NASA TECHNICAL MEMORANDUM

LONGITUDINAL AND TRANSVERSE MAGNETIC FIELD PROGRAM PROCEDURE AND DETAILED SPECIFICATION (FOR SIGMA 5)

By Caroline K. Wang
Space Sciences Laboratory

August 1980

NASA

George C. Marshall Space Flight Center
Marshall Space Flight Center, Alabama

MSFC - Form 3190 (Rev June 1971)

TECHNICAL REPORT STANDARD TITLE PAGE

1. REPORT NO.	2. GOVERNMENT ACCESSION NO.	3. RECIPIENT'S CATALOG NO.
NASA TM-78292		

4. TITLE AND SUBTITLE	5. REPORT DATE
Longitudinal and Transverse Magnetic Field Program Procedure and Detailed Specification (For Sigma 5)	August 1980
	6. PERFORMING ORGANIZATION CODE
7. AUTHOR(S) Caroline K. Wang	8. PERFORMING ORGANIZATION REPORT #
9. PERFORMING ORGANIZATION NAME AND ADDRESS George C. Marshall Space Flight Center Marshall Space Flight Center, Alabama 35812	10. WORK UNIT NO.
	11. CONTRACT OR GRANT NO.
12. SPONSORING AGENCY NAME AND ADDRESS National Aeronautics and Space Administration Washington, D.C. 20546	13. TYPE OF REPORT & PERIOD COVERED Technical Memorandum
	14. SPONSORING AGENCY CODE

15. SUPPLEMENTARY NOTES

Prepared by Space Sciences Laboratory, Science and Engineering Directorate

16. ABSTRACT

This document presents a computer program and procedure for plotting the contour of the data transferred from the Marshall Space Flight Center solar magnetograph. The plotted data then can be easily compared with solar data from other sources, such as the Solar Maximum Mission (SMM).

17. KEY WORDS	18. DISTRIBUTION STATEMENT
	Caroline K. Wang Unclassified—Unlimited

19. SECURITY CLASSIF. (of this report)	20. SECURITY CLASSIF. (of this page)	21. NO. OF PAGES	22. PRICE
Unclassified	Unclassified	95	NTIS

TABLE OF CONTENTS

NASA TECHNICAL MEMORANDUM

1.0 PURPOSE

The purpose of the Longitudinal and Transverse programs is to plot the contour of the data which have been transferred from the Marshall Space Flight Center (MSFC) solar magnetograph. The data contain three different files:

1A1B For circular polarization

1A represents left circular; 1B represents right circular.

2A2B, 3A3B } For linear polarization

1.1 The Longitudinal program is to read the data 1A1B and plot the contours (1A - 1B/1A + 1B) for filtered longitudinal plot and intensity plot by choosing the positive and negative contour levels, intensity levels, and also X,Y plotting ranges which need to be used.

1.2 The Transverse program is to read the data files 2A2B and 3A3B and generate data

$$\sqrt{(2A - 2B/2A + 2B)^2 + (3A - 3B/3A + 3B)^2}$$

to plot the transverse contour plot, azimuth plot

$$\text{azimuth angle } \phi = \tfrac{1}{2} * \tan^{-1}\left[\frac{-(3A - 3B)/(3A + 3B)}{(2A - 2B)/(2A + 2B)}\right],$$

and intensity plot by choosing the positive and negative contour levels, intensity levels, and also X,Y plotting ranges which need to be used.

2.0 COMPUTER AND OPERATION SYSTEM

The program is operational under Sigma 5, using the Tektronics terminal.

3.0 PROCEDURE

3.1 Press Break key and press Return key. System will ask for ID number.

3.2 Key in WANG, ES52000 (ID will not show on the screen).

3.3 Key in PLATEN 132. (This will set the value of the terminal platen width.)

3.4 Key in XEQ KZTIME to execute KZTIME program which will list the housekeeping information from the header record of each file in the data tape. (An example is included in Appendix A.)

Key in PCL.

Key in REW FT#TEST (to rewind the tape).

Press Control key and Z key simultaneously.

3.5 Key in XEQ KZLONGT (to execute Longitudinal program). An example of how to answer all the questions is included in Appendix A. The choice of record/data file equals the beginning record number of 1A1B file from KZTIME program output.

3.6 Key in XEQ KZTRANSV (to execute Transverse program). An example is given in Appendix A. Record/data file equals the beginning record number of 2A2B file from KZTIME program output.

3.7 After running each program, key in PCL and then key in REW FT#TEST to rewind the tape.

Note: The following inputs should be used when a typing error occurs:

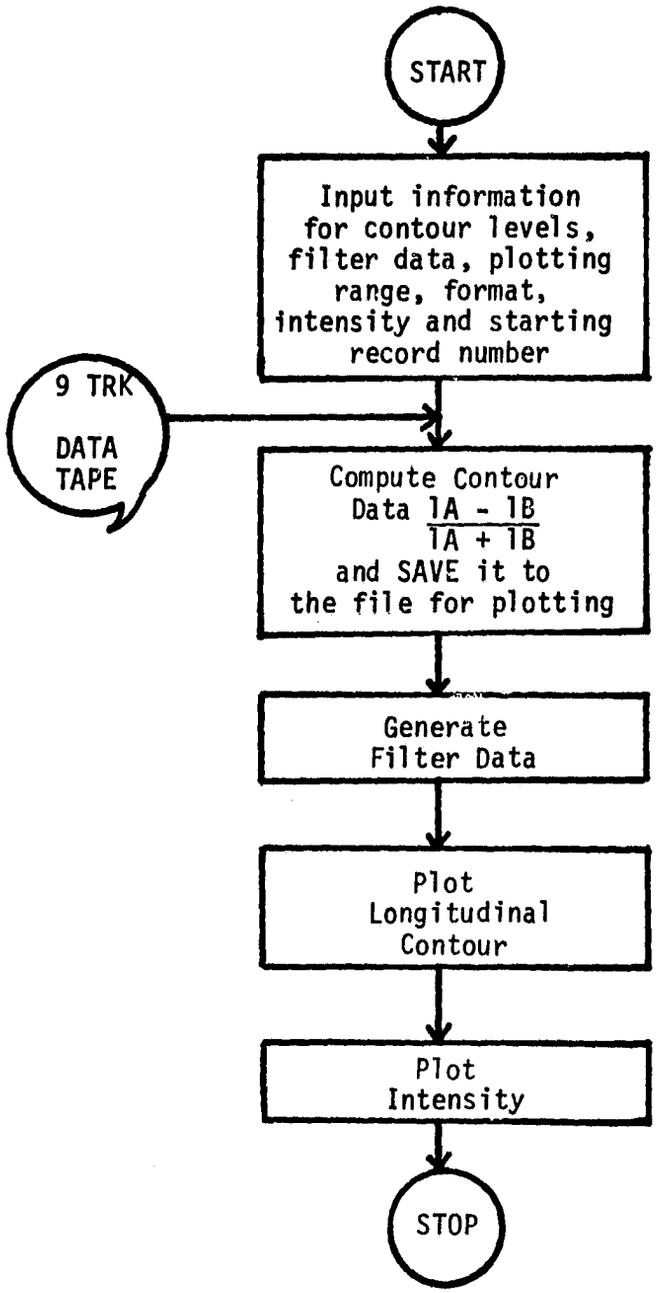
Hit "ROB" - Cancel one character typed on the line.

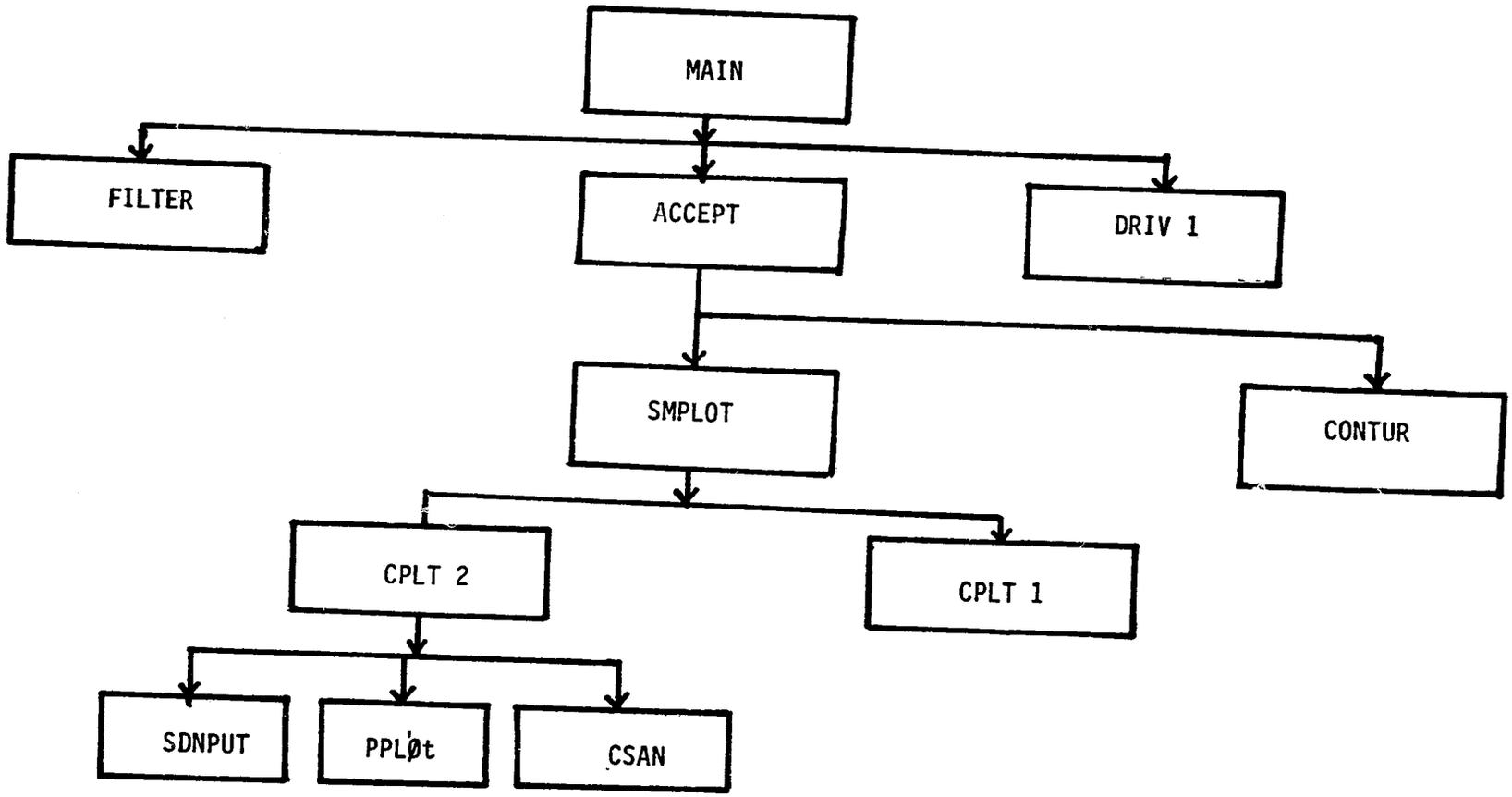
Hit "ESCAPE" and then hit "X" - Cancel all characters typed on the line.

Hit "ESCAPE" and another "ESCAPE" - Terminate the job.

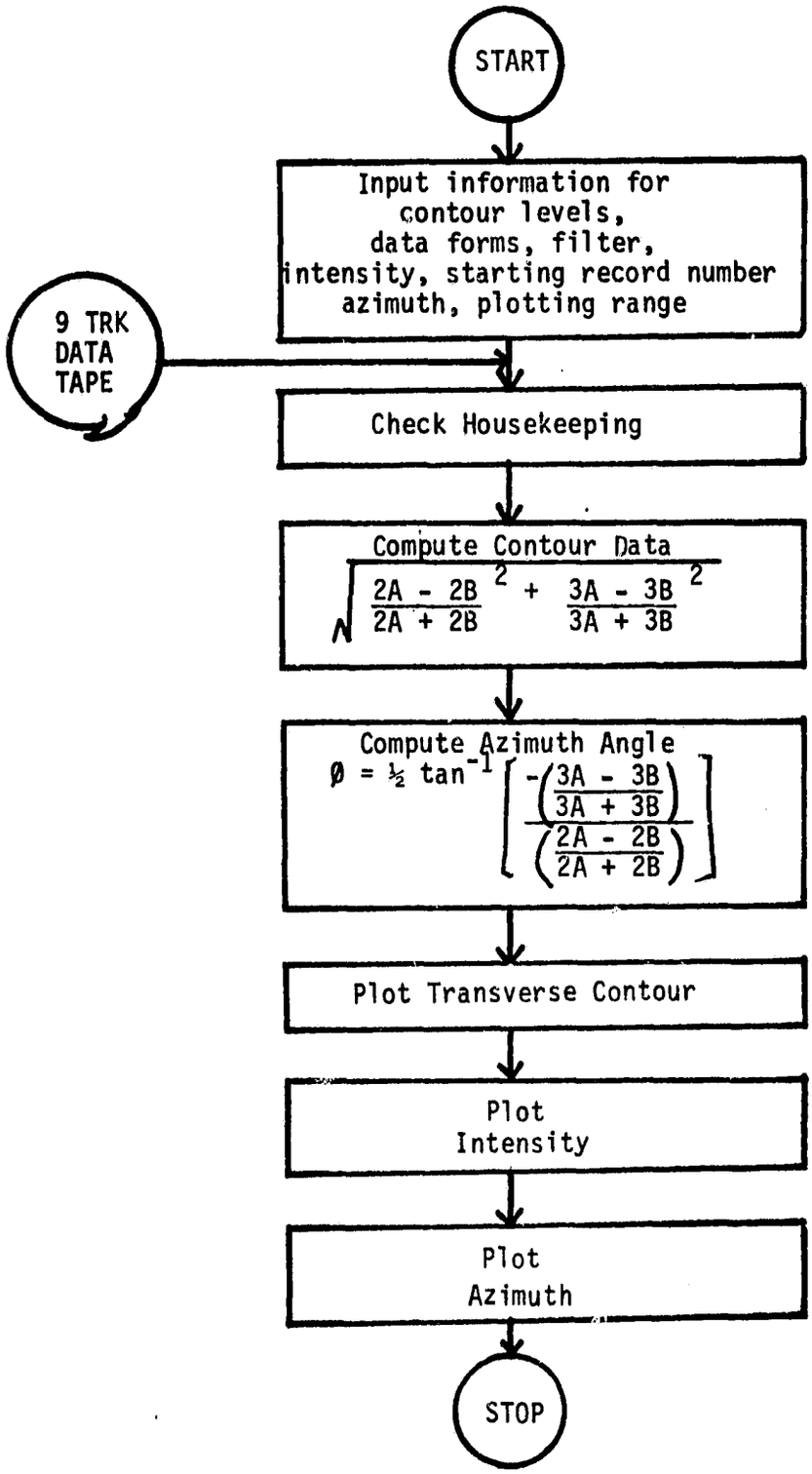
Every time a job is terminated, the tape must be rewound before running another program.

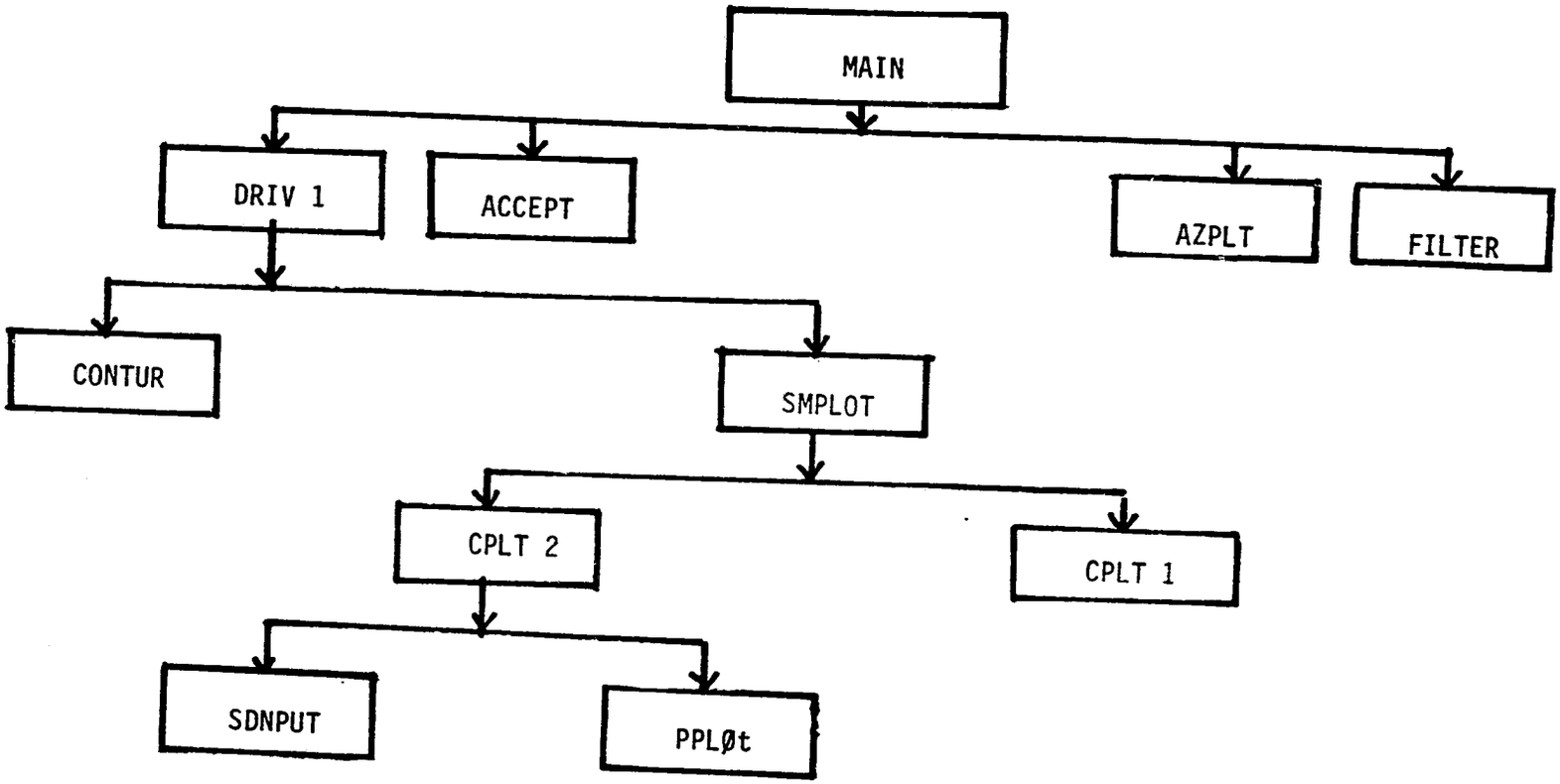
4.0 LONGITUDINAL PROGRAM FLOW DIAGRAM




MAIN
FILTER
ACCEPT
DRIV 1
SMPLOT
CONTUR
CPLT 2
CPLT 1
SDNPUT
PPLØt
CSAN

5.0 TRANSVERSE PROGRAM FLOW DIAGRAM




MAIN
DRIV 1
ACCEPT
AZPLT
FILTER
CONTUR
SMPLOT
CPLT 2
CPLT 1
SDNPUT
PPLØt

APPENDIX A
SAMPLE CASES

Sample Case for KZTIME

```
XEQ X:KZTIME
    0001 - !ECHO
    0002 - !ERROR
    0003 - !SET F:13 FT8TEST
    0004 - !SET F:10 ME
    0005 - !L:KZTIME.
 DO YOU WANT TO SKIP TO AN END OF FILE MARK (1=YE,0=NO
?1
 INPUT THE NUMBER OF EOF TO SKIP?
?1

 EOF 1      RECORD COUNT      1
 STARTING RECORD NUMBER?
?1

    1 REC:     1   TIME:127:13:28:48       ZF:0870
                   ENH.:255    POL.:1    EXP.: 1/15
                   XPOS:    .19       YPOS:  8.30

    2 REC:    33   TIME:127:13:31:56       ZF:0870
                   ENH.:255    POL.:1    EXP.: 1/15
                   XPOS:    .21       YPOS:  8.30

    3 REC:    65   TIME:127:13:31:56       ZF:0870
                   ENH.:255    POL.:1    EXP.: 1/15
                   XPOS:    .13       YPOS:  8.28

    4 REC:    97   TIME:127:13:31:56       ZF:0870
                   ENH.:255    POL.:1    EXP.: 1/15
                   XPOS:    .13       YPOS:  8.30

EOF  REC.:  128


EOF  REC.:  130

$STOP: 0
$$$XEQ TERMINATED$$$
```

