8.0-10139
CR 160592

National Aeronautics and Space Administration

"Made available under NASA sponsorship in the interest of early and wide dissemination of Earth Resources Survey Program information and without liability for any use made thereof."

JSC-13821

Lyndon B. Johnson Space Center
Houston, Texas 77058

EARTH OBSERVATIONS DIVISION

SPACE AND LIFE SCIENCES DIRECTORATE

EARTH OBSERVATIONS DIVISION VERSION OF THE LABORATORY FOR APPLICATIONS OF REMOTE SENSING SYSTEM (EOD-LARSYS) USER GUIDE FOR THE IBM 370/148

VOLUME IV – PROGRAM LISTINGS

Job Order 71-475

Prepared By

Lockheed Electronics Company, Inc.
Systems and Services Division
Houston, Texas

Contract NAS 9-15800

(E80-10139) EARTH OBSERVATIONS DIVISION VERISON OF THE LABORATORY FOR APPLICATIONS OF REMOTE SENSING SYSTEM (EOD-LARSYS) USER GUIDE FOR THE IBM 370/148. VOLUME 4: PROGRAM LISTINGS (Lockheed Electronics Co.) G3/43

N80-26720

Unclas
00139

January 1979

LEC-12565

EARTH OBSERVATIONS DIVISION VERSION OF THE LABORATORY FOR APPLICATIONS OF REMOTE SENSING SYSTEM (EOD-LARSYS) USER GUIDE FOR THE IBM 370/148

VOLUME IV – PROGRAM LISTINGS

Job Order 71-475

PREPARED BY

M. L. Burnell
Technical Publications Department

and

P. J. Aucoin
Earth Observations Data Products Department

APPROVED BY

J. I. Morrow, Supervisor
Scientific Applications Section

Prepared By

Lockheed Electronics Company, Inc.

For

Earth Observations Division

Space and Life Sciences Directorate

NATIONAL AERONATUICS AND SPACE ADMINISTRATION
LYNDON B. JOHNSON SPACE CENTER
HOUSTON, TEXAS

January 1979

LEC-12565

CONTENTS

Section		Page
1.	SCOPE	1-1
2.	APPLICABLE DOCUMENTS	2-1
3.	MONTOR	3-1
4.	MSCAN	4-1
5.	COMMON BLOCKS AND BLOCK DATA	5-1
6.	HIST PROCESSOR	6-1
7.	GRAYMAP PROCESSOR	7-1
8.	STAT PROCESSOR	8-1
9.	ISOCLS PROCESSOR	9-1
10.	SELECT PROCESSOR	10-1
11.	CLASSIFY PROCESSOR	11-1
12.	DISPLAY PROCESSOR	12-1
13.	DATA-TR PROCESSOR	13-1
14.	TRSTAT PROCESSOR	14-1
15.	NDHIST PROCESSOR	15-1
16.	SCTRPL PROCESSOR	16-1
17.	DOTDATA PROCESSOR	17-1
18.	LABEL PROCESSOR	18-1
19.	UTILITY SUBPROGRAMS	19-1
20.	DAMRG PROCESSOR	20-1
21.	GTDDM PROCESSOR	21-1
22.	GTTCN PROCESSOR	22-1
23.	TESTSP PROCESSOR	23-1

PRECEDING PAGE BLANK NOT FILMED

1. SCOPE

This document is one of a four-volume series entitled "Earth Observations Division Version of the Laboratory for Applications of Remote Sensing System (EOD-LARSYS) User Guide for the IBM 370/148" (section 2). This volume IV contains a listing for each subprogram within the EOD-LARSYS. It is modeled after the As-Built Documentation (Volume III), inasmuch as the listings appear in the same order as the subprograms are documented in volume III. Table 1-1 of volume III lists all of the EOD-LARSYS subprograms in alphabetical order, along with the processor to which each belongs. The processors, by section, are as follows:

Section	Processor
6	One-Dimensional Histogram (HIST)
7	GRAYMAP
8	Statistics (STAT)
9	Iterative Self-Organizing Clustering System (ISOCLS)
10	Feature Selection (SELECT)
11	Classification (CLASSIFY)
12	Performance Display (DISPLAY)
13	Data-Transformation (DATATR)
14	Statistics Transformation (TRSTAT)
15	N-Dimensional Histogram (NDHIST)
16	Scatter Plot (SCTRPL)
17	Dot Data (DOTDATA)
18	Automatic Cluster Labeling (LABEL)

Within each of the above sections, the processor driver routine is listed first, followed by the subprogram listings in alphabetical order (the same order as they were documented in volume III). Utility subprograms are listed in section 19. In addition, this document contains subprogram listings for the following new processors:

Section	Processor
20	Data Merge (DAMRG)
21	Ground Truth Data Tape Dump (GTDDM)
22	Ground Truth Tape Conversion (GTTCN)
23	Iterative Self-Organizing Clustering System Using Packed Pixel Storage (TESTSP)

The listing for the EOD-LARSYS monitor routine, MONTOR, is given in section 3, along with a listing for an optional monitor routine, MONPAC. The MONPAC routine was created for use with the TESTSP processor, which clusters pixel values and stores them in packed form on disk storage. It differs from MONTOR in that it stores pixels in packed form (one sample per byte) rather than in floating point (one sample value every four bytes), as is done by the ISOCLS processor. The MONPAC routine may be used with other processors.

The listing for MSCAN, MONTOR's supervisory routine, is given in section 4, and common block listings are given in section 5.

2. APPLICABLE DOCUMENTS

1. Stewart, J.; et al.: EOD-LARSYS User Guide for the IBM 370/148 - vol. I, System Overview. JSC 13821, LEC-12563, NASA/JSC (Houston), Aug. 1978.

2. Stewart, J.; et al.: EOD-LARSYS User Guide for the IBM 370/148 - vol. II, User's Reference Manual. JSC-13821, LEC-12564, NASA/JSC (Houston), Dec. 1978.

3. Burnell, M. L.; et al.: EOD-LARSYS User Guide for the IBM 370/148 - vol. III, As-Built Documentation. JSC-13821, LEC-12565, NASA/JSC (Houston), to be published.

3. MONTOR

FILE: MONTOR FORTRAN A

```
C     //MONTOR                                                                MON00010
C                                                                             MON00020
CCCCC|-----------------------------------------------------------------------|MON00030
CCCCC|                                                                       |MON00040
CCCCC|     CALL..    SYSTEM MONITOR  (// EXEC  LARSYSAA )                    |MON00050
CCCCC|                                                                       |MON00060
CCCCC|     PURPOSE.. MONITORS THE VARIOUS SYSTEM SUPERVISORS                 |MON00070
CCCCC|                                                                       |MON00080
CCCCC|               ROUTINES MSCAN   CLSFY  DSPLAY  STAT                    MON00090
CCCCC|                         SELECT  HIST   GRAYMP  DATATR                 MON00100
CCCCC|                         ISOCLS  TRSTAT  NOHIST  SCTRPL                MON00110
CCCCC|                         DOTODATA    LABEL    EQUPRB  MULBAY           MON00120
CCCCC|                         GTTCN   DAMMG                                 MON00130
CCCCC|                         AMOEBA CLASY TESTSP GTDOM EXIT                MON00140
CCCCC|     RETURNS.. NONE                                                    |MON00150
CCCCC|                                                                       |MON00160
CCCCC|-----------------------------------------------------------------------|MON00170
C                                                                             MON00180
C                                                                             MON00190
      IMPLICIT INTEGER(A-H,O-Z)                                               MON00200
      COMMON ARRAY                                                            MON00210
      DIMENSION ARRAY(10600)                                                  MON00220
C*                                                                            MON00230
C**   'ARRAY' IS A BLOCK OF STORAGE PASSED TO EACH PROCESSOR FOR THE          MON00240
C**   VARIABLE DIMENSIONING OF OTHER ARRAYS. THE ARRAY IS NEVER USED          MON00250
C**   TO PASS INFORMATION  FROM ONE PROCESSOR TO ANOTHER.                     MON00260
C*                                                                            MON00270
      DATA TOP/10600/                                                         MON00280
C     INCLUDE COMMK6,LIST                                                     MON00290
C                                                                             MON00300
C                                                                             MON00310
C     INCLUDE COMMT6,LIST                                                     MON00320
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,               MON00330
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,         MON00340
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                      MON00350
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                          MON00360
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                    MON00370
     *  CPOUNT,PRTUNT,RANDIO                                                  MON00380
C*                                                                            MON00390
C**   GLOBAL COMMON IS USED IN EVERY PROCESSOR. IT IS ALWAYS IN CORE.         MON00400
C**   ALL PARAMETERS ARE INITIALIZED IN THE MONITOR ROUTINE OR BLKCOM         MON00410
C**   EXCEPT AS NOTED BELOW                                                   MON00420
C**   DEFINITIONS                                                             MON00430
C**       HEAD   - STANDARD HEADING PRINTED ON MOST OUTPUT PAGES.             MON00440
C**       MAPTAP - FORTRAN UNIT NUMBER ON WHICH THE MAPTAP FILE IS            MON00450
C**                 WRITTEN (=2)                                              MON00460
C**       DATAPE - UNIT NO. FOR THE IMAGE DATA TAPE (=3)                      MON00470
C**       SAVTAP - UNIT NO. ON WHICH THE STATISTICS FILE IS WRITTEN (=1)      MON00480
C**       BMFILE - UNIT NO. ON WHICH THE B-MATRIX FILE IS WRITTEN (=10)       MON00490
C**       BMKEY - TRIGGER INDICATING THAT THE B-MATRIX FILE HAS BEEN          MON00500
C**                 WRITTEN. CAN BE SET IN SELECT CLASSIFY OR DATA-TP.        MON00510
C**       HISFIL - UNIT NO. ON WHICH THE HISTOGRAM FILE IS WRITTEN (=13)      MON00520
C**       HISKEY - TRIGGER INDICATING THE HISTOGRAM FILE HAS BEEN             MON00530
C**                 WRITTEN. SET IN HIST PROCESSOR.                           MON00540
C**       TRFORM - UNIT NO. ON WHICH THE TRANSFORMED IMAGE IS WRITTEN BY      MON00550
C**                 THE DATA-TRANSFORMATION PROCESSOR. (=14)                  MON00560
C**       ERIPTP - UNIT NO. ON WHICH THE ISOCLS PROCESSOR WRITES              MON00570
C**                 CLUSTER STATISTICS FOR THE ERIPS SYSTEM. (=15)            MON00580
C**       ERPKEY - TRIGGER INDICATING THAT THE ERIPS INTERFACE TAPE           MON00590
C**                 HAS BEEN WRITTEN.                                         MON00600
C**       MAPUNT - UNIT NO. ON WHICH THE ISOCLS OR DISPLAY PROCESSOR          MON00610
C**                 WRITES THE CLUSTERED OR CLASSIFIED DATA                   MON00620
C**                TO BE DISPLAYED ON THE PMIS DAS                            MON00630
C**       NOFILE - NO. OF FILES WRITTEN ON UNIT 16 (MAP OUTPUT TAPE)          MON00640
C**                 BY DISPLAY AND/OR ISOCLS                                  MON00650
C**                 SET EITHER IN ISOCLS OR DISPLAY.                          MON00660
C*        DRUMAD - BEGINNING ADDRESS FOR THE RANDOM ACCESS HIGH SPEED         MON00670
C**   REAL TIME                                                               MON00680
C**                 DRUM FILE. THIS FILE IS USED AS A SCRATCH FILE IN         MON00690
C**                 SEVERAL PROCESSORS. REFERENCES TO SYSTEM ROUTINES         MON00700
C**                 'READ' AND 'WRITE' ACCESS THIS FILE.                      MON00710
C**       DRMWDS - NO. OF WORDS AVAILABLE ON THE RANDOM ACCESS FILE.          MON00720
C**       PAGSIZ - NO. OF LINES AVAILABLE FOR PRINTING ON A PAGE.             MON00730
C**       DATFIL - NO. OF E-O-F'S TO BE READ OVER BY TAPERD ROUTINE IN        MON00740
C**                 ORDER TO POSITION THE DATA TAPE TO DESIRED FILE           MON00750
C**       STAFIL - NO. OF E-O-F'S TO SKIP OVER TO POSITION STAT FILE(1)       MON00760
C**         ASAV - UNIT NO. ON WHICH TRSTAT WRITES THE TRANSFORMED            MON00770
C**                 STATS                                                     MON00780
C**       ASAVFL - NO. OF E-O-F'S TI SUIP OVER TO POSITION TRANSFORMED        MON00790
```

```
FILE: MONTOR   FORTRAN  A

C*                    STATS                                                  MON00800
C*         DOTUNT - UNIT NO. ON WHICH DOT DATA FILE (DOTFIL) IS WRITTEN      MON00810
C*         DOTFIL - NO. OF E-O-F'S TO SKIP OVER TO POSITION DOTFIL FILE      MON00820
C*         NCHPAS - NO. OF CHANNELS PER PASS                                 MON00830
C*         TRNSFL - NO. OF E-O-F'S TO SKIP OVER FOR TRFORM FILE              MON00840
C*         AMTXFL - NO. OF E-O-F'S TO SKIP OVER FOR AMFIL FILE               MON00850
C*         HISTFL - NO. OF E-O-F'S TO SKIP OVER FOR HISFIL FILE              MON00860
C*         PUNCH  - UNIT NO. FOR CARD PUNCH FILE                             MON00870
C*         CRDUNT - UNIT NO. FOR CARD READER                                 MON00880
C*         RANDIO  - SCRATCH UNIT FOR RREAD AND RWRITE ROUTINES              MON00890
C*END                                                                       MON00900
      DRUG=-1                                                              MON00910
C*                                                                         MON00920
C*    SYSTEM ROUTINE RINIT ASSIGNS THE RANDOM ACCESS DRUM FILE.            MON00930
C*                                                                         MON00940
C*          -DRUMAD-- IS THE ADDRESS TO BEGIN WRITING                      MON00950
C*          -DRMWDS- IS THE NO. OF WORDS AVAILABLE ON THE DRUM FILE.       MON00960
C*                                                                         MON00970
C*    THE FOLLOWING PROCESSORS USE THE RANDOM ACCESS DRUM FILE FOR SCRATCH MON00980
C*          -ISOCLS-                                                       MON00990
C*          -DISPLY-                                                       MON01000
C*          -SELECT-                                                       MON01010
C*                                                                         MON01020
C*          -GRAYMP-                                                       MON01030
C*          -SIGEXT-.                                                      MON01040
C                                                                          MON01050
      DEFINE FILE 22(2100,200,U,ID)                                        MON01060
      DRUMAD=1                                                             MON01070
      DRMWDS=420000                                                        MON01080
      WRITE(22'1)DRUMAD                                                    MON01090
   10 CONTINUE                                                             MON01100
      TIME = 0.                                                            MON01110
      CALL CLOCK(TIME)                                                     MON01120
      CALL MSCAN(JGO,DRUG)                                                 MON01130
      GO TO (20,40,60,80,100,120,140,160,175,180,200,220,240,260,          MON01140
     * 280,290,300,310,320,330,340,350,360),JGO                            MON01150
C                                                                          MON01160
   20 CONTINUE                                                             MON01170
      CALL STAT(ARRAY,TOP)                                                 MON01180
      CALL CLOCK(TIME)                                                     MON01190
      WRITE (6,30)TIME                                                     MON01200
   30 FORMAT(' TIME FOR STAT',F10.3)                                       MON01210
      GO TO 10                                                             MON01220
C                                                                          MON01230
C                                                                          MON01240
   40 CONTINUE                                                             MON01250
      CALL CLSFY(ARRAY,TOP)                                                MON01260
      CALL CLOCK(TIME)                                                     MON01270
      WRITE (6,50)TIME                                                     MON01280
   50 FORMAT(' TIME FOR CLASSIFY',F10.3 )                                  MON01290
      GO TO 10                                                             MON01300
C                                                                          MON01310
C                                                                          MON01320
   60 CONTINUE                                                             MON01330
      CALL DSPLAY(ARRAY,TOP)                                               MON01340
      CALL CLOCK(TIME)                                                     MON01350
      WRITE (6,70)TIME                                                     MON01360
   70 FORMAT(' TIME FOR DISPLAY',F10.3)                                    MON01370
      GO TO 10                                                             MON01380
C                                                                          MON01390
C                                                                          MON01400
   80 CONTINUE                                                             MON01410
      CALL SELECT(ARRAY,TOP)                                               MON01420
      CALL CLOCK(TIME)                                                     MON01430
      WRITE (6,90)TIME                                                     MON01440
   90 FORMAT(' TIME FOR SELECT',F10.3)                                     MON01450
      GO TO 10                                                             MON01460
C                                                                          MON01470
C                                                                          MON01480
  100 CONTINUE                                                             MON01490
      CALL HIST(ARRAY,TOP)                                                 MON01500
      CALL CLOCK(TIME)                                                     MON01510
      WRITE(6,110)TIME                                                     MON01520
  110 FORMAT(' TIME FOR HISTOGRAM',F10.3)                                  MON01530
      GO TO 10                                                             MON01540
C                                                                          MON01550
C     GO HERE FOR ISOCLS                                                   MON01560
C                                                                          MON01570
  120 CONTINUE                                                             MON01580
```

```
FILE: MONTOR   FORTHAN  A

      CALL ISOCLS(ARRAY,TOP)                                          MON01590
      CALL CLOCK(TIME)                                                MON01600
      WRITE(6,130) TIME                                               MON01610
  130 FORMAT(' TIME FOR ISOCLS',F10.3)                                MON01620
      GO TO 10                                                        MON01630
C                                                                     MON01640
C     GO HERE FOR GRAYMAP                                             MON01650
  140 CONTINUE                                                        MON01660
      CALL GRAYMP(ARRAY,TOP)                                          MON01670
      CALL CLOCK(TIME)                                                MON01680
      WRITE(6,150) TIME                                               MON01690
  150 FORMAT(' TIME FOR GRAYMAP',F10.3)                               MON01700
      GO TO 10                                                        MON01710
C                                                                     MON01720
C     GO HERE FOR DATA-TRANSFORMATION                                 MON01730
C                                                                     MON01740
  160 CONTINUE                                                        MON01750
      CALL DATATR(ARRAY,TOP)                                          MON01760
      CALL CLOCK(TIME)                                                MON01770
      WRITE(6,170) TIME                                               MON01780
  170 FORMAT(//' TIME FOR DATA-TRANSFORMATION',F10.3)                 MON01790
      GO TO 10                                                        MON01800
C*                                                                    MON01810
C*    GO HERE FOR SIGEXT MODULE                                       MON01820
C*                                                                    MON01830
  175 CONTINUE                                                        MON01840
C  *** SIGEXT                                                        MON01850
      CALL CLOCK(TIME)                                                MON01860
      WRITE(6,176)TIME                                                MON01870
  176 FORMAT(//' TIME FOR SIGEXT ',F10.3, 'MINUTES' )                 MON01880
      GO TO 10                                                        MON01890
C                                                                     MON01900
C     GO HERE FOR TRSTAT                                              MON01910
C                                                                     MON01920
  180 CONTINUE                                                        MON01930
      CALL TRSTAT(ARRAY,TOP)                                          MON01940
      CALL CLOCK(TIME)                                                MON01950
      WRITE(6,190)TIME                                                MON01960
  190 FORMAT(//' TIME FOR TRSTAT',F10.3,' MINUTES')                   MON01970
      GO TO 10                                                        MON01980
C                                                                     MON01990
C     GO HERE FOR NDHIST                                              MON02000
C                                                                     MON02010
  200 CONTINUE                                                        MON02020
      CALL NDHIST(ARRAY,TOP)                                          MON02030
      CALL CLOCK(TIME)                                                MON02040
      WRITE(6,210)TIME                                                MON02050
  210 FORMAT(//' TIME FOR NDHIST',F10.3,' MINUTES')                   MON02060
      GO TO 10                                                        MON02070
C                                                                     MON02080
C     GO HERE FOR SCTRPL                                              MON02090
C                                                                     MON02100
  220 CONTINUE                                                        MON02110
      CALL SCTRPL(ARRAY,TOP)                                          MON02120
      CALL CLOCK(TIME)                                                MON02130
      WRITE(6,230)TIME                                                MON02140
  230 FORMAT(//' TIME FOR SCTRPT',F10.3,' MINUTES')                   MON02150
      GO TO 10                                                        MON02160
C                                                                     MON02170
C                                                                     MON02180
C     GO HERE FOR DOTDATA                                             MON02190
C                                                                     MON02200
  240 CONTINUE                                                        MON02210
      CALL DOTDAT(ARRAY,TOP)                                          MON02220
      CALL CLOCK(TIME)                                                MON02230
      WRITE(6,250)TIME                                                MON02240
  250 FORMAT(//' TIME FOR DOTDATA',F10.3,' MINUTES')                  MON02250
      GO TO 10                                                        MON02260
C                                                                     MON02270
C     GO HERE FOR LABEL                                               MON02280
C                                                                     MON02290
  260 CONTINUE                                                        MON02300
      CALL LABEL(ARRAY,TOP)                                           MON02310
      CALL CLOCK(TIME)                                                MON02320
      WRITE(6,270)TIME                                                MON02330
  270 FORMAT(//' TIME FOR LABEL',F10.3,' MINUTES')                    MON02340
      GO TO 10                                                        MON02350
C                                                                     MON02360
C      GO HERE FOR EQUI-PROBABLE BLOCKS CLASSIFIER                    MON02370
```


ORIGINAL PAGE IS
OF POOR QUALITY

FILE: MONTOR FORTRAN A

```
C                                                                       MON02380
                                                                        MON02390
 280  CONTINUE                                                         MON02400
      CALL EQUPRB(ARRAY,TOP)                                            MON02410
      CALL CLOCK(TIME)                                                  MON02420
      WRITE(6,285) TIME                                                 MON02430
  285 FORMAT(//' TIME FOR EQUPRB',F10.3,'MINUTES')                      MON02440
      GO TO 10                                                          MON02450
C                                                                       MON02460
CCC   GO HERE FOR MULTI-TEMPORAL CLASSIFIER                             MON02470
C                                                                       MON02480
  290 CONTINUE                                                          MON02490
      CALL MULBAY(ARRAY,TOP)                                            MON02500
      CALL CLOCK(TIME)                                                  MON02510
      WRITE(6,295) TIME                                                 MON02520
  295 FORMAT(//' TIME FOR MULBAY',F10.3,'MINUTES')                      MON02530
      GO TO 10                                                          MON02540
C                                                                       MON02550
CCC   GO HERE FOR GROUND TRUTH TO MAPFIL                                MON02560
C                                                                       MON02570
  300 CONTINUE                                                          MON02580
      CALL GTTCN(ARRAY,TOP)                                             MON02590
      CALL CLOCK(TIME)                                                  MON02600
      WRITE(6,305) TIME                                                 MON02610
  305 FORMAT(//' TIME FOR GTTCN',F10.3,'MINUTES')                       MON02620
      GO TO 10                                                          MON02630
C                                                                       MON02640
CC    GO HERE FOTR IMAGE DATA MERGE                                     MON02650
  310  CONTINUE                                                         MON02660
      CALL DAMRG(ARRAY,TOP)                                             MON02670
      CALL CLOCK(TIME)                                                  MON02680
      WRITE(6,315) TIME                                                 MON02690
  315 FORMAT(//' TIME FOR DAMRG',F10.3,'MINUTES')                       MON02700
      GO TO 10                                                          MON02710
C                                                                       MON02720
CCC   GO HERE FOR AMOEBA                                                MON02730
C                                                                       MON02740
  320   CONTINUE                                                        MON02750
        CALL AMOEBA(ARRAY,TOP)                                          MON02760
        CALL CLOCK(TIME)                                                MON02770
        WRITE(6,325) TIME                                               MON02780
  325   FORMAT(//' TIME FOR AMOEBA',F10.3,'MINUTES')                    MON02790
        GO TO 10                                                        MON02800
C                                                                       MON02810
CCC   GO HERE FOR CLASY                                                 MON02820
C                                                                       MON02830
  330   CONTINUE                                                        MON02840
        CALL CLASY(ARRAY,TOP)                                           MON02850
        CALL CLOCK(TIME)                                                MON02860
        WRITE(6,335) TIME                                               MON02870
  335   FORMAT(//' TIME FOR CLASY',F10.3,' MINUTES')                    MON02880
        GO TO 10                                                        MON02890
C                                                                       MON02900
CCC   GO HERE FOR TESTSP                                                MON02910
C                                                                       MON02920
  340   CONTINUE                                                        MON02930
        CALL TESTSP(ARRAY,TOP)                                          MON02940
        CALL CLOCK(TIME)                                                MON02950
        WRITE(6,345) TIME                                               MON02960
  345   FORMAT(//' TIME FOR TESTSP',F10.3,' MINUTES')                   MON02970
        GO TO 10                                                        MON02980
C                                                                       MON02990
CCC   GO HERE FOR GROUND TRUTH DOT UNLOAD                               MON03000
C                                                                       MON03010
  350   CONTINUE                                                        MON03020
        CALL GTODM(ARRAY,TOP)                                           MON03030
        CALL CLOCK(TIME)                                                MON03040
        WRITE(6,355) TIME                                               MON03050
  355   FORMAT(//' TIME FOR GTODM',F10.3,' MINUTES')                    MON03060
        GO TO 10                                                        MON03070
C                                                                       MON03080
CCC   GO HERE TO EXIT                                                   MON03090
C                                                                       MON03100
  360 IF (NOFILE .GT. 0) REWIND MAPUNT                                  MON03110
      END
```

FILE: CLOCK FORTRAN A

```
      SUBROUTINE CLOCK(TIME)                                          CL000010
      INTEGER TIMER                                                   CL000020
      TIME = TIMER(0)                                                 CL000030
      TIME = (TIME)/60000.)-TIME                                      CL000040
      RETURN                                                          CL000050
      END                                                             CL000060
```

```
FILE: MONPAC   FORTRAN  A

C       //MONTOR                                                                MON00010
C                                                                               MON00020
C     ------------------------------------------------------------------------- MON00030
C     |                                                                       | MON00040
C     |   CALL..     SYSTEM MONITOR  (// EXEC  LARSYSAA )                     | MON00050
C     |                                                                       | MON00060
C     |   PURPOSE.. MONITORS THE VARIOUS SYSTEM SUPERVISORS                   | MON00070
C     |                                                                       | MON00080
C     |             ROUTINES MSCAN   CLSFY  DSPLAY  STAT                      | MON00090
C     |                      SELECT  HIST   GRAYMP  DATATP                    | MON00100
C     |                      ISOCLS  TWSTAT  NOHIST  SCTRPL                   | MON00110
C     |                      DOTDATA    LABEL    EQUPRM  MULBAY               | MON00120
C     |                      GTTCN   DAMRG                                    | MON00130
C     |                      AMUEMA CLASY TESTSP GTOOM EXIT                   | MON00140
C     |   RETURNS.. NONE                                                      | MON00150
C     |                                                                       | MON00160
C     ------------------------------------------------------------------------- MON00170
C                                                                               MON00180
      IMPLICIT INTEGER(A-H,O-Z)                                                 MON00190
      COMMON ARRAY                                                              MON00200
      DIMENSION ARRAY(10600)                                                    MON00210
C                                                                               MON00220
C*    'ARRAY' IS A BLOCK OF STORAGE PASSED TO EACH PROCESSOR FOR THE            MON00230
C*    VARIABLE DIMENSIONING OF OTHER ARRAYS. THE ARRAY IS NEVER USED            MON00240
C*    TO PASS INFORMATION  FROM ONE PROCESSOR TO ANOTHER.                       MON00250
C*                                                                              MON00260
      DATA TOP/10600/                                                           MON00270
      INCLUDE COMBKA.LIST                                                       MON00280
C                                                                               MON00290
C                                                                               MON00300
      INCLUDE COMNTA.LIST                                                       MON00310
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,HMFILE,RMKEY,                 MON00320
     *             HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,          MON00330
     *    DRUMAD,DRM.OS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                      MON00340
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                          MON00350
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,RMTPFL,HISTFL,PCHUNT,                    MON00360
     *  CHOUNT,PRTUNT,RANDIO                                                  MON00370
C*                                                                            MON00380
C*    GLOBAL COMMON IS USED IN EVERY PROCESSOR. IT IS ALWAYS IN CORE.         MON00390
C*    ALL PARAMETERS ARE INITIALIZED IN THE MONITOR ROUTINE OR BLKCOM         MON00400
C*    EXCEPT AS NOTED BELOW                                                   MON00410
C*    DEFINITIONS                                                             MON00420
C*        HEAD   - STANDARD HEADING PRINTED ON MOST OUTPUT PAGES.             MON00430
C*        MAPTAP - FORTRAN UNIT NUMBER ON WHICH THE MAPTAP FILE IS            MON00440
C*                 WRITTEN (=2)                                               MON00450
C*        DATAPE - UNIT NO. FOR THE IMAGE DATA TAPE (=3)                      MON00460
C*        SAVTAP - UNIT NO. ON WHICH THE STATISTICS FILE IS WRITTEN (=1)      MON00470
C*        HMFILE - UNIT NO. ON WHICH THE R-MATRIX FILE IS WRITTEN (=10)       MON00480
C*        RMKEY - TRIGGER INDICATING THAT THE R-MATRIX FILE HAS BEEN          MON00490
C*                WRITTEN. CAN BE SET IN SELECT CLASSIFY OR DATA-TP.          MON00500
C*        HISFIL - UNIT NO. ON WHICH THE HISTOGRAM FILE IS WRITTEN (=13)      MON00510
C*        HISKEY - TRIGGER INDICATING THE HISTOGRAM FILE HAS BEEN             MON00520
C*                 WRITTEN. SET IN HIST PROCESSOR.                            MON00530
C*        TRFORM - UNIT NO. ON WHICH THE TRANSFORMED IMAGE IS WRITTEN BY      MON00540
C*                 THE DATA-TRANSFORMATION PROCESSOR. (=14)                   MON00550
C*        ERIPTP - UNIT NO. ON WHICH THE ISOCLS PROCESSOR WRITES              MON00560
C*                 CLUSTER STATISTICS FOR THE ERIPS SYSTEM. (=15)             MON00570
C*        ERPKEY - TRIGGER INDICATING THAT THE ERIPS INTERFACE TAPE           MON00580
C*                 HAS BEEN WRITTEN.                                          MON00590
C*        MAPUNT - UNIT NO. ON WHICH THE ISOCLS OR DISPLAY PROCESSOR          MON00600
C*                 WRITES THE CLUSTERED OR CLASSIFIED DATA                    MON00610
C*                 TO BE DISPLAYED ON THE PMIS DAS                            MON00620
C*        NOFILE - NO. OF FILES WRITTEN ON UNIT 1A (MAP OUTPUT TAPE)          MON00630
C*                 BY DISPLAY AND/OR ISOCLS                                   MON00640
C*                 SET EITHER IN ISOCLS OR DISPLAY.                           MON00650
C*        DRUMAD - BEGINNING ADDRESS FOR THE RANDOM ACCESS HIGH SPEED         MON00660
C*    REAL TIME                                                               MON00670
C*                 DRUM FILE. THIS FILE IS USED AS A SCRATCH FILE IN          MON00680
C*                 SEVERAL PROCESSORS. REFERENCES TO SYSTEM ROUTINES          MON00690
C*                 'READ' AND 'WRITE' ACCESS THIS FILE.                       MON00700
C*        DRMWDS - NO. OF WORDS AVAILABLE ON THE RANDOM ACCESS FILE.          MON00710
C*        PAGSIZ - NO. OF LINES AVAILABLE FOR PRINTING ON A PAGE.             MON00720
C*        DATFIL - NO. OF E-O-F'S TO BE READ OVER BY TAPERD ROUTINE IN        MON00730
C*                 ORDER TO POSITION THE DATA TAPE TO DESIRED FILE            MON00740
C*        STAFIL - NO. OF E-O-F'S TO SKIP OVER TO POSITION STAT FILE.)        MON00750
C*          ASAV - UNIT NO. ON WHICH TWSTAT WRITES THE TRANSFORMED            MON00760
C*                 STATS                                                      MON00770
C*        ASAVFL - NO. OF E-O-F'S TI SJIP OVER TO POSITION TRANSFORMED        MON00780
```

FILE: MONPAC FORTRAN A

```
C*                    STATS                                                  MON00800
C*          DOTUNT - UNIT NO. ON WHICH DOT DATA FILE (DOTFIL) IS WRITTEN     MON00810
C*          DOTFIL - NO. OF E-O-F'S TO SKIP OVER TO POSITION DOTFIL FILE     MON00820
C*          NCHPAS - NO. OF CHANNELS PER PASS                                MON00830
C*          TRNSFL - NO. OF E-O-F'S TO SKIP OVER FOR TREORM FILE             MON00840
C*          AMTXFL - NO. OF E-O-F'S TO SKIP OVER FOR AMFIL FILE              MON00850
C*          HISTFL - NO. OF E-O-F'S TO SKIP OVER FOR HISFIL FILE             MON00860
C*          PUNCH  - UNIT NO. FOR CARD PUNCH FILE                            MON00870
C*          CRDUNT - UNIT NO. FOR CARD READER                                MON00880
C*          RANDIO  - SCRATCH UNIT FOR RREAD AND RWRITE ROUTINES             MON00890
CSEND                                                                        MON00900
      DBUG=-1                                                                MON00910
C*                                                                           MON00920
C*    SYSTEM ROUTINE RINIT ASSIGNS THE RANDOM ACCESS DRUM FILE.              MON00930
C*                                                                           MON00940
C*          -DRUMAD- IS THE ADDRESS TO BEGIN WRITING                         MON00950
C*          -DRWDS- IS THE NO. OF WORDS AVAILABLE ON THE DRUM FILE.          MON00960
C*                                                                           MON00970
C*    THE FOLLOWING PROCESSORS USE THE RANDOM ACCESS DRUM FILE FOR SCRATCH   MON00980
C*          -ISOCLS-                                                         MON00990
C*          -DISPLY-                                                         MON01000
C*          -SELECT-                                                         MON01010
C*                                                                           MON01020
C*          -GWAYMP-                                                         MON01030
C*          -SIGEXT-                                                         MON01040
C*                                                                           MON01050
      DEFINE FILE 22(640,200,U,ID)                                           MON01060
      DRUMAD=1                                                               MON01070
      DRWDS=12*000                                                           MON01080
      WRITE(22*1)DRUMAD                                                      MON01090
   10 CONTINUE                                                               MON01100
      TIME = 0.                                                              MON01110
      CALL CLOCK(TIME)                                                       MON01120
      CALL MSCAN(JGO,DBUG)                                                   MON01130
      GO TO (20,40,60,80,100,120,140,160,175,180,200,220,240,260,            MON01140
     * 280,290,300,310,320,330,340,350,360),JGO                              MON01150
C                                                                            MON01160
   20 CONTINUE                                                               MON01170
      CALL STAT(ARRAY,TOP)                                                   MON01180
      CALL CLOCK(TIME)                                                       MON01190
      WRITE (6,30)TIME                                                       MON01200
   30 FORMAT(' TIME FOR STAT',F10.3)                                         MON01210
      GO TO 10                                                               MON01220
C                                                                            MON01230
C                                                                            MON01240
   40 CONTINUE                                                               MON01250
      CALL CLSFY(ARRAY,TOP)                                                  MON01260
      CALL CLOCK(TIME)                                                       MON01270
      WRITE (6,50)TIME                                                       MON01280
   50 FORMAT(' TIME FOR CLASSIFY',F10.3 )                                    MON01290
      GO TO 10                                                               MON01300
C                                                                            MON01310
C                                                                            MON01320
   60 CONTINUE                                                               MON01330
      CALL DSPLAY(ARRAY,TOP)                                                 MON01340
      CALL CLOCK(TIME)                                                       MON01350
      WRITE (6,70)TIME                                                       MON01360
   70 FORMAT(' TIME FOR DISPLAY',F10.3)                                      MON01370
      GO TO 10                                                               MON01380
C                                                                            MON01390
C                                                                            MON01400
   80 CONTINUE                                                               MON01410
      CALL SELECT(ARRAY,TOP)                                                 MON01420
      CALL CLOCK(TIME)                                                       MON01430
      WRITE (6,90)TIME                                                       MON01440
   90 FORMAT(' TIME FOR SELECT',F10.3)                                       MON01450
      GO TO 10                                                               MON01460
C                                                                            MON01470
C                                                                            MON01480
  100 CONTINUE                                                               MON01490
      CALL HIST(ARRAY,TOP)                                                   MON01500
      CALL CLOCK(TIME)                                                       MON01510
      WRITE(6,110)TIME                                                       MON01520
  110 FORMAT(' TIME FOR HISTOGRAM',F10.3)                                    MON01530
      GO TO 10                                                               MON01540
C                                                                            MON01550
C     GO HERE FOR ISOCLS                                                     MON01560
C                                                                            MON01570
  120 CONTINUE                                                               MON01580
```

```
FILE: MONPAC   FORTRAN  A

      CALL ISOCLS(ARRAY,TOP)                                          MON01590
      CALL CLOCK(TIME)                                                MON01600
      WRITE(6,130) TIME                                               MON01610
  130 FORMAT(' TIME FOR ISOCLS',F10.3)                                MON01620
      GO TO 10                                                        MON01630
C                                                                     MON01640
C     GO HERE FOR GRAYMAP                                             MON01650
C                                                                     MON01660
  140 CONTINUE                                                        MON01670
      CALL GRAYMP(ARRAY,TOP)                                          MON01680
      CALL CLOCK(TIME)                                                MON01690
      WRITE(6,150) TIME                                               MON01700
  150 FORMAT(' TIME FOR GRAYMAP',F10.3)                               MON01710
      GO TO 10                                                        MON01720
C                                                                     MON01730
CC    GO HERE FOR DATA-TRANSFORMATION                                 MON01740
C                                                                     MON01750
  160 CONTINUE                                                        MON01760
      CALL DATATR(ARRAY,TOP)                                          MON01770
      CALL CLOCK(TIME)                                                MON01780
      WRITE(6,170) TIME                                               MON01790
  170 FORMAT(//' TIME FOR DATA-TRANSFORMATION',F10.3)                 MON01800
      GO TO 10                                                        MON01810
C*                                                                    MON01820
C*    GO HERE FOR SIGEXT MODULE                                       MON01830
C*                                                                    MON01840
  175 CONTINUE                                                        MON01850
C     *** SIGEXT                                                      MON01860
      CALL CLOCK(TIME)                                                MON01870
      WRITE(6,176)TIME                                                MON01880
  176 FORMAT(//' TIME FOR SIGEXT ',F10.3, 'MINUTES' )                 MON01890
      GO TO 10                                                        MON01900
C                                                                     MON01910
CC    GO HERE FOR TRSTAT                                              MON01920
C                                                                     MON01930
  180 CONTINUE                                                        MON01940
      CALL TRSTAT(ARRAY,TOP)                                          MON01950
      CALL CLOCK(TIME)                                                MON01960
      WRITE(6,190)TIME                                                MON01970
  190 FORMAT(//' TIME FOR TRSTAT',F10.3,' MINUTES')                   MON01980
      GO TO 10                                                        MON01990
C                                                                     MON02000
CC    GO HERE FOR NOHIST                                              MON02010
C                                                                     MON02020
  200 CONTINUE                                                        MON02030
      CALL NOHIST(ARRAY,TOP)                                          MON02040
      CALL CLOCK(TIME)                                                MON02050
      WRITE(6,210)TIME                                                MON02060
  210 FORMAT(//' TIME FOR NOHIST',F10.3,' MINUTES')                   MON02070
      GO TO 10                                                        MON02080
C                                                                     MON02090
CC    GO HERE FOR SCTRPL                                              MON02100
C                                                                     MON02110
  220 CONTINUE                                                        MON02120
      CALL SCTRPL(ARRAY,TOP)                                          MON02130
      CALL CLOCK(TIME)                                                MON02140
      WRITE(6,230)TIME                                                MON02150
  230 FORMAT(//' TIME FOR SCTRPT',F10.3,' MINUTES')                   MON02160
      GO TO 10                                                        MON02170
C                                                                     MON02180
CC                                                                    MON02190
CC    GO HERE FOR DOTDATA                                             MON02200
C                                                                     MON02210
  240 CONTINUE                                                        MON02220
      CALL DOTDAT(ARRAY,TOP)                                          MON02230
      CALL CLOCK(TIME)                                                MON02240
      WRITE(6,250)TIME                                                MON02250
  250 FORMAT(//' TIME FOR DOTDATA',F10.3,' MINUTES')                  MON02260
      GO TO 10                                                        MON02270
C                                                                     MON02280
CC    GO HERE FOR LABEL                                               MON02290
C                                                                     MON02300
  260 CONTINUE                                                        MON02310
      CALL LABEL(ARRAY,TOP)                                           MON02320
      CALL CLOCK(TIME)                                                MON02330
      WRITE(6,270)TIME                                                MON02340
  270 FORMAT(//' TIME FOR LABEL',F10.3,' MINUTES')                    MON02350
      GO TO 10                                                        MON02360
C                                                                     MON02370
C      GO HERE FOR FOUI-PROBABLE BLOCKS CLASSIFIER
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: MONPAC FORTRAN A

