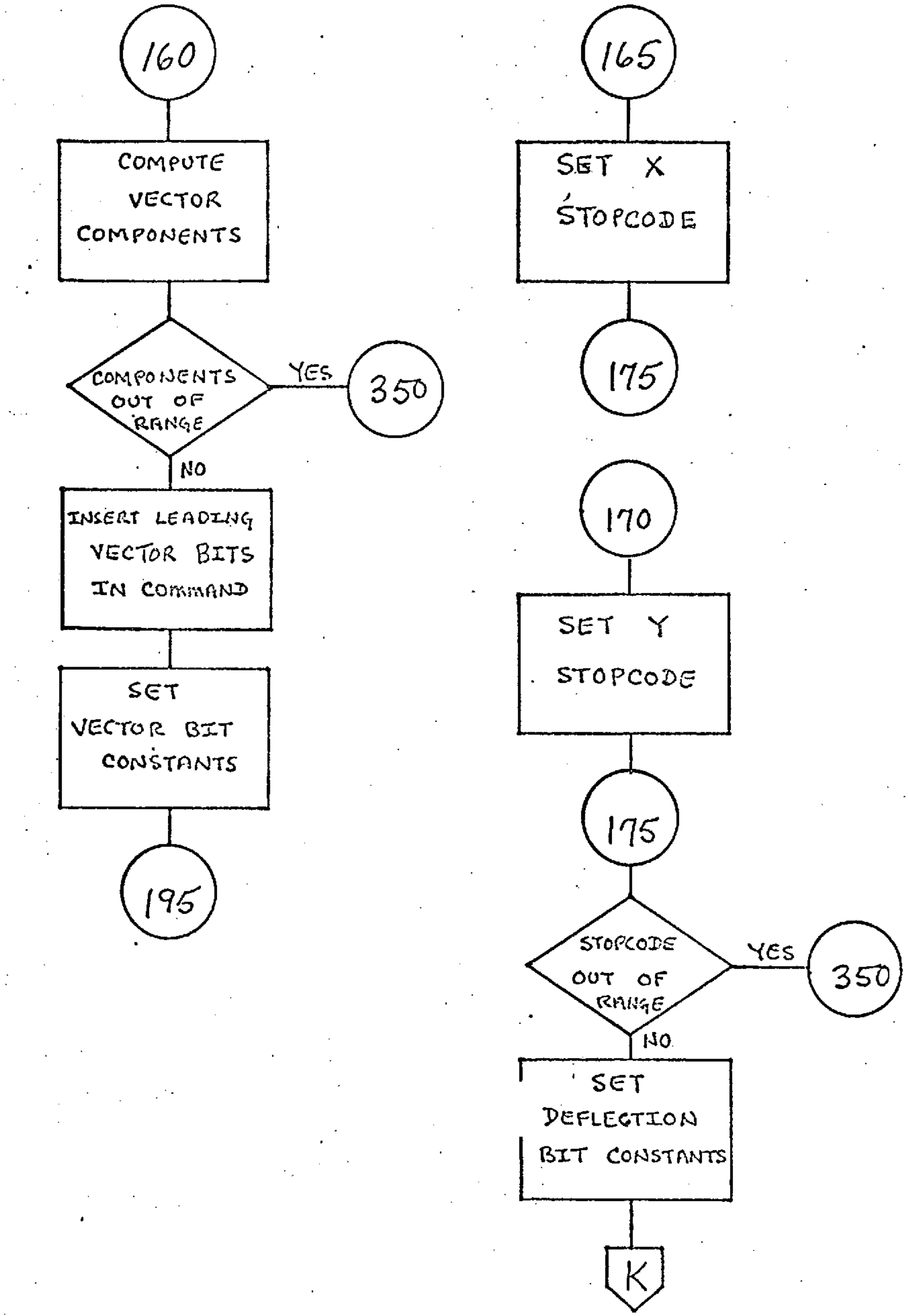
C-3

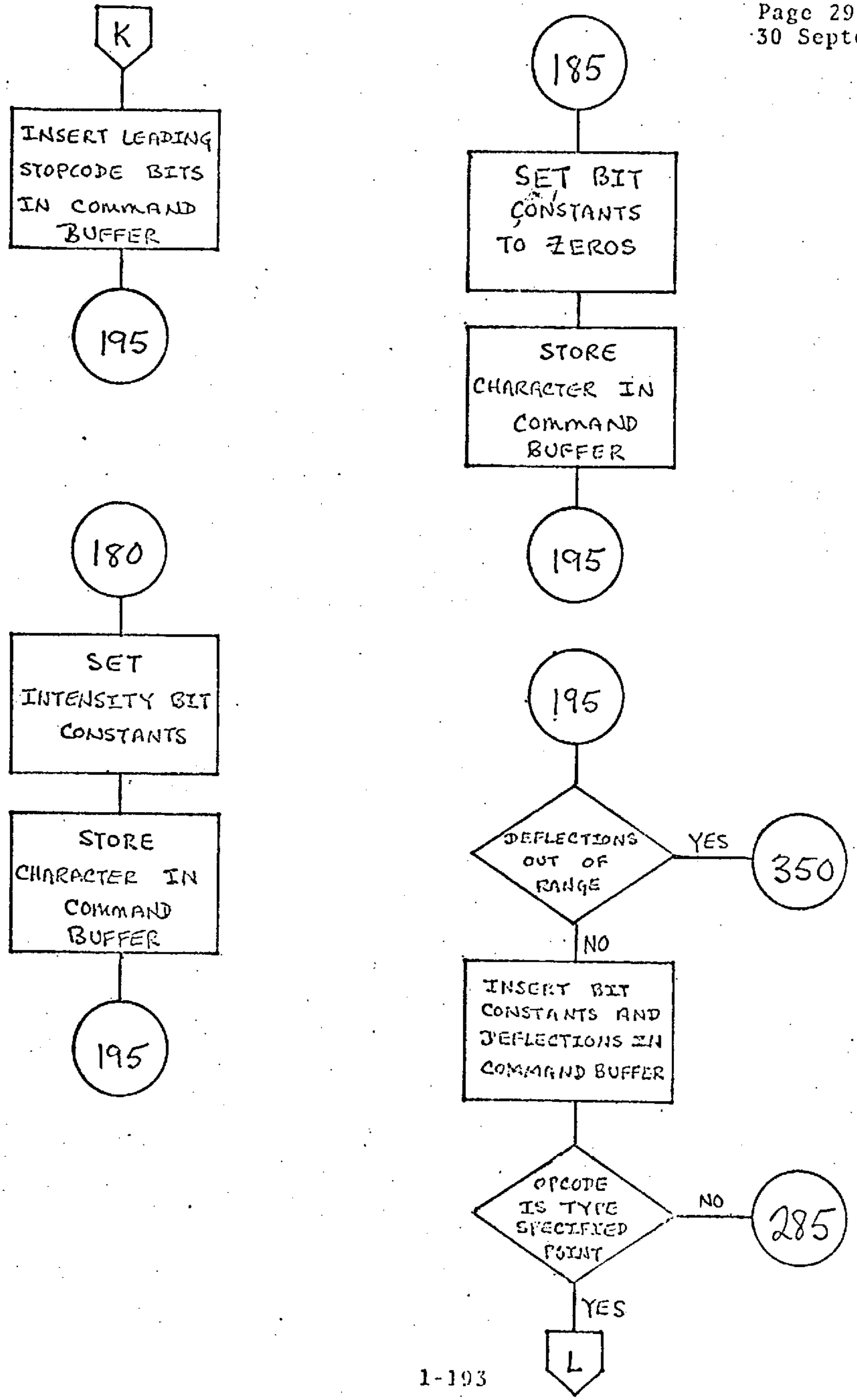

125
ANY ERRORS THIS FRAME
NO
135
YES
INSERT ERROR SLASHES
RESET ERROR SWITCH
135
OUTPUT PRINT BUFFER
INITIALIZE PRINT BUFFER
G
G
RESET COMMAND
NO
145
YES
ZERO CURRENT POINT REGISTER VALUES
150
145
SET CURRENT POINT REGISTER VALUES
150

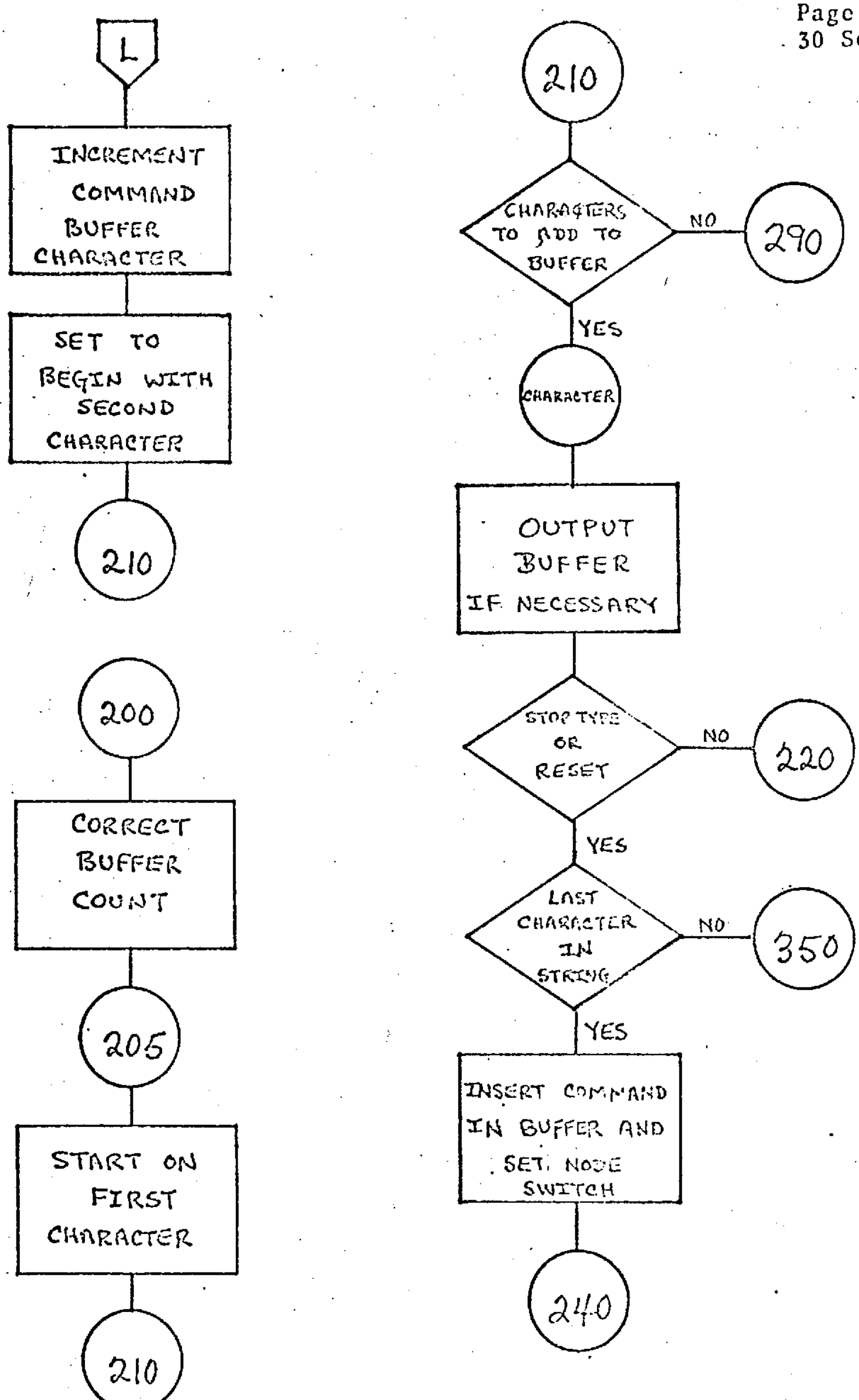

150
SC4020 PLOTS
NO
300
YES
INSERT OPCODE IN COMMAND BUFFER
OPCODE 1
YES
285
NO
OPCODE 2
YES
285
NO
OPCODE 3
YES
285
NO
OPCODE 4
YES
255
NO
H
H
OPCODE 5
YES
185
NO
OPCODE 6
YES
180
NO
OPCODE 7
YES
185
NO
OPCODE 8
YES
185
NO
OPCODE 9
YES
165
NO
OPCODE 10
YES
170
NO
OPCODE 11
YES
160
NO
I

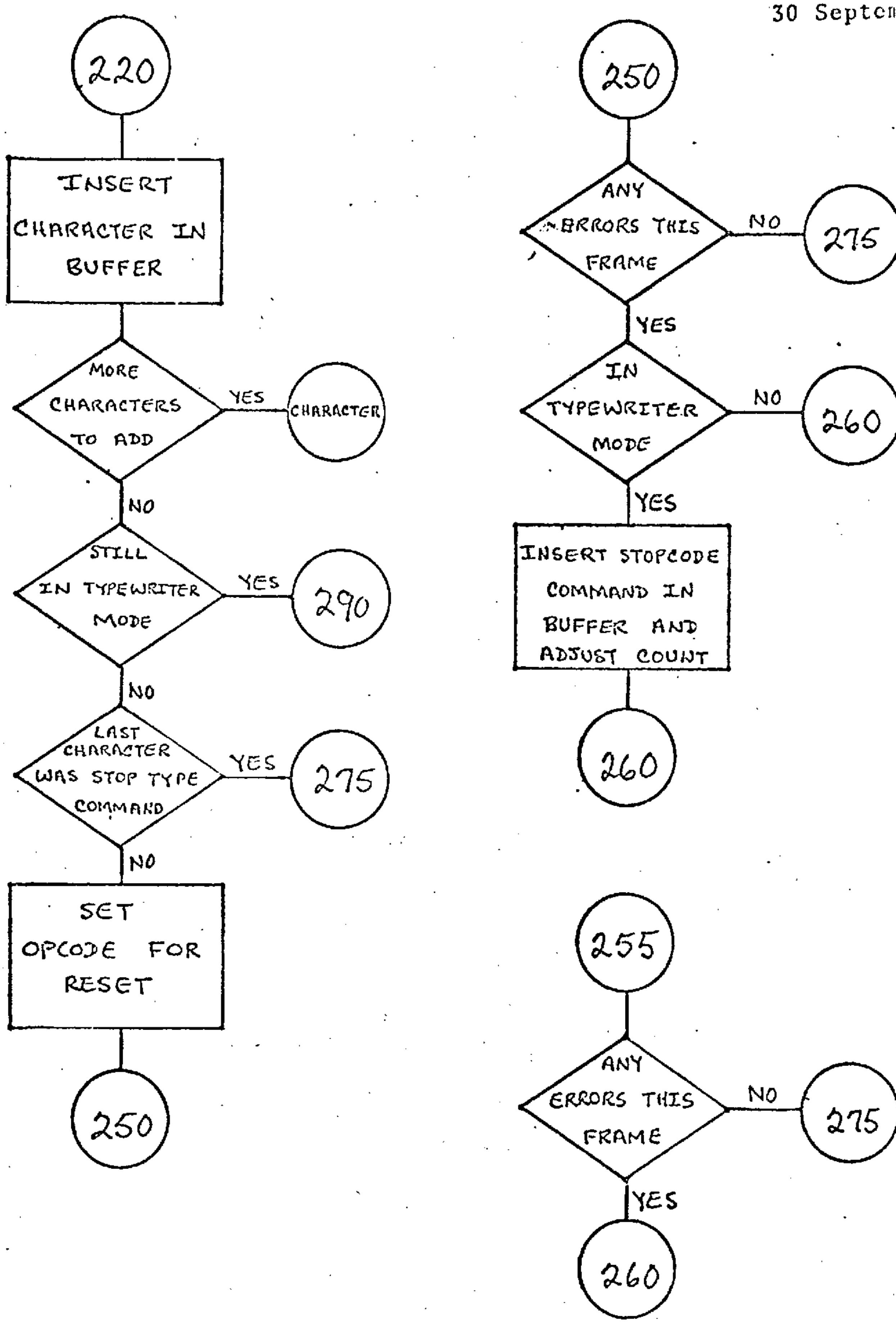

I
OPCODE 12
YES
285
NO
OPCODE 13
YES
285
NO
OPCODE 14
YES
285
NO
OPCODE 15
YES
185
NO
OPCODE 16
YES
200
NO
OPCODE 17
YES
250
NO
J

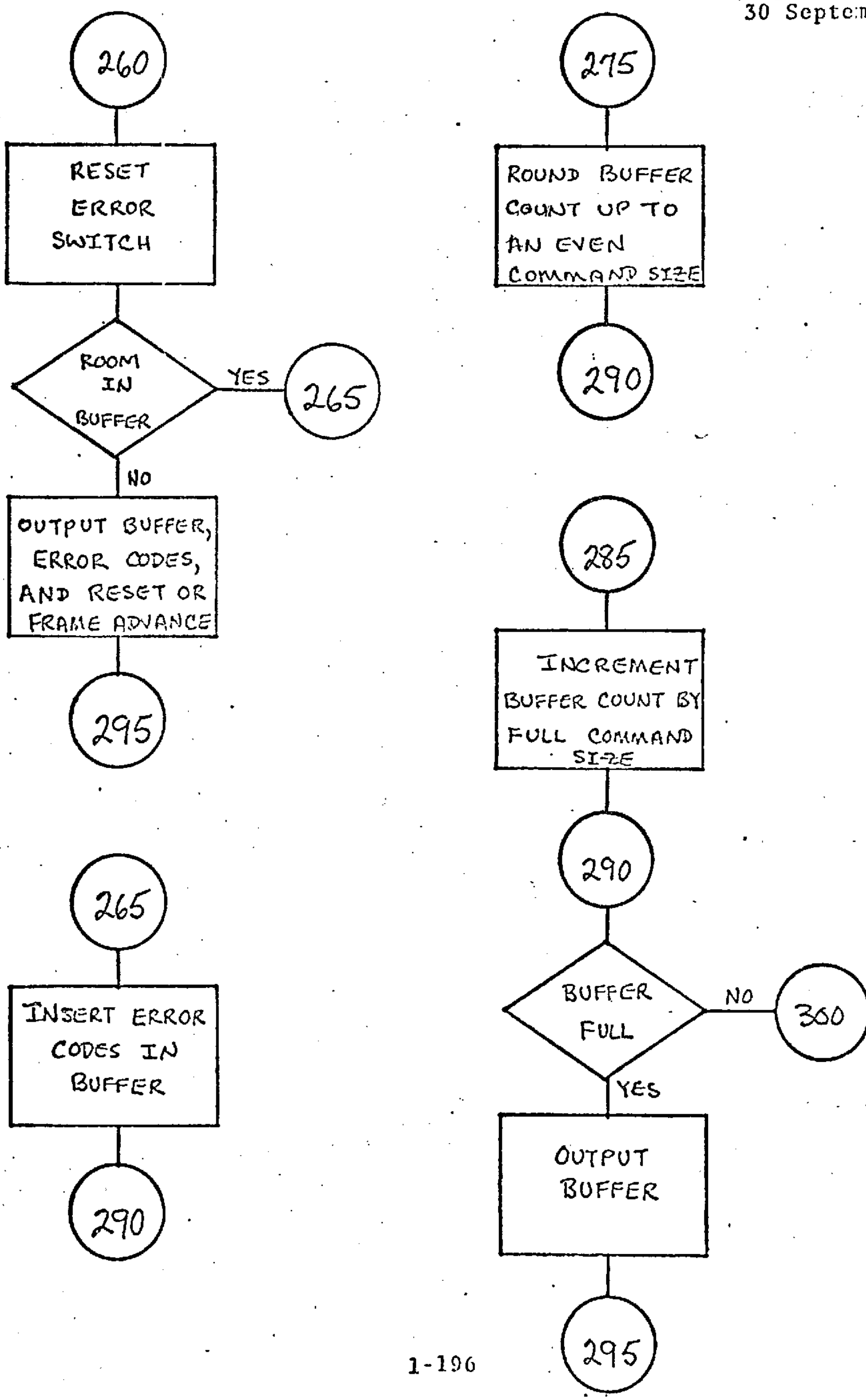

J
OPCODE 18
YES
206
NO
OPCODE 19
YES
205
NO
275

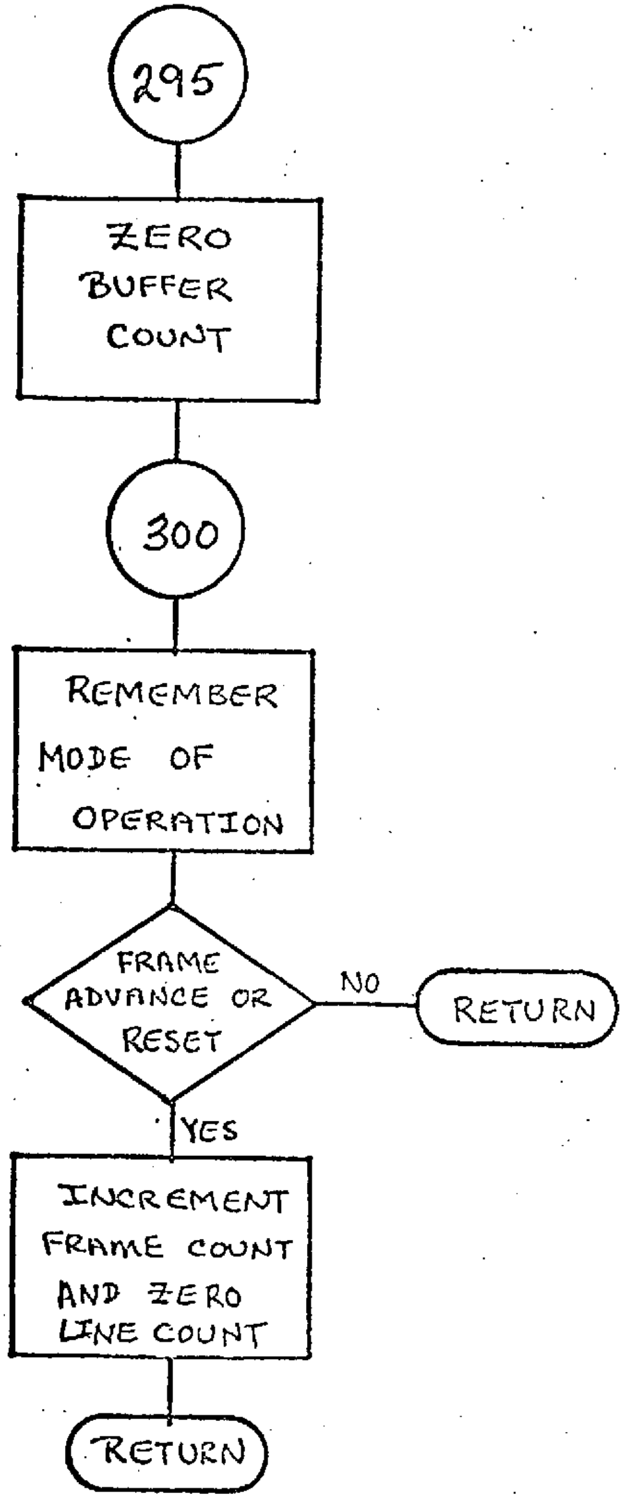

160
COMPUTE VECTOR COMPONENTS
COMPONENTS OUT OF RANGE
YES
350
NO
INSERT LEADING VECTOR BITS IN COMMAND
SET VECTOR BIT CONSTANTS
195
165
SET X STOPCODE
175
170
SET Y STOPCODE
175
STOPCODE OUT OF RANGE
YES
350
NO
SET DEFLECTION BIT CONSTANTS
K

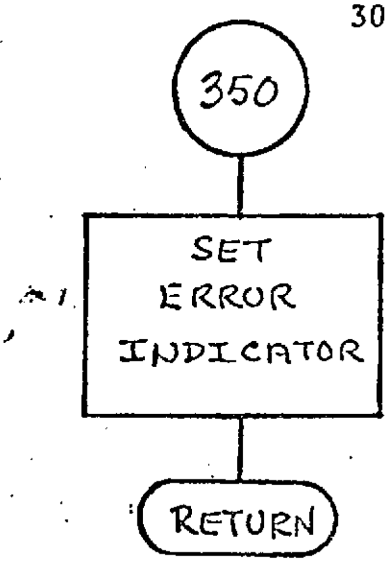

K
INSERT LEADING STOPCODE BITS IN COMMAND BUFFER
195
180
SET INTENSITY BIT CONSTANTS
STORE CHARACTER IN COMMAND BUFFER
195
185
SET BIT CONSTANTS TO ZEROS
STORE CHARACTER IN COMMAND BUFFER
195
195
DEFLECTIONS OUT OF RANGE
YES
350
NO
INSERT BIT CONSTANTS AND DEFLECTIONS IN COMMAND BUFFER
OPCODE IS TYPE SPECIFIED POINT
NO
285
YES
L


L
INCREMENT COMMAND BUFFER CHARACTER
SET TO BEGIN WITH SECOND CHARACTER
210
200
CORRECT BUFFER COUNT
205
START ON FIRST CHARACTER
210
210
CHARACTERS TO ADD TO BUFFER
NO
290
YES
CHARACTER
OUTPUT BUFFER IF NECESSARY
STOP TYPE OR RESET
NO
220
YES
LAST CHARACTER IN STRING
NO
350
YES
INSERT COMMAND IN BUFFER AND SET MODE SWITCH
240


220
INSERT CHARACTER IN BUFFER
MORE CHARACTERS TO ADD
YES
CHARACTER
NO
STILL IN TYPEWRITER MODE
YES
290
NO
LAST CHARACTER WAS STOP TYPE COMMAND
YES
275
NO
SET OPCODE FOR RESET
250
250
ANY ERRORS THIS FRAME
NO
275
YES
IN TYPEWRITER MODE
NO
260
YES
INSERT STOPCODE COMMAND IN BUFFER AND ADJUST COUNT
260
255
ANY ERRORS THIS FRAME
NO
275
YES
260


260
RESET ERROR SWITCH
ROOM IN BUFFER
YES
265
NO
OUTPUT BUFFER, ERROR CODES, AND RESET OR FRAME ADVANCE
295
265
INSERT ERROR CODES IN BUFFER
290
275
ROUND BUFFER COUNT UP TO AN EVEN COMMAND SIZE
290
285
INCREMENT BUFFER COUNT BY FULL COMMAND SIZE
290
BUFFER FULL
NO
300
YES
OUTPUT BUFFER
295


295
ZERO
BUFFER
COUNT
300
REMEMBER
MODE OF
OPERATION
FRAME
ADVANCE OR
RESET
NO
RETURN
YES
INCREMENT
FRAME COUNT
AND ZERO
LINE COUNT
RETURN


350
SET
ERROR
INDICATOR
RETURN

SCHAR

DESCRIPTION

SCHAR is a function routine which when given an SC4020 character value determines the corresponding EBCDIC character value.

The method involves an array XCHAR which is set up by a data statement so that if I is the input and SCHAR is the output, then basically SCHAR=XCHAR(I).

NAME	SCHAR
PURPOSE	THE FUNCTION VALUE IS THE EBCDIC CHARACTER VALUE CORRESPONDING TO THE INPUT SC4020 CHARACTER VALUE
CALLING SEQUENCE	SCHAR(I)

SYMBOL	TYPE	DESCRIPTION
I	I	INPUT - SC4020 CHARACTER VALUE
SCHAR	I	OUTPUT - EBCDIC CHARACTER VALUE

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

```
      INTEGER FUNCTION SCHAR (I)
      LOGICAL *1 LX,CHAR,XCHAR,DUM(4)
      DIMENSION CHAR(1),XCHAR(64)
      EQUIVALENCE (IW,DUM(1)),(LX,DUM(4))
      EQUIVALENCE (XCHAR(2),CHAR(1))
      DATA IW /0/
C TABLE CONTAINS EBCDIC CHARACTER VALUES CORRESPONDING TO SC4020
C CHARACTER SET - NOTE THAT APPROPRIATE SC4020 VALUES RANGE FROM
C 0 TO 63
C Z4A=CENT,Z5A=EXCLAMATION POINT,ZE0=0-2-8 PUNCH
      DATA XCHAR  /
     • '0','1','2','3','4','5','6','7',
     • '8','9',':','=','"',1H','#','@',
     • '+','A','B','C','D','E','F','G',
     • 'H','I',Z4A,'.',')','%','&','?',
     • '-','J','K','L','M','N','O','P',
     • 'Q','R',Z5A,'$','*','<',';','¬',
     • ' ','/','S','T','U','V','W','X',
     • 'Y','Z',ZE0,',','(','_','>','|'/
C PERFORM TABLE LOOKUP
      LX=CHAR(I)
      SCHAR=IW
      RETURN
      END
```

TIMING

DESCRIPTION

TIMING has one entry, NOW. This is used (by DATE, for example) to determine the date and time of day. A system macro is used to get the date in YYDDD integer format and the time of day in hundredths of seconds.

CONTROL SECTION NAME TIMING

ENTRY POINT — PURPOSE

NOW — OBTAINS THE CURRENT DATE IN IBM PACKED INTEGER FORMAT (YYDDD) AND THE TIME OF DAY IN INTEGER HUNDREDTHS OF SECONDS

CALLING SEQUENCE CALL NOW(IYDD,IHM)

SYMBOL	TYPE	DESCRIPTION
IYDD	I	OUTPUT - YYDDD FOR CURRENT DATE
IHM	I	OUTPUT - TIME OF DAY IN HUNDREDTHS OF SECONDS

SUBROUTINES USED NONE

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

RESTRICTIONS NONE

REFERENCES NONE

```
*
TIMING   START 0
         ENTRY NOW
NOW      SAVE (14,12),,*
         BALR  4,0
         USING *,4
         LM    5,6,0(1) ADDRESSES OF OUTUT ARGUMENTS
         TIME  BIN
         ST    0,0(6) RETURN TIME OF DAY IN HUNDREDTHS OF SECONDS
         ST    1,TEMP+4 MUST CONVERT YYDDD TO BINARY INTEGER
         CVB   1,TEMP
         ST    1,0(5) BEFORE RETURNING IT
         RETURN (14,12)
*
TEMP     DC    D'0'
         END
```

TYPLIN

DESCRIPTION

TYPLIN is a routine which is used to type line information on the SC4020.

The SC4020 is put into typewriter mode and the carriage control character is inspected. A '1' causes a frame advance for example. The line is typed and the SC4020 is put back into plotting mode.

SETPAG is an entry in TYPLIN so the user can specify where the typing should start. Otherwise, typing starts where it ended last or at the beginning if typing has been just started.