Sample Case for Longitudinal Program

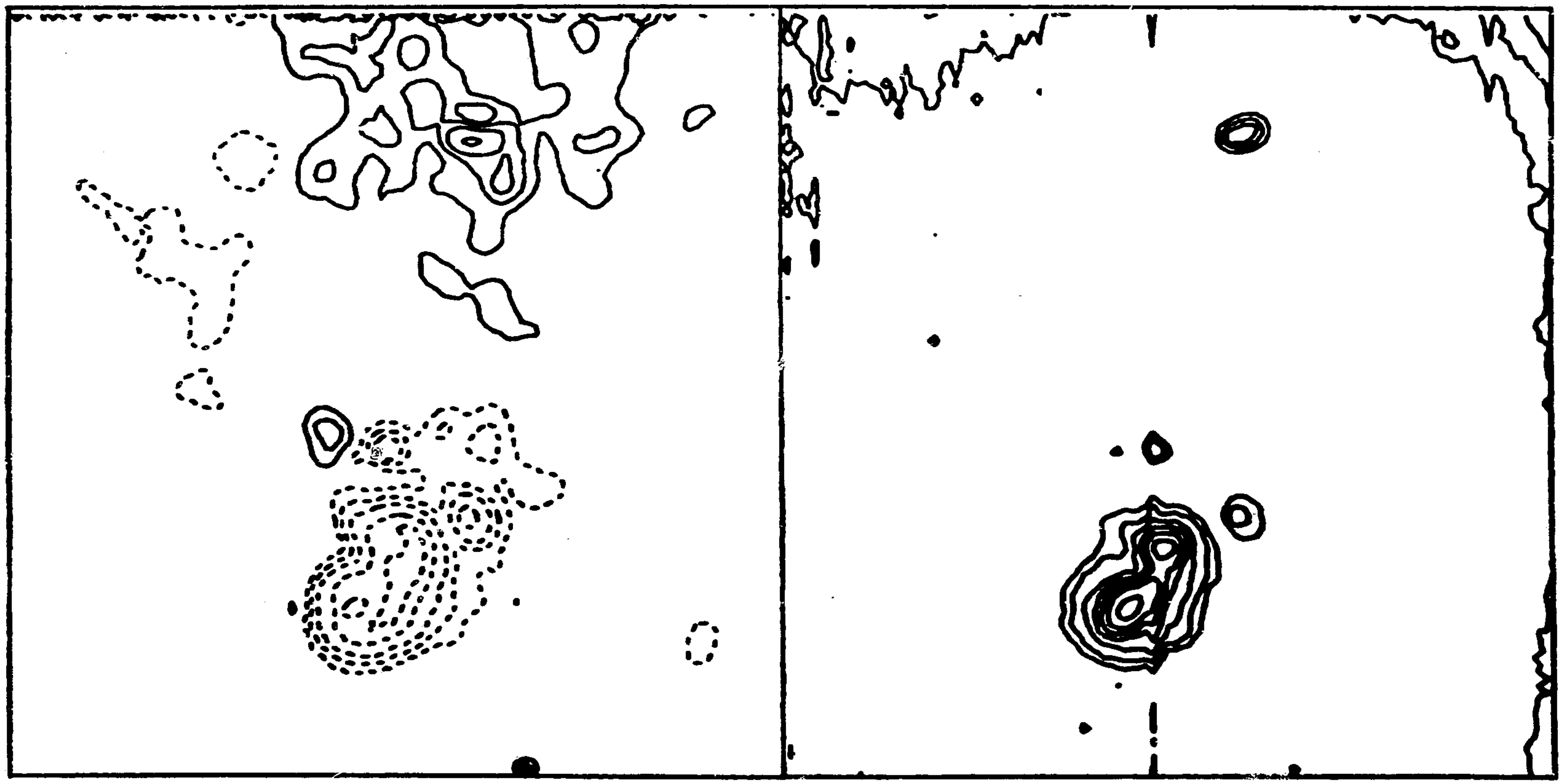
```
XEQ X:KZLONGT
    0001 - !ECHO
    0002 - !ERROR
    0003 - !SET F:13 FTSTEST
    0004 - !SET F:2 /1A;OUTIN;SAVE;KEYED;DIRECT
    0005 - !SET F:3 /1B;OUTIN;SAVE;KEYED;DIRECT
    0006 - !SET F:5 /FHL;OUTIN;SAVE;KEYED;DIRECT
    0007 - !SET F:6 /HL;OUTIN;SAVE;KEYED;DIRECT
    0008 - !SET F:7 /HI;OUTIN;SAVE;KEYED;DIRECT
    0009 - !SET F:12 /A;OUTIN;SAVE;KEYED;DIRECT
    0010 - !SET F:9 /B;OUTIN;SAVE;KEYED;DIRECT
    0011 - !SET F:11 /C;OUTIN;SAVE;KEYED;DIRECT
    0012 - !SET F:14 LF
    0013 - !SET F:20 /DMDS;OUT;SAVE
    0014 - !SET F:22 /DMDS
    0015 - !SET F:10 ME
    0016 - !SET F:101 ME
    0017 - !L:KZLONGT.
 NUMBER OF POSITIVE CONTOUR LEVELS
??
 A=
?10
 B=
?25
 C=
?50
 D=
?100
 E=
?150
 F=
?200
 G=
?250

 DO YOU WANT THE POSITIVE AND NEGATIVE CONTUR VALUES
 TO BE THE SAME (1=YES,0=NO)?
?1

 FILTER DATA (1=YES,0=NO)?
?1
 THERE ARE TWO DATA FORMATS.
 1.  4.25X4.25 INCHES WHERE BOTH THE MAGNETOGRAM
     AND INTENSITY CAN BE PLACED ON SAME PAGE

 2.  6.5X8.5 INCHES ONLY ONE PLOT PER PAGE
 FORMAT (1 OR 2) ?
?1
 DELETE THE CHARACTER LABELS? (1=YES,0=NO)
?0
 DELETE THE FIRST NEGATIVE AND POSITIVE CONTURS LABLES(1=YES,0=NO   )?
?1
 XMIN=
?10
 XMAX=
?120
 YMIN=
?10
 YMAX=
?120
 DO YOU WANT INTENSITY PLOT?  (1=YES,0=NO)
?1
 NUMBER OF INTENSITY VALUES  (MAX. OF 7 )?
??
 A=
?4000
 B=
?3500
 C=
?3000
 D=
?2500
 E=
?2000
 F=
?1500
 G=
?1000
 WAS PROCESSING INTERRUPTED?  (1=YES,0=NO
?0
 ARE THERE EOF MARKS BETWEEN 1A1B/2A2B/3A3B
 (1=YES;0=NO)?
?0
 NUMBER OF MAGNETOGRAMS TO PROCESS
?1
 INPUT RECORD/DATA FILE NUMBERS OF MAGNETOGRAMS

 RECORD/DATA FILE =
?2


 DATA TO BE PLOTTED:  127:13:28:48
 NZFT=0870 NEMH=  255 NEXP=4 NPOL=1
 M1A(1)=D4481274 IDAT=1274D448
```

FILTERED LONGITUDINAL PLOT
TIME 127:13:28:46
ZEISS FILTER POSITION:0070   ENHANCEMENTS: 255
EXPOSURE: 1/15   POLARIZATIONS: 1
XPOS: .19   YPOS: 2.30
APERTURE:X(10,120)   Y(10,120)

LABEL	VALUE	LABEL	VALUE
A	10	H	-10
B	25	I	-25
C	50	J	-50
D	100	K	-100
E	150	L	-150
F	200	M	-200
G	250	N	-250

INTENSITY PLOT
(1A1B)

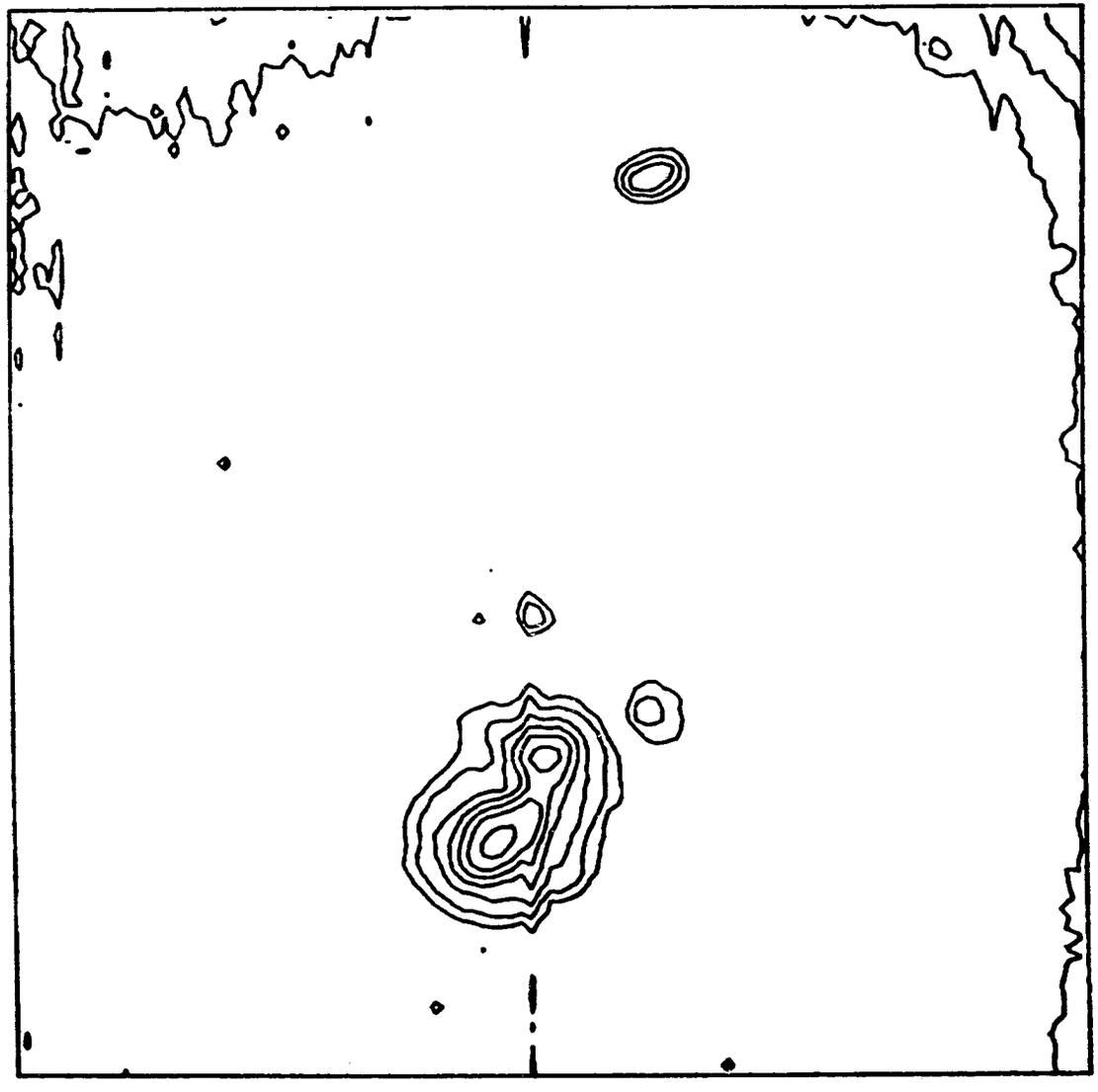
LABEL	VALUE
A	4000
B	3500
C	3000
D	2500
E	2000
F	1500
G	1000



```
XEQ X:KZLONGT
    0001 - !ECHO
    0002 - !ERROR
    0003 - !SET F:13 FT8TEST
    0004 - !SET F:2 /1A;OUTIN;SAVE;KEYED;DIRECT
    0005 - !SET F:3 /1B;OUTIN;SAVE;KEYED;DIRECT
    0006 - !SET F:5 /FHL;OUTIN;SAVE;KEYED;DIRECT
    0007 - !SET F:6 /HL;OUTIN;SAVE;KEYED;DIRECT
    0008 - !SET F:7 /HI;OUTIN;SAVE;KEYED;DIRECT
    0009 - !SET F:12 /A;OUTIN;SAVE;KEYED;DIRECT
    0010 - !SET F:9 /B;OUTIN;SAVE;KEYED;DIRECT
    0011 - !SET F:11 /C;OUTIN;SAVE;KEYED;DIRECT
    0012 - !SET F:14 LP
    0013 - !SET F:20 /BNDS;OUT;SAVE
    0014 - !SET F:22 /BNDS
    0015 - !SET F:10 RE
    0016 - !SET F:101 RE
    0017 - !L:KZLONGT.
 NUMBER OF POSITIVE CONTOUR LEVELS
??
 A=
?10
 B=
?25
 C=
?50
 D=
?100
 E=
?150
 F=
?200
 G=
?250

 DO YOU WANT THE POSITIVE AND NEGATIVE CONTUR VALUES
 TO BE THE SAME (1=YES,0=NO)?
?1

 FILTER DATA (1=YES,0=NO)?
?1
 THERE ARE TWO DATA FORMATS.
 1.  4.25X4.25 INCHES WHERE BOTH THE MAGNETOGRAM
     AND INTENSITY CAN BE PLACED ON SAME PAGE

 2.  6.5X8.5 INCHES ONLY ONE PLOT PER PAGE
 FORMAT (1 OR 2) ?
?2
 DELETE THE CHARACTER LABELS? (1=YES,0=NO)
?0
 DELETE THE FIRST NEGATIVE AND POSITIVE CONTURS LABLES(1=YES,0=NO   )?
?1
 XMIN=
?10
 XMAX=
?120
 YMIN=
?10
 YMAX=
?120
 DO YOU WANT INTENSITY PLOT?  (1=YES,0=NO)
?1
 NUMBER OF INTENSITY VALUES  (MAX. OF 7 )?
??
 A=
?4000
 B=
?3500
 C=
?3000
 D=
?2500
 E=
?2000
 F=
?1500
 G=
?1000
 WAS PROCESSING INTERRUPTED?  (1=YES,0=NO
?0
 ARE THERE EOF MARKS BETWEEN 1A1B/2A2B/3A3B
 (1=YES;0=NO)?
?0
 NUMBER OF MAGNETOGRAMS TO PROCESS
?1
 INPUT RECORD/DATA FILE NUMBERS OF MAGNETOGRAMS

 RECORD/DATA FILE =
?2


 DATA TO BE PLOTTED:  127:13:28:48
 NZFT=0870 MEMM=  255 MEXP=4 MPOL=1
 M1A(1)=D44?1274 IDAT=1274D448
```

INTENSITY PLOT (2AB3)
TIME 127:13:28:48
ZEISS FILTER POSITION:0870 ENHANCEMENTS: 255
EXPOSURE: 1/15 POLARIZATIONS: 1
XPOS: .19 YPOS: 2.30
APERTURE:X(10,120) Y(10,120)