```
C
 280  CONTINUE                                                          MON02380
      CALL EQUPRR(ARRAY,TOP)                                            MON02390
      CALL CLOCK(TIME)                                                  MON02400
      WRITE(6,285) TIME                                                 MON02410
 285  FORMAT(//' TIME FOR EQUPRR',F10.3,'MINUTES')                      MON02420
      GO TO 10                                                          MON02430
C                                                                       MON02440
CCC   GO HERE FOR MULTI-TEMPORAL CLASSIFIER                             MON02450
C                                                                       MON02460
 290  CONTINUE                                                          MON02470
      CALL MULBAY(ARRAY,TOP)                                            MON02480
      CALL CLOCK(TIME)                                                  MON02490
      WRITE(6,295) TIME                                                 MON02500
 295  FORMAT(//' TIME FOR MULBAY',F10.3,'MINUTES')                      MON02510
      GO TO 10                                                          MON02520
C                                                                       MON02530
CCC   GO HERE FOR GROUND TRUTH TO MAPFIL                                MON02540
C                                                                       MON02550
 300  CONTINUE                                                          MON02560
      CALL GTTCN(ARRAY,TOP)                                             MON02570
      CALL CLOCK(TIME)                                                  MON02580
      WRITE(6,305) TIME                                                 MON02590
 305  FORMAT(//' TIME FOR GTTCN',F10.3,'MINUTES')                       MON02600
      GO TO 10                                                          MON02610
C                                                                       MON02620
C     GO HERE FOTR IMAGE DATA MERGE                                     MON02630
 310   CONTINUE                                                         MON02640
      CALL DAMRG(ARRAY,TOP)                                             MON02650
      CALL CLOCK(TIME)                                                  MON02660
      WRITE(6,315) TIME                                                 MON02670
 315  FORMAT(//' TIME FOR DAMRG',F10.3,'MINUTES')                       MON02680
      GO TO 10                                                          MON02690
C                                                                       MON02700
CCC   GO HERE FOR AMOEBA                                                MON02710
C                                                                       MON02720
 320   CONTINUE                                                         MON02730
      CALL AMOEBA(ARRAY,TOP)                                            MON02740
      CALL CLOCK(TIME)                                                  MON02750
      WRITE(6,325) TIME                                                 MON02760
 325  FORMAT(//' TIME FOR AMOEBA',F10.3,'MINUTES')                      MON02770
      GO TO 10                                                          MON02780
C                                                                       MON02790
CCC   GO HERE FOR CLASY                                                 MON02800
C                                                                       MON02810
 330   CONTINUE                                                         MON02820
      CALL CLASY(ARRAY,TOP)                                             MON02830
      CALL CLOCK(TIME)                                                  MON02840
      WRITE(6,335) TIME                                                 MON02850
 335  FORMAT(//' TIME FOR CLASY',F10.3,' MINUTES')                      MON02860
      GO TO 10                                                          MON02870
C                                                                       MON02880
CCC   GO HERE FOR TESTSP                                                MON02890
C                                                                       MON02900
 340   CONTINUE                                                         MON02910
      CALL TESTSP(ARRAY,TOP)                                            MON02920
      CALL CLOCK(TIME)                                                  MON02930
      WRITE(6,345) TIME                                                 MON02940
 345  FORMAT(//' TIME FOR TESTSP',F10.3,' MINUTES')                     MON02950
      GO TO 10                                                          MON02960
C                                                                       MON02970
CCC   GO HERE FOR GROUND TRUTH DOT UNLOAD                               MON02980
C                                                                       MON02990
 350   CONTINUE                                                         MON03000
      CALL GTDDM(ARRAY,TOP)                                             MON03010
      CALL CLOCK(TIME)                                                  MON03020
      WRITE(6,355) TIME                                                 MON03030
 355  FORMAT(//' TIME FOR GTDDM',F10.3,' MINUTES')                      MON03040
      GO TO 10                                                          MON03050
C                                                                       MON03060
CCC   GO HERE TO EXIT                                                   MON03070
C                                                                       MON03080
 360  IF (NOFILE .GT. 0) REWIND MAPUNT                                  MON03090
      END                                                               MON03100
                                                                        MON03110
```

FILE: MSCAN FORTRAN A

```
      SUBROUTINE MSCAN(MGO,DBUG)                                        MSC00010
      IMPLICIT INTEGER (A-H,O-Z)                                        MSC00020
      DIMENSION CODTAB(23),COMENT(15),DATE(3)                           MSC00030
     1 ,        HED1(15), HED2(15),ACARD(20)                           MSC00040
C                                                                       MSC00050
CCI---------------------------------------------------------------------IMSC00060
CCCI--------------------------------------------------------------------IMSC00070
CCCCI                                                                   IMSC00080
CCCCI    CALL..    CALL MSCAN(MGO,DBUG)                                 MSC00090
CCCCI                                                                   IMSC00100
CCCCI      CONTINUE                                                     MSC00110
CCCCI    ARGS..    JGO    - PROCESSOR PTR                             IMSC00120
CCCCI              DBUG   - DEBUG KEY  -1 FIRST ENTRY                 IMSC00130
CCCCI                                   0 - INCLUDE FLASH             IMSC00140
CCCCI                                   1 - EXCLUDE FLASH             IMSC00150
CCCCI                                                                 IMSC00160
CCCCI                                                                 IMSC00170
CCCCI    PURPOSE.. ANALYZES ALL MONITOR CONTROL CARDS                 IMSC00180
CCCCI                                                                 IMSC00190
CCCCI    RETURNS.. JGO - 1 $STAT                                      IMSC00200
CCCCI                   2 $CLASS                                      IMSC00210
CCCCI                   3 $DISPLAY                                    IMSC00220
CCCCI                   4 $SELECT                                     IMSC00230
CCCCI                   5 $HIST                                        MSC00240
CCCCI                   6 $ISOCLS                                      MSC00250
CCCCI                   7 $GRAYMAY                                     MSC00260
CCCCI                   8 $DATA-TRANSFORMATION                         MSC00270
CCC*                    9 $SIG EXT                                     MSC00280
CCC*                   10 $TRSTAT                                      MSC00290
CCC*                   11 $NDHIST                                      MSC00300
CCC*                   12 $SCTRPL                                      MSC00310
CCC*                   13 $DOTDAT                                      MSC00320
CCC*                   14 $LABEL                                       MSC00330
CCC*                   15 $EQUPROB BLOCKS                              MSC00340
CCC*                   16 $MULTI-TEMPORAL BAYES CLASSIFIER             MSC00350
CCC*                   17 $GROUND TRUTH TO CLUSTER MAP                 MSC00360
CCC*                   18 $DATA MERGING                                MSC00370
CC*                    19 $AMOEBA                                      MSC00380
C*         CONTINUE                                                     MSC00390
CC*                    20 $CLASY                                       MSC00400
CCC*                   21 $TESTSP                                      MSC00410
CCCC*                 22 $GTODM                                        MSC00420
CCCC*                 23 $EXIT                                         MSC00430
CCCI                                                                  IMSC00440
CCCI--------------------------------------------------------------------IMSC00450
CCCI---------------------------------------------------------------------IMSC00460
      EQUIVALENCE      (HED1(1), HEAD(4)),  (DATE(1), HEAD(22)),       MSC00470
     1                 (HED2(1), HEAD(30)), (COMENT(1), HEAD(48))      MSC00480
C                                                                       MSC00490
C     INCLUDE COMRKO.LIST                                               MSC00500
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         MSC00510
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,   MSC00520
     *    DRUMAD,DRMNOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               MSC00530
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                   MSC00540
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,             MSC00550
     *  CROUNT,PRTUNT,RANDIO                                           MSC00560
C*END                                                                   MSC00570
      COMMON /TAPERD/ IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,      MSC00580
     *  LINC,ID(200),DSL,LBUF(30),JREC(30),IBYTE(30),NBUFS,FILENO,      MSC00590
     *  LINEND,LININC,NSAMP,JCHAN,FORMT                                MSC00600
      DATA CODTAB /'$STA','$CLA','$DIS','$SEL','$HIS','$ISO','$GRA',   MSC00610
     *            '$DAT','$SIG','$TRS','$NDH','$SCT','$DOT',          MSC00620
     * '$LAB','$EQU','$MUL','$GTT','$DAM',                            MSC00630
     * '$AMO','$CLS','$TES','$GTO','$EXI'/                            MSC00640
      DATA FF /'FORM'/, LFORM /'2'/                                     MSC00650
      DIMENSION CARD(62)                                                MSC00660
C                                                                       MSC00670
      ------------------------------------------------------------------MSC00680
      ------------------------------------------------------------------MSC00690
                                                                        MSC00700
                                                                        MSC00710
      INIT                                                              MSC00720
      ------------------------------------------------------------------MSC00730
C                                                                       MSC00740
                                                                        MSC00750
        WRITE(6,200)                                                    MSC00760
  200   FORMAT(// 5X,'PLEASE BE ADVISED OF THE CHANGE TO CLASSIFY       MSC00770
     *  AND DISPLAY PROCESSOR CARD IMAGES')                            MSC00780
        WRITE(6,210)                                                    MSC00790
  210   FORMAT(5X,'MAPTAP FILE AND UNIT NUMBER NOW APPEAR AS
```

```
FILE: MSCAN    FORTRAN  A

     * ORDINARY CONTROL CARDS  FIRST ONES AFTER PROCESSOR CARD*)          MSC00800
      CODDEM = 23                                                        MSC00810
      IF (DBUG.GE.0) GO TO 10                                            MSC00820
      DBUG = 0                                                           MSC00830
       FORMT = 1                                                         MSC00840
C                                                                        MSC00850
CC    SET THE DATE FROM GTDATE                                           MSC00860
C                                                                        MSC00870
   10 WRITE (6,HEAD)                                                     MSC00880
C                                                                        MSC00890
CC    READ AND DECODE THE MONITOR CARD                                   MSC00900
CC    ------------------------------------------------------------------MSC00910
CC                                                                       MSC00920
CC    SET UP RREAD BUFFER                                                MSC00930
C                                                                        MSC00940
      RRUNIT=30                                                          MSC00950
      CALL REREAD(RRUNIT,80)                                             MSC00960
C                                                                        MSC00970
CC    NOW PUT THE CARD IN BUFFER                                         MSC00980
C                                                                        MSC00990
   20 READ(21,25)(ACARD(I),I=1,20)                                       MSC01000
   25 FORMAT(20A4)                                                       MSC01010
      WRITE(RRUNIT,25)(ACARD(I),I=1,20)                                  MSC01020
      REWIND RRUNIT                                                      MSC01030
      READ(RRUNIT,30)CODE1,CODE2                                         MSC01040
   30 FORMAT(2A4)                                                        MSC01050
      REWIND RRUNIT                                                      MSC01060
      WRITE(6,40)CODE1,CODE2                                             MSC01070
   40 FORMAT(1X,2A4/)                                                    MSC01080
      DO 50 JGO=1,CODDEM                                                 MSC01090
      MGO = JGO                                                          MSC01100
      IF (CODTAB(JGO) .EQ. CODE1) RETURN                                 MSC01110
   50 CONTINUE                                                           MSC01120
      IF(CODE1.NE.FF) GO TO 55                                           MSC01130
      READ(30,52) CARD                                                   MSC01140
      REWIND 30                                                          MSC01150
   52 FORMAT(10X,62A1)                                                   MSC01160
      COL=0                                                              MSC01170
      LFOR= NXTCHR(CARD,COL)                                             MSC01180
      IF(LFOR.EQ.LFORR) FORMT=2                                          MSC01190
      GO TO 20                                                           MSC01200
   55 CONTINUE                                                           MSC01210
      WRITE (6,60)                                                       MSC01220
      GO TO 20                                                           MSC01230
   60 FORMAT(// 5X,'*****  MSCANN  ---  PROCESSOR CARD REQUIRED  --- ABOMSC01240
     1VE CARD NOT A VALID PROCESSOR CARD  ---  CONTINUING SCAN  *****'/)MSC01250
C                                                                        MSC01260
       END                                                               MSC01270
```

5. COMMON BLOCKS AND BLOCK DATA

The common block listings are given in this section. For specific descriptions, definitions of the parameters, and processor and subprogram interfaces, see section 5 of volume III of this user guide. The common block listings are given here in alphabetical order, as they appear in volume III.

```
COMMON/BESTKN/ KPPPTS(60), IPRIOR, KBEST, NCPASS


COMMON /BMTRX/BMATRX(450)


COMMON /CLASS/ APRFLG,BMCOMB,BMFEAT,BMFLG,NOCAT,THIJ1,IDATA1,
               NFILE,STATKY,CATNAM(60),
               CLSSYM(60),CON(60),DET(60),FLDESC,FLDINF(6),
               KCLSNA(60),NOCTCL(60),SUBCAT(60)
    ,NOCHAN,CHNVEC(30)


COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60),
             SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,
             PCFDKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,
             PLTKEY,BMFLG,BMCOMB,BMFEAT,CDATE(2),
             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,
             THRES(60),SYMMTX(66),HIGH(60),CON(60)
             ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)
             ,NOSUB2,NOTREE,TOTVT2,NOCLS2
             ,KATNO(60),NOCAT,FILTER,MAPFMT
             ,DESKEY,DESUNI,DESOTH,CROP  ,ACROP,AOTHER,ATOTAL
             ,SITE(6),ANALYS(5),CAM(15),CRPKEY,KEPPTS(60)
             ,DOTKEY,DOTERR


COMMON /DOTVEC/TYPE,CATNAM(60),NOCAT,TOTVEC,FLDINF(6),PRTKEY
               ,SIZE ,LACIF


COMMON/DUMBLK/DFDK,CAYMIN,FII,CCAY,IID,IIDMEN,ITT,ICNT,N


COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,
    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)
     ,NOFET4,VARS74,CORBAS,DTAB4,WGHS14,BESTVC(10),DIVSI7
    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2


COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,
              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,
    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL
  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL
  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,
  CRDUNT,PRTUNT,RANDIO
```

```
COMMON /GRCHLK/MAXFET,NOFEAT,NOFET2,FETVEC(30),
               FETVC2(30),FLDINF(6),INFMT,FILESV,NOHIST,
               HISVEC(30),NOFLD,          FLDPTS
,XSIZ,XLOW,XHGH,YSIZ


COMMON /HISTOR/HF


COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,
              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,
              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),
              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),
              GRPCHK(61),GROUPS(124)


COMMON/ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP


COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),
             SUBRAY(120),PTR(60),CATPTR(250),CATDOT(500),
              DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,
  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,
   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),
              CLSSYM(62),STADRS,MEANAD,TABADR,MAPADR,SUNCOR(30) ,
       ODOTUN,ODOTFI,MANSTA,MANDOT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,
  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)
    ,TOTDT3,FLDADR,VTXADR


COMMON /LISTMM/ NPGA(3,2),NAMPGA(209,3,2),LINPGA(209,3,2),


COMMON /NDIM/NCLRCH,CLRVEC(30),MAXVEC,MAPKEY,     ,
      CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(6),SIZE,TOTMNS
    ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,BUFLEN,ID3,COLOR3,NODUMP
    ,IDATA1,TOTVEC


COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,
                       MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,
  ICHN,CHNTHS,ICHAIN(62),NWDS,IREGIN,BEGIN1,
              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,
              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT
 ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)
,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR
,IORDER,INUNIT,INFILE,INITM,PMIN,SURVEC(62),NOSUB2,CHNVC(30)
 ,NOCHAN,ERCORR,NOSEQ,MEANDO,MEANDU,
 SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,
  DUFLAG,DODU,STDOTS(60),NSDOTS,SUNCOR(30),LLNCAT,
 DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)
  ,MXFET1,MAXPOP


COMMON/PASSA/NOFET1,FTVEC1(30)


COMMON /PASSB/ NOCLS,NOSUB,NOFEAT,NOFLD,TOTVRT,
               FETVEC(30),FLDSV1,CLSID1
```

```
      COMMON /SCRACH/SCR1(2000),SCR2(10500)


      COMMON/SCTTER/RSCALF,XYSCLF,CLRVEC(30),NCLRCH,CLRKEY,LOG,
     FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,
     YHI,YSIZ,NBINS,SYMMTX(32),BMATRX(60),BVEC(30),NBVCHN,NOFEAT
      ,SCALKY,MENADR,FLDADR,PNTADR,IDADR,NC,BMFEAT,BMCOMB
      ,NOVEC,TOTMNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,
     DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD
     ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE
     ,NOSUB


      COMMON /STBASE/SUBSV1,SUBMN1,SUBVR1,SUBSD1,SUBCL1,SAVER1,HSTAL1,
     SPEC1,COVAR1,AVAR1,CLSID1,FLMEN1,FLVAR1,HFTAL1,FLDSV1
     COMMON /STCRLK/          MAXFET,MAXCLS,MAXFLD,NOFEAT,NOFET2,
     VARSIZ,NOSPEC,NOHIST,SPCBAS,IBLOCK(30),FETVEC(30),
     FETVC2(30),HISVEC(30),NOFLD,NOCLS,
        FLDINF(6),FLDPTS,CLSPTS,XSIZ,XHGH,XLOW,YSIZ


      COMMON /STCRLK/          MAXFET,MAXCLS,MAXFLD,NOFEAT,NOFET2,
     VARSIZ,NOSPEC,NOHIST,SPCBAS,IBLOCK(30),FETVEC(30),
     FETVC2(30),HISVEC(30),NOFLD,NOCLS,
        FLDINF(6),FLDPTS,CLSPTS,XSIZ,XHGH,XLOW,YSIZ


      COMMON /TAPERD/ IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,
        LINC,ID(200),DSL,LBUF(30),JREC(30),IBYTE(30),NBUFS,FILENO,LINEND
        ,LININC,NSAMP,NOCHAN,FORMT


      COMMON/TRBLCK/OUTFMT,NOFEAT,FLDINF(6),          FETVEC(30)


      COMMON /WRTAPE/ICOUNT,FORMT,UNIT,VARBL(600),IREND
```

ORIGINAL PAGE IS
OF POOR QUALITY

6. HIST PROCESSOR

FILE: HIST FORTRAN A

```
      SUBROUTINE HIST(ARRAY,TOP)                                        HIS00010
C                                                                       HIS00020
C     IMPLICIT INTEGER (F-T)                                            HIS00030
C                                                                       HIS00040
C     DIMENSION ARRAY(1)                                                HIS00050
C                                                                       HIS00060
C     //HIST                                                            HIS00070
CCCCI-----------------------------------------------------------------  HIS00080
CCCCI-----------------------------------------------------------------  HIS00090
CCCCC                                                                   HIS00100
CCCCC     CALL.      CALL HIST(ARRAY,TOP)                               HIS00110
CCCCC                                                                   HIS00120
CCCCC     REQUIRES.  NO COMMON BLOCKS                                   HIS00130
CCCCC                ROUTINES  SETUPS                                   HIS00140
CCCCC                          HISTGM                                   HIS00150
CCCCC                                                                   HIS00160
CCCCC     PURPOSE.   COORDINATES THE LOGICAL STEPS FOR HISTOGRAMMING    HIS00170
CCCCC                DATA                                               HIS00180
CCCCC                                                                   HIS00190
CCCCC     RETURNS.   NONE                                               HIS00200
CCCCI-----------------------------------------------------------------  HIS00210
CCCCI-----------------------------------------------------------------  HIS00220
CCCCC                                                                   HIS00230
CCCCC     CALL SETUPS TO READ IN CONTROL CARDS                          HIS00240
C     C                                                                 HIS00250
C     C   CALL SETUPS(FILHS,FLDTL,TOTTL,TOP)                           HIS00260
C     C                                                                 HIS00270
C     C                                                                 HIS00280
C     C   CALL HISTGM(ARRAY(FILHS),ARRAY(FLDTL),ARRAY(TOTTL))          HIS00290
      RETURN                                                            HIS00300
      END                                                               HIS00310
```

FILE: SETUP5 FORTRAN A

```
      SUBROUTINE SETUP5(FILNS,FLDTL,TOTTL,TOP)                          SET00010
      IMPLICIT INTEGER(A-Z)                                             SET00020
CI    PURPOSE  -- READS THE CONTROL CARDS FOR HIST PROCESSOR            SET00030
C                                                                       SET00040
CCC   INCLUDE COMBK3,LIST                                               SET00050
CCC   INCLUDE COMBK4,LIST                                               SET00060
C     INCLUDE COMBK6,LIST                                               SET00070
      COMMON /GRCHLK/MAXFET,NOFEAT,NOFET2,FETVEC(30),                   SET00080
     *               FETVC2(30),FLDINF(6),INFMT,FILESV,NOHIST,          SET00090
     *               HISVEC(30),NOFLD,          FLDPTS                 SET00100
     *,XSIZ,XLOW,XHGH,YSIZ                                              SET00110
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                    SET00120
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                 SET00130
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               SET00140
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         SET00150
     *              HISFIL,HISKEY,TRFORM,ERIPTR,ERRKEY,MAPUNT,NOFILE,   SET00160
     *    DWUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                SET00170
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    SET00180
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              SET00190
     *   CRDUNT,PRTUNT,RANDIO                                           SET00200
CSEND                                                                   SET00210
C                                                                       SET00220
      DIMENSION CARD(62),ACARD(20)                                      SET00230
      DIMENSION EQUVEC(2)                                               SET00240
      DIMENSION INVECT(9),NUMVEC(30),IPTVEC(30),IPT(2),CHAR(2)          SET00250
     *,         NUMVC1(30)                                             SET00260
C                                                                       SET00270
C                                                                       SET00280
      DATA INVECT /'CHAN','HED1','HED2','COMM','DATE',                  SET00290
     * 'DISP','END','SIZE','DATA'/                                      SET00300
      DATA EQUVEC/1,'='/                                                SET00310
      DATA BLANK/' '/,CHAR/1,'='/,EQUAL/'='/,IBCD/'I'/                  SET00320
      DATA UBCD/'U'/ , LBCD/'L'/ , HBCD/'H'/                            SET00330
      DATA YBCD/'Y'/, FBCD/'F'/ , XBCD/'X'/                             SET00340
C                                                                       SET00350
      INFMT=2                                                           SET00360
      XSIZ = 101                                                        SET00370
      YSIZ = 15                                                         SET00380
      XHGH = 255                                                        SET00390
      XLOW = 0                                                          SET00400
      NOFEAT = 0                                                        SET00410
      NOHIST = 0                                                        SET00420
C     CALL IDATE(DATE)                                                  SET00430
      WRITE(6,630)                                                      SET00440
C                                                                       SET00450
CCC   SETUP REREAD BUFFER                                               SET00460
C                                                                       SET00470
      CALL REREAD(30,80)                                                SET00480
C   NOW PUT THE NEXT CARD IN THE BUFFER                                 SET00490
   10 READ(21,15)(ACARD(I),I=1,20)                                      SET00500
   15 FORMAT(20A4)                                                      SET00510
      WRITE(30,15)(ACARD(I),I=1,20)                                     SET00520
      REWIND 30                                                         SET00530
      READ(30,480)CODE,CARD                                             SET00540
      REWIND 30                                                         SET00550
      WRITE(6,550) CODE,CARD                                            SET00560
      COL = 0                                                           SET00570
      DO 20 I=1,9                                                       SET00580
      IF(CODE.EQ.INVECT(I)) GO TO (30,40,50,60,70,                      SET00590
     * 80,150,100,1701),I                                              SET00600
   20 CONTINUE                                                          SET00610
      WRITE(6,490) CODE,CARD                                            SET00620
      GO TO 10                                                          SET00630
C                                                                       SET00640
CCC   CHANNEL CARD                                                      SET00650
CCC                                                                     SET00660
   30 J = NXTCHR(CARD,COL)                                              SET00670
      IF (J .EQ. BLANK)   GO TO 10                                      SET00680
      COL = COL - 1                                                     SET00690
      NOFEAT = NUMBER(CARD,COL,NUMVEC,NOFEAT)                           SET00700
      CALL SORT(NOFEAT,IPT,NUMVEC,IPTVEC)                               SET00710
      KA   = IPT(1)                                                     SET00720
      DO 35       I= 1,NOFEAT                                           SET00730
      FETVEC(I) = NUMVEC(KA)                                            SET00740
   35 KA        = IPTVEC(KA)                                          SET00750
      GO TO 10                                                          SET00760
C                                                                       SET00770
CCC   HED1 CARD                                                         SET00780
C                                                                       SET00790
```

FILE: SETUPS FORTRAN A

```
   40 READ(30,500)HED1                                                  SET00800
      REWIND 30                                                         SET00810
      GO TO 10                                                          SET00820
C                                                                       SET00830
C     HED2 CARD                                                         SET00840
C                                                                       SET00850
   50 READ(30,500)HED2                                                  SET00860
      REWIND 30                                                         SET00870
      GO TO 10                                                          SET00880
C                                                                       SET00890
C     COMMENT CARD                                                      SET00900
C                                                                       SET00910
   60 READ(30,500) COMENT                                               SET00920
      REWIND 30                                                         SET00930
      GO TO 10                                                          SET00940
C                                                                       SET00950
C     DATE CARD                                                         SET00960
C                                                                       SET00970
   70 READ(30,510) DATE                                                 SET00980
      REWIND 30                                                         SET00990
      GO TO 10                                                          SET01000
C                                                                       SET01010
C     DISPLAY CARD                                                      SET01020
C                                                                       SET01030
   90 J = NXTCHR(CARD,COL)                                              SET01040
      IF (J .EQ. BLANK)  GO TO 10                                       SET01050
      COL = COL - 1                                                     SET01060
      NOHIST = NUMBER(CARD,COL,NUMVC1,NOHIST)                           SET01070
      GO TO 10                                                          SET01080
C                                                                       SET01090
C     SIZE CARD                                                         SET01100
C                                                                       SET01110
   97 COL = COL - 1                                                     SET01120
  100 J = NXTCHR(CARD,COL)                                              SET01130
      IF (J .EQ. BLANK)  GO TO 10                                       SET01140
      IF (J .EQ. YRCD)  GO TO 130                                       SET01150
      IF (J .EQ. XRCD)  GO TO 140                                       SET01160
      GO TO 120                                                         SET01170
  130 M = FIND12(CARD,COL,CHAR)                                         SET01180
      IF (CHAR(M) .NE. EQUAL)  GO TO 120                                SET01190
      M = NUMBER(CARD,COL,NUMVEC,0)                                     SET01200
      YSIZ = NUMVEC(1)                                                  SET01210
      GO TO 97                                                          SET01220
  140 J = NXTCHR(CARD,COL)                                              SET01230
      M = FIND12(CARD,COL,CHAR)                                         SET01240
      IF (CHAR(M) .NE. EQUAL)  GO TO 120                                SET01250
      M = NUMBER(CARD,COL,NUMVEC,0)                                     SET01260
      IF (J .EQ. HRCD)  XHGH = NUMVEC(1)                                SET01270
      IF (J .EQ. HRCD)  GO TO 97                                        SET01280
      IF (J .NE. LRCD)  GO TO 120                                       SET01290
      XLOW = NUMVEC(1)                                                  SET01300
      GO TO 97                                                          SET01310
C                                                                       SET01320
C  DATAFILE POSITIONING CARD                                            SET01330
C                                                                       SET01340
 1701 M=NXTCHR(CARD,COL)                                                SET01350
      IF(M.EQ.BLANK) GO TO 10                                           SET01360
      IF(M.EQ.URCD) GO TO 1702                                          SET01370
      IF(M.EQ.FRCD) GO TO 1703                                          SET01380
 1705 WRITE(6,1704)                                                     SET01390
 1704 FORMAT(' ERROR ON DATA FILE CARD ')                               SET01400
      GO TO 10                                                          SET01410
 1702 J=FIND12(CARD,COL,EQUVEC)                                         SET01420
      IF(J.EQ.-1) GO TO 1705                                            SET01430
      M=NUMBER(CARD,COL,DATAPE,ZERO)                                    SET01440
      COL=COL-1                                                         SET01450
      GO TO 1701                                                        SET01460
 1703 J=FIND12(CARD,COL,EQUVEC)                                         SET01470
      IF(J.EQ.-1) GO TO 1705                                            SET01480
      FILNO=NUMBER(CARD ,COL,DATFIL,FILNO)                              SET01490
      DATFIL=DATFIL-1                                                   SET01500
      COL=COL-1                                                         SET01510
      GO TO 1701                                                        SET01520
C                                                                       SET01530
C     *END* CARD                                                        SET01540
C                                                                       SET01550
  150 CONTINUE                                                          SET01560
      IF(NOHIST.EQ.0) GO TO 1                                           SET01570
C                                                                       SET01580
```

~~6-5~~

20

FILE: SETUPS FORTRAN A

```
C       IS DISPLAY A SUBSET OF CHANNEL CARD                                    SET01590
C                                                                              SET01600
        DO 160    J=1,NOHIST                                                   SET01610
        DO 155    I=1,NOFEAT                                                   SET01620
        IF (NUMVC1(J) .EQ. FETVEC(I))  GO TO 160                               SET01630
  155 CONTINUE                                                                 SET01640
        WRITE(6,560)  NUMVC1(J)                                                SET01650
        NUMVC1(J) = 0                                                          SET01660
  160 CONTINUE                                                                 SET01670
        CALL SORT(NOHIST,IPT,NUMVC1,IPTVEC)                                    SET01680
        KA = IPT(1)                                                            SET01690
        DO 165    I=1,NOHIST                                                   SET01700
        HISVEC(I) = NUMVC1(KA)                                                 SET01710
  165 KA    =    IPTVEC(KA)                                                    SET01720
    1 CONTINUE                                                                 SET01730
C                                                                              SET01740
CCC     CHECKING XHIGH AND XLOW                                                SET01750
C                                                                              SET01760
        IF((XHIGH - XLOW) .GE. 100)  GO TO 170                                 SET01770
        XLOW = XHIGH - 100                                                     SET01780
        IF (XLOW .LT. 0)    XHIGH = 100                                        SET01790
        IF (XLOW .LT. 0)    XLOW = 0                                           SET01800
        WRITE(6,570) XHIGH,XLOW                                                SET01810
  170 CONTINUE                                                                 SET01820
C                                                                              SET01830
CCC     BASES FOR ARRAY                                                        SET01840
C                                                                              SET01850
        FILMS = 1                                                              SET01860
        FLOTL = NOFEAT*256 + FILMS                                             SET01870
        TOTTL = NOHIST* XSIZ + FLOTL                                           SET01880
        TOTPTS = NOHIST * XSIZ + TOTTL - 1                                     SET01890
        IF(TOTPTS .LE. TOP)  RETURN                                            SET01900
C                                                                              SET01910
CCC     RESET NO OF CHANNELS TO BE DISPLAYED AND RECALCULATE ADDRESS TOTTL     SET01920
C                                                                              SET01930
        NOHST1 = NOHIST                                                        SET01940
        NOHIST = (TOP - (FLOTL-1) ) / (2*XSIZ)                                 SET01950
        TOTTL = NOHIST*XSIZ + FLOTL                                            SET01960
        NOHST = NOHIST + 1                                                     SET01970
C                                                                              SET01980
CCC     ZERO OUT CHANNELS THAT ARE NOT TO BE PLOTTED                           SET01990
C                                                                              SET02000
        DO 200    I=NOHST,NOHST1                                               SET02010
  200 HISVEC(I) = 0                                                            SET02020
        WRITE(6,650) NOHIST                                                    SET02030
  650 FORMAT(/ ' TOO MANY CHANNELS ARE BEING HISTOGRAMMED AND PLOTTED.'/SET02040
     *       ' NO. OF CHANNELS PLOTTED WAS RESET TO',I5)                       SET02050
        RETURN                                                                 SET02060
  480 FORMAT(A4,6X,62A1)                                                       SET02070
  490 FORMAT('INVALID CARD -- IGNORED'/T5,A4,6X,62A1)                          SET02080
  500 FORMAT(10X,15A4)                                                         SET02090
  510 FORMAT(10X, 3A4)                                                         SET02100
  570 FORMAT(' XHIGH - XLOW WAS LESS THAN 100. XHIGH WAS RESET TO ',I3,        SET02110
     * 2X,' OR XLOW WAS RESET TO ', I3)                                        SET02120
  550 FORMAT(5X,A4,6X,62A1)                                                    SET02130
  560 FORMAT(' CHANNEL ',I2,' IS NOT A SUBSET OF THE CHANNELS GIVEN ON C SET02140
     *HANNEL CARD' )                                                          SET02150
  630 FORMAT(//' INPUT SUMMARY')                                               SET02160
  120 WRITE(6,640) CODE,CARD                                                   SET02170
  640 FORMAT(/ 1X,A4,6X,62A1/ ' BAD SUPERVISOR CONTROL CARD'/)                 SET02180
        GO TO 10                                                               SET02190
        END                                                                    SET02200
```

FILE: SORT   FORTRAN A

```
      SUBROUTINE SORT(IA,IPT,NUMVEC,IPTVEC)                             C
C     ******************************************************************C
C     CALL     -- SORT(IA,IPT,NUMVEC,IPTVEC)                            C
C                 IA     -- NO. OF ELEMENTS TO BE SORTED IN ASCEN       C
C                           DING ORDER                                  C
C                 IPT    -- CONTAINS BEG. AND ENDING POINTER FOR        C
C                           IPTVEC ARRAY                                C
C                 NUMVEC -- ARRAY CONTAINING ELEMENTS TO BE SORTED      C
C                 IPTVEC -- ARRAY CONTAINING POINTERS FOR NUMVEC        C
C                                                                       C
C                           EXAMPLE:                                    C
C                                     IPT = 2                           C
C                                     IPT = 8                           C
C                           SMALLEST NO. = NUMVEC(IPT(1))               C
C                           NEXT NO.     = NUMVEC(IPTVEC(IPT(1))        C
C                                        ....                           C
C                           LAST NO.     = NUMVEC(IPTVEC(IPT(2))        C
C     ******************************************************************C
C
      DIMENSION  IPTVEC(  30  ),IPT(2),NUMVEC( 30  )
C
      ICT      = 0
      NOHST1= IA
      DO 50       I = 1,IA
      ICT      = ICT + 1
      IF ( ICT .GT. 1)        GO TO 10
      IPT(1) = 1
      IPT(2) = 1
      IF ( ICT .EQ. IPT(1) ) GO TO 60
   10 KA       = IPT(1)
      KB       = IPT(2)
      IF (NUMVEC(I) .EQ. 0)  NOHST1 = NOHST1 - 1
      IF (NUMVEC(I) .EQ. 0)    GO TO 60
      IF (NUMVEC(I) .LE. NUMVEC(KA))  GO TO 30
      IF (NUMVEC(I) .GE. NUMVEC(KB))  GO TO 40
   20 KB       = KA
      KA       = IPTVEC(KA)
      IF (NUMVEC(ICT) .LE. NUMVEC(KA))GO TO 50
      GO TO 20
   30 IPTVEC(ICT) = IPT(1)
      IPT(1)       = ICT
      KB           = IPT(2)
      GO TO 60
   40 IPT(2)       = ICT
      IPTVEC(KB)   = ICT
      GO TO 60
   50 IPTVEC(ICT) = KA
      IPTVEC(KB)   = ICT
   60 CONTINUE
      IA = NOHST1
      RETURN
      END
```

7. GRAYMAP PROCESSOR

```
FILE: GRAYMP   FORTRAN  A

      SUBROUTINE GRAYMP(ARRAY,TOP)                                          GRA00010
C                                                                           GRA00020
      IMPLICIT INTEGER(A-Z)                                                 GRA00030
C                                                                           GRA00040
      DIMENSION ARRAY(TOP),BINLEV(30,16)                                    GRA00050
      DIMENSION SYMBOL(16,2)                                                GRA00060
C                                                                           GRA00070
CCC   INCLUDE COMBK3.LIST                                                   GRA00080
CCC   INCLUDE COMBK4.LIST                                                   GRA00090
CCC   INCLUDE COMBK6.LIST                                                   GRA00100
      COMMON /GRCHLR/MAXFET,NOFEAT,NOFET2,FETVEC(30),
     *               FETVC2(30),FLDINF(6),INFMT,FILESV,NOHIST,
     *               HISVEC(30),NOFLD,          FLOPTS
     *,XSIZ,XLOW,XHGH,YSIZ
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                        COM00010
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                     COM00020
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                   COM00030
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             COM00010
     *              HISFIL,HISKEY,TRFORM,ERRPTP,ERRKEY,MAPUNT,NOFILE,       COM00020
     *    ,DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                   COM00030
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                        COM00040
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                  COM00050
     *  CROUNT,PRTUNT,RANDIO                                                COM00060
C$END                                                                       GRA00120
C                                                                           GRA00130
      COMMON /HISTOR/HF                                                     GRA00140
C                                                                           GRA00150
      CALL SETUPS(ARRAY,BINCNT,BINLEV,NUMBIN,SYMBOL,SYMCNT,SYMDIM)          GRA00160
      IF((BINCNT.EQ.1).OR.(HISKEY.EQ.1))  GO TO 1                           GRA00170
      FILHS=1                                                               GRA00180
      FLOTL=9000                                                            GRA00190
      TOTTL=9000                                                            GRA00200
      HF=1                                                                  GRA00210
      CALL HISTGM(ARRAY(FILHS),ARRAY(FLOTL),ARRAY(TOTTL))                   GRA00220
      CALL SETUPS(ARRAY,BINCNT,BINLEV,NUMBIN,SYMBOL,SYMCNT,SYMDIM)          GRA00230
    1 CALL PICT(ARRAY,BINLEV,NUMBIN,SYMBOL,SYMCNT,SYMDIM)                   GRA00240
      HF=0                                                                  GRA00250
      RETURN                                                                GRA00260
      END                                                                   GRA00270
```

FILE: HEADNG FORTRAN A