NAME	TYPLIN
ENTRY POINT	PURPOSE
TYPLIN	TO TYPE A LINE OF INFORMATION
SETPAG	TO SET LINE NUMBER AND COLUMN NUMBER OF TYPLIN

CALLING SEQUENCE CALL TYPLIN(LINE,N)

SYMBOL	TYPE	DESCRIPTION
LINE	A	INPUT - ARRAY OF CHARACTERS (FIRST CHARACTER IN "CARRIAGE CONTROL")
N	I	INPUT - LENGTH OF LINE

CALLING SEQUENCE CALL SETPAG(LINES,ICOL)

SYMBOL	TYPE	DESCRIPTION
LINES	I	INPUT - LINE NUMBER FOR NEXT CALL TO TYPLIN
ICOL	I	INPUT - COLUMN NUMBER FOR SUBSEQUENT CALLS TO TYPLIN

SUBROUTINE USED	SC4020
COMMON BLOCK	CPLOT$
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE TYPLIN(LINE,N)
      COMMON /CPLOT$/ G1(15),LININC,LINECT
      LOGICAL*1 LINE(1),LX,BLANK,DUM(4),STOPCD
      EQUIVALENCE (IW,DUM(1)),(LX,DUM(4))
      DATA BLANK,STOPCD/1H ,Z0A /
      DATA IONE,IZERO /Z000000F1,Z000000F0 /
      DATA IW/0/
      DATA IX /0/
C PERFORM INDICATED CARRIAGE CONTROL OPERATION
      LX=LINE(1)
      IF(IW.EQ.IONE) GO TO 10
      IF(IW.EQ.IZERO) LINECT=LINECT+1
      IF(LINECT*LININC.LT.1024) GO TO 50
   10 LINECT=0
```

```
      CALL SC4020 (17,0,0,0,0)
   50 IY=1023-LINECT*LININC
      IF(N.EQ.1) RETURN
C TYPE LINE
      N1=MIN0(N+1,130)
      LX=LINE(N1)
      LINE(N1)=STOPCD
      CALL SC4020 (15,IX,IY,LINE(2),N)
      LINE(N1)=LX
      LINECT=LINECT+1
      RETURN
C SETPAG ENTRY
      ENTRY SETPAG (LINES,ICOL)
C SET LINE NUMBER AND STARTING COLUMN
      IX=ICOL*8-8
      IF(ICOL.GT.128.OR.ICOL.LT.1) IX=0
      LINECT=LINES
      RETURN
      END
```

UCS

DESCRIPTION

UCS contain the character description arrays for the standard EBCDIC character font. These are set up in data statements.

If BLKLET is to be used (to produce block letters) than some character description must be input via CSET in BLKLET. In order to aid the programmer, a simple call to UCS will accomplish the above.

NAME	UCS
PURPOSE	TO CALL CSET WITH A STANDARD 360 CHARACTER SET
CALLING SEQUENCE	CALL UCS
SUBROUTINE USED	CSET
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE UCS
C CHARACTER SET
      LOGICAL*1 ICHAR(62)/
     *'A','B','C','D','E','F','G','H','I','J','K','L','M','N','O','P',
     *'Q','R','S','T','U','V','W','X','Y','Z','0','1','2','3','4','5',
     *'6','7','8','9',' ','.','<','(','+','|','&','!','$','*',')',';',
     *'¬','-','/',',','%','_','>','?',':','#','@',1H','=','"'/
C INDEX OF ABOVE CHARACTERS IN IVEC
      INTEGER*2 IPOS(63)/
     *  1,  4, 14, 21, 27, 31, 34, 44, 47, 50, 56, 59, 61, 65, 68, 76,
     * 82, 91, 98,109,111,116,118,122,124,127,131,139,142,151,162,165,
     *173,184,187,202,213,221,225,227,232,234,235,249,257,267,270,275,
     *284,286,287,288,293,310,311,313,321,329,333,350,351,353,355/
C RASTER COORDINATES OF VECTORS FOR EACH CHARACTER -
C PACKED AS X,Y,DX,DY
      INTEGER*2 IVEC(190)/
     *Z0099,Z6039,Z13A0,Z0069,Z09A0,Z6361,Z6662,Z4581,Z4520,Z6441,
     *Z6163,Z4C81,Z4020,Z6742,Z4940,Z0782,Z0265,Z2042,Z2080,Z4082,
     *Z0069,Z09A0,Z6742,Z6265,Z4082,Z4020,Z0069,Z09C0,Z05A0,Z00C0,
     *Z0069,Z09C0,Z05A0,Z4361,Z4480,Z6262,Z4082,Z4040,Z2042,Z0265,
     *Z0782,Z2980,Z6742,Z0069,Z6069,Z05C0,Z3069,Z2080,Z2980,Z6267,
     *Z4082,Z4040,Z2042,Z0262,Z4270,Z0069,Z03C6,Z6025,Z0069,Z00C0,
     *Z0069,Z3534,Z3594,Z6069,Z0069,Z6609,Z6069,Z0265,Z0782,Z2980,
     *Z6742,Z6265,Z4082,Z4040,Z2042,Z0069,Z09A0,Z6851,Z6563,Z5471,
     *Z5410,Z0265,Z0782,Z2980,Z6742,Z6265,Z4082,Z4040,Z2042,Z6042,
     *Z0069,Z09E0,Z6851,Z6563,Z5471,Z5410,Z6044,Z6851,Z5920,Z0871,
     *Z0662,Z1551,Z15A0,Z6451,Z6163,Z5071,Z5020,Z1051,Z3069,Z09C0,
     *Z0267,Z2042,Z2080,Z4082,Z6267,Z3039,Z3099,Z2049,Z2074,Z4054,
     *Z4089,Z0069,Z6009,Z3534,Z3594,Z3060,Z00C0,Z00C9,Z09C0,Z15A0,
     *Z0167,Z0871,Z19A0,Z6851,Z1051,Z10A0,Z5071,Z6167,Z2080,Z3069,
     *Z2772,Z0782,Z2980,Z6742,Z6661,Z4462,Z444C,Z0183,Z0061,Z09C0,
     *Z0871,Z19A0,Z6851,Z6662,Z5571,Z5530,Z6451,Z6163,Z5071,Z5020,
     *Z1051,Z5069,Z0485,Z04C0,Z6900,Z0663,Z0680,Z6551,Z6164,Z5071,
     *Z5020,Z1C51,Z6851,Z5920,Z0871,Z0167,Z1051,Z10AC,Z5071,Z6163,
```

```
     .Z6451,Z5520,Z0471,Z0861,Z09C0,Z20A9,Z19A0,Z6851,Z6662,Z5571/
      EQUIVALENCE (IVEC(191),IVEC1(1))
      INTEGER*2 IVEC1(164)/
     .Z5520,Z1551,Z0662,Z0871,Z6451,Z6163,Z5571,Z5020,Z1051,Z0163,
     .Z0471,Z1051,Z10A0,Z5071,Z6167,Z6851,Z5920,Z0871,Z0563,Z1451,
     .Z14A0,Z5471,Z2251,Z2280,Z4271,Z1363,Z1671,Z2780,Z5651,Z2187,
     .Z2170,Z2161,Z2270,Z3151,Z0584,Z5014,Z5051,Z4152,Z3363,Z3672,
     .Z4871,Z05C0,Z3266,Z3069,Z3070,Z4071,Z41C0,Z3041,Z1151,Z0262,
     .Z0481,Z2570,Z2541,Z0661,Z0771,Z1881,Z3970,Z5851,Z2070,Z2061,
     .Z2170,Z3061,Z2270,Z2267,Z2970,Z3267,Z3069,Z2142,Z2180,Z4182,
     .Z6361,Z6401,Z0561,Z0682,Z2880,Z6642,Z04C0,Z11A6,Z5126,Z1071,
     .Z2172,Z3363,Z3652,Z2851,Z2570,Z2561,Z3561,Z2670,Z2270,Z2261,
     .Z2370,Z3261,Z2171,Z05C0,Z6362,Z05C0,Z00C9,Z2170,Z2161,Z2270,
     .Z3161,Z2071,Z2651,Z2670,Z3671,Z4761,Z4851,Z3950,Z1761,Z1871,
     .Z00C9,Z3051,Z2161,Z2271,Z3070,Z4071,Z5161,Z5251,Z3370,Z00C0,
     .Z10B4,Z6414,Z3061,Z3262,Z3470,Z4482,Z6561,Z6742,Z0782,Z2960,
     .Z2570,Z2561,Z3561,Z2670,Z2270,Z2261,Z2370,Z3261,Z1386,Z3386,
     .Z05C0,Z07C0,Z2070,Z3082,Z2042,Z0264,Z0682,Z2870,Z5642,Z4271,
     .Z5363,Z2270,Z3271,Z4263,Z2251,Z1362,Z1571,Z2670,Z4551,Z3662,
     .Z04C0,Z06C0,Z2662,Z4663/
      CALL CSET(62,ICHAR,IPOS,IVEC)
      RETURN
      END
```

1.2 GEODYN DATA HANDLING SUPPORT PROGRAMS

There are five data handling programs used by the GEODYN program: DODS SORT-MERGE, GEOS SORT-MERGE, EPHEMERIS TAPE GENERATOR, ORB1 CONVERSION and TDIF TABLE GENERATOR.

DODS SORT-MERGE sorts and merges DODS formatted data from two tapes onto one tape. The data can be from any number of satellites. GEOS SORT-MERGE performs the same task; however, data from only one satellite should be used. EPHEMERIS TAPE GENERATOR generates various ephemerides by precessing and nutating the values found on the JPL ephemeris. ORB1 CONVERSION converts an IBM 360 system 9-track tape to the same format on a 7-track tape. TDIF TABLE GENERATOR generates tabular information for use with subroutine TDIF to compute time differences between systems A.1 and UT1.

Detailed descriptions of the formats of the data tapes are found in Appendix C of Volume III -- GEODYN SYSTEM OPERATIONS DESCRIPTION.

1.2.1 DODS SORT-MERGE

INTRODUCTION

The DODS SORT-MERGE program sorts data from DODS format data tapes by satellite identification numbers into chronological, station and then measurement type order, eliminating duplicate data records.

MAIN-DODS SRTMRG

DESCRIPTION

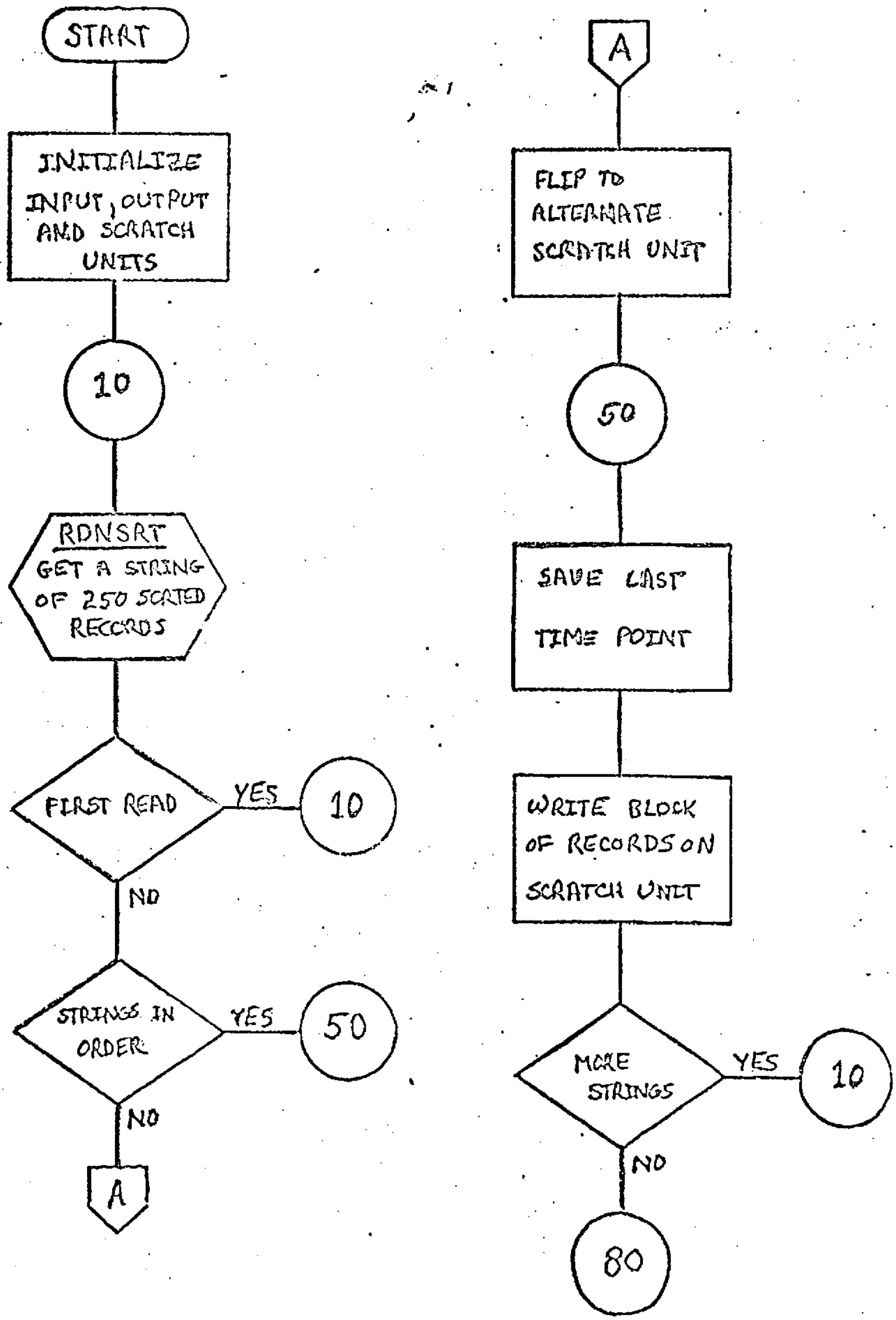
The main program SRTMRG sorts and merges blocks of 250 sorted records which are obtained from the subroutine RDNSRT. The blocks are sorted onto two scratch disk units, which are then merged and sorted again onto two alternate scratch disk units. The process is repeated until all the records are sorted by satellite identification number and in chronological order. Then the subroutine WRITE is called to write out the data records onto a tape.

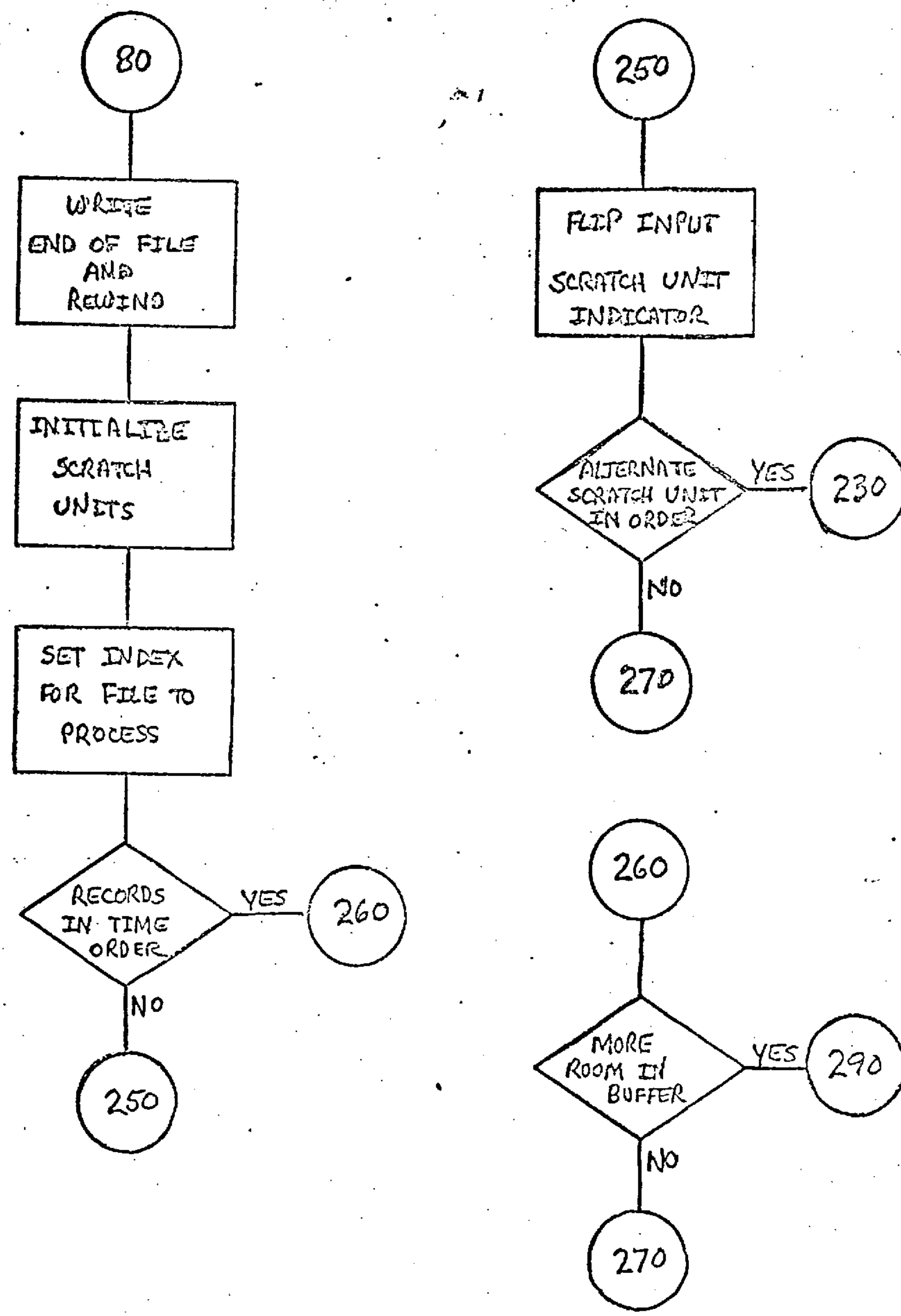
NAME	MAIN - DODS SRTMRG
PURPOSE	SORTS AND MERGES TWO INPUT DATA TAPES ONTO ONE TAPE
SUBROUTINES USED	RDNSRT WRITE
COMMON BLOCKS	OSORT UNITS
INPUT FILES	NONE
OUTPUT FILES	NONE
SCRATCH FILES	UNITS - 20,21,22,23
RESTRICTIONS	NONE
REFERENCES	NONE

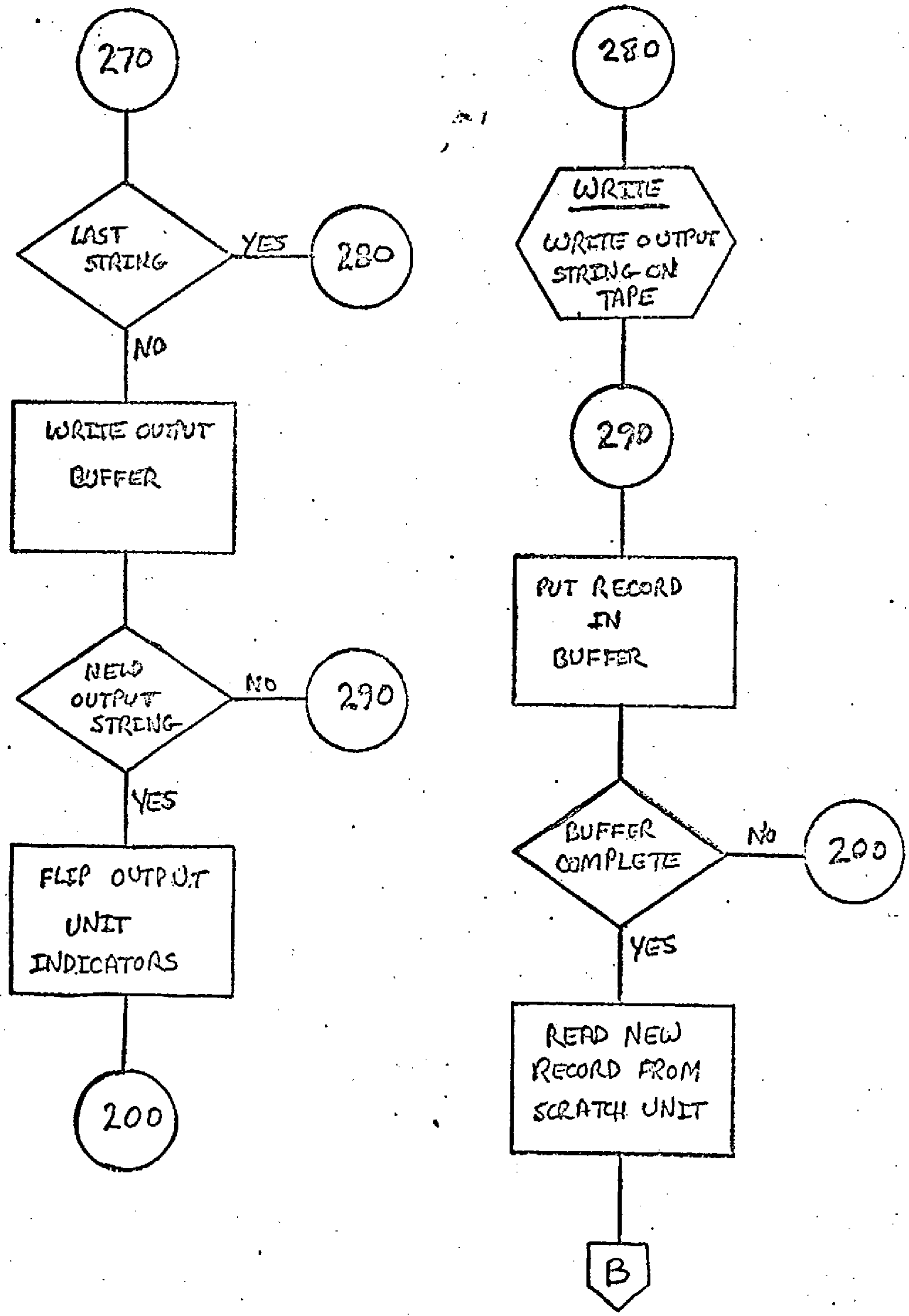
```
      COMMON/OSORT/NO,G,OB(14,250),OB1(14,250,2)
      COMMON/UNITS/NIN,NOUT,UNIT
      DOUBLE PRECISION OB,OB1,LAST(4),EOF
      INTEGER NN(2),NT(2),UNIT(2,2)
      INTEGER FLIP
      EQUIVALENCE (N1,NN(1)),(N2,NN(2))
      LOGICAL*1 FRSTIM,LSTPAS,REV,MERGE
      DATA FRSTIM,LSTPAS,REV/.TRUE.,2*.FALSE./,
     *     IOU,IOH,INH/2*1,2/,
     *     NSTRNG,EOF/1,99999999./
      FLIP(I)=MOD(I,2)+1
C INITIALIZE INPUT, OUTPUT, AND SCRATCH UNITS
      NIN=10
      NOUT=11
      K=19
      DO 5 I=1,2
      DO 5 J=1,2
      K=K+1
  5   UNIT(I,J)=K
C READ AND SORT 250 RECORDS
 10   CALL RDNSRT
      I1=1
      IF(NO.EQ.0) GO TO 80
      IF(FRSTIM) GO TO 50
C TEST ORDER OF STRINGS
 20   DO 30 I=1,4
      IF(LAST(I)-OB(I,I1))50,30,40
30    CONTINUE
      IF(I1.EQ.NO) GO TO 70
      I1=I1+1
      GO TO 20
C FLIP TO ALTERNATE SCRATCH UNIT
40    IOU=FLIP(IOU)
      NSTRNG=NSTRNG+1
```

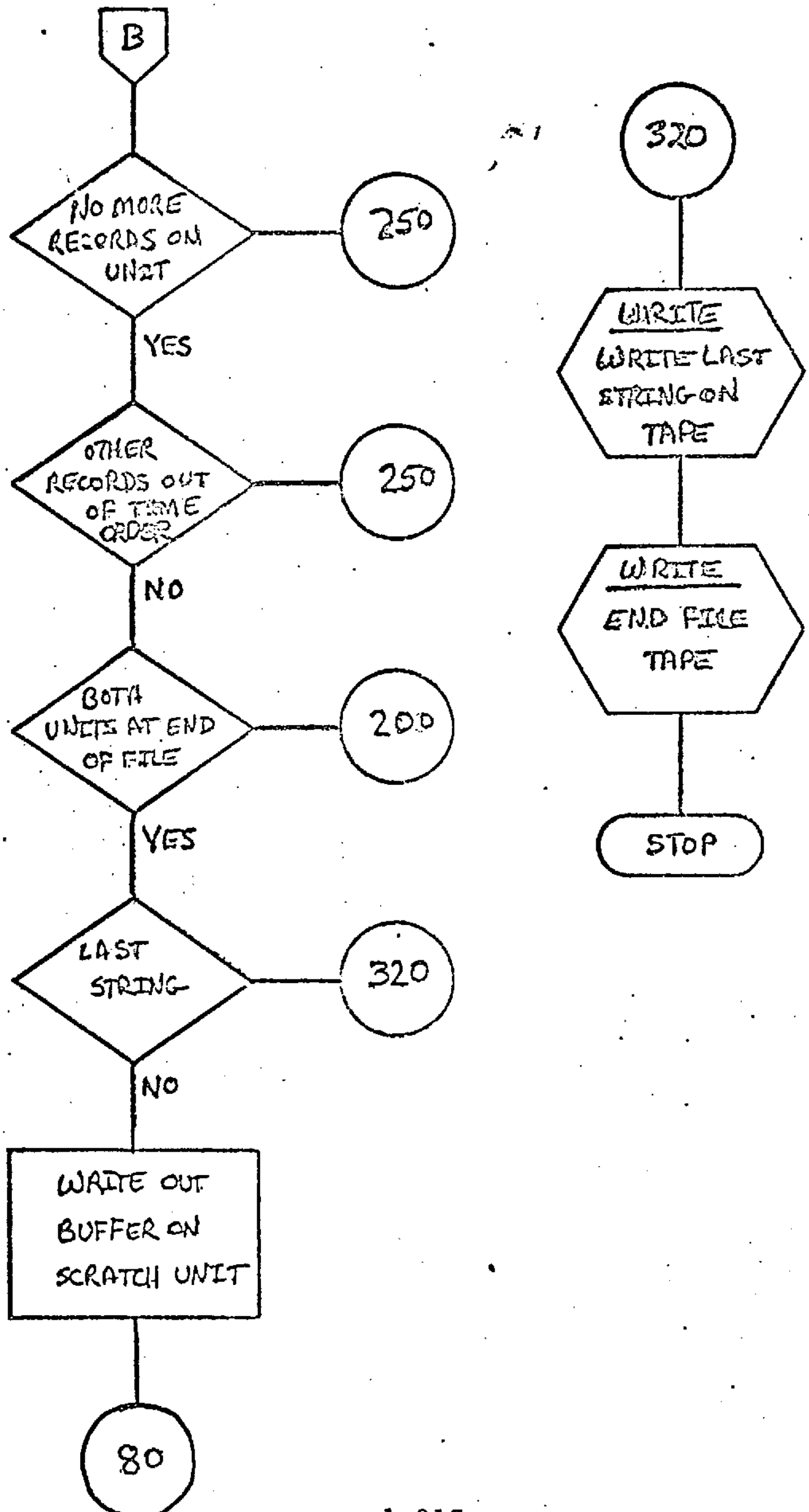
```
C SAVE LAST TIME POINT
 50     DO 60 I=1,4
 60     LAST(I)=CB(I,NC)
        M=NO-I1+1
        IU=UNIT(IOU,IOH)
C WRITE BLOCK OF RECORDS ON SCRATCH UNIT
        WRITE(IU)M,((CB(I,J),I=1,14),J=I1,NO)
        FRSTIM=.FALSE.
C TEST IF MORE STRINGS
 70     IF(NO.GE.250) GO TO 10
 80     M=0
        DO 90 K=1,2
        IU=UNIT(K,IOH)
C WRITE EOF AND REWIND
        WRITE(IU)M,((EOF,I=1,14),J=1,M)
        ENDFILE IU
        REWIND IU
C INITIALIZE SCRATCH UNITS
        IU=UNIT(K,INH)
 90     REWIND IU
        IOH=INH
        INH=FLIP(INH)
        IOU=1
        LSTPAS=NSTRNG.LE.2
        PRINT 1010,NSTRNG
        NSTRNG=1
        FRSTIM=.TRUE.
        DO 100 K=1,2
        IU=UNIT(K,INH)
C READ NEW STRING
        READ(IU)M,((OB1(I,J,K),I=1,14),J=1,M)
        NT(K)=M
 100    NN(K)=1
        MERGE=.FALSE.
        IF(NT(1)+NT(2).NE.0) GO TO 200
        PRINT 1000
        STOP
 200    IF(MERGE) GO TO 230
C SET INDEX FOR FILE TO PROCESS
        IT=1
        DO 210 I=1,4
        IF(OB1(I,N1,1)-OB1(I,N2,2))230,210,220
 210    CONTINUE
        GO TO 305
 220    IT=2
 230    N=NN(IT)
        IF(FRSTIM) GO TO 285
C TEST IF RECORDS IN TIME ORDER
        DO 240 I=1,4
        IF(OB1(I,N,IT)-OB(I,NO))250,240,260
 240    CONTINUE
        GO TO 305
 250    MERGE=.NOT.MERGE.AND.NT(1).NE.0.AND.NT(2).NE.0
C FLIP INPUT SCRATCH UNIT INDICATOR
        IT=FLIP(IT)
C TEST IF ALTERNATE SCRATCH UNIT IS IN ORDER
```

```
      IF(MERGE) GO TO 230
      REV=.TRUE.
      GO TO 270
C TEST IF BUFFER IS FULL
260   IF(NO.LT.250) GO TO 290
C TEST IF LAST STRING IS PROCESSED
270   IF(LSTPAS) GO TO 280
      IU=UNIT(ICU,ICH)
C WRITE OUTPUT BUFFER
      WRITE(IU)NO,((OB(I,J),I=1,14),J=1,NO)
      NO=0
C TEST IF NEW OUTPUT STRING
      IF(.NOT.REV) GO TO 290
      REV=.FALSE.
C FLIP OUTPUT INDICATOR
      IOU=FLIP(ICU)
      NSTRNG=NSTRNG+1
      FRSTIM=.TRUE.
      GO TO 200
C WRITE OUTPUT STRING ON TAPE
 280  CALL WRITE(.FALSE.)
285   NO=0
290   NO=NO+1
      FRSTIM=.FALSE.
      DO 300 I=1,14
C PUT RECORD IN BUFFER
300   OB(I,NO)=OB1(I,N,IT)
305   IF(NN(IT).EQ.NT(IT)) GO TO 310
      NN(IT)=NN(IT)+1
      GO TO 200
310   IU=UNIT(IT,INH)
C READ NEW RECORD FROM SCRATCH UNIT
      READ(IU)M,((OB1(I,J,IT),I=1,14),J=1,M)
      NT(IT)=M
      NN(IT)=1
      IF(MERGE.AND.NT(IT).EQ.0)GO TO 250
      IF(NT(1)+NT(2).NE.0) GO TO 200
      IF(LSTPAS) GO TO 320
      IU=UNIT(ICU,ICH)
C WRITE BUFFER ON SCRATCH UNIT
      WRITE(IU)NO,((OB(I,J),I=1,14),J=1,NO)
      GO TO 80
C WRITE OUTPUT TAPE
 320  CALL WRITE(.FALSE.)
      CALL WRITE (.TRUE.)
1000  FORMAT('1NO SORT INPUT')
1010  FORMAT(' ',I8,' STRINGS')
      STOP
      END
```


START
INITIALIZE INPUT, OUTPUT AND SCRATCH UNITS
10
RDNSRT
GET A STRING OF 250 SORTED RECORDS
FIRST READ
YES
10
NO
STRINGS IN ORDER
YES
50
NO
A
A
FLIP TO ALTERNATE SCRATCH UNIT
50
SAVE LAST TIME POINT
WRITE BLOCK OF RECORDS ON SCRATCH UNIT
MORE STRINGS
YES
10
NO
80