LABEL	VALUE
A	4000
B	3500
C	3000
D	2500
E	2000
F	1500
G	1000

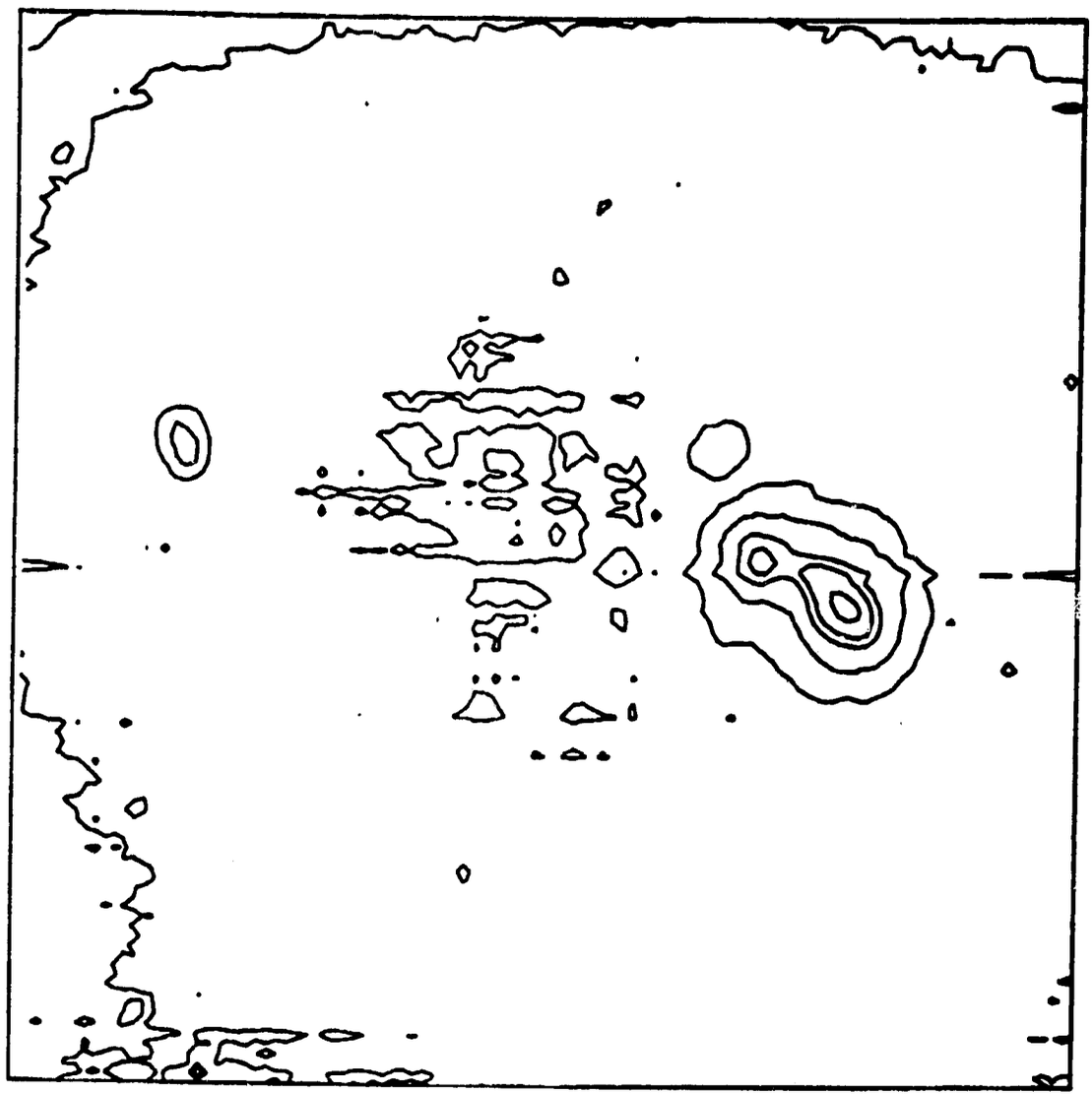


Sample Case for Transverse Program

```
XEQ X:KZTRANSV
    0001 - !ECHO
    0002 - !ERROR
    0003 - !SET F:13 FT0TEST
    0004 - !SET F:2 /2A;OUTIN;SAVE;KEYED;DIRECT
    0005 - !SET F:3 /2B;OUTIN;SAVE;KEYED;DIRECT
    0006 - !SET F:4 /3A;OUTIN;SAVE;KEYED;DIRECT
    0007 - !SET F:5 /3B;OUTIN;SAVE;KEYED;DIRECT
    0008 - !SET F:6 /HT;OUTIN;SAVE;KEYED;DIRECT
    0009 - !SET F:7 /HIU;OUTIN;SAVE;KEYED;DIRECT
    0010 - !SET F:8 /HIR;OUTIN;SAVE;KEYED;DIRECT
    0011 - !SET F:9 /B;OUTIN;SAVE;KEYED;DIRECT
    0012 - !SET F:11/C;OUTIN;SAVE;KEYED;DIRECT
    0013 - !SET F:12/A;OUTIN;SAVE;KEYED;DIRECT
    0014 - !SET F:20 /BNDS;OUT;SAVE
    0015 - !SET F:22 /BNDS
    0016 - !SET F:10 ME
    0017 - !SET F:101 ME
    0018 - !L:KZTRANSV.

 TRANSVERSE PROGRAM

 WAS DATA TO BE PLOTTED CREATED BEFORE APRIL 18,1980
 (1=YES, 0=NO) ?
?0

 IF AZIMUTH PLOTS ONLY;SET NO. OF CONTOUR LEVELS
 EQUAL TO ZERO
 NUMBER OF CONTOUR LEVELS (MAX. NO. 7 ) ?
?7
 A=
?50
 B=
?75
 C=
?100
 D=
?125
 E=
?150
 F=
?200
 G=
?250

 THERE ARE TWO DATA FORMATS.
 1. 4.25*4.25 INCHES WHERE BOTH THE MAGNETOGRAM
    AND INTENSITY CAN BE PLACED ON THE SAME PAGE

 2. 8.5*8.5 INCHES ONLY ONE PLOT PER PAGE
 FORMAT (1 OR 2 ) ?
?2

 DELETE ALL OF THE CHARACTER LABELS? (1=YES,0=NO)
?0

 DELETE THE FIRST CONTOUR LEVEL? (1=YES,0=NO)
?1

 ENTER ARRAY SIZE (MAXIMUM 128X128)
 XMIN=
?10
 XMAX=
?120
 YMIN=
?10
 YMAX=
?120

 DO YOU WANT TO INPUT THE MIN. AND MAX. VALUES
 FOR THE AZIMUTH PLOTS (1=YES,0=NO)?
?0

 DO YOU WANT INTENSITY PLOTS (1=YES,0=NO)?
?1
 2A,2B (1=YES,0=NO)?
?1
 3A,3B (1=YES,0=NO)?
?1

 NUMBER OF INTENSITY VALUES (MAX. NO. 7)?
?7
 INPUT INTENSITY VALUES DIVIDED BY 10
 A=
?5000
 B=
?4000
 C=
?3000
 D=
?2000
 E=
?1500
 F=
?1000
 G=
?500

 WAS ROCESSING INTERRUPTED (1=YES,0=NO)?
?0

 NUMBER OF MAGNETOGRAMS TO PROCESS?
?1
 ARE THERE EOF MARKS BETWEEN 1A1B/2A2B/3A3B
 (1=YES, 0=NO) ?
?0

 INPUT RECORD/DATA NUMBERS OF MAGNETOGRAMS
 RECORD/DATA FILE =
?34

 DATA TO BE PLOTTED:  127:13:31:56
 MZFT=0070 MENM=  255 MEXP=4 MPOL=1
 M1A(1)=00061274 IDAT=12740006
```

MARSHALL SPACE FLIGHT CENTER MAGNETOGRAMS
TELEPHONE:205-453-5687 FTS:878-5687

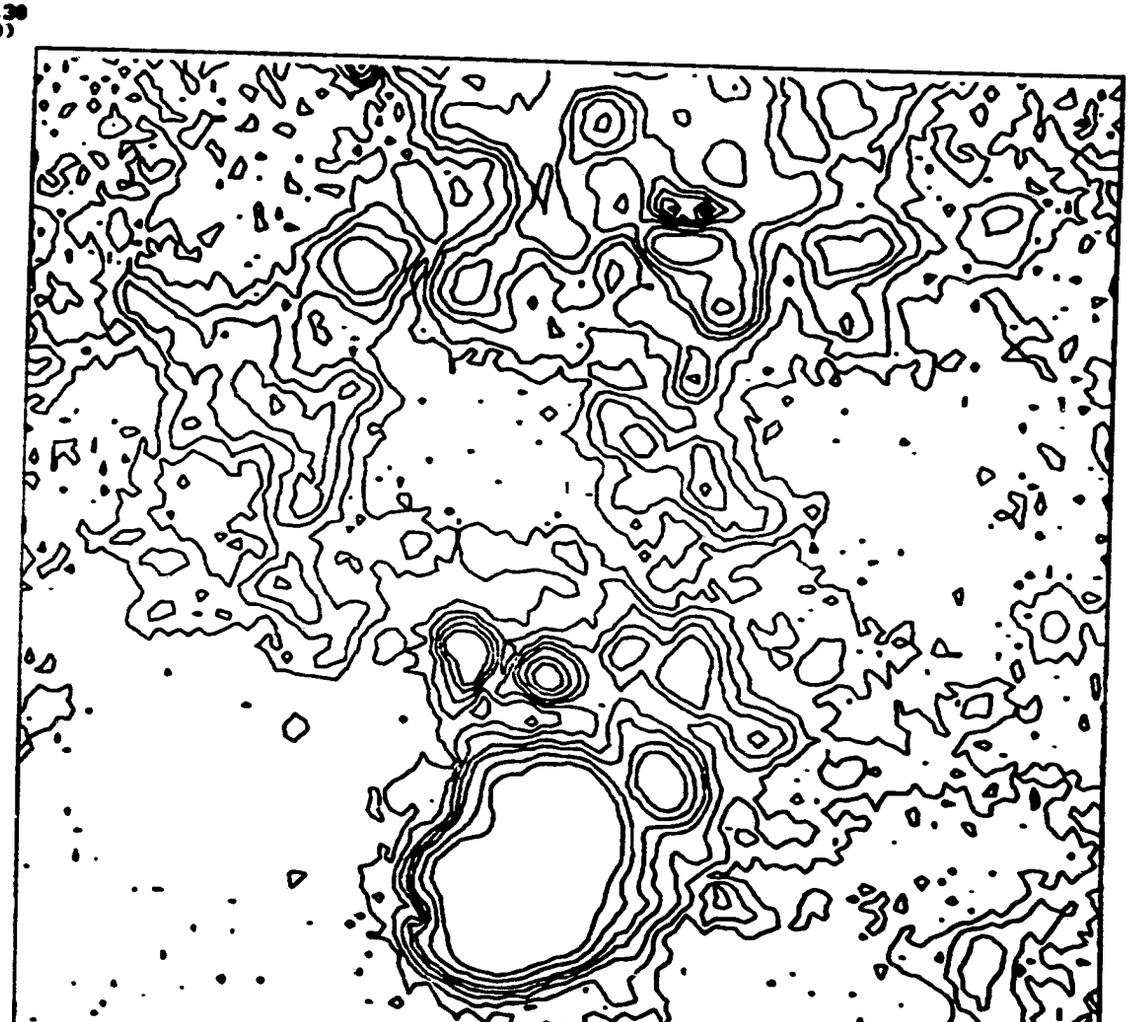
AZIMUTH PLOT
TIME127:13:31:56
ZEISS FILTER:6870 ENHANCEMENTS: 255
EXPOSURE: 1/15 POLARIZATIONS: 1
XPOS: .21 YPOS: 2.30
APERTURE: X(10,120) Y(10,120)
MINIMUM LENGTH SET AT 60.
MAXIMUM LENGTH SET AT 200.

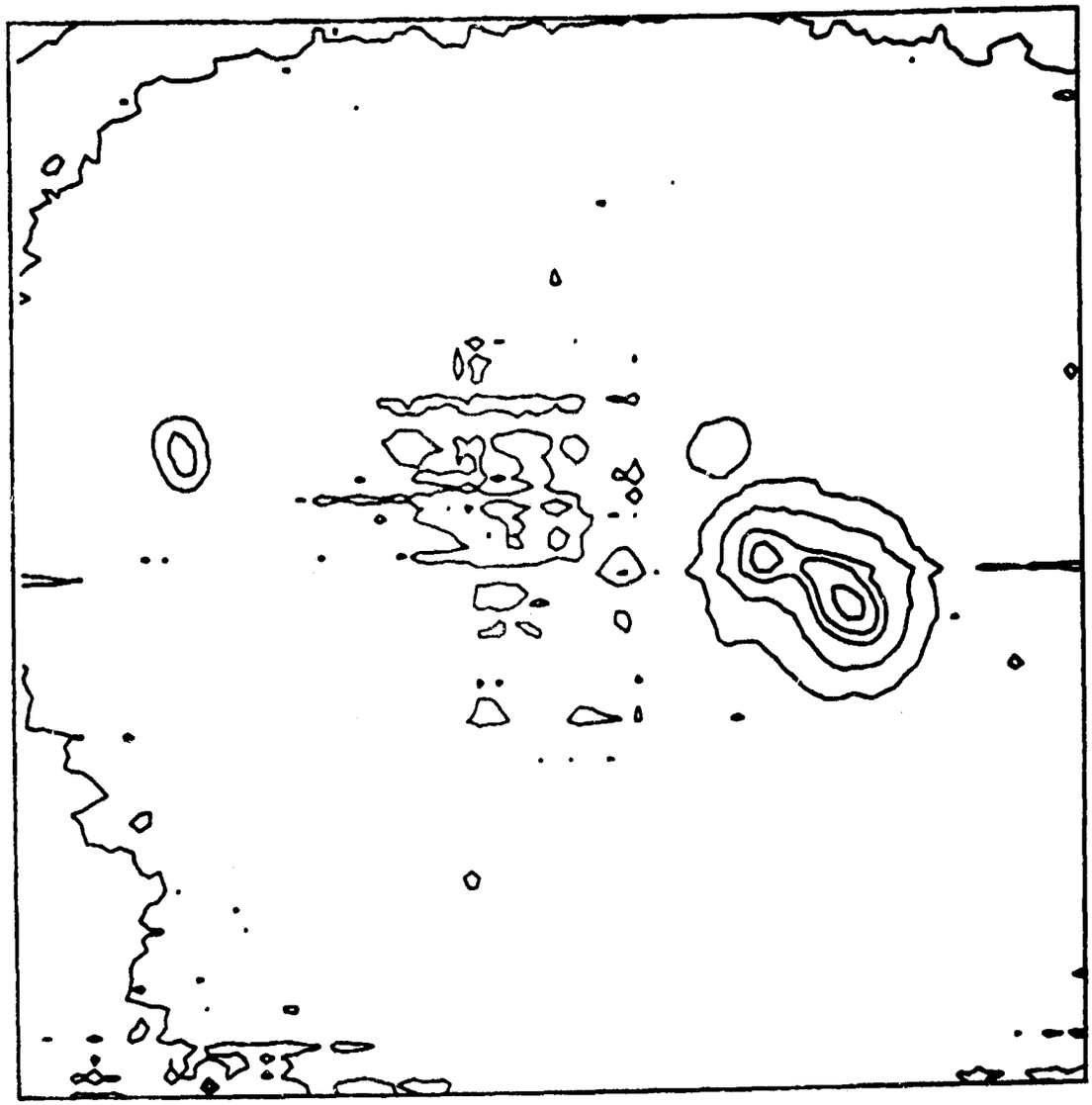


INTENSITY PLOT (8423)
TIME 127:13:31:58
ZEISS FILTER POSITION:8570 ENHANCEMENTS: 255
EXPOSURE: 1/15 POLARIZATIONS: 1
XPOS: .21 YPOS: 2.30
APERTURE:X(10,120) Y(10,120)

LABEL	VALUE
A	5000
B	4500
C	4000
D	3500
E	3000
F	2500
G	2000
H	1500
I	1000
J	500

TRANSVERSE PLOT
TIME 127:13:31:56
ZEISS FILTER POSITION:8870 ENHANCEMENTS: 255
EXPOSRE: 1/15 POLARIZATIONS: 1
XPOS: .21 YPOS: 2.30
APERTURE:X(10,120) Y(10,120)





INTENSITY PLOT (3438)
TIME 127:13:31:56
ZEISS FILTER POSITION:8870 ENHANCEMENTS: 255
EXPOSURE: 1/15 POLARIZATIONS: 1
XPOS: .21 YPOS: 2.30
APERTURE:X(10,120) Y(10,120)

LABEL	VALUE
A	5000
B	4000
C	3000
D	2000
E	1500
F	1000
G	500

APPENDIX B
PROGRAM LISTING

PROGRAM LISTING INDEX

EXT. FORTRAN IV, VERSION F02

Program KZTIME

```
      DIMENSION M00(15),M01(10),M02(8)
      INTEGER DT
      COMMON /HEAD/ INPBUF(512)
      COMMON /IDAT/ DT(4),NZFT,NENH,NEXP,NPOL,XPOS,YPOS
      DATA M00 /60H(1X,"DO YOU WANT TO SKIP TO AN END OF FILE MARK (1=YES
     *,0=NO")/
      DATA M01 /40H(1X,"INPUT THE NUMBER OF EOF TO SKIP?")  /
      DATA M02 /32H(1X,"STARTING RECORD NUMBER?")   /
C
C
C
 1000 FORMAT(/,1X,"EOF",I2,5X,"RECORD COUNT",I6)
 2000 FORMAT(1X,I4," REC:",I5,2X,"TIME:",Z3,3(":",Z2),6X,"ZF:",Z4)
 3000 FORMAT(17X,"ENH.:",I3,3X,"POL.:",I1,3X,"EXP.: 1/",I2)
 4000 FORMAT(17X,"XPOS:",F6.2,6X,"YPOS:",F6.2)
 5000 FORMAT(1X,"EOF",2X,"REC.:",I5)
 9991 FORMAT(1X,"INPUT TAPE ERROR , STATUS CODE =",I2)
C
C     INPUT INFORMATION ABOUT END OF FILE MARK, STARTING RECORD NO.
C     FROM TERMINAL
C
      IREC=0
      CALL ACCEPT (M00,IEOF)
      IF(IEOF .EQ. 0) GO TO 40
      CALL ACCEPT (M01,NEOF)
      DO 30 I=1,NEOF
      IRCD =1
   10 CALL BUFFERIN (13,1,INPBUF,512,ISTAT)
      IF (ISTAT .EQ. 3) GO TO 20
      IRCD = IRCD +1
      GO TO 10
   20 WRITE(10,1000) I,IRCD
   30 CONTINUE
   40 CONTINUE
      CALL ACCEPT (M02,NSKIP)
      NSKIP = NSKIP -1
      NP = 0
      J =NSKIP
      IF (NSKIP .EQ. 0) GO TO 60
      DO 50 I=1,NSKIP
      CALL BUFFERIN (13,1,INPBUF,512,ISTAT)
   50 CONTINUE
   60 CONTINUE
      IREC = IREC+1
      CALL BUFFERIN (13,1,INPBUF,512,ISTAT)
      J=J + 1
      IF( ISTAT .EQ. 3) GO TO 90
      IF(ISTAT .EQ. 2) GO TO 70
      WRITE(10,9991) ISTAT
   70 IF( IREC .EQ. 1) GO TO 80
      IF(IREC .EQ. 32) IREC = 0
      GO TO 60
```

```
80    CONTINUE
      CALL PTIME
      NP =NP + 1
      OUTPUT ' '
      WRITE(10,2000) NP,J,DT(4),DT(3),DT(2),DT(1),NZFT
      WRITE(10,3000) NENH,NPOL,NEXP
      WRITE(10,4000) XPOS,YPOS
      GO TO 60
90    CONTINUE
      OUTPUT ' '
      WRITE(10,5000) J
      OUTPUT ' '
      EOF1 = J
      IF(EOF1 .EQ. EOF2) STOP
      EOF2 =EOF1 + 1
      IREC =0
      GO TO 60
      END

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

                     DEC      HEX
                    WORDS    WORDS
                    -----    -----
  GENERATED CODE:    226     000E2
       CONSTANTS:      4     00004
 LOCAL VARIABLES:     44     0002C
           TEMPS:      0     00000
                   -----    -----
  TOTAL PROGRAM:     274     0011[illegible]   (PLUS LABELED COMMON)
```