```
      SUBROUTINE HEADNG(TYPE,FETNUM,BINLEV,NUMBIN,FLDINP,                HEA00010
     *  SYMBOL,NSAMP,FET,SYMDIM,TCOL)                                    HEA00020
      IMPLICIT INTEGER(A-Z)                                            HEA00030
C                                                                      HEA00040
C-----------------------------------------------------------------------HEA00050
C                                                                      HEA00060
C     CALL.. CALL HEADNG(TYPE,FETNUM)                                  HEA00070
C                                                                      HEA00080
C     ARGS.. TYPE - REFERS TO COLUMN HEADING                           HEA00090
C             FETNUM - REFERS TO LOCATION OF FEATURE IN FETVC2 ARRAY   HEA00100
C                                                                      HEA00110
C     REQUIRES. COMMONS /INFORM/  /INFORS/                             HEA00120
C                       /GLOBAL/  /HELP/                               HEA00130
C                                                                      HEA00140
C     PURPOSE.. PRINTS OUT HEADING INFORMATION                         HEA00150
C                                                                      HEA00160
C     RETURNS NONE                                                     HEA00170
C                                                                      HEA00180
C-----------------------------------------------------------------------HEA00190
C                                                                      HEA00200
C     INCLUDE COMPR3.LIST                                              HEA00210
C     INCLUDE COMPR4.LIST                                              HEA00220
C     INCLUDE COMPR5.LIST
      COMMON /GLCHLK/MAXFET,NOFEAT,NOFET2,FETVEC(30),
     *               FETVC2(30),FLDINF(6),INFMT,FILESV,NOHIST,
     *               HISVEC(30),NOFLD,          FLOPTS
     *,XSIZ,XLO,XHGH,YSIZ
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                   COM00010
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                COM00020
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(4A))              COM00030
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,        COM00010
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,  COM00020
     *    DRUM30,DWMODS,MAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               COM00030
     *  ,NMSTUN,NMSTFI,SCTPUN,MAPFIL                                   COM00040
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTHFL,HISTFL,PCHUNT,             COM00050
     *  CRDUNT,PRTUNT,RANDIO                                           COM00060
CSEND                                                                  HEA00240
      DIMENSION SYMBOL(16,2),FET(1),FLDINP(7)                          HEA00250
      DIMENSION ICOL(3,110),BINLEV(30,16)                              HEA00260
      EQUIVALENCE (FLDINF(1),LINSTR),(FLDINF(2),LINEND),               HEA00270
     *            (FLDINF(3),LININC),(FLDINF(4),SAMSTR),               HEA00280
     *            (FLDINF(5),SAMEND),(FLDINF(6),SAMINC)                HEA00290
      IF(TYPE.EQ.2) GO TO 110                                          HEA00300
      IF(TCOL.EQ.1) GO TO 120                                          HEA00310
      J=FETNUM                                                         HEA00320
  100 FIRST=0                                                          HEA00330
      WRITE(4,101) FIRST,(BINLEV(J,MA),MA=1,NUMBIN)                    HEA00340
  101 FORMAT(//12,16(3X,I3))                                           HEA00350
      DO 102 IZ=1,4                                                    HEA00360
      WRITE(4,104)((SYMBOL(MA,1),SYMBOL(MA,1),SYMBOL(MA,1),SYMBOL(MA,1),HEA00370
     * SYMBOL(MA,1),SYMBOL(MA,1)),MA=1,NUMBIN)                         HEA00380
  104 FORMAT(2X,96A1)                                                  HEA00390
      IF(SYMDIM.EQ.1) GO TO 102                                        HEA00400
      WRITE(4,105)((SYMBOL(MA,2),SYMBOL(MA,2),SYMBOL(MA,2),SYMBOL(MA,2),HEA00410
     * SYMBOL(MA,2),SYMBOL(MA,2)),MA=1,NUMBIN)                         HEA00420
  105 FORMAT(1H+,1X,96A1)                                              HEA00430
  102 CONTINUE                                                         HEA00440
  120 CONTINUE                                                         HEA00450
      CALL SETMRG(66,0,66)                                             HEA00460
C CALCULATE AND PRINT SAMPLE NUMBERS                                   HEA00470
      JG=0                                                             HEA00480
      SS=FLDINP(4)                                                     HEA00490
      SE=FLDINP(5)                                                     HEA00500
      DO 106 I=SS,SE,SAMINC                                            HEA00510
      JG=JG+1                                                          HEA00520
      ICOL(1,JG)=I/100                                                 HEA00530
      ICOL(2,JG)=MOD(I,100)/10                                         HEA00540
      ICOL(3,JG)=MOD(I,10)                                             HEA00550
  106 CONTINUE                                                         HEA00560
  110 IF(TYPE.EQ.2) WRITE(6,111)                                       HEA00570
      DO 107 I=1,3                                                     HEA00580
  107 WRITE(6,108) (ICOL(I,J),J=1,NSAMP)                               HEA00590
  108 FORMAT(10X,110I1)                                                HEA00600
      WRITE(4,111)                                                     HEA00610
  111 FORMAT(1H0)                                                      HEA00620
      IF(TYPE.EQ.2) CALL SETMRG(66,4,62)                               HEA00630
      RETURN                                                           HEA00640
      END                                                              HEA00650
```

FILE: PICT FORTRAN A

```
      SUBROUTINE PICT(BUF,BINLEV,NUMBIN,SYMBOL,SYMCNT,SYMDIM)                PIC00010
C                                                                            PIC00020
      IMPLICIT INTEGER(A-Z)                                                  PIC00030
C                                                                            PIC00040
C-------------------------------------------------------------------------PIC00050
CCCCC                                                                        PIC00060
CCCCC  CALL.. CALL PICT(IDATA)                                               PIC00070
CCCCC                                                                        PIC00080
CCCCC  ARGS.. IDATA - SCANNER DATA                                           PIC00090
CCCCC                                                                        PIC00100
CCCCC          ROUTINES HEADNG  TAPHDR  FLDINF  LINERD  LAREAD               PIC00110
CCCCC                                                                        PIC00120
CCCCC  PURPOSE. PICTORIALLY DISPLAYS FEATURES REQUESTED                      PIC00130
CCCCC                                                                        PIC00140
CCCCC  RETURNS NONE                                                          PIC00150
CCCCC                                                                        PIC00160
C-------------------------------------------------------------------------PIC00170
CCCCC                                                                        PIC00180
CCCCC  INCLUDE COMBK6.LIST                                                   PIC00190
CCCCC  INCLUDE COMBK3.LIST                                                   PIC00200
C      INCLUDE COMBK4.LIST                                                   PIC00210
       COMMON /GRCBLK/MAXFET,NOFEAT,NOFET2,FETVEC(30),                       PIC00220
      *               FETVC2(30),FLDINF(6),INFMT,FILESV,NOHIST,              PIC00230
      *               HISVEC(30),NOFLD,          FLDPTS                      PIC00240
      *,XSIZ,XLOW,XHGH,YSIZ                                                  PIC00250
       DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                        PIC00260
       EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                     PIC00270
      2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                  PIC00280
       COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             PIC00290
      *              HISFIL,HISKEY,TRFORM,ERIPTR,EPPKEY,MAPUNT,NOFILE,       PIC00300
      *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    PIC00310
      *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                       PIC00320
      *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                 PIC00330
      *   CRDUNT,PRTUNT,RANDIO                                               PIC00340
C$END                                                                        PIC00350
C                                                                            PIC00360
       EQUIVALENCE (FLDINF(1),LINSTR),(FLDINF(2),LINEND),                    PIC00370
      *            (FLDINF(3),LININC),(FLDINF(4),SAMSTR),                   PIC00380
      *            (FLDINF(5),SAMEND),(FLDINF(6),SAMINC)                    PIC00390
C                                                                            PIC00400
       DIMENSION BINLEV(30,16)                                               PIC00410
       DIMENSION LCHAR(2,256),LIN(110),IDATA(12000),BUF(110,20),JSTAT(20)    PIC00420
       DIMENSION SYMBOL(16,2),FET(1),FLDINP(6)                               PIC00430
       DIMENSION VERTCS(2,11),FL(8)                                          PIC00440
C                                                                            PIC00450
       DATA BLANK/' '/                                                       PIC00460
       DATA OPAR/'('/,CPAR/')'/,COMMA/','/                                   PIC00470
C                                                                            PIC00480
C READ HEADER RECORD ON DATA TAPE                                            PIC00490
       CALL TAPHDR(DATAPE,DATFIL)                                            PIC00500
C READ FIELD DEFINITION CARDS                                                PIC00510
   20  RUNNO=LAREAD(FLDNAM,VERTCS,FLDINF,NC)                                 PIC00520
       IF(RUNNO.LE.0) GO TO 1                                                PIC00530
C CHECK TO SEE IF INFORMATION WILL FIT ON DRUM                               PIC00540
       NBUFS=20                                                              PIC00550
       PTS=(FLDINF(5)-FLDINF(4))/FLDINF(6)+1                                 PIC00560
       SPTS=PTS                                                              PIC00570
       LINES=(FLDINF(2)-FLDINF(1))/FLDINF(3)+1                               PIC00580
       NOFET4=NOFET2                                                         PIC00590
   27  NBUFSZ=PTS*NOFET4                                                     PIC00600
       TSAMP=LINES*NBUFSZ                                                    PIC00610
       IF(TSAMP.LE.DRMWDS) GO TO 26                                          PIC00620
C REDUCE NO. OF CHANNELS BY ONE,CHECK TO SEE IF FIELD WILL FIT               PIC00630
       NOFET4=NOFET4-1                                                       PIC00640
       GO TO 27                                                              PIC00650
   26  IF(NOFET2.NE.NOFET4) WRITE(6,28) NOFET4                               PIC00660
   28  FORMAT(' THE NO. OF CHANNELS FOR THIS FIELD HAS BEEN REDUCED TO',     PIC00670
      *I3,' SO ALL THE INFORMATION WILL FIT ON DRUM'/' MAKE ANOTHER RUN      PIC00680
      *TO GRAYMAP OTHER CHANNELS')                                           PIC00690
       CALL FLDINT(FLDINF,FETVC2,NOFET4)                                     PIC00700
       ADRES=DRUMAD                                                          PIC00710
       DO 29 I=1,LINES                                                       PIC00720
       CALL LINERD(IDATA,ENDTAP)                                             PIC00730
       IF(ENDTAP.NE.0) GO TO 30                                              PIC00740
       CALL RWRITE(ADRES,IDATA,NBUFSZ,LSTAT)                                 PIC00750
   31  IF(LSTAT.EQ.1) GO TO 31                                               PIC00760
       ADRES=ADRES+NBUFSZ                                                    PIC00770
       IF(ADRES.LE.DRUMAD+DRMWDS) GO TO 29                                   PIC00780
       WRITE(6,33)                                                           PIC00790
```

```
FILE: PICT     FORTRAN  A

   33 FORMAT(' FIELD TOO LARGE,TERMINATING')                                PIC00800
      CALL CMERR                                                            PIC00810
   29 CONTINUE                                                              PIC00820
   30 CONTINUE                                                              PIC00830
      FLDINP(1)=FLDINF(1)                                                   PIC00840
      FLDINP(2)=FLDINF(2)                                                   PIC00850
      FLDINP(3)=FLDINF(3)                                                   PIC00860
      FLDINP(6)=FLDINF(6)                                                   PIC00870
      NFET=1                                                                PIC00880
C FOR EACH FEATURE                                                          PIC00890
      DO 4 J=1,NOFET4                                                       PIC00900
      PTS=SPTS                                                              PIC00910
      NADRES=DRUMAD+(J-1)*SPTS                                              PIC00920
      FLDINP(4)=0                                                           PIC00930
      FLDINP(5)=0                                                           PIC00940
      FET(1)=FETVC2(J)                                                      PIC00950
      WRITE(6,HEAD)                                                         PIC00960
      MM=NC-1                                                               PIC00970
  300 FORMAT(T24,'SAMPLE  LINE     NO. OF'/                                 PIC00980
     *2X,'CHANNEL   FIELDNAME    INC    INC    VERTICES    VERTICES(SAMP   PIC00990
     *LE,LINE)'                                                             PIC01000
     */3X,I3,7X,A4,7X,I4,2X,I4,7X,I2,7X,5(A1,I4,A1,I4,A1,2X)/               PIC01010
     *T51,5(A1,I4,A1,I4,A1,2X))                                             PIC01020
      WRITE(6,300) FET(1),FLDNAM   ,SAMINC,LININC,MM,                       PIC01030
     *((OPAR,VERTCS  (1,NM),COMMA,VERTCS(2,NM),CPAR),NM=1,MM)               PIC01040
      TCOL =0                                                               PIC01050
C SET UP VALUE-SYMBOL TABLE                                                 PIC01060
      DO 9 JD=1,SYMDIM                                                      PIC01070
      I2=BINLEV(J,1)+1                                                      PIC01080
      DO 8 I3=1,I2                                                          PIC01090
    8 LCHAR(JD,I3)=SYMBOL(1,JD)                                             PIC01100
      DO 10 J1=2,NUMBIN                                                     PIC01110
      I1=BINLEV(J,J1-1)+2                                                   PIC01120
      I2=BINLEV(J,J1)+1                                                     PIC01130
      DO 10 I3=I1,I2                                                        PIC01140
   10 LCHAR(JD,I3)=SYMBOL(J1,JD)                                            PIC01150
    9 CONTINUE                                                              PIC01160
      FLDINP(4)=FLDINF(4)                                                   PIC01170
      AD=0                                                                  PIC01180
    7 PPTS=PTS                                                              PIC01190
      ADRES=NADRES+AD                                                       PIC01200
      IF(PTS.LE.110) GO TO 5                                                PIC01210
      FLDINP(5)=FLDINP(4)+109*FLDINP(6)                                     PIC01220
      GO TO 6                                                               PIC01230
    5 FLDINP(5)=FLDINF(5)                                                   PIC01240
    6 PTS=PTS-110                                                           PIC01250
C INITIALIZE TAPE READING FOR THIS FIELD                                    PIC01260
      CALL FLDINT(FLDINP,FET,NFET)                                          PIC01270
      NSAMP=(FLDINP(5)-FLDINP(4))/FLDINP(6)+1                               PIC01280
      IF(NSAMP.GE.110) GO TO 101                                            PIC01290
      IF(NSAMP.EQ.PPTS) GO TO 101                                           PIC01300
      WRITE(6,100) FLDINP(5)                                                PIC01310
      PTS=0                                                                 PIC01320
      FLDINF(5)=FLDINP(5)                                                   PIC01330
  100 FORMAT(' YOU HAVE ASKED FOR TOO MANY SAMPLES, THE LAST SAMPLE IS',I5, PIC01340
     * I5)                                                                  PIC01350
  101 CONTINUE                                                              PIC01360
      LINES=(FLDINP(2)-FLDINP(1))/FLDINP(3)+1                               PIC01370
      TYPE=1                                                                PIC01380
      FETNUM=J                                                              PIC01390
      CALL HEADNG(TYPE,FETNUM,BINLEV,NUMBIN,FLDINP,SYMBOL,                  PIC01400
     *  NSAMP,FET,SYMDIM,TCOL)                                              PIC01410
      TCOL=1                                                                PIC01420
      LINCNT=0                                                              PIC01430
C READ AND FILL 20 BUFFERS                                                  PIC01440
      DO 44 JA=1,NBUFS                                                      PIC01450
      CALL RREAD(ADRES,BUF(1,JA),NSAMP,JSTAT(JA))                           PIC01460
      ADRES=ADRES+NBUFSZ                                                    PIC01470
      LINCNT=LINCNT+1                                                       PIC01480
   44 CONTINUE                                                              PIC01490
      LINE=LINSTR                                                           PIC01500
      IBUF=1                                                                PIC01510
C FINISHED READING                                                          PIC01520
   36 IF(JSTAT(IBUF).EQ.1) GO TO 36                                         PIC01530
      DO 200 MM=1,NSAMP                                                     PIC01540
  200 LIN(MM)=BLANK                                                         PIC01550
      CALL FDLINT(VERTCS,NC,FL,LINE,NS,JJ)                                  PIC01560
      DO 14 JD=1,SYMDIM                                                     PIC01570
      L=1                                                                   PIC01580
```

FILE: PICT FORTRAN A

```
      DO 201 IA=1,NSAMP                                                 PIC01590
      KA=(IA-1)*SAMINC+FLDINP(4)                                        PIC01600
      DO 202 JK=L,JJ,2                                                  PIC01610
      IF(KA.LT.FL(JK)) GO TO 201                                        PIC01620
      IF(KA.GT.FL(JK+1)) GO TO 203                                      PIC01630
      IDT=BUF(IA,IBUF)+1                                                PIC01640
      LIN(IA)=LCHAR(JD,IDT)                                             PIC01650
      GO TO 201                                                         PIC01660
  203 L=L+2                                                             PIC01670
      IF(L.GT.JJ) GO TO 205                                             PIC01680
  202 CONTINUE                                                          PIC01690
  201 CONTINUE                                                          PIC01700
  205 CONTINUE                                                          PIC01710
      IF(JD.NE.2) GO TO 21                                              PIC01720
      WRITE(6,17) LINE,(LIN(JK),JK=1,NSAMP)                             PIC01730
      WRITE(6,102) LINE                                                 PIC01740
  102 FORMAT(1H+,T122,I5)                                               PIC01750
      GO TO 14                                                          PIC01760
   17 FORMAT(1H+,I5,4X,110A1)                                           PIC01770
   21 WRITE(6,15) LINE,(LIN(JK),JK=1,NSAMP)                             PIC01780
   15 FORMAT(I6,4X,110A1)                                               PIC01790
   14 CONTINUE                                                          PIC01800
      LINE=LINE+LININC                                                  PIC01810
      LINCNT=LINCNT+1                                                   PIC01820
      IF(LINCNT.GT.LINES) GO TO 37                                      PIC01830
      CALL RREAD(ADRES,BUF(1,IBUF),NSAMP,JSTAT(IBUF))                   PIC01840
      ADRES=ADRES+NBUFSZ                                                PIC01850
   37 IBUF=IBUF+1                                                       PIC01860
      IF(IBUF.GT.NBUFS) IBUF=1                                          PIC01870
      IF(LINE.LE.LINEND) GO TO 36                                       PIC01880
      TYPE=2                                                            PIC01890
      CALL HEADNG(TYPE,FETNUM,BINLEV,NUMBIN,FLDINP,SYMBOL,             PIC01900
     *  NSAMP,FET,SYMDIM,TCOL)                                          PIC01910
      FLDINP(4)=FLDINP(6)+FLDINP(5)                                     PIC01920
      AD=AD+110                                                         PIC01930
      IF(PTS.GT.0) GO TO 7                                              PIC01940
    4 CONTINUE                                                          PIC01950
      GO TO 20                                                          PIC01960
    1 IF(RUNNO.EQ.0) RETURN                                             PIC01970
      GO TO 20                                                          PIC01980
      END                                                               PIC01990
```

FILE: SETUP6 FORTRAN A

```
      SUBROUTINE SETUP6(FILHIS,BINCNT,BINLEV,NUMBIN,SYMBOL,SYMCNT,      SET00010
     * SYMD(M)                                                          SET00020
C                                                                       SET00030
      IMPLICIT INTEGER(A-Z)                                             SET00040
C                                                                       SET00050
C-----------------------------------------------------------------------SET00060
C                                                                       SET00070
C     CALL.. CALL SETUP6(FILHIS)                                        SET00080
C                                                                       SET00090
C     ARGS.. FILHIS - HISTOGRAM DATA ARRAY                              SET00100
C                                                                       SET00110
C     REQUIRES COMMONS /INFORM/   /INFORS/                              SET00120
C                      /GLOBAL/   /HELP/                                SET00130
C              ROUTINES NXTCHR FIND12 NUMBER                            SET00140
C                                                                       SET00150
C     PURPOSE.. READS AND ANALYSES CONTROL CARDS FOR 'GRAYMAP' STEP     SET00160
C                                                                       SET00170
C     RETURNS.. SUPERVISOR INFORMATION                                  SET00180
C                                                                       SET00190
C-----------------------------------------------------------------------SET00200
C                                                                       SET00210
C     INCLUDE COMBK3.LIST                                               SET00220
C     INCLUDE COMBK4.LIST                                               SET00230
C     INCLUDE COMBK6.LIST                                               SET00240
      COMMON /GRCBLK/MAXFET,NOFEAT,NOFET2,FETVEC(30),                   SET00250
     *               FETVC2(30),FLDINF(6),INFMT,FILESV,NOHIST,          SET00260
     *               HISVEC(30),NOFLD,           FLDPTS                SET00270
     *,XSIZ,XLOW,XHGH,YSIZ                                              SET00280
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                    SET00290
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                 SET00300
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               SET00310
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         SET00320
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,   SET00330
     *    DRUMAD,DRMSDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                SET00340
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                   SET00350
     *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,             SET00360
     *   CRDUNT,PRTUNT,RANDIO                                           SET00370
CSEND                                                                   SET00380
C                                                                       SET00390
      COMMON /HISTOR/HF                                                 SET00400
C                                                                       SET00410
      EQUIVALENCE (FLDINF(1),LINSTR),(FLDINF(2),LINEND),                SET00420
     *            (FLDINF(3),LININC),(FLDINF(4),SAMSTR),                SET00430
     *            (FLDINF(5),SAMEND),(FLDINF(6),SAMINC)                 SET00440
C                                                                       SET00450
      DIMENSION FRVEC1(3),FRVEC2(3),NUMVEC(30)                          SET00460
      DIMENSION SYMBOL(15,2),ACARD(20)                                  SET00470
      DIMENSION CINDEX(10),BINLEV(30,16),EQUVEC(2),                     SET00480
     *SINVEC(3),CARD2(62),HGTPT(30),FILHIS(NOFEAT,256),COMMA(2)         SET00490
C                                                                       SET00500
      EQUIVALENCE (SINVEC(3),EQUAL)                                     SET00510
C                                                                       SET00520
      DATA CINDEX/'CHAN','BINL','SYMB','FORM','HED1',                   SET00530
     * 'HED2','COMM','DATE','END','DATA'/                               SET00540
      DATA EQUVEC/1,'='/                                                SET00550
      DATA SINVEC/2,',','='/,CINMAX/10/,                                SET00560
     * BLANK/' '/,COMMA/1,','/,FRVEC1/2,'I','O'/,                       SET00570
     *FRVEC2/2,'U','L'/                                                 SET00580
      DATA MRCD/'M'/,ORCD/'O'/,XRCD/'X'/,DLRBCD/'$'/,DOTBCD/'.'/,       SET00590
     1 EQLRCD/'='/,MNSRCD/'-'/,SLHBCD/'/'/,STRACD/'*'/,BLKBCD/' '/ ,    SET00600
     2 FRCD/'F'/,ORCD/'U'/                                              SET00610
C                                                                       SET00620
      IK=1                                                              SET00630
      NOFET2=0                                                          SET00640
      MAXFET=30                                                         SET00650
      IF(HF.EQ.1) GO TO 80                                              SET00660
      SYMDIM=0                                                          SET00670
      NUMBIN=0                                                          SET00680
      BINCNT=0                                                          SET00690
C                                                                       SET00700
C   SETUP REREAD BUFFER                                                 SET00710
C                                                                       SET00720
      CALL REREAD(30,80)                                                SET00730
   14 COL=0                                                             SET00740
C   PUT NEXT CARD IN BUFFER                                             SET00750
      READ(21,200)(ACARD(I),I=1,20)                                     SET00760
  200 FORMAT(20A4)                                                      SET00770
      WRITE(30,200)(ACARD(I),I=1,20)                                    SET00780
      REWIND 30                                                         SET00790
```

FILE: SETUP6 FORTRAN A

```
      GO TO 14                                                          SET01590
C  HED1 CARD                                                           SET01600
    8 READ(30,25) HED1                                                 SET01610
      REWIND 30                                                        SET01620
      GO TO 14                                                         SET01630
C HED2 CARD                                                            SET01640
    9 READ(30,25) HED2                                                 SET01650
      REWIND 30                                                        SET01660
      GO TO 14                                                         SET01670
   25 FORMAT(10X,15A4)                                                 SET01680
C COMMENT CARD                                                         SET01690
   10 READ(30,25) COMENT                                               SET01700
      REWIND 30                                                        SET01710
      GO TO 14                                                         SET01720
C DATE CARD                                                            SET01730
   11 M=NXTCHR(CARD2,COL)                                              SET01740
      IF(M.EQ.BLANK) GO TO 14                                          SET01750
      READ(30,25) DATE                                                 SET01760
      REWIND 30                                                        SET01770
      GO TO 14                                                         SET01780
C                                                                      SET01790
C DATAFILE POSITIONING CARD                                            SET01800
C                                                                      SET01810
 1701 M=NXTCHR(CARD2,COL)                                              SET01820
      IF(M.EQ.BLANK) GO TO 14                                          SET01830
      IF (M .EQ. UBCD) GO TO 1702                                      SET01840
      IF (M .EQ. FBCD) GO TO 1703                                      SET01850
 1705 WRITE(6,1704)                                                    SET01860
 1704 FORMAT(' ERROR ON DATA FILE CARD ')                              SET01870
      GO TO 14                                                         SET01880
 1702 J=FIND12(CARD2,COL,EQUVEC)                                       SET01890
      IF(J.EQ.-1) GO TO 1705                                           SET01900
      M=NUMBER(CARD2,COL,DATAPE,ZERO)                                  SET01910
      COL=COL-1                                                        SET01920
      GO TO 1701                                                       SET01930
 1703 J=FIND12(CARD2,COL,EQUVEC)                                       SET01940
      IF(J.EQ.-1) GO TO 1705                                           SET01950
      FILNO=NUMBER(CARD2,COL,DATFIL,FILNO)                             SET01960
      DATFIL=DATFIL-1                                                  SET01970
      COL=COL-1                                                        SET01980
      GO TO 1701                                                       SET01990
C *END* CARD                                                           SET02000
   12 IF(MSYM.EQ.1) GO TO 26                                           SET02010
C DEFAULT SYMBOLS                                                      SET02020
      SYMCNT=10                                                        SET02030
      SYMDIM=2                                                         SET02040
      SYMBOL(1,1) = MBCD                                               SET02050
      SYMBOL(1,2) = DLBBCD                                             SET02060
      SYMBOL(2,1) = XBCD                                               SET02070
      SYMBOL(2,2) = EQLBCD                                             SET02080
      SYMBOL(3,1) = ORCD                                               SET02090
      SYMBOL(3,2) = MNSBCD                                             SET02100
      SYMBOL(4,1) = ORCD                                               SET02110
      SYMBOL(4,2) = ORCD                                               SET02120
      SYMBOL(5,1) = STRBCD                                             SET02130
      SYMBOL(5,2) = MNSBCD                                             SET02140
      SYMBOL(6,1) = EQLHCD                                             SET02150
      SYMBOL(6,2) = EQLBCD                                             SET02160
      SYMBOL(7,1) = DOTBCD                                             SET02170
      SYMBOL(7,2) = DOTBCD                                             SET02180
      SYMBOL(8,1) = MNSHCD                                             SET02190
      SYMBOL(8,2) = MNSBCD                                             SET02200
      SYMBOL(9,1) = SLHBCD                                             SET02210
      SYMBOL(9,2) = SLHBCD                                             SET02220
      SYMBOL(10,1) = BLKHCD                                            SET02230
      SYMBOL(10,2) = BLKBCD                                            SET02240
C CHECK TO SEE IF BINLEVELS INPUT                                      SET02250
   26 IF(BINCNT.EQ.1) GO TO 27                                         SET02260
      NUMBIN=SYMCNT                                                    SET02270
      IF(HISFIL.NE.13) STOP 2                                          SET02280
C READ HISTOGRAM AND CALCULATE BINLEVELS                               SET02290
      IF(HISKEY.EQ.1) GO TO 80                                         SET02300
      NOFEAT=NOFET2                                                    SET02310
      DO 81 I7=1,NOFET2                                                SET02320
   81 FETVEC(IZ)=FETVC2(IZ)                                            SET02330
      RETURN                                                           SET02340
   80 READ(HISFIL)  NOFEAT,(FETVEC(I),I=1,NOFEAT)                      SET02350
      READ(HISFIL) ((FILHIS(I,J),J=1,256),I=1,NOFEAT)                  SET02360
      REWIND HISFIL                                                    SET02370
```

FILE: SETUP6 FORTRAN A

```
      DO 44 I=1,NOFEAT                                                  SET02380
   44 HGTPT(I)=0                                                        SET02390
      DO 45 I=1,NOFEAT                                                  SET02400
      DO 45 J=1,256                                                     SET02410
   45 HGTPT(I)=HGTPT(I)+FILHIS(I,J)                                     SET02420
C IF FEATURES CARD NOT INPUT,FETVC2 ARRAY= FETVEC ARRAY,NOFET2=NOFEAT   SET02430
      IF(NOFET2.NE.0) GO TO 60                                          SET02440
      NOFET2=NOFEAT                                                     SET02450
      DO 61 I=1,NOFEAT                                                  SET02460
   61 FETVC2(I)=FETVEC(I)                                               SET02470
      GO TO 100                                                         SET02480
   60 IK=1                                                              SET02490
      KT=NOFET2                                                         SET02500
  102 DO 28 I=1,KT                                                      SET02510
      DO 29 J=1,NOFEAT                                                  SET02520
C CHECK TO SEE IF FEATURES HISTOGRAMMED                                 SET02530
      IF(FETVC2(I).NE.FETVEC(J)) GO TO 29                               SET02540
      GO TO 28                                                          SET02550
   29 CONTINUE                                                          SET02560
      WRITE(6,30) FETVC2(I)                                             SET02570
   30 FORMAT(1X,'THIS CHANNEL IS NOT HISTOGRAMMED',I3//)                SET02580
      NOFET2=NOFET2-1                                                   SET02590
      IF(I.GT.KT) GO TO 100                                             SET02600
      IK=I+1                                                            SET02610
      DO 101 II=IK,KT                                                   SET02620
  101 FETVC2(II-1)=FETVC2(II)                                           SET02630
      KT=NOFET2                                                         SET02640
      IK=I                                                              SET02650
      GO TO 102                                                         SET02660
   28 CONTINUE                                                          SET02670
  100 DO 103 I=1,NOFET2                                                 SET02680
      DO 104 J=1,NOFEAT                                                 SET02690
      IF(FETVC2(I).NE.FETVEC(J)) GO TO 104                              SET02700
      GO TO 32                                                          SET02710
  104 CONTINUE                                                          SET02720
C COMPUTE BINLEVELS FOR FEATURE                                         SET02730
   32 L=HGTPT(J)/NUMBIN                                                 SET02740
      LNUM=NUMBIN-1                                                     SET02750
      M=L                                                               SET02760
      N=0                                                               SET02770
      K=0                                                               SET02780
      DO 33 JJ=1,LNUM                                                   SET02790
   34 K=K+1                                                             SET02800
      N=N+FILHIS(J,K)                                                   SET02810
      IF(N.LT.M) GO TO 34                                               SET02820
      M=M+L                                                             SET02830
      BINLEV(I,JJ)=K-1                                                  SET02840
      IF(JJ.NE.1) GO TO 111                                             SET02850
      IF(K.EQ.1) BINLEV(I,1)=1                                          SET02860
      GO TO 33                                                          SET02870
  111 IF(BINLEV(I,JJ).EQ.BINLEV(I,JJ-1)) BINLEV(I,JJ)=BINLEV(I,JJ-1)+1 SET02880
   33 CONTINUE                                                          SET02890
      BINLEV(I,NUMBIN)=255                                              SET02900
  103 CONTINUE                                                          SET02910
   27 CONTINUE                                                          SET02920
C PRINT OUT SETUP6 CARDS                                                SET02930
      WRITE(6,40) (FETVC2(I),I=1,NOFET2)                                SET02940
   40 FORMAT(1X,'SUPERVISOR INFORMATION FOR GRAYMAP'/1X,'CHANNELS GRAYMASET02950
     *PPED ARE',36I4)                                                   SET02960
C WRITE OUT SYMBOLS                                                     SET02970
      WRITE(6,41) (SYMBOL(KZ,1),KZ=1,SYMCNT)                           SET02980
   41 FORMAT(' SYMBOLS USED IN GRAYMAP ARE',16(2X,A1))                  SET02990
      IF(SYMDIM.EQ.2) WRITE(6,42) (SYMBOL(KZ,2),KZ=1,SYMCNT)            SET03000
   42 FORMAT(1H+,27X,16(2X,A1))                                         SET03010
      RETURN                                                            SET03020
      END                                                               SET03030
```

8. STAT PROCESSOR

```
FILE: STAT     FORTRAN  A

      SUBROUTINE STAT(ARRAY,TOP)                                                STA00010
CI                                                                             ISTA00020
CI    PURPOSE.. COORDINATES THE VARIOUS ROUTINES                               ISTA00030
CI              FOR 'STATISTICS' STEP                                          ISTA00040
C                                                                              STA00050
      IMPLICIT INTEGER (A-H,O-Z)                                               STA00060
      DOUBLE PRECISION ARRAY(1500)                                             STA00070
C                                                                              STA00080
      DIMENSION KEPPTS(60)                                                     STA00090
C                                                                              STA00100
C     INCLUDE COMMON8,LIST                                                     STA00110
C STAT COMMON BLOCK                                                            STA00120
      COMMON /STBASE/SUBSV1,SUBMN1,SURVR1,SURSD1,SURCL1,SAVER1,HSTAL1,         STA00130
     *SPEC1,COVAR1,AVAR1,CLSID1,FLMEN1,FLVAR1,HFTAL1,FLDSV1                    STA00140
      COMMON /STCBLK/       MAXFET,MAXCLS,MAXFLD,NOFEAT,NOFET2,              STA00150
     *VARSIZ,NOSPEC,NOHIST,SPCBAS,IBLOCK(30),FETVEC(30),                       STA00160
     *FETVC2(30),HISVEC(30),NOFLD,NOCLS,                                       STA00170
     *   FLDINF(6),FLDPTS,CLSPTS,XSIZ,XHGH,XLOW,YSIZ                           STA00180
C COMMON BLOCK  STBASE  CONTAINS THE BASE ADDRESSES FOR THE STATISTICS         STA00190
C STORED IN 'ARRAY'                                                            STA00200
C                                                                              STA00210
C     SUBSV1 - BASE ADDRESS IN 'ARRAY' FOR SUBCLASS INFORMATION                STA00220
C              (5*SUBNO)                                                       STA00230
C                  FOR EACH SUBCLASS INDEX  1-CLASS NUMBER                     STA00240
C                                           2-STARTING FIELD NUMBER            STA00250
C                                           3-ENDING FIELD NUMBER              STA00260
C                                           4-SUBCLASS NAME                    STA00270
C     SUBMN1 - BASE ADDRESS IN 'ARRAY' FOR SUBCLASS MEANS                      STA00280
C     SURVR1 - BASE ADDRESS IN 'ARRAY' FOR SUBCLASS VARIANCES                  STA00290
C     SURSD1 - BASE ADDRESS IN 'ARRAY' FOR SUBCLASS NAMES                      STA00300
C     SURCL1 - BASE ADDRESS IN 'ARRAY' FOR CLASS NUMBER                        STA00310
C     SAVER1 - BASE ADDRESS IN 'ARRAY' FOR TRAINING FIELD VERTICES             STA00320
C     HSTAL1 - BASE ADDRESS IN 'ARRAY' FOR SUBCLASS HISTOGRAM TOTALS           STA00330
C     SPEC1  - BASE ADDRESS IN 'ARRAY' FOR SPECTOGRAM INFORMATION              STA00340
C              (5*NOSPEC)                                                      STA00350
C     COVAR1 - BASE ADDRESS IN 'ARRAY' FOR FIELD COVARIANCES                   STA00360
C     AVAR1  - BASE ADDRESS IN 'ARRAY' FOR FIELD MEANS                         STA00370
C     CLSID1 - BASE ADDRESS IN 'ARRAY' FOR CLASS NAMES                         STA00380
C     FLMEN1 - BASE ADDRESS IN 'ARRAY' FOR FIELD MEANS                         STA00390
C     FLVAR1 - BASE ADDRESS IN 'ARRAY' FOR FIELD VARIANCES                     STA00400
C     HFTAL1 - BASE ADDRESS IN 'ARRAY' FOR FIELD HISTOGRAM TOTALS              STA00410
C     FLDSV1 - BASE ADDRESS IN 'ARRAY' FOR FIELD INFOPMATION (10*MAXFLD)       STA00420
      CONTINUE                                                                 STA00430
C                 FOR EACH FIELD  INDEX  1-FIELD NAME                        STA00440
C                                        2-CLASS NUMBER                        STA00450
C                                        3-SUBCLASS NUMBER                     STA00460
C                                        4-NUMBER OF VERTICES                  STA00470
C                                        5-STARTING LINE NUMBER                STA00480
C                                        6-ENDING LINE NUMBER                  STA00490
C                                        7-STARTING SAMPLE NUMBER              STA00500
C                                        8-ENDING SAMPLE NUMBER                STA00510
C                                        9-LINE INCREMENT                      STA00520
C                                        10-SAMPLE INCREMENT                   STA00530
      CONTINUE                                                                 STA00540
C COMMON BLOCK  STCBLK  CONTAINS INFORMATION NEEDED BY ROUTINES IN STAT        STA00550
C     MAXFET- MAXIMUM NUMBER OF CHANNELS                                       STA00560
C     MAXCLS - MAXIMUM NUMBER OF CLASSES                                       STA00570
C     MAXFLD - MAXIMUM NUMBER OF FIELDS                                        STA00580
C     NOFEAT - NUMBER OF CHANNELS REQUESTED                                    STA00590
C     VARSIZ - SIZE OF EACH COVARIANCE MATRIX (NOFEAT*(NOFEAT+1)/2             STA00600
C     NOSPEC - NUMBER OF GROUPS OF SUBCLASSES TO PLOT                          STA00610
C     NOHIST - NUMBER OF CHANNELS TO HISTOGRAM                                 STA00620
C     SPCBAS - MINIMUM RADIANCE VALUE ON Y AXIS OF SPECTRAL PLOT               STA00630
C     IBLOCK - ARRAY CONTAINING TRIGGERS TO CERTAIN OPTIONS IN STAT            STA00640
C     FETVEC - ARRAY OF CHANNELS SELECTED                                      STA00650
C     HISVEC - ARRAY OF CHANNELS TO HISTOGRAM                                  STA00660
C     NOFLD - NUMBER OF FIELDS                                                 STA00670
C     NOCLS - NUMBER OF CLASSES                                                STA00680
C     FLDINF - FIELD INFORMATION ARRAY                                         STA00690
C     FLDPTS - NUMBER OF POINTS IN FIELD                                       STA00700
C     CLSPTS - NUMBER OF POINTS IN CLASS                                       STA00710
C     XSIZ - XHGH-XLOW  =101                                                   STA00720
C     XLOW - MINIMUM X VALUE TO BE HISTOGRAMMED  =0                            STA00730
C     XHGH -MAXIMUM X VALUE TO BE HISTOGRAMMED   =255                          STA00740
C     YSIZ - HEIGHT OF Y AXIS IN HISTOGRAM  =15                                STA00750
C$END                                                                          STA00760
C                                                                              STA00770
      CALL SETUP1(ARRAY,TOP,MAXSUB)                                            STA00780
      CALL LEARN(ARRAY(SPEC1),ARRAY(COVAR1),ARRAY(AVAR1),                      STA00790
```