80
WRITE END OF FILE AND REWIND
INITIALIZE SCRATCH UNITS
SET INDEX FOR FILE TO PROCESS
RECORDS IN TIME ORDER
YES
260
NO
250
250
FLIP INPUT SCRATCH UNIT INDICATOR
ALTERNATE SCRATCH UNIT IN ORDER
YES
230
NO
270
260
MORE ROOM IN BUFFER
YES
290
NO
270


270
LAST STRING
YES
280
NO
WRITE OUTPUT BUFFER
NEW OUTPUT STRING
NO
290
YES
FLIP OUTPUT UNIT INDICATORS
200
280
WRITE
WRITE OUTPUT STRING ON TAPE
290
PUT RECORD IN BUFFER
BUFFER COMPLETE
NO
200
YES
READ NEW RECORD FROM SCRATCH UNIT
B


B
NO MORE RECORDS ON UNIT
250
YES
OTHER RECORDS OUT OF TIME ORDER
250
NO
BOTH UNITS AT END OF FILE
200
YES
LAST STRING
320
NO
WRITE OUT BUFFER ON SCRATCH UNIT
80
320
WRITE
WRITE LAST STRING ON TAPE
WRITE
END FILE TAPE
STOP

DODS RDNSRT

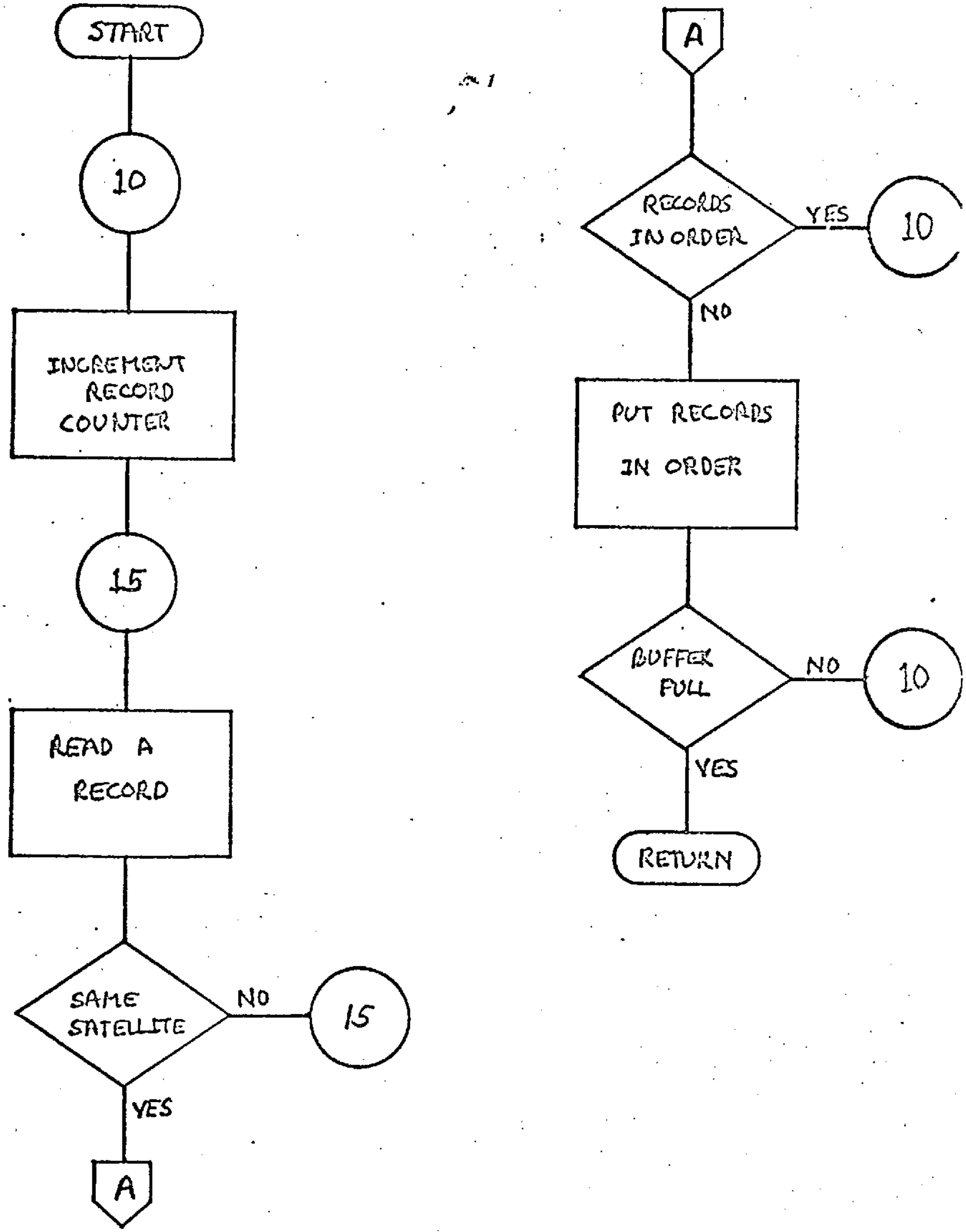
DESCRIPTION

RDNSRT reads a DODS data tape, sorting each record until a block of 250 records is filled and checking the satellite identification number. If the satellite identification numbers are not the same, a new record is read. When either a block of 250 is reached or the end of the tape is reached, control is returned to SRTMRG.

NAME	DODS RDNSRT
PURPOSE	READS AND SORTS 250 RECORDS INTO CORE
CALLING SEQUENCE	CALL RDNSRT
SUBROUTINES USED	NONE
COMMON BLOCKS	OSORT UNITS
INPUT FILE	DODS INPUT TAPE
OUTPUT FILES	NONE
RESTRICTIONS	TAPE MUST BE IN DODS FORMAT
REFERENCES	NONE

```
      SUBROUTINE RDNSRT
      COMMON/OSORT/N,G,OB(14,250)
      COMMON/UNITS/NIN,NOUT,SCR(4)
      DOUBLE PRECISION OB,SAVE,D
      INTEGER*2 ID(4)
      EQUIVALENCE (D,ID)
      N=0
10    N=N+1
C READ A RECORD
15    READ(NIN,ERR=100,END=80)OB(2,N),OB(3,N),(OB(I,N),I=5,8),ISATID,
     *                         (OB(I,N),I=9,14)
      IF(OB(2,N).LT.0.) GO TO 15
      OB(1,N)=ISATID
      D=OB(13,N)
      OB(4,N)=ID(3)
C IF FIRST READ, GO READ ANOTHER
20    IF(N.LT.2) GO TO 10
      DO 30 J1=2,N
      J=N+2-J1
C CHECK TIME ORDER OF RECORDS AND SATELLITE ID NUMBER
      DO 25 K=1,4
      IF(OB(K,N)-OB(K,J-1))30,25,40
25    CONTINUE
      GO TO 15
30    CONTINUE
      J=1
40    IF(J.EQ.N) GO TO 70
      J1=N-J
C ARRANGE RECORDS IN ORDER
      DO 60 K=1,14
      SAVE=OB(K,N)
      DO 50 M1=1,J1
      M=N-M1
50    OB(K,M+1)=OB(K,M)
```

```
60      CB(K,J)=SAVE
70      IF(N.LT.250) GO TO 10
C IF ARRAY IS FULL, RETURN
        RETURN
80      N=N-1
        RETURN
 100    READ(NIN,ERR=100,END=80)
        GO TO 15
        END
```


START
10
INCREMENT RECORD COUNTER
15
READ A RECORD
SAME SATELLITE
NO
15
YES
A
A
RECORDS IN ORDER
YES
10
NO
PUT RECORDS IN ORDER
BUFFER FULL
NO
10
YES
RETURN

DODS WRITE

DESCRIPTION

The subroutine WRITE is the output routine of the program. If WRITE is called with a false logical argument, it will write out a block of data records on the output tape. If the satellite identification number changes, it will write out a flagged record with the new identification number. If WRITE is called with a true logical argument, it will write an endfile on the output tape. A flowchart would be superfluous.

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

NAME	DODS WRITE
PURPOSE	WRITES DATA RECORDS AND ENDFILES TAPES
CALLING SEQUENCE	CALL WRITE(ENDSW)
SYMBOL TYPE	DESCRIPTION
ENDSW L	TRUE WILL ENDFILE WHEN TAPE IS COMPLETELY WRITTEN
SUBROUTINES USED	NONE
COMMON BLOCKS	OSORT UNITS
INPUT FILES	NONE
OUTPUT FILE	MAGNETIC TAPE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE WRITE(ENDSW)
      COMMON/OSORT/NT,G,OB(14,250)
      DOUBLE PRECISION OB
      COMMON/UNITS/NIN,NOUT,SCR(4)
      INTEGER ZERO/0/,JSATID/-1/
      REAL*8 FLAG/-1.D0/
      LOGICAL ENDSW
C IF REQUESTED, ENDFILE TAPE  ONLY
      IF(ENDSW) GO TO 20
      DO 10 N=1,NT
      ISATID=OB(1,N)+.5
      IF(ISATID.NE.JSATID)WRITE(NOUT)FLAG,(ZERO,I=1,10),ISATID,
     *                                     (ZERO,I=1,12)
      JSATID=ISATID
C IF SATELLITE ID NUMBERS ARE THE SAME, WRITE OUT THE RECORD
 10   WRITE(NOUT)OB(2,N),OB(3,N),(OB(I,N),I=5,8),ISATID,
     *           (OB(I,N),I=9,14)
      RETURN
20    ENDFILE NOUT
      REWIND NOUT
      RETURN
      END
```

1.2.2 GEOS SORT-MERGE

INTRODUCTION

The GEOS SORT-MERGE program sorts data from GEOS format data tapes into chronological, station, and then measurement type order, eliminating duplicate data records.

MAIN-GEOS SRTMRG

DESCRIPTION

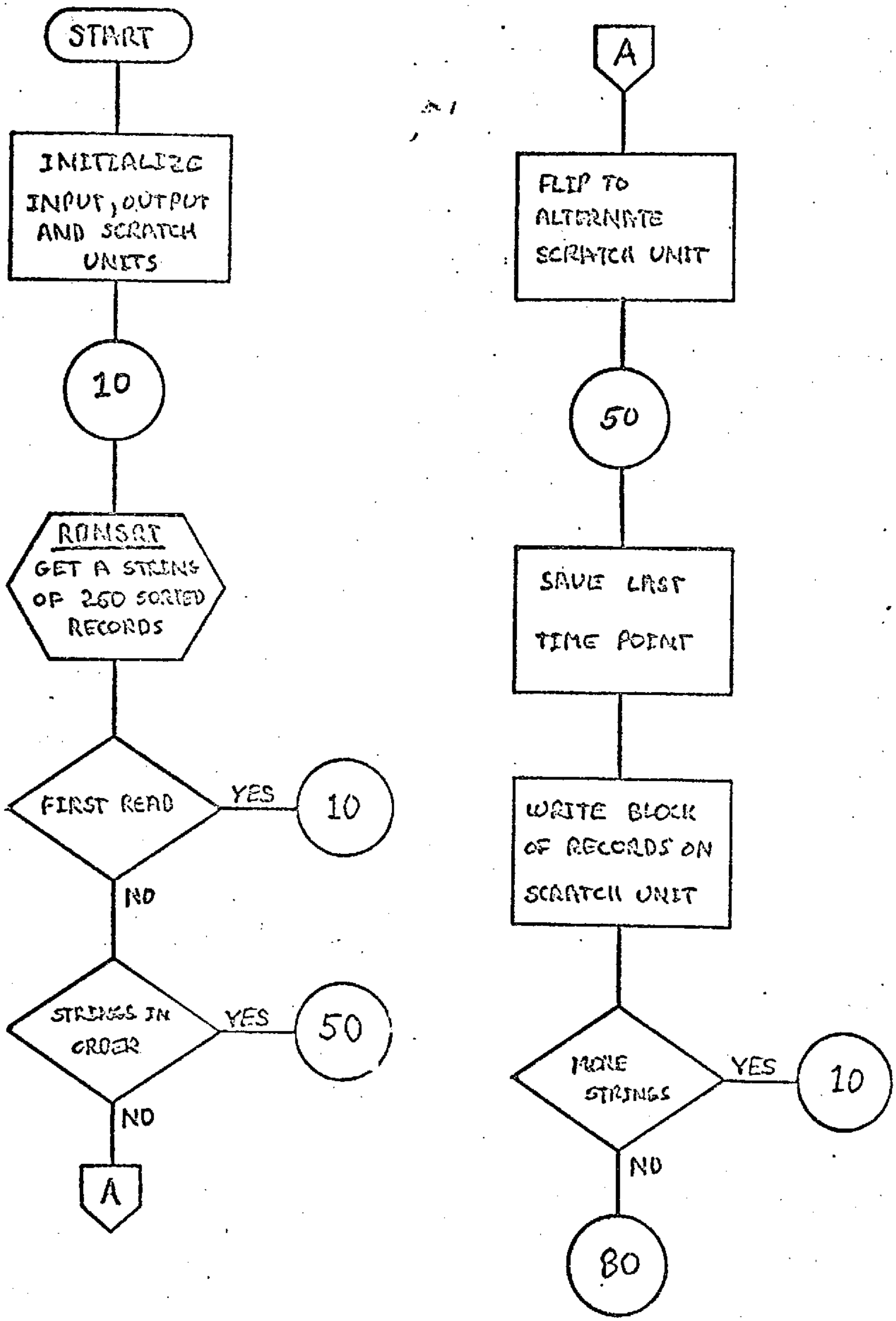
The main program SRTMRG sorts and merges blocks of 250 sorted records which are obtained from the subroutine RDNSRT. The blocks are sorted onto two scratch disk units which are then merged and sorted again onto two alternate scratch disk units. The process is repeated until all the records are in chronological order. This program does not sort by satellite identification number. Then the subroutine WRITE is called to write out the data records onto a tape.

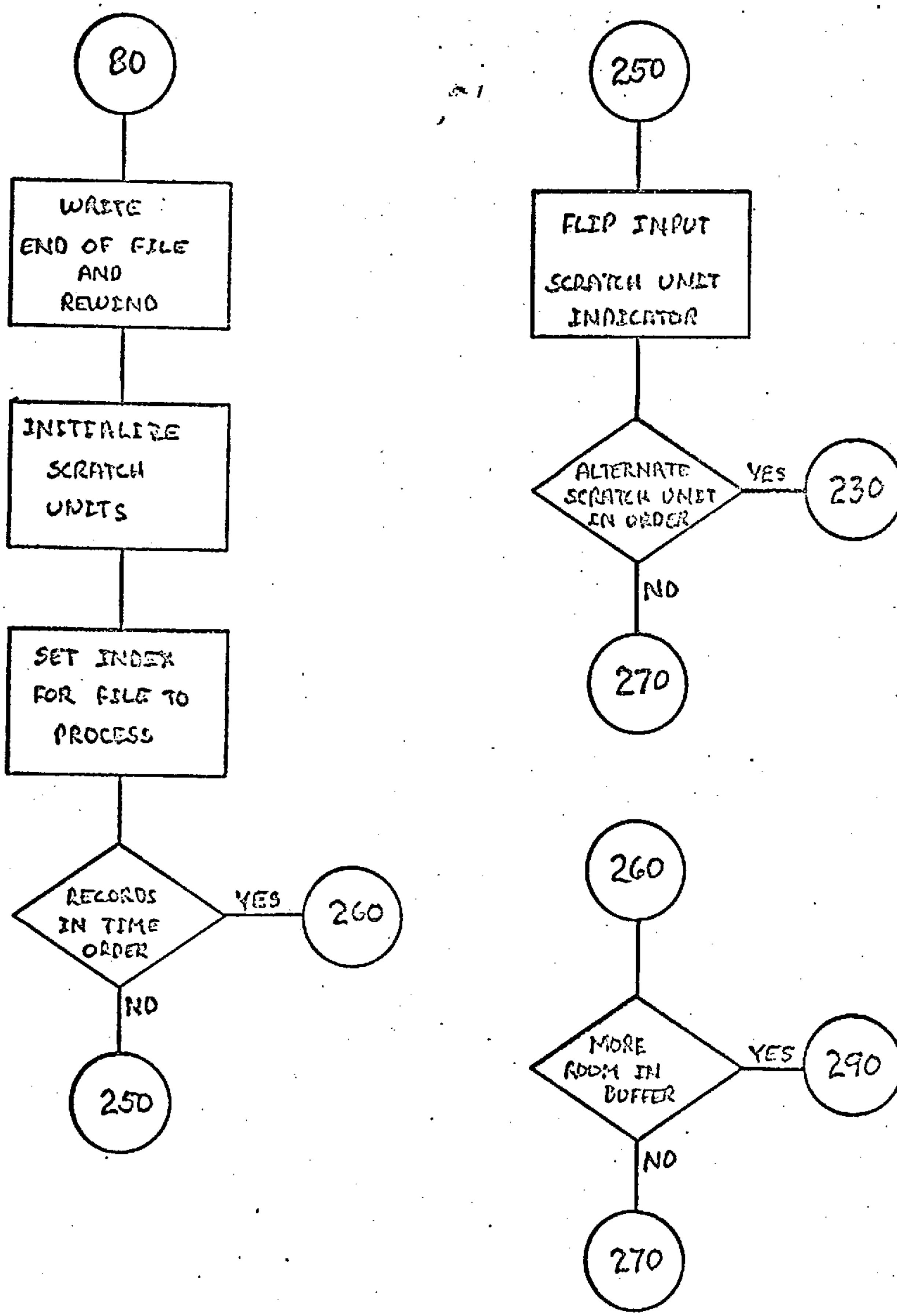
NAME	MAIN - GEOS SRTMRG
PURPOSE	SORTS AND MERGES TWO INPUT DATA TAPES ONTO ONE TAPE
SUBROUTINES USED	RDNSRT WRITE
COMMON BLOCKS	OSORT UNITS
INPUT FILES	NONE
OUTPUT FILES	NONE
SCRATCH FILES	UNITS - 20,21,22,23
RESTRICTIONS	NONE
REFERENCES	NONE

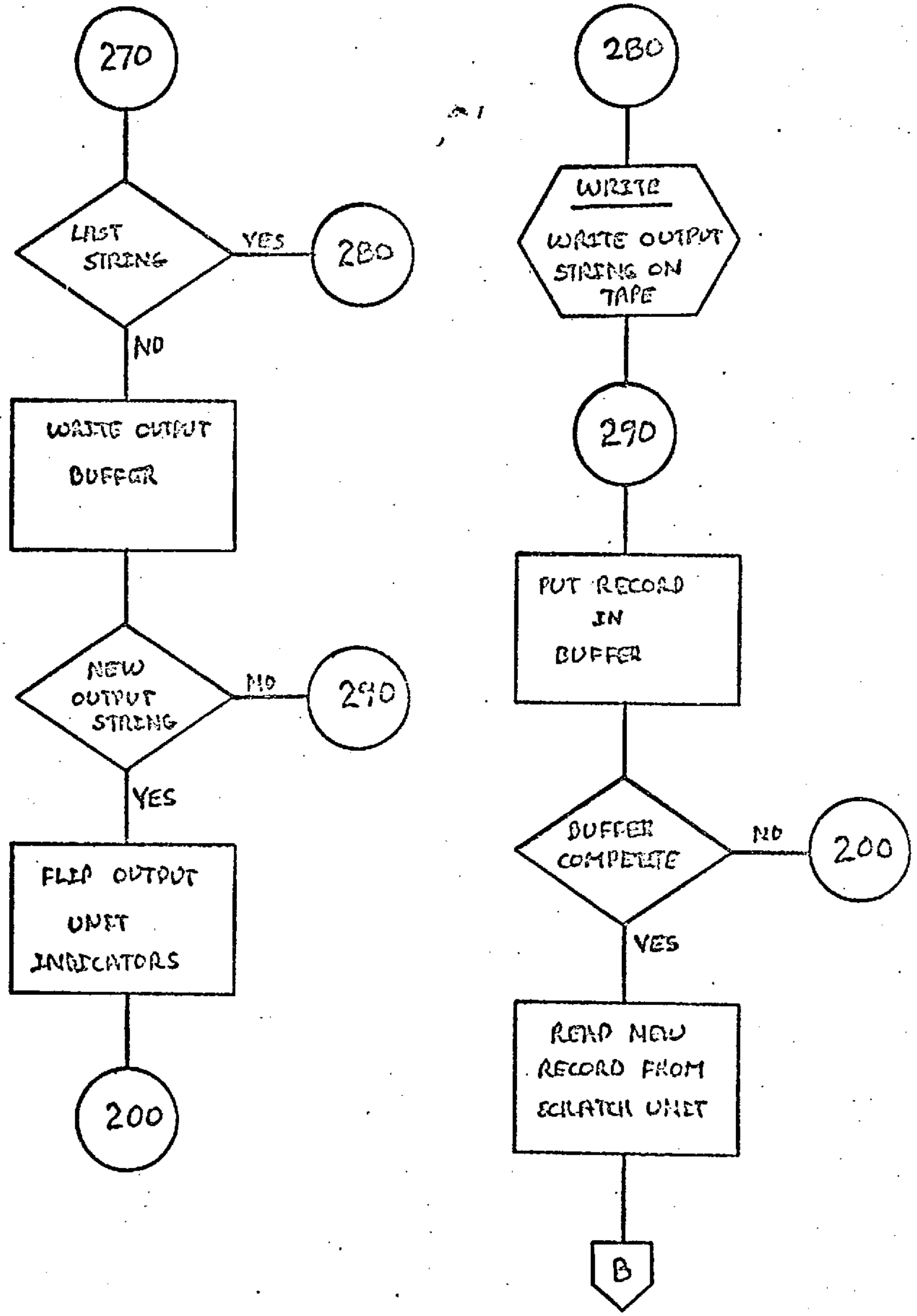
```
      COMMON/OSORT/NO,OB(19,250),OB1(19,250,2)
      COMMON/UNITS/NIN,NOUT,UNIT
      INTEGER OB,OB1,NN(2),NT(2),LAST(6),UNIT(2,2),EOF
      INTEGER FLIP
      EQUIVALENCE (N1,NN(1)),(N2,NN(2))
      LOGICAL*1 FRSTIM,LSTPAS,REV,MERGE
      DATA FRSTIM,LSTPAS,REV/.TRUE.,2*.FALSE./,
     *     IOU,IOH,INH/2*1,2/,
     *     NSTRNG,EOF/1,9999999/
      FLIP(I)=MOD(I,2)+1
C INITIALIZE INPUT, OUTPUT, AND SCRATCH UNITS
      NIN=10
      NOUT=11
 2    K=19
      DO 5 I=1,2
      DO 5 J=1,2
      K=K+1
      REWIND K
 5    UNIT(I,J)=K
      REWIND NIN
C READ AND SORT 250 RECORDS
 10   CALL RDNSRT
      I1=1
      IF(NO.EQ.0) GO TO 80
      IF(FRSTIM) GO TO 50
C TEST ORDER OF STRINGS
20    DO 30 I=1,3
      IF(LAST(I)-OB(I,I1))50,30,40
30    CONTINUE
      IF(I1.EQ.NO) GO TO 70
      I1=I1+1
      GO TO 20
C FLIP TO ALTERNATE SCRATCH UNIT
40    IOU=FLIP(IOU)
```

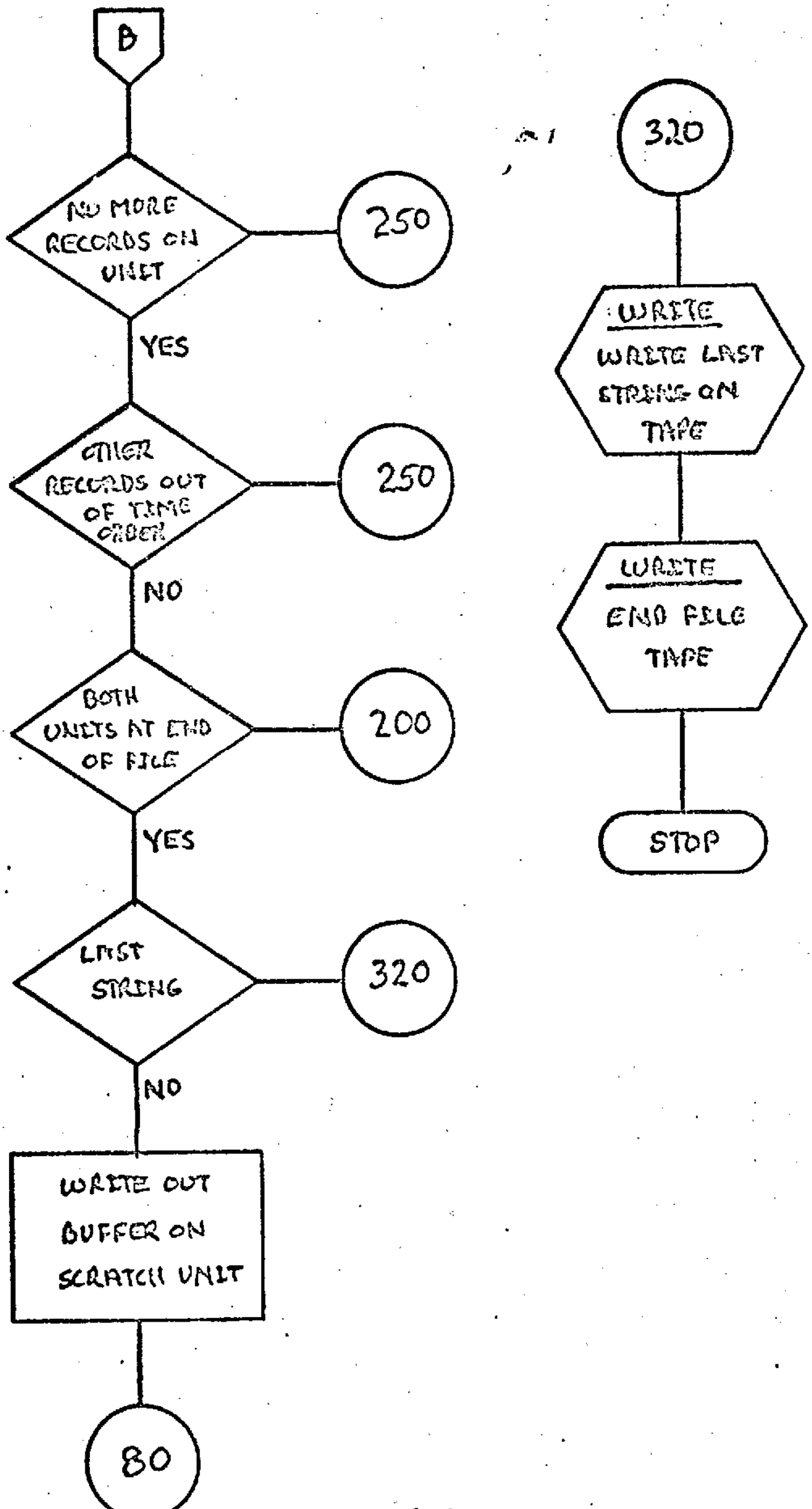
```
      NSTRNG=NSTRNG+1
50    DO 60 I=1,3
C SAVE LAST TIME POINT
60    LAST(I)=OB(I,NO)
      M=NO-I1+1
      IU=UNIT(IOU,IOH)
C WRITE BLOCK OF RECORDS ON SCRATCH UNIT
      WRITE(IU)M,((OB(I,J),I=1,19),J=I1,NO)
      FRSTIM=.FALSE.
C TEST IF MORE STRINGS
70    IF(NO.GE.250) GO TO 10
80    M=0
      DO 90 K=1,2
      IU=UNIT(K,IOH)
C WRITE EOF AND REWIND
      WRITE(IU)M,((EOF,I=1,19),J=1,M)
      ENDFILE IU
      REWIND IU
C INITIALIZE SCRATCH UNITS
      IU=UNIT(K,INH)
90    REWIND IU
      IOH=INH
      INH=FLIP(INH)
       IOU=1
      LSTPAS=NSTRNG.LE.2
      PRINT 777,NSTRNG
 777  FORMAT(' *** NUMBER OF STRINGS=',I4)
      NSTRNG=1
      FRSTIM=.TRUE.
      DO 100 K=1,2
      IU=UNIT(K,INH)
C READ NEW STRING
      READ(IU)M,((OB1(I,J,K),I=1,19),J=1,M)
      NT(K)=M
100   NN(K)=1
      MERGE=.FALSE.
      IF(NT(1)+NT(2).NE.0) GO TO 200
      PRINT 1000
      STOP
200   IF(MERGE) GO TO 230
C SET INDEX FOR FILE TO PROCESS
      IT=1
      DO 210 I=1,3
      IF(OB1(I,N1,1)-OB1(I,N2,2))230,210,220
210   CONTINUE
      GO TO 305
220   IT=2
230   N=NN(IT)
      IF(FRSTIM) GO TO 285
C TEST IF RECORDS IN TIME ORDER
      DO 240 I=1,3
      IF(OB1(I,N,IT)-OB(I,NO))250,240,260
240   CONTINUE
      GO TO 305
250   MERGE=.NOT.MERGE.AND.NT(1).NE.0.AND.NT(2).NE.0
C FLIP INPUT SCRATCH UNIT INDICATOR
```