ORIGINAL PAGE IS
OF POOR QUALITY

EXT. FORTRAN IV, VERSION F02

LONGT

```
      C
      C     SUBROUTINE LONGT
      C     THIS ROUTINE SUBTRACTS THE 1A AND 1B MATRICES
      C     AND STORES THE RESULTS IN FILE "HL",
      C     WHICH IS USED BY SUBROUTINE DRIV1 TO PROCEDURE CONTURS
      C
            COMMON /BLNK/ NBT(4),MF(10),ID(4)
            COMMON /IVAL/ NCTR, KTRVAL(7),NN1,NCTVAL(7),INTVAL(7)
            COMMON /BND/  XMIN,XMAX,YMIN,YMAX,IBIT
            COMMON /NCTVAL/NCTVAL(7),INUM
            COMMON /HL/MHL,HI1
            COMMON /ISA/ISA,INHOUS(8)
            COMMON /IDAT/DT(4),NZFT,NENH,NEXP,NPOL,XPOS,YPOS
            COMMON /ISB/ISB,NS,IDLCH,LLCH,LEXTRA,NFORM
            COMMON /IBIT/IBIT1,IBIT2,IBIT3,IBIT4,IBIT5,IBIT6,IBIT7
            COMMON /ABC/A(500),B(500),C(500),Z(130)
            INTEGER DT ,INPBUF(512),M1A(64),M1B(64)
            REAL    MHL(128),HI1(128)
            DATA  NBT/7,7,6,10/
            DATA  MF /1,2,4,8,10,20,40,80,100,200/
            DATA  ID /2Z7F,2Z7F,2Z3F,3Z3FF/
            DATA KK/4ZFFFF/
            DATA  NNN /1/
            DATA KZF/4Z1FFF/
            DATA KEH/2ZFF/
            DATA KM1A/8ZFFFF0000/
            DATA KEXP/1Z7/
            DIMENSION     IRN(50)
            DIMENSION MM35(10),MM36(5),MM37(12),MM38(9)
            DIMENSION MM00(10),MM01(3),MM02(3),MM03(3),MM04(3),
           *          MM05 (3),MM06(3),MM07(3),MM08(10),MM09(3),MM10(3),
           *          MM11 (3),MM12(3),MM13(3),MM14(3),MM15(3),MM16(3),
           *          MM17 (3),MM18(3),MM19(3) ,MM20(3),MM21(3),MM22(8),
           *          MM23(12),MM24(3),MM25(3),MM26(3),MM27(3),MM28(12),
           *          MM29(12),MM30(12),MM31(10),MM32(6),MM33(19),MM34(11)
            DATA MM00/40H(1X,"NUMBER OF POSITIVE CONTOUR LEVELS")/
            DATA MM01/12H(1X,"A=    ")/
            DATA MM02/12H(1X,"B=    ")/
            DATA MM03/12H(1X,"C=    ")/
            DATA MM04/12H(1X,"D=    ")/
            DATA MM05/12H(1X,"E=    ")/
            DATA MM06/12H(1X,"F=    ")/
            DATA MM07/12H(1X,"G=    ")/
            DATA MM08/40H(1X,"NUMBER OF NEGATIVE CONTOUR LEVELS")/
            DATA MM09/12H(1X,"B=-   ")/
            DATA MM10/12H(1X,"C=-   ")/
            DATA MM11/12H(1X,"D=-   ")/
            DATA MM12/12H(1X,"E=-   ")/
            DATA MM13/12H(1X,"F=-   ")/
            DATA MM14/12H(1X,"G=-   ")/
            DATA MM15/12H(1X,"H=-   ")/
            DATA MM16/12H(1X,"I=-   ")/
```

```
      DATA MM17/12H(1X,"J=-  ")/
      DATA MM18/12H(1X,"K=-  ")/
      DATA MM19/12H(1X,"L=-  ")/
      DATA MM20/12H(1X,"M=-  ")/
      DATA MM21/12H(1X,"N=-  ")/
      DATA MM22/32H(1X,"FILTER DATA (1=YES,0=NO)?")/
      DATA MM23/48H(1X,"DELETE THE CHARACTER LABELS? (1=YES,0=NO)")/
      DATA MM24/12H(1X,"XMIN=")/
      DATA MM25/12H(1X,"XMAX=")/
      DATA MM26/12H(1X,"YMIN=")/
      DATA MM27/12H(1X,"YMAX=")/
      DATA MM28/48H(1X,"DO YOU WANT INTENSITY PLOT?  (1=YES,0=NO)")/
      DATA MM29/48H(1X,"NUMBER OF INTENSITY VALUES  (MAX. OF 7 )?")/
      DATA MM30/48H(1X,"WAS PROCESSING INTERRUPTED?  (1=YES,0=NO" )/
      DATA MM31/40H(1X,"NUMBER OF MAGNETOGRAMS TO PROCESS")/
      DATA MM32/24H(1X,"FORMAT (1 OR 2) ?")/
      DATA MM33/76H(1X,"DELETE THE FIRST NEGATIVE AND POSITIVE CONTURS L
     *ABLES(1=YES,0=NO   )?")/
      DATA MM34/44H(1X,"REPEAT LAST CONTOURS? (1=YES , 0=NO) ")/
      DATA MM35/40H(1X,"INPUT FROM TAPE(1)  OR DISK (0) ?")/
      DATA MM36/20H(1X,"(1=YES;0=NO)?")/
      DATA MM37/28H(1X,"RECORD/DATA FILE =    )/
      DATA MM38/36H(1X,"TO BE THE SAME (1=YES,0=NO)?" )/
      DO 10  I=1,7
      NCTVAL(I) =0
      NCIVAL(I) =0
      INTVAL(I) =0
   10 CONTINUE
      DO 20 I=1,50
   20 IRN(I)=0
      DO 33 I=1,500
      A(I)=0
      B(I)=0
      C(I)=0
   33 CONTINUE
      DO 44 I=1,130
   44 Z(I)=0
      DO 66 I=1,128
      MHL(I)=0
      HII(I)=0
   66 CONTINUE
      NF1  =0
      NKEOF=0
      NEG  =0
      IBIT3=0
      IBIT4=0
      IBIT5=0
      IBIT6=0
      IBIT7=0
    1 CALL ACCEPT(MM00,NF1)
      NCTR =NF1
      NF=NF1
```

```
      DO 4  I=1,NF
      IF( I .EQ. 1) CALL ACCEPT(MM01,NCTVAL(I))
      IF( I .EQ. 2) CALL ACCEPT(MM02,NCTVAL(I))
      IF( I .EQ. 3) CALL ACCEPT(MM03,NCTVAL(I))
      IF (I .EQ. 4) CALL ACCEPT(MM04,NCTVAL(I))
      IF( I .EQ. 5) CALL ACCEPT(MM05,NCTVAL(I))
      IF( I .EQ. 6) CALL ACCEPT(MM06,NCTVAL(I))
      IF( I .EQ. 7) CALL ACCEPT(MM07,NCTVAL(I))
      IF( I .GT. 7) OUTPUT 'ERROR = MAXIMUM OF 7 POSITIVE CONTOURS'
      IF( I .GT. 7) GO TO 1
    4 CONTINUE
      OUTPUT ' '
      OUTPUT 'DO YOU WANT THE POSITIVE AND NEGATIVE CONTUR VALUES'
      CALL ACCEPT (MM38,ISM)
      IF(ISM .EQ. 1) GO TO 6
      OUTPUT ' '
    2 CALL ACCEPT (MM08,NN1)
      DO 3  I=1,NN1
      IF(NC .EQ. 1) OUTPUT 'NO POSITIVE VALUES?'
      IF(NC .EQ. 2) CALL ACCEPT(MM09,NCTVAL(I))
      IF(NC .EQ. 3) CALL ACCEPT(MM10,NCTVAL(I))
      IF(NC .EQ. 4) CALL ACCEPT(MM11,NCTVAL(I))
      IF(NC .EQ. 5) CALL ACCEPT(MM12,NCTVAL(I))
      IF(NC .EQ. 6) CALL ACCEPT(MM13,NCTVAL(I))
      IF(NC .EQ. 7) CALL ACCEPT(MM14,NCTVAL(I))
      IF(NC .EQ. 8) CALL ACCEPT(MM15,NCTVAL(I))
      IF(NC .EQ. 9) CALL ACCEPT(MM16,NCTVAL(I))
      IF(NC .EQ. 10) CALL ACCEPT(MM17,NCTVAL(I))
      IF(NC .EQ.11) CALL ACCEPT(MM18,NCTVAL(I))
      IF(NC .EQ.12) CALL ACCEPT(MM19,NCTVAL(I))
      IF(NC .EQ.13) CALL ACCEPT(MM20,NCTVAL(I))
      IF(NC .EQ.14) CALL ACCEPT(MM21,NCTVAL(I))
      IF(NC .GT. 14) STOP
      NCTVAL(I) =-NCTVAL(I)
      IF(I.GT. 7) GO TO 2
      NC=NC+1
    3 CONTINUE
      GO TO 9
    6 CONTINUE
      NN1=NF1
      NEG=NF1
      DO 7 I=1,NEG
    7 NCTVAL(I)=-NCTVAL(I)
    9 CONTINUE
      OUTPUT ' '
      CALL ACCEPT (MM22,IBIT3)
      OUTPUT 'THERE ARE TWO DATA FORMATS.'
      OUTPUT '1.  4.25X4.25 INCHES WHERE BOTH THE MAGNETOGRAM'
      OUTPUT '    AND INTENSITY CAN BE PLACED ON SAME PAGE'
      OUTPUT ' '
      OUTPUT '2.  6.5X6.5 INCHES ONLY ONE PLOT PER PAGE'
      CALL ACCEPT (MM32,NFORM)
```

```
      CALL ACCEPT (MM23,IDLCH)
      CALL ACCEPT (MM33,LLCH)
      CALL ACCEPTR (MM24,XMIN)
      CALL ACCEPTR (MM25,XMAX)
      CALL ACCEPTR (MM26,YMIN)
      CALL ACCEPTR (MM27,YMAX)
      CALL ACCEPT (MM28,IBIT5)
      IF (IBIT5 .EQ. 0) GO TO 35
      CALL ACCEPT (MM29,INUM)
      DO 34 I=1,INUM
      IF(I .EQ. 1) CALL ACCEPT (MM01,INTVAL(I))
      IF(I .EQ. 2) CALL ACCEPT (MM02,INTVAL(I))
      IF(I .EQ. 3) CALL ACCEPT (MM03,INTVAL(I))
      IF(I .EQ. 4) CALL ACCEPT (MM04,INTVAL(I))
      IF(I .EQ. 5) CALL ACCEPT (MM05,INTVAL(I))
      IF(I .EQ. 6) CALL ACCEPT (MM06,INTVAL(I))
      IF(I .EQ. 7) CALL ACCEPT (MM07,INTVAL(I))
   34 CONTINUE
   35 CONTINUE
      CALL ACCEPT (MM30,IBIT4)
      IBIT6 =0
      IF(IBIT4 .EQ. 1) CALL ACCEPT (MM34,IBIT6)
      IF(IBIT4 .EQ. 1) GO TO 16
      OUTPUT 'ARE THERE EOF MARKS BETWEEN 1A1B/2A2B/3A3B'
      CALL ACCEPT (MM36,MKEOF)
      CALL ACCEPT(MM31,NP)
      OUTPUT 'INPUT RECORD/DATA FILE NUMBERS OF MAGNETOGRAMS'
      OUTPUT ' '
      DO 1000 I=1,NP
 1000 CALL ACCEPT (MM37,IRN(I))
   16 CONTINUE
C
C     CREATE ZERO FILES
C
      JREC=0
      DO 4000 I3=1,NP
C
C     MAG. TAPE INPUT
C
  130 IF(JREC .EQ. (IRN(I3)-1)) GO TO 120
      CALL BUFFERIN(13,1,INPBUF,512,ISTAT)
      JREC =JREC + 1
      GO TO 130
  120 CONTINUE
      DO 5 K=2,3
      DO 110 I=1,16
      CALL BUFFERIN(13,1,INPBUF,512,ISTAT)
      IF (ISTAT .EQ. 2) GO TO 15
      STOP
   15 CONTINUE
      IF(I .GT. 1) GO TO 220
      IREC =0
```

```
  220 CONTINUE
      IWD = 1
      DO 45 I1 =1,8
      DO 50 J1 =1,64
      M1A(J1) = INPBUF(IWD)
      IWD =IWD + 1
   50 CONTINUE
      CALL DIRECT WRITE (K,M1A,64,IREC)
      IREC = IREC + 1
   45 CONTINUE
      JREC = JREC +1
  110 CONTINUE
    5 CONTINUE
C
C     READ END OF FILE
C
      IF(MKEOF .EQ. 0) GO TO 55
      CALL BUFFERIN(13,1,INPBUF,512,ISTAT)
      JREC =JREC +1
   55 CONTINUE
   56 CONTINUE
      IRN3 = IRN(I3)
C
C     CHECK HOUSEKEEPING
C
      CALL DIRECT READ (2,M1A,64,0)
      IDAT=ISL(M1A(1),-16)+IAND(ISL(M1A(1),16),KM1A)
      DO 252 I1=1,4
      IF (I1 .EQ. 1) NBB =0
      IF (I1 .GT. 1) NBB = NBT(I1-1) +NBB
      DT(I1)     = IAND(ID(I1),ISL(IDAT,-NBB))
  252 CONTINUE
      WRITE (10,254) DT(4),DT(3),DT(2),DT(1)
  254 FORMAT(//,1X,'DATA TO BE PLOTTED:  ',Z3,3(':',Z2))
      NZFT=IAND(ISL(M1A(2),-16),KZF)
      NENH=IAND(ISL(M1A(2),-8),KEH)
      NEXP=IAND(ISL(M1A(3),-25),KEXP)
      NPOL=IAND(ISL(M1A(3),-19),7)
      WRITE(10,264) NZFT,NENH,NEXP,NPOL
  264 FORMAT(1X,'NZFT=',Z4,1X,'NENH=',I5,1X,'NEXP=',I1,1X,'NPOL=',I1)
      WRITE(10,274) M1A(1),IDAT
  274 FORMAT(1X,'M1A(1)=',Z8,1X'IDAT=',Z8)
      DO 25 I1=1,8
      INHOUS (I1) = M1A(I1)
   25 CONTINUE
      IF(NEXP .EQ. 2) NEXP=60
      IF(NEXP .EQ. 3) NEXP=30
      IF(NEXP .EQ. 4) NEXP=15
      IF(NEXP .EQ. 5) NEXP=8
      IXX =ISL(M1A(3),-16)+IAND(ISL(M1A(3),16),KM1A)
      ISNGX=IAND(ISL(IXX,-13),1)
      XPOS=0
```

```
      TXPOS=0
      MX=14
C     DO 333 II=1,3
      II=3
  333 CONTINUE
      NN=4
  353 CONTINUE
      TXPOS=TXPOS+IAND(ISL(IXX,-MX),1)*(2**(NN-1))
      NN = NN -1
      MX=MX+1
      IF(NN .GE. 1) GO TO 353
      XPOS=XPOS+TXPOS*(10**(II-1))
      TXPOS=0
      IF(II .LE. 1) GO TO 363
      II=II-1
      GO TO 333
  363 CONTINUE
      IF(ISNGX .EQ. 0) XPOS=-XPOS
      ISNGY=IAND(ISL(IXX,-MX),1)
      YPOS=0
      TYPOS=0
      MX=MX+1
      NN=4
  454 CONTINUE
      TYPOS=TYPOS+(IAND(ISL(IXX,-MX),1))*2**(NN-1)
      NN=NN-1
      MX=MX+1
      IF(NN .GE. 1) GO TO 454
      YPOS=YPOS+TYPOS
      YPOS=YPOS*(10**2)
      TYPOS=0
      MY=23
C     DO 444 II=1,2
      II=2
  444 CONTINUE
      NN=4
  464 CONTINUE
      IYY=ISL(M1A(4),-16)+IAND(ISL(M1A(4),16),KM1A)
      TYPOS=TYPOS + (IAND(ISL(IYY,-MY),1))*2**(NN-1)
      NN=NN - 1
      MY = MY + 1
      IF (NN .GE. 1) GO TO 464
      YPOS=YPOS + TYPOS*(10**(II-1))
      TYPOS=0
C 444 CONTINUE
      IF(II .LE. 1) GO TO 474
      II=II-1
      GO TO 444
  474 CONTINUE
      IF(ISNGY .EQ. 0) YPOS=-YPOS
      XPOS=XPOS/100.
      YPOS=YPOS/100.
```

```
C
C     SUBTRACT MATRICES
C
      ICOR =2*10
      C1   =1000./ICOR
      IXMIN=XMIN
      IXMAX=XMAX
      IYMIN=YMIN
      IYMAX=YMAX
C     INVERT YMIN & YMAX TO GET S-W DIRECTION CORRECT
C
      DO 30 I=IYMIN,IYMAX
      IREC = I-1
      CALL DIRECT READ (2,M1A,64,IREC)
      CALL DIRECT READ (3,M1B,64,IREC)
      DO 31 J =1,64
      INPBUF (2*J-1) =ISL(M1A(J),-16)
      INPBUF (2*J)   =IAND(M1A(J),KK)
      INPBUF (128+2*J-1) = ISL(M1B(J), -16)
      INPBUF (128+2*J  ) = IAND(M1B(J), KK)
   31 CONTINUE
      DO 40 J=IXMIN,IXMAX
      R1A  = INPBUF (J)
      R1B  = INPBUF (128+J)
      IF( R1A .GT. 0 .AND. R1B .GT. 0) GO TO 100
      MHL(J)=0
      GO TO 40
  100 HI1(J)=(R1A + R1B)/ICOR
      MHL(J)=C1*((R1A - R1B)/HI1(J))
   40 CONTINUE
      CALL DIRECT WRITE (6,MHL,128,IREC)
      CALL DIRECT WRITE (7,HI1,128,IREC)
   30 CONTINUE
      IF(IBIT3 .EQ. 1) CALL FILTER
   26 CONTINUE
C     PLOT LONGITUDINAL CONTOURS
      ISB =1
      IF(IBIT3 .EQ. 1) ISB =-1
      NCTR = NF1
      DO 90 IC =1, NCTR
   99 KTRVAL(IC) = MCTVAL(IC)
      CALL DRIV1
      IF(NFORM .EQ. 2 .OR. IBIT5 .EQ. 0) CALL HDCOPY
      IF(IBIT5 .EQ. 0) GO TO 36
C     PLOT INTENSITY
      ISB = 3
      NCTR = INUM
      DO 37 II =1,NCTR
   37 KTRVAL(II) = INTVAL(II)
      CALL DRIV1
      CALL HDCOPY
   36 CONTINUE
```

```
    365.           CALL CLOSE(2)
    366.           CALL CLOSE(3)
    367.           CALL CLOSE(6)
    368.           CALL CLOSE(7)
    369.           IF (IBIT3 .EQ. 1) CALL CLOSE(5)
    370.           IF (IBIT4 .EQ. 1 .OR. NP .EQ. 1) STOP
    371.      4000 CONTINUE
    372.           CALL FINITT(0,450)
    373.           STOP
    374.           END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
GENERATED CODE:	1271	004F7
CONSTANTS:	16	00010
LOCAL VARIABLES:	965	003C5
TEMPS:	3	00003
TOTAL PROGRAM:	2255	008CF (PLUS LABELED COMMON)

```
      C
      C
            SUBROUTINE DRIV1
            COMMON /TVAL/ NCTR,KTRVAL(7),NN1,NCTVAL(7),INTVAL(7)
            COMMON /BND/ XMIN,XMAX,YMIN,YMAX,IBIT
            COMMON /CONCOM/ KFLAG1,KFLAG2,FUNCTN,NF,NDIM,NX,NY,X,Y,EXCLUD
            COMMON /ISB/ ISB,NS,IDLCH,LLCH,LEXTRA,NFORM
            COMMON /ABC/A(500),B(500),C(500),Z(130)
            COMMON /MCTVAL/MCTVAL(7),INUM
            COMMON /HL/MRL,HI1
            COMMON /IA/IA
            COMMON /IDAT/DT(4),NZFT,NENH,NEXP,NPOL,XPOS,YPOS
            COMMON /TSA/TSA(60),INHOUS(8)
            COMMON /IBIT/IBIT1,IBIT2,IBIT3,IBIT4,IBIT5,IBIT6,IBIT7
            INTEGER ALPHA(14),DT
            REAL HL(128),HI1(128)
            DIMENSION CVAL(14)
            DATA ALPHA/1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HI,1HJ,1HK,1HL,1HM,
           *          1HN/
            DATA NNN /1/
            IA=0
            IF (ISB .EQ. -1) KU=5
            IF (ISB .EQ.  1) KU=6
            IF (ISB .EQ.  2) KU=6
            IF (ISB .EQ.  3) KU=7
            IF (ISB .EQ.  4) KU=7
            IF (ISB .EQ.  5) KU=8
            IF (ISB .GT. 5  .OR.  ISB .LT. -1) STOP
            IF (NCTR .EQ. 0 .AND.  ISB .EQ.  2) GO TO 1100
      C
      C     PAUSE AT DRIV1
            IPASS =0
            YMIN1 = YMIN
            YMAX1 = YMAX
            NX  = XMAX - XMIN +1
            NY  = YMAX - YMIN +1
            YMAX= 129. - YMIN1
            YMIN= 129. - YMAX1
      C
      C     CHECK TO SEE OF PLOT IS LONGT.
      C
            NDIM =500
            EXCLUD =-1
      C
      C     POSITIVE CONTOUR VALUES
      C
            IF (ISB .GE. 3) GO TO 103
            DO  5  I=1,NCTR
            CVAL(I)=KTRVAL(I)
            A(I) = MCTVAL(I)
          5 CONTINUE
            NC  =NCTR
```