```
FILE: STAT     FORTRAN  A

     *            ARRAY(CLSID1),ARRAY(SUBSV1),ARRAY(FLMEN1),ARRAY(FLVAR1)STA00800
     *,           ARRAY(SUBMN1),ARRAY(SUBVR1),ARRAY(SUBSD1),ARRAY(SUBCL1)STA00810
     *,           ARRAY(MFTAL1),ARRAY(MSTAL1),ARRAY(FLOSV1),             STA00820
     *            ARRAY(SAVER1),KEPPTS,MAXSUB)                           STA00830
      WRITE (6,2)                                                       STA00840
   2  FORMAT(////////// 2X,'*** $STAT - COMPLETED ***' //////)          STA00850
      RETURN                                                            STA00860
       END                                                              STA00870
```

FILE: CLSSPC FORTRAN A

```
      SUBROUTINE CLSSPC(MEAN,SUBSTD,IDVEC,PTRVEC,PLOT,                  CLS00010
     *TITLE,NOFEAT,FETVEC,SPCBAS)                                       CLS00020
C                                                                       CLS00030
      IMPLICIT INTEGER (A-H,O-Z)                                        CLS00040
      LOGICAL OVRFLG                                                    CLS00050
      REAL MEAN(1),SUBSTD(1),BIAS,INCR,MENI,DEVI                        CLS00060
      DOUBLE PRECISION DEV(1),DMEAN(1)                                  CLS00070
C                                                                       CLS00080
      DIMENSION PLOT(4,NOFEAT,49),PTRVEC(5)                             CLS00090
      DIMENSION FETVEC(30)                                              CLS00100
     *, SYMVEC(4),TAB(12),ERRLIN(7)                                     CLS00110
C                                                                       CLS00120
C     INCLUDE COMRK6.LIST                                               CLS00130
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         CLS00140
     *          HISFIL,HISKEY,THFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,       CLS00150
     *    DRUMAD,DRMSOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                CLS00160
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    CLS00170
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTAFL,HISTFL,PCHUNT,              CLS00180
     *  CRDUNT,PRTUNT,RANDIO                                            CLS00190
C$END                                                                   CLS00200
C                                                                       CLS00210
      DATA SYMVEC/'$','+','=','*'/,  INCR/3.0/, NOLINE/49/,             CLS00220
     1     BLANK /' '/                                                  CLS00230
C                                                                       CLS00240
      DATA  TAB /'12','20','28','36','44','52'                          CLS00250
     1          ,'60','68','76','84','92','99'/                         CLS00260
      DATA ERRLIN/'(1H+',',T  ','     ',',1H(',',A1,',',1H),',',I3) '/  CLS00270
C                                                                       CLS00280
      DATA  DASH/'----'/                                                CLS00290
C                                                                       CLS00300
CCC                                                                     CLS00310
CC     SPEC INIZ                                                        CLS00320
CC                                                                      CLS00330
      WRITE(6,HEAD)                                                     CLS00340
      WRITE(6,10021)                                                    CLS00350
10021 FORMAT(     34X,'SPECTRAL PLOT (MEAN,PLUS AND MINUS ONE STD. DEV. )CLS00360
     1 FOR:' / )                                                        CLS00370
      WRITE(6,1002)  TITLE   ,(DASH,I=1,2)                              CLS00380
 1002 FORMAT(44X,'TRAINING SUBCLASS  ',A4/44X,4A4,A2/)                  CLS00390
      WRITE(6,2002)                                                     CLS00400
      WRITE(6,9002) SYMVEC(2),IDVEC                                     CLS00410
      GO TO 250                                                         CLS00420
CCC   ------------------------------------------------------------------CLS00430
CC                                                                      CLS00440
      ENTRY FLDSPC(DMEAN,DEV,IDVEC,PTRVEC,PLOT,MEAN,SUBSTD,FLDNAM,      CLS00450
     *NOFEAT,FETVEC,SPCBAS)                                             CLS00460
      WRITE(6,HEAD)                                                     CLS00470
      WRITE(6,10021)                                                    CLS00480
      WRITE(6,1004) IDVEC,(DASH,I=1,4)                                  CLS00490
 1004 FORMAT(47X,'TRAINING FIELD  ',A4/46X,4A6/)                        CLS00500
      WRITE(6,20021)                                                    CLS00510
20021 FORMAT( / 47X, 'PLOT LEGEND:' / 47X, '------------' )             CLS00520
      WRITE(6,2002) SYMVEC(2),IDVEC                                     CLS00530
 2002 FORMAT(50X,A1,' = FIELD  ',A4)                                    CLS00540
C                                                                       CLS00550
      DO 100 J=1,NOFEAT                                                 CLS00560
  100 MEAN(J) = DMEAN(J)                                                CLS00570
      DO 399 I=1,NOFEAT                                                 CLS00580
  399 SUBSTD(I)=DEV(I)                                                  CLS00590
  250 JPSTR = 2                                                         CLS00600
      JPLOT = 2                                                         CLS00610
      JBIAS = 1                                                         CLS00620
C                                                                       CLS00630
CC     COMBINED INIZ                                                    CLS00640
C                                                                       CLS00650
  300 BIAS = SPCBAS                                                     CLS00660
      WRITE(6,4002)BLANK                                                CLS00670
      CNT = BIAS                                                        CLS00680
      OVRFLW = 0                                                        CLS00690
      OVRFLG = .TRUE.                                                   CLS00700
      DO 350 J=1,NOFEAT                                                 CLS00710
      DO 350 I=1,4                                                      CLS00720
      DO 350 K=1,49                                                     CLS00730
  350 PLOT(I,J,K) = BLANK                                               CLS00740
      LOC=1                                                             CLS00750
      ISTOP=NOFEAT                                                      CLS00760
      IF(NOFEAT.GT.12)ISTOP=12                                          CLS00770
                                                                        CLS00780
                                                                        CLS00790
```

```
FILE: CLSSPC   FORTRAN  A

C      SET UP 'PLOT' MATRIX                                                    CLS00800
C                                                                              CLS00810
       JKSV=0                                                                  CLS00820
  400  DO 690 JP=JPSTR,JPLOT                                                   CLS00830
       MENBAS = (PTRVEC(JP-JBIAS)-1)*NOFEAT                                    CLS00840
       IF( OVRFLG) WRITE(6,4002)                                               CLS00850
 4002  FORMAT(A4)                                                              CLS00860
       OVRFLG = .FALSE.                                                        CLS00870
       JK=JKSV                                                                 CLS00880
       JF = 0                                                                  CLS00890
       DO 500 JFEAT=LOC,ISTOP                                                  CLS00900
            JK = JK+JFEAT                                                      CLS00910
            MENI = MEAN(MENBAS+JFEAT)                                          CLS00920
       DEVI=SUBSTD(MENBAS+JFEAT)                                               CLS00930
            MENLOW = (MENI-DEVI-BIAS)/INCR+0.5                                 CLS00940
            MENHGH = (MENI+DEVI-BIAS)/INCR+0.5                                 CLS00950
            IF( MENLOW .GE. 1) GO TO 430                                       CLS00960
       OVRFLW = MENI-DEVI+0.5                                                  CLS00970
            MENLOW = 1                                                         CLS00980
  430       IF(MENHGH .LE. NOLINE)  GO TO 450                                  CLS00990
       OVRFLW = MENI+DEVI+0.5                                                  CLS01000
            MENHGH = NOLINE                                                    CLS01010
  450       DO 490 J =MENLOW,MENHGH                                            CLS01020
  490          PLOT(JP,JFEAT,J) = SYMVEC(JP)                                   CLS01030
       JF = JF + 1                                                             CLS01040
       IF(OVRFLW .EQ. 0 ) GOTO 500                                             CLS01050
C*****JF = JF + 1 THIS STATEMENT MOVED UP ONE LINE **********                  CLS01060
       ERRLIN(3)=TAR(JF)                                                       CLS01070
       WRITE(6,ERRLIN)  SYMVEC(JP),OVRFLW                                      CLS01080
       OVRFLG = .TRUE.                                                         CLS01090
       OVRFLW = 0                                                              CLS01100
  500     CONTINUE                                                             CLS01110
  690   CONTINUE                                                               CLS01120
C                                                                              CLS01130
C      PRINT OUT 'PLOT' MATRIX                                                 CLS01140
       WRITE(6,3004) CNT,CNT                                                   CLS01150
 3004  FORMAT( 4X,I3, 2X, 'I------I', 12('-------I'), 2X, I3)                  CLS01160
       CNT = CNT+INCR                                                          CLS01170
  700  DO 790 K= 1,NOLINE                                                      CLS01180
       WRITE(6,7002)CNT,((PLOT(I,J,K),I=1,4),J=LOC,ISTOP)                      CLS01190
       WRITE(6,7003)CNT                                                        CLS01200
 7002  FORMAT(4X,I3,2X,'I ',12(4X,4A1))                                        CLS01210
 7003  FORMAT('+',111X,'I',2X,I3)                                              CLS01220
          CNT = CNT+INCR                                                       CLS01230
  790   CONTINUE                                                               CLS01240
C                                                                              CLS01250
  800  WRITE(6,3004)  CNT, CNT                                                 CLS01260
C                                                                              CLS01270
       WRITE(6,8002)  ( FETVEC(I), I=LOC,ISTOP)                                CLS01280
 8002  FORMAT( / 1X, 'CHANNEL NO.' , 3X, I2, ( 11(6X,I2) )  )                  CLS01290
       WRITE(6,8003)  (DASH,I=1,3)                                             CLS01300
 8003  FORMAT(1X, 3A4 ///)                                                     CLS01310
C                                                                              CLS01320
       IF(ISTOP.EQ.NOFEAT)RETURN                                               CLS01330
       CNT=BIAS                                                                CLS01340
       JKSV=JK                                                                 CLS01350
       LOC=LOC+12                                                              CLS01360
       ISTOP=ISTOP+12                                                          CLS01370
       IF(ISTOP.GT.NOFEAT)ISTOP=NOFEAT                                         CLS01380
       GO TO 400                                                               CLS01390
C                                                                              CLS01400
CCC                                                                            CLS01410
C      ------------------------------------------------------------------------CLS01420
C                                                                              CLS01430
       ENTRY MULSPC(MEAN,SUBSTD,JDVEC,PTRVEC,PLOT,                             CLS01440
      *NOFEAT,FETVEC,SPCBAS)                                                   CLS01450
       DIMENSION JDVEC(1,1),BUF(4)                                             CLS01460
       WRITE(4,20021)                                                          CLS01470
       JPSTR =1                                                                CLS01480
       JPLOT = PTRVEC(5)                                                       CLS01490
       IF(JPLOT .NE. 1) GOTO 900                                               CLS01500
       JPSTR = 2                                                               CLS01510
       JPLOT = 2                                                               CLS01520
       JBIAS = 1                                                               CLS01530
       WAT=PTRVEC(1)                                                           CLS01540
       WRITE(4,9002) SYMVEC(2),JDVEC(1,WAT)                                    CLS01550
       GOTO 300                                                                CLS01560
  900  DO 22 JKL = JPSTR,JPLOT                                                 CLS01570
       WAT=PTRVEC(JKL)                                                         CLS01580
```

FILE: CLSSPC FORTRAN A

```
   22 BUF(JKL)=JDVEC(1,WAT)                                          CLS01590
      WRITE(6,9002)  (SYMVEC(I),BUF(I),I=JPSTR,JPLOT)                CLS01600
 9002 FORMAT(57X,A1,' = SUBCLASS ',A4)                               CLS01610
      JBIAS = 0                                                      CLS01620
      GO TO 300                                                      CLS01630
       END                                                           CLS01640
```

```
FILE: FLDCOV   FORTRAN  A

      SUBROUTINE FLDCOV(COR,DEV,MEAN,VAR,PTS,GO,FLDNAM,                 FLD00010
     *NOFEAT,MAXFFT,VARSIZ)                                             FLD00020
C                                                                       FLD00030
      IMPLICIT INTEGER (A-H,O-Z)                                        FLD00040
C     INCLUDE COMMON.LIST                                               FLD00050
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         FLD00060
     *              HISFIL,HISKEY,TMFORM,ERRTAP,ERRKEY,MAPUNT,NOFILE,   FLD00070
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                FLD00080
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    FLD00090
     *  ,DDTUNT,DDTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              FLD00100
     *  CRDUNT,PRTUNT,RANDIO                                            FLD00110
CSEND                                                                   FLD00120
CCCC|                                                                 |FLD00130
CCCC|  PURPOSE.. CALCULATES THE COVARIANCE AND CORRELATION MATRICES    |FLD00140
CCCC|            FROM THE RAW DATA  FOR THE FIELDS                     |FLD00150
CCCC|-----------------------------------------------------------------|FLD00160
      DOUBLE PRECISION COR(VARSIZ),DEV(NOFEAT),MEAN(NOFEAT),VAR(VARSIZ) FLD00170
CCC   ---------------------------------------------------------------FLD00180
      DATA  RCDT*0/ '2' / , DASH/'----' /                               FLD00190
CCC   ---------------------------------------------------------------FLD00200
      DATA  RCDT*0/ '2' / , DASH/'----' /                               FLD00210
CCC   ---------------------------------------------------------------FLD00220
CCC   ---------------------------------------------------------------FLD00230
CCC   ---------------------------------------------------------------FLD00240
      IF( GO .NE. 1)  GO TO 20                                          FLD00250
      WRITE(6,HEAD)                                                     FLD00260
      WRITE(6,10)  NOFEAT                                               FLD00270
   10 FORMAT('0'/' THE MEAN,STANDARD DEVIATION,COVARIANCE,AND CORRELATIOFLD00280
     1N (' , I2, ' CHANNELS)   FOR:' //)                                FLD00290
      WRITE(6,11) FLDNAM    ,(DASH,I=1,3)                               FLD00300
   11 FORMAT(T50,'TRAINING FIELD    ',A4/T49,4A4/)                     FLD00310
CCC                                                                     FLD00320
CCC                                                                     FLD00330
CCC   EQUATIONS :                                                       FLD00340
CCC                                                                     FLD00350
CCC                                                                     FLD00360
CCC                                                                     FLD00370
CCC                    **  N                **                        FLD00380
CCC                    *  ***               *                         FLD00390
CCC                 1  *    *               *                         FLD00400
CCC   COVAR(1,2) = ---*     * X  X  - U  U   *                         FLD00410
CCC                N-1*     *  1  2    1  2  *                         FLD00420
CCC                    *  ***               *                         FLD00430
CCC                    **  J=1              **                         FLD00440
CCC                     N                                               FLD00450
CCC                                                                     FLD00460
CCC                    ***                                              FLD00470
CCC                 1  *                                                FLD00480
CCC   MEAN(1)     = -  *  X  = U                                        FLD00490
CCC                 N  *   1    1                                       FLD00500
CCC                    ***                                              FLD00510
CCC                    I=1                                              FLD00520
CCC                                                                     FLD00530
CCC                   **  *************                                 FLD00540
CCC                    *  *                                            FLD00550
CCC   STDEV(2)    =    ** COVAR(2,2)                                    FLD00560
CCC                     *                                              FLD00570
CCC                                                                     FLD00580
   20 JK = 0                                                            FLD00590
      PTS1 = PTS                                                        FLD00600
      PTS2 =PTS-1                                                       FLD00610
      IF (PTS2 .LT. 1) PTS2 = 1                                         FLD00620
      N = NOFEAT                                                        FLD00630
      DO 40 JA=1,N                                                      FLD00640
      J = JA                                                            FLD00650
      DO 30 K=1,J                                                       FLD00660
      JK = JK+1                                                         FLD00670
      VAR(JK) = (VAR(JK) -MEAN(J)*MEAN(K)/PTS1)/PTS2                    FLD00680
   30 CONTINUE                                                          FLD00690
      DEV(J ) = DSQRT(DABS(VAR(JK)))                                    FLD00700
   40 CONTINUE                                                          FLD00710
      JK = 0                                                            FLD00720
      DO 50 J=1,NOFEAT                                                  FLD00730
      MEAN(J) = MEAN(J)/PTS1                                            FLD00740
      DO 50 K=1,J                                                       FLD00750
      JK = JK +1                                                        FLD00760
      COR(JK) = 0.0                                                     FLD00770
      IF (DEV(K)*DEV(J).LT.1.0E-25) GO TO 50                            FLD00780
      COR(JK) = VAR(JK)/(DEV(J)*DEV(K))                                 FLD00790
```

FILE: FLDCOV FORTRAN A

```
   50 CONTINUE                                                                  FLD00800
                                                                                FLD00810
C     ---------------------------------------------------------------------------FLD00820
CCC                                                                             FLD00830
      IF (GO .EQ. 0)  RETURN                                                    FLD00840
      DO 70 LOC=1,NOFEAT,12                                                     FLD00850
      STOP = LOC+11                                                             FLD00860
      IF(STOP .GT. NOFEAT)  STOP = NOFEAT                                       FLD00870
      WRITE(6,100)  ( MEAN(I), I=LOC,STOP)                                      FLD00880
      WRITE(6,1001) (DASH,I=1,2)                                                FLD00890
      WRITE(6,110)  ( DEV(I), I=LOC,STOP)                                       FLD00900
      WRITE(6,1001) (DASH,I=1,2)                                                FLD00910
      WRITE (6,120)                                                             FLD00920
   70 CONTINUE                                                                  FLD00930
  100 FORMAT( '0MEAN:',12F9.2 )                                                 FLD00940
 1001 FORMAT(1X,A4,A2/)                                                         FLD00950
  110 FORMAT( '0ST DEV:', 12F9.2 )                                              FLD00960
  120 FORMAT('0')                                                               FLD00970
                                                                                FLD00980
C                                                                               FLD00990
      WRITE(6,130)  ( DASH, I=1,5)                                              FLD01000
  130 FORMAT('0' / ' COVARIANCE MATRIX' / 1X,4A4,A2 )                           FLD01010
      CALL DWRTMX(VAR,NOFEAT,NCOLTWO)                                           FLD01020
      IF (NOFEAT.LE.MAXFET) GO TO 140                                           FLD01030
      WRITE (6,HEAD)                                                            FLD01040
                                                                                FLD01050
C                                                                               FLD01060
  140 WRITE(6,150)  ( DASH, I=1,5)                                              FLD01070
  150 FORMAT( '1' //' CORRELATION MATRIX' /1X,5A4 )                             FLD01080
      CALL DWRTMX(COR,NOFEAT,NCOLTWO)                                           FLD01090
      RETURN                                                                    FLD01100
C     ---------------------------------------------------------------------------FLD01110
CCC                                                                             FLD01120
      ENTRY CLSCOV(COR,DEV,MEAN,VAR,PTS,GO,                                     FLD01130
     *TITLE,NOFEAT,MAXFET,VARSIZ)                                               FLD01140
      IF (GO.EQ.0) GO TO 20                                                     FLD01150
      WRITE (6,HEAD)                                                            FLD01160
      WRITE(6,10)  NOFEAT                                                       FLD01170
      WRITE(6,160)  TITLE   ,(DASH,I=1,5)                                       FLD01180
  160 FORMAT(T47,'TRAINING SUBCLASS    ',A4/T48,4A4,A2)                         FLD01190
      GO TO 20                                                                  FLD01200
       END                                                                      FLD01210
```

```
FILE  LEARN    FORTRAN  A

      SUBROUTINE LEARN(SPEC,COVAR,AVAR,CLSDES,SUBSAV,                          LEA00010
     *                 FLDMEN,FLDVAR,SUBMEN,SUBVAR,SUBSTD,                     LEA00020
     *                 SUBCLS,MFTALY,MSTALY,FLDSAV,SAVERT,KEPPTS,MAXSUB)       LEA00030
C                                                                              LEA00040
      IMPLICIT INTEGER(A-Z)                                                    LEA00050
      REAL ASCALE,XSCPT,                                                       LEA00060
     * COVAR(VARSIZ),AVAR(NUMFEAT,MAXSUB),SUBSTD(NUMFEAT,MAXSUB)               LEA00070
      DOUBLE PRECISION FLDMEN(NUMFEAT),FLDVAR(VARSIZ),SUBMEN(NUMFEAT),        LEA00080
     * SUBVAR(VARSIZ),COR(465),DEV(30)                                         LEA00090
C                                                                              LEA00100
C     INCLUDE COMBLK4,LIST                                                     LEA00110
C     INCLUDE COMBLK5,LIST                                                     LEA00120
C     INCLUDE COMBLK6,LIST                                                     LEA00130
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                           LEA00140
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                        LEA00150
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                      LEA00160
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,               LEA00170
     *              MISFIL,MISKEY,TRFORM,ERRPTR,ERRKEY,MAPUNT,NUFILE,          LEA00180
     *    DRUMAD,DRMWD,MAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                        LEA00190
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                           LEA00200
     *  ,DUTUNT,DUTFIL,NCMPAS,TRNSFL,BMTAPE,MISTFL,PCHUNT,                     LEA00210
     *  CRDUNT,PRTUNT,RANDID                                                   LEA00220
C STAT COMMON BLOCK                                                            LEA00230
      COMMON /STBASE/SUBSV1,SUBVR1,SUBVR1,SUBSU1,SUBCL1,SAVER1,MSTAL1,         LEA00240
     *SPEC1,COVAR1,AVAR1,CLSTD1,FLMEN1,FLVAR1,MFTAL1,FLDSV1                    LEA00250
      COMMON /STCBLK/          MAXFEA,MAXCLS,MAXFLD,NUMFEAT,NUMFEAT2,         LEA00260
     *VARSIZ,NUMSPEC,NUMIST,SPCBAS,IBLOCK(30),FETVEC(30),                      LEA00270
     *FETVC2(30),MISVEC(30),NUMFLD,NOCLS,                                      LEA00280
     *   FLDINF(6),FLDPTS,CLSPTS,XSIZ,ARROW,ALOW,YSIZ                          LEA00290
CSEND                                                                          LEA00300
C                                                                              LEA00310
      DIMENSION LMIST(30),VERTCS(2,11),PL(8)                                   LEA00320
      DIMENSION SPEC(5,NUMSPEC),FLDSAV(10,MAXFLD),CLSDES(MAXSUB),              LEA00330
     * MFTALY(NUMIST,XSIZ),MSTALY(NUMIST,XSIZ),KEPPTS(MAXSUB),                 LEA00340
     * IDATA(12000),DUMPTR(5),SUBSAV(4,MAXSUB),                                LEA00350
     *SAVERT(22,MAXFLD),SUBCLS(1),SUBDES(300)                                  LEA00360
C                                                                              LEA00370
      EQUIVALENCE (LMIST,BESTVC)                                               LEA00380
      EQUIVALENCE          (IBLOCK( 1),NUMREC),(IBLOCK( 2),PCHKEY),            LEA00390
     *          (IBLOCK(3),SSPKEY),(IBLOCK(4),CPUKEY),                         LEA00400
     *               (IBLOCK(5),MSBKEY),(IBLOCK(6),MFDKEY),                    LEA00410
     *               (IBLOCK(7), SSLKEY),(IBLOCK(8),SFDKEY),                   LEA00420
     4                    (IBLOCK( 9),NUMREC2),(IBLOCK(10),CALKEY),           LEA00430
     5                    (IBLOCK(11),PCPCKY),(IBLOCK(12),PCCLKY),            LEA00440
     6                    (IBLOCK(13),ISTKEY),(IBLOCK(14),IRRKEY),            LEA00450
     7                    (IBLOCK(15),IMRSKY),(IBLOCK(16),STATKY),            LEA00460
     8                    (IBLOCK(17),PCALKY)                                 LEA00470
      EQUIVALENCE (IDATA(1),COR(1))                                            LEA00480
      EQUIVALENCE          (FLDINF(1), LINSTR), (FLDINF(2),LINEAD),            LEA00490
     1                     (FLDINF(3),LINEAC), (FLDINF(4), SAMSTR),            LEA00500
     2                     (FLDINF(5),SAMEND),(FLDINF(6),SAMINC),              LEA00510
     3                     (FLDINF(7),FLDTYP)                                  LEA00520
C                                                                              LEA00530
      DATA ENDCRD/'SEND'/,DUMPTR/1,3*0,1/,DASHV'----'/,                        LEA00540
     * BLANK/' '/,OPAR/'('/,CPAR/')'/,COMMA/','/,PUNCH//                       LEA00550
C                                                                              LEA00560
C     ---------- ---------- ---------- ---------- ---------- ---------- -----LEA00570
C     ---------- ---------- ---------- ---------- ---------- ---------- -----LEA00580
C                                                                              LEA00590
C     INIZ                                                                     LEA00600
C     ---------- ---------- ---------- ---------- ---------- ---------- -----LEA00610
C                                                                              LEA00620
C*    SET UP LOGICAL ARRAY FOR FEATURES TO BE HISTOGRAMMED.                    LEA00630
      SUBNO=0                                                                  LEA00640
      DO 7 I=1,NUMIST                                                          LEA00650
      DO 5 J=1,NUMFEAT                                                         LEA00660
      IF(MISVEC(I).EQ.FETVEC(J))GO TO 6                                        LEA00670
    5 CONTINUE                                                                 LEA00680
    6 LMIST(I)=J                                                               LEA00690
    7 CONTINUE                                                                 LEA00700
      DO 922 I=1,MAXSUB                                                        LEA00710
  922 SUBCLS(I)=0                                                              LEA00720
C*    READ HEADER RECORD ON DATA TAPE                                          LEA00730
      CALL TAPHDR(DATAPE,DATFIL)                                               LEA00740
      REWIND SAVTAP                                                            LEA00750
      IF(STAFIL.EQ.0) GO TO 941                                                LEA00760
      CALL FSASPL(SAVTAP,STAFIL,NSTAT)                                         LEA00770
      IF(NSTAT.EQ.0) GO TO 941                                                 LEA00780
      WRITE(6,942) NSTAT                                                       LEA00790
```

OFFICIAL USE ONLY
OF FOR OFFICIAL USE

FILE LEARN FORTRAN A

```
  542 FORMAT(' BAD POSITIONING OF SAVTAP, TERMINATING ',I3)                    LEA00800
      CALL CMERR                                                               LEA00810
  541 CONTINUE                                                                 LEA00820
      CALSW = CALKEY                                                           LEA00830
      BADFLG = 0                                                               LEA00840
      NOFLD =0                                                                 LEA00850
      NOCLS = 0                                                                LEA00860
   10 CONTINUE                                                                 LEA00870
      WRITE(6,HEAD)                                                            LEA00880
      IF(HSOKEY+HFOKEY.EQ.0) GO TO 14                                          LEA00890
      XSCALE=FLOAT(1-ASIZ)/(XHIGH-XLOW)                                        LEA00900
      XSHFT=-XHIGH*XSCALE+1.0                                                  LEA00910
      DO 20 I=1,XSIZ                                                           LEA00920
      DO 20 J=1,NUMIST                                                         LEA00930
   20 HSTALY(J,I) = 0                                                          LEA00940
      IF(HSOKEY+HFOKEY.NE.2) GO TO 14                                          LEA00950
      DO 30 I=1,XSIZ                                                           LEA00960
      DO 30 J=1,NUMIST                                                         LEA00970
   30 HSTALY(J,I)=0                                                            LEA00980
      GO TO 14                                                                 LEA00990
C CLASSES                                                                      LEA01000
   11 READ(30,12) TITLE                                                        LEA01010
   12 FORMAT(10X,A4)                                                           LEA01020
      REWIND 30                                                                LEA01030
      NOCLS=NOCLS+1                                                            LEA01040
      CLSTOT=NOCLS                                                             LEA01050
      CLSDES(NOCLS)=TITLE                                                      LEA01060
      GO TO 14                                                                 LEA01070
C SUBCLASSES                                                                   LEA01080
   13 READ(30,12) TITLE                                                        LEA01090
      REWIND 30                                                                LEA01100
      SUBNO=SUBNO+1                                                            LEA01110
      IF(SUBNO.GT.MAXSUB) GO TO 490                                            LEA01120
      SCLTOT=SUBNO                                                             LEA01130
      SUBSAV(4,SUBNO)=TITLE                                                    LEA01140
C  STARTING FIELD                                                              LEA01150
      SUBSAV(2,SUBNO) = NOFLD + 1                                              LEA01160
      SUBSAV(1,SUBNO)=NOCLS                                                    LEA01170
      SUBCLS(NOCLS)=SUBCLS(NOCLS)+1                                            LEA01180
C READ FIELD CARD                                                              LEA01190
   14 CFLAG=LAREAD(FLDNAM,VERTCS,FLDINF,NC)                                    LEA01200
C  END, CLASS, AND SUBCLASS                                                    LEA01210
      IF (CFLAG .EQ. 0) GO TO 60                                               LEA01220
      IF (CFLAG .EQ. -1) GO TO 11                                              LEA01230
      IF (CFLAG .EQ. -2) GO TO 13                                              LEA01240
      NOFLD=NOFLD+1                                                            LEA01250
      IF(NOFLD.GT.MAXFLD) GO TO 510                                            LEA01260
      FLDSAV(1,NOFLD)=FLDNAM                                                   LEA01270
      FLDSAV(2,NOFLD)=NOCLS                                                    LEA01280
      FLDSAV(3,NOFLD)=SUBNO                                                    LEA01290
      FLDSAV(4,NOFLD)=NC                                                       LEA01300
      FLDSAV(5,NOFLD)=LINSTR                                                   LEA01310
      FLDSAV(6,NOFLD)=LINEND                                                   LEA01320
      FLDSAV(7,NOFLD)=SAMSTR                                                   LEA01330
      FLDSAV(8,NOFLD)=SAMEND                                                   LEA01340
      FLDSAV(9,NOFLD)=LININC                                                   LEA01350
      FLDSAV(10,NOFLD)=SAMINC                                                  LEA01360
      SUBSAV(3,SUBNO) = NOFLD                                                  LEA01370
      K=0                                                                      LEA01380
      DO 111 J=1,11                                                            LEA01390
      DO 111 I=1,2                                                             LEA01400
      K=K+1                                                                    LEA01410
  111 SAVERT(K,NOFLD)=VERTCS(I,J)                                              LEA01420
      TOTVRT=TOTVRT+NC                                                         LEA01430
      GO TO 14                                                                 LEA01440
   60 CONTINUE                                                                 LEA01450
      WRITE(SAVTAP) NOCLS,SUBNO,NOFEAT,NOFLD,TOTVRT,(FETVEC(I),I=1,NOFEAL EA01460
     *T)                                                                       LEA01470
      DO 61 I=1,NOFLD                                                          LEA01480
      TNC=2*FLDSAV(4,I)                                                        LEA01490
      WRITE(SAVTAP) (FLDSAV(J,I),J=1,4)                                        LEA01500
   61 WRITE(SAVTAP) (SAVERT(J,I),J=1,TNC)                                      LEA01510
      WRITE(SAVTAP)(CLSDES(J),J=1,NOCLS),(SUBCLS(J),J=1,NOCLS),                LEA01520
     *              (SUBSAV(4,J),J=1,SUBNO)                                   LEA01530
      IF(PCHKEY.NE.1) GO TO 62                                                 LEA01540
      WRITE(PCHUNT,63)                                                         LEA01550
   63 FORMAT('MODULE     TRAINING FIELD DECK')                               LEA01560
      WRITE(PCHUNT,64) NOCLS,SUBNO,NOFEAT,NOFLD,TOTVRT                         LEA01570
   64 FORMAT('NOCLS ',I4,' NOSUB  ',I2,' NOFEAT ',I2,' NOFLD ',I3,            LEA01580
```

```
FILE  LEARN    FORTRAN  A

     * ' TOTVRT ',I4)                                                          LEA01590
       WRITE(PCHUNT,165) (FETVEC(I),I=1,NUFEAT)                               LEA01600
  165 FORMAT('CHNVEC',4X,3012)                                                LEA01610
       DO 65 I=1,NUFLD                                                        LEA01620
       WRITE(PCHUNT,66) (FLDSAV(J,I),J=1,4)                                   LEA01630
   66 FORMAT(A4,6X,I2,6X,I2,6X,I2)                                            LEA01640
       TNC=2*FLDSAV(4,I)                                                      LEA01650
   65 WRITE(PCHUNT,67) (SAVERT(J,I),J=1,TNC)                                  LEA01660
   67 FORMAT('VERTICES  ',1415)                                               LEA01670
       WRITE(PCHUNT,68) (CLSDES(J),J=1,NUCLS)                                 LEA01680
   68 FORMAT(('CLSDES  ',9(2X,A4)))                                           LEA01690
       WRITE(PCHUNT,69) (SUBCLS(J),J=1,NUCLS)                                 LEA01700
   69 FORMAT(('SUBNU  ',24(1X,I2)))                                           LEA01710
       WRITE(PCHUNT,90) (SUBSAV(4,J),J=1,SUBNU)                               LEA01720
   90 FORMAT(('SUBDES  ',10(A4,1X)))                                          LEA01730
   62 CONTINUE                                                                LEA01740
       WRITE(6,41) (DASH,I=1,20)                                              LEA01750
       DO 40 K=1,NUFLD                                                        LEA01760
       JJ=2*(FLDSAV(4,K)-1)                                                   LEA01770
       MP=FLDSAV(2,K)                                                         LEA01780
       KJ=10                                                                  LEA01790
       IF(JJ.LE.10) KJ=JJ                                                     LEA01800
       MPP=FLDSAV(3,K)                                                        LEA01810
       WRITE(6,42) K,FLDSAV(1,K),CLSDES(MP),SUBSAV(4,MPP),FLDSAV(10,K),       LEA01820
      *FLDSAV(9,K),((UPAR,SAVERT(I,K),COMMA,SAVERT(I+1,K),CPAR),I=1,KJ,2)LEA01830
       IF(JJ.LE.10) GO TO 2017                                                LEA01840
       WRITE(6,43) ((UPAR,SAVERT(I,K),COMMA,SAVERT(I+1,K),CPAR),I=11,JJ,2LEA01850
      *)                                                                      LEA01860
 2017 CONTINUE                                                               LEA01870
   42 FORMAT(4X,I4,2X,A4,4X,A4,4X,A4,5X,I4,3X,I4,4X,                         LEA01880
      *5(A1,I4,A1,I4,A1,2X))                                                 LEA01890
   43 FORMAT(50X,5(A1,I4,A1,I4,A1,2X))                                       LEA01900
   40 CONTINUE                                                               LEA01910
   41 FORMAT(//T50,'TRAINING FIELDS'/T49,4A4//                               LEA01920
      *7X,'FIELD',T36,'SAMPLE LINE'/                                         LEA01930
      *5X,'NO.  NAME',4X,'CLASS',3X,'SUBCLASS  INC     INC',                LEA01940
      *4X,'VERTICES(SAMPLE,LINE)'/                                           LEA01950
      *4X,3A4,2X,A4,A2,2X,2A4,1X,3A4,4X,5A4,A1)                              LEA01960
       SUBNU=0                                                                LEA01970
   70 SUBNU=SUBNU+1                                                          LEA01980
       SUBPTS=0                                                               LEA01990
       DO 71 I=1,NUFEAT                                                       LEA02000
   71 SUBMEN(I)=0.                                                           LEA02010
       DO 72 I=1,VARSIZ                                                       LEA02020
   72 SUBVAR(I)=0.                                                           LEA02030
       FIELD1=SUBSAV(2,SUBNU)                                                 LEA02040
       FIELDL=SUBSAV(3,SUBNU)                                                 LEA02050
       DO 301 N=FIELD1,FIELDL                                                 LEA02060
       DO 73 I=1,NUFEAT                                                       LEA02070
   73 FLDMEN(I)=0.                                                           LEA02080
       DO 74 I=1,VARSIZ                                                       LEA02090
   74 FLDVAR(I)=0.                                                           LEA02100
       LINSTR=FLDSAV(5,N)                                                     LEA02110
       LINEND=FLDSAV(6,N)                                                     LEA02120
       LININC=FLDSAV(9,N)                                                     LEA02130
       SAMSTR=FLDSAV(7,N)                                                     LEA02140
       SAMEND=FLDSAV(8,N)                                                     LEA02150
       SAMINC=FLDSAV(10,N)                                                    LEA02160
       CALL FLDINT(FLDINF,FETVEC,NUFEAT)                                      LEA02170
       LINES=(LINEND-LINSTR)/LININC+1                                         LEA02180
       PTS=(SAMEND-SAMSTR)/SAMINC+1                                           LEA02190
       NSAMP=PTS                                                              LEA02200
       FLDPTS=0                                                               LEA02210
       DO 17 JLINES=1,LINES                                                   LEA02220
       CALL LINERD(IDATA,ENDTAP)                                              LEA02230
       IF(ENDTAP.EQ.-1) GO TO 16                                              LEA02240
       IF(JLINES.NE.1) GO TO 8                                                LEA02250
       NLINES=LINSTR                                                          LEA02260
       GO TO 9                                                                LEA02270
    8  NLINES=NLINES+LININC                                                   LEA02280
    9  CONTINUE                                                               LEA02290
       K=0                                                                    LEA02300
       DO 93 I=1,11                                                           LEA02310
       DO 93 J=1,2                                                            LEA02320
       K=K+1                                                                  LEA02330
   93 VERTCS(J,I)=SAVERT(K,N)                                                 LEA02340
       CALL FDLINT(VERTCS,FLDSAV(4,N),FL,NLINES,NS,JJJ)                       LEA02350
       KK=0                                                                   LEA02360
       NN=1                                                                   LEA02370
```


ORIGINAL PAGE IS
OF POOR QUALITY

FILE LEARN FORTRAN A

ORIGINAL PAGE IS
OF POOR QUALITY