```
      IT=FLIP(IT)
C TEST IF ALTERNATE SCRATCH UNIT IS IN ORDER
      IF(MERGE) GO TO 230
      REV=.TRUE.
      GO TO 270
C TEST IF BUFFER IS FULL
260   IF(NO.LT.250) GO TO 290
C TEST IF LAST STRING IS PROCESSED
270   IF(LSTPAS) GO TO 280
      IU=UNIT(ICU,ICH)
C WRITE OUTPUT BUFFER
      WRITE(IU)NO,((OB(I,J),I=1,19),J=1,NO)
      NO=0
C TEST IF NEW OUTPUT STRING
      IF(.NOT.REV) GO TO 290
      REV=.FALSE.
C FLIP OUTPUT INDICATOR
      IOU=FLIP(IOU)
      NSTRNG=NSTRNG+1
      FRSTIM=.TRUE.
      GO TO 200
C WRITE OUTPUT STRING ON TAPE
280   CALL WRITE(.FALSE.)
285   NO=0
290   NO=NO+1
      FRSTIM=.FALSE.
C PUT RECORD IN BUFFER
      DO 300 I=1,19
300   OB(I,NO)=OB1(I,N,IT)
305   IF(NN(IT).EQ.NT(IT)) GO TO 310
      NN(IT)=NN(IT)+1
      GO TO 200
310   IU=UNIT(IT,INH)
C READ NEW RECORD FROM SCRATCH UNIT
      READ(IU)M,((OB1(I,J,IT),I=1,19),J=1,M)
      NT(IT)=M
      NN(IT)=1
      IF(MERGE.AND.NT(IT).EQ.0) GO TO 250
      IF(NT(1)+NT(2).NE.0) GO TO 200
      IF(LSTPAS) GO TO 320
      IU=UNIT(ICU,ICH)
C WRITE BUFFER ON SCRATCH UNIT
      WRITE(IU)NO,((OB(I,J),I=1,19),J=1,NO)
      GO TO 80
C WRITE OUTPUT TAPE
 320  CALL WRITE(.FALSE.)
      CALL WRITE(.TRUE.)
1000  FORMAT('1NO SORT INPUT')
      STOP
      END
```


START
INITIALIZE INPUT, OUTPUT AND SCRATCH UNITS
10
RDNSRT
GET A STRING OF 260 SORTED RECORDS
FIRST READ
YES
10
NO
STRINGS IN ORDER
YES
50
NO
A
A
FLIP TO ALTERNATE SCRATCH UNIT
50
SAVE LAST TIME POINT
WRITE BLOCK OF RECORDS ON SCRATCH UNIT
MORE STRINGS
YES
10
NO
80


80
WRITE
END OF FILE
AND
REWIND
INITIALIZE
SCRATCH
UNITS
SET INDEX
FOR FILE TO
PROCESS
RECORDS
IN TIME
ORDER
YES
260
NO
250
250
FLIP INPUT
SCRATCH UNIT
INDICATOR
ALTERNATE
SCRATCH UNIT
IN ORDER
YES
230
NO
270
260
MORE
ROOM IN
BUFFER
YES
290
NO
270


270
LAST STRING
YES
280
NO
WRITE OUTPUT BUFFER
NEW OUTPUT STRING
NO
290
YES
FLIP OUTPUT UNIT INDICATORS
200
280
WRITE
WRITE OUTPUT STRING ON TAPE
290
PUT RECORD IN BUFFER
BUFFER COMPLETE
NO
200
YES
READ NEW RECORD FROM SCRATCH UNIT
B


B
NO MORE RECORDS ON UNIT
250
YES
OTHER RECORDS OUT OF TIME ORDER
250
NO
BOTH UNITS AT END OF FILE
200
YES
LAST STRING
320
NO
WRITE OUT BUFFER ON SCRATCH UNIT
80
320
WRITE
WRITE LAST STRING ON TAPE
WRITE
END FILE TAPE
STOP

GEOS RDNSRT

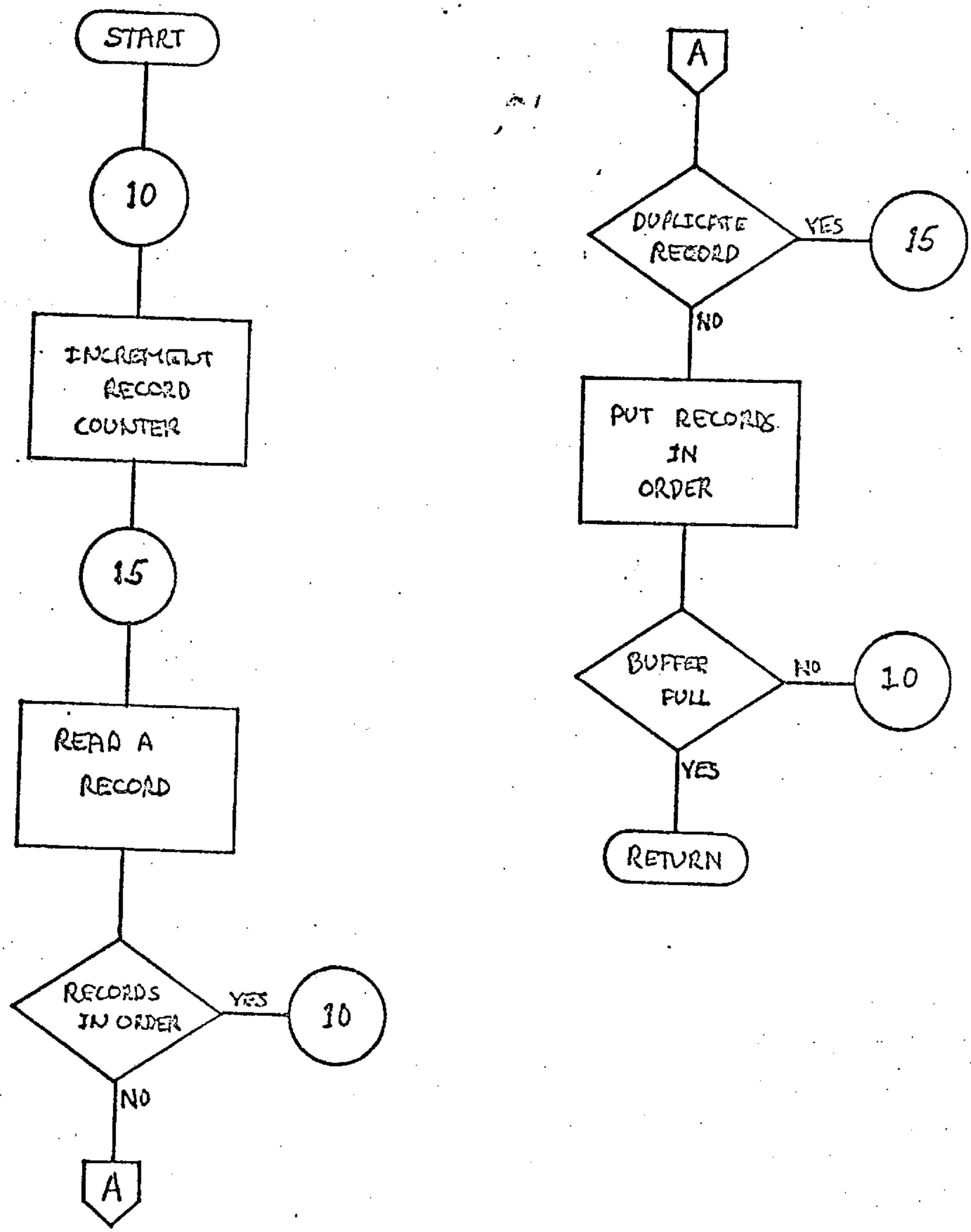
DESCRIPTION

RDNSRT reads a GEOS data tape, sorting each record into a block of 250 records. When either a block is full or the end of the tape is reached, control is returned to SRTMRG.

NAME	GEOS RDNSRT
PURPOSE	TO READ AND SORT 250 RECORDS INTO CORE
CALLING SEQUENCE	CALL RDNSRT
SUBROUTINES USED	NONE
COMMON BLOCKS	OSORT UNITS
INPUT FILE	GEOS INPUT TAPE
OUTPUT FILES	NONE
RESTRICTIONS	TAPE MUST BE IN GEOS FORMAT AND CONTAIN DATA FROM THE SAME SATELLITE
REFERENCES	NONE

```
      SUBROUTINE RDNSRT
      COMMON/OSORT/N,OB(19,250)
      COMMON/UNITS/NIN,NOUT,SCR(4)
      INTEGER OB,SAVE
      N=0
10    N=N+1
C READ A RECORD
15    READ(NIN,10000,END=80,ERR=15)OB(4,N),OB(5,N),MTYPE,ID,OB(6,N),
     *            OB(7,N),ISTANO,OB(1,N),OB(2,N),(OB(I,N),I=8,19)
      OB(3,N)=ISTANO*100+MTYPE*10+9-ID
C IF FIRST READ, GO READ ANOTHER
20    IF(N.LT.2) GO TO 10
      DO 30 J1=2,N
      J=N+2-J1
C CHECK TIME ORDER OF RECORDS
      DO 25 K=1,3
      IF(OB(K,N)-OB(K,J-1))30,25,40
25    CONTINUE
      GO TO 15
30    CONTINUE
      J=1
40    IF(J.EQ.N) GO TO 70
      J1=N-J
C ARRANGE RECORDS IN ORDER
      DO 60 K=1,19
      SAVE=OB(K,N)
      DO 50 M1=1,J1
      M=N-M1
50    OB(K,M+1)=OB(K,M)
60    OB(K,J)=SAVE
70    IF(N.LT.250) GO TO 10
C IF ARRAY IS FULL, RETURN
      RETURN
```

```
80    N=N-1
      RETURN
10000 FORMAT(A4,A2,2I1,A4,A1,I5,2I8,11A4,A2)
      END
```


START
10
INCREMENT RECORD COUNTER
15
READ A RECORD
RECORDS IN ORDER
YES
10
NO
A
A
DUPLICATE RECORD
YES
15
NO
PUT RECORDS IN ORDER
BUFFER FULL
NO
10
YES
RETURN

GEOS WRITE

DESCRIPTION

The subroutine WRITE is the output routine of the program. If WRITE is called with a false logical argument it will write out a block of data records on the output tape. If it is called with a true logical argument, it will write an endfile on the output tape. A flowchart would be superfluous.

NAME	GEOS WRITE
PURPOSE	WRITES DATA RECORDS AND ENDFILES TAPES
CALLING SEQUENCE	CALL WRITE(ENDSW)

SYMBOL	TYPE	DESCRIPTION
ENDSW	L	TRUE WILL ENDFILE WHEN TAPE IS COMPLETELY WRITTEN

SUBROUTINES USED	NONE
COMMON BLOCKS	OSORT UNITS
INPUT FILES	NONE
OUTPUT FILE	MAGNETIC TAPE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE WRITE(ENDSW)
      COMMON/OSORT/NT,OB(19,250)
      INTEGER OB
      COMMON/UNITS/NIN,NOUT,SCR(4)
      LOGICAL ENDSW
C IF REQUESTED, ENDFILE TAPE  ONLY
      IF(ENDSW) GO TO 20
      DO 12 N=1,NT
      ISTANO=OB(3,N)/100
      ITEMP=OB(3,N)-ISTANO*100
      MTYPE=ITEMP/10
      ID=9-(ITEMP-MTYPE*10)
C WRITE OUT THE RECORD
10    WRITE(NOUT,10000)OB(4,N),OB(5,N),MTYPE,ID,OB(6,N),OB(7,N),
     *            ISTANO,OB(1,N),OB(2,N),(OB(I,N),I=8,19)
 12   CONTINUE
      RETURN
20    ENDFILE NOUT
      REWIND NOUT
      RETURN
10000 FORMAT(A4,A2,2I1,A4,A1,I5,2I8,11A4,A2)
      END
```

1.2.3 EPHEMERIS TAPE GENERATOR

INTRODUCTION

The ephemeris tape contains the following ephemerides:

- geocentric lunar positions at half day intervals
- heliocentric positions of the earth-moon barycenter at 4-day intervals
- heliocentric positions of the planets, Venus, Mars, Jupiter and Saturn at 4-day intervals
- nutation in obliquity at half day intervals

The ephemerides are obtained by precessing and nutating to true of date coordinates the values found on the JPL planetary ephemeris tape.

The positions of the Earth-moon barycenter and the planets are heliocentric on the JPL tape; however, the moon is geocentric. Subroutine READE uses Everett's 5th order interpolation formula which is written as follows:

$$y(t_j+sh) \cong P(s) \equiv y_j F_o(1-s)+d_j^2 F_2(1-s)$$
$$+ d_j^4 F_4(1-s)$$
$$+ y_{j+1} F_o(s)+d_{j+1}^2 F_2(s)$$
$$+ d_{j+1}^4 F_4(s)$$

where

$F_o(s) = s$

$F_2(s) = [(s-1)\ (s)\ (s+1)]/6$

$F_4(s) = [(s-2)\ (s-1)\ (s)\ (s+1)\ (s+2)]/120$

d_j^2, d_j^4 are the second and fourth modified central differences contained on the JPL tape.

$y_j, j=1,2,\ldots,n$ denotes successive tabular values of one of the quantities contained in the ephemeris.

t_j denotes the corresponding time points.

$h = t_{j+1}-t_j$

$s = \frac{t-t_j}{h}$, t is time at which information is requested.

All the coordinates are converted to geocentric positions and placed in common. The MAIN program, by calling subroutine EQUATR, precesses and nutates the coordinate system from Mean of 1950 to True of Date. The planets are reconverted by MAIN to heliocentric positions, while the moon remains geocentric. The sun is converted to Earth-moon barycentered positions. The second and fourth modified central differences are then recomputed. These differences and the positions are written on the tape.

The modified second and fourth differences for Everett interpolation are computed as follows:

$$d_j^2 = \delta_j^2 + a_{26}\,\delta_j^6 + a_{28}\,\delta_j^8$$

$$d_j^4 = \delta_j^4 + a_{46}\,\delta_j^6 + a_{48}\,\delta_j^8$$

where

$$a_{26} = -0.013120 \qquad a_{28} = 0.004299$$

$$a_{46} = -0.278269 \qquad a_{48} = 0.068489$$

and the ordinary central differences are defined:

$$\delta_j^0 = y_j$$

$$\delta_{j+0.5}^1 = \delta_{j+1}^0 - \delta_j^0$$

$$\delta_j^2 = \delta_{j+0.5}^1 - \delta_{j-0.5}^1$$

$$\delta_{j+0.5}^3 = \delta_{j+1}^2 - \delta_j^2 \qquad \text{etc.}$$

SUBROUTINE CROSS REFERENCE CHART

CALLING ROUTINES

CALLED ROUTINES	MAIN	DATES	DIFF	DJUL	EQUATR	MATRIX	NUTATE	PRECES	READE	YMDAY
ADDYMD		○								
CLEAR	○									
DATES	○									
DIFF										○
EQN							○			
EQUATR	○									
GETTAP									○	
MULMAT							○	○		
NUTATE					○	○				
PRECES					○					
READE	○									
ROTMAT							○	○		
RYMDI			○							
YMDAY	○			○			○	○		

COMMON BLOCK CROSS REFERENCE

ROUTINES

COMMON BLOCKS	MAIN	GETTAP	READE
CETBL1	O		O
CETBL2	O	O	O
CETBL3	O	O	O
CETBL4	O		O
CETBL5			O
CETBL9		O	O
REC1		O	
REC2		O	
TAPE	O	O	

MAIN-EPHEM

DESCRIPTION

The MAIN program, EPHEM, reads the JPL ephemeris tape for positions of the Earth-moon barycenter, the planets and the moon. All the coordinates are converted to geocentric positions and placed in common. MAIN, by calling subroutine EQUATR, precesses and nutates the coordinate systems from Mean of 1950 to True of Date. The planets are reconverted by MAIN to heliocentric positions while the moon remains geocentric. The sun is converted to Earth-moon bary-centered positions. The second and fourth modified central differences are then recomputed. These differences and the positions are written on the tape.

NAME	MAIN - EPHEM
PURPOSE	GENERATES HELIOCENTRIC POSITIONS OF VENUS,MARS, JUPITER,SATURN AND THE EARTH-MOON BARYCENTER, GEOCENTRIC LUNAR POSITIONS AND NUTATIONS IN OBLIQUITY
SUBROUTINES USED	CLEAR DATES EQUATR READE TDIF YMDAY
COMMON BLOCKS	CETBL1 CETBL2 CETBL3 CETBL4 TAPE
INPUT FILES	5 - READER
OUTPUT FILES	6 - PRINTER
REFERENCES	JPL DEVELOPMENT EPHEMERIS NO. 19 TECHNICAL REPORT 32-1181 - C.J. DEVINE JPL, CALIF. INST. OF TECH., PASADENA,CALIF. NOV. 15,1967

```
      REAL*8 JD1,DJ,SEC1,DELJD,DELSEC,AU,REM,TPD,EMRAT,TABOUT
     1TB2,SUN,NUT,NUTATE,JED,TSEC,SOLARS(3,81,10),BASE,DJBASE,
     2BUF1(3,3,17),BUF3(3,3,3,5),DOUT,E,TAB3,BUFM(153),BUFP(135),
     3DAYEND,DAY,YMDAY,A(4),D6,D8
      EQUIVALENCE(BUFM(1),BUF1(1,1,1)),(BUF3(1,1,1,1),BUFP(1))
      REAL*8 FACTOR
      DATA FACTOR/1.2150373016452D-02/
      DOUBLE PRECISION TIMES(6)
      DATA ITIME/1/
      COMMON/TAPE/IN
      COMMON/CETBL1/AU,REM,TPD,EMRAT
      COMMON/CETBL2/ICW,NCENTR,IREQ(13)
      COMMON/CETBL3/TAB3(829),NUTAT(204),CKSUM
      COMMON/CETBL4/SUN(6,12),NUT(4)
      DATA NEQ/6/
      INTEGER IEQ(10)/11,10,2,4,5,6,7,8,9,1/
      REAL NUTAT
      LOGICAL TIN
      DIMENSION NUTATE(81),BUF2(3,17)
      INTEGER RECORD/0/
      INTEGER OUT/10/,IY/50/
      DATA NBODY,NPTS/10,16/
      DATA A/-1.312D-2,4.299D-3,-2.78269D-1,6.8489D-2/
      DATA JD1,SEC1,DELJD,DELSEC/2437608.5D0,0.0D0,0.5D0,0.0D0/
      DATA DJBASE,DAYEND/2433281.5D0,2440784.5D0/
      DATA TIN/.FALSE./
      E(X)=((.190037D-21*X-.021441D-15)*X-6.217959D-9)*X+.40931976D0
      READ 1003,TIMES
      JD1=TIMES(ITIME)
      DAYEND=TIMES(ITIME+1)
      IN=12
```

```
      ICW=1
      NCENTR=3
      AU=1.495979D11
      REV=6378149.5D0
      TPD=8.64D4
      TSEC=0
      EMRAT=81.302D0
      CALL CLEAR(IREQ,13,1)
      DO 605 I=1,NEQ
      I1=IEQ(I)
  605 IREQ(I1)=1
      IREQ(13)=1
      CALL READE(JD1,TSEC,IERR)
      JD1=TAB3(1)
      JED=JD1
C DAYS FROM 1900.0
      JD1=JD1-2415020.0D0
      DJ=JED+16.D0
C 1950.0
      BASE=YMDAY(500100,0,0.)
      DOUT=BASE+JED-DJBASE
C SET UP ARRAY FIRST TIME
      DO 606 I=1,81
      CALL READE(JED,TSEC,IERR)
C PRECESS AND NUTATE
      CALL EQUATR(SUN,BASE,TIN,SOLARS(1,I,1),DOUT,.TRUE.,IEQ,NEQ)
C SUBTRACT VECTOR TO SUN FROM PLANETS
      DO 900 J=3,NEQ
      DO 900 L=1,3
  900 SOLARS(L,I,J)=SOLARS(L,I,J)-SOLARS(L,I,2)
C SUBTRACT EARTH-MOON BARYCENTER FROM VECTOR TO SUN
      DO 810 J=1,3
      SOLARS(J,I,2)=SOLARS(J,I,2)-FACTOR*SOLARS(J,I,1)
  810 CONTINUE
      NUTATE(I)=DCOS(E(JD1)+NUT(2))*NUT(1)
      DOUT=DOUT+DELJD
      JD1=JD1+DELJD
      JED=JED+DELJD
  606 CONTINUE
      GO TO 610
C READ ONE RECORD
  620 CONTINUE
      DO 621 I=66,81
      CALL READE(JED,TSEC,IERR)
C PRECESS AND NUTATE
      CALL EQUATR(SUN,BASE,TIN,SOLARS(1,I,1),DOUT,.TRUE.,IEQ,NEQ)
C SUBTRACT VECTOR TO SUN FROM PLANETS
      DO 800 J=3,NEQ
      DO 800 L=1,3
  800 SOLARS(L,I,J)=SOLARS(L,I,J)-SOLARS(L,I,2)
C SUBTRACT EARTH-MOON BARYCENTER FROM VECTOR TO SUN
      DO 910 J=1,3
      SOLARS(J,I,2)=SOLARS(J,I,2)-FACTOR*SOLARS(J,I,1)
  910 CONTINUE
      NUTATE(I)=DCOS(E(JD1)+NUT(2))*NUT(1)
      DOUT=DOUT+DELJD
```

```
      JD1=JD1+DELJD
      JED=JED+DELJD
  621 CONTINUE
C COMPUTE DIFFERENCES
  610 DO 608 I=1,17
      II=I+32
      BUF2(1,I)=NUTATE(II)
      BUF2(2,I)=NUTATE(II-1)+NUTATE(II+1)-2.D0*NUTATE(II)
      BUF2(3,I)=NUTATE(II+2)+NUTATE(II-2)-4.D0*NUTATE(II+1)
     *-4.D0*NUTATE(II-1)+6.D0*NUTATE(II)
      D6=NUTATE(II-3)+NUTATE(II+3)-6.D0*(NUTATE(II-2)+NUTATE(II+2))
     *+15.D0*(NUTATE(II-1)+NUTATE(II+1))-20.D0*NUTATE(II)
      D8=NUTATE(II-4)+NUTATE(II+4)-8.D0*(NUTATE(II-3)+NUTATE(II+3))
     *+28.D0*NUTATE(II-2)+28.D0*NUTATE(II+2)-56.D0*(NUTATE(II-1)+
     *NUTATE(II+1))+70.D0*NUTATE(II)
      BUF2(2,I)=BUF2(2,I)+A(1)*D6+A(2)*D8
      BUF2(3,I)=BUF2(3,I)+A(3)*D6+A(4)*D8
      DO 608 J=1,3
      BUF1(1,J,I)=SOLARS(J,II,1)
      BUF1(2,J,I)=SOLARS(J,II-1,1)+SOLARS(J,II+1,1)-2.D0*SOLARS(J,II,1)
      BUF1(3,J,I)=SOLARS(J,II+2,1)+SOLARS(J,II-2,1)-4.D0*SOLARS(J,II-1,1
     *)-4.D0*SOLARS(J,II+1,1)+6.D0*SOLARS(J,II,1)
      D6=SOLARS(J,II-3,1)+SOLARS(J,II+3,1)-6.D0*(SOLARS(J,II-2,1)+
     *SOLARS(J,II+2,1))+15.D0*(SOLARS(J,II-1,1)+SOLARS(J,II+1,1))
     *-20.D0*SOLARS(J,II,1)
      D8=SOLARS(J,II-4,1)+SOLARS(J,II+4,1)-8.D0*(SOLARS(J,II-3,1)+
     *SOLARS(J,II+3,1))+28.D0*(SOLARS(J,II-2,1)+SOLARS(J,II+2,1))-56.D0
     **(SOLARS(J,II-1,1)+SOLARS(J,II+1,1))+70.D0*SOLARS(J,II,1)
      BUF1(2,J,I)=BUF1(2,J,I)+A(1)*D6+A(2)*D8
      BUF1(3,J,I)=BUF1(3,J,I)+A(3)*D6+A(4)*D8
  608 CONTINUE
      DO 611 I=1,3
      II=(I-1)*8+33
      DO 611 K=2,NEQ
      DO 611 J=1,3
      BUF3(1,J,I,K-1)=SOLARS(J,II,K)
      BUF3(2,J,I,K-1)=SOLARS(J,II-8,K)+SOLARS(J,II+8,K)-2.D0*
     *SOLARS(J,II,K)
      BUF3(3,J,I,K-1)=SOLARS(J,II+16,K)+SOLARS(J,II-16,K)-4.D0*
     *(SOLARS(J,II-8,K)+SOLARS(J,II+8,K))+6.D0*SOLARS(J,II,K)
      D6=SOLARS(J,II-24,K)+SOLARS(J,II+24,K)-6.D0*(SOLARS(J,II-16,K)+
     *SOLARS(J,II+16,K))+15.D0*(SOLARS(J,II-8,K)+SOLARS(J,II+8,K))
     *-20.D0*SOLARS(J,II,K)
      D8=SOLARS(J,II-32,K)+SOLARS(J,II+32,K)-8.D0*(SOLARS(J,II-24,K)+
     *SOLARS(J,II+24,K))+28.D0*(SOLARS(J,II-16,K)+SOLARS(J,II+16,K))
     *-56.D0*(SOLARS(J,II-8,K)+SOLARS(J,II+8,K))+70.D0*SOLARS(J,II,K)
      BUF3(2,J,I,K-1)=BUF3(2,J,I,K-1)+A(1)*D6+A(2)*D8
      BUF3(3,J,I,K-1)=BUF3(3,J,I,K-1)+A(3)*D6+A(4)*D8
  611 CONTINUE
C WRITE OUTPUT
      DAY=DJ-DJBASE+BASE
      DAY=DAY/8.64E4-32.15/8.64E4
      CALL DATES(DAY,IY,IYMD,IHM,SEC)
      WRITE(OUT) IYMD,IHM,SEC,BUF2,(BUFP(I),I=1,27)
      WRITE(OUT)(BUFM(I),I=1,51)
      WRITE(OUT)(BUFM(I),I=52,102)
```

```
      WRITE(OUT)(BUFM(I),I=103,153)
      WRITE(OUT)(BUFP(I),I=28,81)
      WRITE(OUT)(BUFP(I),I=82,135)
      DJ=DJ+8.D0
      RECORD=RECORD+1
      IF(RECORD.LT.11) WRITE(6,1001) IYMD,IHM,SEC,DJ,BUF1,BUF2,BUF3
      IF(RECORD.GT.10) WRITE(6,1002) IYMD,IHM,SEC,DJ,RECORD
C SHIFT BACK VALUES
      DO 630 I=1,65
      II=I+16
      NUTATE(I)=NUTATE(II)
      DO 630 J=1,3
      DO 630 K=1,NEQ
      SOLAR5(J,I,K)=SOLAR5(J,II,K)
  630 CONTINUE
      IF(ICW.EQ.3) GO TO 100
         IF(DJ.GE.DAYEND) GO TO 100
      GO TO 620
C TEST TO SEE IF LAST TAPE HAS BEEN READ
  100 IF(IN.EQ.14) GO TO 200
      ITIME=ITIME+2
      IF(TIMES(ITIME).LE.0.) GO TO 200
      DAYEND=TIMES(ITIME+1)
      REWIND IN
      IN=IN+1
      ICW=1
      GO TO 620
  200 WRITE(6,1000) DJ
      IYMD=0
      WRITE(OUT) IYMD,IHM,SEC,BUF2,(BUFP(I),I=1,27)
      WRITE(OUT)(BUFM(I),I=1,51)
      WRITE(OUT)(BUFM(I),I=52,102)
      WRITE(OUT)(BUFM(I),I=103,153)
      WRITE(OUT)(BUFP(I),I=28,81)
      WRITE(OUT)(BUFP(I),I=82,135)
      END FILE OUT
      REWIND OUT
      REWIND IN
      STOP
 1000 FORMAT('1EPHEMERIS TAPE GENERATION COMPLETE'/'0LAST DATE',G25.16)
 1001 FORMAT(1H1/61(5G25.16/)////)
 1002 FORMAT(1H0,5G25.16)
 1003 FORMAT(6D12.6)
      END
```

ADDYMD

DESCRIPTION

(See GEODYN)

CLEAR

DESCRIPTION

(See GEODYN)

DATES

DESCRIPTION

DATES converts a number of days elapsed from Jan 0.0 of the arc reference year into a three-word date of the form: YYMMDD, HHMM, SEC.

NAME DATES

PURPOSE CONVERTS DAYS ELAPSED FROM JAN 0.0 OF THE ARC REFERENCE YEAR INTO A 3-WORD DATE OF THE FORM: YYMMDD, HHMM, SEC

CALLING SEQUENCE CALL DATES(DAYNR,IY,IYMD,IHM,SEC)

SYMBOL	TYPE	DESCRIPTION
DAYNR	DP	INPUT - DAYS ELAPSED FROM JAN 0.0 OF THE REFERENCE YEAR
IY	I	INPUT - 1950 REPRESENTED BY THE LAST TWO DIGITS IN THE FORM YY
IYMD	I	OUTPUT - YEAR, MONTH, DAY IN THE FORM YYMMDD
IHM	I	OUTPUT - HOUR, MINUTES IN THE FORM HHMM
SEC	R	OUTPUT - SECONDS

SUBROUTINES USED ADDYMD

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
      SUBROUTINE DATES(DAYNR,IY,IYMD,IHM,SEC)
      DOUBLE PRECISION DAYNR,S,DAY
C     NUMBER OF DAYS FROM JAN 1 OF REFERENCE YEAR
      DAY=DAYNR+0.5D-4/8.64D4
      IDAY=DAY-1.
C NUMBER OF DAYS FROM JAN 1 OF THE REFERENCE YEAR
      IYMD=IY*10000+101
C CALCULATE YEAR,MONTH, DAY OF INTEREST
      CALL ADDYMD(IYMD,IDAY)
C CALCULATE THE NUMBER OF SECONDS REMAINING
      S=8.64D4*(DAY-FLOAT(IDAY+1))
      ISEC=S
C CONVERT TO HOUR,MINUTE FORMAT
      IHM=40*(ISEC/3600)+ISEC/60
C REMAINING SECONDS
      SEC=S-FLOAT(60*(ISEC/60))-0.5E-4
      RETURN
      END
```

DIFF

DESCRIPTION

DIFF calculates the difference in days and seconds between any two time points in the 20th century.

NAME	DIFF
PURPOSE	CALCULATES THE DIFFERENCE BETWEEN ANY TWO TIME POINTS IN THE 20TH CENTURY
CALLING SEQUENCE	CALL DIFF(IYMD1,IHMS1,IYMD2,IHMS2,IDAY,ISEC)

SYMBOL	TYPE	DESCRIPTION
IYMD1	I	INPUT - FIRST DATE IN THE FORM YYMMDD
IHMS1	I	INPUT - TIME ON IYMD1 IN THE FORM HHMMSS
IYMD2	I	INPUT - SECOND DATE IN THE FORM YYMMDD
IHMS2	I	INPUT - TIME ON IYMD2 IN THE FORM HHMMSS
IDAY	I	OUTPUT - ELAPSED FULL DAYS DIFFERENCE IDAY IS NEGATIVE IF IYMD2,IHMS2 IS THE EARLIER TIME
ISEC	I	OUTPUT - REMAINDER OF DIFFERENCE IN SECONDS ISEC HAS THE SAME SIGN CONVENTION AS IDAY

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

```
      SUBROUTINE DIFF(IYMD1,IHMS1,IYMD2,IHMS2,IDAY,ISEC)
      DIMENSION MONTH(13,2)
      DATA MONTH/0,31,60,91,121,152,182,213,244,274,305,335,366,
     .           0,31,59,90,120,151,181,212,243,273,304,334,365/
      ISUB(IY)=MIN0(MOD(IY,4),1)+1
      ISEC=0
      IF(IYMD1.EQ.IYMD2) GOTO 4000
      CALL RYMDI(IYMD1,IY1,IM1,ID1)
      CALL RYMDI(IYMD2,IY2,IM2,ID2)
      L1=ISUB(IY1)
      IYEAR1=36525*(IY1-1)/100+MONTH(IM1,L1)+ID1
      L2=ISUB(IY2)
      IYEAR2=36525*(IY2-1)/100+MONTH(IM2,L2)+ID2
      ISEC=(IYEAR2-IYEAR1)*86400
 4000 ISEC1=IHMS1-40*(IHMS1/100)-2400*(IHMS1/10000)
      ISEC2=IHMS2-40*(IHMS2/100)-2400*(IHMS2/10000)
      ISEC=ISEC+ISEC2-ISEC1
      IDAY=ISEC/86400
      ISEC=ISEC-IDAY*86400
      RETURN
      END
```

DJUL

DESCRIPTION

DJUL computes the Julian date for a time input in days from Jan. 0.0 of the reference year.

NAME	DJUL
PURPOSE	TO COMPUTE JULIAN DATE FOR AN INPUT TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR FOR THE ARC
CALLING SEQUENCE	X=DJUL(DAY)

SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT - TIME IN DAYS FROM JAN 0.0 OF THE REFERENCE YEAR
DJUL	DP	OUTPUT - JULIAN DATE

SUBROUTINE USED	YMDAY
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      DOUBLE PRECISION FUNCTION DJUL(DAY)
      REAL*8 DJ,DAY,YMDAY
      LOGICAL NOT1ST/.FALSE./
      IF(NOT1ST) GO TO 10
      NOT1ST=.TRUE.
      DJ=2433281.5D0-YMDAY(500100,0,0.)
   10 DJUL=DJ+DAY
      RETURN
      END
```

EQN

DESCRIPTION

(See GEODYN)

EQUATR

DESCRIPTION

EQUATR rotates a set of vectors from mean or true equator and equinox of one epoch to mean or true equator and equinox of another epoch.