```
C
C     NEGATIVE CONTOUR VALUES
C
          NEG =NN1
          IF(NEG .EQ. 0)  GO TO 104
          DO 6  I=1,NEG
          NC  =NC +1
          A(NC) = NCTVAL(I)
        6 CONTINUE
      104 CONTINUE
          NF = NC
          GO TO 102
      103 DO 7  I=1,INUM
          A(I) = INTVAL(I)
        7 CONTINUE
          NF = INUM
      102 CONTINUE
          IRECP =128-YMIN+1
          KFLAG1 =1
          IBIT =1
       11 CONTINUE
          NNN=NNN+1
          CALL CONTUR
      988 GO TO (21,22,22,24),KFLAG2
       21 IX =X
          IY =Y
          IREC =128-IY
          IF(IREC .EQ. IRECP) GO TO 10
          CALL DIRECT READ (KU,HL,128,IREC)
          IRECP = IRECP - 1
       10 FUNCTN= HL(IX)
          GO TO 11
       22 CONTINUE
          IPASS = IPASS + 1
          CALL DIRECT WRITE (12,A,500,IA)
          CALL DIRECT WRITE (9,B,500,IA)
          CALL DIRECT WRITE (11,C,500,IA)
          IA =IA + 1
          GO TO (99,11,99), KFLAG2
       24 WRITE(10,95)
       95 FORMAT(1X,31H CONTUR NEEDS MORE BUFFER SPACE)
       99 CONTINUE
       40     CONTINUE
          CALL SMPLOT
          CALL RESTAT(TSA)
C         CALL NCNVT(300,30,XMAX)
C         CALL NCNVT(950,30,XMIN)
C         CALL NCNVT(270,50,YMAX1)
C         CALL NCNVT(270,750,YMIN1)
          CALL MOVABS(0,767)
          CALL ANMODE
C         PRINT CONTOUR LEVELS
```

```
      IF(NFORM .EQ. 1 .AND. ISB .GT. 2) GO TO 2000
      IF(ISB .EQ. -1) OUTPUT "FILTERED LONGITUDINAL PLOT"
      IF (ISB .EQ. 1) OUTPUT "TRANSVERSE PLOT"
      IF (ISB .EQ. 2) OUTPUT "INTENSITY PLOT (1A1B)"
      IF (ISB .EQ. 3) OUTPUT "INTENSITY PLOT (2A2B)"
      IF (ISB .EQ. 4) OUTPUT "INTENSITY PLOT (3A3B)"
      WRITE(10,300) DT(4),DT(3),DT(2),DT(1)
  300 FORMAT(1X,"TIME ",Z3,":",Z2,":",Z2,":",Z2)
      WRITE(10,301) NZFT,NENH
  301 FORMAT(1X,"ZEISS FILTER POSITION:",Z4,3X,"ENHANCEMENTS:",I4)
      WRITE(10,303) NEXP,NPOL
  303 FORMAT(1X,"EXPOSRE: 1/",I2,4X,"POLARIZATIONS:",I3)
      WRITE(10,309) XPOS,YPOS
  309 FORMAT(1X,"XPOS:",F6.2,15X,"YPOS:",F6.2)
      WRITE(10,313) XMIN,XMAX,YMIN,YMAX
  313 FORMAT(1X,"APERTURE:X(",I3,",",I3,"),7X,"Y(",I3,",",I3,")")
      IF(ISB .GT. 2) GO TO 314
      IF(NEG .EQ. 0) WRITE(10,305)
      IF(NEG .GT. 0) WRITE(10,306)
      DO 200 I=1,NCTR
      NC=I+7
      INC=I+NCTR
      IF(NEG.EQ.0) WRITE(10,307) ALPHA(I),MCTVAL(I)
      IF(NEG.GT.0) WRITE(10,308)ALPHA(I),MCTVAL(I),ALPHA(INC),NCTVAL(I)
  200 CONTINUE
      GO TO 3000
  314 CONTINUE
      WRITE(10,305)
      DO 315 I=1,INUM
      WRITE(10,307) ALPHA(I),INTVAL(I)
  315 CONTINUE
      GO TO 3000
 2000 CONTINUE
  310 FORMAT(1X,66X,"INTENSITY PLOT")
      WRITE(10,310)
      IF(ISB .EQ. 3) WRITE(10,323)
      IF(ISB .EQ. 4) WRITE(10,324)
      IF(ISB .EQ. 5) WRITE(10,325)
  323 FORMAT(67X,"(1A1B)")
  324 FORMAT(67X,"(2A2B)")
  325 FORMAT(67X,"(3A3B)")
      WRITE(10,311)
      DO 201 I=1,NCTR
      WRITE(10,312) ALPHA(I),INTVAL(I)
  311 FORMAT(/////,67X,"LABEL",4X,"VALUE")
  312 FORMAT(69X,A1,6X,I5)
  201 CONTINUE
 3000 CONTINUE
C     PAUSE BEFORE SWAP PLOT.SV IN DRIV1.SV
  305 FORMAT(/,1X,"LABEL",4X,"VALUE")
  306 FORMAT(/,1X,"LABEL",4X,"VALUE",8X,"LABEL",4X,"VALUE")
  307 FORMAT(3X,A1,6X,I5)
```

EXT. FORTRAN IV, VERSION F02

Program TRANSV

```
      COMMON /TVAL/NCTR,RTRVAL(7),NN1,NCTVAL(7),INTVAL(7)
      COMMON /ABND/AMIN,AMAX
      COMMON /ISB/ISB,NS,IDLCH,LLCH,LEXTRA,NFORM
      COMMON /IDAT/DT(4),NZFT,NENH,NEXP,NPOL,XPOS,YPOS
      COMMON /BND/XMIN,XMAX,YMIN,YMAX,IBIT
      COMMON /NCTVAL/NCTVAL(7),INUM
      COMMON /ABC/A(500),B(500),C(500),Z(130)
      COMMON /BLNK/NBT(4),MF(10),ID(4)
      COMMON /TSA/TSA(60),INHOUS(8)
      INTEGER M2A(128),M2B(128),M3A(128),M3B(128),INPBUF(512)
      INTEGER DT
      DIMENSION IRN(50)
      DIMENSION M00(12),M01(3),M02(3),M03(3),M04(3),M05(3)
      DIMENSION M06(3),M07(3),M08(7),M09(14),M10(13),M11(3)
      DIMENSION M12(3),M13(3),M14(3),M15(11),M16(8),M17(8)
      DIMENSION M18(12),M19(7),M20(7),M21(12),M22(12)
      DIMENSION M23(11),M24(10),M25(11),M26(6),M27(7)
      DATA NBT/7,7,6,10/
      DATA MF/1,2,4,8,10,20,40,80,100,200/
      DATA ID/2Z7F,2Z7F,2Z3F,3Z3FF/
      DATA KK/4ZFFFF/
      DATA KM1A/8ZFFFF0000/
      DATA KZF/4Z1FFF/
      DATA KEH/2ZFF/
      DATA KEXP/1Z7/
      DATA M00/48H(1X,"NUMBER OF CONTOUR LEVELS (MAX. NO. 7 ) ? ")/
      DATA M01/12H(1X,"A=    ")/
      DATA M02/12H(1X,"B=    ")/
      DATA M03/12H(1X,"C=    ")/
      DATA M04/12H(1X,"D=    ")/
      DATA M05/12H(1X,"E=    ")/
      DATA M06/12H(1X,"F=    ")/
      DATA M07/12H(1X,"G=    ")/
      DATA M08/28H(1X,"FORMAT (1 OR 2 ) ?"   )/
      DATA M09/56H(1X,"DELETE ALL OF THE CHARACTER LABELS? (1=YES,0=NO)"
     *) /
      DATA M10/52H(1X,"DELETE THE FIRST CONTOUR LEVEL? (1=YES,0=NO)") /
      DATA M11/12H(1X,"XMIN=")/
      DATA M12/12H(1X,"XMAX=")/
      DATA M13/12H(1X,"YMIN=")/
      DATA M14/12H(1X,"YMAX=")/
      DATA M15/44H(1X,"FOR THE AZIMUTH PLOTS (1=YES,0=NO)?")  /
      DATA M16/32H(1X,"MAXIMUM VALUE (EX. 200)?") /
      DATA M17/32H(1X,"MINIMUM VALUE (EX. 60 )?") /
      DATA M18/48H(1X,"DO YOU WANT INTENSITY PLOTS (1=YES,0=NO)?")/
      DATA M19/28H(1X,"2A,2B (1=YES,0=NO)?")  /
      DATA M20/28H(1X,"3A,3B (1=YES,0=NO)?")  /
      DATA M21/48H(1X,"NUMBER OF INTENSITY VALUES (MAX. NO. 7)?") /
      DATA M22/48H(1X,"WAS ROCESSING INTERRUPTED (1=YES,0=NO)? ") /
      DATA M23/44H(1X,"REPEAT LAST CONTOURS? (1=YES,0 =NO)" ) /
      DATA M24/40H(1X,"INPUT FROM TAPE (1) OR DISK (0) ?")/
      DATA M25/44H(1X,"NUMBER OF MAGNETOGRAMS TO PROCESS? " ) /
```

```
      DATA M26/24H(1X,"(1=YES, 0=NO) ?")  /
      DATA M27/28H(1X,"RECORD/DATA FILE = ")  /
C     IBIT1 : 2A2B
C     IBIT2 : 3A3B
C     IBIT3 : DATA CREATED BEFORE APRIL 18,1980
C     IBIT4 : WAS PROCESSING INTERRUPTED
C     IBIT5 : INTENSITY PLOT
      DO 10 I=1,7
      MCTVAL(I)=0
      NCTVAL(I)=0
   10 INTVAL(I)=0
      IBIT1=0
      IBIT2=0
      IBIT3=0
      IBIT4=0
      IBIT5=0
      IBIT6=0
      IBIT7=0
      NEG=0
      MKEOF=0
      ISB=0
      DO 20 I=1,50
   20 IRN(I)=0
      OUTPUT " "
      OUTPUT " "
      OUTPUT "TRANSVERSE PROGRAM"
      OUTPUT " "
      OUTPUT " "
   17 CONTINUE
      OUTPUT "WAS DATA TO BE PLOTTED CREATED BEFORE APRIL 18,1980"
C CORRECT SEQUENCE SWITCH(2A2B NOW 3A3B)
      CALL ACCEPT(M26,IBIT3)
      IF(IBIT3 .EQ. 0 .OR. IBIT3 .EQ. 1) GO TO 18
      GO TO 17
   18 CONTINUE
      OUTPUT " "
      OUTPUT "IF AZIMUTH PLOTS ONLY;SET NO. OF CONTOUR LEVELS"
      OUTPUT "EQUAL TO ZERO"
      CALL ACCEPT(M00,NF1)
      IF(NF1 .EQ. 0) GO TO 90
      NF=NF1
      NCTR=NF1
      DO 4 I=1,NF
      IF(I .EQ. 1) CALL ACCEPT(M01,MCTVAL(I))
      IF(I .EQ. 2) CALL ACCEPT(M02,MCTVAL(I))
      IF(I .EQ. 3) CALL ACCEPT(M03,MCTVAL(I))
      IF(I .EQ. 4) CALL ACCEPT(M04,MCTVAL(I))
      IF(I .EQ. 5) CALL ACCEPT(M05,MCTVAL(I))
      IF(I .EQ. 6) CALL ACCEPT(M06,MCTVAL(I))
      IF(I .EQ. 7) CALL ACCEPT(M07,MCTVAL(I))
    4 CONTINUE
   90 CONTINUE
```

```
      OUTPUT ' '
      OUTPUT 'THERE ARE TWO DATA FORMATS.'
      OUTPUT '1. 4.25=4.25 INCHES WHERE BOTH THE MAGNETOGRAM'
      OUTPUT '   AND INTENSITY CAN BE PLACED ON THE SAME PAGE '
      OUTPUT ' '
      OUTPUT '2. 6.5=6.5 INCHES ONLY ONE PLOT PER PAGE'
      CALL ACCEPT (M08,NFORM)
      OUTPUT ' '
      CALL ACCEPT(M09,IDLCH)
      OUTPUT ' '
      CALL ACCEPT(M10,LLCH)
      OUTPUT ' '
      OUTPUT 'ENTER ARRAY SIZE (MAXIMUM 128*128)'
      CALL ACCEPTR(M11,XMIN)
      CALL ACCEPTR(M12,XMAX)
      CALL ACCEPTR(M13,YMIN)
      CALL ACCEPTR(M14,YMAX)
      OUTPUT ' '
      OUTPUT 'DO YOU WANT TO INPUT THE MIN. AND MAX. VALUES'
      CALL ACCEPT (M15,IAZ)
      IF(IAZ .EQ. 1) GO TO 15
      AMAX = 200.
      AMIN = 60.
      GO TO 16
   15 CONTINUE
      CALL ACCEPT (M16,AMAX)
      CALL ACCEPT (M17,AMIN)
   16 CONTINUE
      OUTPUT ' '
      CALL ACCEPT(M18,IBIT5)
      IF(IBIT5 .EQ. 0) GO TO 80
      CALL ACCEPT(M19,IBIT1)
      CALL ACCEPT(M20,IBIT2)
      OUTPUT ' '
      CALL ACCEPT(M21,INUM)
      OUTPUT 'INPUT INTENSITY VALUES DIVIDED BY 10'
      DO 81 I=1,INUM
      IF(I .EQ. 1) CALL ACCEPT (M01,INTVAL(I))
      IF(I .EQ. 2) CALL ACCEPT (M02,INTVAL(I))
      IF(I .EQ. 3) CALL ACCEPT(M03,INTVAL(I))
      IF(I .EQ. 4) CALL ACCEPT(M04,INTVAL(I))
      IF(I .EQ. 5) CALL ACCEPT (M05,INTVAL(I))
      IF(I .EQ. 6) CALL ACCEPT(M06,INTVAL(I))
      IF(I .EQ. 7) CALL ACCEPT(M07,INTVAL(I))
   81 CONTINUE
   80 CONTINUE
      OUTPUT ' '
      CALL ACCEPT(M22,IBIT4)
      IBIT6 = 0
      OUTPUT ' '
C     IF(IBIT4 .EQ. 1) CALL ACCEPT(M23,IBIT6)
C     IF(IBIT4 .EQ. 1) GO TO 26
```

```
      OUTPUT ' '
      CALL ACCEPT(M25,NP)
      OUTPUT 'ARE THERE EOF MARKS BETWEEN 1A1B/2A2B/3A3B'
      CALL ACCEPT(M26,NKEOF)
      OUTPUT ' '
      OUTPUT 'INPUT RECORD/DATA NUMBERS OF MAGNETOGRAMS'
      DO 1000 I=1,NP
 1000 CALL ACCEPT(M27,IRN(I))
C
C  SET UP INPUT FILES
C
   26 CONTINUE
C
C   MAG. TAP INPUT
C
      JREC=0
      DO 4000 I3=1,NP
  130 IF(JREC .EQ. (IRN(I3)-1)) GO TO 120
      CALL BUFFERIN(13,1,INPBUF,512,ISTAT)
      JREC=JREC+1
      GO TO 130
  120 CONTINUE
      DO 5 K=2,5
      DO 11 I=1,16
   60 CALL BUFFERIN(13,1,INPBUF,512,ISTAT)
      IF(I .GT. 1) GO TO 201
      IREC=0
  201 CONTINUE
      IWD=1
      DO 45 I1=1,8
      DO 50 J1=1,64
      M2A(J1)=INPBUF(IWD)
      IWD=IWD+1
   50 CONTINUE
      CALL DIRECT WRITE(K,M2A,64,IREC)
      IREC=IREC+1
   45 CONTINUE
      JREC=JREC+1
   11 CONTINUE
    5 CONTINUE
C
C   CHECK HOUSEKEEPING
C
      CALL DIRECT READ (2,M2A,64,0)
      IDAT=ISL(M2A(1),-16)+IAND(ISL(M2A(1),16),KM1A)
      DO 252 I1=1,4
      IF (I1 .EQ. 1) NBB =0
      IF (I1 .GT. 1) NBB = NBT(I1-1) +NBB
      DT(I1)     = IAND(ID(I1),ISL(IDAT,-NBB))
  252 CONTINUE
      WRITE (10,254) DT(4),DT(3),DT(2),DT(1)
  254 FORMAT(//,1X,'DATA TO BE PLOTTED: ',Z3,3('-',Z2))
```

```
      NZFT=IAND(ISL(M2A(2),-16),KZF)
      NENH=IAND(ISL(M2A(2),-8),KEH)
      NEXP=IAND(ISL(M2A(3),-25),KEXP)
      NPOL=IAND(ISL(M2A(3),-19),7)
      WRITE(10,264) NZFT,NENH,NEXP,NPOL
  264 FORMAT(1X,'NZFT=',Z4,1X,'NENH=',I5,1X,'NEXP=',I1,1X,'NPOL=',I1)
      WRITE(10,274) M2A(1),IDAT
  274 FORMAT(1X,'M1A(1)=',Z8,1X'IDAT=',Z8)
      DO 25 I1=1,8
      INHOUS (I1) = M2A(I1)
   25 CONTINUE
      IF(NEXP .EQ. 2) NEXP=60
      IF(NEXP .EQ. 3) NEXP=30
      IF(NEXP .EQ. 4) NEXP=15
      IF(NEXP .EQ. 5) NEXP=8
      IF(NEXP .EQ. 6) NEXP=4
      IXX =ISL(M2A(3),-16)+IAND(ISL(M2A(3),16),KM1A)
      ISNGX=IAND(ISL(IXX,-13),1)
      XPOS=0
      TXPOS=0
      MX=14
C     DO 333 II=1,3
      II=3
  333 CONTINUE
      NN=4
  353 CONTINUE
      TXPOS=TXPOS+IAND(ISL(IXX,-MX),1)*(2**(NN-1))
      NN = NN -1
      MX=MX+1
      IF(NN .GE. 1) GO TO 353
      XPOS=XPOS+TXPOS*(10**(II-1))
      TXPOS=0
      IF(II .LE. 1) GO TO 363
      II=II-1
      GO TO 333
  363 CONTINUE
      IF(ISNGX .EQ. 0) XPOS=-XPOS
      ISNGY=IAND(ISL(IXX,-MX),1)
      YPOS=0
      TYPOS=0
      MX=MX+1
      NN=4
  454 CONTINUE
      TYPOS=TYPOS+(IAND(ISL(IXX,-MX),1))*2**(NN-1)
      NN=NN-1
      MX=MX+1
      IF(NN .GE. 1) GO TO 454
      YPOS=YPOS+TYPOS
      YPOS=YPOS*(10**2)
      TYPOS=0
      MY=23
C     DO 444 II=1,2
```