```
      KC=0                                                              LEAU2380
      DO 1 J=1,NUFEAT                                                   LEAU2390
      INDEX1=(J-1)*NSAMP                                                LEAU2400
      JJ=J                                                              LEAU2410
      DO 3 K=1,JJ                                                       LEAU2420
      KK=KK+1                                                           LEAU2430
      INDEX2=(K-1)*NSAMP                                                LEAU2440
      L=1                                                               LEAU2450
      DO 2 JPT=1,NSAMP                                                  LEAU2460
      KPT=(JPT-1)*SAMINC+SAMSTR                                         LEAU2470
      DO 4 JK=L,JJJ,2                                                   LEAU2480
      IF(KPT.LT.FL(JK)) GO TO 2                                         LEAU2490
      IF(KPT.GT.FL(JK+1)) GO TO 15                                      LEAU2500
      IDJ=IDATA(INDEX1+JPT)                                             LEAU2510
      IF(J.EQ.1) FLOPTS=FLOPTS+1                                        LEAU2520
      IF(K.EQ.1) FLOMEN(J)=FLOMEN(J)+IDJ                                LEAU2530
      FLOVAR(KK)=FLOVAR(KK)+IDJ*IDATA(INDEX2+JPT)                       LEAU2540
      IF(LHIST(NN).NE.J) GO TO 2                                        LEAU2550
      KC=1                                                              LEAU2560
      IPUT=IDATA(JPT+INDEX1)*XSCALE+XSHFT+0.501                         LEAU2570
      IF(IPUT.LT.1) IPUT=1                                              LEAU2580
      IF(IPUT.GT.XSIZ) IPUT=XSIZ                                        LEAU2590
      HFTALY(NN,IPUT)=HFTALY(NN,IPUT)+1                                 LEAU2600
      IF(JPT.EQ.NSAMP) NN=NN+1                                          LEAU2610
      GO TO 2                                                           LEAU2620
   15 L=L+2                                                             LEAU2630
      IF(L.GT.JJJ) GO TO 53                                             LEAU2640
    4 CONTINUE                                                          LEAU2650
    2 CONTINUE                                                          LEAU2660
      IF(JPT.EQ.NSAMP) GO TO 3                                          LEAU2670
   53 IF(KC.EQ.1) NN=NN+1                                               LEAU2680
      KC=0                                                              LEAU2690
    3 CONTINUE                                                          LEAU2700
    1 CONTINUE                                                          LEAU2710
   17 CONTINUE                                                          LEAU2720
   16 CONTINUE                                                          LEAU2730
    / SUBPTS=SUBPTS+FLOPTS                                              LEAU2740
      DO 200 I=1,NUFEAT                                                 LEAU2750
  200 SUBMEN(I)=SUBMEN(I)+FLOMEN(I)                                     LEAU2760
      DO 21 I=1,VARSIZ                                                  LEAU2770
   21 SUBVAR(I)=SUBVAR(I)+FLOVAR(I)                                     LEAU2780
      TITLE=SUBSAV(4,SUBNO)                                             LEAU2790
      IF(CFDKEY+SFDKEY+HFDKEY.EQ.0) GO TO 301                           LEAU2800
      IF(CFDKEY+SFDKEY.EQ.0) GO TO 280                                  LEAU2810
      CALL FLOCOV(COR,DEV,FLOMEN,FLOVAR,FLOPTS,CFDKEY,FLDSAV(1,N),      LEAU2820
     *NUFEAT,MAXFET,VARSIZ)                                             LEAU2830
C PLOT SPECTRAL RESPONSE FOR FIELDS                                     LEAU2840
      IF(SFDKEY.EQ.0) GO TO 270                                         LEAU2850
      CALL FLOSPC(FLOMEN,DEV,FLDSAV(1,N),DUMPTR,IDATA,                  LEAU2860
     *FLOMEN,FLOVAR,TITLE,NUFEAT,FETVEC,SPCBAS)                         LEAU2870
  270 IF(HFDKEY.EQ.0) GO TO 301                                         LEAU2880
  280 IF(HSBKEY.EQ.0) GO TO 300                                         LEAU2890
      DO 290 I=1,XSIZ                                                   LEAU2900
      DO 290 J=1,NUMIST                                                 LEAU2910
  290 HSTALY(J,I)=HSTALY(J,I)+HFTALY(J,I)                               LEAU2920
  300 CONTINUE                                                          LEAU2930
      CALL FLOHIS(HFTALY,IDATA,FLDSAV(1,N),XSIZ,XHGH,XLOW,YSIZ,         LEAU2940
     *NUMIST,FLOPTS,TITLE,HISVEC)                                       LEAU2950
  301 CONTINUE                                                          LEAU2960
C CALCULATE COVAR MTX AND MEAN VECTOR FOR SUBCLASS                      LEAU2970
      CALL CLSCOV(COR,DEV,SUBMEN,SUBVAR,SUBPTS,SSFKEY,                  LEAU2980
     *TITLE,NUFEAT,MAXFET,VARSIZ)                                       LEAU2990
C SAVE SUBCLASS MEAN,COVAR,STD DEV                                      LEAU3000
  130 DO 31 I=1,NUFEAT                                                  LEAU3010
      AVAR(I,SUBNO)=SUBMEN(I)                                           LEAU3020
   31 SUBSTD(I,SUBNO)=DEV(I)                                            LEAU3030
      DO 32 J=1,VARSIZ                                                  LEAU3040
   32 COVAR(J)=SUBVAR(J)                                                LEAU3050
      KEPPTS(SUBNO)=SUBPTS                                              LEAU3060
C PLOT SPECTRAL RESPONSE FOR EACH SUBCLASS                              LEAU3070
      IF(SSLKEY.EQ.0) GO TO 33                                          LEAU3080
      CALL CLSSPC(AVAR(1,SUBNO),SUBSTD(1,SUBNO),TITLE,DUMPTR,IDATA,     LEAU3090
     *TITLE,NUFEAT,FETVEC,SPCBAS)                                       LEAU3100
C PRINT SUBCLASS HIST                                                   LEAU3110
   33 IF(HSBKEY.EQ.0) GO TO 390                                         LEAU3120
      TITLE=SUBSAV(4,SUBNO)                                             LEAU3130
      IF(HFDKEY.EQ.0) GO TO 380                                         LEAU3140
      CALL CLSHIS(HSTALY,IDATA,TITLE    ,XSIZ,XHGH,XLOW,YSIZ,           LEAU3150
     *NUMIST,FLOPTS,HISVEC)                                             LEAU3160
```

```
FILE  LEAKN    FORTRAN  A

      GO TO 390                                                          LEA03170
  380 CALL CLSMIS(HFTALY,IDATA,TITLE    ,XSIZ,XMGN,XLOW,YSIZ,           LEA03180
     *NOHIST,FLOPTS,HISVEC)                                              LEA03190
  390 WRITE(SAVTAP) KEPPTS(SUBNO),(COVAR(I),I=1,VARSIZ),                 LEA03200
     *                 (AVAR(I,SUBNO),I=1,NOFEAT)                       LEA03210
      IF(PCHKEY.NE.1) GO TO 94                                           LEA03220
      WRITE(PCHUNT,95) KEPPTS(SUBNO)                                     LEA03230
   95 FORMAT('NOPTS ',6X,I8)                                             LEA03240
      WRITE(PCHUNT,96) (AVAR(I,SUBNO),I=1,NOFEAT)                        LEA03250
   96 FORMAT('MEANS',5E15.8)                                             LEA03260
      WRITE(PCHUNT,97) (COVAR(I),I=1,VARSIZ)                             LEA03270
   97 FORMAT('COVAR',5E15.8)                                             LEA03280
   94 IF(SUBNO.LT.SCLTOT) GO TO 70                                       LEA03290
      ENDFILE SAVTAP                                                     LEA03300
      REWIND SAVTAP                                                      LEA03310
C                                                                        LEA03320
CCC   PUBLISH THE MULTISPECTRAL PLOTS                                    LEA03330
CCC   ------------------------------------------------------------------LEA03340
C                                                                        LEA03350
  410 IF (SPEC(1,1).NE.0) GO TO 450                                      LEA03360
      JK = 0                                                             LEA03370
      DO 430 I=1,NOSPEC                                                  LEA03380
      DO 420 J=1,4                                                       LEA03390
      II=I                                                               LEA03400
      JJ=J                                                               LEA03410
      JK =JK+1                                                           LEA03420
      SPEC(J,I) = JK                                                     LEA03430
      IF(JK.EQ.SUBNO) GO TO 440                                          LEA03440
  420 CONTINUE                                                           LEA03450
  430 SPEC(5,I) = 4                                                      LEA03460
  440 SPEC(5,II)=JJ                                                      LEA03470
      NOSPEC=II                                                          LEA03480
  450 CONTINUE                                                           LEA03490
      DO 480 I=1,NOSPEC                                                  LEA03500
      K = SPEC(5,I)                                                      LEA03510
      JJ = 0                                                             LEA03520
      DO 460 J=1,K                                                       LEA03530
      IF(SPEC(J,I).GT.SUBNO) GO TO 460                                   LEA03540
      JJ = JJ +1                                                         LEA03550
      SPEC(JJ,I) = SPEC(J,I)                                             LEA03560
  460 CONTINUE                                                           LEA03570
      IF (JJ.EQ.0) GO TO 480                                             LEA03580
      SPEC(5,I) = JJ                                                     LEA03590
      WRITE (6,HEAD)                                                     LEA03600
      WRITE(6,465)                                                       LEA03610
  465 FORMAT(// 27X,'COMPOSITE SPECTRAL PLOT (MEAN,PLUS AND MINUS ONE ST LEA03620
     1D. DEV. ) FOR ' / )                                                LEA03630
      WRITE (6,470)(SPEC(J,I),J=1,JJ)                                    LEA03640
  470 FORMAT(38X,'TRAINING SUBCLASS(ES) ',4I4/41X,'----------------')     LEA03650
      DO 98 JI=1,SUBNO                                                   LEA03660
   98 SUBDES(JI)=SUBSAV(4,JI)                                            LEA03670
      IF( JJ .LT. 4)  WRITE(6,471)  ( DASH,II=1,5 )                      LEA03680
  471 FORMAT(41X,4A4,A2 / )                                              LEA03690
      CALL MULSPC(AVAR(1,1),SUBSTD(1,1),SUBDES,SPEC(1,1),IDATA,          LEA03700
     *NOFEAT,FETVEC,SPCBAS)                                              LEA03710
  480 CONTINUE                                                           LEA03720
      CALL SETMRG(66,4,62)                                               LEA03730
      RETURN                                                             LEA03740
C                                                                        LEA03750
CC    ERROR ROUTINES                                                     LEA03760
CC    ------------------------------------------------------------------LEA03770
C                                                                        LEA03780
  490  BADFLG = 2                                                        LEA03790
      WRITE(6,500) MAXSUB,MAXSUB                                         LEA03800
  500 FORMAT(//5X,'*** STAT/LEAKNN   MAX NO. OF',I8,3X, 'SUBCLASSES EXCELEA03810
     *EEDED---FIRST',I8,3X,'SUBCLASSES USED--REMAINDER IGNORED'///)      LEA03820
      GO TO 530                                                          LEA03830
C                                                                        LEA03840
  510 BADFLG = 1                                                         LEA03850
      WRITE(6,520) MAXFLD,MAXFLD,SUBNO                                   LEA03860
  520  FORMAT(//// 5X,'*****  STAT/LEAKNN --- MAX. OF',I8,3X,            LEA03870
     1 'FIELDS EXCEEDED --- ',I8,3X,'FIELDS RETAINED FOR',I8,3X,         LEA03880
     *'SUBCLASSES'/5X,'***REMAINDER OF INPUT TRAINING FIELDS NOT USED')/)LEA03890
C                                                                        LEA03900
  530 READ (21,540)I                                                     LEA03910
  540 FORMAT (A4)                                                        LEA03920
      IF (I.NE.ENDCRD) GO TO 530                                         LEA03930
      GO TO 60                                                           LEA03940
C                                                                        LEA03950
```

```
FILE  LEARN    FORTRAN  A
      END                                                    LEA03960
```

FILE: SETUP1 FORTRAN A

```
      SUBROUTINE SETUP1(SPCVEC,TOP,MAXSUB)                                  SET00010
C                                                                           SET00020
CI    PURPOSE.. READS AND ANALYZES SUPERVISOR CONTROL CARDS                 ISET00030
CI              FOR THE 'STATISTICS' STEP                                   ISET00040
C                                                                           SET00050
      IMPLICIT INTEGER (A-H,O-Z)                                            SET00060
C                                                                           SET00070
      DIMENSION  SINVEC(3), SPCVEC(5,20), CARD2(62),NUMVEC(30)              SET00080
      DIMENSION CINDEX(13),OPTCOD(6),EQUVEC(2),ACARD(20)                    SET00090
CI                                                                          1SET00100
C     INCLUDE COMBK4                                                        SET00110
C     INCLUDE  COMBK6,LIST                                                  SET00120
C     INCLUDE COMBKA,LIST                                                   SET00130
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                        SET00140
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                     SET00150
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                  SET00160
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             SET00170
     *     HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,                SET00180
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    SET00190
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                       SET00200
     *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                 SET00210
     *   CRDUNT,PRTUNT,RANDIO                                               SET00220
C STAT COMMON BLOCK                                                         SET00230
      COMMON /STBASE/SUBSV1,SUBMN1,SUBVR1,SUBSD1,SUBCL1,SAVER1,HSTAL1,      SET00240
     *SPEC1,COVAR1,AVAR1,CLSID1,FLMEN1,FLVAR1,HFTAL1,FLDSV1                 SET00250
      COMMON /STCALK/        MAXFET,MAXCLS,MAXFLD,NOFEAT,NOFET2,            SET00260
     *VARSIZ,NOSPEC,NOHIST,SPCBAS,IBLOCK(30),FETVEC(30),                    SET00270
     *FETVC2(30),HISVEC(30),NOFLD,NOCLS,                                    SET00280
     *   FLDINF(6),FLDPTS,CLSPTS,XSIZ,XHGH,XLOW,YSIZ                        SET00290
C$END                                                                       SET00300
      EQUIVALENCE       (IBLOCK( 1),NOTHNG),(IBLOCK( 2),PCHKEY),          SET00310
     *          (IBLOCK(3),SSFKEY),(IBLOCK(4),CFOKEY),                    SET00320
     *               (IBLOCK(5),HSBKEY),(IBLOCK(6),HFOKEY),               SET00330
     *               (IBLOCK(7), SSLKEY),(IBLOCK(8),SFOKEY),              SET00340
     4                    (IBLOCK( 9),NOTHG2),(IBLOCK(10),CALKEY),        SET00350
     5                    (IBLOCK(11),PCFOKY),(IBLOCK(12),PCCLKY),        SET00360
     6                    (IBLOCK(13),TSTKEY),(IBLOCK(14),TRNKEY),        SET00370
     7                    (IBLOCK(15),THRSKY),(IBLOCK(16),STATKY),        SET00380
     8                    (IBLOCK(17),PCALKY)                             SET00390
C                                                                           SET00400
C                                                                           SET00410
      DATA  SINVEC/2,',',' ', '=' / , EQUAL/'=' /, BLANK/ ' ' /            SET00420
      DATA        CBCD/'C'/,FBCD/'F'/,LBCD/'L'/,SBCD/'S'/,XBCD/'X'/,        SET00430
     1            YBCD/'Y'/,BLANK/' '/,HBCD/'H'/                          SET00440
     1            ,NBCD/'N'/,OBCD/'O'/,ABCD/'A'/                          SET00450
     3            ,UBCD/'U'/                                              SET00460
C                                                                           SET00470
      DATA CINMAX/13/,EQUVEC/1,'='/                                         SET00480
C                                                                           SET00490
      DATA        OPTCOD/'M','P','C','H','S'/                             SET00500
      DATA CINDEX/'OPTI','CHAN','HIST','SPEC','IBLO',                       SET00510
     1'SIZE','DATE','*END','COMM',                                          SET00520
     2'HED1','HED2','DATA','STAT'/                                          SET00530
C                                                                           SET00540
C                                                                           SET00550
C     INIZ                                                                  SET00560
C                                                                           SET00570
C*                                                                          SET00580
C*    READ AND UNPACK HEADER RECORD TO SET MAXFET ACCORDING TO ID(S)        SET00590
C*                                                                          SET00600
    1 DO 2 I=1,30                                                           SET00610
      FETVEC(I) = I                                                         SET00620
      HISVEC(I) = I                                                         SET00630
    2 IBLOCK(I) = 0                                                         SET00640
      NBLOCK = 0                                                            SET00650
      CALKEY =0                                                             SET00660
      CCLKEY = 1                                                            SET00670
      MAXCLS = 15                                                           SET00680
      MAXSUB=15                                                             SET00690
      MAXFET=30                                                             SET00700
      SYMMAX=30                                                             SET00710
      NOSPEC = 0                                                            SET00720
      SPCBAS = 75                                                           SET00730
      YSIZ = 14                                                             SET00740
      XSIZ=0                                                                SET00750
      XLOW=120                                                              SET00760
      XHGH=220                                                              SET00770
      INFMT = 0                                                             SET00780
      NOFEAT=0                                                              SET00790
```


ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SETUP1 FORTRAN A

```
C                                                                               SET00800
CCCCC                                                                           SET00810
CCCCC    READ AND ANALYZE SUPERVISOR CONTROL CARDS                              SET00820
CCCCC    ----------------------------------------                               SET00830
CCCCC                                                                           SET00840
C                                                                               SET00850
C     SETUP REREAD BUFFER                                                       SET00860
C                                                                               SET00870
      CALL REREAD(30,80)                                                        SET00880
  200 COL = 0                                                                   SET00890
C     NOW READ THE CARD INTO THE BUFFER                                         SET00900
      READ(21,150)(ACARD(I),I=1,20)                                             SET00910
  150 FORMAT(20A4)                                                              SET00920
      WRITE(30,150)(ACARD(I),I=1,20)                                            SET00930
      REWIND 30                                                                 SET00940
      READ(30,2002) CODE,CARD2                                                  SET00950
 2002 FORMAT (A4,6X,62A1)                                                       SET00960
      REWIND 30                                                                 SET00970
  225 WRITE(6,2252)  CODE,CARD2                                                 SET00980
 2252 FORMAT (T5,A4,6X,62A1)                                                    SET00990
      DO 230 I=1,CINMAX                                                         SET01000
      IF (CINDEX(I) .EQ. CODE)                                                  SET01010
     1 GO TO(10,600,700,800,1100,1200,1400,900,1500,1600,                       SET01020
     2  1700,1710,1720),I                                                      SET01030
  230 CONTINUE                                                                  SET01040
      GO TO 1000                                                                SET01050
C                                                                               SET01060
C     OPTION CARD                                                               SET01070
C     -----------                                                               SET01080
C                                                                               SET01090
   10 M = NXTCHR(CARD2,COL)                                                     SET01100
      IF(M .EQ. BLANK ) GO TO 200                                               SET01110
      IF( M .EQ. OPTCOD(1) ) GO TO 20                                           SET01120
      SETFLG = 1                                                                SET01130
      IF ( M .NE. NBCD ) GOTO 14                                                SET01140
      J = COL-1                                                                 SET01150
      M = NXTCHR(CARD2,COL)                                                     SET01160
      IF (M .NE. OBCD ) GOTO 12                                                 SET01170
      SETFLG = 0                                                                SET01180
      J = COL                                                                   SET01190
C                                                                               SET01200
   12 COL = J                                                                   SET01210
      M = NXTCHR(CARD2,COL)                                                     SET01220
   14 DO 15 I=2,5                                                               SET01230
      IF( M .EQ. OPTCOD(I) )  GO TO (40,30,25,30,30), I                         SET01240
   15 CONTINUE                                                                  SET01250
C                                                                               SET01260
      IF( M .EQ. BLANK)  GO TO 200                                              SET01270
   40 M = COL + 10                                                              SET01280
      WRITE(6,402)   M                                                          SET01290
  402 FORMAT(/ 1X,'***  STAT/SETUP1 --- ERROR IN OPTION(S) REQUESTED - SSET01300
     1CAN OF OPTION(S) DISCONTINUED AT CARD COLUMN',I5,2X,'***' /)              SET01310
      GO TO 200                                                                 SET01320
C                                                                               SET01330
   20 M = FIND12(CARD2,COL,SINVEC)                                              SET01340
      IF ( SINVEC(M) .NE. EQUAL ) GO TO 40                                      SET01350
      M = NUMBER(CARD2,COL,NUMVEC,29)                                           SET01360
      IF( NUMVEC(30) .LE. 0)  GO TO 40                                          SET01370
      MAXSUB=NUMVEC(30)                                                         SET01380
      GO TO 10                                                                  SET01390
C                                                                               SET01400
   25 J = 20                                                                    SET01410
      M = NXTCHR(CARD2,COL)                                                     SET01420
      IF ( M .EQ. OBCD ) J=3                                                    SET01430
      IF ( M .EQ. ABCD ) J=9                                                    SET01440
      IF ( J .LT. 20 ) GOTO 32                                                  SET01450
      GOTO 40                                                                   SET01460
C                                                                               SET01470
   30 J = I*2-3                                                                 SET01480
   32 M = FIND12(CARD2,COL,SINVEC)                                              SET01490
      IF ( SINVEC(M) .NE. EQUAL ) GOTO 38                                       SET01500
      M = NXTCHR(CARD2,COL)                                                     SET01510
      IF ( M .EQ. CBCD ) IBLOCK(J) = SETFLG                                     SET01520
      IF ( M .EQ. FBCD ) IBLOCK(J+1) = SETFLG                                   SET01530
      M = FIND12(CARD2,COL,SINVEC)                                              SET01540
      IF ( M .LE. 0 ) GOTO 200                                                  SET01550
      GOTO 10                                                                   SET01560
C                                                                               SET01570
   38 IBLOCK(J) = SETFLG                                                        SET01580
```

FILE: SETUP1 FORTRAN A

```
      IBLOCK(J+1) = SETFLG                                                  SET01590
      IF ( M .LE. 0 ) GOTO 200                                              SET01600
      GOTO 10                                                               SET01610
C                                                                           SET01620
CCCC  CHANNELS                                                              SET01630
CCCCC ========                                                              SET01640
CCCCC ------------------------                                              SET01650
C                                                                           SET01660
  600 J = NXTCHR(CARD2,COL)                                                 SET01670
      IF  ( J .EQ. BLANK ) GOTO 200                                         SET01680
      COL = COL-1                                                           SET01690
      NOFEAT = NUMBER(CARD2,COL,FETVEC,NOFEAT)                              SET01700
C                                                                           SET01710
CCCC  ELIMINATE OUT-OF-RANGE REQUESTED FEATURES,IF ANY,                     SET01720
CCCC  ORDER THE RESULTING FEATURE VECTOR                                    SET01730
C                                                                           SET01740
      NM1 = NOFEAT - 1                                                      SET01750
      IF( NM1 .LE. 0)  NM1 = 1                                              SET01760
      DO  611  I=1,NM1,1                                                    SET01770
      IP1 = I + 1                                                           SET01780
      IF( IP1 .GT. NOFEAT)  GO TO 611                                       SET01790
      DO  610  J=IP1,NOFEAT,1                                               SET01800
      IF( FETVEC(I) .EQ. FETVEC(J)) FETVEC(J) = I*MAXFET+J                  SET01810
  610 CONTINUE                                                              SET01820
  611 CONTINUE                                                              SET01830
      II = 0                                                                SET01840
      DO  612  I=1,NOFEAT,1                                                 SET01850
      CHK = FETVEC(I)                                                       SET01860
      IF( CHK .LE. 0 .OR. CHK .GT. MAXFET)  GO TO 612                       SET01870
      II = II + 1                                                           SET01880
      FETVEC(II) = FETVEC(I)                                                SET01890
  612 CONTINUE                                                              SET01900
      NOFEAT = II                                                           SET01910
      IF( NOFEAT .EQ. 0)  GO TO 1303                                        SET01920
      NM1 = NOFEAT - 1                                                      SET01930
      IF( NM1 .LE. 0)  NM1 = 1                                              SET01940
      DO  614  I=1,NM1,1                                                    SET01950
      IP1 = I + 1                                                           SET01960
      IF( IP1 .GT. NOFEAT)  GO TO 614                                       SET01970
      DO  613  J=IP1,NOFEAT,1                                               SET01980
      IF( FETVEC(I) .LT. FETVEC(J))  GO TO 613                              SET01990
      TEMP = FETVEC(I)                                                      SET02000
      FETVEC(I) = FETVEC(J)                                                 SET02010
      FETVEC(J) = TEMP                                                      SET02020
  613 CONTINUE                                                              SET02030
  614 CONTINUE                                                              SET02040
      GO TO 200                                                             SET02050
C                                                                           SET02060
CCCC  HISTOGRAM CARD                                                        SET02070
CCCC  ------------------------                                              SET02080
C                                                                           SET02090
  700 J = NXTCHR(CARD2,COL)                                                 SET02100
      IF  ( J .EQ. BLANK ) GOTO 200                                         SET02110
      COL = COL-1                                                           SET02120
      NOHIST = NUMBER(CARD2,COL,HISVEC,NOHIST)                              SET02130
C                                                                           SET02140
CCC   ELIMINATE OUT OF RANGE REQUESTED SUBCLASSES,IF ANY, AND               SET02150
CCC   ORDER THE RESULTING SUBCLASS VECTOR                                   SET02160
C                                                                           SET02170
      NM1 = NOHIST - 1                                                      SET02180
      IF( NM1 .LE. 0)  NM1 = 1                                              SET02190
      DO  711  I=1,NM1,1                                                    SET02200
      IP1 = I + 1                                                           SET02210
      IF( IP1 .GT. NOHIST)  GO TO 711                                       SET02220
      DO  710  J=IP1,NOHIST,1                                               SET02230
      IF( HISVEC(I) .EQ. HISVEC(J)) HISVEC(J) = I*SYMMAX+J                  SET02240
  710 CONTINUE                                                              SET02250
  711 CONTINUE                                                              SET02260
      II = 0                                                                SET02270
      DO  712  I=1,NOHIST,1                                                 SET02280
      CHK = HISVEC(I)                                                       SET02290
      IF( CHK .LE. 0 .OR. CHK .GT. SYMMAX)  GO TO 712                       SET02300
      II = II + 1                                                           SET02310
      HISVEC(II) = HISVEC(I)                                                SET02320
  712 CONTINUE                                                              SET02330
      NOHIST = II                                                           SET02340
      IF( NOHIST .EQ. 0)  GO TO 1303                                        SET02350
      NM1 = NOHIST - 1                                                      SET02360
      IF( NM1 .LE. 0)  NM1 = 1                                              SET02370
```

FILE: SETUP1 FORTRAN A

```
      DO  714  I=1,NM1,1                                                SET02380
      IP1 = I + 1                                                       SET02390
      IF( IP1 .GT. NOHIST)  GO TO 714                                   SET02400
      DO  713  J=IP1,NOHIST,1                                           SET02410
      IF( HISVEC(I) .LT. HISVEC(J))  GO TO 713                          SET02420
      TEMP = HISVEC(I)                                                  SET02430
      HISVEC(I) = HISVEC(J)                                             SET02440
      HISVEC(J) = TEMP                                                  SET02450
 713  CONTINUE                                                          SET02460
 714  CONTINUE                                                          SET02470
      GO TO 200                                                         SET02480
C                                                                       SET02490
CCCC  SPEC CARD                                                         SET02500
CCCC  ------------------------                                         SET02510
C                                                                       SET02520
 800  J = NXTCHR(CARD2,COL)                                             SET02530
      IF  ( J .EQ. BLANK ) GOTO 200                                     SET02540
      COL = COL-1                                                       SET02550
      NOSPEC = NOSPEC + 1                                               SET02560
      IF( NOSPEC .GT. 20)  GO TO 200                                    SET02570
      J = NUMBER(CARD2,COL,NUMVEC,0)                                    SET02580
      IF ( J .GT.  4 ) J = 4                                            SET02590
      DO 810 I=1,J                                                      SET02600
      IF (NUMVEC(I) .LE. 0 ) GOTO 815                                   SET02610
 810  SPCVEC(I,NOSPEC) = NUMVEC(I)                                      SET02620
      I = J+1                                                           SET02630
 815  SPCVEC(5,NOSPEC) = I-1                                            SET02640
      GOTO 200                                                          SET02650
C                                                                       SET02660
CCCC  IBLOCK CARD                                                       SET02670
CCCC  ------------------------                                         SET02680
C                                                                       SET02690
1100  J = NXTCHR(CARD2,COL)                                             SET02700
      IF  ( J .EQ. BLANK ) GOTO 200                                     SET02710
      COL = COL-1                                                       SET02720
      NBLOCK = NUMBER(CARD2,COL,NUMVEC,NBLOCK)                          SET02730
      DO  1110  I=1,NBLOCK,1                                            SET02740
      IF (NUMVEC(I) .EQ. 1) IBLOCK(I) = 1                               SET02750
1110  CONTINUE                                                          SET02760
      GO TO 200                                                         SET02770
C                                                                       SET02780
CCCC  SIZE CARD                                                         SET02790
CCCC  ------------------------                                         SET02800
C                                                                       SET02810
  97  COL=COL-1                                                         SET02820
1200  J = NXTCHR(CARD2,COL)                                             SET02830
      IF( J .EQ. BLANK ) GOTO 200                                       SET02840
      IF ( J .EQ. XHCD) GO TO 1220                                      SET02850
      IF ( J .EQ. SHCD) GO TO 1230                                      SET02860
      IF ( J .EQ. YHCD) GO TO 1240                                      SET02870
      GO TO 1000                                                        SET02880
C                                                                       SET02890
1220  J = NXTCHR(CARD2,COL)                                             SET02900
      M = FIND12(CARD2,COL,SINVEC)                                      SET02910
      IF( SINVEC(M) .NE. EQUAL ) GO TO 1000                             SET02920
      M = NUMBER(CARD2,COL,NUMVEC,29)                                   SET02930
      IF( J .EQ. LHCD)  XLOW = NUMVEC(30)                               SET02940
      IF( J .EQ. HHCD)  XHGH = NUMVEC(30)                               SET02950
      IF(J.NE.SHCD) GO TO 97                                            SET02960
      XSIZ = NUMVEC(30)                                                 SET02970
      GO TO 97                                                          SET02980
C                                                                       SET02990
1230  M = FIND12(CARD2,COL,SINVEC)                                      SET03000
      IF( SINVEC(M) .NE. EQUAL ) GO TO 1000                             SET03010
      M = NUMBER( CARD2,COL,NUMVEC,29)                                  SET03020
      SPCBAS = NUMVEC(30)                                               SET03030
      GO TO 97                                                          SET03040
C                                                                       SET03050
1240  M = FIND12(CARD2,COL,SINVEC)                                      SET03060
      IF( SINVEC(M) .NE. EQUAL ) GO TO 1000                             SET03070
      M = NUMBER(CARD2,COL,NUMVEC,29)                                   SET03080
      YSIZ = NUMVEC(30)                                                 SET03090
      GO TO 97                                                          SET03100
C                                                                       SET03110
CCCC  DATE CARD                                                         SET03120
CCCC  ------------------------                                         SET03130
C                                                                       SET03140
1400  M = NXTCHR(CARD2,COL)                                             SET03150
      IF( M .EQ. BLANK ) GO TO 200                                      SET03160
```

```
FILE: SETUP1   FORTRAN  A

      READ (30,999) DATE                                                    SET03170
  999 FORMAT (10X,15A4)                                                     SET03180
      REWIND 30                                                             SET03190
      GO TO 200                                                             SET03200
C                                                                           SET03210
CCCC  COMMENT CARD                                                          SET03220
CCCC  ------------------------                                             SET03230
C                                                                           SET03240
 1500 READ (30,999) COMENT                                                  SET03250
      REWIND 30                                                             SET03260
      GO TO 200                                                             SET03270
C                                                                           SET03280
CCCC  HED1 CARD                                                             SET03290
CCCC  ------------------------                                             SET03300
C                                                                           SET03310
 1600 READ (30,999) HED1                                                    SET03320
      REWIND 30                                                             SET03330
      GO TO 200                                                             SET03340
C                                                                           SET03350
CCCC  HED2 CARD                                                             SET03360
CCCC  ------------------------                                             SET03370
C                                                                           SET03380
 1700 READ (30,999) HED2                                                    SET03390
      REWIND 30                                                             SET03400
      GOTO 200                                                              SET03410
C                                                                           SET03420
CCC   DATA FILE CARD                                                        SET03430
C                                                                           SET03440
 1710 M = NXTCHR(CARD2,COL)                                                 SET03450
      IF (M .EQ. BLANK) GO TO 200                                           SET03460
      IF (M .EQ. UHCD) GO TO 1715                                           SET03470
      IF (M .EQ. FHCD) GO TO 1717                                           SET03480
 1713 WRITE(6,753)                                                          SET03490
  753 FORMAT(' ERROR ON DATA FILE CARD')                                    SET03500
      GO TO 200                                                             SET03510
 1715 J = FIND12(CARD2,COL,EQUVEC)                                          SET03520
      IF ( J .EQ. -1)  GO TO 1713                                           SET03530
      M = NUMBER(CARD2,COL,DATAPE,ZERO)                                     SET03540
      COL = COL - 1                                                         SET03550
      GO TO 1710                                                            SET03560
 1717 J = FIND12(CARD2,COL,EQUVEC)                                          SET03570
      IF(J.EQ.-1) GO TO 1713                                                SET03580
      M = NUMBER(CARD2,COL,DATFIL,ZERO)                                     SET03590
      DATFIL = DATFIL - 1                                                   SET03600
      IF (DATFIL .LT. 0)  DATFIL = 0                                        SET03610
      COL = COL - 1                                                         SET03620
      GO TO 1710                                                            SET03630
C                                                                           SET03640
CCC   STAT FILE CARD                                                        SET03650
C                                                                           SET03660
 1720 M = NXTCHR(CARD2,COL)                                                 SET03670
      IF (M .EQ. BLANK) GO TO 200                                           SET03680
      IF (M .EQ. UHCD)  GO TO 1725                                          SET03690
      IF (M .EQ. FHCD)  GO TO 1727                                          SET03700
 1723 WRITE(6,755)                                                          SET03710
  755 FORMAT(' ERROR ON STAT FILE CARD')                                    SET03720
      GO TO 200                                                             SET03730
 1725 J = FIND12(CARD2,COL,EQUVEC)                                          SET03740
      IF ( J .EQ. -1)  GO TO 1723                                           SET03750
      M = NUMBER(CARD2,COL,SAVTAP,ZERO)                                     SET03760
      COL = COL - 1                                                         SET03770
      GO TO 1720                                                            SET03780
 1727 J = FIND12(CARD2,COL,EQUVEC)                                          SET03790
      IF (J .EQ. -1)  GO TO 1723                                            SET03800
      M = NUMBER(CARD2,COL,STAFIL,ZERO)                                     SET03810
      STAFIL = STAFIL - 1                                                   SET03820
      IF (STAFIL .LT. 0) STAFIL = 0                                         SET03830
      COL = COL - 1                                                         SET03840
      GO TO 1720                                                            SET03850
C                                                                           SET03860
CCCC  CALCULATE BASES OF THE ARRAYS                                         SET03870
CCCC  ------------------------------------------------------------------    SET03880
C                                                                           SET03890
 900  CONTINUE                                                              SET03900
      IF( NOSPEC .GT. 20)  NOSPEC = 20                                      SET03910
      IF( NOSPEC .NE. 0)  GO TO 950                                         SET03920
      NOSPEC = (MAXCLS+3)/4                                                 SET03930
      SPCVEC(1,1) = 0                                                       SET03940
                                                                            SET03950
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SETUP1 FORTRAN A

```
  950 VARSIZ = NOFEAT*(NOFEAT+1)/2                                          SET03960
      IF(XSIZ.LE.0) XSIZ=XHIGH-XLOW+1                                       SET03970
      IF(XSIZ.GT.101) XSIZ=101                                              SET03980
      SPEC1=(5*NOSPEC+1)/2*2                                                SET03990
      COVAR1=(VARSIZ+1)/2*2                                                 SET04000
      AVAR1=(NOFEAT*MAXSUB+1)/2*2                                           SET04010
      CLSID1=(MAXSUB+1)/2*2                                                 SET04020
      SUBSV1=(5*MAXSUB+1)/2*2                                               SET04030
      FLMEN1=NOFEAT*2                                                       SET04040
      FLVAR1=VARSIZ*2                                                       SET04050
      SUBMN1=NOFEAT*2                                                       SET04060
      SURVR1=VARSIZ*2                                                       SET04070
      SURSD1=(NOFEAT*MAXSUB+1)/2*2                                          SET04080
      SUBCL1=(MAXSUB+1)/2*2                                                 SET04090
      HFTAL1=(XSIZ*NOHIST+1)/2*2+HFDKEY                                     SET04100
      HSTAL1=(XSIZ*NOHIST+1)/2*2+HSBKEY                                     SET04110
      SIZE=SPEC1+COVAR1+AVAR1+CLSID1+SUBSV1+FLMEN1+FLVAR1+                  SET04120
     * SUBMN1+SUBVR1+SURSD1+SUBCL1+HFTAL1+HSTAL1                            SET04130
      MAXFLD=(TOP-SIZE-32)/32                                               SET04140
      IF(MAXFLD.LE.0) GO TO 1300                                            SET04150
      SPEC1=1                                                               SET04160
      COVAR1=SPEC1+(5*NOSPEC+1)/2                                           SET04170
      AVAR1=COVAR1+(VARSIZ+1)/2                                             SET04180
      CLSID1=AVAR1+(NOFEAT*MAXSUB+1)/2                                      SET04190
      SUBSV1=CLSID1+(MAXSUB+1)/2                                            SET04200
      FLMEN1=SUBSV1+(5*MAXSUB+1)/2                                          SET04210
      FLVAR1=FLMEN1+NOFEAT                                                  SET04220
      SUBMN1=FLVAR1+VARSIZ                                                  SET04230
      SURVR1=SUBMN1+NOFEAT                                                  SET04240
      SURSD1=SURVR1+VARSIZ                                                  SET04250
      SURCL1=SURSD1+(NOFEAT*MAXSUB+1)/2                                     SET04260
      HFTAL1=SUBCL1+(MAXSUB+1)/2                                            SET04270
      HSTAL1=HFTAL1+(XSIZ*NOHIST+1)/2*HFDKEY                                SET04280
      FLDSV1=HSTAL1+(XSIZ*NOHIST+1)/2*HSBKEY                                SET04290
      SAVER1=FLDSV1+(10*MAXFLD)/2                                           SET04300
      TIPTOP=SAVER1+(22*MAXFLD)/2                                           SET04310
      BADCOR=TOP-2*TIPTOP                                                   SET04320
      IF(BADCOR.LT.0) GO TO 1300                                            SET04330
C     PRINT OUT OPTIONS                                                     SET04340
CC    ------------------------------------------------------------------    SET04350
C                                                                           SET04360
      WRITE(6,HEAD)                                                         SET04370
      IF(PCHKEY+SSFKEY+CFDKEY+HSBKEY+HFDKEY+SSLKEY                          SET04380
     1   +SFDKEY+CALKEY .LE. 0 )  GOTO 960                                  SET04390
      WRITE(6,9001)                                                         SET04400
      IF(CFDKEY .EQ.  1)  WRITE(6,9002)                                     SET04410
      IF(SFDKEY .EQ.  1)  WRITE(6,9004)                                     SET04420
      IF(SSLKEY.EQ.1) WRITE(6,9006)                                         SET04430
      IF(PCHKEY .EQ. 1) WRITE(6,9008)                                       SET04440
      IF( HFDKEY .EQ. 1)  WRITE(6,9012)                                     SET04450
      IF(HSBKEY.EQ.1) WRITE(6,9014)                                         SET04460
      IF ( CALKEY .EQ. 1 ) WRITE(6,9016)                                    SET04470
      IF(SSFKEY.EQ.1) WRITE(6,9018)                                         SET04480
C                                                                           SET04490
 9001 FORMAT(1H,'YOU HAVE SELECTED THE FOLLOWING  $STAT  PROCESSOR OPTIO    SET04500
     1NS:'//)                                                               SET04510
 9002 FORMAT(T5,' PRINT MEAN AND COVARIANCE FOR EACH FIELD')                SET04520
 9004 FORMAT(T5,' PRINT SPECTRAL PLOT FOR EACH FIELD')                      SET04530
 9006 FORMAT(T5,' PRINT SPECTRAL PLOT FOR EACH SUBCLASS')                   SET04540
 9008 FORMAT(T5,' PUNCH MEAN AND COVARIANCE MATRIX FOR EACH SUBCLASS')      SET04550
 9012 FORMAT(T5,' PRINT A HISTOGRAM FOR EACH FIELD')                        SET04560
 9014 FORMAT(T5,' PRINT A HISTOGRAM  FOR EACH SUBCLASS')                    SET04570
 9016 FORMAT(T5,' *** USE CALIBRATED DATA ***')                             SET04580
 9018 FORMAT(T5,' PRINT MEAN AND COVARIANCE FOR EACH SUBCLASS')             SET04590
  960 WRITE(6,9502) BADCOR,MAXFLD,MAXSUB,(FETVEC(I),I=1,NOFEAT)            SET04600
 9502 FORMAT('0'//'0SUPERVISOR INFORMATION:'//T5,'UNUSED CORE',I6,' LOCA    SET04610
     *TIONS'/T5,'MAXIMUM NO. OF FIELDS...',I3/T5,'MAXIMUM NO. OF SUBCLASS    SET04620
     2ES...',I3/T5,'CHANNELS SELECTED ARE  ',15(I3,','))/T27,15(I3,',') )   SET04630
      IF( HFDKEY+HCLKEY .NE. 0) WRITE(6,9504)  (HISVEC(I),I=1,NOHIST)       SET04640
 9504 FORMAT(T5,'HISTOGRAM CHANNELS ARE ',15(I3,',')/T28,15(I3,',') )       SET04650
      IF( SPCVEC(1,1) .EQ. 0 )  GOTO 971                                    SET04660
      WRITE(6,9505)                                                         SET04670
 9505 FORMAT(T5,'MULTISPECTRAL PLOTS ARE...')                               SET04680
      DO 970 J =1,NOSPEC                                                    SET04690
      K = SPCVEC(5,J)                                                       SET04700
      WRITE(6,9506)  (SPCVEC(I,J),I=1,K)                                    SET04710
 9506 FORMAT(T44,')',T31,'(',4(I2,','))                                     SET04720
  970  CONTINUE                                                             SET04730
  971 CONTINUE                                                              SET04740
```