NAME EQUATR

PURPOSE TO ROTATE A SET OF VECTORS FROM MEAN OR TRUE EQUATOR AND EQUINOX OF ONE EPOCH TO MEAN OR TRUE EQUATOR AND EQUINOX OF ANOTHER EPOCH

CALLING SEQUENCE CALL EQUATR(X,DIN,TIN,Y,DOUT,TOUT,IEQ,NEQ)

SYMBOL	TYPE	DESCRIPTION
X	DP	INPUT - SET OF VECTORS TO BE ROTATED
DIN	DP	INPUT - DAY NUMBER OF THE COORDINATES SINCE JAN 0.0 OF THE REFERENCE YEAR
TIN	L	INPUT - TYPE OF INPUT .TRUE. = TRUE COORDINATE SYSTEM .FALSE. = MEAN COORDINATE SYSTEM
Y	DP	OUTPUT - ROTATED SET OF VECTORS
DOUT	DP	OUTPUT - DAY NUMBER OF OUTPUT VECTOR SET SINCE JAN 0.0 OF THE REFERENCE YEAR
TOUT	L	INPUT - TYPE OF OUTPUT .TRUE. = TRUE COORDINATE SYSTEM .FALSE. = MEAN COORDINATE SYSTEM
IEQ	I	INPUT - INDICATES WHICH MEMBERS OF THE SET ARE TO BE ROTATED
NEQ	I	INPUT - NUMBER OF MEMBERS OF THE SET TO BE ROTATED

SUBROUTINES USED NUTATE PRECES

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
      SUBROUTINE EQUATR(S,DIN,TIN,Y,DOUT,TOUT,IEQ,NEQ)
      REAL*8 X(6,1),Y(3,1),NP(3,3,4),T(3),DIN,DOUT,TEMP
      DIMENSION IEQ(1)
      LOGICAL TIN,TOUT
      M=2
      IF(.NOT.TIN) GO TO 10
      M=1
C OBTAIN MATRIX TO NUTATE FROM TRUE TO MEAN OF INPUT EPOCH
      CALL NUTATE(DIN,NP(1,1,1))
C OBTAIN MATRIX TO PRECESS FROM INPUT EPOCH TO 1950
 10   CALL PRECES(DIN,NP(1,1,2))
```

```
C OBTAIN MATRIX TO PRECESS FROM OUTPUT EPOCH TO 1950
      CALL PRECES(DOUT,NP(1,1,3))
      N=3
      IF(.NOT.TOUT) GO TO 20
      N=4
C OBTAIN MATRIX TO NUTATE FROM TRUE TO MEAN OF OUTPUT EPOCH
      CALL NUTATE(DOUT,NP(1,1,4))
C TRANSPOSE OUTPUT EPOCH PRECESSION AND NUTATION MATRICES
   20 DO 30 I=1,3
      DO 30 J=I,3
      DO 30 K=3,N
      TEMP=NP(I,J,K)
      NP(I,J,K)=NP(J,I,K)
   30 NP(J,I,K)=TEMP
      DO 70 II=1,NEQ
      JJ=IEQ(II)
      LL=(II-1)*81+1
      DO 40 I=1,3
   40 Y(I,LL)=X(I,JJ)
C ROTATE INPUT VECTOR TO OBTAIN OUTPUT VECTOR
      DO 60 K=M,N
      DO 50 I=1,3
      T(I)=Y(I,LL)
   50 Y(I,LL)=0.D0
      DO 60 I=1,3
      DO 60 J=1,3
   60 Y(I,LL)=Y(I,LL)+NP(I,J,K)*T(J)
   70 CONTINUE
      RETURN
      END
```

GETTAP

DESCRIPTION

GETTAP obtains a Julian date through common block CETBL9 and then reads the JPL ephemeris tape one record at a time. The record containing the information desired is loaded into common block CETBL3.

NAME GETTAP

PURPOSE READS JPL TAPE ONE RECORD AT A TIME GIVEN A JULIAN DATE

CALLING SEQUENCE CALL GETTAP

SUBROUTINES USED NONE

COMMON BLOCKS CETBL2 CETBL3 CETBL9 REC1 REC2 TAPE

INPUT FILES IN - JPL EPHEMERIS TAPE

OUTPUT FILES NONE

RESTRICTIONS NONE

REFERENCE JPL DEVELOPMENT EPHEMERIS NUMBER 19
TECHNICAL REPORT 32-1181 - C.J. DEVINE
JPL. CALIF. INST. OF TECH., PASADENA, CALIF.
NOV. 15, 1967

```
      SUBROUTINE GETTAP
      COMMON/CETBL2/ICW,ICENT,IREQ(13)
      COMMON/CETBL3/TAB3(829),NUTAT(204),CKSUM
      COMMON/CETBL9/JD1,TDAY,JDIF,IERR1
      COMMON/REC1/REC1(24)
      COMMON/REC2/TBODY,TYPE,AJD,BJD,STEP,DUM20(20)
      COMMON/TAPE/IN
      REAL REC2(25)
      DOUBLE PRECISION TAB3,DJ,JD1,TDAY,JDIF
      EQUIVALENCE (REC2(1),TBODY)
      IF(ICW.EQ.1) DJ=1.0D20
      IERR1=0
      JDIF=JD1-DJ
      IF(JD1.GE.DJ+8.0D0) GO TO 100
      IF(JD1.GE.DJ) RETURN
      REWIND IN
      READ(IN) REC1
      READ(IN) REC2
      READ(IN) TAB3,NUTAT,CKSUM
      ICW=2
      DJ=TAB3(1)
      JDIF=JD1-DJ
      IF(JD1.GE.DJ+8.0D0) GO TO 100
      IF(JD1.GE.DJ) RETURN
      PRINT 200,JD1,DJ,IN
      PRINT 300,TAB3,NUTAT,CKSUM
      STOP 51816
  100 READ(IN,END=150) TAB3,NUTAT,CKSUM
      DJ=TAB3(1)
```

```
      JDIF=JD1-DJ
      IF(JD1.GE.DJ+8.000) GO TO 100
      RETURN
  150 PRINT 250,JD1,DJ,IN
      PRINT 300,TAB3,NUTAT,CKSUM
      STOP 51316
  200 FORMAT('1***** DATA REQUESTED AT JULIAN DATE ',G16.9,' SMALLER ',
     .   'THAN FIRST DATE ',G16.9,' ON INPUT UNIT',I3)
  250 FORMAT('1***** DATA REQUESTED AT JULIAN DATE ',G16.9,' GREATER ',
     .   'THAN LAST DATE ',G16.9,' ON INPUT UNIT',I3)
  300 FORMAT(1H0/(5G25.16))
      END
```

MATRIX

DESCRIPTION

MATRIX calls subroutine NUTATE to find the nutation matrix for a time specified in the calling sequence. It then multiplies the nutation matrix by another 3x3 matrix passed in through the calling sequence.

NAME	MATRIX
PURPOSE	MULTIPLIES TWO 3X3 MATRICES
CALLING SEQUENCE	CALL MATRIX(DAY,A,B)

SYMBOL	TYPE	DESCRIPTION
DAY	DP	INPUT - TIME OF NUTATION MATRIX
A	DP	OUTPUT - PRODUCT OF THE TWO MATRICES
B	DP	INPUT - MATRIX TO BE MULTIPLIED BY THE NUTATION MATRIX

SUBROUTINES USED	NUTATE
COMMON BLOCKS	NONE
INPUT FILES	NONE
OUTPUT FILES	NONE
RESTRICTIONS	NONE
REFERENCES	NONE

```
      SUBROUTINE MATRIX(DAY,A,B)
      DOUBLE PRECISION A(3,3),B(3,3),CT(3,3),DAY
      CALL NUTATE(DAY,CT)
      DO 10 I=1,3
      DO 10 J=1,3
      A(I,J)=0.0D0
      DO 10 K=1,3
   10 A(I,J)=A(I,J)+B(I,K)*CT(J,K)
      RETURN
      END
```

MULMAT

DESCRIPTION

(See GEODYN)

NUTATE

DESCRIPTION

(See GEODYN)

PRECES

DESCRIPTION

(See GEODYN)

READE

DESCRIPTION

READE interpolates the JPL ephemeris quantities to find values on the date and time specified in the calling sequence. It then performs coordinate transformations as specified in common block CETBL2 and unit transformations as specified in common block CETBL1.

```
      SUBROUTINE READE(JED,TSEC,IERR)
      J.E. EKELUND, MESA SCIENTIFIC CORP., 1965 SEPT 15
      C.L. LAWSON, JPL, 1966 MAR 17
      READ JPL EPHEMERIS AT THE JULIAN EPHEMERIS DATE
      GIVEN BY (JED+TSEC/86400.D0)

**    ITEMS COMMUNICATED THROUGH THE CALLING SEQUENCE **

      JED        REFERENCE JULIAN EPHEMERIS DATE.
      TSEC       SECONDS OF EPHEMERIS TIME PAST JED.
                 ANY COMBINATION OF VALUES OF JED AND TSEC
                 IS ACCEPTABLE AS LONG AS (JED+TSEC/86400.D0)
                 IS WITHIN THE RANGE OF THE EPHEMERIS TAPE
                 BEING USED. HOWEVER TO OBTAIN THE
                 FINEST POSSIBLE RESOLUTION IN INTERPOLATION
                 THE NUMBER JED MUST BE AN EXACT MACHINE
                 NUMBER. FOR EXAMPLE JED COULD BE A DATE ENDING
                 WITH .0 OR .5 .
      IERR       ERROR FLAG
                 0=NO ERROR
                 1=(JED+TSEC/86400.D0) LESS THAN FIRST DATE
                    ON TAPE
                 2=(JED+TSEC/86400.D0) GREATER THAN LAST DATE
                    ON TAPE
                 3=SOME IREQ(I) IS NOT 0,1, OR 2
                 4=ICENT    IS NOT IN THE RANGE 1 THRU 11
                 5=ICW IS NOT 1,2, OR 3

** THE FOLLOWING ITEMS ARE INPUT THROUGH COMMON **
 *   COMMON BLOCK CETBL1 *
      AU               A.U. EXPRESSED IN DESIRED OUTPUT UNITS
      RE               EQUATORIAL RADIUS OF EARTH IN DESIRED OUTPUT UNITS
                       RE IS USED TO SCALE THE LUNAR EPHEMERIS
      TPD                 DESIRED NUMBER OF TIME UNITS PER DAY
      EMRAT      EARTH MOON MASS RATIO. SUGGESTED VALUE=81.3D0

      SUGGESTED VALUES FOR AU AND RE  DEPEND UPON
      DESIRED OUTPUT UNITS AS FOLLOWS..
      FOR OUTPUT IN EARTH RADII AU=23454.794001225117D0, RE =1.D0
      FOR OUTPUT IN KILOMETERS  AU=149599354C.D0,          RE =6378.169D0
      FOR OUTPUT IN A.U.   AU=1.D0,  RE =4.2635237111508509D-5

      SET TPD=86400.D0 FOR VELOCITY IN LINEAR UNITS PER SECOND.
      SET TPD=     1.D0 FOR VELOCITY IN LINEAR UNITS PER DAY.

 *   COMMON BLOCK CETBL2 *
      ICW        FLAG INDICATING STATUS OF COMMON BLOCKS REC2 AND CETBL3
                 1 MEANS NEITHER BLOCKS CONTAIN VALID DATA
                 2 MEANS BOTH BLOCKS CONTAIN VALID DATA
                 3 MEANS REC2 IS VALID, CETBL3 IS NOT
                 USER MUST SET ICW=1 BEFORE INITIAL CALL
      ICENTR     SPECIFIES CENTRAL BODY FOR COORDINATE
                 TRANSLATION AS FOLLOWS..
                  1 MERC      5 JUP       9 PLUTO
                  2 VENUS     6 SAT      10 SUN
```

```
             3 EARTH      7 URANUS   11 MOON
             4 MARS       8 NEP
   IREQ()    IREQ(J) SPECIFIES OUTPUT DESIRED FOR
             BODY NO. J.
             IREQ(J)=0  NO OUTPUT
                      1 POSITION
                      2 POSITION AND VELOCITY
             J RUNS FROM 1 TO 11 AS FOLLOWS..
             1 MERC       5 JUP       9 PLUTO
             2 VENUS      6 SAT      10 SUN
             3 EARTH      7 URANUS   11 MOON
             4 MARS       8 NEP      12 ERTH-MN-BARYCENTER
                                     13 NUTATION
 *  COMMON BLOCK CETBL3 *
  TAB3 829 DOUBLE PREC. WORD BUFFER TO ACCOMODATE  J.D. AND EPHEMERIS.
  NUTAT 204 SINGLE PREC. WORD BUFFER TO ACCOMODATE NUTATION DATA.
  CKSUM 1 S.P. WORD FOR CHECKSUM.
** THE FOLLOWING ITEMS ARE OUTPUT THROUGH COMMON **
 *  COMMON BLOCK CETBL4 *
   TABOUT( , )   PLANETARY AND LUNAR OUTPUT, SCALED AND
     TRANSLATED WITH RESPECT TO CENTER.
                 TABOUT (I,J) CONTAINS OUTPUT FOR
                 BODY NO. J. (1 .LE. J .LE. 12)
                 THE INDEX I IDENTIFIES COMPONENTS AS FOLLOWS..
                 1=X        2=Y        3=Z
                 4=XDOT     5=YDOT     6=ZDOT
   NUT( )        NUTATION OUTPUT
                 NUT(1)=DELTA LONGITUDE
                 NUT(2)=DELTA OBLICUITY
                 NUT(3)=TIME DERIVATIVE OF NUT(1)
                 NUT(4)=TIME DERIVATIVE OF NUT(2)
 *  COMMON BLOCK CETBL5 *
   BIVECT( , )   WORKING ARRAY. CONTENTS ARE INTERPOLATED
                 AND SCALED BUT NOT TRANSLATED. 1ST INDEX RUNS
                 OVER X,Y,Z,XDOT,YDOT,ZDOT AS IN TABOUT
                 BUT 2ND INDEX IS DIFFERENT AS FOLLOWS..
             BODIES 1 THRU 9 ARE HELIOCENTRIC.
                 1 MERC       5 JUP        9 PLUTO
                 2 VENUS      6 SAT       10 MOON REL TO EARTH
                 3 ERTHMN     7 URANUS    11 ERTHMN REL TO EARTH
                 4 MARS       8 NEP       12 ERTHMN REL TO MOON
                                          13 SEE 4092+

   THE COMMON BLOCK 'CETBL9' IS FOR COMMUNICATION
   BETWEEN RDEP2 AND GETR2.


   SUBROUTINE READE(JED,TSEC,IERR)                                  READ 105
   COMMON /CETBL1/            AU,RE,TPD,EMRAT                       READ 106
   COMMON/CETBL2/ICW,ICENT   ,IREQ(13)                              READ 107
   COMMON /CETBL3/ TAB3(829),NUTAT(204),CKSUM                       READ 108
   COMMON/CETBL4/TABOUT(6,12),NUT(4)                                READ 109
   COMMON /CETBL5/ BIVECT( 6,13)                                    READ 110
   COMMON/CETBL9/JD1,TDAY,JDIF,IERR1                                READ 111
```

```
      LOGICAL WFLAG
      INTEGER KREQ(12),MCENT(11),M1(20),JREQ(11), IPOS(11),IVEL(11)
      REAL NUTAT ,STP(11)
      DOUBLE PRECISION      AU,RE,EMRAT,  TPD,TAB3,BIVECT,TABOUT,NUT
      DOUBLE PRECISION JD1,TDAY,JDIF ,JED
      DOUBLE PRECISION TSEC,RAT, FAC,U(2,3),C,TEMP
C                                       FAC=1/86400
      DATA FAC/1.1574074074074074D-5/
      DATA STP/ 2.,8*4.,2*.5/
      DATA KREQ/5,5,2,6*5,4,3,1/
      DATA MCENT/15,15,0,6*15,10,5/
      DATA M1/11,0,10,11,0,12,10,0,12,0,3,3,3,0,0,3,13,13,0,0/
      DATA
     * IPOS  /02,092,146,200,254,308,362,416,470,524,1/
     *,IVEL  /47,119,173,227,281,335,389,443,497,677,103/
      JD1=JED
      TDAY=TSEC*FAC
      CALL GETTAP
      IF(IERR1 .NE. 0) GO TO 5000
      IF(ICENT .GE. 1 .AND. ICENT .LE. 11) GO TO 10
      IERR1=4
      GO TO 5000
   10 CONTINUE
C                             SET JREQ()TO CONTROL INTERPOLATION
      DO 20 I=1,10
      IF(IREQ(I) .GE. 0 .AND. IREQ(I) .LE. 2) GO TO 20
      IERR1=3
      GO TO 5000
   20 JREQ(I)=IREQ(I)
C                             BARYCENTER FLAG
      JREQ(3)=IREQ(12)
      MAXPL=JREQ(1)
      DO 24 I=2,10
   24 MAXPL=MAX0(MAXPL,JREQ(I))
      MAXEM=MAX0(IREQ(3),IREQ(11))
      MAXALL=MAX0(MAXPL,MAXEM)
      IF(ICENT.EQ.3.OR.ICENT.EQ.11) GO TO 28
C                             CENTER IS NOT EARTH OR MOON
C                             10=MOON,3=ERTHMN
      JREQ(10)=MAXEM
      JREQ(3)=MAX0(JREQ(3),MAXEM)
      JREQ(ICENT)=MAXALL
      GO TO 32
C                             CENTER IS EARTH OR MOON
C                             10=MOON,3=ERTHMN
   28 JREQ(10)=MAXALL
      JREQ(3)=MAXPL
   32 JREQ(11)=IREQ(13)
      LUNAR=JREQ(10)
      IBARY=JREQ(3)*3
C                             JREQ( )IS NOW SET
      SAVE=0.
      DO 240 IBODY=1,11
      IF(JREQ(IBODY)) 240,240,40
   40 IF(STP(IBODY).EQ.SAVE) GO TO 165
      SAVE=STP(IBODY)
```

```
  160 TEMP=JDIF/SAVE
      KK=TEMP
      U(1,1)=TEMP-FLOAT(KK)
      IF(U(1,1))161,165,161
  161 CONTINUE
      U(2,1)=1.D0-U(1,1)
      DO 163 IU=1,2
      U(IU,3)=U(IU,1)*U(IU,1)
      U(IU,2)=(U(IU,3)-1.D0)/6.D0
  163 U(IU,3)=(U(IU,3)-4.D0)/20.D0
  165 IF(IBODY-10) 169,167,220
  167 C=RE
      GO TO 172
  169 C=AU
C                           INTERPOLATE IBODY=1,2,....,10
  172 IGET1=IPOS(IBODY)+KK*9
      IC1=1
  200 CONTINUE
      IF(U(1,1))203,201,203
  203 IGET2=IGET1+6
      DO 204 IGET=IGET1,IGET2,3
      BIVECT(IC1,IBODY)=
     *            C*(U(2,1)*(TAB3(IGET    )+
     *                U(2,2)*(TAB3(IGET+ 1)+
     *                U(2,3)* TAB3(IGET+ 2)))+
     *                U(1,1)*(TAB3(IGET+ 9)+
     *                U(1,2)*(TAB3(IGET+10)+
     *                U(1,3)* TAB3(IGET+11))) )
  204 IC1=IC1+1
      GO TO 205
  201 IC2=IC1+2
      DO 202 I=IC1,IC2
      BIVECT(I,IBODY)=C*TAB3(IGET1)
  202 IGET1=IGET1+3
  205 CONTINUE
      JREQ(IBODY)=JREQ(IBODY)-1
      IF(JREQ(IBODY)) 240,240,207
  207 IGET1=IVEL(IBODY)+KK*9
      IC1=4
      C=C/TPD
      GO TO 200
C                           INTERPOLATE   IBODY=11   NUTATION
  220 C=1.D0
      IGET1=IPOS(IBODY)+KK*6
      IC1=1
  222 IGET2=IGET1+3
  225 IF(U(1,1))228,226,228
  228 DO 230 IGET=IGET1,IGET2,3
      NUT(IC1)=
     *   C*(U(2,1)*(NUTAT(IGET   )+
     *      U(2,2)*(NUTAT(IGET+1)+
     *      U(2,3)* NUTAT(IGET+2)))+
     *      U(1,1)*(NUTAT(IGET+6)+
     *      U(1,2)*(NUTAT(IGET+7)+
     *      U(1,3)* NUTAT(IGET+8))) )
  230 IC1=IC1+1
```

```
      GO TO  232
  226 DO 227 IGET=IGET1,IGET2,3
      NUT(IC1)=C*NUTAT(IGET)
  227 IC1=IC1+1
  232 CONTINUE
      JREQ(IBODY)=JREQ(IBODY)-1
      IF(JREQ(IBODY)) 240,240,236
  236 C=C/TPD
      IGET1=IVEL(IBODY)+KK*6
      IC1=3
      GO TO 222
  240 CONTINUE
C                              INTERPOLATION IS FINISHED
C                              RESULTS ARE IN BIVECT( , )AND NUT( )
C                              TEST MOON REQUEST
      IF(LUNAR   ) 4020,4020,4010
C                              NOTE..EMRAT=EARTH MASS/MOON MASS
C                              SET BIVECT( ,11)=ERTHMN CENTERED AT EARTH
C                              SET BIVECT( ,12)=ERTHMN CENTERED AT MOON
 4010 RAT=1.D0/(EMRAT+1.D0)
      IMAX=LUNAR * 3
      DO 4016 I=1,IMAX
      BIVECT(I,11)=RAT*BIVECT(I,10)
 4016 BIVECT(I,12)=-EMRAT*BIVECT(I,11)
 4020 WFLAG=.FALSE.
      KCENT=MCENT(ICENT)
C                              BEGIN TRANSLATION LOOP
      DO 4108  IBODY=1,12
      IF(IREQ(IBODY)) 4108,4108,4024
 4024 IMAX=IREQ(IBODY)*3
      KASE=KCENT+KREQ(IBODY)
      K1=M1(KASE)
      GO TO (4032,4060,4032,4048,4052,
     *       4032,4068,4040,4048,4056,
     *       4032,4076,4076,4040,4028,
     *       4064,4088,4088,4064,4080),KASE
C                              KASE=15
 4028 K1=IBODY
C                              KASE=1,3,6,11
 4032 DO 4036 I=1,IMAX
 4036 TABOUT(I,IBODY)=BIVECT(I,K1)
      GO TO 4108
C                              KASE=2,8,14
 4040 DO 4044 I=1,IMAX
 4044 TABOUT(I,IBODY)=0.D0
      GO TO 4108
C                              KASE=4,9
 4048 K2=3
      GO TO 4100
C                              KASE=5
 4052 L2=11
      GO TO 4060
C                              KASE=10
 4056 L2=12
 4060 K1=IBODY
      K2=13
```

```
      GO TO 4092
C                                 KASE=19
 4064 K1=ICENT
C                                 KASE=7
 4068 DO 4072 I=1,IMAX
 4072 TABOUT(I,IBODY)=-BIVECT(I,K1)
      GO TO 4108
C                                 KASE=12,13
 4076 K2=KASE-1
      GO TO 4100
C                                 KASE=20
 4080 K1=IBODY
C                                 KASE=16
 4084 K2=ICENT
      GO TO 4100
C                                 KASE=17,18
 4088 L2=ICENT
      K2=KASE-6
 4092 IF(WFLAG) GO TO 4100
      WFLAG=.TRUE.
C                                 BIVECT( ,13)IS AN AUXILIARY VECTOR
C                                 NEEDED WHEN KASE=5,10,17,18.
C                FOR KASE=05       BIVECT( ,13)=EARTH  CENTERED AT SUN
C                FOR KASE=10       BIVECT( ,13)=MOON  CENTERED AT SUN
C                FOR KASE=17,18       BIVECT( ,13)=ERTHMN  CENTERED AT ICENT
      DO 4096 I=1,IBARY
 4096 BIVECT(I,13)=BIVECT(I,3)-BIVECT(I,L2)
 4100 DO 4104 I=1,IMAX
 4104 TABOUT(I,IBODY)=BIVECT(I,K1)-BIVECT(I,K2)
 4108 CONTINUE
 5000 IERR=IERR1
      RETURN
      END
```

ROTMAT

DESCRIPTION

(See GEODYN)

RYMDI

DESCRIPTION

RYMDI separates a six digit number representing a date in the form YYMMDD into three two digit numbers representing the year, month, and day.

NAME   RYMDI

PURPOSE   TO SEPARATE PACKED SIX-DIGIT DECIMAL DATES INTO
                    TWO-DIGIT YEAR, MONTH, AND DAY

CALLING SEQUENCE   CALL RYMDI(YMD,Y,M,D)

SYMBOL	TYPE	DESCRIPTION
YMD	I	INPUT - DATE TO BE SEPARATED
Y	I	OUTPUT - TWO-DIGIT YEAR
M	I	OUTPUT - TWO-DIGIT MONTH
D	I	OUTPUT - TWO-DIGIT DAY

SUBROUTINES USED   NONE

COMMON BLOCKS   NONE

INPUT FILES   NONE

OUTPUT FILES   NONE

RESTRICTIONS   NONE

REFERENCES   NONE

```
      SUBROUTINE RYMDI (YMD,Y,M,D)
         INTEGER YMD,Y,M,D
         Y=YMD/10000
         I=YMD/100
         M=I-Y*100
         D=YMD-I*100
         RETURN
         END
```

YMDAY

DESCRIPTION

YMDAY is a real valued DOUBLE PRECISION function used to compute from a given date and time the number of days from January 0.0 of a given reference year.

NAME YMDAY

PURPOSE GIVEN A DATE COMPUTES THE NUMBER OF DAYS FROM JAN 0.0 OF THE REFFRENCE YEAR FOR THE ARC

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

SYMBOL	TYPE	DESCRIPTION
X	DP	OUTPUT - NUMBER OF DAYS FROM JAN 0.0 OF THE REFFRENCE YEAR
IYMD	I	INPUT - DATE IN THE FORM YYMMDD
IHM	I	INPUT - HOURS AND MINUTES IN THE FORM HHMM
SEC	R	INPUT - SECONDS

SUBROUTINES USED ADDYMD

COMMON BLOCKS NONE

INPUT FILES NONE

OUTPUT FILES NONE

```
      DOUBLE PRECISION FUNCTION YMDAY(IYMD,IHM,SEC)
      LOGICAL NOT1ST
      DATA NOT1ST/.FALSE./
      IF(NOT1ST) GO TO 10
      NOT1ST=.TRUE.
      IY=(IYMD/10000)*10000+101
   10 IHMS=IHM*100
      CALL DIFF(IY,0,IYMD,IHMS,ID,IS)
      YMDAY=86400*(ID+1)+IS
      YMDAY=(YMDAY+SEC)/8.64D4
      RETURN
      END
```

1.2.4 ORB1 CONVERSION

INTRODUCTION

The ORB1 CONVERSION program is used to convert a 9-track 360 double-precision ORB1 tape to a 7-track 7094 single-precision ORB1 tape.

The main routine reads in IBM 360 double-precision words and writes on a 7-track tape the equivalent IBM 7094 single-precision words.

The subroutine WORD94 does the conversion from the IBM 360 64-bit floating point format to the IBM 7094 36-bit floating point format.

MAIN - ORB1 CONVERSION

DESCRIPTION

The main program for ORB1 reads a block, converts each double precision word in the block to the IBM 7094 single precision format using subroutine WORD94, and then outputs the converted block. This procedure continues until all blocks on the input tape have been processed.

NAME	MAIN - ORB1 CONVERSION
PURPOSE	TO CONVERT A 9-TRACK IBM 360 FORMAT ORB1 TAPE TO 7-TRACK IBM 7094 FORMAT
SUBROUTINE USED	WORD94
COMMON BLOCKS	NONE
INPUT FILE	IN - FORTRAN LOGICAL UNIT NUMBER FOR INPUT TAPE
OUTPUT FILE	OUT - FORTRAN LOGICAL UNIT NUMBER FOR OUTPUT TAPE
RESTRICTIONS	NONE
REFERENCES	GSFC ORBIT TAPE - FORMAT 1

```
      LOGICAL*1 BUF(6,350)
      REAL*8 DBUF(350)
      INTEGER IN/10/,OUT/11/
C READ EACH RECORD
   10 READ(IN,END=30) DBUF
C CALL WORD94 TO CONVERT EACH INPUT WORD TO OUTPUT FORMAT
      DO 20 I=1,350
   20 CALL WORD94(DBUF(I),BUF(1,I))
C OUTPUT RECORD
      WRITE(OUT,1000) BUF
      GO TO 10
C END FILE OUTPUT TAPE AND TERMINATE
   30 ENDFILE OUT
      STOP
 1000 FORMAT(2100A1)
      END
```

WORD94

DESCRIPTION

Subroutine WORD94 converts a word in 64 bit IBM 360 floating point format to 36 bit IBM 7094 floating point format.

The order of computation is as follows:

- Bits 8-38 (the fraction) of the 360 word are extracted and placed in bits 2-31 of an integral word (NUM).
- The sign (bit 0) and exponent (bits 1-7) are extracted and stored as integer.
- 40_{16} is subtracted from the exponent and the result multiplied by 4 to change to base 2.
- Bits 30-28 are sequentially tested for non zero to obtain a normalization count, N.

- N is added to the exponent and the fraction (NUM) is shifted right 4-N bits.

- The fraction is then stored six bits at a time from the right (bits 76-31, 20-25 into the output characters (6,5...1).

- In 2d character WORD94 stores the low order 3 bits of the exponent and bits 28-30 of the fraction.

- In the 1st character WORD94 stores the high order bits of the exponent and the sign.

C-4

NAME		WORD94
PURPOSE		TO CONVERT FROM 360, 64 BIT FORMAT TO 7094, 36 BIT FORMAT (FLOATING POINT)
CALLING SEQUENCE		CALL WORD94(W360,W94)
SYMBOL	TYPE	DESCRIPTION
W360 (8)	L*1	INPUT - 360 DOUBLE PRECISION WORD
W94	L*1	OUTPUT - 7094 SINGLE PRECISION WORD
SUBROUTINES USED		NONE
COMMON BLOCKS		NONE
INPUT FILES		NONE
OUTPUT FILES		NONE
RESTRICTIONS		NONE
REFERENCES		IBM 360 AND 7094 PRINCIPLES OF OPERATION MANUALS

```
      SUBROUTINE WORD94(W360,W94)
      LOGICAL*1 W360(8),W94(6),L1(4),L
      EQUIVALENCE (L1,IN),(L1(4),L)
C BIT CONTAINS ALL INTEGER POWERS OF 2 WHICH FIT IN I*4 WORD
      INTEGER BIT(32)/Z00000001,Z00000002,Z00000004,Z00000008,
     *                Z00000010,Z00000020,Z00000040,Z00000080,
     *                Z00000100,Z00000200,Z00000400,Z00000800,
     *                Z00001000,Z00002000,Z00004000,Z00008000,
     *                Z00010000,Z00020000,Z00040000,Z00080000,
     *                Z00100000,Z00200000,Z00400000,Z00800000,
     *                Z01000000,Z02000000,Z04000000,Z08000000,
     *                Z10000000,Z20000000,Z40000000,Z80000000/
C EXTRACT MANTISSA (FRACTIONAL PART)
      IN=0
      L=W360(5)
      NUM=IN/4
      L=W360(4)
      NUM=NUM+IN*BIT(7)
      L=W360(3)
      NUM=NUM+IN*BIT(15)
      L=W360(2)
      NUM=NUM+IN*BIT(23)
C EXTRACT SIGN BIT
      L=W360(1)
      ISGN=IN/BIT(8)
C EXTRACT EXPONENT
      IEXP=(IN-ISGN*BIT(8)-BIT(6))*BIT(3)
      N=0
C TEST FOR ZERO WORD
```