```
          II=2
      444 CONTINUE
          NN=4
      464 CONTINUE
          IYY=ISL(M2A(4),-16)+IAND(ISL(M2A(4),16),KM1A)
          TYPOS=TYPOS + (IAND(ISL(IYY,-MY),1))*2**(NN-1)
          NN=NN - 1
          MY = MY + 1
          IF (NN .GE. 1) GO TO 464
          YPOS=YPOS + TYPOS*(10**(II-1))
          TYPOS=0
C 444 CONTINUE
          IF(II .LE. 1) GO TO 474
          II=II-1
          GO TO 444
      474 CONTINUE
          IF(ISNGY .EQ. 0) YPOS=-YPOS
          XPOS=XPOS/100.
          YPOS=YPOS/100.
C
C     **********
C
C    SUBTRACT MATRICES
C
          ICOR=2*10
          UOBAR=0.
          ROBAR=0.
          K2=1
          C2=1000.
          IYMIN=YMIN
          IYMAX=YMAX
          IXMIN=XMIN
          IXMAX=XMAX
          DO 30 I=IYMIN,IYMAX
          IREC=I-1
          CALL DIRECT READ(2,M2A,64,IREC)
          CALL DIRECT READ(3,M2B,64,IREC)
          CALL DIRECT READ(4,M3A,64,IREC)
          CALL DIRECT READ(5,M3B,64,IREC)
C    CONVERT 16-BIT INTEGERS TO 32 - BIT INTEGERS
          DO 31 J=1,64
          ITEMP=M2A(65-J)
          M2A(130-2*J)=IAND(ITEMP,KK)
          M2A(129-2*J)=ISL(ITEMP,-16)
          ITEMP=M2B(65-J)
          M2B(130-2*J)=IAND(ITEMP,KK)
          M2B(129-2*J)=ISL(ITEMP,-16)
          ITEMP=M3A(65-J)
          M3A(130-2*J)=IAND(ITEMP,KK)
          M3A(129-2*J)=ISL(ITEMP,-16)
          ITEMP=M3B(65-J)
          M3B(130-2*J)=IAND(ITEMP,KK)
```

```
      M3B(129-2*J)=ISL(ITEMP,-16)
   31 CONTINUE
      DO 40 J=IXMIN,IXMAX
      R2A=M2A(J)
      R2B=M2B(J)
      IF(R2A .GT. 0.0 .AND. R2B .GT. 0.0) GO TO 100
      U=0
      GO TO 43
  100 U=((R2A-R2B)/(R2A+R2B))-UOBAR
   43 R3A=M3A(J)
      R3B=M3B(J)
      IF(R3A .GT. 0.0 .AND. R3B .GT. 0.0) GO TO 1001
      R=0
      GO TO 44
 1001 R=((R3A-R3B)/(R3A+R3B))-ROBAR
   44 A(J)=C2*((U**2+R**2)**.25)
      B(J)=(R3A+R3B)/ICOR
      C(J)=(R2A+R2B)/ICOR
C     TEST WHEN DATA WAS TAKEN
C
      IF(IBIT3 .EQ. 1) GO TO 39
      Z(J)=-45
      IF(R .EQ. 0. .AND. U.GE. 0.) GO TO 40
      Z(J)=+45
      IF(R .EQ. 0. .AND. U.LT. 0.) GO TO 40
      W=-U/R
      Z(J)=.5*ATAN(W)*(180./3.14159)
      IF(U .GE. 0. .AND. R .LT. 0.)Z(J)=-90+Z(J)
      IF(U .LT. 0. .AND. R.LT.0.) Z(J)=Z(J)+90
      GO TO 40
   39 CONTINUE
      Z(J)=-45
      IF(U .EQ. 0.0 .AND. R .GE. 0.0) GO TO 40
      Z(J)=45
      IF(U .EQ. 0.0 .AND. R.LT. 0.0) GOTO 40
      W=-R/U
      Z(J)=.5*ATAN(W)*(180./3.14159)
      IF(R .GE. 0.0 .AND. U .LT. 0.0)Z(J)=-90+Z(J)
      IF(R .LT. 0.0 .AND. U .LT. 0.0)Z(J)=Z(J)+90
   40 CONTINUE
      KREC=IREC+128
C     TRANSVERSE PLOT ISB=2
      CALL DIRECT WRITE(6,A,128,IREC)
C     AZPLOT ISB=1
      CALL DIRECT WRITE(6,Z,128,KREC)
C     2A2B PLOT ISB=4
      CALL DIRECT WRITE(7,C,128,IREC)
C     3A3B PLOT ISB=5
      CALL DIRECT WRITE(8,B,128,IREC)
   30 CONTINUE
C
C  PLOT AZIMUTHE PLOTS
```

```
      C
            CALL AZPLT
            IF(NFORM .EQ. 2 .OR. IBIT1 .EQ. 0 .OR. IBIT5 .EQ. 0) CALL HDCOPY
      C
      C   PLOT 2A2B INTENSITY
      C
            IF(IBIT5 .EQ. 0 .OR. IBIT1 .EQ. 0) GO TO 98
            ISB=4
            CALL DRIV1
            CALL HDCOPY
      C
      C   PLOT TRANSVERSE CONTOURS
      C
         98 CONTINUE
            ISB=2
            IF(NGTR .EQ. 0) GO TO 92
            CALL DRIV1
            IF(NFORM .EQ. 2 .OR. IBIT5 .EQ. 0 .OR. IBIT2 .EQ. 0)CALL HDCOPY
         92 CONTINUE
      C
      C   PLOT 3A3B INTENSITY
      C
            ISB=5
            IF(IBIT5 .EQ. 0 .OR. IBIT2 .EQ. 0) GO TO 99
            CALL DRIV1
            CALL HDCOPY
         99 CONTINUE
            IF(IBIT4 .EQ. 1 .OR. NP .EQ. 1) STOP
       4000 CONTINUE
            CALL FINITT(0,450)
            STOP
            END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
GENERATED CODE:	1365	00555
CONSTANTS:	28	0001C
LOCAL VARIABLES:	1333	00535
TEMPS:	3	00003
TOTAL PROGRAM:	2729	00AA9 (PLUS LABELED COMMON)

ACCEPT

```
1.          SUBROUTINE ACCEPT(IAR,IDUM)
2.          DIMENSION IAR(1)
3.          WRITE(101,IAR)
4.          READ(101,10)IDUM
5.    10    FORMAT(I)
6.          RETURN
7.          END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
GENERATED CODE:	18	00012
CONSTANTS:	0	00000
LOCAL VARIABLES:	1	00001
TEMPS:	3	00003
TOTAL PROGRAM:	22	00016

```
50      1.            SUBROUTINE ACCEPTR(IAR,XDUM)
        2.            DIMENSION IAR(1)
        3.            WRITE(101,IAR)
        4.            READ(101,10)XDUM
        5.      10    FORMAT(F)
        6.            RETURN
        7.            END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS
GENERATED CODE:	18	00012
CONSTANTS:	0	00000
LOCAL VARIABLES:	1	00001
TEMPS:	3	00003
TOTAL PROGRAM:	22	00016

EXT. FORTRAN IV, VERSION F02

FILTER

```
      1.          SUBROUTINE FILTER
      2.          COMMON /BND/ SMIN,XAX,YMIN,YMAX,IBIT
      3.          INTEGER  WF1,WF2,WF3
      4.          REAL FHL(128),MHL(384),MHL2,MHL3
      5.          EQUIVALENCE (MHL2,MHL(129)),(MHL3,MHL(257))
      6.          IXMIN=XMIN
      7.          IXMAX=XMAX
      8.          IYMIN=YMIN-1
      9.          IYMAX=YMAX
     10.          WF1=20
     11.          WF2=3
     12.          WF3=2
     13.          TW=WF1+4*WF2+4*WF3
     14.      * THE FOPEN FUNCTION WILL BE ACHIEVED VIA
     15.      * !SET F:5 /FHL;OUTIN;SAVE;KEYED;DIRECT
     16.      *     CALL FOPEN(5,'FHL',512)
     17.          DO 10 I=1,128
     18.      10  FHL(I)=0
     19.      C   ZERO EDGES OF FILTERED MAGNETOGRAM
     20.      *   CALL WRITR(5,0,FHL,1,IER)
     21.      *   CALL WRITR(5,127,FHL,1,IER)
     22.          CALL DIRECT WRITE (5,FHL,128,0)
     23.          CALL DIRECT WRITE (5,FHL,128,127)
     24.      C   READ 3 LINES OF DATA FOR AVERAGING
     25.      * PRIME THE INPUT BUFFERS WITH THE FIRST TWO RECORDS
     26.          CALL DIRECT READ (6,MHL,128,0)
     27.          CALL DIRECT READ (6,MHL2,128,1)
     28.          DO 20 IR=IYMIN,IYMAX
     29.      *   IREC=IR-1
     30.          IREC = IR+1
     31.      *   CALL READR(6,IREC,MHL,3,IER)
     32.          CALL DIRECT READ (6,MHL3,128,IREC)
     33.          FHL(1)=0
     34.          FHL(128)=0
     35.          DO 30 J=2,127
     36.          IC1=J-1
     37.          IC2=J
     38.          IC3=J+1
     39.          IC4=J+127
     40.          IC5=J+128
     41.          IC6=J+129
     42.          IC7=J+255
     43.          IC8=J+256
     44.          IC9=J+257
     45.          A=WF1*MHL(IC5)
     46.          B=WF2*(MHL(IC2)+MHL(IC4)+MHL(IC6)+MHL(IC8))
     47.          C=WF3*(MHL(IC1)+MHL(IC3)+MHL(IC7)+MHL(IC9))
     48.          FHL(J)=(A+B+C)/TW
     49.      30  CONTINUE
     50.      *   CALL WRITR(5,IR,FHL,1,IER)
     51.          CALL DIRECT WRITE (5,FHL,128,IR)
     52.      * IN-CORE SHUFFLE TO PREPARE FOR NEXT READ
```

```
53.           DO 15 J=1,256
54.     15    MHL(J) = MHL(J+128)
55.     20    CONTINUE
56.     *  THE CP-V IMPLEMENTATION CLOSES F:5 IN LONGT
57.     *     CALL FCLOS(5)
58.           RETURN
59.           END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS	
GENERATED CODE:	163	000A3	
CONSTANTS:	7	00007	
LOCAL VARIABLES:	539	00218	
TEMPS:	2	00002	
TOTAL PROGRAM:	711	002C7	(PLUS LABELED COMMON)

SMPLOT

```
      C     SMPLOT.SV
      C     THIS ROUTINE CALLS EITHER CPLOT1 OR CPLOT2 AFTER
      C     BEING SWAPPED
            SUBROUTINE SMPLOT
            COMMON /TVAL/NCTR,KTRVAL(7),NN1,NCTVAL(7),INTVAL(7)
            COMMON /ISB/ISB,NS,IDLCH,LLCH,LEXTRA,NFORM
            COMMON /BND/XMIN,XMAX,YMIN,YMAX,IBIT
            COMMON /ABC/A(500),B(500),C(500),Z(130)
            COMMON /TSA/TSA(60),INHOUS(8)
            COMMON /IA/IA
            YMIN1=YMIN
            YMAX1=YMAX
            YMIN=129.-YMAX1
            YMAX=129.-YMIN1
            IF(NFORM .NE. 1) GO TO 10
            IF(NFORM .EQ. 1 .AND. ISB .LE. 2) GO TO 10
            GO TO 20
         10 CALL INITT(960)
            CALL NEWPAG
            GO TO 30
         20 CALL RESTAT(TSA)
         30 CONTINUE
            CALL CPLT1
            IF (ISB .EQ. 2 .AND . NCTR .EQ. 0) GO TO 1001
            DO 1000 IA2=1,IA
            IP=IA2-1
            CALL DIRECT READ (12,A,500,IP)
            CALL DIRECT READ (9,B,500,IP)
            CALL DIRECT READ (11,C,500,IP)
      C
      C     SET FOR SOLID CONTOURS, CSCAN SETS THE DASHED LINE FLAG
      C C
            NS=0
            CALL CPLT2
       1000 CONTINUE
       1001 CONTINUE
            CALL SVSTAT(TSA)
            RETURN
            END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS	
GENERATED CODE:	74	0004A	
CONSTANTS:	6	00006	
LOCAL VARIABLES:	5	00005	
TEMPS:	1	00001	
TOTAL PROGRAM:	86	00056	(PLUS LABELED COMMON)

CPLT1

```
    1.          SUBROUTINE CPLT1
    2.      C   THIS ROUTINE DRAWS AN AXIS AND LABELS IT.
    3.          COMMON /TVAL/NF1,MCTVAL(7),NN1,NCTVAL(7),INTVAL(7)
    4.          COMMON /ISB/ ISB,NS,IDLCH,LLCH,LEXT,NFORM
    5.          COMMON /BND/ XMIN,XMAX,YMIN,YMAX,IBIT
    6.          COMMON /ABC/ A(500),B(500),C(500),Z(130)
    7.          COMMON /TSA/TSA(60),INHOUS(8)
    8.          COMMON /VAR/XD,YD
    9.          COMMON /VAR2/IALPHA
   10.          XD=0.
   11.          YD=0.
   12.          XRANGE=XMAX-XMIN
   13.          YRANGE=YMAX-YMIN
   14.          IF(XMIN.LT.0) XD=-XMIN
   15.          IF(YMIN.LT.0) YD=-YMIN
   16.          XMIN1=XMIN+XD
   17.          YMIN1=YMIN+YD
   18.          XMAX1=XMAX+XD
   19.          YMAX1=YMAX+YD
   20.          CALL VWINDO(XMIN1,XRANGE,YMIN1,YRANGE)
   21.          IF(NFORM .EQ. 2) CALL SWINDO(300,700,0,700)
   22.          IF(NFORM.EQ.1.AND.ISB.LT.3)CALL SWINDO(0,510,0,510)
   23.          IF(NFORM.EQ.1.AND.ISB.GE.3)CALL SWINDO(510,510,0,510)
   24.          CALL MOVEA(XMIN1,YMIN1)
   25.          CALL DRAWA(XMAX1,YMIN1)
   26.          CALL DRAWA(XMAX1,YMAX1)
   27.          CALL DRAWA(XMIN1,YMAX1)
   28.          CALL DRAWA(XMIN1,YMIN1)
   29.    100   CONTINUE
   30.          RETURN
   31.          END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS	
GENERATED CODE:	94	0005E	
CONSTANTS:	4	00004	
LOCAL VARIABLES:	7	00007	
TEMPS:	1	00001	
TOTAL PROGRAM:	106	0006A	(PLUS LABELED COMMON)

EXT. FORTRAN IV, VERSION F02

CPLT2

```
      *     COMPILER NOSTACK
            SUBROUTINE CPLT2
            COMMON /TVAL/NF1,MCTVAL(7),NN1,NCTVAL(7),INTVAL(7)
            COMMON /ISB/ISB,NS,IOLCH,LLCH,LEXTRA,NFORM
            COMMON /BND/XMIN,XMAX,YMIN,YMAX,IBIT
            COMMON /ABC/A1(500),B1(500),C1(500),Z1(130)
            COMMON /TSA/TSA(60),INHOUS(8)
            COMMON /VAR/XD,YD
            COMMON /VAR2/IALPHA
            COMMON /SCOM/KSTART,KGO,HLABEL,TLABEL,IZ,Z,IARC,X,Y
            LOGICAL HLABEL,TLABEL,TLSAVE
            INTEGER IALPHA
            LOGICAL FIRST
      C     RESET THE WINDOW AFTER BEING SWAPPED
            XD=0.
            YD=0.
            NPTS=0
            XRANGE=XMAX-XMIN
            YRANGE=YMAX-YMIN
            IF(XMIN.LT.0) XD=-XMIN
            IF(YMIN.LT.0) YD=-YMIN
            XMIN1=XMIN+XD
            YMIN1=YMIN+YD
            CALL VWINDO(XMIN1,XRANGE,YMIN1,YRANGE)
            IF (NFORM.EQ.1.AND.ISB.LT.3)CALL SWINDO(0,510,0,510)
            IF (NFORM .EQ. 1. AND.ISB.GE.3)CALL SWINDO(510,510,0,510)
            IF(NFORM .EQ. 2) CALL SWINDO(300,700,0,700)
      C
            FIRST=.TRUE.
            KSTART=1
     10     CALL CSCAN
      C     KGO IS SET BY CSCAN
      C     KGO=1 NEW CONTUR VALUE
      C         2 NEW ARC
      C         3 NEW SUB-ARC AFTER GAP
      C         4 NEW POINT
      C         5 FINISHED
            GO TO (12,24,15,32,20),KGO
     12     IF(.NOT. FIRST) GO TO 24
            FIRST=.FALSE.
            GO TO 28
      C      GAP
     15     NPTS=0
            GO TO 32
     20     IF(FIRST) GO TO 40
      C      TERMINATE PREVIOUS ARC
     24     NPTS=0
      C      LABEL END OF PREVIOUS ARC
            IF(TLSAVE) CALL PPLOT(XX,YY)
            GO TO (28,30,99,99,36),KGO
     28     IALPHA=1+MOD(IZ-1,26)
      C      LABEL BEGINNING OF NEW ARC
```

```
53.     30      TLSAVE=TLABEL
54.             IF(HLABEL) CALL PPLOT(X,Y)
55.     C        AD A POINT TO ARC
56.     32      CONTINUE
57.             IALPHA=1+MOD(IZ-1,26)
58.             CALL SDMPUT(X,Y,NPTS)
59.             XX=X
60.             YY=Y
61.             GO TO 10
62.     C        PROGRAM NEVER GOES TO 99
63.     99      CONTINUE
64.     36      CONTINUE
65.     40      RETURN
66.             END

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

                    DEC       HEX
                   WORDS     WORDS
                   -----     -----
  GENERATED CODE:    148     00094
       CONSTANTS:      5     00005
  LOCAL VARIABLES:    10     0000A
           TEMPS:      1     00001
                   -----     -----
   TOTAL PROGRAM:    164     000A4   (PLUS LABELED COMMON)
```

EXT. FORTRAN IV, VERSION F02

SDNPUT

```
      SUBROUTINE SDNPUT(XX,YY,NPTS)
      COMMON /ISB/ISB,NS,IDLCH,LLCH,LEXTRA,NFORM
      COMMON /BND/XMIN,XMAX,YMIN,YMAX
      COMMON /STAT/TSA(60),INHOUS(8)
      COMMON /VAR/XD,YD
      COMMON /VAR2/IALPHA
      NPTS=NPTS+1
      XX1=XX+XD
      YY1=YY+YD
      IF(NPTS.EQ.1) GO TO 10
      IF(NFORM.EQ.2)CALL SWINDO(300,700,0,700)
      IF(NFORM.EQ.1.AND.ISB.LT.3)CALL SWINDO(0,510,0,510)
      IF(NFORM.EQ.1.AND.ISB.GE.3)CALL SWINDO(510,510,0,510)
      CALL MOVEA(X1,Y1)
      IF(NS.EQ.0  ) CALL DRAWA(XX1,YY1)
      IF(NS.EQ.0)NN=0
      IF(NS.EQ.0)GO TO 30
      XM=XMAX-XMIN
      IF(XM.GT.115.AND.NFORM.EQ.1)GO TO 20
      IF(NS.EQ.1) CALL DASHA(XX1,YY1,12)
      GO TO 30
   20 CONTINUE
      NN=NN+1
      NTEST=MOD(NN,3)
      IF(NTEST.EQ. 0) CALL MOVEA(XX1,YY1)
      IF(NTEST .EQ. 1) CALL DRAWA(XX1,YY1)
      IF(NTEST.EQ.2) CALL DRAWA(XX1,YY1)
   30 CONTINUE
      IF (NS.LT.0 .OR. NS.GT.1) OUTPUT " NS IS NOT DEFINED CORRECTLY"
   10 X1=XX1
      Y1=YY1
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS	
GENERATED CODE:	132	00084	
CONSTANTS:	7	00007	
LOCAL VARIABLES:	8	00008	
TEMPS:	4	00004	
TOTAL PROGRAM:	151	00097	(PLUS LABELED COMMON)