FILE: SETUP1 FORTRAN A

```
  980 RETURN                                                                    SET04750
                                                                                SET04760
C      ERROR  ROUTINES                                                        SET04770
CCC    ------------------------------------------------------------------     SET04780
C                                                                             SET04790
 1000 WRITE (6,10002) CODE, CARD2                                             SET04800
10002 FORMAT(///5X,'/////  FROM SUBR. SETUP1  ---  BAD CONTROL CARD ENC      SET04810
     1OUNTERED  ---  INPUT CARD IS ...'//8X,2H'',A4,6A,62A1,2H'' ///)         SET04820
      GO TO 200                                                               SET04830
C                                                                             SET04840
 1300 WRITE (6,1302)                                                          SET04850
 1302 FORMAT(////  5X,'/////  FROM SUBR. SETUP1  ---  DECREASE OPTIONS'      SET04860
     1 // 5X,'***** TERMINATING PROGRAM EXECUTION FROM SUBR. SETUP1  **      SET04870
     2***' /1H1)                                                              SET04880
      GO TO 1305                                                              SET04890
 1303 WRITE(6,10002)  CODE, CARD2                                             SET04900
      WRITE(6,13031)  MAXFET                                                  SET04910
13031 FORMAT(//5X,'CHECK CHANNELS OR SUBCLASS NO.S REQUESTED-CANNOT BE        SET04920
     1 LESS THAN OR EQUAL ZERO, OR GREATER THAN',I5///5X,                     SET04930
     2 '*****  TERMINATING PROGRAM EXECUTION FROM SUBR. SETUP1  *****'       SET04940
     3 /1H1)                                                                  SET04950
                                                                              SET04960
 1305 CALL EXIT                                                               SET04970
      END
```

9. ISOCLS PROCESSOR

See listings for the TESTSP processor (section 23) for an iterative self-organizing clustering procedure using sample values of pixels clustered in packed form on disk storage.

```
FILE: ISOCLS   FORTRAN  A

      SUBROUTINE ISOCLS(ARRAY,TOP)                                          !S000010
C*                                                                          !S000020
C*************************************************************************!S000030
C*                                                                          !S000040
C*    THIS PROGRAM PERFORMS A MODIFIED VERSION OF THE CLUSTERING            !S000050
C*    ALGORITHM (ISODATA) ORIGINALLY DEVELOPED BY BALL AND HALL OF          !S000060
C*    STANFORD RESEARCH INSTITUTE.  THE ALGORITHM HAS BEEN MODIFIED         !S000070
C*    ON THE RECOMMENDATIONS OF ED KAN (LEC).                               !S000080
C*                                                                          !S000090
C*    THE PROGRAM EXPECTS MULTISPECTRAL SCANNER DATA                        !S000100
C*    IN EITHER THE LARSYS 22 OR THE UNIVERSAL                              !S000110
C*    FORMAT.  THE DATA TAPE SHOULD BE ASSIGNED TO FORTRAN UNIT 3.          !S000120
C*                                                                          !S000130
C*************************************************************************!S000140
C*                                                                          !S000150
      IMPLICIT INTEGER (A-Z)                                                !S000160
CCCC  INCLUDE COMMON.LIST                                                   !S000170
CCCC  INCLUDE COMMTS.LIST                                                   !S000180
CCCC  INCLUDE COMMRD.LIST                                                   !S000190
CCCC  INCLUDE CMMKIA.LIST                                                   !S000200
      COMMON/PASS/STOP,LNCAT,NMIN,KPN,STDMAX,DLMIN,SEP,                     !S000210
     *                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,          !S000220
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                          !S000230
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTRD,TOTPTS,            !S000240
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT             !S000250
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                          !S000260
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR         !S000270
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUMVEC(62),NOSUB2,CHNVC(30)         !S000280
     * ,NOCMAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                                  !S000290
     * SYMDO,SYMDU,ITHIGO,ITRIGU,DOFLAG,                                    !S000300
     *  DUFLAG,DODU,STDOTS(60),NSDOTS,SUNCOR(30),LLNCAT,                    !S000310
     * DVCNT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDDU(2)                !S000320
     *  ,MXFET1,MAXPOP                                                      !S000330
      REAL SUNCOR                                                           !S000340
C*                                                                          !S000350
CCCC  COMMON BLOCK 'PASS' IS USED ONLY BY THE ISOCLS PROCESSOR.             !S000360
CCCC                                                                        !S000370
CCCC  ISOCLS USES THE RANDOM ACCESS DRUM FILE AS FOUR DISTINCT FILES.       !S000380
CCCC  SEE DEFINITIONS OF IBEGIN,BEGIN1,BEGIN2,BEGIN3 BELOW                 !S000390
CCCC                                                                        !S000400
CCCC  DEFINITIONS                                                           !S000410
CCCC                                                                        !S000420
CCCC     ISTOP  - MAX. NO. OF ITERATIONS FOR THE CLUSTERING PROCEDURE       !S000430
CCCC              SET IN SETUP7 ROUTINE. (USER INPUT)                       !S000440
CCCC     LNCAT  - CURRENT NO. OF CLUSTERS. SET INITIALLY IN RDFILE OR IS    !S000450
CCCC              ISOCLS. THEN ONLY IN ISODAT.                              !S000460
CCCC     NMIN   - MIN. NO. OF POINTS TO ALLOW PER CLUSTER                   !S000470
CCCC              SET IN SETUP7 ROUTINE. (USER INPUT)                       !S000480
CCCC     KPN    - PRINT CLUSTER SUMMARY EVERY 'KPN' ITERATION(S)           !S000490
CCCC              SET IN SETUP7 ROUTINE. (USER INPUT)                       !S000500
CCCC     STDMAX - STANDARD DEVIATION FOR SPLITTING CLUSTERS                 !S000510
CCCC              SET IN SETUP7 ROUTINE. (USER INPUT)                       !S000520
CCCC     DLMIN  - MIN. DISTANCE BETWEEN CLUSTERS FOR COMBINING.             !S000530
CCCC     SEP    - DISTANCE TO SEPARATE CLUSTERS.  SET EITHER IN SETUP7.     !S000540
CCCC              BY USER INPUT, OR IN ID                                   !S000550
CCCC              BY USER INPUT, OR IN ISODAT.                              !S000560
CCCC     MAP    - PRINT A CLUSTER MAP EVERY 'MAP' ITERATION(S)- SETUP7      !S000570
CCCC     SPTRIG - TRIGGER TELLING WHETHER OR NOT 'SEP' WAS INPUT. -SETUP    !S000580
CCCC     IRD    - NO. OF RECORDS TO READ FROM DATA FILE. COMPUTED IN        !S000590
CCCC              ISOCLS                                                    !S000600
C*       NOPTS  - NO. OF POINTS IN EACH RECORD. COMPUTER IN ISOCLS          !S000610
      CONTINUE                                                              !S000620
C*       KPTS   - NO. OF POINTS IN LAST RECORD. COMPUTER IN ISOCLS          !S000630
CCCC     PUNCH  - TRIGGER TELLING WHETHER OR NOT TO PUNCH THE MODULE        !S000640
CCCC              STAT DECK. - SETUP7                                       !S000650
CCCC     ICHN   - TRIGGER TELLING WHETHER OR NOT CHAINING IS TO BE DONE     !S000660
CCCC     CHNTHS - MIN. DISTANCE BETWEEN CLUSTERS FOR CHAINING - SETUP7      !S000670
CCCC     ICHAIN - ARRAY CONTAINING CHAINED CLUSTER NUMBERS. SET IN          !S000680
CCCC              'CHAIN' ROUTINE.                                          !S000690
CCCC     NWDS   - TOTAL NO. OF WORDS AVAILABLE FOR DRUM STORAGE OF          !S000700
CCCC              IMAGE DATA TO BE CLUSTERED - SET IN ISOCLS                !S000710
CCCC     IBEGIN - BEGINNING DRUM FILE ADDRESS FOR INPUT INITIAL CLUSTER     !S000720
CCCC              CENTERS - SET IN ISOCLS                                   !S000730
CCCC     BEGIN3 - BEGINNING DRUM FILE ADDRESS FOR TEMPORARY STORAGE OF      !S000740
CCCC              CLASS STATISTICS - SET IN ISOCLS ROUTINE                  !S000750
CCCC     BEGIN1 - BEGINNING DRUM FILE ADDRESS FOR IMAGE DATA                !S000760
CCCC     BEGIN2 - BEGINNING DRUM FILE ADDRESS FOR 'IPLACE'. (CLUSTER TO     !S000770
CCCC              WHICH CORRESPONDING POINT BELONGS.)                       !S000780
CCCC     CLSNAM - NAME OF CLASS CURRENTLY BEING PROCESSED. - RDDATA         !S000790
```

FILE: ISOCLS FORTRAN A

```
C*        NOFLD  - NO. OF FIELDS INPUT FOR THIS CLASS - RDDATA                  ISO00800
C*        IPT    - NO. OF WORDS OF STORAGE USED IN 'ARRAY' FOR FIELD AND        ISO00810
C*                 CLASS INFORMATION FOR THIS CLASS. - RDDATA                   ISO00820
C*        TOTWRD - TOTOL WORDS WRITTEN ON DRUM FILE BEGINNING AT ADDRESS        ISO00830
C*                 BEGIN1 - RDDATA                                             ISO00840
C*        TOTPTS - TOTOL POINTS TO BE CLUSTERED FOR CURRENT CLASS - RDDATISO00850
C*        NCLASS - NO. OF CLASSES TO BE CLUSTERED FOR CURRENT CALL TO           ISO00860
C*                 ISOCLS. USER INPUT - SETUP7.                                ISO00870
C*        NOCLS  - CURRENT CLASS NO. - ISOCLS                                   ISO00880
C*        TOTSUB - TOTAL CLUSTERS FOR THIS CALL TO ISOCLS                       ISO00890
C*        TOTFLD - TOTAL FIELDS FOR ALL CLASSES - ISOCLS                        ISO00900
C*        TOTVRT - TOTAL VERTICES FOR ALL FIELDS - ISOCLS                       ISO00910
C*        NOCL   - NO. OF CLASSES SINCE LAST CALL TO SETUP - RDDATA             ISO00920
C*        NVRT   - NO.OF VERTICES FOR ALL FIELDS IN CURRENT CLASS               ISO00930
C*        NXTCLS - NAME OF NEXT CLASS  (RDDATA)                                 ISO00940
C*        MAPFMT - FORMAT OF OUTPUT MAP TAPE                                    ISO00950
C*        NOFEAT - NO. OF CHANNELS USED IN CLUSTERING                           ISO00960
C*        MAXCLS - MAX. NO. OF CLUSTERS PER CLASS                               ISO00970
C*        FETVEC - CHANNELS USED IN CLUSTERING                                  ISO00980
C*        SYMMTX - SYMBOLS FOR PRINTING MAP                                     ISO00990
C*        VARSIZ - SIZE OF EACH CLUSTER'S COVARIANCE MATRIX                     ISO01000
C*        STATKY - FLAG FOR PRINTING COVARIANCES.                               ISO01010
C*      MAPKEY-INDICATES WHETHER TO USE MEAN VECTOR OR CLUSTER NUMBER           ISO01020
C*             IN MAPTAPE                                                       ISO01030
C*      SEQUEN-ARRAY CONTAINING SEQUENCE OF 'S' AND 'C' CHARACTERS FOR          ISO01040
C*             SPLIT-COMBINE ITERATION CONTROL.                                 ISO01050
C*      PERCEN-CUT OFF PERCENTAGE FOR STABILIZED CLUSTERS IN INITIAL            ISO01060
C*             SEQUENCE.                                                        ISO01070
C*      SIMERP-TRIGGER FOR SIMULATING ERIPS CLUSTERING ALGORITHM.               ISO01080
C*             SIMERP=1    SIMULATE ERIPS-SIMERP=0    STANDARD ISOCLS          ISO01090
C*      ORDER--TRIGGER FOR ORDERING THE COLOR KEYS BASE ON THE RANKING          ISO01100
C*             OF GREENNESS OF EACH CLUSTER AS COMPUTED BY THE SUBROUTINE       ISO01110
C*             RANK.SFT IN SETUP7.                                              ISO01120
C*      INUNIT-UNIT NUMBER FOR INPUT STATISTICS FILE.                           ISO01130
C*      INFILE-FILE NUMBER FOR INPUT STATISTICS FILE.                           ISO01140
C*      INITM--TRIGGER FOR INITIAL MEANS FROM STATISTICS FILE.                  ISO01150
C*      PMIN---POPULATION MINIMUM FOR EACH CLUSTER.                             ISO01160
C*             NOFEAT.                                                          ISO01170
C*                                                                              ISO01180
C                                                                               ISO01190
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,                 ISO01200
     *             HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,            ISO01210
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                        ISO01220
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                            ISO01230
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTHFL,HISTFL,PCHUNT,                      ISO01240
     *  CPOUNT,PRTUNT,HAND10                                                    ISO01250
      COMMON/ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP              ISO01260
C$END                                                                           ISO01270
      DIMENSION KVAR(11500)                                                     ISO01280
      KVARDM = 11500                                                            ISO01290
      DIMENSION ARRAY(1)                                                        ISO01300
      DIMENSION COVAR(465)                                                      ISO01310
      DIMENSION NN(60)                                                          ISO01320
      DATA SYMDA /'*    '/,SYMDB /'#    '/                                      ISO01330
      MAXPOP=62                                                                 ISO01340
      MXFET1=30                                                                 ISO01350
      IBEGIN=DRUMAD                                                             ISO01360
C*                                                                              ISO01370
C*    RESERVE ENOUGH DRUM STORAGE FOR MAXIMUM INITIAL MEANS                     ISO01380
C*                                                                              ISO01390
      BEGIN3=IBEGIN + MAXPOP*MXFET1 + MXFET1 + 2                                ISO01400
C*                                                                              ISO01410
C*    CALL SETUP TO READ CARD INPUT AND INITIALIZE DEFAULT VALUES               ISO01420
C*                                                                              ISO01430
      ITIME=1                                                                   ISO01440
      NOCLS = 0                                                                 ISO01450
      TOTFLD = 0                                                                ISO01460
      TOTVRT = 0                                                                ISO01470
      TOTSUB = 0                                                                ISO01480
      CORRAS=1                                                                  ISO01490
      ITRIGU = 0                                                                ISO01500
      ITRIGO=0                                                                  ISO01510
      SYMDO = SYMDA                                                             ISO01520
      SYMDU = SYMDB                                                             ISO01530
      MEANDO = 0                                                                ISO01540
      MEANDU = 255                                                              ISO01550
    1 CALL SETUP7(ARRAY(CORRAS),TOP,ITIME)                                      ISO01560
      IDUM = MAXCLS                                                             ISO01570
      IF(ITIME.GT.1)GO TO 2                                                     ISO01580
```


ORIGINAL PAGE IS
OF POOR QUALITY

```
FILE: ISOCLS   FORTRAN  A

      VARSIZ=NOFEAT*(NOFEAT+1)/2                                                ISO01590
      BEGIN1 = BEGIN3 + NCLASS*MAXPOP*(VARSIZ + NOFEAT + 1)                     ISO01600
      NWDS=DRMWDS-(BEGIN1-DRUMAD)                                               ISO01610
    2 ITIME=ITIME+1                                                             ISO01620
      NOCL=0                                                                    ISO01630
C*                                                                              ISO01640
C     CALL RDDATA  TO COORDINATE READING OF DATA                                ISO01650
C*                                                                              ISO01660
    5 MAXDIM = TOP-CORBAS                                                       ISO01670
      FD1=CORBAS                                                                ISO01680
      CALL RDDATA(ARRAY(FD1),MAXDIM,KVAR,KVARDM,LAST)                           ISO01690
      MAXCLS = IDUM * DDDU                                                      ISO01700
      WRITE(6,210) NDDU(1),NDDU(2)                                              ISO01710
  210 FORMAT(1X,///' DD/DU CLUSTER POP FOR THIS CLASS ',2I7)                    ISO01720
      BEGIN2=BEGIN1 + TOTWRD                                                    ISO01730
      N1 = FD1 + IPT                                                            ISO01740
      MEANS1=N1 + MAXCLS                                                        ISO01750
      STDEV1=MEANS1 + MAXCLS*NOFEAT                                             ISO01760
      TTOP = STDEV1 + MAXCLS*NOFEAT                                             ISO01770
      MAXDIM=TOP-TTOP                                                           ISO01780
      NOPTS = MAXDIM/(NOFEAT+1)                                                 ISO01790
      IDAT1 = TTOP                                                              ISO01800
      IF (NSDOTS.EQ.0) GO TO 4                                                  ISO01810
      DOTDMF = NOCHAN                                                           ISO01820
      TYPSWT = 1                                                                ISO01830
      CALL RDDOTS(ARRAY(MEANS1),STDOTS,NSDOTS,                                  ISO01840
     *  TYPSWT,DOTDMF,DOTDMC,DOTDUM,COVAR,                                      ISO01850
     *  NOCHAN,CHNVC,DTDM,COVAR,                                                ISO01860
     * DOTDM,DOTDM,DOTDM,DOTDM,DOTDM,DOTDM,KVAR)                                ISO01870
      LNCAT = NSDOTS                                                            ISO01880
      DO 500 I = 1,NSDOTS                                                       ISO01890
      DO 500 K = 1,NOFEAT                                                       ISO01900
      III = (I-1)*NOFEAT + K                                                    ISO01910
      II = III + MEANS1 - 1                                                     ISO01920
  500  ARRAY(II) = KVAR(III)                                                    ISO01930
      IF (NOCHAN.EQ.NOFEAT) GO TO 8                                             ISO01940
      WRITE(6,110)                                                              ISO01950
  110 FORMAT(1H ,'NO CHANNELS FOR STARTING NOT EQUAL THAT FOR CLUSTER')         ISO01960
      GO TO 9                                                                   ISO01970
    4 CONTINUE                                                                  ISO01980
       IF(ISOKEY.EQ.1)GO TO 7                                                   ISO01990
C*     SUBVEC-SUBCLASSES FROM STATISTICS FILE FOR INTIAL MEANS.                 ISO02000
C*     NOSUB2-NUMBER OF INITIAL MEANS.                                          ISO02010
C*     CHNVEC-NUMBER OF CHANNELS FROM STATISTICS FILE. NOCHAN MUST EQUAL        ISO02020
       IF(INITM.EQ.1)GO TO 6                                                    ISO02030
       LNCAT=1                                                                  ISO02040
       GO TO 8                                                                  ISO02050
    6  LNCAT=NOSUB2                                                             ISO02060
      CALL GETST(INUNIT,INFILE,ARRAY(MEANS1),DUM,NOSUB2,SUBVEC,NOCHAN           ISO02070
     *  ,CHNVC,ARRAY(TTOP),COVAR,0)                                             ISO02080
      LNCAT = NOSUB2                                                            ISO02090
       GO TO 8                                                                  ISO02100
    7  CONTINUE                                                                 ISO02110
      IF(ISOKEY.EQ.1)CALL RDFILE(ARRAY(MEANS1),ARRAY(TTOP))                     ISO02120
    8  CONTINUE                                                                 ISO02130
      IF(NOPTS.GT.0)GOTO 10                                                     ISO02140
      WRITE(6,100)MAXDIM                                                        ISO02150
  100 FORMAT(' DIMENSION LIMITS EXCEEDED IN ISOCLS BY',I6,                      ISO02160
     *  ' REDUCE CHANNELS OR MAX.CLUSTERS')                                     ISO02170
    9 CALL CMERR                                                                ISO02180
   10 CONTINUE                                                                  ISO02190
      IRD=TOTPTS/NOPTS                                                          ISO02200
      IF(MOD(TOTPTS,NOPTS).EQ.0)GO TO 20                                        ISO02210
      KPTS=MOD(TOTPTS,NOPTS)                                                    ISO02220
      IRD=IRD+1                                                                 ISO02230
      IF(IRD.EQ.1)NOPTS=KPTS                                                    ISO02240
      GO TO 25                                                                  ISO02250
   20 KPTS=NOPTS                                                                ISO02260
   25 CONTINUE                                                                  ISO02270
C*                                                                              ISO02280
C*     CALL ISODAT TO PERFORM CLUSTERING                                        ISO02290
C*                                                                              ISO02300
      A1=1                                                                      ISO02310
      A2=A1+ MAXCLS*NOFEAT                                                      ISO02320
      CLD1=A2 + MAXCLS*NOFEAT                                                   ISO02330
      KPLCE = NOPTS*NOFEAT + IDAT1                                              ISO02340
      CALL ISODAT(ARRAY(IDAT1),ARRAY(KPLCE),ARRAY(MEANS1),ARRAY(N1),            ISO02350
     *            ARRAY(STDEV1),KVAR (CLD1),ARRAY(FD1),KVAR(A1),              ISO02360
     *            KVAR(A2))                                                   ISO02370
```

FILE: ISOCLS FORTRAN A

```
C*                                                                      ISO02380
C*    CHAIN CLUSTERS WHOSE DISTANCES ARE LESS THAN DLMIN                ISO02390
C*                                                                      ISO02400
      LNCAT=LNCAT+DODU                                                  ISO02410
      IF(ICHN.GT.0)CALL CHAIN(KVAR(CLD1))                               ISO02420
C*                                                                      ISO02430
C*    PRINT FINAL RESULTS                                               ISO02440
C*                                                                      ISO02450
      CALL PRINT(-1,ARRAY(KPLCE),ARRAY(MEANS1),ARRAY(STDEV1),           ISO02460
     *           KVAR (CLD1),ARRAY(FD1),ARRAY(N1))                      ISO02470
C*                                                                      ISO02480
C*    CREATE MAP OUTPUT TAPE FOR PMIS DAS IF DESIRED                    ISO02490
C*                                                                      ISO02500
      IF(MAPFMT.GT.0)CALL OSTAPE(ARRAY(KPLCE),KVAR(1),ARRAY(MEANS1),    ISO02510
     *                           ARRAY(FD1))                            ISO02520
      LNCAT=LNCAT-DODU                                                  ISO02530
C*                                                                      ISO02540
C*    CALCULATE COVARIANCE MATRIX FOR EACH CLUSTER                      ISO02550
C*                                                                      ISO02560
      IF(VARSIZ*LNCAT.GT.KVARDM)GO TO 30                                ISO02570
      CALL COVAR1(KVAR,ARRAY(IDAT1),ARRAY(KPLCE),ARRAY(MEANS1),         ISO02580
     *            ARRAY(N1))                                            ISO02590
C*                                                                      ISO02600
      DO 26  II=1,LNCAT                                                 ISO02610
   26 NN(TOTSUB+II)    = ARRAY(N1+II-1)                                 ISO02620
      TOTSUB = TOTSUB + LNCAT                                           ISO02630
      NOCLS = NOCLS + 1                                                 ISO02640
      TOTFLD = TOTFLD + NOFLD                                           ISO02650
      TOTVRT = TOTVRT + NVRT                                            ISO02660
      ARRAY(FD1+1)=IPT + FD1                                            ISO02670
      ARRAY(FD1+2)=LNCAT                                                ISO02680
      ARRAY(FD1+3)=NOFLD                                                ISO02690
C*                                                                      ISO02700
C*    WRITE STATS FOR THESE CLUSTERS ON SCRATCH FILE 18                 ISO02710
C*                                                                      ISO02720
      IF(NOCLS.EQ.1) ADRES=BEGIN3                                       ISO02730
      IN=NOFEAT*LNCAT                                                   ISO02740
      CALL RWRITE(ADRES,ARRAY(MEANS1),IN,JSTAT)                         ISO02750
      ADRES=ADRES+IN                                                    ISO02760
      IN=VARSIZ*LNCAT                                                   ISO02770
      CALL RWRITE(ADRES,KVAR,IN,LSTAT)                                  ISO02780
      ADRES=ADRES+IN                                                    ISO02790
C*    WAIT FOR I/O COMPLETION                                           ISO02800
   60 IF(LSTAT.EQ.1) GO TO 60                                           ISO02810
C*                                                                      ISO02820
C*    GO READ IN ANOTHER CLASS                                          ISO02830
C*                                                                      ISO02840
      CORBAS=CORBAS+IPT                                                 ISO02850
      IF(LAST.NE.1)GO TO 5                                              ISO02860
      IF(NOCLS.LT.NCLASS)GO TO 1                                        ISO02870
C*                                                                      ISO02880
C*    NOW READ SCRATCH FILE AND STORE ON SAVTAP FILE AND PUNCH ON       ISO02890
C*    CARDS IF REQUESTED.                                               ISO02900
C*                                                                      ISO02910
      FLD1 = 1                                                          ISO02920
      VERTX1 = FLD1 + TOTFLD*4                                          ISO02930
      CLSNM1 = VERTX1 + TOTVRT*2                                        ISO02940
      NOSUB1 = CLSNM1 + NOCLS                                           ISO02950
      SUBNM1 = NOSUB1 + NOCLS                                           ISO02960
C                                                                       ISO02970
C     RETRIEVE INFORMATION FROM 'ARRAY'                                 ISO02980
C                                                                       ISO02990
      CALL GETINF(ARRAY(1),KVAR(FLD1),KVAR(VERTX1),KVAR(CLSNM1),        ISO03000
     *     KVAR(NOSUB1),KVAR(SUBNM1),NOCLS,TOTSUB)                      ISO03010
C                                                                       ISO03020
      SWTCH = 1                                                         ISO03030
C                                                                       ISO03040
C     OUTPUT STATS                                                      ISO03050
C                                                                       ISO03060
      CALL LARMAN(SAVTAP,STAFIL,NOCLS,TOTSUB,NOFEAT,TOTFLD,TOTVRT,      ISO03070
     *     FETVEC,KVAR(FLD1),KVAR(VERTX1),KVAR(CLSNM1),KVAR(NOSUB1),    ISO03080
     *  KVAR(SUBNM1),NN,BEGIN3,VARSIZ,PUNCH,DUMMY,STATKY,SWTCH)          ISO03090
      RETURN                                                            ISO03100
   30 KV=KVARDM                                                         ISO03110
      WRITE(6,200)KV                                                    ISO03120
      CALL CMERR                                                        ISO03130
  200 FORMAT(' DIMENSION LIMIT OF',I6,' FOR COVARIANCES EXCEEDED')      ISO03140
      RETURN                                                            ISO03150
      END                                                               ISO03160
```

FILE: CHAIN FORTRAN A

```
      SUBROUTINE CHAIN(CLD)                                                  CHA00010
C*****************************************************************************CHA00020
C*                                                                            *CHA00030
C*    THIS SUBROUTINE CHAINS ALL CLUSTERS WHOSE MEANS ARE LESS THAN           *CHA00040
C*    DLMIN UNITS APART.                                                      *CHA00050
C*         IF - DISTANCE BETWEEN CLUSTERS L AND M < DLMIN                     *CHA00060
C*              DISTANCE BETWEEN CLUSTERS L AND N = DLMIN                     *CHA00070
C*              DISTANCE BETWEEN CLUSTERS M AND N < DLMIN                     *CHA00080
C*         THEN-CLUSTERS L,M, AND N ARE CHAINED                               *CHA00090
C*                                                                            *CHA00100
C*    INPUT  CLD-CLUSTER DISTANCES                                            *CHA00110
C*           DLMIN-MINIMUM DISTANCE BETWEEN CLUSTERS                          *CHA00120
C*           LNCAT-NUMBER OF CLUSTERS                                         *CHA00130
C*                                                                            *CHA00140
C*    OUTPUT   ICHAIN-ARRAY CONTAINING NUMBERS OF CHAINED CLUSTERS            *CHA00150
C*           PRINTED SUMMARY OF CLUSTERS WHICH WERE CHAINED                   *CHA00160
C*****************************************************************************CHA00170
C*                                                                            CHA00180
      IMPLICIT INTEGER (A-X)                                                 CHA00190
C     INCLUDE COMRKS.LIST                                                    CHA00200
C     INCLUDE COMRKA.LIST                                                    CHA00210
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STOMAX,DLMIN,SEP,                      COM00010
     *                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,           COM00020
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                           COM00030
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,            COM00040
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT              COM00050
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                           COM00060
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP          COM00070
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)          COM00080
     * ,NOCHAN,FRCOMP,NOSEQ,MEANDO,MEANDU,                                  COM00090
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                     COM00100
     *  DUFLAG,DODU,STDOTS(60),NSDOTS,SUNCOR(30),LLNCAT,                     COM00110
     * DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)                 COM00120
     *  ,MXFET1,MAXPOP                                                       COM00130
      REAL SUNCOR                                                            COM00140
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,              COM00010
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,        COM00020
     *    DRUMAD,DRMADS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                     COM00030
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                         COM00040
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                   COM00050
     *  CRDUNT,PRTUNT,RANDIO                                                 COM00060
CSEND                                                                        CHA00230
      EQUIVALENCE(SYMBLS,SYMMTX)                                             CHA00240
      DIMENSION JP(62),CLD(MAXCLS,MAXCLS),SYMBLS(62)                         CHA00250
      REAL CHNTHS,CLD                                                        CHA00260
      IHD=0                                                                  CHA00270
      DO 10 I=1,LNCAT                                                        CHA00280
   10 ICHAIN(I) = I                                                          CHA00290
   20 DO 30 I=1,LNCAT                                                        CHA00300
   30 JP(I) = ICHAIN(I)                                                      CHA00310
      I=0                                                                    CHA00320
   40 I=I+1                                                                  CHA00330
      IF (I.GE.LNCAT) GO TO 60                                               CHA00340
      M=I+1                                                                  CHA00350
      DO 50 J=M,LNCAT                                                        CHA00360
      IF(CLD(I,J).GT.CHNTHS)GO TO 50                                         CHA00370
      ICHAIN(I) = MIN0(ICHAIN(I),ICHAIN(J))                                  CHA00380
      ICHAIN(J) = ICHAIN(I)                                                  CHA00390
   50 CONTINUE                                                               CHA00400
      GO TO 40                                                               CHA00410
   60 DO 70 I=1,LNCAT                                                        CHA00420
      IF(ICHAIN(I).NE.JP(I))GO TO 20                                         CHA00430
   70 CONTINUE                                                               CHA00440
      M=1                                                                    CHA00450
      KNCAT=LNCAT                                                            CHA00460
   80 K=0                                                                    CHA00470
      IM=M + 1                                                               CHA00480
      DO 90 I=IM,LNCAT                                                       CHA00490
      IF(ICHAIN(I) .NE. M)      GO TO 90                                     CHA00500
      KNCAT=KNCAT-1                                                          CHA00510
      K=K+1                                                                  CHA00520
      SYMBLS(I)=SYMBLS(M)                                                    CHA00530
      JP(K)=I                                                                CHA00540
   90 CONTINUE                                                               CHA00550
      IF (K.EQ.0) GO TO 100                                                  CHA00560
      IF (IHD.EQ.0) WRITE (6,140)                                            CHA00570
      IF (IHD.EQ.0) WRITE (6,HEAD)                                           CHA00580
      IHD=1                                                                  CHA00590
      WRITE (6,110)M,(JP(I),I=1,K)                                           CHA00600
```

FILE: CHAIN FORTRAN A

```
      WRITE (6,120)M                                                    CHA00610
                                                                        CHA00620
  100 M=M+1                                                             CHA00630
      IF (M.LT.LNCAT) GO TO 80                                          CHA00640
      IF(KNCAT.EQ.LNCAT)RETURN                                          CHA00650
      WRITE (6,130)KNCAT                                                CHA00660
      RETURN                                                            CHA00670
  110 FORMAT(/' THE FOLLOWING CLUSTERS SHOULD BE CHAINED---',20I4)      CHA00680
  120 FORMAT(/' IN THE FINAL OUTPUT MAP ALL OF THE ABOVE CLUSTERS WILL BCHA00690
     *E REPRESENTED BY THE SYMBOL FOR CLUSTER',I4//)                     CHA00700
  130 FORMAT(' THE ABOVE CHAINING REDUCES THE EFFECTIVE NUMBER OF CLUSTECHA00710
     *RS TO ',I5)                                                       CHA00720
  140 FORMAT(1H1)                                                       CHA00730
      END
```

FILE: CLDIST FORTRAN A

```
C*                                                                              CLD00010
C*******************************************************************************CLD00020
C*                                                                              *CLD00030
C*     THIS SUBROUTINE CALCULATES THE WEIGHTED DISTANCE BETWEEN                 *CLD00040
C*     CLUSTER MEANS                                                            CLD00050
C*                                                                              CLD00060
C*                                                                              *CLD00070
C*     INPUT     AMN(MEANS) - MEANS OF EACH FEATURE OF EACH CLUSTER             *CLD00080
C*               STDEV      - STANDARD DEVIATIONS FOR EACH FEATURE/CLUSTER      CLD00090
C*               LNCAT      - NUMBER OF CLUSTERS                               *CLD00100
C*               NOFEAT     - NUMBER OF FEATURES (CHANNELS)                    *CLD00110
C*                                                                             *CLD00120
C*     OUTPUT    CLD        - ARRAY CONTAINING DISTANCE BETWEEN CLUSTERS       *CLD00130
C*                            CLD(N,M)=DISTANCE BETWEEN CLUSTERS N AND M       *CLD00140
C*                                                                              CLD00150
C*******************************************************************************CLD00160
C*                                                                              CLD00170
      SUBROUTINE CLDIST(CLD,STDEV,MEANS)                                        CLD00180
       IMPLICIT INTEGER (A-Z)                                                   CLD00190
C      INCLUDE COMRKS.LIST                                                      CLD00200
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                         COM00010
     *                       MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,           COM00020
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                             COM00030
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,              COM00040
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT                COM00050
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                             COM00060
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP             COM00070
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)             COM00080
     * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                                     COM00090
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                       COM00100
     *  DUFLAG,DODU,STDOTS(60),NSDOTS,SUNCOR(30),LLNCAT,                       COM00110
     * DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)                   COM00120
     *  ,MXFETI,MAXPOP                                                         COM00130
      REAL SUNCOR                                                             COM00140
C$END                                                                         CLD00220
      REAL STDEV,MEANS,CLD                                                    CLD00230
      DIMENSION MEANS(NOFEAT,MAXCLS)                                          CLD00240
      DIMENSION CLD(MAXCLS,MAXCLS),STDEV(NOFEAT,MAXCLS)                       CLD00250
      DO 30 I=1,LNCAT                                                         CLD00260
      CLD(I,I)=0.                                                             CLD00270
      JJ=I+1                                                                  CLD00280
       IF (I.EQ.LNCAT) JJ=I                                                   CLD00290
      DO 20 J=JJ,LNCAT                                                        CLD00300
      CLD(I,J)=0.0                                                            CLD00310
      DO 10 K=1,NOFEAT                                                        CLD00320
      IF(STDEV(K,I).GT.0.0  .AND.   STDEV(K,J).GT.0.0)GO TO 5                 CLD00330
      CLD(I,J)=999.99                                                         CLD00340
      GO TO 15                                                                CLD00350
    5 CONTINUE                                                                CLD00360
      CLD(I,J)=CLD(I,J)+(MEANS(K,I)-MEANS(K,J))**2/(STDEV(K,I)*STDEV(K,J      CLD00370
     *))                                                                      CLD00380
   10 CONTINUE                                                                CLD00390
      CLD(I,J)=SQRT(CLD(I,J))                                                 CLD00400
   15  CLD(J,I) = CLD(I,J)                                                    CLD00410
   20 CONTINUE                                                                CLD00420
   30 CONTINUE                                                                CLD00430
      RETURN                                                                  CLD00440
      END                                                                     CLD00450
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: COVAR1 FORTRAN A