```
      IF(IEXP.GT.0) GO TO 10
      IEXP=0
      NUM=0
C ADJUST EXPONENT FOR BINARY NORMALIZATION INSTEAD OF HEX NORMALIZATION
 10   DO 20 I=1,3
      IF(NUM.GE.BIT(31-I).OR.IEXP.EQ.0) GO TO 30
      IEXP=IEXP-1
 20   N=N+1
C SHIFT MANTISSA TO ACCOUNT FOR NORMALIZATION
 30   NUM=NUM/BIT(4-N)
C OUTPUT 7094 MANTISSA
      DO 40 I=1,5
      IN=NUM
      W94(7-I)=L
 40   NUM=NUM/BIT(7)
C PUT LOW ORDER THREE BITS OF EXPONENT IN WITH FIRST 3 BITS OF MANTISSA
      IEXP=IEXP+ISGN*BIT(9)
      IN=IN+IEXP*BIT(4)
      W94(2)=L
C INSERT HIGH ORDER BITS OF EXPONENT AND SIGN
      IN=IEXP/BIT(4)
      W94(1)=L
C ALL DONE
      RETURN
      END
```

1.2.5 TDIF TABLE GENERATOR

INTRODUCTION

The TDIF TABLE GENERATOR generates tabular differences between time systems A.1 and UT1. It reads as input the differences between systems UT1 and UTC which are obtained from B.I.H. Using the differences between A.1 and UTC computed by subroutine TDIF in conjunction with the difference between UT1 and UTC, the TDIF TABLE GENERATOR computes the differences between A.1 and UT1.

Continual maintenance is required to keep these tables up-to-date.

SUBROUTINE CROSS REFERENCE CHART

CALLING ROUTINES

CALLED ROUTINES	MAIN	DJUL*	TDIF	YMDAY
DIFF				●
DJUL	●			
TDIF	●			
YMDAY	●	●	●	

*DJUL IS AN ENTRY POINT IN DPFCT.

COMMON BLOCK CROSS REFERENCE CHART

ROUTINES

COMMON BLOCKS	MAIN	BLOCK DATA	DPFCT	DIFF	TDIF	YMDAY
CONSTS		●	●			
CSTHET			●			
CTIME	●					●
INITBK	●		●		●	
MONTHS		●		●		

MAIN-TDFGEN

DESCRIPTION

The MAIN routine reads the UT1-UTC time differences and uses subroutine TDIF to obtain the A.1-UTC differences. Then MAIN subtracts these differences [(A.1-UTC) - (UT1-UTC)] to obtain the time differences between systems A.1 and UT1 (A.1-UT1).

NAME	MAIN - TDFGEN
PURPOSE	COMPUTES TIME DIFFERENCES BETWEEN A.1 AND UT1
SUBROUTINES USED	DJUL TIF YMDAY
COMMON BLOCKS	CTIME INITBK
INPUT FILE	5 - CARD INPUT
OUTPUT FILES	6 - PRINTER 7 - PUNCHED CARDS
RESTRICTIONS	NONE
REFERENCES	NONE

```
      REAL*8 YMDAY,DAY,DJUL,DJ
      DIMENSION A1UT1(1000)
      COMMON/CTIME/CUM(22),IYREF
      COMMON/INITBK/NOT1ST(57)
      DO 5 I=1,57
    5 NOT1ST(I)=0
      IYREF=66
      NUM=0
      IHM=0
      SEC=0
   10 READ(5,1000,END=100) IYMD,UT1UTC
      DAY=YMDAY(IYMD,IHM,SEC)
      A1UTC=TDIF(4,3,DAY)
      TA1UT1=TDIF(4,1,DAY)
      NUM=NUM+1
      DJ=DJUL(DAY)
      A1UT1(NUM)=A1UTC-UT1UTC
      IF(MOD(NUM,50).EQ.1) PRINT 2000
      IF(MOD(NUM,5).EQ.1) PRINT 2005
      PRINT 2010,IYMD,IHM,SEC,DJ,A1UTC,UT1UTC,A1UT1(NUM),TA1UT1
      GO TO 10
  100 PUNCH 3000,(A1UT1(I),I=1,NUM)
      PRINT 3000,(A1UT1(I),I=1,NUM)
      STOP 41
 1000 FORMAT(I6,F10.5)
 2000 FORMAT('1YYMMDD HHMM SS.SSSS      JULIAN      A1-UTC   UT1-UTC   ',
     . 'A1-UT1',4X,'A1-UT1 TDIF')
 2005 FORMAT(1X)
 2010 FORMAT(1X,I6,I5,F8.4,2X,F11.1,3X,F7.4,F9.4,F3.4,3F11.4)
 3000 FORMAT(5X,1H.,1X,F7.4,1H.,F7.4,1H.,F7.4,1H.,F7.4,1H.,
     .  F7.4,1H.,F7.4,1H.,F7.4,1H.,F7.4,1H.)
      END
```

TDIF

DESCRIPTION

(See GEODYN)

BLOCK DATA

DESCRIPTION

The block data routine initializes values for π, 2π, and the conversion factors for converting degrees to radians and arc seconds to radians. It also gives the day number of the first day of each month in a regular year and in a leap year starting from Jan. 0.0 of that year.

NAME	BLOCK DATA
PURPOSE	DATA INITIALIZING OF PI, 2*PI, CONVERSION FACTOR OF DEGREES TO RADIANS, CONVERSION FACTOR OF ARC SECONDS TO RADIANS, AND THE DAY NUMBER OF THE FIRST DAY OF EACH MONTH IN A YEAR
COMMON BLOCK	MONTH

```
      BLOCK DATA
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON/CONSTS/PI,TWOPI,DRAD,DRSEC
      COMMON/MONTHS/MONTH(26)
      DATA PI/3.14159265358979320D0/,
     *  TWOPI/6.28318530717958640D0/,
     *   DRAD/.0174532925199432960D0/,
     *  DRSEC/.4848136811095360D-5/
      DATA MONTH/0,31,60,91,121,152,182,213,244,274,305,335,366,
     *           0,31,59,90,120,151,181,212,243,273,304,334,365/
      END
```

DIFF

DESCRIPTION

(See GEODYN)

DPFCT

DESCRIPTION

(See GEODYN)

YMDAY

DESCRIPTION

(See GEODYN)

SECTION 2.0

OPERATIONS DESCRIPTION OF GEODYN SUPPORT PROGRAMS

2.1 GEODYN ANALYSES AND GRAPHICS SUPPORT PROGRAMS

The GEODYN Analyses and Graphics Support Programs constitute an integral part of the GEODYN System. Included within this set of programs are the following:

- DELTA - DELTA computes, prints and plots satellite trajectory differences.
- GEORGE - GEORGE performs a linear regression analysis on GEODYN residuals.
- GROUNDTRACK - GROUNDTRACK plots subsatellite groundtracks used for analysis of tracking station-satellite pass geometric relationships.
- WRDC SC4020 PLOT PACKAGE - The Plot Package is a group of subroutines that may be used to generate plots.

The operation of the programs DELTA, GEORGE, and GROUNDTRACK will be described in the following pages of this section.

2.1.1 DELTA

DELTA is a GEODYN support program which reads satellite trajectory tapes written by GEODYN and computes, prints and plots orbital differences. DELTA reads inertial Cartesian coordinates and computes trajectory differences in the more physically meaningful radial, cross track, and along track directions. Optional output from DELTA is a plot of these trajectory differences. By calling WRDC SC4020 PLOT PACKAGE subroutines, DELTA will plot these differences on the printer and/or will write an SC4020 Plotter Driver Tape which may be used to obtain microfilm and/or hard copy plots of the DELTA trajectory differences from the SC4020 plotter.

The following pages will describe in detail the setup and operation of the DELTA support program of the GEODYN System.

2.1.1.1 DELTA Input Cards

The entire card input to DELTA consists of four cards per case with no limit on the number of cases. The four cards input to DELTA consist of the DELTA Option Card and three title cards. These cards are described below.

1. The DELTA Option Card

COLUMNS	FORMAT	DESCRIPTION
1-2	I2	Change of unit for first satellite trajectory tape (default is 21 or that value used by the previous case).
3-4	I2	Change of unit for second satellite trajectory tape (default is 22 or that value used by the previous case).
5	L1	T- Plot requested. F or blank--no plot requested.
6	I1	Request for type of output 1 = microfilm 2 = hardcopy 4 = printer Any combination of the above may be used by simple summation. (Default is 7).
7	I1	=1 Specifies that the input tapes are ORB1 tapes. (Default is RV tapes)
8-9	I2	$\left.\begin{array}{l}\leq 0\\ =1\end{array}\right\}$ Plots every point =n Plots every n^{th} point.

COLUMNS	FORMAT	DESCRIPTION
10	I1	≠0 Specifies that another case will follow. =0 This is the last case.
11-22	F12.6	Y-scale upper limit for plots. A suitable default value will be used if no value is input here.
23-34	F12.6	Y-scale lower limit for plots. A suitable default value will be used if no value is input here.
35-46	F12.6	Y-scale divisions interval. A suitable default value will be used if no value is input here. If a value is input here DELTA will assume that values also have been input in columns 11-22 and 23-34.

2. The DELTA Title Cards

Any information may be punched on these title cards in columns 1-56. Information punched on these cards will appear on the first frame of all plots for this case. These cards should be present only when plotting is requested.

2.1.1.2 DELTA Job Control Language and Hardware and Software Restrictions

2.1.1.2.1 Job Control Language

The DELTA program may be executed by use of the LINKGO procedure as follows:

```
// EXEC LINKGO,REGION.GO=250K
//LINK.SYSLIN DD *
 INCLUDE LOADLIB(ZCTVMDEL)
 INCLUDE LOADLIB(ZCRGWTYP)
 ENTRY MAIN
/*
//GO.FT20F001 DD LABEL=(,BLP),UNIT=2400-7,
//   DCB=(RECFM=FB,LRECL=6,BLKSIZE=4092,DEN=1),
//   VOL=SER=PLOT2
//GO.FT21F001 DD UNIT=2400-9,VOL=SER=RVTAP1,
//   DCB=(RECFM=VBS,LRECL=72,BLKSIZE=7204),
//   LABEL=(,BLP)
//GO.FT22F001 DD UNIT=2400-9,VOL=SER=RVTAP2,
//   DCB=(RECFM=VBS,LRECL=72,BLKSIZE=7204),
//   LABEL=(,BLP)
//GO.DATA5 DD *

    <<<The DELTA Input Cards go here.<<<

/*
```

Unit 20 is used for output of the SC4020 Plotter Driver tape.

Units 21 and 22 and any other units which the user wishes to specify are used for input of RV tapes and/or ORB1 tapes.

2.1.1.2.2 Hardware and Software Restrictions

DELTA requires an IBM 360 computer with a minimum of 250K bytes of user accessable core, two 9 track tape drives, one 7 track tape drive, one card reader and one high speed printer.

The current DELTA program is executable under versions 14, 16, and 18 of the IBM 360 operating system.

The compilation of DELTA requires an IBM FORTRAN IV Level G compiler.

There are no DELTA software or hardware restrictions other than that the above mentioned hardware and software be available and working properly.

2.1.1.3 DELTA Example Job

The example job for DELTA is included with Example Three for GEODYN in section 4.3 of Volume 3. Shown in this example is the normal mode of operation for the DELTA program. However, as shown in section 2.1.1.2.1, tape input may also be used for DELTA rather than concatenating the DELTA execution with a GEODYN execution.

2.1.1.4 DELTA Error Messages

There are no DELTA error messages other than those which may be printed by the WRDC SC4020 PLOT PACKAGE. These messages are described in section 2.1.4 of this document.

2.1.2 GEORGE

GEORGE is a GEODYN support program which reads a Binary Residual tape written by GEODYN and using the information obtained from this tape performs statistical linear regression computations to determine tracking instrument zero-set biases and timing biases in the GEODYN residuals. Optional output from GEORGE is a residual plot which may be obtained on the printer and/or a WRDC SC4020 Plotter Driver tape from which may be obtained microfilm and/or hard copy plots of the GEODYN residuals. To perform such plotting functions, GEORGE calls subroutines in the WRDC SC4020 PLOT PACKAGE.

The following pages will describe in detail the setup and operation of the GEORGE support program of the GEODYN System.

2.1.2.1 GEORGE Input Cards

Specific functions of the GEORGE program may be requested by input cards. The GEORGE Input Cards are separated into two categories:

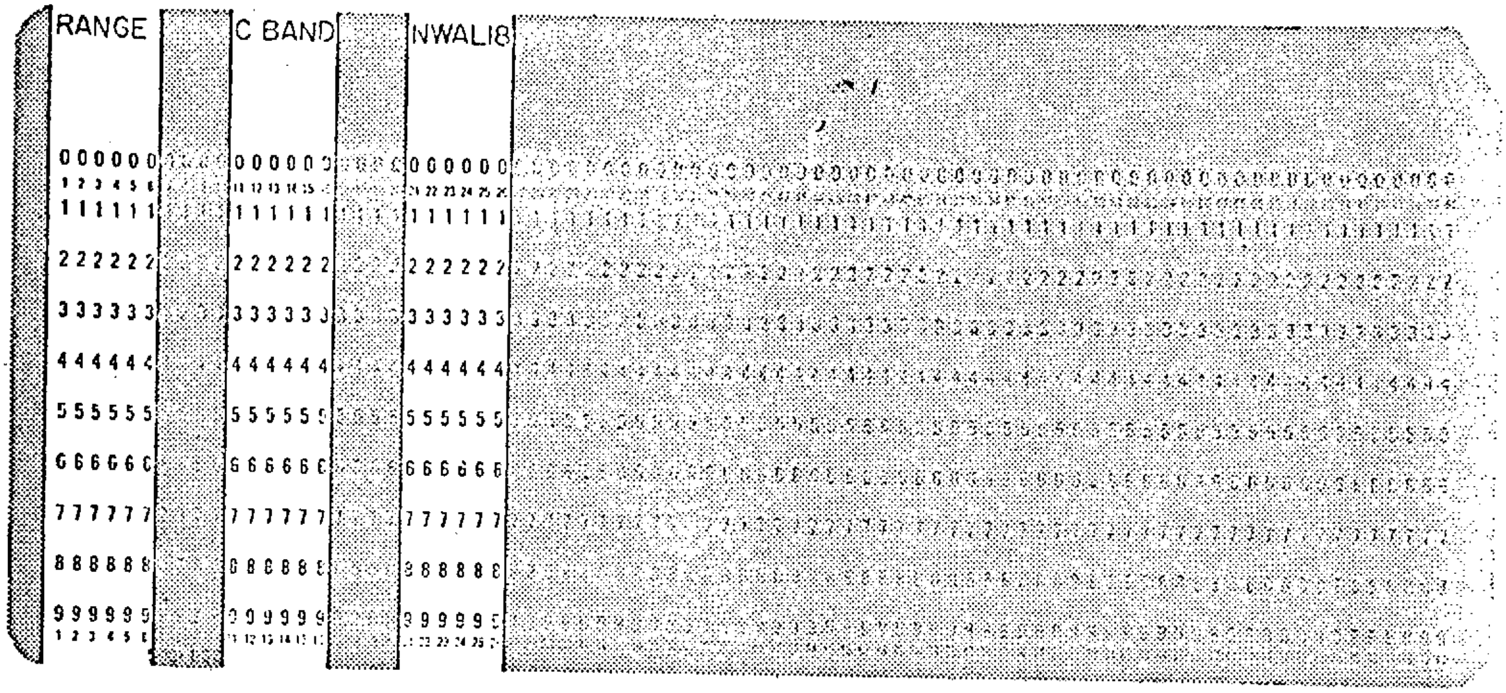
- GEORGE Mandatory Cards - these are cards that <u>must</u> be present for each case.

- GEORGE Option Cards - these are cards that as the name implies are optional.

A set of these cards will define a case. More than one case may be present.

This section of the manual describes the format and usage of the GEORGE Input Cards.

MEASUREMENT CARD*



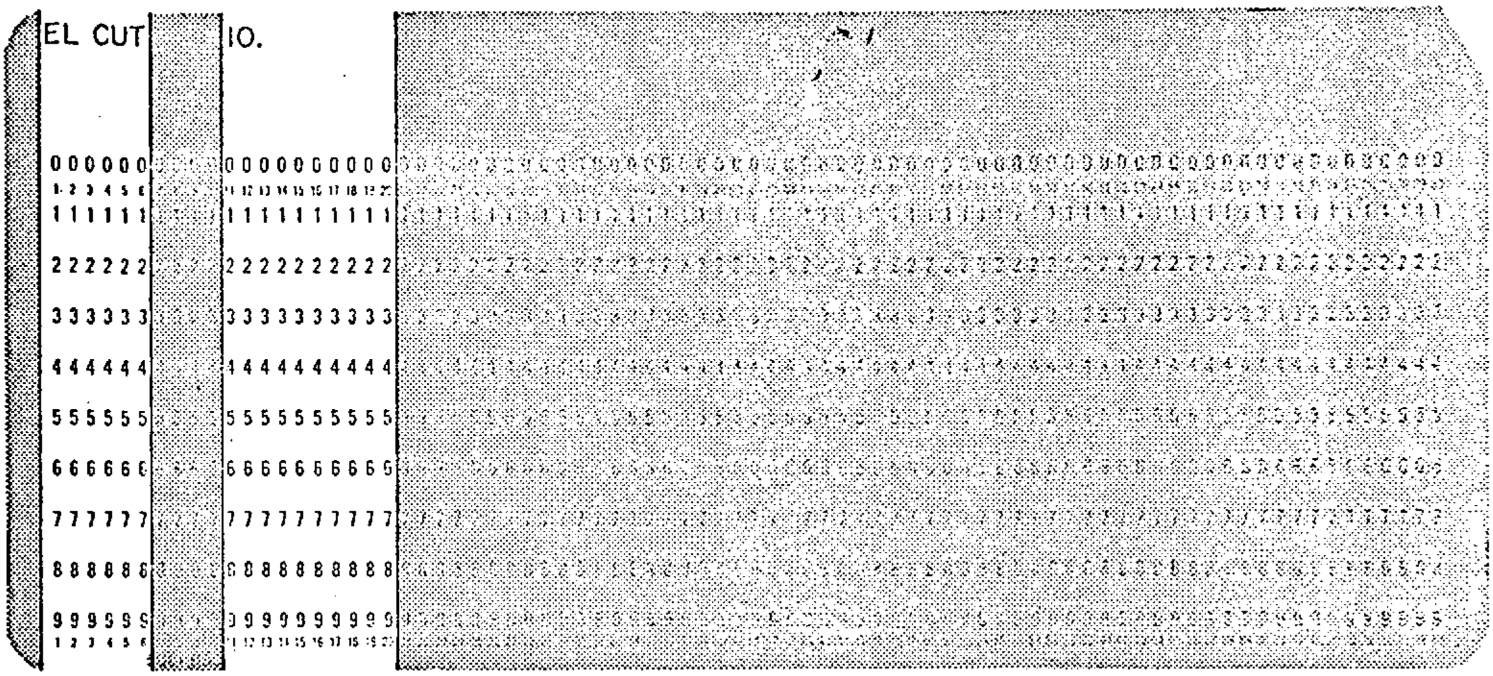
COLUMNS	FORMAT	DESCRIPTION
1-6	A6	Alphanumeric measurement type, left adjusted in field. Measurement types are: RT ASC, R RATE, X ANGL, RANGE, ALPHA, AZMUTH.
11-16	A6	Alphanumeric network name, left adjusted in field. Network names are: STADAN, DOPLER, USAF, C BAND, SECOR, USC+GS, SPEOPT, INTERL, SAO.
21-26	A6	Alphanumeric station name.

Notes: * One Measurement Card is mandatory for each case.

The measurement type must be specified.

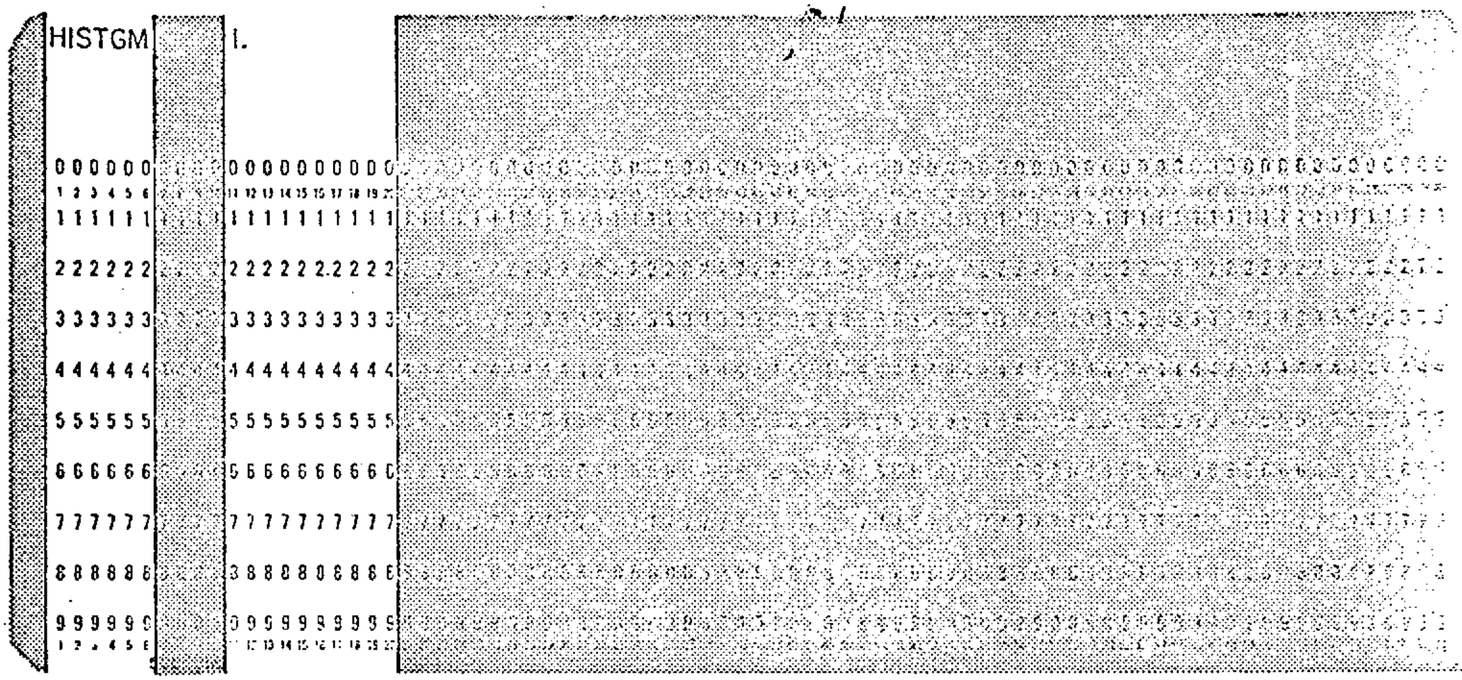
The network and station name are optional. If left blank, all networks and stations will be analyzed.

OPTION CARD
EL CUT



COLUMNS	FORMAT	DESCRIPTION
1-6	A6	The word "EL CUT" requests that data elevation cutoff be made.
11-20	F10.5	Desired data elevation cutoff angle.

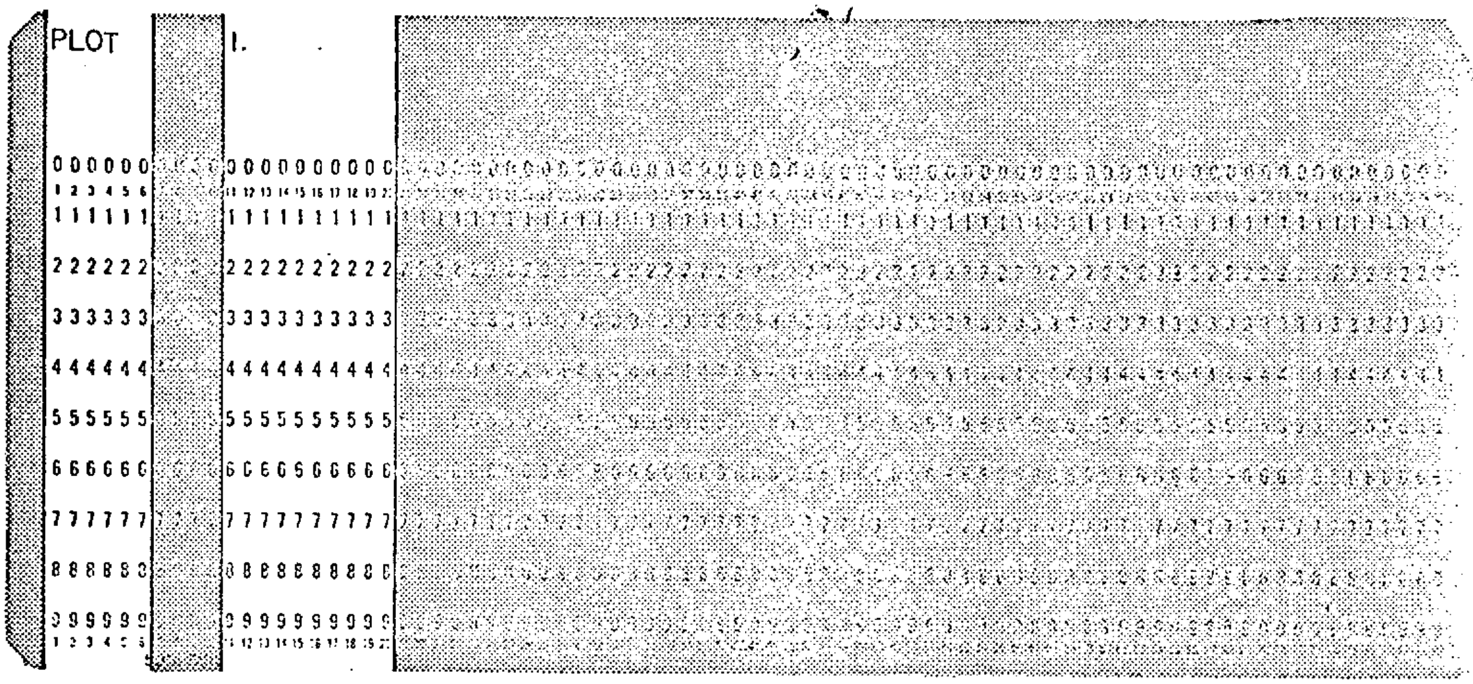
OPTION CARD
HISTGM



COLUMNS	FORMAT	DESCRIPTION
1-6	A6	The word "HISTGM" requests that histogram plots be made.
11-20	F10.5	=0. Histograms of residuals. =1. Histograms of residual ratios to sigma. =2. Histograms of residuals plus a final histogram of all residuals. =3. Histograms of residual ratios to sigma plus a final histogram of all residual ratios to sigma.

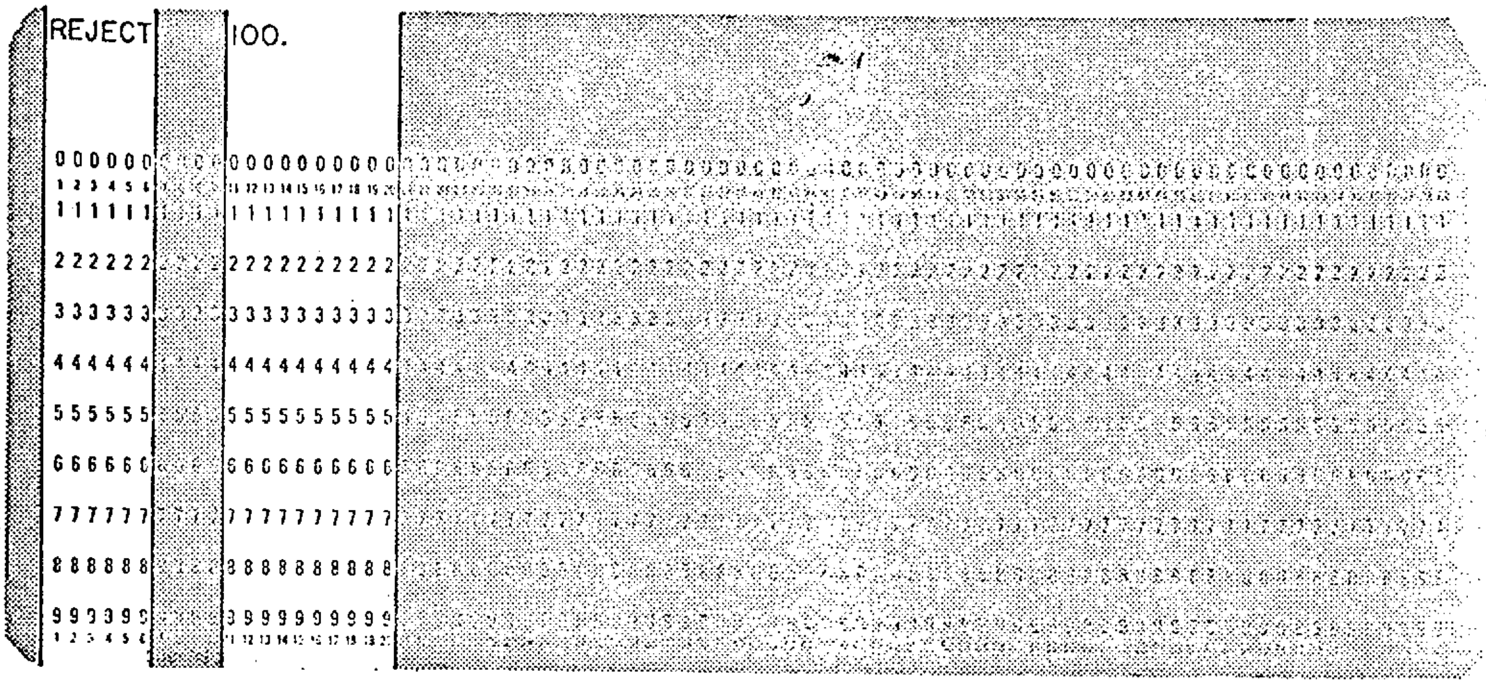
OPTION CARD

PLOT



COLUMNS	FORMAT	DESCRIPTION
1-4	A4	The word "PLOT" requests that plots of analysis be made.
11-20	F10.5	=0. Printer plots only. =1. Printer plots and SC4020 Plotter Driver tape.

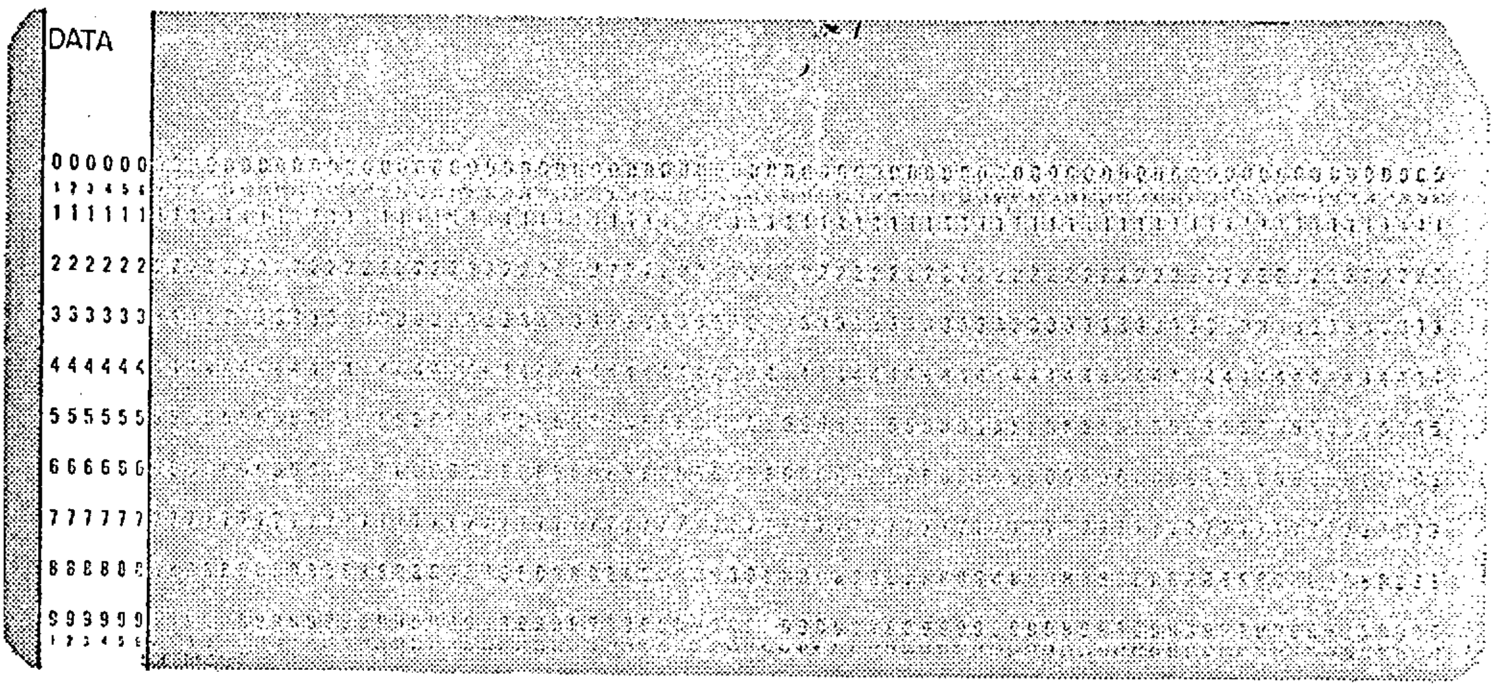
OPTION CARD
REJECT



COLUMNS	FORMAT	DESCRIPTION
1-6	A6	The word "REJECT" requests data editing.
11-20	F10.5	Value of the rejection criterion.

OPTION TERMINATION CARD*

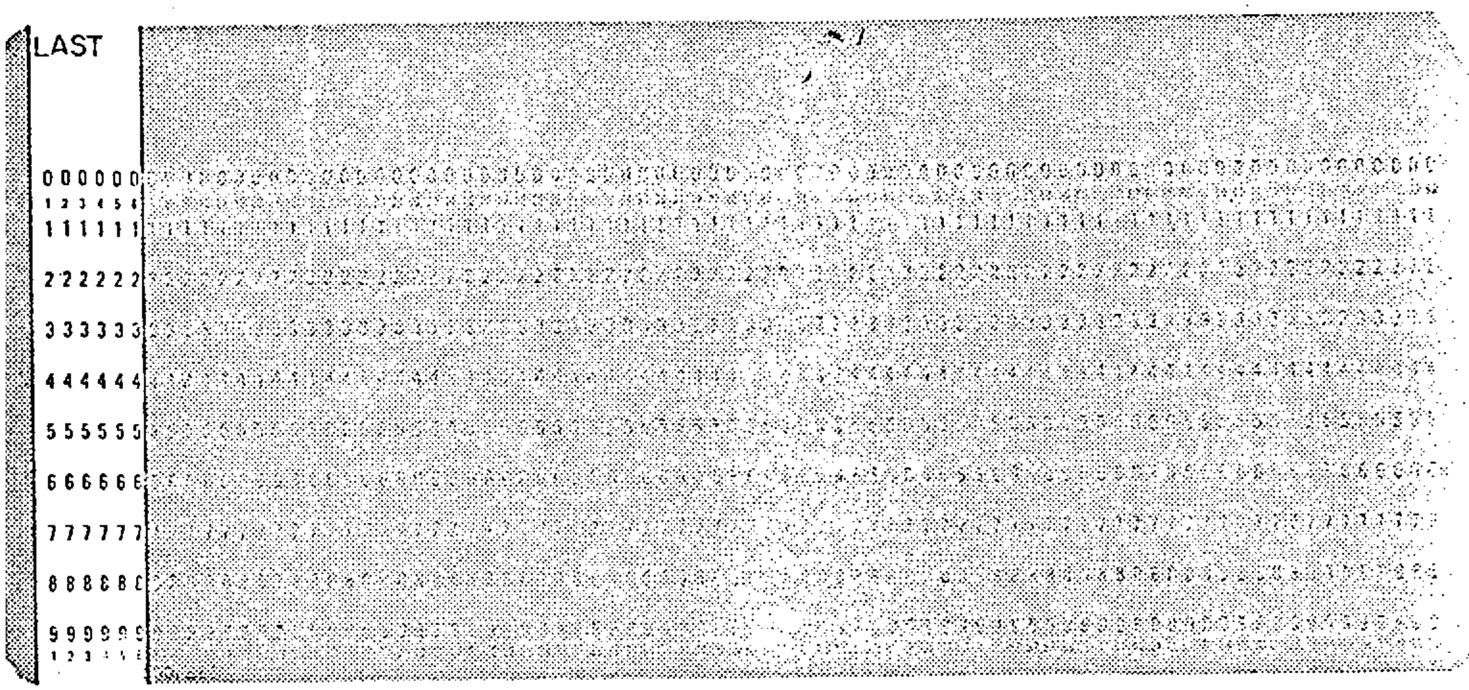
DATA



COLUMNS	FORMAT	DESCRIPTION
1-4	A4	The word "DATA" signifies the end of the option cards.

Note: Termination Card - This card must always be present, even if no option cards are used.

CASE TERMINATION CARD*



COLUMNS	FORMAT	DESCRIPTION
1-4	A4	The word "LAST" appearing here indicates that this is the last case. If left blank this card indicates that another case will follow.

Note: *CASE TERMINATION CARD - This card <u>must</u> always be present to terminate each case.

2.1.2.2 GEORGE Job Control Language and Hardware and Software Restrictions

2.1.2.2.1 Job Control Language

The GEORGE program may be executed by use of the LINKGO procedure as follows:

```
// EXEC LINKGO,REGION.GO=525K
//LINK.SYSLIN DD *
 INCLUDE LOADLIB(ZCMLDGRG)
 INCLUDE LOADLIB(ZCRGWTYP)
 ENTRY MAIN
/*
//GO.FT15F001 DD UNIT=2400-9,VOL=SER=BRESID,
//   DCB=(RECFM=VBS,LRECL=80,BLKSIZE=3204),
//   LABEL=(,BLP)
//GO.FT20F001 DD LABEL=(,BLP),UNIT=2400-7,
//   DCB=(DEN=1,RECFM=FB,LRECL=6,BLKSIZE=4092),
//   VOL=SER=PLOT1
//GO.DATA5 DD *

       The GEORGE Input Card deck goes here.

/*
```

Unit 15 is used to input the Binary Residual Tape.

Unit 20 is used for output of the SC4020 Plotter Driver tape.

2.1.2.2.2 Hardware and Software Restrictions

GEORGE requires a large scale IBM 360 computer with a minimum of 525K bytes of user accessable core, one 9 track tape drive, one 7 track tape drive, one card reader and one high speed printer.

The current GEORGE program is executable under version 18 of the IBM 360 operating system.

The compilation of GEORGE requires an IBM FORTRAN IV Level G compiler.

There are no GEORGE software or hardware restrictions other than that the above mentioned hardware and software be available and working properly.

2.1.2.3 GEORGE Example Job

The example job for GEORGE is included with Example Two for GEODYN in Volume 3, Section 4.2. Shown in this example is the normal mode of operation for the GEORGE program. However, as shown in Section 2.2.2.1, tape input may also be used for GEORGE rather than concatenating the GEORGE execution with a GEODYN execution.

2.1.2.4 GEORGE Error Messages

In addition to those error messages that may be printed by the WRDC SC4020 PLOT PACKAGE the following error messages may be printed during the execution of the GEORGE program.

a) ILLEGAL MEASUREMENT TYPE--SKIPPING TO NEXT CASE

b) ILLEGAL NETWORK NAME--SKIPPING TO NEXT CASE

c) ILLEGAL OPTION CARD--__________
REMAINING OPTIONS IGNORED--SKIPPING TO DATA

d) NO DATA OF THE TYPE SPECIFIED FOUND--SKIPPING TO NEXT CASE

e) TOO MANY OBSERVATIONS--REMAINDER IGNORED

With the exception of error e all of these messages are self-explanatory.

e) The GEORGE program will process a maximum of 4000 observations per case. Observations in excess of 4000 will be ignored.

2.1.3 GROUNDTRACK

GROUNDTRACK is a GEODYN support program which reads a subsatellite groundtrack tape written by GEODYN and plots, using the WRDC SC4020 PLOT PACKAGE, the geometry of satellite passes across the stations tracking the satellite. Only those passes on which tracking data is taken are written on the groundtrack tape. Plots from GROUNDTRACK may be obtained on the printer and/or an SC4020 Plotter Driver tape.

The following pages will describe in detail the setup and operation of the GROUNDTRACK support program of the GEODYN System.

2.1.3.1 GROUNDTRACK Input Cards

Specific functions of the GROUNDTRACK program may be requested by cards. The GROUNDTRACK Input Cards are separated into two categories:

- GROUNDTRACK Mandatory Cards -- these are cards that must be present for each case.

- GROUNDTRACK Option Cards -- these are cards that, as the name implies, are optional.

A set of these cards will define a case. More than one case may be present.

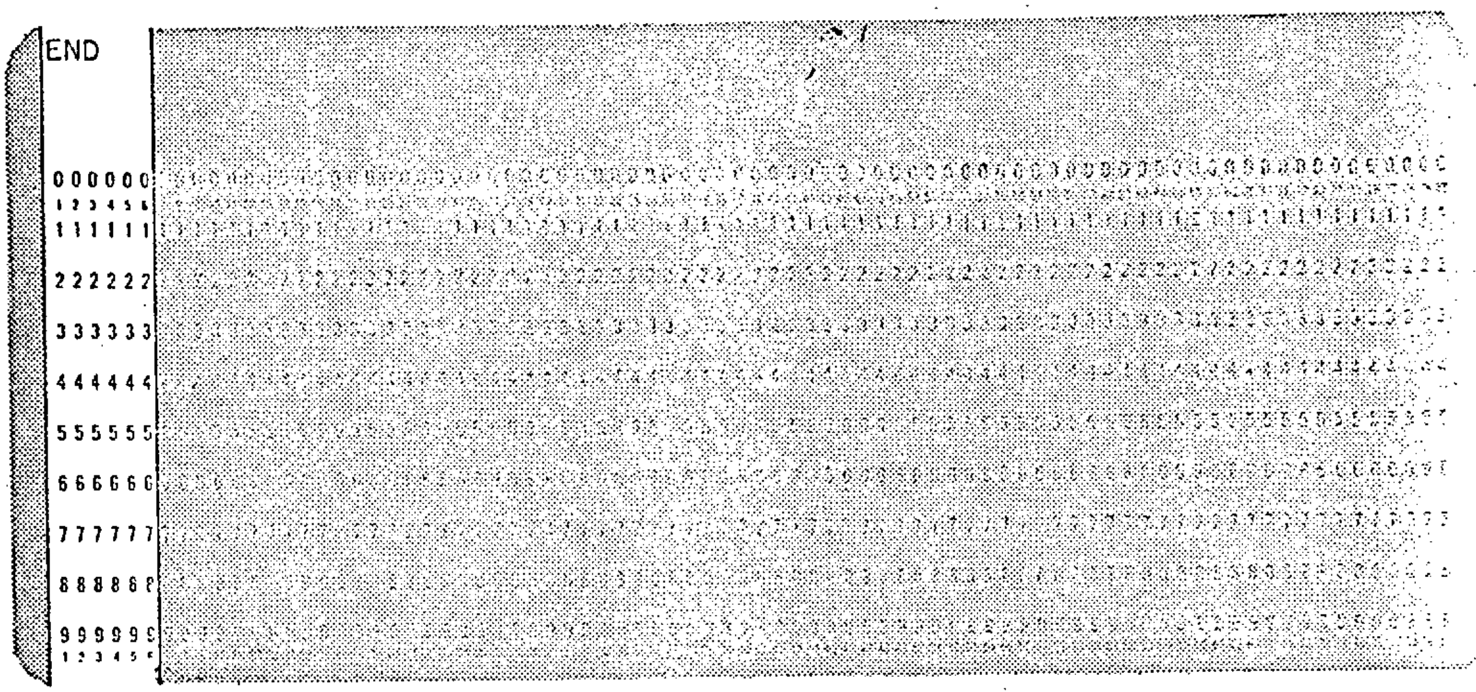
This section of the manual describes the format and usage of the GROUNDTRACK Input Cards.

STATION POSITION CARDS

COLUMNS	FORMAT	DESCRIPTION		
1-6	A6	Station name.		
7-10	I4	Station number.		
11	A1	Sign of Latitude		
12-13	I2	Degrees	Station	
14-15	I2	Minutes	Geodetic	ϕ
16-25	F10.5	Seconds	Latitude	
26-28	I3	Degrees	Station	
29-30	I2	Minutes	East	λ
31-40	F10.5	Seconds	Longitude	
41-50	F10.5	Station height in meters.		h

Note : Station position cards are mandatory for all stations for which plotting is requested. A maximum of 10 stations is permitted per case.

STATION POSITION
TERMINATION CARD

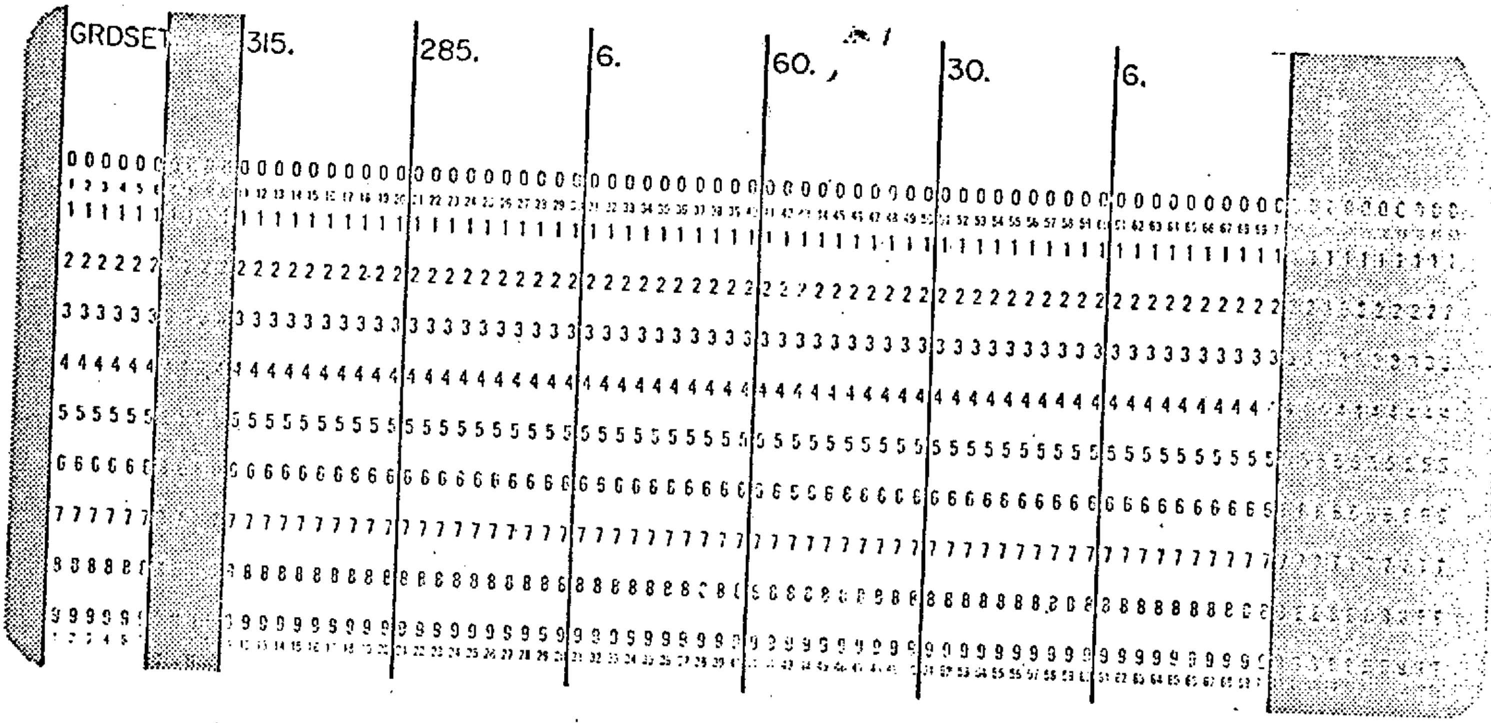


COLUMNS	FORMAT	DESCRIPTION
1-3	A3	The word "END" specified here indicates the end of the Station Position cards for the case.

This card must be present.

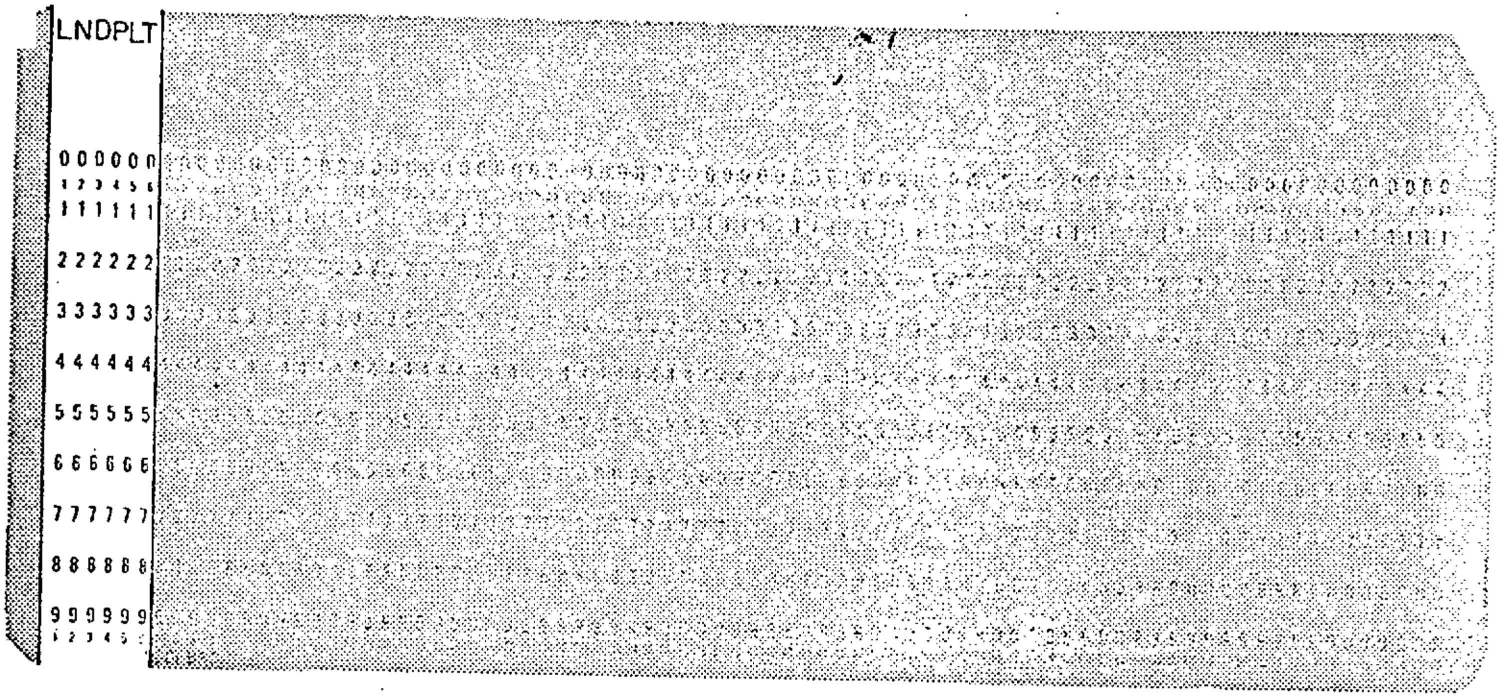
OPTION CARD

GRDSET



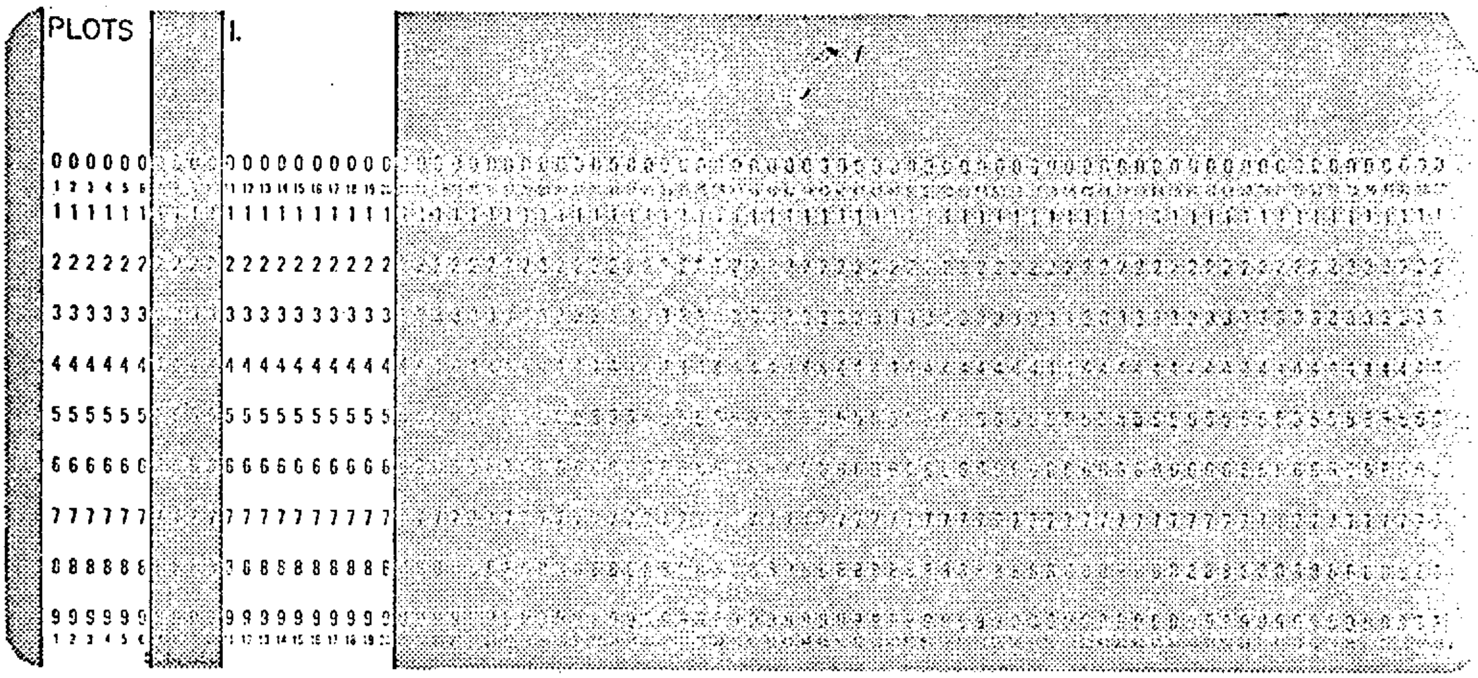
COLUMNS	FORMAT	DESCRIPTION
1-6	A6	The word "GRDSET" specified here indicates that this card will specify the grid parameters. If this card is not present, GROUNDTRACK will compute appropriate grid limits.
11-20	F10.5	Maximum longitude west of station.
21-30	F10.5	Minimum longitude east of station.
31-40	F10.5	Number of longitudinal grid intervals.
41-50	F10.5	Minimum latitude.
51-60	F10.5	Maximum latitude.
61-70	F10.5	Number of latitudinal grid intervals.

OPTION CARD
LNDPLT



COLUMNS	FORMAT	DESCRIPTION
1-6	A6	The word "LNDPLT" specified here indicates the plot is to be superimposed over the land contour plot for the region of the earth specified.

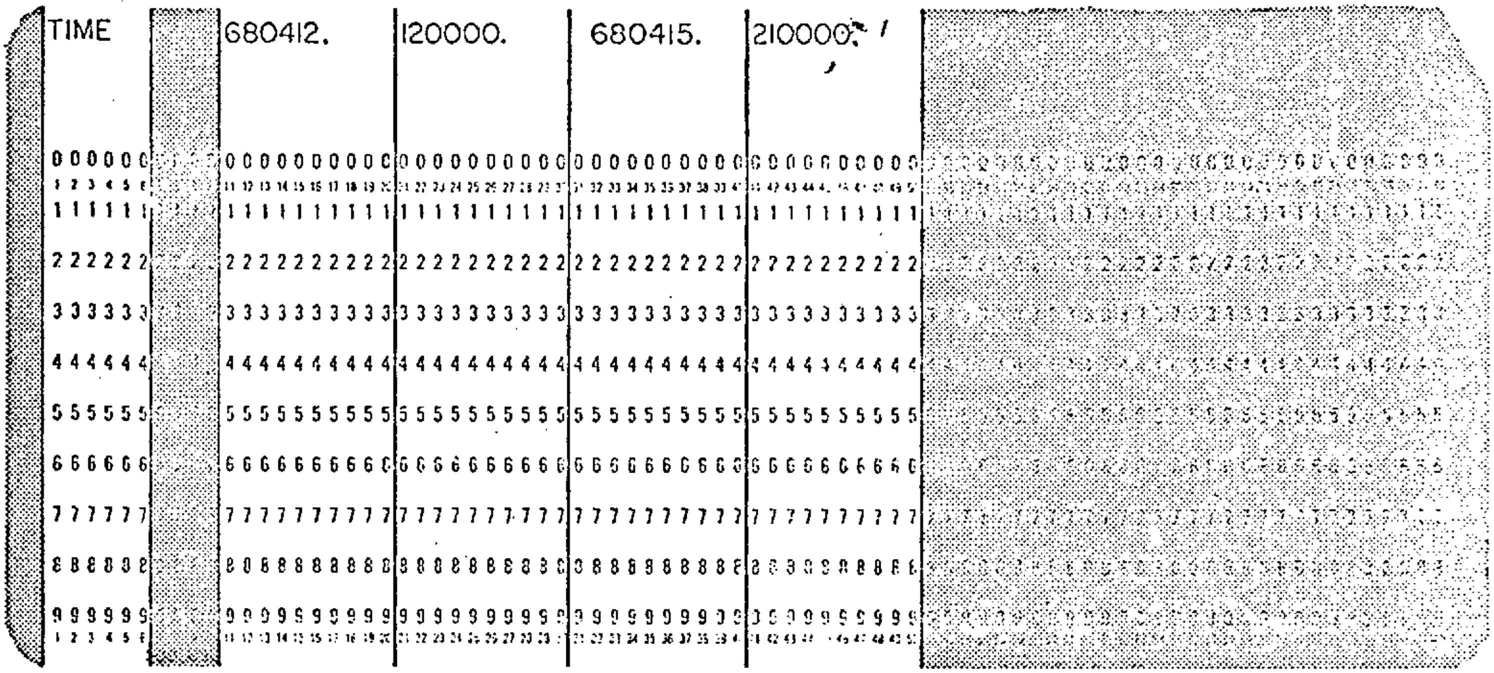
OPTION CARD
PLOTS



COLUMNS	FORMAT	DESCRIPTION
1-5	A5	The word "PLOTS" specified here indicates that groundtrack plots are requested with the option specified on this card.
11-20	F10.5	=0. Printer plots only. =1. Printer plots and an SC4020 Plotter Driver tape.

OPTION CARD

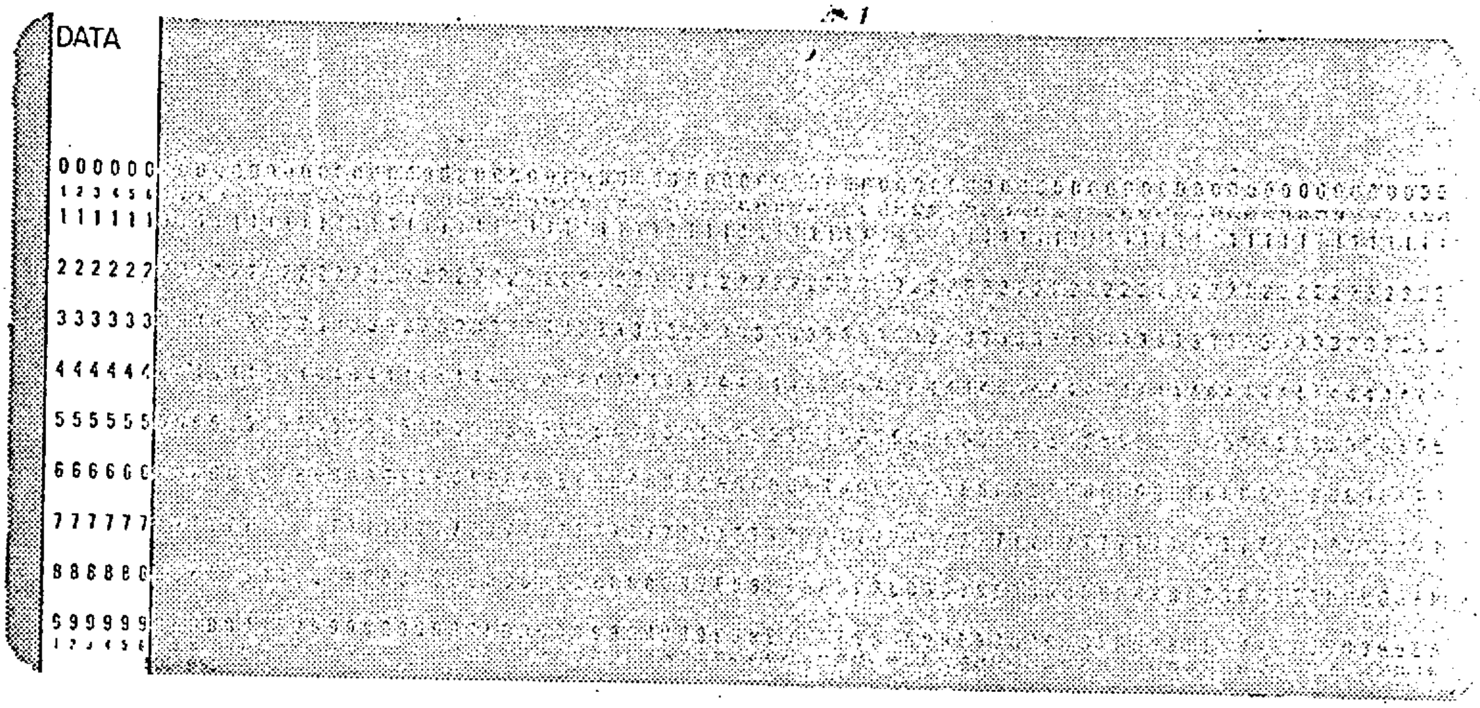
TIME



COLUMNS	FORMAT	DESCRIPTION
1-4	A4	The word "TIME" specified here indicates that the groundtrack plot times will be specified on this card.
11-20	F10.5	Start date in YYMMDD.
21-30	F10.5	Start time in HHMM.
31-40	F10.5	Stop date in YYMMDD.
41-50	F10.5	Stop time in HHMM.

OPTION TERMINATION CARD*

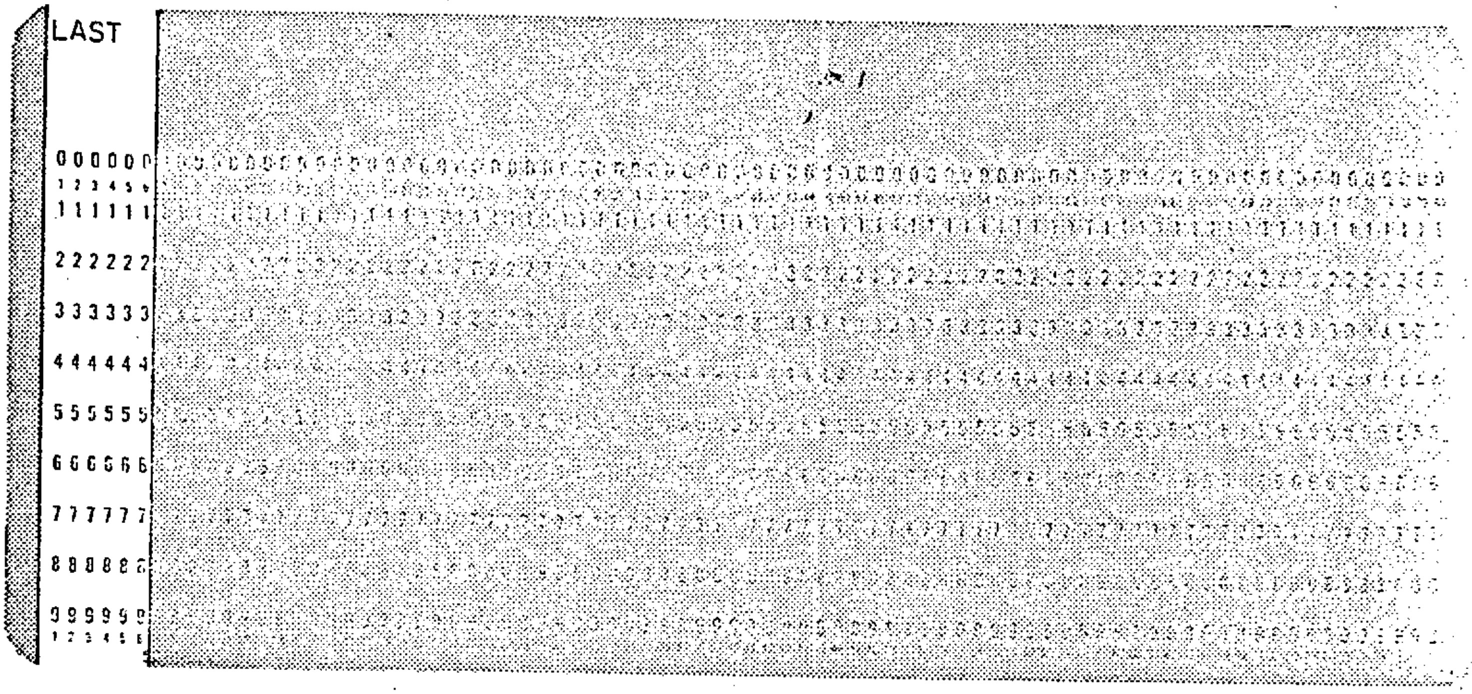
DATA



COLUMNS	FORMAT	DESCRIPTION
1-4	A4	The word "DATA" specified here indicates the end of the optional GROUNDTRACK Input Cards.

Note: *OPTION TERMINATION CARD--This card must always be present for each arc.

CASE TERMINATION CARD*



COLUMNS	FORMAT	DESCRIPTION
1-4	A4	The word "LAST" appearing here indicates that this is the last case. If left blank this card indicates that another case will follow.

Note: *CASE TERMINATION CARD--This card <u>must</u> always be present to terminate each case.

2.1.3.2 GROUNDTRACK Job Control Language and Hardware and Software Restrictions

2.1.3.2.1 Job Control Language

The GROUNDTRACK program may be executed by use of the LINKGO procedure as follows:

```
// EXEC LINKGO,REGION.GO=500K
//LINK.SYSLIN DD *
 INCLUDE LOADLIB(ZCMLDGRK)
 INCLUDE LOADLIB(ZCRJGWRL)
 INCLUDE LOADLIB(ZCRGWTYP)
 ENTRY MAIN
/*
//GO.FT11F001 DD UNIT=2400-9,VOL=SER=GTRACK,
//  DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200),
//  LABEL=(,BLP)
//GO.FT20F001 DD UNIT=2400-7,LABEL=(,BLP),
//  DCB=(DEN=1,RECFM=FB,LRECL=6,BLKSIZE=4092),
//  VOL=SER=PLOT3
//GO.DATA5 DD *

      The GROUNDTRACK Input Card deck goes here.

/*
```

Unit 11 is used for input of the groundtrack tape.

Unit 20 is used for output of the SC4020 Plotter Driver tape.

2.1.3.2.2 Hardware and Software Restrictions

GROUNDTRACK requires a large scale IBM 360 computer with a minimum of 500K bytes of user accessable core, one 9 track tape drive, one 7 track tape drive, one card reader and one high speed printer.

The current GROUNDTRACK program is executable under version 18 of the IBM 360 operating system.

The compilation of GROUNDTRACK requires an IBM FORTRAN IV Level G compiler.

There are no GROUNDTRACK software or hardware restrictions other than that the above mentioned hardware and software be available and working properly.

2.1.3.3 GROUNDTRACK Example Job

The example job for GROUNDTRACK is included with Example One for GEODYN in Section 4.1 of Volume 3. Shown in this example is the normal mode of operation for the GROUNDTRACK program. However, as shown in Section 2.1.3.2.1, tape input may also be used for GROUNDTRACK rather than concatenating the GROUNDTRACK execution with a GEODYN execution.

2.1.3.4 GROUNDTRACK Error Messages

In addition to those error messages that may be printed by the WRDC SC4020 PLOT PACKAGE the following error message may be printed during the execution of the GROUNDTRACK program.

ILLEGAL OPTION CARD ___ IGNORED REMAINING OPTIONS, EXECUTION CONTINUING

2.1.4 WRDC SC4020 PLOT PACKAGE

The WRDC SC4020 PLOT PACKAGE is a group of subroutines which may be called from FORTRAN programs and which may be used to plot information. The WRDC SC4020 PLOT PACKAGE has no main program and therefore no setup and operation procedures. However, since DELTA, GEORGE, and GROUNDTRACK all use the WRDC SC4020 PLOT PACKAGE, it is appropriate to herein describe all error messages which may be printed by the WRDC SC4020 PLOT PACKAGE during the execution of the above mentioned GEODYN support programs.

WRDC SC4020 PLOT PACKAGE
ERROR MESSAGES

The WRDC SC4020 PLOT PACKAGE prints three different error messages, all of which result from inappropriate PLOT PACKAGE input. Consequently, any errors which have occured will be caused by errors in the calling programs (i.e. DELTA, GEORGE, and GROUNDTRACK). The proper response to all of these messages is to examine the input to these calling programs for misplaced, out-of-order, or mispunched input cards or incorrectly specified input tape parameters.

The WRDC SC4020 PLOT PACKAGE error messages are:

1) SETGRD ARGUMENTS OUT OF RANGE -- LIMITS NOT RESET

2) EMPTY ARRAY OR ALL ITEMS EQUAL IN QUICKY

3) ////////

Slashes in upper right corner of a plot indicate an attempt was made to plot outside of the device limits.

These errors always result in the following program action.

1) Plot frame advance.
2) No program corrective action.
3) No program corrective action.

2.2 GEODYN DATA HANDLING SUPPORT PROGRAMS

The GEODYN Data Handling Support Programs are used for data management. The five data handling support programs are:

- DODS SORT-MERGE
- GEOS SORT-MERGE
- EPHEMERIS TAPE GENERATOR
- ORB1 CONVERSION (9-7 track)
- TDIF TABLE GENERATOR

The operation of these programs is described in the following pages.

2.2.1 DODS SORT-MERGE

DODS SORT-MERGE reads an unspecified number of data tapes in DODS Data Tape Format assuming these tapes to be one continuous file not in time order. Scratch files are written containing strings of time ordered data which are iteratively merged with other strings decreasing the number of strings by half until one time ordered string of data in DODS Data Tape Format exists.

There is no card input to the DODS SORT-MERGE program. On the following pages will be described

- Job Control Language and Job Submittal and
- DODS SORT-MERGE Printer Output.

2.2.1.1 Job Control Language and Job Submittal

To submit a DODS SORT-MERGE job requires only the preparation of the job control language (JCL) and submittal of the job to the computer with the proper job identification slip.

DODS SORT-MERGE may be invoked by usage of the FORTRAN and LINKGO procedures.

```
// EXEC FORTRAN
//SOURCE.SYSIN DD *

<<<The DODS SORT-MERGE FORTRAN deck goes here.<<<

/*
// EXEC LINKGO,REGION.GO=400K,TIME=1440
//GO.FT10F001 DD UNIT=2400-9,LABEL=(,BLP),
// DCB=(RECFM=VBS,LRECL=104,BLKSIZE=1044),
// VOL=SER=(INPUT1,INPUT2,INPUT3,......)
//GO.FT11F001 DD UNIT=2400-4,VOL=SER=OUTPUT,
// DCB=(RECFM=VBS,LRECL=104,BLKSIZE=8324),
// LABEL=(,BLP)
//GO.FT20F001 DD UNIT=DISK,SPACE=(28008,100),
// DCB=(RECFM=VST,BLKSIZE=28008)
//GO.FT21F001 DD UNIT=DISK,SPACE=(28008,100),
// DCB=(RECFM=VST,BLKSIZE=28008)
//GO.FT22F001 DD UNIT=DISK,SPACE=(28008,100),
// DCB=(RECFM=VST,BLKSIZE=28008)
//GO.FT23F001 DD UNIT=DISK,SPACE=(28008,100),
// DCB=(RECFM=VST,BLKSIZE=28008)
```

Data tape input is on unit 10.
Data tape output is on unit 11.
Units 20,21,22,23 are used for temporary scratch data storage and will each hold 25,000 observations.

The combined disk space requested by units 20-23 comes to a total of 1600 tracks. To allocate 1600 tracks is extremely difficult; therefore, if more than 25,000 observations are to be processed, units 20-23 should be specified as 9-track, high density tapes with the following DCB parameters.

```
DCB=(RECFM=VBS,LRECL=28008,BLKSIZE=28012,DEN=3)
```

DODS SORT-MERGE EXAMPLE SETUP DECK

```
//...      JOB ...
// EXEC FORTRAN
//SOURCE.SYSIN DD *

         THE DODS SORT-MERGE FORTRAN DECK GOES HERE

/*
// EXEC LINKGO,REGION.GO=400K
//GO.FT10F001 DD UNIT=2400-9,DCB=(RECFM=VBS,LRECL=104,BLKSIZE=1044,
//   DEN=3),LABEL=(,BLP),VOL=SER=INPUT
//GO.FT11F001 DD UNIT=2400-9,DCB=(RECFM=VBS,LRECL=104,BLKSIZE=8324,
//   DEN=3),LABEL=(,BLP),VOL=SER=SCRTCH
//GO.FT20F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=28004),
//   SPACE=(28004,10)
//GO.FT21F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=28004),
//   SPACE=(28004,10)
//GO.FT22F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=28004),
//   SPACE=(28004,10)
//GO.FT23F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=28004),
//   SPACE=(28004,10)
```

2.2.1.2 DODS Sort-Merge Printer Output

During normal operation DODS SORT-MERGE prints the number of strings of data before each merge process.

Only one error message may be printed and that message is

NO SORT INPUT

The following example job sorted and merged a data tape with 2130 measurements in random time order.

The core and time required were

CORE = 396k CPU = 0.91m I/O = 0.19m

```
IEF285I   MPLOAD                                                   KEPT
IEF285I   VOL SER NOS= MPLOAD.
IEF285I   SYS1.DUMMY                                               KEPT
IEF285I   VOL SER NOS= M2SYS4.
IEF285I   SYS1.DUMMY                                               KEPT
IEF285I   VOL SER NOS= M2SYS4.
IEF285I   SYS1.FORTLIB                                             KEPT
IEF285I   VOL SER NOS= M2OR41.
IEF285I   SYS1.PLILIB                                              KEPT
IEF285I   VOL SER NOS= M2SYS4.
IEF285I   SYS1.COBCLIB                                             KEPT
IEF285I   VOL SER NOS= M2SYS2.
IEF285I   SYS1.SSP                                                 KEPT
IEF285I   VOL SER NOS= M2SYS2.
IEF285I   SYS1.COBLIB                                              KEPT
IEF285I   VOL SER NOS= M2SYS2.
IEF285I   SYS70351.T161145.RV000.ZCTVMDSR.LOADGO                   PASSED
IEF285I   VOL SER NOS= M2SCR2.
IEF285I   SYS70351.T161145.RV000.ZCTVMDSR.R0000579                 SYSOUT
IEF285I   VOL SER NOS= M2SCR2.
IEF285I   SYS70351.T161145.RV000.ZCTVMDSR.R0000500                 DELETED
IEF285I   VOL SER NOS= M2SCR2.
IEF285I   SYS70351.T161145.RV000.ZCTVMDSR.COJMOD                   DELETED
IEF285I   VOL SER NOS= M2OR41.
IEF373I STEP /LINK     / START 70351.2105
IEF374I STEP /LINK     / STOP  70351.2105 CPU   00MIN 00.29SEC MAIN 154K LCS    0K
                                                       STEP TIME =      .10 MINS=(CPU=      .00.IO=       .10)
- STEP 02 -
                                 IO IN SECS. DISK=    4.45.DRUM=    1.90.TAPE=      .00.CELL=      .00.OTHR=    .00
XXGO      EXEC   PGM=*.LINK.SYSLMOD.COND=(5.LT).REGION=100K          00002000
XXFT05F001 DD  DDNAME=DATA5                                          00002100
XXFT06F001 DD SYSOUT=A.DCB=(RECFM=VBA.LRECL=137.BLKSIZE=7265)        00002200
XXFT07F001 DD DUMMY.DCB=(RECFM=FB.LRECL=80.BLKSIZE=3200)            X00002300
XX*              TO GET A DECK. ADD THE FOLLOWING CARD TO YOUR DECK.  X00002400
XX*              //GO.FT07F001 DD DDNAME=DECK.SYSOUT=B                00002500
XXSYSPRINT DD SYSOUT=A.DCB=(RECFM=VBA.LRECL=137.BLKSIZE=7265)        00002600
//GO.FT10F001 DD UNIT=2400-9.DCB=(RECFM=VBS.LRECL=104.BLKSIZE=1044.
//  DEN=3).LABEL=(.BLP).VOL=SER=TC5046
//GO.FT11F001 DD UNIT=2400-9.DCB=(RECFM=VBS.LRECL=104.BLKSIZE=8324.
//  DEN=3).LABEL=(.BLP).VOL=SER=SCRTCH
//GO.FT20F001 DD UNIT=DISK.DCB=(RECFM=VST.BLKSIZE=28004).
//  SPACE=(28004.10)
//GO.FT21F001 DD UNIT=DISK.DCB=(RECFM=VST.BLKSIZE=28004).
//  SPACE=(28004.10)
//GO.FT22F001 DD UNIT=DISK.DCB=(RECFM=VST.BLKSIZE=28004).
//  SPACE=(28004.10)
//GO.FT23F001 DD UNIT=DISK.DCB=(RECFM=VST.BLKSIZE=28004).
//  SPACE=(28004.10)
//
IEF236I ALLOC. FOR ZCTVMDSR GO
IEF237I 233    ALLOCATED TO PGM=*.DD
IEF237I 234    ALLOCATED TO FT06F001
IEF237I 234    ALLOCATED TO SYSPRINT
IEF237I 0C3    ALLOCATED TO FT10F001
IEF237I 0C4    ALLOCATED TO FT11F001
IEF237I 234    ALLOCATED TO FT20F001
IEF237I 233    ALLOCATED TO FT21F001
IEF237I 232    ALLOCATED TO FT22F001
IEF237I 234    ALLOCATED TO FT23F001

        7 STRINGS
        4 STRINGS
        2 STRINGS
```

2.2.2 GEOS SORT-MERGE

GEOS SORT-MERGE reads an unspecified number of data tapes in GEOS Data Tape Format assuming these tapes to be one continuous file not in time order. Scratch files are written containing strings of time ordered data which are iteratively merged with other strings decreasing the number of strings by half until one time ordered string of data in GEOS Data Tape Format exists.

There is no card input to the GEOS SORT-MERGE program. On the following pages will be described

- Job Control Language and Job Submittal.

- GEOS SORT-MERGE Printer Output.

2.2.2.1 Job Control Language and Job Submittal

To submit a GEOS SORT-MERGE job requires only the preparation of the job control language (JCL) and submittal of the job to the computer with the proper job identification slip.

GEOS SORT-MERGE may be invoked by usage of the FORTRAN and LINKGO procedures.

```
// EXEC FORTRAN
//SOURCE.SYSIN DD *
```

The GEOS SORT-MERGE FORTRAN deck goes here.

```
/*
// EXEC LINKGO,REGION.GO=250K,TIME=1440
//GO.FT10F001 DD UNIT=(2400-9,,2),LABEL=(,BLP),
// DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200),
// VOL=SER=(INPUT1,INPUT2,INPUT3,......)
//GO.FT11F001 DD UNIT=2400-4,LABEL=(,BLP),
// DCB=(RECFM=FBS,LRECL=80,BLKSIZE=8000),
// VOL=SER=OUTPUT
//GO.FT20F001 DD UNIT=DISK,SPACE=(19008,100),
// DCB=(RECFM=VST,BLKSIZE=19008)
//GO.FT21F001 DD UNIT=DISK,SPACE=(19008,100),
// DCB=(RECFM=VST,BLKSIZE=19008)
//GO.FT22F001 DD UNIT=DISK, SPACE=(19008,100),
// DCB=(RECFM=VST,BLKSIZE=19008)
//GO.FT23F001 DD UNIT=DISK, SPACE=(19008,100),
// DCB=(RECFM=VST,BLKSIZE=19008)
```

Data tape input is on unit 10.
Data tape output is on unit 11.
Units 20, 21, 22, 23 are used for temporary scratch
data storage and will each hold 25,000 observations.

The combined disk space requested by units 20-23 comes to a total of 1100 tracks. To allocate 1100 tracks is extremely difficult; therefore, if more than 25,000 observations are to be processed, units 20-23 should be specified as 9-track, high density tapes with the following DCB parameters.

DCB=(RECFM=VBS,LRECL=19008,BLKSIZE=19012,DEN=3)

GEOS SORT-MERGE EXAMPLE SETUP DECK

```
//...      JOB ...
// EXEC FORTRAN
//SOURCE.SYSIN DD *

          THE GEOS SORT-MERGE FORTRAN DECK GOES HERE

/*
// EXEC LINKGO,REGION.GO=275K
//GO.FT11F001 DD UNIT=2400-9,DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000),
//  LABEL=(,BLP),VOL=SER=SCRTCH
//GO.FT20F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=19008),
//  SPACE=(19008,10)
//GO.FT21F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=19008),
//  SPACE=(19008,10)
//GO.FT22F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=19008),
//  SPACE=(19008,10)
//GO.FT23F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=19008),
//  SPACE=(19008,10)
//GO.FT10F001 DD *

          THE GEOS DATA CARDS TO BE MERGED GO HERE

/*
```

2.2.2.2 GEOS Sort-Merge Printer Output

During normal operation GEOS SORT-MERGE prints the number of strings data before each merge process.

Only one error message may be printed and that message is

NO SORT INPUT

The following example job sorted and merged 720 GEOS data cards in random time order.

The core and time required were

CORE = 256k   CPU = 0.12m   I/O = 0.08m

```
IEF285I   VMLOAD                                   KEPT
IEF285I   VOL SER NOS= MOLOAD.
IEF285I   SYS1.DUMMY                               KEPT
IEF285I   VOL SER NOS= M2SYS4.
IEF285I   SYS1.DUMMY                               KEPT
IEF285I   VOL SER NOS= M2SYS4.
IEF285I   SYS1.FORTLIB                             KEPT
IEF285I   VOL SER NOS= M20HM1.
IEF285I   SYS1.PLILIB                              KEPT
IEF285I   VOL SER NOS= M3SYS4.
IEF285I   SYS1.COBLIB                              KEPT
IEF285I   VOL SER NOS= M2SYS2.
IEF285I   SYS1.SSP                                 KEPT
IEF285I   VOL SER NOS= M2SYS2.
IEF285I   SYS1.COBLIB                              KEPT
IEF285I   VOL SER NOS= M2SYS2.
IEF285I   SYS72351.T161132.RV000.ZCTVMSRT.LOADMOD  PASSED
IEF285I   VOL SER NOS= M2SCR6.
IEF285I   SYS72351.T161132.SV000.ZCTVMSRT.R0000975 SYSOUT
IEF285I   VOL SER NOS= M2SCR5.
IEF285I   SYS72351.T161132.RV000.ZCTVMSRT.R0000976 DELETED
IEF285I   VOL SER NOS= M20HM1.
IEF285I   SYS72351.T161132.RV000.ZCTVMSRT.GOSET    DELETED
IEF285I   VOL SER NOS= M20HM1.
IEF373I STEP /LINK    / START 72351.1934
IEF374I STEP /LINK    / STOP  72351.1934 CPU   0MIN 03.27SEC MAIN 154K LCS    OK
- STEP 02 -                                   STEP TIME =    .12 MINS=(CPU=    .09,IO=    .10)
                   IO IN SECS. DISK=    3.40,DRUM=     2.64,TAPE=     .00,CELL=     .00,OTHR=     .00
XXGO     EXEC  PGM=*.LINK.SYSLMOD,COND=(4,LT),REGION=132K        00002000
XXFT05F001 DD  DDNAME=DATA5                                       00002100
XXFT06F001 DD  SYSOUT=A,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=7265)    00002200
XXFT07F001 DD  DUMMY,DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)        X00002300
XX*          TO GET A DECK, ADD THE FOLLOWING CARD TO YOUR DECK  X00002400
XX*             //GO.FT07F001 DD DSNAME=DECK,SYSOUT=B             00002500
XXFT09F001 DD  SYSOUT=A,DCB=(RECFM=VBA,LRECL=137,BLKSIZE=7265)    00002600
//GO.FT11F001 DD UNIT=2400-9,DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000),
//  LABEL=(,BLP),VOL=SER=SCRTCH
//GO.FT20F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=19008),
//  SPACE=(19008,10)
//GO.FT21F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=19008),
//  SPACE=(19008,10)
//GO.FT22F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=19008),
//  SPACE=(19008,10)
//GO.FT23F001 DD UNIT=DISK,DCB=(RECFM=VST,BLKSIZE=19008),
//  SPACE=(19008,10)
//DATA5 DD *
//
IEF236I ALLOC. FOR ZCTVMSRT GO
IEF237I 330   ALLOCATED TO PGM=*.DD
IEF237I 332   ALLOCATED TO FT10F001
IEF237I 330   ALLOCATED TO SYSPRINT
IEF237I 336   ALLOCATED TO FT11F001
IEF237I 333   ALLOCATED TO FT20F001
IEF237I 334   ALLOCATED TO FT21F001
IEF237I 333   ALLOCATED TO FT22F001
IEF237I 334   ALLOCATED TO FT23F001
IEF237I 333   ALLOCATED TO FT10F001

*** NUMBER OF STRINGS=   2
```

2.2.3 EPHEMERIS TAPE GENERATOR

The EPHEMERIS TAPE GENERATOR generates geocentric lunar positions at half day intervals, heliocentric positions of the Earth-moon barycenter, and the planets, Venus, Mars, Jupiter and Saturn at four day intervals and the nutation in obliquity at half day intervals.

2.2.3.1 Input Card

The EPHEMERIS TAPE GENERATOR can read a maximum of three tapes since the JPL ephemeris is broken into three pieces. The input card of the EPHEMERIS TAPE GENERATOR consists of specification of start and stop times for taking information from each input tape. This is done in the following manner:

Columns	Format	Description
1-12	F12.5	Julian start time of ephemeris.
13-24	F12.5	Julian stop time for taking information from first input tape.
25-36	F12.5	Julian start time for taking information from second input tape. If zero, second input tape will not be read. If second tape is to be read, start time must be the same as the stop time of first tape.
37-48	F12.5	Julian stop time for taking information from second input tape.
49-60	F12.5	Julian start time for taking information from third input tape. If zero, third tape will not be read. If third tape is to be read, start time must be the same as the stop time of second tape.
61-72	F12.5	Julian stop time for taking information from third input tape.

First, second and third tapes must use units 12, 13, and 14, respectively.

2.2.3.2 Job Control Language and Job Submittal

To submit an EPHEMERIS TAPE GENERATOR job requires only the preparation of the job control language (JCL) and input card.

The EPHEMERIS TAPE GENERATOR may be invoked using the following procedure:

```
//----JOB----
// EXEC FORTRAN
//SYSIN DD *

<<<The EPHEMERIS TAPE GENERATOR Source Deck goes here.<<<

/*
// EXEC LINKGO,REGION=250K
//GO.FT10F001 DD UNIT=2400-9,DCB=(RECFM=VBS,
//    LRECL=436,BLKSIZE=7294,DEN=3),LABEL=(1,BLP),
//    VOL=SER=OUTPUT
//GO.FT12F001 DD UNIT=2400-9,DCB=(RECFM=VBS,
//    LRECL=7456,BLKSIZE=29828,DEN=3),LABEL=(,BLP),
//    VOL=SER=INPUT1
//GO.SYSUDUMP DD SYSOUT=C,SPACE=(CYL,(10,2))
//GO.DATA5 DD *

<<<The EPHEMERIS TAPE GENERATOR Input Card goes here.<<<

/*
```

2.2.4 ORB1 CONVERSION (9-7) Tracks

The ORB1 CONVERSION program reads a double precision, 9-track, IBM 360 ORB1 tape written by GEODYN and writes a single precision, 7-track, IBM 7094 ORB1 tape in the same format.

There is no card input and no printer output for the ORB1 CONVERSION program and therefore, complete program operation is described by Job Control Language (JCL). The JCL necessary is described below and requires only to be submitted to the computer with the proper job identification slip.

ORB1 CONVERSION may be invoked by usage of the FORTRAN and LINKGO procedures.

```
// EXEC FORTRAN
//SOURCE.SYSIN DD *
```

The ORB1 CONVERSION FORTRAN deck goes here.

```
/*
//  EXEC LINKGO
//GO.FT10F001 DD UNIT=2400-9,LABEL=(,BLP).
// DCB=(RECFM=VBS,LRECL=2804,BLKSIZE-2808),
// VOL=SER=INPUT9
//GO.FT11F001 DD UNIT=2400-7,LABEL=(,BLP),
// DCB=(RECFM=FB,BLKSIZE=2100,LRECL=21)
// VOL=SER=OUTPUT
```

Unit 10 is the IBM 360, 9-track, ORB1 tape input.
Unit 11 is the IBM 7094, 7-track, ORB1 tape output.

The example job for ORB1 CONVERSION is included with Example Three for GEODYN in Volume 3, Section 4.3.

2.2.5 TDIF TABLE GENERATOR

The TDIF TABLE GENERATOR generates tabular differences between the time systems A.1 and UT1. It reads tables showing the differences between systems UT1 and UTC (UT1-UTC) which are obtained from B.I.H. The tables require continual up-dating as this information is received directly from B.I.H.

2.2.5.1 Data Deck

The values of UT1-UTC are received from B.I.H. on Circular D.

The values of UT1-UTC are input to the program at 10-day intervals in the following manner:

Columns	Format	Description
1-6	I6	Date in YYMMDD
7-16	F10.5	Value in seconds of UT1-UTC as given on B.I.H. circular D.

An example of this circular is given on the following page.

The program then punches the values of A.1-UT1 on cards in tabular form.

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

Circular D69
Paris, 1972 August 1

1 - UNIVERSAL TIME AND COORDINATES OF THE POLE

Date	J.D.	smoothed values				raw values			
(Oh UT) 1972	2400000.5 +	x 0"001	y 0"000	UT2-UTC 0.0001s	UT1-UTC 0.0001s	x 0"001	y 0"001	UT1-UTC 0.0001s	UT1-IA s
June 1	41 469	-145	+356	-5252	-5553	-151	+341	-5541	-10.555
6	474	-134	+366	-5416	-5710	-129	+371	-5743	571
11	479	-120	+376	-5579	-5861	-114	+359	-5872	586
16	484	-105	+385	-5740	-6006	-.92	+374	-6019	600
21	489	- 89	+394	-5899	-6145	- 92	+421	-6165	614
26	494	- 72	+402	-6057	-6279	- 51	+382	-6263	627
July 1	499	- 54	+409	+3786	+3591	- 58	+407	+3616	640

IAT-UTC is exactly 10s in June 1972
IAT-UTC is exactly 11s since 1972 July 1st, Oh 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	HBG	0	OLB5	(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 i. Ref. CCIR Recommendation 458, 1970)

a - From LORAN-C and Television pulses receptions

Date 1972	June 11	June 21	July 1
J.D. 2400000.5 +	41 479	41 489	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
OP (Paris)	+ 1.6	+ 1.6	+ 1.6
NBS (Boulder)	- 2.4	- 2.5	- 2.7
RGO (Herstmonceux)	+ 3.2	+ 3.8	+ 4.4
NRC (Ottawa)	+ 0.8	+ 0.9	+ 1.2
FOA (Stockholm)	+ 23.9	+ 26.3	+ 28.6
DHI (Hamburg)	- 16.9	- 15.3	- 13.9
ON (Neuchâtel)	+ 20.6	+ 20.7	+ 20.6

P. T. O.

2.2.5.2 Job Control Language and Job Submittal

To submit a TDIF TABLE GENERATOR job requires only the preparation of the job control language (JCL) and the data deck.

The TDIF TABLE GENERATOR may be invoked using the following procedure:

```
//----JOB----
// EXEC FORTRANH,PARM='ID,OPT=2'
//SOURCE.SYSIN DD *

<<<THE TDIF TABLE GENERATOR Fortran deck goes here.<<<

// EXEC LOADER
//GO.DATAS DD *

<<<The TDIF TABLE GENERATOR data deck goes here<<<

/*
```