64 EXT. FORTRAN IV, VERSION F02

CSCAN

```
      SUBROUTINE CSCAN
      COMMON /ISB/ISB,NS,IDLCH,LLCH,LEXTRA,NFORM
      COMMON /BND/ XMIN,XMAX,YMIN,YMAX
      COMMON /ABC/ BUFA(500),BUFB(500),BUFC(500),BUFZ(130)
      COMMON /TSA/TSA(60),INHOUS(8)
      COMMON /SCOM/KSTART,KGO,HLABEL,TLABEL,IZ,Z,IARC,X,Y
      COMMON /BLK3/LABF
      COMMON /IVAL/ NCTR,KIRVAL(7),NN1,NCTVAL(7),INTVAL(7)
C     KSTART IS SET =1 BY USER TO INITIATE NEW CASE
C     KGO=1 NEW CONTUR VALUE
C         2 NEW ARC AND NEW POINT AND LABEL FLAGS
C         3 NEW SUB-ARC AFTER GAP, AND NEW POINT
C         4 NEW POINT
C         5 FINISHED, NO NEW POINT
      LOGICAL HLABEL,TLABEL
      REAL IBUFA(500),IBUFB(500),IBUFC(500)
      INTEGER KSTART,KGO,IZ,IARC
      INTEGER JCVL,JADL,JAPL
      REAL X,Y,Z
      REAL BUFA,BUFB,BUFC
      EQUIVALENCE(IBUFA,BUFA),(IBUFB,BUFB),(IBUFC,BUFC)
      DATA LABF/32000/
      NF1=NCTR
      GO TO (11,15),KSTART
11    KSTART=2
      JCVL=1
      IZ=1
12    JADL=IBUFB(JCVL)
      IF(JADL.EQ.0) GO TO 16
      IARC=1
      KGO =1
      Z=BUFA(JCVL)
C      HERE AT BEGINING OF NEW ARC
13    JAPL=ABS(IBUFA(JADL))
      JTRY=JAPL-LABF
      IF(JTRY.GT.0) JAPL=JTRY
      HLABEL=(IBUFA(JADL).LT.0).AND.(IBUFA(JADL).GT.(-LABF))
      TLABEL=(IBUFB(JADL).LT.0).AND.(IBUFB(JADL).GT.(-LABF))
148   X=XMAX-BUFA(JAPL)+XMIN
      Y=BUFB(JAPL)
      RETURN
15    JAPL=IBUFC(JAPL)
      IF(JAPL) 151,155,152
C      CROSSING A GAP
151   JAPL=-JAPL
      KGO=3
      GO TO 148
C      ORDINARY NEW POINT
152   KGO=4
      IF(X.EQ.BUFA(JAPL).AND.Y.EQ.BUFB(JAPL)) GO TO 15
      GO TO 148
155   KGO =2
```

```
99   53.            JADL=IBUFC(JADL)
     54.            IARG=IARG+1
     55.            IF(JADL.GT.0) GO TO 13
     56.     16     JCVL=IBUFG(JCVL)
     57.            IZ=IZ+1
     58.            NDASH=NF1+1
     59.            IF(NS.EQ.1) GO TO 20
     60.            IF(NDASH.EQ.JCVL) NS=1
     61.     20     CONTINUE
     62.            IF(JCVL.NE.0) GO TO 12
     63.            KGO=5
     64.            RETURN
     65.            END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS	
GENERATED CODE:	127	0007F	
CONSTANTS:	0	00000	
LOCAL VARIABLES:	7	00007	
TEMPS:	1	00001	
TOTAL PROGRAM:	135	00087	(PLUS LABELED COMMON)

EXT. FORTRAN IV, VERSION F02

CONTUR

```
      1.     *     COMPILER NOSTACK
      2.           SUBROUTINE CONTUR
      3.           COMMON /TVAL/ NP1,MCTVAL(7),NN1,NCTVAL(7),INTVAL(7)
      4.           COMMON /ISB/ISB,NS,IOLCH
      5.           COMMON /BND/ XMIN,XMAX,YMIN,YMAX,IBIT
      6.           COMMON /ABC/BUFA(500),BUFB(500),BUFC(500),Z(130)
      7.           COMMON /TSA/TSA(60),INHOUS(8)
      8.           COMMON /CONCOM/KFLAG1,KFLAG2,FUNCTN,NF,NDIM,NX,NY,XX,YY,EXCLUD
      9.           COMMON /AVLCOM/AVAIL
     10.           COMMON /JUNK/IZ,LABT,KX,KY,KZ1,KZ2,LABF,BIG
     11.     C     KFLAG1 IS INITIALLY SET=1 BY USER. CONTUR RESETS KFLAG=2 OR 3,
     12.     C     HOWEVER USER IS PERMITTED TO RESET KFLAG1=3. GOOD LUCK!
     13.     C     KFLAG=1 BEGIN NEW CASE
     14.     C     KFLAG1=2 GO TO NEXT TO CONTINUE PROCESSING
     15.     C     KFLAG1=3 CALL CTRIM TO TRIM LISTS, THEN GO TO NEXT TO
     16.     C      CONTINUE PROCESSING
     17.     C     KFLAG2 IS SET BY CONTUR TO BE USED IN A COMPUTED GO-TO
     18.     C     IN THE USER PROGRAM
     19.     C     KFLAG2=1 REQUESTING FUNCTION EVALUATION
     20.     C     KFLAG2=2 PROCESSING INTERUPTED. AVAILABLE SPACE USED UP.
     21.     C      USER SHOULD PLOT AND/OR PRINT EXISTING
     22.     C      CONTOUR LIST AND THEN RE-ENTER CONTUR
     23.     C      FOR FURTHER PROCESSING
     24.     C     KFLAG2=3 PROCESSING COMPLETED. USER SHOULD PLOT
     25.     C      AND/OR PRINT EXISTING CONTOUR LISTS
     26.     C     KFLAG2=4 CATASTROPHIC SHORTAGE OF AVAILABLE SPACE.
     27.     C      THE CASE CAN NOT BE COMPLETED.
     28.           REAL X(5),Y(5),XCOOR(2),YCOOR(2),ZLO(5),ZHI(5)
     29.           INTEGER SAVE0,SAVE1,SAVE2,SIGN1,SIGN2,AVAIL,AVAILN
     30.           INTEGER POINT1,POINT2,SWTCH1,SWTCH2
     31.           INTEGER KZ1(4),KZ2(4),KX(4),KY(4),LAB(3)
     32.           REAL IBUFA(500),IBUFB(500),IBUFC(500)
     33.           INTEGER RULE(4),MODE(5),POINT,POINTR,SWITCH
     34.           EQUIVALENCE (BUFA(1),IBUFA(1)),(BUFB(1),IBUFB(1)),
     35.          * (BUFC(1),IBUFC(1))
     36.           EQUIVALENCE (IZ,JCOL)
     37.           DATA KX/1,1,2,1/
     38.           DATA KY/1,1,1,2/
     39.           DATA KZ1/2,2,3,0/
     40.           DATA KZ2/0,3,1,1/
     41.           DATA LABF/32000/,LABT/0/
     42.           DATA BIG/.17E37/
     43.     C     ************************************************************
     44.     C       BRANCH ON KFLAG1
     45.           GO TO (104,102,101),KFLAG1
     46.     C      TRIM LISTS TO RETRIEVE AVAILABLE SPACE
     47.     101   CALL CTRIM(AVAIL)
     48.           IF(IBUFA(AVAIL).GT.0) GO TO 102
     49.     C       CATASTROPHIC SPACE SHORTAGE
     50.           KFLAG2=4
     51.           RETURN
     52.     102   GO TO NEXT,(115,141,18,20)
```

```
104   CONTINUE
C     ****************************************************************
      IF(NF.NE.0) GO TO 106
C       SET BRANCHES FOR NF .EQ. 0
      ASSIGN 1165 TO LINK1
      ASSIGN 145 TO LINK2
      ASSIGN 831 TO LINK4
      ASSIGN 855 TO LINK5
      IBUFB(1)=0
      IBUFC(1)=0
      GO TO 111
C       SET BRANCHES FOR NF .GT. 0
106   ASSIGN 116 TO LINK1
      ASSIGN 143 TO LINK2
      ASSIGN 185 TO LINK4
      ASSIGN 845 TO LINK5
C     INITIALIZE AVAILABLE SPACE
      NF1=NF+1
      NDIMM5=NDIM-5
      DO 110 I=NF1,NDIM
      IBUFA(I)=NDIMM5-I
110   IBUFC(I)=I+1
      IBUFC(NDIM)=0
      AVAIL=NF1
C     ****************************************************************
C     INITIALIZE CVL (CONTOUR VALUE LIST)
      DO 108 I=1,NF
      IBUFB(I)=0
108   IBUFC(I)=I+1
      IBUFC(NF)=0
C     ****************************************************************
C     INITIALIZE RULE ARRAY
C     RULE(I)=RULE TO BE USED FOR SIDE I
C     WHERE    -1=INACTIVE
C       0=SEARCH
C      +1=ACTIVE
      RULE(1)=-1
      RULE(2)=-1
      RULE(3)=+1
      RULE(4)=+1
C     ****************************************************************
C     SET UP X AND Y ARRAYS
111   CONTINUE
      NX1=NX-1
      NY1=NY-1
      STEPX=(XMAX-XMIN)/FLOAT(NX1)
      STEPY=(YMAX-YMIN)/FLOAT(NY1)
      XCOOR(1)=XMIN
      YCOOR(1)=YMIN
      XCOOR(2)=XMIN+STEPX
      YCOOR(2)=YMIN+STEPY
C     SET UP INITIAL Z ARRAY
```

```
      NZMAX=NX1+3
      K=1
      XX=XCOOR(1)
      YY=YCOOR(1)
      ASSIGN 115 TO NEXT
C     ***************************************************************
C       RETURN FOR FUNCTION EVALUATION
112   KFLAG2=1
      KFLAG1=2
      RETURN
C     ***************************************************************
115   CONTINUE
      GO TO LINK1,(116,1165)
116   CONTINUE
      Z(K+2)=FUNCTN
1165  CONTINUE
      XX=XX+STEPX
      K=K+1
      IF(K.LE.NX) GO TO 112
C     ***************************************************************
C     BEGIN LOOP ON ROWS
      IROW=1
C     RETURN HERE FROM 85+ FOR NEXT ROW OF MATRIX
118   CONTINUE
      XX=XCOOR(1)
      YY=YCOOR(2)
      ASSIGN 141 TO NEXT
      GO TO 112
C     ***************************************************************
141   CONTINUE
      GO TO LINK2,(143,145)
143   CONTINUE
      Z(1)=FUNCTN
145   CONTINUE
C     ***************************************************************
C     SET UP LOOP ON COLUMNS
      JCOL=1
C     RETURN HERE FROM 83+ FOR NEXT COLUMN OF MATRIX
149   CONTINUE
      XX=XCOOR(2)
      YY=YCOOR(2)
      ASSIGN 18 TO NEXT
      GO TO 112
C     ***************************************************************
C     ***************************************************************
18    CONTINUE
      GO TO LINK4,(185,831)
185   CONTINUE
      Z(IZ+1)=FUNCTN
C     ***************************************************************
C       SET XLO( ),ZHI( ) FOR NEW SQUARE
      IEXC=0
```

```
157.            ZLO(5)=BIG
158.            ZHI(5)=-BIG
159.            DO 188 ISIDE=1,4
160.            I1=KZ1(ISIDE)+IZ
161.            I2=KZ2(ISIDE)+IZ
162.            IF(Z(I1).EQ.EXCLUD.OR.Z(I2).EQ.EXCLUD) GO TO 186
163.            ZLO(ISIDE)=AMIN1(Z(I1),Z(I2))
164.            ZHI(ISIDE)=AMAX1(Z(I1),Z(I2))
165.            ZLO(5)=AMIN1(ZLO(5),ZLO(ISIDE))
166.            ZHI(5)=AMAX1(ZHI(5),ZHI(ISIDE))
167.            GO TO 188
168.    186     ZLO(ISIDE)=EXCLUD
169.            ZHI(ISIDE)=EXCLUD
170.            IEXC=IEXC+1
171.    188     CONTINUE
172.            IF(IEXC.EQ.4) GO TO 831
173.    C       SET UP LOOP ON CVL
174.            ICNTR=1
175.    C       ************************************************************
176.    C       RETURN HERE FROM 83+ FOR NEXT CONTUR VALUE
177.    19      CONTINUE
178.            VALUE=BUFA(ICNTR)
179.            IF((ZLO(5).GT.VALUE).OR.(VALUE.GE.ZHI(5))) GO TO 83
180.    C       ************************************************************
181.    C         TEST AVAILABLE SPACE
182.            IF(IBUFA(AVAIL).GT.0) GO TO 20
183.    C       INSUFFICIENT SPACE IN BUFFERS
184.            KFLAG1=3
185.            KFLAG2=2
186.            ASSIGN 20 TO NEXT
187.            RETURN
188.    20      CONTINUE
189.            IS=1
190.            DO 36 ISIDE=1,4
191.            IF((ZLO(ISIDE).GT.VALUE).OR.(VALUE.GE.ZHI(ISIDE)))GO TO 36
192.            IX=KX(ISIDE)
193.            IY=KY(ISIDE)
194.            I1=KZ1(ISIDE)+IZ
195.            I2=KZ2(ISIDE)+IZ
196.            DELZ=Z(I2)-Z(I1)
197.            FRACTN=(VALUE-Z(I1))/DELZ
198.            GO TO (32,33,32,33),ISIDE
199.    32      X(IS)=XCOOR(IX)
200.            Y(IS)=YCOOR(IY)+FRACTN*STEPY
201.            GO TO 35
202.    33      X(IS)=XCOOR(IX)+FRACTN*STEPX
203.            Y(IS)=YCOOR(IY)
204.    35      MODE(IS)=RULE(ISIDE)
205.            IS=IS+1
206.    36      CONTINUE
207.    C       ************************************************************
208.    C       ************************************************************
```

```
C     TEST FOR NUMBER OF CROSSINGS
      GO TO (83,362,37,363,364),IS
C     ***************************************************************
C     HERE FOR 1 CROSSING - SHOULD HAPPEN WITH EXCLUDED MESHPOINTS
362   MODE(2)=-1
      X(2)=X(1)
      Y(2)=Y(1)
      IS=3
C     ***************************************************************
      GO TO 37
C     HERE FOR ERRONEOUS CROSSING COUNT
363   WRITE(10,3631)
3631  FORMAT(2X,25HINDICATION OF 3 CROSSINGS)
      CALL EXEN(1000)
C     ***************************************************************
C     HERE FOR 4 CROSSINGS
364   IF(X(2).LE.X(4)) GO TO 37
      MODE(5)=MODE(4)
      X(5)=X(4)
      Y(5)=Y(4)
      MODE(4)=MODE(2)
      X(4)=X(2)
      Y(4)=Y(2)
      MODE(2)=MODE(5)
      X(2)=X(5)
      Y(2)=Y(5)
C     ***************************************************************
C     RETURN TO 37 FROM 714 FOR SECOND PAIR OF CROSSINGS (IF ANY)
37    CONTINUE
      LAB(1)=LABT
      LAB(2)=LABT
C     RETURN TO 38 FROM 545- IF SEARCH FOR P1 FAILS
38    CONTINUE
C       BRANCH ACCORDING TO MODE(1) AND MODE(2)
      KEY=3*MODE(1)+MODE(2)+5
      GO TO (60,42,60,43,44,43,60,42,60),KEY
C     ***************************************************************
C     HERE IF KEY=2,8
C     (I,S),(A,S)
C     EXCHANGE ROLES OF POINTS 1 AND 2 AND USE 43
42    MODE(5)=MODE(1)
      X(5)=X(1)
      Y(5)=Y(1)
      LAB(3)=LAB(1)
      MODE(1)=MODE(2)
      X(1)=X(2)
      Y(1)=Y(2)
      LAB(1)=LAB(2)
      MODE(2)=MODE(5)
      X(2)=X(5)
      Y(2)=Y(5)
      LAB(2)=LAB(3)
```

```
      GO TO 43
C     ************************************************************
C     HERE IF KEY=4,6
C     (S,I),(S,A)
C     SEARCH ON P1 AND ATTACH P2
43    POINT=0
      ASSIGN 435 TO KRTN
      GO TO 49
435   CONTINUE
C       ATTACH P2 TO EXISTING ARC
      BUFA(AVAIL)=X(2)
      BUFB(AVAIL)=Y(2)
      AVAILN=IBUFC(AVAIL)
C     IF SWITCH=0--ATTACH NEW POINT TO HEAD OF ARC LIST
C     IF SWITCH=1--ATTACH NEW POINT OF TAIL OF ARC LIST
      IF(SWITCH.NE.0) GO TO 437
      IBUFC(AVAIL)=POINTR
      IBUFA(IXADL)=ISIGN(AVAIL,MODE(2))-LAB(2)
      GO TO 439
C     HERE TO ATTACH TO TAIL LIST
437   IBUFC(AVAIL)=0
      IBUFC(POINTR)=AVAIL
      IBUFB(IXADL)=ISIGN(AVAIL,MODE(2))-LAB(2)
439   AVAIL=AVAILN
      GO TO 70
C     ************************************************************
C     HERE IF KEY=5
C     (C,C)
C     (S,S)
C     SEARCH FOR P1 AND P2 -- JOIN THE ARCS
44    POINT=0
      ASSIGN 445 TO KRTN
      GO TO 49
445   CONTINUE
      POINT1=POINTR
      SWTCH1=SWITCH
      IXADL1=IXADL
      POINT=1
      ASSIGN 475 TO KRTN
      GO TO 49
475   CONTINUE
      POINT2=POINTR
      SWTCH2=SWITCH
      IXADL2=IXADL
      IF(IXADL1.NE.IXADL2) GO TO 4753
C       COMPLETE A CLOSED CURVE
C       SET FLAGS TO LABEL HEAD BUT NOT TAIL
      IHEAD=IBUFA(IXADL1)
      IBUFA(IXADL1)=-IHEAD
      ITAIL=IBUFB(IXADL1)
      IBUFC(ITAIL)=AVAIL
      AVAILN=IBUFC(AVAIL)
```