```
      SUBROUTINE COVAR1(COVAR,C,IPLACE,MEANS,N)                                 COV00010
      IMPLICIT INTEGER (A-X)                                                    COV00020
C*                                                                              COV00030
C*    SUBROUTINE COVARR CALCULATES AND PRINTS THE COVARIANCE MATRIX FOR          COV00040
C*    EACH CLUSTER                                                              COV00050
C*                                                                              COV00060
C     INCLUDE COMQKS.LIST                                                       COV00070
C     INCLUDE COMRK6.LIST                                                       COV00080
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                         COV00090
     *                        MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,          COV00100
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                             COV00110
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,              COV00120
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT                COV00130
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                            COV00140
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP             COV00150
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)             COV00160
     * ,NOCHAN,FRCOMP,NOSEQ,MEANDO,MEANDU,                                     COV00170
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                       COV00180
     *  DUFLAG,DODU,STDOTS(60),NSDOTS,SUNCOR(30),LLNCAT,                       COV00190
     * DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDDU(2)                   COV00200
     *  ,MXFFT1,MAXPOP                                                         COV00210
      REAL SUNCOR                                                               COV00220
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,                 COV00230
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,          COV00240
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                       COV00250
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                           COV00260
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                     COV00270
     *  CRDUNT,PRTUNT,RANDIO                                                   COV00280
CSEND                                                                           COV00290
      REAL MEANS,COVAR,C                                                        COV00300
      DIMENSION C(NOFEAT,NOPTS)                                                 COV00310
      DIMENSION COVAR(VARSIZ,LNCAT),IPLACE(NOPTS)                               COV00320
      DIMENSION MEANS(NOFEAT,MAXCLS),N(MAXCLS)                                  COV00330
      DATA CH/'CH'/                                                             COV00340
      DO 10 I=1,LNCAT                                                           COV00350
      DO 10 J=1,VARSIZ                                                          COV00360
   10 COVAR(J,I)=0.0                                                            COV00370
C*                                                                              COV00380
      ADRES1=BEGIN1                                                             COV00390
      ADRES2= BEGIN2                                                            COV00400
      ICCT=NOPTS                                                                COV00410
      IRC=IRD                                                                   COV00420
   20 IF(IRC.LE.1)ICCT=KPTS                                                     COV00430
      IF (IRD.EQ.0) GO TO 30                                                    COV00440
      IWRDS=ICCT*NOFEAT                                                         COV00450
      CALL RREAD(ADRES1,C,IWRDS,ISTAT)                                          COV00460
      ADRES1=ADRES1+IWRDS                                                       COV00470
   22 IF(ISTAT.EQ.1)GO TO 22                                                    COV00480
      CALL RREAD(ADRES2,IPLACE,ICCT,ISTAT)                                      COV00490
      ADRES2=ADRES2+ICCT                                                        COV00500
   25 IF(ISTAT.EQ.1)GO TO 25                                                    COV00510
C*                                                                              COV00520
C*    SINCE THE COVARIANCE MATRIX IS SYMMETRICAL ONLY THE LOWER                 COV00530
C*    TRIANGULAR PORTION OF THE MATRIX IS CALCULATED.                           COV00540
C*                                                                              COV00550
   30 DO 45 I = 1,ICCT                                                          COV00560
      KK=0                                                                      COV00570
      ICLS=IPLACE(I)                                                            COV00580
      IF(ICLS.GT.LNCAT) GO TO 45                                                COV00590
      DO 40 J=1,NOFEAT                                                          COV00600
      DO 40 K=1,J                                                               COV00610
      KK=KK+1                                                                   COV00620
      COVAR(KK,ICLS)=COVAR(KK,ICLS)+C(J,I)*C(K,I)                               COV00630
   40 CONTINUE                                                                  COV00640
   45 CONTINUE                                                                  COV00650
      IRC=IRC-1                                                                 COV00660
      IF (IRC.GT.0) GO TO 20                                                    COV00670
      DO 50 I=1,LNCAT                                                           COV00680
      KK=0                                                                      COV00690
      DO 50 J=1,NOFEAT                                                          COV00700
      DO 50 K=1,J                                                               COV00710
      KK=KK+1                                                                   COV00720
      COVAR(KK,I)=COVAR(KK,I)/N(I) - MEANS(K,I)*MEANS(J,I)                      COV00730
   50 CONTINUE                                                                  COV00740
      RETURN                                                                    COV00750
C     IF(STATKY.NE.1)RETURN                                                     COV00760
   55 WRITE (6,HEAD)                                                            COV00770
      WRITE(6,150)CLSNAM                                                        COV00780
      DO 80 I=1,LNCAT                                                           COV00790
```

FILE: COVAR1 FORTRAN A

```
      WRITE (6,90)I                                                     COV00800
      DO 70 LOC=1,NOFEAT,12                                             COV00810
      ISTOP=LOC+11                                                      COV00820
      IF(ISTOP.GT.NOFEAT)ISTOP=NOFEAT                                   COV00830
      WRITE(6,140)(CH,FETVEC(J),J=LOC,ISTOP)                            COV00840
      II=1                                                              COV00850
      KINC=1                                                            COV00860
      DO 60 J=LOC,NOFEAT                                                COV00870
      K=J*(J+1)/2-II+1                                                  COV00880
      JK=K+KINC-1                                                       COV00890
      WRITE(6,100)(COVAR(M,I),M=K,JK)                                   COV00900
      II=II+1                                                           COV00910
   60 IF(KINC.LT.ISTOP.AND.KINC.LT.12)KINC=KINC+1                       COV00920
      WRITE (6,110)                                                     COV00930
   70 CONTINUE                                                          COV00940
   80 CONTINUE                                                          COV00950
      RETURN                                                            COV00960
   90 FORMAT(//' COVARIANCE MATRIX FOR CLUSTER',I4/)                    COV00970
  100 FORMAT(/6X,12F9.2)                                                COV00980
  110 FORMAT(///)                                                       COV00990
  120 FORMAT(1H1)                                                       COV01000
  140 FORMAT(9X,12(A3,I2,')',3X))                                       COV01010
  150 FORMAT(/' COVARIANCES FOR CLASS',2X,A4//)                         COV01020
      END                                                               COV01030
```

FILE: DESCEN FORTRAN A

```
      SUBROUTINE DESCEN(SCN,LNCAT,PTR1,PTR2)                            DES00010
      IMPLICIT INTEGER(A-Z)                                             DES00020
      DIMENSION PTR1(LNCAT),PTR2(LNCAT)                                 DES00030
      REAL SCN(LNCAT),SAVE                                              DES00040
      J=0                                                               DES00050
 60   J=J+1                                                             DES00060
      IF(J.GT.LNCAT)GO TO 90                                            DES00070
      IF(J.EQ.LNCAT)GO TO 75                                            DES00080
      IF(SCN(J).LT.SCN(J+1))GO TO 70                                    DES00090
      GO TO 60                                                          DES00100
C                                                                       DES00110
 70   SAVE=SCN(J)                                                       DES00120
      SCN(J)=SCN(J+1)                                                   DES00130
      SCN(J+1)=SAVE                                                     DES00140
C                                                                       DES00150
      SAVE1=PTR1(J)                                                     DES00160
      PTR1(J)=PTR1(J+1)                                                 DES00170
      PTR1(J+1)=SAVE1                                                   DES00180
C                                                                       DES00190
      SAVE2=PTR2(J)                                                     DES00200
      PTR2(J)=PTR2(J+1)                                                 DES00210
      PTR2(J+1)=SAVE2                                                   DES00220
 75   K=J                                                               DES00230
 80   IF(K.EQ.1)GO TO 60                                                DES00240
      IF(SCN(K).LT.SCN(K-1))GO TO 60                                    DES00250
C                                                                       DES00260
      SAVE=SCN(K-1)                                                     DES00270
      SCN(K-1)=SCN(K)                                                   DES00280
      SCN(K)=SAVE                                                       DES00290
C                                                                       DES00300
      SAVE1=PTR1(K-1)                                                   DES00310
      PTR1(K-1)=PTR1(K)                                                 DES00320
      PTR1(K)=SAVE1                                                     DES00330
C                                                                       DES00340
      SAVE2=PTR2(K-1)                                                   DES00350
      PTR2(K-1)=PTR2(K)                                                 DES00360
      PTR2(K)=SAVE2                                                     DES00370
      K=K-1                                                             DES00380
      GO TO 80                                                          DES00390
 90   CONTINUE                                                          DES00400
      RETURN                                                            DES00410
      END                                                               DES00420
```

FILE: DSTAPE FORTRAN A

```
C*****************************************************************************DST00010
C*                                                                           DST00020
C*        ISOCLS SUBROUTINE                                                  DST00030
C*                                                                           DST00040
C*      THIS SUBROUTINE GENERATES A CLUSTER IMAGE TAPE IN EITHER             DST00050
C*      UNIVERSAL OR LARSYS II FORMAT. THE IMAGE MAY OPTIONALLY BE A         DST00060
C*      ONE CHANNEL TAPE REFLECTING THE CLUSTER NUMBER OF EACH PIXEL,        DST00070
C*      OR A 'NOFEAT' CHANNEL TAPE REFLECTING THE MEAN VECTOR OF THE         DST00080
C*      CLUSTER TO WHICH THE PIXEL WAS ASSIGNED.                             DST00090
C*                                                                           DST00100
C*                                                                           DST00110
      SUBROUTINE DSTAPE(IPLACE,IBUF,MEANS,/FLDINF/)                        DST00120
      IMPLICIT INTEGER(A-Z)                                                DST00130
C     INCLUDE COMMON.LIST                                                  DST00140
C     INCLUDE COMMON.LIST                                                  DST00150
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                    DST00160
     *                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,         DST00170
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                         DST00180
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,          DST00190
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT            DST00200
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                         DST00210
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR         DST00220
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SURVEC(62),NOSUB2,CHNVC(30)         DST00230
     * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                                 DST00240
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                   DST00250
     *  DUFLAG,DODU,STDOTS(60),NSDOTS,SUNCOR(30),LLNCAT,                   DST00260
     * DVENT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)               DST00270
     *  ,MXFET1,MAXPOP                                                     DST00280
      REAL SUNCOR                                                          DST00290
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,           DST00300
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,      DST00310
     *    DRUMAD,DRUMOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                   DST00320
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                       DST00330
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                 DST00340
     *  CPOUNT,PRTUNT,HANDIO                                               DST00350
CSEND                                                                      DST00360
      DIMENSION IBUF(1),FETVC2(30)                                         DST00370
      REAL MEANS(1)                                                        DST00380
      NC=1                                                                 DST00390
      IF(MAPKEY.EQ.1)NC=NOFEAT                                             DST00400
       WRITE(6,HEAD)                                                       DST00410
C*                                                                         DST00420
      DIMENSION FINF(6),FL(22)                                             DST00430
      DIMENSION IPTT(62)                                                   DST00440
      EQUIVALENCE (FINF(1),LINSTR),(FINF(4),SAMSTR)                        DST00450
      EQUIVALENCE (FINF(2),LINEND),(FINF(5),SAMEND)                        DST00460
      EQUIVALENCE (FINF(3),LININC),(FINF(6),SAMINC)                        DST00470
      DIMENSION IPLACE(NOPTS)                                              DST00480
      DIMENSION FLDINF(1)                                                  DST00490
      DIMENSION FORM(3,2)                                                  DST00500
      DATA FORM/'UNIV','ERSA','L   ','LARS','YS I','I   '/                 DST00510
       ICOUNT=0                                                            DST00520
       DO 3 I=1,LNCAT                                                      DST00530
    3   IPTT(I)=I                                                          DST00540
      IF(ICHN.GT.0)GO TO 5                                                 DST00550
      DO 1 I=1,LNCAT                                                       DST00560
    1 ICHAIN(I)=I                                                          DST00570
    5 CONTINUE                                                             DST00580
      ADRES=BEGIN2                                                         DST00590
      K=0                                                                  DST00600
      IREC=IRD                                                             DST00610
      IPTS=NOPTS                                                           DST00620
   10 IF(IREC.LE.1)IPTS=KPTS                                               DST00630
      IF (IRD.EQ.0) GO TO 20                                               DST00640
      CALL MREAD(ADRES,IPLACE,IPTS,ISTAT)                                  DST00650
   15 IF(ISTAT.EQ.1)GO TO 15                                               DST00660
      ADRES=ADRES+IPTS                                                     DST00670
   20 CONTINUE                                                             DST00680
      IV=5                                                                 DST00690
      DO 50 IFLD=1,NOFLD                                                   DST00700
      NOREC = 1                                                            DST00710
      NV=FLDINF(IV+1)                                                      DST00720
      IB=IV+2+NV+2                                                         DST00730
      DO 25 I=1,6                                                          DST00740
   25 FINF(I)=FLDINF(IB+I)                                                 DST00750
      LINES= (LINEND-LINSTR)/LININC +1                                     DST00760
      LPTS = (SAMEND-SAMSTR)/SAMINC +1                                     DST00770
      NPTS=NC*LPTS                                                         DST00780
      IF(NPTS.GT.11500)GO TO 80                                            DST00790
```

9-12

62

FILE: DSTAPE FORTRAN A

```
      LINE = LINSTR-LININC                                                  DST00800
      DO 40 I=1,LINES                                                       DST00810
C*    ZERO IBUF                                                             DST00820
C*                                                                          DST00830
       DO 26 J=1,NPTS                                                       DST00840
   26 IBUF(J)=0                                                             DST00850
      LINE = LINE + LININC                                                  DST00860
      CALL FDLINT(FLOINF(IV+2),NV,FL,LINE,SAMPS,NFL)                        DST00870
      DO 31 L=1,NFL,2                                                       DST00880
      IB=(FL(L)-SAMSTR)/SAMINC+1                                            DST00890
      IE=(FL(L+1)-SAMSTR)/SAMINC + 1                                        DST00900
      IF(MOD(SAMSTR,SAMINC).NE.MOD(FL(L),SAMINC))IB=IB+1                    DST00910
      IF(IB.GT.IE)GO TO 31                                                  DST00920
      DO 30 J=IB,IE                                                         DST00930
      K=K+1                                                                 DST00940
      KP=IPLACE(K)                                                          DST00950
      IF(MAPKEY.EQ.1)GO TO 70                                               DST00960
      IBUF(J)=ICHAIN(KP)                                                    DST00970
      GO TO 75                                                              DST00980
70    DO 71 LZ=1,NC                                                         DST00990
      JJ=(LZ-1)*LPTS+J                                                      DST01000
      KK=(KP-1)*NOFEAT+LZ                                                   DST01010
   71 IBUF(JJ) = MEANS(KK) + 0.5                                            DST01020
75    IF(K.LT.IPTS)GO TO 30                                                 DST01030
      IREC=IREC-1                                                           DST01040
      IF (IREC.EQ.0) GO TO 30                                               DST01050
      IF(IREC.EQ.1)IPTS=KPTS                                                DST01060
      CALL MREAD(ADRES,IPLACE,IPTS,ISTAT)                                   DST01070
      ADRES=ADRES+IPTS                                                      DST01080
   27 IF(ISTAT.EQ.1)GO TO 27                                                DST01090
      K=0                                                                   DST01100
   30 CONTINUE                                                              DST01110
   31 CONTINUE                                                              DST01120
      GO TO (35,37),HDREC                                                   DST01130
C                                                                           DST01140
CC    WRITE HEADER RECORD                                                   DST01150
C                                                                           DST01160
35    CONTINUE                                                              DST01170
      LSTLIN = 0                                                            DST01180
      LNES = 0                                                              DST01190
      IF(MAPKEY.EQ.1)GO TO 72                                               DST01200
      FETVC2(1)=1                                                           DST01210
      GO TO 73                                                              DST01220
72    DO 74 KK=1,NOFEAT                                                     DST01230
74    FETVC2(KK)=KK                                                         DST01240
73    HDREC=2                                                               DST01250
      NOFILE = NOFILE + 1                                                   DST01260
      CALL WRTHED(NC,FETVC2,LPTS,MAPFMT,MAPUNT)                             DST01270
C                                                                           DST01280
CC    WRITE DATA RECORD                                                     DST01290
C                                                                           DST01300
 37   LNES = LNES + 1                                                       DST01310
       IF(LNES.EQ.LINES.AND.MAPKEY.NE.1)LSTLIN=-1                           DST01320
      CALL WRTLN(IBUF,LSTLIN)                                               DST01330
 40   CONTINUE                                                              DST01340
       IF(MAPKEY.NE.1)GO TO 48                                              DST01350
       IF(IORDER.EQ.1)CALL RANK(NOFEAT,FETVC2,LNCAT,MEANS,IPTT)             DST01360
      DO 41 I=1,NPTS                                                        DST01370
   41 IBUF(I)=0                                                             DST01380
      CALL WRTLN(IBUF,LSTLIN)                                               DST01390
      NTEN=10                                                               DST01400
      JTEN=NTEN                                                             DST01410
      NCLUS=LNCAT                                                           DST01420
      DIV=11                                                                DST01430
      II=1                                                                  DST01440
      NRL=LPTS/DIV                                                          DST01450
      IF(NRL.LE.0) NRL=1                                                    DST01460
      IF(LPTS.LT.11) JTEN=LPTS                                              DST01470
      IF(LPTS.LT.11) DIV=LPTS                                               DST01480
   51 CNT=NRL                                                               DST01490
      IF((NCLUS-NRL).LT.0) CNT=NCLUS                                        DST01500
      NCLUS=NCLUS-NRL                                                       DST01510
      IST=NRL*(II-1)+1                                                      DST01520
      IEND=IST+CNT-1                                                        DST01530
      II=II+1                                                               DST01540
      III=0                                                                 DST01550
      DO 42 I=IST,IEND                                                      DST01560
      III=III+1                                                             DST01570
      DO 42 K=1,NC                                                          DST01580
```

FILE: DSTAPE FORTRAN A

```
       KK=(IPTT(I)-1)*NC*K                                              DST01590
       DO 42 J=1,JTEN                                                   DST01600
       L=(III-1)*DIV*(K-1)*LPTS*J                                       DST01610
    42 IBUF(L) = MEANS(KK) + 0.5                                        DST01620
       DO 43 I=1,NTEN                                                   DST01630
       ICOUNT=ICOUNT+1                                                  DST01640
    43 CALL WRTLN(IBUF,LSTLIN)                                          DST01650
       DO 44 I=1,NPTS                                                   DST01660
    44 IBUF(I)=0                                                        DST01670
       IF(NCLUS.LE.0) LSTLIN=-1                                         DST01680
       CALL WRTLN(IBUF,LSTLIN)                                          DST01690
       IF(NCLUS.GT.0) GO TO 51                                          DST01700
    48 CONTINUE                                                         DST01710
       WRITE(6,60) NOFILE,FLDINF(IV),(FORM(I,MAPFMT),I=1,3),LNES        DST01720
      *,ICOUNT                                                          DST01730
   60  FORMAT(///T55,'FILE NO.     - ',I6,/T55,'FIELD NAME  - ',A4,/     DST01740
      *  T55,'FORMAT       - ',3A4,/T55,'NO. OF SCAN LINES - ',I6,/T55,  DST01750
      *'NO. OF COLOR KEY SCAN LINES - ',I6)                              DST01760
       IV = IV + NV*2 + 9                                               DST01770
    50 CONTINUE                                                         DST01780
       RETURN                                                           DST01790
80     WRITE(6,81)                                                      DST01800
81     FORMAT('THE NUMBER OF CHANNELS TIMES THE NUMBER OF SAMPLES HAS   DST01810
      1EXCEEDED 11500.DECREASE THE NUMBER OF CHANNELS OR THE NUMBER OF  DST01820
      2SAMPLES.TERMINATING RUN FROM DSTAPE')                            DST01830
       CALL  CMEAR                                                      DST01840
       END                                                              DST01850
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: GETINF FORTRAN A

```
      SUBROUTINE GETINF(ARRAY,FLOSAV,VERTEX,CLSNMS,NOSUBS,SUBNM,NOCLS,
     *   TOTSUB)
C
      IMPLICIT INTEGER (A-Z)
      DIMENSION ARRAY(1),FLOSAV(4,1),VERTEX(1),CLSNMS(1),NOSUBS(1),
     *   SUBNM(1),CLSVEC(60)
C
      JJ = 0
      KPT = 1
      NFS = 0
      L = 0
C
C     SEE SUBROUTINE RODATA FOR STORAGE ARRANGEMENT OF 'ARRAY'
C
      DO 80  CLS=1,NOCLS
      L = L + NFS
      CLSNMS(CLS) = ARRAY(KPT)
      NOSUBS(CLS) = ARRAY(KPT+2)
      NFS = ARRAY(KPT+3)
      IKP = KPT + 4
C
      DO 100 I=1,NFS
      FLOSAV(1,I+L) = ARRAY(IKP)
      FLOSAV(2,I+L) = CLS
      FLOSAV(3,I+L) = 0
      FLOSAV(4,I+L) = ARRAY(IKP+1)
C
      NV = FLOSAV(4,I+L)+2
C
      DO 90  J=1,NV
   90 VERTEX(JJ+J) = ARRAY(IKP+1+J)
      JJ = NV + JJ
C
  100 CONTINUE
C
      IKP = IKP + NV + 9
C
      KPT = ARRAY(KPT+1)
C
   80 CONTINUE
C
      K = 0
      DO 120  I=1,NOCLS
      NSUB = NOSUBS(I)
C
      DO 120  J=1,NSUB
      K = K + 1
  120 CLSVEC(K) = I
C
      CALL NAMSTA(SUBNM,CLSVEC,NOSUBS,TOTSUB,CLSNMS,NOCLS)
C
      RETURN
C
      END
```

FILE: ISODAT FORTRAN A

```
      SUBROUTINE ISODAT(C,IPLACE,MEANS,N,STDEV,CLD,FLDINF,AVP,AMN)        ISO00010
       IMPLICIT INTEGER (A-Z)                                             ISO00020
        IMPLICIT INTEGER (A-Z)                                            ISO00030
C        INCLUDE COMMK5.LIST                                              ISO00040
C         INCLUDE COMMK6.LIST                                             ISO00050
CMS360                                                                    ISO00060
CMS160                                                                    ISO00070
C         INCLUDE CMBK16.LIST                                             ISO00080
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                   ISO00090
     *                    MAP,SPTRIG,    IND,  KPTS, NOPTS, PUNCH,       ISO00100
     *  ICHN,CHNTHS,ICHAIN(62),NMOS,IBEGIN,BEGIN1,                        ISO00110
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTRD,TOTPTS,          ISO00120
     *              NCLASS,NOCLS,TOTSUM,TOTFLD,TOTVRT,NUCL,NVRT           ISO00130
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                        ISO00140
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SINGRP        ISO00150
     *,IORDER,IMUNIT,INFILE,INITM,PMIN,SUMVEC(62),NOSUM2,CHNVC(30)        ISO00160
     * ,NOCHAN,EPCOMP,NOSEQ,MEANDO,MEANDU,                               ISO00170
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                  ISO00180
     *  DUFLAG,DODU,STOPTS(50),NSOPTS,SUNCOR(30),LLNCAT,                  ISO00190
     * DVERT(250,2),DMECT(60,2),DVPNT(11,2),IUCNT(2),NDOU(2)              ISO00200
     *  ,MAFFIL,MAXPOP                                                   ISO00210
      REAL SUNCOR                                                         ISO00220
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,           ISO00230
     *              HISFIL,HISKEY,TRFORM,ERIPTR,ERRKEY,MAPUNT,NOFILE,     ISO00240
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  ISO00250
     *  ,NMSTUN,NMSTFI,SCTWIN,MAPFIL                                      ISO00260
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTAFL,HISTFL,PCHUNT,                ISO00270
     *  CRDUNT,PRTUNT,HANDIO)                                            ISO00280
      COMMON/ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP        ISO00290
CSEND                                                                     ISO00300
      EQUIVALENCE (SGMIN,STDMAX)                                          ISO00310
       REAL MEANS,STDEV,STDMAX,SEP,AVP,C,AMN,SGMA,RND,                    ISO00320
     *TEST,DMIN,DLMIN,CLD,TIME,PERCEN,DIJ                                 ISO00330
       REAL ESUM,ESQT,MEAN(30,62),SDIJ                                    ISO00340
      LOGICAL DEL                                                         ISO00350
      DIMENSION AVP(NOFEAT,MAXCLS),ISGMA(62)                              ISO00360
      DIMENSION C(NOFEAT,NOPTS),IPLACE(NOPTS)                             ISO00370
      DIMENSION AMN(NOFEAT,MAXCLS),SGMA(62)                               ISO00380
      DIMENSION MEANS(NOFEAT,MAXCLS),N(MAXCLS)                            ISO00390
      DIMENSION STDEV(NOFEAT,MAXCLS),CLD(MAXCLS,MAXCLS)                   ISO00400
      DIMENSION FLDINF(1)                                                 ISO00410
       REAL SDUM                                                          ISO00420
      DIMENSION PTR(62)                                                   ISO00430
      DATA SS/'S'/,CC/'C'/                                                ISO00440
      EQUIVALENCE (KDIM,NOFEAT),(LNCAT,INCAT)                             ISO00450
      DEL=.FALSE.                                                         ISO00460
       ISEQ=0                                                             ISO00470
      MAXCL = MAXCLS - DODU                                               ISO00480
      IDUM=LNCAT+DODU-MAXCLS                                              ISO00490
      IF(IDUM.GT.0) LNCAT=LNCAT-IDUM                                      ISO00500
      ISTOP=STOP                                                          ISO00510
      SPLFIN=0                                                            ISO00520
      KKT=1                                                               ISO00530
       DO 5 I = 1,30                                                      ISO00540
 5      SUNCOR(I) = 1.                                                    ISO00550
       IF (ISUNC.NE.0.OR.ISUNT.NE.0) CALL SUNFAC(SUNCOR,SUNANG,           ISO00560
     *  FETVEC,NOFEAT,ISUNC,ISUNT)                                       ISO00570
C*                                                                        ISO00580
       LK=K                                                               ISO00590
C*     ASSIGN DATA TO CLUSTERS                                            ISO00600
C*                                                                        ISO00610
   10  CONTINUE                                                           ISO00620
       LLNCAT = LNCAT + DODU                                              ISO00630
       IF(DOFLAG.EQ.0) GO TO 12                                           ISO00640
       DO 11 J=1,NOFEAT                                                   ISO00650
 11    MEANS(J,LNCAT+1)=MEANDO                                            ISO00660
 12    IF(DUFLAG.EQ.0) GO TO 14                                           ISO00670
       DO 13 J=1,NOFEAT                                                   ISO00680
 13    MEANS(J,LLNCAT) = MEANDU                                           ISO00690
 14     CONTINUE                                                          ISO00700
       DO 15 K=1,LLNCAT                                                   ISO00710
       DO 15 J=1,NOFEAT                                                   ISO00720
   15  MEAN(J,K) = MEANS(J,K)                                             ISO00730
       IF (LNCAT.LE.1.AND.KKT.GT.1) GO TO 530                             ISO00740
       CALL PSPLIT(MEANS,STDEV,N,CLD,C,IPLACE,AVP,AMN,MEANS)              ISO00750
C      CALL CLOCK(TIME)                                                   ISO00760
C      IF(MOD(KKT,KRN).EQ.0)WRITE(6,120)KKT,TIME                          ISO00770
C 120 FORMAT(' CUMULATIVE TIME AFTER ASSIGNING DATA TO CLUSTERS FOR ITER  ISO00780
C    *ATION',I4,' IS',F10.6)                                             ISO00790
```

FILE: ISODAT FORTRAN A

```
      IF(ERCOMP.NE.1)GO TO 135                                          ISO000800
      ESUM=0.0                                                          ISO000810
      DO 132 J=1,NOFEAT                                                 ISO000820
      DO 132 K=1,LNCAT                                                  ISO000830
      ESUM=ESUM+N(K)*(STDEV(J,K))**2/TOTPTS                             ISO000840
  132 CONTINUE                                                          ISO000850
      ESQT=SQRT(ESUM/NOFEAT)                                            ISO000860
      WRITE(6,133) ESQT,PERCEN,STDMAX                                   ISO000870
  133 FORMAT(1X,///' ERCOMP= ',F7.3,' PERCEN = ',F5.3,' STDMAX = ',     ISO000880
     *F7.3/)                                                            ISO000890
C*                                                                      ISO000900
C*    CALCULATE DISTANCES BETWEEN CLUSTER CENTERS                       ISO000910
C*                                                                      ISO000920
  135  CALL CLDIST(CLD,STDEV,MEANS)                                     ISO000930
C*                                                                      ISO000940
C*    IF STOP EQUALS ZERO RETURN AFTER INITIAL CLASSIFICATION           ISO000950
C*                                                                      ISO000960
      IF (ISTOP .EQ. 0) RETURN                                          ISO000970
      LNCAT=LLNCAT                                                      ISO000980
      IF (MOD(KKT,MAP)) 150,140,150                                     ISO000990
  140 CALL PRINT(KKT,IPLACE,MEANS,STDEV,CLD,FLDINF,N)                   ISO001000
      GO TO 161                                                         ISO001010
  150 IF (MOD(KKT,KRN))161,160,161                                      ISO001020
  160 CONTINUE                                                          ISO001030
      CALL PRINT(KKT,IPLACE,MEANS,STDEV,CLD,FLDINF,N)                   ISO001040
  161 CONTINUE                                                          ISO001050
      LNCAT=LLNCAT-DODU                                                 ISO001060
                                                                        ISO001070
CC    FOR ITERATION N CHECK N(K) AGAINST PMIN * NOFEAT                  ISO001080
C                                                                       ISO001090
      IF (ISEQ .NE. NOSEQ) GO TO 169                                    ISO001100
      ISTOP = 0                                                         ISO001110
  162  DO 163 K = 1,LNCAT                                               ISO001120
       IF (N(K) - (PMIN * NOFEAT)) 167,163,163                          ISO001130
  163  CONTINUE                                                         ISO001140
       IF (.NOT.DEL) RETURN                                             ISO001150
       DO 164 KK=1,LNCAT                                                ISO001160
       DO 164 KKK=1,NOFEAT                                              ISO001170
  164  MEANS(KKK,KK) = MEAN(KKK,KK)                                     ISO001180
       CALL PSPLIT(MEANS,STDEV,N,CLD,C,IPLACE,AVP,AMN,MEANS)            ISO001190
       DO 165 KK=1,LNCAT                                                ISO001200
       DO 165 KKK=1,NOFEAT                                              ISO001210
  165  MEAN(KKK,KK) = MEANS(KKK,KK)                                     ISO001220
       CALL CLDIST(CLD,STDEV,MEANS)                                     ISO001230
      RETURN                                                            ISO001240
  167 WRITE(6,168)K,N(K),PMIN,NOFEAT                                    ISO001250
  168 FORMAT(/' CLUSTER',I3,' REMOVED FOR HAVING ONLY',I6,' POINTS.'/   ISO001260
     *' MIN. POINTS IS  (',I4,' * ',I2,')')                             ISO001270
      RETF=1                                                            ISO001280
       LK=K                                                             ISO001290
      GO TO 570                                                         ISO001300
  171  K=LK                                                             ISO001310
      DEL = .TRUE.                                                      ISO001320
      GO TO 162                                                         ISO001330
  169 CONTINUE                                                          ISO001340
  170 CONTINUE                                                          ISO001350
C                                                                       ISO001360
CC    ON ITERATIONS 1 THRU N-1 CHECK N(K) AGAINST NMIN                  ISO001370
C                                                                       ISO001380
      DO 180 K=1,LNCAT                                                  ISO001390
      IF (N(K)-NMIN) 190,180,180                                        ISO001400
  180 CONTINUE                                                          ISO001410
      IF(DEL)CALL CLDIST(CLD,STDEV,MEANS)                               ISO001420
      GO TO 220                                                         ISO001430
  190 IF (MOD(KKT,KRN)) 200,195,200                                     ISO001440
  195    WRITE (6,210)K,N(K),NMIN                                       ISO001450
  200  RETF=2                                                           ISO001460
      LK=K                                                              ISO001470
       GO TO 570                                                        ISO001480
  201  K=LK                                                             ISO001490
       DEL=.TRUE.                                                       ISO001500
      GO TO 170                                                         ISO001510
  210 FORMAT('0 CLUSTER ',I2,' REMOVED FOR HAVING  ONLY ',I6,           ISO001520
     2       ' ELEMENTS. MIN. NO. ELEMENTS IS ',I6)                     ISO001530
  220 CONTINUE                                                          ISO001540
                                                                        ISO001550
C                                                                       ISO001560
C                                                                       ISO001570
C*    SPLIT ITERATION                                                   ISO001580
```

FILE: ISODAT FORTRAN A

```
C                                                                               ISO01590
C                                                                               ISO01600
      DO 225 I=1,INCAT                                                          ISO01610
  225 PTR(I)=I                                                                  ISO01620
      ISPLT=0                                                                   ISO01630
      DO 260 K=1,INCAT                                                          ISO01640
C*                                                                              ISO01650
C*    FIND MAXIMUM STANDARD DEVIATION PER CLUSTER                               ISO01660
C*                                                                              ISO01670
      SGMA(K) = 0.                                                              ISO01680
      DO 250 J = 1,NOFEAT                                                       ISO01690
      SDUM = STDEV(J,K)*SUNCOR(J)                                               ISO01700
      IF (SDUM - SGMA(K)) 250,240,240                                           ISO01710
  240 ISGMA(K) = J                                                              ISO01720
      SGMA(K) = SDUM                                                            ISO01730
  250 CONTINUE                                                                  ISO01740
      IF(SGMA(K).GE.STDMAX)ISPLT=ISPLT+1                                        ISO01750
  260 CONTINUE                                                                  ISO01760
      IF (2*LNCAT.GT.MAXCL) CALLDESCEN(SGMA,LNCAT,ISGMA,PTR)                    ISO01770
      TEST=FLOAT(ISPLT)/FLOAT(LNCAT)                                            ISO01780
      IF(TEST.LE.PERCEN)SPLFIN=1                                                ISO01790
      IF (KKT .GT. ISTOP) SPLFIN = 1                                            ISO01800
      IF (SPLFIN .EQ. 0) GO TO 270                                              ISO01810
      IF (MOD(KKT,KRN) .EQ. 0) WRITE(6,503)                                     ISO01820
  503 FORMAT( /)                                                                ISO01830
      IF(MOD(KKT,KRN).EQ.0)WRITE(6,502)                                         ISO01840
  502 FORMAT(1X,'USER INPUT*SPLIT-COMBINE SEQUENCE OF ITERATIONS')              ISO01850
      ISEQ=ISEQ+1                                                               ISO01860
      IF (SEQUEN(ISEQ).EQ.SS) GO TO 270                                         ISO01870
      IF (SEQUEN(ISEQ).EQ.CC) GO TO 410                                         ISO01880
C                                                                               ISO01890
CC    IS SPLITTING REQUIRED                                                     ISO01900
C                                                                               ISO01910
  270 K=1                                                                       ISO01920
      NCAT=INCAT                                                                ISO01930
  280 IF (K-NCAT) 290,290,500                                                   ISO01940
  290 IF (STDMAX-SGMA(K)) 300,300,310                                           ISO01950
  300 IF (N(K)-(NMIN+NMIN+2)) 310,310,320                                       ISO01960
  310 K=K+1                                                                     ISO01970
      GO TO 280                                                                 ISO01980
C                                                                               ISO01990
CC    SPLIT CLUSTER K                                                           ISO02000
C                                                                               ISO02010
  320 TRIG1=1                                                                   ISO02020
      DEL=.TRUE.                                                                ISO02030
      KX=ISGMA(K)                                                               ISO02040
  330 INCAT=INCAT+1                                                             ISO02050
      LLNCAT=LLNCAT+1                                                           ISO02060
      IF(LLNCAT.LE.MAXCLS) GO TO 350                                            ISO02070
      IF(MOD(KKT,KRN).EQ.0)WRITE(6,340)KKT                                      ISO02080
  340 FORMAT(/' MAXIMUM CLUSTERS ON ITERATION',I4/' SPLITTING REQUIRED BISO02090
     *UT NOT PERFORMED'/)                                                       ISO02100
      LNCAT = MAXCL                                                             ISO02110
      LLNCAT=MAXCLS                                                             ISO02120
      GO TO 500                                                                 ISO02130
  350 INC=INCAT                                                                 ISO02140
      LL=PTR(K)                                                                 ISO02150
  360 DO 370 I=1,KDIM                                                           ISO02160
  370 AMN(I,INC)=AMN(I,LL)                                                      ISO02170
  380 AMN(KX,LL)=AMN(KX,LL)+SEP*SGMA(K)                                         ISO02180
      AMN(KX,INC)=AMN(KX,INC)-SEP*SGMA(K)                                       ISO02190
      SGMA(K)=0.0                                                               ISO02200
      IF (MOD(KKT,KRN)) 400,401,400                                             ISO02210
  401 WRITE(6,390)LL,KX,INC                                                     ISO02220
  390 FORMAT('0 CLUSTER ',I2,' IS SPLIT IN THE ',I2,'TH PARAMETER INTO C ISO02230
     2LUSTER ',I2)                                                              ISO02240
  400 CONTINUE                                                                  ISO02250
      K=K+1                                                                     ISO02260
      GO TO 280                                                                 ISO02270
C                                                                               ISO02280
CC    EVEN ITERATION                                                            ISO02290
CC                                                                              ISO02300
CC    ARE CLUSTERS TO BE COMBINED                                               ISO02310
C                                                                               ISO02320
  410 CONTINUE                                                                  ISO02330
      DO 405 L=1,LNCAT                                                          ISO02340
  405 PTR(L)=I                                                                  ISO02350
C                                                                               ISO02360
      NOCOMB=0                                                                  ISO02370
```