```
      IBUFB(IXADL1)=-AVAIL-LABF
      BUFA(AVAIL)=BUFA(IHEAD)
      BUFB(AVAIL)=BUFB(IHEAD)
      IBUFC(AVAIL)=0
      AVAIL=AVAILN
      GO TO 70
4753  IF(SWTCH1.EQ.SWTCH2) GO TO 477
      IF(SWTCH1.NE.0) GO TO 476
C     HERE IF P1 IS HEAD , P2IS TAIL
      IBUFC(POINT2)=POINT1
      IBUFB(IXADL2)=IBUFB(IXADL1)
      IREMOV=IXADL1
      GO TO 65
C     HERE IF P1 IS TAIL, P2 IS HEAD
476   IBUFC(POINT1)=POINT2
      IBUFB(IXADL1)=IBUFB(IXADL2)
      IREMOV=IXADL2
      GO TO 65
C      HERE MUST CONNECT ARCS HEAD TO HEAD OR TAIL TO TAIL
477   IF(SWTCH1.EQ.0) GO TO 478
C     HERE IF P1 AND P2 ARE TAILS
      IBUFB(IXADL2)=IBUFA(IXADL1)
      IBUFC(POINT2)=POINT1
C      UNPACK IBUFA(IXADL1)
      SAVE0=ABS(IBUFA(IXADL1))
      SAVE1=SAVE0-LABF
      IF(SAVE1.GT.0) SAVE0=SAVE1
      SIGN1=IBUFC(SAVE0)
      SAVE1=IABS(SIGN1)
      IBUFC(SAVE0)=0
      GO TO 479
C     HERE IF P1 AND P2 ARE HEADS
478   IBUFA(IXADL2)=IBUFB(IXADL1)
      SAVE0=POINT1
      SIGN1=IBUFC(SAVE0)
      SAVE1=IABS(SIGN1)
      IBUFC(SAVE0)=POINT2
C      REVERSE THE NO. 1 LIST
479   SIGN2=IBUFC(SAVE1)
      SAVE2=IABS(SIGN2)
      IBUFC(SAVE1)=ISIGN(SAVE0,SIGN1)
      IF(SAVE2.EQ.0) GO TO 4791
      SAVE0=SAVE1
      SAVE1=SAVE2
      SIGN1=SIGN2
      GO TO 479
4791  IREMOV=IXADL1
      GO TO 65
C     ************************************************************
C     SEARCH FOR POINT ON APL (ARC POINT LIST)
49    IXADL=IBUFB(ICNTR)
      XNEW=X(POINT+1)
```

```
      YNEW=Y(POINT+1)
50    IF(IXADL.EQ.0) GO TO 54
      IHEAD=IBUFA(IXADL)
      ITAIL=IBUFB(IXADL)
      POINTR=IHEAD
      SWITCH=0
      GO TO 52
51    POINTR=ITAIL
      SWITCH=1
C     EXAMINE ONLY ACTIVE ENDS OF ARCS
52    IF(POINTR.LE.0) GO TO 53
      XOLD=BUFA(POINTR)
      YOLD=BUFB(POINTR)
C     COMPARE POINTS (XNEW,YNEW) AND (XOLD,YOLD)
      IF((XNEW.NE.XOLD).OR.(YNEW.NE.YOLD)) GO TO 53
      IF(POINT.EQ.0) GO TO 525
      IF(POINT1.EQ.POINTR) GO TO 53
525   KFOUND=1
      GO TO 55
53    IF(SWITCH.EQ.0) GO TO 51
      IXADL=IBUFC(IXADL)
      GO TO 50
54    KFOUND=0
C     ***************************************************************
C     SEARCH FAILED
      IF(POINT.NE.0) GO TO 545
      MODE(1)=-1
      LAB(1)=LABF
      GO TO 38
545   MODE(2)=-1
      LAB(2)=LABF
      KEY=4
      SWITCH=SWTCH1
      POINTR=POINT1
      IXADL=IXADL1
      GO TO 435
55    CONTINUE
      GO TO KRTN,(435,445,475)
C     ***************************************************
C     HERE IF KEY=1,3,7,9
C     (I,I),(I,A),(A,I),(A,A)
C     HERE TO START NEW ARC
C      FIRST OBTAIN 3 TRIPLE WORDS FROM AVAILABLE LIST
60    IXADL=AVAIL
      IHEAD=IBUFC(IXADL)
      ITAIL=IBUFC(IHEAD)
      AVAIL=IBUFC(ITAIL)
C      THEN CONSTRUCT NEW ARC
      BUFA(IHEAD)=X(1)
      BUFB(IHEAD)=Y(1)
      IBUFC(IHEAD)=ITAIL
      BUFA(ITAIL)=X(2)
```

```
      BUFB(ITAIL)=Y(2)
      IBUFC(ITAIL)=0
      IBUFA(IXADL)=ISIGN(IHEAD,MODE(1))-LAB(1)
      IBUFB(IXADL)=ISIGN(ITAIL,MODE(2))-LAB(2)
      IBUFC(IXADL)=IBUFB(ICNTR)
      IBUFB(ICNTR)=IXADL
      GO TO 70
C     ********************************************************
C     HERE TO RESTORE A WORD TO AVAILABLE SPACE
65    CONTINUE
C     FIRST, EXAMINE CVL LIST
      IXADL=IBUFB(ICNTR)
      IF(IXADL.NE.IREMOV) GO TO 653
      IBUFB(ICNTR)=IBUFC(IREMOV)
      GO TO 655
652   IXADL=IBUFC(IXADL)
653   IF(IBUFC(IXADL).NE.IREMOV) GO TO 652
      IBUFC(IXADL)=IBUFC(IREMOV)
655   IBUFC(IREMOV)=AVAIL
      IBUFA(IREMOV)=IBUFA(AVAIL)+1
      AVAIL=IREMOV
C     ********************************************************
70    CONTINUE
      IF(IS.LT.5) GO TO 83
      DO 71 I=1,2
      MODE(I)=MODE(I+2)
      X(I)=X(I+2)
71    Y(I)=Y(I+2)
      IS=3
      GO TO 37
C     ********************************************************
C     HERE AFTER PROCESSING ONE SQUARE FOR A SINGLE CONTUR
83    CONTINUE
      ICNTR=ICNTR+1
      IF(ICNTR.LE.NF) GO TO 19
831   CONTINUE
      XCOOR(1)=XCOOR(2)
      XCOOR(2)=XCOOR(2)+STEPX
      JCOL=JCOL+1
      RULE(1)=0
      IF(JCOL-NX1) 149,833,84
833   RULE(3)=-1
      GO TO 149
C     ********************************************************
C     HERE AFTER ALL COLUMNS OF MATRIX HAVE BEEN PROCESSED
C     MOVE Z-VALUES DOWN 2 POSITIONS AND INCREMENT YCOOR(1),
C     YCOOR(2), FOR NEXT ROW
84    CONTINUE
      GO TO LINK5,(845,855)
845   CONTINUE
      DO 85 K=3,NZMAX
      KBACK=3+NZMAX-K
```

```
  469.      85    Z(KBACK)=Z(KBACK-2)
  470.      855   CONTINUE
  471.            YCOOR(1)=YCOOR(2)
  472.            YCOOR(2)=YCOOR(2)+STEPY
  473.            XCOOR(1)=XMIN
  474.            XCOOR(2)=XMIN+STEPX
  475.            IROW=IROW+1
  476.            RULE(1)=-1
  477.            RULE(2)=0
  478.            RULE(3)=+1
  479.            IF(IROW-NY1) 118,853,86
  480.      853   RULE(4)=-1
  481.            GO TO 118
  482.      C     ********************************************************
  483.      C     ********************************************************
  484.      C     HERE WHEN FINISHED WITH ALL CONTURS FOR ENTIRE MAT.
  485.      86    CONTINUE
  486.            KFLAG2=3
  487.            KFLAG1=1
  488.            RETURN
  489.            END
```

HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS	
GENERATED CODE:	925	0039D	
CONSTANTS:	2	00002	
LOCAL VARIABLES:	90	0005A	
TEMPS:	2	00002	
TOTAL PROGRAM:	1019	003FB	(PLUS LABELED COMMON)

78EXT. FORTRAN IV, VERSION F02

EXT. FORTRAN IV, VERSION F02

PTIME

```
      1.          SUBROUTINE PTIME
      2.          DIMENSION M1A(64),NBT(4),MF(10),ID(4)
      3.          COMMON /HEAD/ INPBUF(512)
      4.          COMMON /IDAT/DT(4),NZFT,NENH,NEXP,NPOL,XPOS,YPOS
      5.          INTEGER DT
      6.          DATA  NBT/7,7,6,10/
      7.          DATA  MF /1,2,4,8,10,20,40,80,100,200/
      8.          DATA  ID /27F,27F,23F,33FF/
      9.          DATA KK/4ZFFFF/
     10.          DATA  NNN /1/
     11.          DATA KZF/4Z1FFF/
     12.          DATA KEH/2ZFF/
     13.          DATA KM1A/8ZFFFF0000/
     14.          DATA KEXP/1Z7/
     15.          IWD =1
     16.          DO 50 J1=1,64
     17.          M1A(J1) =INPBUF(IWD)
     18.          IWD =IWD +1
     19.     50   CONTINUE
     20.          IDAT=ISL(M1A(1),-16)+IAND(ISL(M1A(1),16),KM1A)
     21.          DO 252 I1=1,4
     22.          IF (I1 .EQ. 1) NBB =0
     23.          IF (I1 .GT. 1) NBB = NBT(I1-1) +NBB
     24.          DT(I1)      = IAND(ID(I1),ISL(IDAT,-NBB))
     25.      252 CONTINUE
     26.          NZFT=IAND(ISL(M1A(2),-16),KZF)
     27.          NENH=IAND(ISL(M1A(2),-8),KEH)
     28.          NEXP=IAND(ISL(M1A(3),-25),KEXP)
     29.          NPOL=IAND(ISL(M1A(3),-19),7)
     30.          DO 25 I1=1,8
     31.       25 CONTINUE
     32.          IF(NEXP .EQ. 2) NEXP=60
     33.          IF(NEXP .EQ. 3) NEXP=30
     34.          IF(NEXP .EQ. 4) NEXP=15
     35.          IF(NEXP .EQ. 5) NEXP=8
     36.          IXX =ISL(M1A(3),-16)+IAND(ISL(M1A(3),16),KM1A)
     37.          ISNGX=IAND(ISL(IXX,-13),1)
     38.          XPOS=0
     39.          TXPOS=0
     40.          MX=14
     41.     C    DO 333 II=1,3
     42.          II=3
     43.      333 CONTINUE
     44.          NN=4
     45.      353 CONTINUE
     46.          TXPOS=TXPOS+IAND(ISL(IXX,-MX),1)*(2**(NN-1))
     47.          NN = NN -1
     48.          MX=MX+1
     49.          IF(NN .GE. 1) GO TO 353
     50.          XPOS=XPOS+TXPOS*(10**(II-1))
     51.          TXPOS=0
     52.          IF(II .LE. 1) GO TO 363
```

```
53.           II=II-1
54.           GO TO 333
55.       363 CONTINUE
56.           IF(ISNGX .EQ. 0) XPOS=-XPOS
57.           ISNGY=IAND(ISL(IXX,-MX),1)
58.           YPOS=0
59.           TYPOS=0
60.           MX=MX+1
61.           NN=4
62.       454 CONTINUE
63.           TYPOS=TYPOS+(IAND(ISL(IXX,-MX),1))*2**(NN-1)
64.           NN=NN-1
65.           MX=MX+1
66.           IF(NN .GE. 1) GO TO 454
67.           YPOS=YPOS+TYPOS
68.           YPOS=YPOS*(10**2)
69.           TYPOS=0
70.           MY=23
71.     C     DO 444 II=1,2
72.           II=2
73.       444 CONTINUE
74.           NN=4
75.       464 CONTINUE
76.           IYY=ISL(M1A(4),-16)+IAND(ISL(M1A(4),16),KM1A)
77.           TYPOS=TYPOS + (IAND(ISL(IYY,-MY),1))*2**(NN-1)
78.           NN=NN - 1
79.           MY = MY + 1
80.           IF (NN .GE. 1) GO TO 464
81.           YPOS=YPOS + TYPOS*(10**(II-1))
82.           TYPOS=0
83.     C 444 CONTINUE
84.           IF(II .LE. 1) GO TO 474
85.           II=II-1
86.           GO TO 444
87.       474 CONTINUE
88.           IF(ISNGY .EQ. 0) YPOS=-YPOS
89.           XPOS=XPOS/100.
90.           YPOS=YPOS/100.
91.           RETURN
92.           END
```

28 HIGHEST ERROR SEVERITY: 0 (NO ERRORS)

	DEC WORDS	HEX WORDS	
GENERATED CODE:	267	00108	
CONSTANTS:	5	00005	
LOCAL VARIABLES:	104	00068	
TEMPS:	4	00004	
TOTAL PROGRAM:	380	0017C	(PLUS LABELED COMMON)

APPENDIX C

COMMAND FILES FOR KZTIME, KZLONGT, AND KZTRANSV

```
C C:ICZTIME
!ECHO
!SET F:13 FTGTEST
!SET F:10 RE
!C B:ICZTIME,B:ICZPTIME,B:07 OVER B:TEMP
!LYNX B:TEMP OVER L:ICZTIME(SL,F) (R)

!C X:ICZTIME
!ECHO
!ERROR
!SET F:13 FTGTEST
!SET F:10 RE
!L:ICZTIME.
```

PAGE 84 INTENTIONALLY BLANK

```
C C:KZLONGT
!ECHO
!SET F:13 FTSTEST
!SET F:2 /1A;OUTIN;SAVE;KEYED;DIRECT
!SET F:3 /1B;OUTIN;SAVE;KEYED;DIRECT
!SET F:5 /FHL;OUTIN;SAVE;KEYED;DIRECT
!SET F:6 /HL;OUTIN;SAVE;KEYED;DIRECT
!SET F:7 /HI;OUTIN;SAVE;KEYED;DIRECT
!SET F:12 /A;OUTIN;SAVE;KEYED;DIRECT
!SET F:9 /B;OUTIN;SAVE;KEYED;DIRECT
!SET F:11 /C;OUTIN;SAVE;KEYED;DIRECT
!SET F:14 LP
!SET F:20 /BNDS;OUT;SAVE
!SET F:22 /BNDS
!SET F:10 ME
!SET F:101 ME
!C B:KZLONGT,B:GT,B:KZCLOSE,B:KZFILTER,B:KZSMPLOT OVER B:TEMP1
!C B:KZCPLT1,B:KZCPLT2,B:KZSINPUT,B:KZPPLOT,B:KZCSCAN OVER B:TEMP2
!C B:KZPTIME,B:NCNUT,B:KZCONTUR,B:KZCTRIM    OVER B:TEMP3
!LYNX B:TEMP1,B:TEMP2,B:TEMP3 OVER L:KZLONGT(SL,F) (M) ;LXLIB.MUGRAF

!C X:KZLONGT
!ECHO
!ERROR
!SET F:13 FTSTEST
!SET F:2 /1A;OUTIN;SAVE;KEYED;DIRECT
!SET F:3 /1B;OUTIN;SAVE;KEYED;DIRECT
!SET F:5 /FHL;OUTIN;SAVE;KEYED;DIRECT
!SET F:6 /HL;OUTIN;SAVE;KEYED;DIRECT
!SET F:7 /HI;OUTIN;SAVE;KEYED;DIRECT
!SET F:12 /A;OUTIN;SAVE;KEYED;DIRECT
!SET F:9 /B;OUTIN;SAVE;KEYED;DIRECT
!SET F:11 /C;OUTIN;SAVE;KEYED;DIRECT
!SET F:14 LP
!SET F:20 /BNDS;OUT;SAVE
!SET F:22 /BNDS
!SET F:10 ME
!SET F:101 ME
!L:KZLONGT.

!
```

```
C C:KZTRANSU
!ECHO
!SET F:13 FT$TEST
!SET F:2 /2A;OUTIN;SAVE;KEYED;DIRECT
!SET F:3 /2B;OUTIN;SAVE;KEYED;DIRECT
!SET F:4 /3A;OUTIN;SAVE;KEYED;DIRECT
!SET F:5 /3B;OUTIN;SAVE;KEYED;DIRECT
!SET F:6 /HT;OUTIN;SAVE;KEYED;DIRECT
!SET F:7 /HIU;OUTIN;SAVE;KEYED;DIRECT
!SET F:8 /HIR;OUTIN;SAVE;KEYED;DIRECT
!SET F:9 /B;OUTIN;SAVE;KEYED;DIRECT
!SET F:11 /C;OUTIN;SAVE;KEYED;DIRECT
!SET F:12 /A;OUTIN;SAVE;KEYED;DIRECT
!SET F:20 /BNDS;OUT;SAVE
!SET F:22 /BNDS
!SET F:10 ME
!SET F:101 ME
!C B:KZTRANSU,B:GT,B:KZCLOSE,B:KZAZPLT,B:KZSMPLOT,B:KZDRIV1 OVER B:TEMP
!C B:KZCPLT1,B:KZCPLT2,B:KZSDNPUT,B:KZPPLOT,B:KZCSCAN OVER B:TEMP1
!C B:KZPTIME,B:NCNUT,B:KZCONTUR,B:KZCTRIM OVER B:TEMP2
!LYNX B:TEMP,B:TEMP1,B:TEMP2 OVER L:KZTRANSU(SL,F)(M);LXLIB.MUGRAF

!C X:KZTRANSU
!ECHO
!ERROR
!SET F:13 FT$TEST
!SET F:2 /2A;OUTIN;SAVE;KEYED;DIRECT
!SET F:3 /2B;OUTIN;SAVE;KEYED;DIRECT
!SET F:4 /3A;OUTIN;SAVE;KEYED;DIRECT
!SET F:5 /3B;OUTIN;SAVE;KEYED;DIRECT
!SET F:6 /HT;OUTIN;SAVE;KEYED;DIRECT
!SET F:7 /HIU;OUTIN;SAVE;KEYED;DIRECT
!SET F:8 /HIR;OUTIN;SAVE;KEYED;DIRECT
!SET F:9 /B;OUTIN;SAVE;KEYED;DIRECT
!SET F:11/C;OUTIN;SAVE;KEYED;DIRECT
!SET F:12/A;OUTIN;SAVE;KEYED;DIRECT
!SET F:20 /BNDS;OUT;SAVE
!SET F:22 /BNDS
!SET F:10 ME
!SET F:101 ME
!L:KZTRANSU.

!
```

APPENDIX D
DESCRIPTIONS FOR PROGRAM SUBROUTINES

KZTIME: To read the 9TRK data tape and search for the beginning record of each file and also print out the housekeeping information for each file.

KZLONGT: The main program for longitudinal plot.
To read data file 1A1B from 9TRK data tape, compute contour data (1A - 1B)/(1A + 1B) and save it for future use.
To read information from terminal input and call subroutines to do contour plots.

KZTRANSV: The main program for transverse.
To read input data 2A2B, 3A3B, compute contour data, azimuth data and plot.

KZPTIME: To generate housekeeping information.

KZFILTER: To generate filter data.

KZDRIV 1: To choose saved data file for different cases, plot longitudinal contour and intensity plot and also display the label with the plots.

KZSMPLOT: To set up X, Y range, clear screen and call subroutines to plot save status words.

KZCONTUR: Generate contour plot points.

KZCPLT 1: Draw plotting frame.

KZCPLT 2: To decide starting or ending contour and call subroutine KZSDNPUT to plot.

KZSDNPUT: To plot contour.

KZPPLOT: Options 1. Delete all characters not in box.
2. Delete all characters.
3. Delete only low level characters.

KZCSCAN: Set up KGO
KGO = 1 New contour value
KGO = 2 New ARC
KGO = 3 New SUB-ARC After gap

KGO = 4 New point

KGO = 5 Finished

APPROVAL

LONGITUDINAL AND TRANSVERSE MAGNETIC FIELD PROGRAM PROCEDURE
AND DETAILED SPECIFICATION (FOR SIGMA 5)

By Caroline K. Wang

The information in this report has been reviewed for technical content. Review of any information concerning Department of Defense or nuclear energy activities or programs has been made by the MSFC Security Classification Officer. This report, in its entirety, has been determined to be unclassified.

ANTHONY C. DeLOACH
Chief, Solar Sciences Branch

CHARLES R. CHAPPELL
Chief, Solar-Terrestrial Division

CHARLES A. LUNDQUIST
Director, Space Sciences Laboratory

☆ U.S. GOVERNMENT PRINTING OFFICE 1980—640-247/80 REGION NO.4