FILE: ISODAT FORTRAN A

```
      NOCLST=LNCAT-1                                                    ISO02380
      L=-1                                                              ISO02390
406   L=L+2                                                             ISO02400
      IF (L.GT.NOCLST) GO TO 480                                        ISO02410
      NOCLTR = LNCAT - 1                                                ISO02420
      KK=0                                                              ISO02430
      DMIN=DLMIN                                                        ISO02440
      DO 430 I=1,NOCLTR                                                 ISO02450
C                                                                       ISO02460
      IF(PTR(I).EQ.0)GO TO 430                                          ISO02470
      II=I+1                                                            ISO02480
      DO 425 J=II,LNCAT                                                 ISO02490
      IF(PTR(J).EQ.0)GO TO 425                                          ISO02500
      SDIJ = 0.0                                                        ISO02510
      DO 420 JJ=1,KDIM                                                  ISO02520
      SDIJ=SDIJ+((AMN(JJ,I)-AMN(JJ,J))**2/(STDEV(JJ,I)*STDEV(JJ,J)))   ISO02530
420   CONTINUE                                                          ISO02540
      DIJ=SQRT(SDIJ)                                                    ISO02550
C                                                                       ISO02560
      IF(DIJ.GT.DMIN)GO TO 425                                          ISO02570
      DMIN=DIJ                                                          ISO02580
      KK=I                                                              ISO02590
      KKK=J                                                             ISO02600
425   CONTINUE                                                          ISO02610
430   CONTINUE                                                          ISO02620
C                                                                       ISO02630
      IF(KK.EQ.0)GO TO 480                                              ISO02640
      PTR(KK)=0                                                         ISO02650
C                                                                       ISO02660
C       COMBINE CLUSTERS KK AND KKK                                     ISO02670
C                                                                       ISO02680
      DEL=.TRUE.                                                        ISO02690
      RND=1.0 /FLOAT(N(KK)+N(KKK))                                      ISO02700
C                                                                       ISO02710
      DO 460 K=1,KDIM                                                   ISO02720
460   AMN(K,KK)=(N(KK)*AMN(K,KK)+N(KKK)*AMN(K,KKK))*RND                ISO02730
C                                                                       ISO02740
      RETF=3                                                            ISO02750
      LK=KKK                                                            ISO02760
      GO TO 570                                                         ISO02770
461   KKK=LK                                                            ISO02780
      IF (KKK.EQ.(LNCAT+1)) GO TO 435                                   ISO02790
C                                                                       ISO02800
C     MOVE POINTERS UP                                                  ISO02810
C                                                                       ISO02820
      DO 175 K=KKK,LNCAT                                                ISO02830
175   PTR(K) = PTR(K+1)                                                 ISO02840
C                                                                       ISO02850
435   IF(MOD(KKT,KPN))440,441,440                                       ISO02860
441     WRITE(6,490)KK,KKK,KK                                           ISO02870
440   IF (L.LT.NOCLST) GO TO 406                                        ISO02880
C                                                                       ISO02890
480   CONTINUE                                                          ISO02900
490 FORMAT('  CLUSTERS ',I2,' AND ',I2,' HAVE BEEN COMBINED INTO CLUST  ISO02910
    2ER ',I2)                                                           ISO02920
C*                                                                      ISO02930
C*    REINITIALIZE                                                      ISO02940
C*                                                                      ISO02950
500 CONTINUE                                                            ISO02960
      DO 510 J=1,MAXCLS                                                 ISO02970
      SGMA(J)=0.0                                                       ISO02980
      ISGMA(J)=0                                                        ISO02990
      DO 510 K=1,KDIM                                                   ISO03000
      AVP(K,J)=0.0                                                      ISO03010
      STDEV(K,J)=0.0                                                    ISO03020
      MEANS(K,J)=AMN(K,J)                                               ISO03030
      AMN(K,J)=0.0                                                      ISO03040
510 CONTINUE                                                            ISO03050
      KKT=KKT+1                                                         ISO03060
      DEL=.FALSE.                                                       ISO03070
      GO TO 10                                                          ISO03080
C                                                                       ISO03090
530 IF (KKT.NE.2) GO TO 550                                             ISO03100
      WRITE (6,540)                                                     ISO03110
540 FORMAT(' THE ORIGINAL CLUSTER WAS NOT SPLIT - EXAMINE THE INPUT VA  ISO03120
    *LUE FOR STDMAX')                                                   ISO03130
      KKT=1                                                             ISO03140
      ISTOP=0                                                           ISO03150
      GO TO 10                                                          ISO03160
```

FILE: ISODAT FORTRAN A

```
  550 WRITE (6,560)KKT                                                          ISO03170
  560 FORMAT(//' AFTER ',I4,' ITERATIONS ALL DATA HAS BEEN ASSIGNED TO O ISO03180
     *NE CLUSTER'/)                                                             ISO03190
      KKT=1                                                                     ISO03200
      ISTOP=0                                                                   ISO03210
      GO TO 10                                                                  ISO03220
  570 CONTINUE                                                                  ISO03230
C*                                                                              ISO03240
C       ROUTINE TO DELETE A CLUSTER                                             ISO03250
C*                                                                              ISO03260
      INCAT=INCAT-1                                                             ISO03270
      LLNCAT=LLNCAT-1                                                           ISO03280
      IF (LK.EQ.(INCAT+1)) GO TO (171,201,461),RETF                             ISO03290
       DO 561 J=LK,INCAT                                                        ISO03300
       DO 552 L=1,KOIM                                                          ISO03310
      AMN(L,J)=AMN(L,J+1)                                                       ISO03320
      MEANS(L,J)=MEANS(L,J+1)                                                   ISO03330
      MEAN(L,J) = MEAN(L,J+1)                                                   ISO03340
  552  STDEV(L,J)=STDEV(L,J+1)                                                  ISO03350
      N(J)=N(J+1)                                                               ISO03360
  561  CONTINUE                                                                 ISO03370
       GO TO (171,201,461),RETF                                                 ISO03380
      END                                                                       ISO03390
```


ORIGINAL PAGE IS
OF POOR QUALITY

FILE: PRINT FORTRAN A

```
      SUBROUTINE PRINT(KKT,IPLACE,MEANS,STDEV,CLD,FLDINF,N)                 PRI00010
      IMPLICIT INTEGER(A-X)                                                 PRI00020
C     INCLUDE COMMKS.LIST                                                   PRI00030
C     INCLUDE COMMK6.LIST                                                   PRI00040
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                     PRI00050
     *                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,          PRI00060
     *  ICHN,CHNTHS,ICHAIN(42),NWDS,IBEGIN,BEGIN1,                         PRI00070
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,          PRI00080
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT           PRI00090
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                        PRI00100
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP        PRI00110
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)        PRI00120
     * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                                PRI00130
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                  PRI00140
     *  DUFLAG,DODU,STDOTS(60),NSDOTS,SUNCOR(30),LLNCAT,                  PRI00150
     * DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)              PRI00160
     *  ,MXFET1,MAXPOP                                                    PRI00170
      REAL SUNCOR                                                         PRI00180
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,           PRI00190
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,     PRI00200
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  PRI00210
     *  ,NHSTUN,NHSTFI,SETRUN,MAPFIL                                      PRI00220
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                PRI00230
     *  CRDUNT,PRTUNT,RANDIO                                              PRI00240
CSEND                                                                     PRI00250
      DIMENSION IPLACE(NOPTS)                                             PRI00260
      DIMENSION MEANS(NOFEAT,MAXCLS),STDEV(NOFEAT,MAXCLS)                 PRI00270
      DIMENSION SYMBLS(1),FL(12)                                          PRI00280
      DIMENSION FLDINF(1)                                                 PRI00290
      DIMENSION CLD(MAXCLS,1),N(MAXCLS),NBLK(62),FINF(6)                  PRI00300
      REAL MEANS,STDEV,CLD                                                PRI00310
      DIMENSION COL(3,110),OUT(110)                                       PRI00320
      EQUIVALENCE (FINF(1),LINSTR),  (FINF(4),SAMSTR),                    PRI00330
     *            (FINF(2),LINEND),  (FINF(5),SAMEND),                    PRI00340
     *            (FINF(3),LININC),  (FINF(6),SAMINC)                     PRI00350
      EQUIVALENCE(SYMMTX,SYMBLS)                                          PRI00360
      EQUIVALENCE(COL(1,1),OUT(1)), (COL(1,111),NBLK)                     PRI00370
      DATA BLANK/' '/                                                     PRI00380
      IF(DODU.EQ.0) GO TO 20                                              PRI00390
      SAVEP=SYMMTX(LNCAT-DODU+DOFLAG)                                     PRI00400
      SAVEB=SYMMTX(LNCAT-DODU+DODU)                                       PRI00410
      IF (DOFLAG.NE.0) SYMMTX(LNCAT-DODU + 1) = SYMDO                     PRI00420
      IF (DUFLAG.NE.0) SYMMTX(LNCAT) = SYMDU                              PRI00430
 20   CONTINUE                                                            PRI00440
      WRITE (6,HEAD)                                                      PRI00450
      IF(KKT.GT.0)WRITE(6,240)KKT                                         PRI00460
      IF(KKT.LT.0)WRITE(6,245)CLSNAM                                      PRI00470
      WRITE (6,250)LNCAT                                                  PRI00480
      TOTPTS=TOTWRD/NOFEAT                                                PRI00490
      WRITE(6,260)TOTPTS                                                  PRI00500
      WRITE (6,270)                                                       PRI00510
      DO 30 J=1,LNCAT                                                     PRI00520
      WRITE (6,280)J,SYMBLS(J),N(J)                                       PRI00530
   30 CONTINUE                                                            PRI00540
      WRITE (6,290)                                                       PRI00550
      ISTART = 1                                                          PRI00560
      IEND = 12                                                           PRI00570
      LOOPCT = NOFEAT / 12                                                PRI00580
      LOOPC1 = MOD(NOFEAT,12)                                             PRI00590
      IF (LOOPC1 .GT. 0)  LOOPCT = LOOPCT + 1                             PRI00600
      IF(LOOPCT .EQ. 1)  IEND = NOFEAT                                    PRI00610
      DO 45  M=1,LOOPCT                                                   PRI00620
      WRITE(6,300) (BLANK,FETVEC(J),J=ISTART,IEND)                        PRI00630
      DO 40  J=1,LNCAT                                                    PRI00640
      WRITE(6,310)J,(MEANS(I,J),I=ISTART,IEND)                            PRI00650
   40 CONTINUE                                                            PRI00660
      WRITE(6,315)                                                        PRI00670
  315 FORMAT( )                                                           PRI00680
      ISTART = IEND + 1                                                   PRI00690
      IEND = ISTART + IEND - 1                                            PRI00700
      IF (IEND .GT. NOFEAT)  IEND = NOFEAT                                PRI00710
   45 CONTINUE                                                            PRI00720
      ISTART = 1                                                          PRI00730
      IEND = 12                                                           PRI00740
      IF(LOOPCT .EQ. 1)  IEND = NOFEAT                                    PRI00750
      WRITE( 6,320)                                                       PRI00760
      DO 55  M=1,LOOPCT                                                   PRI00770
      WRITE(6,300) (BLANK,FETVEC(J),J=ISTART,IEND)                        PRI00780
      DO 50  J=1,LNCAT                                                    PRI00790
```

FILE: PRINT FORTRAN A

```
      WRITE(6,310)J,(STDEV(I,J),I=ISTART,IEND)                          PRI00800
   50 CONTINUE                                                          PRI00810
      WRITE(6,315)                                                      PRI00820
      ISTART = IEND + 1                                                 PRI00830
      IEND = ISTART + IEND - 1                                          PRI00840
      IF (IEND .GT. NOFEAT)  IEND = NOFEAT                              PRI00850
   55 CONTINUE                                                          PRI00860
      L=1                                                               PRI00870
      J=LNCAT                                                           PRI00880
      IF(J.GT.15)J=15                                                   PRI00890
   60 WRITE (6,340)(K,K=L,J)                                            PRI00900
      DO 70 I=1,LNCAT                                                   PRI00910
   70 WRITE (6,350)I,(CLD(I,K),K=L,J)                                   PRI00920
      IF (J.EQ.LNCAT) GO TO 80                                          PRI00930
      L=L+15                                                            PRI00940
      J=J+15                                                            PRI00950
      IF(J.GE.LNCAT)J=LNCAT                                             PRI00960
      GO TO 60                                                          PRI00970
   80 CONTINUE                                                          PRI00980
      IF (KKT.EQ.-1) GO TO 90                                           PRI00990
      IF(MOD(KKT, MAP).NE.0)RETURN                                      PRI01000
   90 CONTINUE                                                          PRI01010
      IRC=IRD                                                           PRI01020
      ICCT=NOPTS                                                        PRI01030
      IF(IRD.EQ.0)ICCT=KPTS                                             PRI01040
      IF (IRD.EQ.0) GO TO 110                                           PRI01050
      ADRES2=BEGIN2                                                     PRI01060
      CALL BREAD(ADRES2,IPLACE,ICCT,ISTAT)                              PRI01070
  105 IF(ISTAT.EQ.1)GO TO 105                                           PRI01080
      ADRES2=ADRES2+ICCT                                                PRI01090
      IREC=1                                                            PRI01100
  110 JPTS=0                                                            PRI01110
      IRC=IRC-1                                                         PRI01120
      CALL SETMPG(66,0,66)                                              PRI01130
      IV=5                                                              PRI01140
      DO 200 IFLD=1,NOFLD                                               PRI01150
C*    ZERO NRLK                                                         PRI01160
      DO 115 I=1,LNCAT                                                  PRI01170
  115 NRLK(I)=0                                                         PRI01180
      FLDNAM = FLDINF(IV)                                               PRI01190
      NV= FLDINF(IV+1)                                                  PRI01200
      IB=IV+2*NV+2                                                      PRI01210
      DO 116 I=1,6                                                      PRI01220
  116 FINF(I)=FLDINF(IB+I)                                              PRI01230
      J=0                                                               PRI01240
  120 DO 130 I=SAMSTR,SAMEND,SAMINC                                     PRI01250
      J=J+1                                                             PRI01260
      COL(1,J)=I/100                                                    PRI01270
      COL(2,J)=MOD(I,100)/10                                            PRI01280
      COL(3,J)=MOD(I,10)                                                PRI01290
      IF (J.EQ.110) GO TO 140                                           PRI01300
  130 CONTINUE                                                          PRI01310
  140 LPTS=J                                                            PRI01320
      WRITE (6,220)                                                     PRI01330
      WRITE (6,HEAD)                                                    PRI01340
      TPTS=FLDINF(IV+NV*2+2)                                            PRI01350
      WRITE(6,330)FLDNAM,TPTS                                           PRI01360
      DO 150 I=1,3                                                      PRI01370
  150 WRITE (6,210)(COL(I,J),J=1,LPTS)                                  PRI01380
      WRITE (6,220)                                                     PRI01390
      DO 180 I=LINSTR,LINEND,LININC                                     PRI01400
C*    FIND FIELD INTERSECTIONS FOR THIS LINE                            PRI01410
      CALL FDLINT(FLDINF(IV+2),NV,FL,I,PTS,NFL)                         PRI01420
      DO 155 J=1,110                                                    PRI01430
  155 OUT(J)=BLANK                                                      PRI01440
      DO 175 IJ=1,NFL,2                                                 PRI01450
      IB=(FL(IJ)-SAMSTR)/SAMINC +1                                      PRI01460
      IE=(FL(IJ+1)-SAMSTR)/SAMINC + 1                                   PRI01470
      IF(MOD(SAMSTR,SAMINC).NE.MOD(FL(IJ),SAMINC))IB=IB+1               PRI01480
      IF(IB.GT.IE)GO TO 175                                             PRI01490
      DO 170 J=IB,IE                                                    PRI01500
      JPTS=JPTS+1                                                       PRI01510
      IF (JPTS.LE.ICCT) GO TO 160                                       PRI01520
      IF(IRC.EQ.1)ICCT=KPTS                                             PRI01530
      CALL BREAD(ADRES2,IPLACE,ICCT,ISTAT)                              PRI01540
      ADRES2=ADRES2+ICCT                                                PRI01550
  156 IF(ISTAT.EQ.1)GO TO 156                                           PRI01560
      IRC=IRC-1                                                         PRI01570
      JPTS=1                                                            PRI01580
```

FILE: PRINT FORTRAN A

```
  160 CONTINUE                                                          PRI01590
      K=IPLACE(JPTS)                                                    PRI01600
      NBLK(K)=NBLK(K)+1                                                 PRI01610
      LINE=I                                                            PRI01620
      IF (J.GT.110) GO TO 170                                           PRI01630
      OUT(J)=SYMBLS(K)                                                  PRI01640
  170 CONTINUE                                                          PRI01650
  175 CONTINUE                                                          PRI01660
      WRITE (6,230)LINE,(OUT(J),J=1,LPTS)                               PRI01670
  180 CONTINUE                                                          PRI01680
      IV=IV + NV*2 + 9                                                  PRI01690
      WRITE (6,370)                                                     PRI01700
      DO 190 I=1,LNCAT                                                  PRI01710
  190 WRITE(6,380)I,SYMBLS(I),NBLK(I)                                   PRI01720
  200 CONTINUE                                                          PRI01730
      IF(DODU.EQ.0) GO TO 205                                           PRI01740
      SYMMTX(LNCAT-DODU+DOFLAG)=SAVEP                                   PRI01750
      IF (DOFLAG.NE.0) SYMMTX(LNCAT) = SAVEB                            PRI01760
  205 CONTINUE                                                          PRI01770
      CALL SETMPG(66,4,62)                                              PRI01780
      RETURN                                                            PRI01790
  210 FORMAT(9X,110I1)                                                  PRI01800
  220 FORMAT(/)                                                         PRI01810
  230 FORMAT(2X,I5,2X,110A1)                                            PRI01820
  240 FORMAT(///' INTERMEDIATE PRINTOUT FOR ITERATION',I5//)            PRI01830
  245 FORMAT(///' FINAL CLUSTER SUMMARY FOR CLASS',1X,A4//)             PRI01840
  250 FORMAT(///' TOTAL NUMBER OF CLUSTERS =',I3)                       PRI01850
  260 FORMAT(/' TOTAL NUMBER OF POINTS =',I7)                           PRI01860
  270 FORMAT(///' CLUSTER    SYMBOL    POINTS IN CLUSTER')              PRI01870
  280 FORMAT(4X,I2,9X,A1,10X,I7)                                        PRI01880
  290 FORMAT(////15X,'MEANS'/)                                          PRI01890
  300 FORMAT(/2X,'CLUSTER',5X,12(A1,'CH(',I2,')',1X))                   PRI01900
  310 FORMAT(5X,I2,7X,12(F7.2,1X))                                      PRI01910
  320 FORMAT(///10X,' STANDARD DEVIATIONS'/)                            PRI01920
  330 FORMAT(//2X,A4,//' TOTAL NUMBER OF POINTS IN THIS FIELD',I7/      PRI01930
     *)                                                                 PRI01940
  340 FORMAT(//15X,'DISTANCES BETWEEN CLUSTERS'//1X,'CLUSTER',15I8)     PRI01950
  350 FORMAT(3X,I2,5X,15F8.2)                                           PRI01960
  370 FORMAT(//2X,'POINTS PER CLUSTER IN THIS FIELD'/3X,'CLUSTER',      PRI01970
     *  5X,'SYMBOL',5X,'POINTS'/)                                      PRI01980
  380 FORMAT(6X,I2,10X,A1,7X,I5)                                        PRI01990
      END                                                               PRI02000
```

~~9~~ 73

FILE: PSPLIT FORTRAN A

```
      SUBROUTINE PSPLIT(MEANS,STDEV,N,CLD,C,IPLACE,AVP,AMN,MEN)          PSP00010
      IMPLICIT INTEGER (A-Z)                                             PSP00020
C                                                                        PSP00030
C       INCLUDE COMRK5.LIST                                              PSP00040
C       INCLUDE CMGK16.LIST                                              PSP00050
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STOMAX,OLMIN,SEP,                  PSP00060
     *                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,       PSP00070
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                      PSP00080
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,       PSP00090
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT         PSP00100
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                      PSP00110
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR      PSP00120
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)      PSP00130
     * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                             PSP00140
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                               PSP00150
     *  DUFLAG,DODU,STDOTS(40),NSDOTS,SUNCOR(30),LLNCAT,               PSP00160
     * DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)           PSP00170
     *  ,MXFET1,MAXPOP                                                 PSP00180
      REAL SUNCOR                                                      PSP00190
      COMMON/ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP     PSP00200
CSEND                                                                  PSP00210
      DIMENSION C(NOFEAT,NOPTS),IPLACE(NOPTS),AMN(NOFEAT,MAXCLS)       PSP00220
      DIMENSION STDEV(NOFEAT,MAXCLS),CLD(MAXCLS,MAXCLS),N(MAXCLS)      PSP00230
      DIMENSION AVP(NOFEAT,MAXCLS),MEANS(NOFEAT,MAXCLS)                PSP00240
      REAL MEN(NOFEAT,MAXCLS)                                          PSP00250
     REAL AMN,STDEV,AVP,SDIST,DIST,C,RND,MEANS                         PSP00260
      DIMENSION CSUN(30)                                               PSP00270
      REAL CSUN                                                        PSP00280
        REAL DUM,DUMA                                                  PSP00290
        DUM = .00001                                                   PSP00300
        ICOUNT = 0                                                     PSP00310
        IDELET = 0                                                     PSP00320
 2      ICOUNT = ICOUNT + 1                                            PSP00330
        IDELET = 0                                                     PSP00340
C                                                                      PSP00350
      IF(DOFLAG.NE.0) N(LNCAT+1)=NDOU(DOFLAG)                          PSP00360
      IF(DUFLAG.NE.0) N(LNCAT+DODU)=NDOU(DODU)                         PSP00370
       DO 5 I=1,LNCAT                                                  PSP00380
       N(I)=0                                                          PSP00390
       DO 5 J=1,NOFEAT                                                 PSP00400
       AMN(J,I)=0.0                                                    PSP00410
      STDEV(J,I) = 0.0                                                 PSP00420
 5     AVP(J,I)=0.0                                                    PSP00430
C*                                                                     PSP00440
C*      ASSIGN DATA TO CLUSTERS                                        PSP00450
C*                                                                     PSP00460
       ADRES1=BEGIN1                                                   PSP00470
       ADRES2=BEGIN2                                                   PSP00480
       ICCT=NOPTS                                                      PSP00490
       IRC=IRD                                                         PSP00500
 20    IF(IRC.LE.1)ICCT=KPTS                                           PSP00510
       IF(IRD.EQ.0)GO TO 40                                            PSP00520
       IWRDS=NOFEAT*ICCT                                               PSP00530
       CALL RREAD(ADRES1,C,IWRDS,ISTAT)                                PSP00540
       ADRES1=ADRES1+IWRDS                                             PSP00550
 25    IF(ISTAT.EQ.1)GO TO 25                                          PSP00560
       IF(ISTAT.EQ.0)GO TO 40                                          PSP00570
       IF(ISTAT.GE.0)GO TO 40                                          PSP00580
       WRITE(6,30)ISTAT                                                PSP00590
 30    FORMAT(' ERROR READING DRUM---ISTAT=',I4)                       PSP00600
 40    CONTINUE                                                        PSP00610
      IF (ISUNT.EQ.0.AND.ISUNC.EQ.0) GO TO 50                          PSP00620
      DO 49 I=1,ICCT                                                   PSP00630
      IF(DODU.EQ.0) GO TO 42                                           PSP00640
      DO 41 K=1,NOFEAT                                                 PSP00650
       CDUM = C(K,I)                                                   PSP00660
      IF (CDUM.NE.MEANDO.AND.CDUM.NE.MEANDU) GO TO 42                  PSP00670
 41   CONTINUE                                                         PSP00680
      IF (CDUM.EQ.MEANDO) IPLACE(I) = LNCAT + 1                        PSP00690
      IF (CDUM.EQ.MEANDU) IPLACE(I) = LNCAT + DODU                     PSP00700
      GO TO 49                                                         PSP00710
 42   KK=1                                                             PSP00720
      SDIST=10.0E+20                                                   PSP00730
      DO 46 J=1,LNCAT                                                  PSP00740
      DIST=0.                                                          PSP00750
      DO 44 K=1,NOFEAT                                                 PSP00760
      CSUN(K)=C(K,I)                                                   PSP00770
 44   DIST=DIST+ABS(MEANS(K,J)-CSUN(K))*SUNCOR(K)                      PSP00780
       IF (DIST - SDIST) 45,46,46                                      PSP00790
```

FILE: PSPLIT FORTRAN A

```
 45    KK=J                                                                    PSP00800
       SDIST=DIST                                                              PSP00810
 46    CONTINUE                                                                PSP00820
 47    CONTINUE                                                                PSP00830
       N(KK)=N(KK)+1                                                           PSP00840
       IPLACE(I)=KK                                                            PSP00850
       DO 48 K=1,NOFEAT                                                        PSP00860
       AMN(K,KK)=AMN(K,KK)+CSUN(K)                                             PSP00870
       AVP(K,KK)=AVP(K,KK)+CSUN(K)**2                                          PSP00880
 48    CONTINUE                                                                PSP00890
 49    CONTINUE                                                                PSP00900
       GO TO 101                                                               PSP00910
 50    DO 100 I = 1,ICCT                                                       PSP00920
        KK=1                                                                   PSP00930
       IF(DODU.EQ.0) GO TO 52                                                  PSP00940
       DO 51 K=1,NOFEAT                                                        PSP00950
       CDUM=C(K,I)                                                             PSP00960
       IF (CDUM.NE.MEANDO.AND.CDUM.NE.MEANDU) GO TO 52                         PSP00970
 51    CONTINUE                                                                PSP00980
       IF (CDUM.EQ.MEANDO) IPLACE(I) = LNCAT + 1                               PSP00990
       IF (CDUM.EQ.MEANDU) IPLACE(I) = LNCAT + DODU                            PSP01000
       GO TO 100                                                               PSP01010
 52    CONTINUE                                                                PSP01020
       KK = 1                                                                  PSP01030
        SDIST=10.0E+20                                                         PSP01040
        DO 70 J=1,LNCAT                                                        PSP01050
        DIST=0.0                                                               PSP01060
       DO 55 K = 1,NOFEAT                                                      PSP01070
       CSUN(K) = C(K,I)                                                        PSP01080
 55    DIST = DIST + ABS(MEANS(K,J) - CSUN(K))                                 PSP01090
        IF(DIST-SDIST)60,70,70                                                 PSP01100
 60     KK=J                                                                   PSP01110
        SDIST=DIST                                                             PSP01120
 70     CONTINUE                                                               PSP01130
 80     CONTINUE                                                               PSP01140
        N(KK)=N(KK)+1                                                          PSP01150
        IPLACE(I)=KK                                                           PSP01160
        DO 90 K=1,NOFEAT                                                       PSP01170
       AMN(K,KK) = AMN(K,KK) + CSUN(K)                                         PSP01180
       AVP(K,KK) = AVP(K,KK) + CSUN(K)**2                                      PSP01190
 90     CONTINUE                                                               PSP01200
 100    CONTINUE                                                               PSP01210
 101   CONTINUE                                                                PSP01220
        IF(IRD.EQ.0)GO TO 110                                                  PSP01230
        CALL RWRITE(ADRES2,IPLACE,ICCT,ISTAT)                                  PSP01240
        ADRES2=ADRES2+ICCT                                                     PSP01250
 105    IF(ISTAT.EQ.1)GO TO 105                                                PSP01260
 110    IRC=IRC-1                                                              PSP01270
        IF(IRC.GT.0)GO TO 20                                                   PSP01280
          KA = 1                                                               PSP01290
  115     CONTINUE                                                             PSP01300
       DO 130 K=KA,LNCAT                                                       PSP01310
          KAA = K                                                              PSP01320
          IF (N(K).EQ.0) GO TO 140                                             PSP01330
       RND=FLOAT(N(K))                                                         PSP01340
       DO 130 J=1,NOFEAT                                                       PSP01350
       AMN(J,K)=AMN(J,K)/RND                                                   PSP01360
       MEANS(J,K)=AMN(J,K)                                                     PSP01370
       STDEV(J,K)=SQRT(AVP(J,K)/RND-AMN(J,K)*AMN(J,K))                         PSP01380
        DUMA = STDEV(J,K)                                                      PSP01390
        IF (DUMA.LT.DUM)   STDEV(J,K) = DUM                                    PSP01400
 130     CONTINUE                                                              PSP01410
         IF (ICOUNT.GT.2) RETURN                                               PSP01420
         IF (IDELET.NE.0) GO TO 2                                              PSP01430
         RETURN                                                                PSP01440
 140     LNCAT = LNCAT - 1                                                     PSP01450
         LLNCAT = LLNCAT - 1                                                   PSP01460
          IDELET = IDELET + 1                                                  PSP01470
         KAAA = KAA - 1                                                        PSP01480
         IF (KAAA.EQ.LNCAT) GO TO 160                                          PSP01490
         DO 150 K = KAA,LNCAT                                                  PSP01500
         N(K) = N(K+1)                                                         PSP01510
         DO 150 J=1,NOFEAT                                                     PSP01520
         AMN(J,K) = AMN(J,K+1)                                                 PSP01530
         AVP(J,K) = AVP(J,K+1)                                                 PSP01540
 150      CONTINUE                                                             PSP01550
 160     IF (DOFLAG.NE.0) N(LNCAT+1) = NDOU(DOFLAG)                            PSP01560
         IF (DOFLAG.NE.0) N(LNCAT + DODU) = NDOU(DODU)                         PSP01570
         IF (DOFLAG.EQ.0) GO TO 170                                            PSP01580
```

FILE: PSPLIT FORTRAN A

```
          DO 165 J = 1,NOFEAT                                          PSP01590
 165      MEANS(J,LNCAT+1) = MEANDO                                    PSP01600
 170      IF (DUFLAG.EQ.0) GO TO 180                                   PSP01610
          DO 175 J = 1,NOFEAT                                          PSP01620
 175      MEANS(J,LLNCAT) = MEANDU                                     PSP01630
 180      CONTINUE                                                     PSP01640
          KA = KAA                                                     PSP01650
          IF (KAAA.NE.LNCAT) GO TO 115                                 PSP01660
C*                                                                     PSP01670
C                                                                      PSP01680
          IF (ICOUNT.GT.2) RETURN                                      PSP01690
          IF (IDELET.NE.0) GO TO 2                                     PSP01700
      RETURN                                                           PSP01710
C                                                                      PSP01720
       END                                                             PSP01730
```

FILE: RANK FORTRAN A

```
      SUBROUTINE RANK(NOFEAT,FETVC2,LNCAT,MEANS,IPTT)                    RAN00010
      IMPLICIT INTEGER(A-X)                                              RAN00020
      REAL MEANS(NOFEAT,LNCAT),SAVE,G(60)                                RAN00030
      DIMENSION FETVC2(28),IPTT(LNCAT)                                   RAN00040
C     INCLUDE COMRK4                                                     RAN00050
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          COM00010
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,    COM00020
     *    ,DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                COM00030
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    COM00040
     *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              COM00050
     *   CRDUNT,PRTUNT,RANDIO                                            COM00060
C$END                                                                    RAN00070
      IF(MOD(NOFEAT,4))10,20,10                                          RAN00080
 10   WRITE(6,500)                                                       RAN00090
      GO TO 99                                                           RAN00100
 20   DO 30 I=1,LNCAT                                                    RAN00110
      IPTT(I)=I                                                          RAN00120
 30   CONTINUE                                                           RAN00130
      DO 40 J=1,LNCAT                                                    RAN00140
      G(J)=0                                                             RAN00150
 40   CONTINUE                                                           RAN00160
      DO 50 I=1,NOFEAT,NCHPAS                                            RAN00170
      DO 50 J=1,LNCAT                                                    RAN00180
      G(J)=G(J)+(-0.29*MEANS(I,J)-0.56*MEANS(I+1,J)+0.6*MEANS(I+2,J)     RAN00190
     *-0.49*MEANS(I+3,J))                                                RAN00200
 50   CONTINUE                                                           RAN00210
      J=0                                                                RAN00220
 60   J=J+1                                                              RAN00230
      IF(J.GT.LNCAT)GO TO 90                                             RAN00240
      IF(J.EQ.LNCAT)GO TO 75                                             RAN00250
      IF(G(J).LT.G(J+1))GO TO 70                                         RAN00260
      GO TO 60                                                           RAN00270
 70   SAVE=G(J)                                                          RAN00280
      G(J)=G(J+1)                                                        RAN00290
      G(J+1)=SAVE                                                        RAN00300
      ISAVE=IPTT(J)                                                      RAN00310
      IPTT(J)=IPTT(J+1)                                                  RAN00320
      IPTT(J+1)=ISAVE                                                    RAN00330
      K=J                                                                RAN00340
 75   IF(K.EQ.1)GO TO 60                                                 RAN00350
 80   IF(G(K).LT.G(K-1))GO TO 60                                         RAN00360
      SAVE=G(K-1)                                                        RAN00370
      G(K-1)=G(K)                                                        RAN00380
      G(K)=SAVE                                                          RAN00390
      ISAVE=IPTT(K-1)                                                    RAN00400
      IPTT(K-1)=IPTT(K)                                                  RAN00410
      IPTT(K)=ISAVE                                                      RAN00420
      K=K-1                                                              RAN00430
      GO TO 80                                                           RAN00440
 90   CONTINUE                                                           RAN00450
      WRITE(6,510)(I,IPTT(I),G(I),I=1,LNCAT)                             RAN00460
 99   CONTINUE                                                           RAN00470
      RETURN                                                             RAN00480
 500  FORMAT(1X,'THE NUMBER OF CHANNELS ARE NOT A MULTIPLE OF 4.         RAN00490
     *THE COLOR KEYS WILL BE ORDERED BY CLUSTER NUMBER.')                RAN00500
 510  FORMAT(1X,'COLOR KEY ',I2,' IS CLUSTER ',I2,' GREENNESS=',F7.2/)   RAN00510
      END                                                                RAN00520
```

FILE: RDDATA FORTRAN A

```
C*     THIS SUBROUTINE COORDINATES THE ROUTINES TO READ FIELDS OF DATA    RDD00010
C*     FROM THE IMAGE TAPE AND STORE IT ON A DRUM FILE FOR                RDD00020
C*     THE ISOCLS ROUTINES.                                               RDD00030
C*                                                                        RDD00040
       SUBROUTINE RDDATA(ARRAY,TOP,IDATA,IDIM,LAST)                       RDD00050
       IMPLICIT INTEGER (A-Z)                                             RDD00060
       DIMENSION ARRAY(TOP),FLDINF(6),IDATA(IDIM),FL(12),LSTAT(3)         RDD00070
C      INCLUDE COMARG.LIST                                                RDD00080
C      INCLUDE COMPRS.LIST                                                RDD00090
       COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                  RDD00100
      *                    MAP,SPTRIG,    IRD,  KPTS, NOPTS, PUNCH,      RDD00110
      *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                       RDD00120
      *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,        RDD00130
      *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT          RDD00140
      * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                       RDD00150
      *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR       RDD00160
      *,IORDER,INUNIT,INFILE,INITM,PMIN,SURVEC(62),NOSUB2,CHNVC(30)       RDD00170
      * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                               RDD00180
      * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                 RDD00190
      *  DUFLAG,DODU,STDOTS(50),NSDOTS,SUNCOR(30),LLNCAT,                 RDD00200
      * DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)             RDD00210
      *  ,MXFET1,MAXPOP                                                   RDD00220
       REAL SUNCOR                                                        RDD00230
       COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          RDD00240
      *              HISFIL,HISKEY,TRFORM,EPIPTP,ERPKEY,MAPUNT,NOFILE,    RDD00250
      *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 RDD00260
      *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                     RDD00270
      *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,               RDD00280
      *  CRDUNT,PRTUNT,RANDIO                                             RDD00290
CSEND                                                                     RDD00300
       RRUNIT = 30                                                        RDD00310
       DIMENSION  CARD(20)                                                RDD00320
       EQUIVALENCE  (FLDINF(1),LINSTR),  (FLDINF(4),SAMSTR),              RDD00330
      *             (FLDINF(2),LINEND),  (FLDINF(5),SAMEND),             RDD00340
      *             (FLDINF(3),LININC),  (FLDINF(6),SAMINC)              RDD00350
       DATA LPRN/'('/                                                     RDD00360
       DIMENSION LND(2),LDOU(11,2),IDB(35),IDE(35),NDINT(11,2),           RDD00370
      *  DINT(120,2),DPINT(12,2),DIN(70)                                  RDD00380
       DATA DONAME/'OTHE'/                                                RDD00390
       DATA DUNAME/'UNID'/                                                RDD00400
       DIMENSION FLDSAV(4,10),VERTEX(220)                                 RDD00410
C                                                                         RDD00420
C*     RESERVE 2000 LOCATIONS OF 'ARRAY' FOR FIELD DEFINITION INFORMATION.RDD00430
C*     THE REMAINDER OF 'ARRAY' IS USED FOR I/O BUFFERS.                  RDD00440
C*                                                                        RDD00450
C*     CLASS AND FIELD INFORMATION STORED AS FOLLOWS                      RDD00460
C*                                                                        RDD00470
C*                                                                        RDD00480
C*        ARRAY(1)    =CLASS NAME                                         RDD00490
C*        ARRAY(2)    =RESERVED FOR INDEX POINTER TO NEXT CLASS NAME      RDD00500
C*        ARRAY(3)    =RESERVED FOR NO. OF CLUSTERS IN THIS CLASS         RDD00510
C*        ARRAY(4)    =NO. OF FIELDS FOR THIS CLASS                       RDD00520
C*        ARRAY(5)    =FIRST FIELD NAME FOR THIS CLASS                    RDD00530
C*             (6)    =NO. OF VERTICES FOR THIS FIELD  (NV)               RDD00540
C*             (7)-(7+NV*2)  = ACTUAL VERTEX NUMBERS                      RDD00550
C*             (8+NV*2)  =TOTAL PIXELS IN THIS FIELD                      RDD00560
C*             (9+NV*2)-(10+NV*2) = FLDINF BLOCK FOR THIS FIELD           RDD00570
       CALL TAPHDR(DATAPE,DATFIL)                                         RDD00580
     1 CONTINUE                                                           RDD00590
       RESERV=2000                                                        RDD00600
       ADDRES=BEGIN1                                                      RDD00610
       IBUF=1                                                             RDD00620
       IWRD=0                                                             RDD00630
       NVRT=0                                                             RDD00640
       LAST=0                                                             RDD00650
       TOTWRD=0                                                           RDD00660
       TOP = 0                                                            RDD00670
       DOFLAG = 0                                                         RDD00680
       DUFLAG = 0                                                         RDD00690
       DODU = 0                                                           RDD00700
       NDOU(1) = 0                                                        RDD00710
       NDOU(2) = 0                                                        RDD00720
     2 RFINDX=RESERV+1                                                    RDD00730
       NBUFS=3                                                            RDD00740
       MAXDIM=TOP-RESERV                                                  RDD00750
       BUFSIZ= MAXDIM/(NBUFS*NOFEAT)  * NOFEAT                            RDD00760
       IF(BUFSIZ.GT. 100)GO TO 3                                          RDD00770
       RESERV=RESERV-100                                                  RDD00780
       IF(RESERV.GT.30 )GO TO 2                                           RDD00790
```


ORIGINAL PAGE IS
OF POOR QUALITY

FILE: RDDATA FORTRAN A

```
      GO TO 70                                                          RDD000800
    3 CONTINUE                                                          RDD000810
      NOFLD=0                                                           RDD000820
      IPT=1                                                             RDD000830
      TOTVT2=0                                                          RDD000840
      IF(NOCL.EQ.0)GO TO 5                                              RDD000850
    4 ARRAY(IPT)=NXTCLS                                                 RDD000860
      IPT=IPT+4                                                         RDD000870
      WRITE(6,HEAD)                                                     RDD000880
      WRITE(6,500)NXTCLS                                                RDD000890
C*                                                                      RDD000900
C*    READ A FIELD DESCRIPTION FROM CARDS.                              RDD000910
C*                                                                      RDD000920
    5 ICK = LAREAD(ARRAY(IPT),ARRAY(IPT+2),FLDINF,ARRAY(IPT+1) ).       RDD000930
      IF(ICK.NE.-3) GO TO 1000                                          RDD000940
       WRITE (6,140)                                                    RDD000950
       READ (RRUNIT,150) (CARD(I), I=1,20)                              RDD000960
       WRITE (6,150)  (CARD(I), I=1,20)                                 RDD000970
  150  FORMAT(20A4)                                                     RDD000980
  140  FORMAT(1X,20A4)                                                  RDD000990
       REWIND RRUNIT                                                    RDD001000
      IDP=IDP+1                                                         RDD001010
      IDCNT(IDP)=0                                                      RDD001020
      DVPNT(1,IDP)=1                                                    RDD001030
      READ(30,100) DNAME                                                RDD001040
      REWIND 30                                                         RDD001050
      IF(DNAME.EQ.CONAME) ITRIGO=1                                      RDD001060
      IF(DNAME.EQ.DUNAME) ITRIGU=1                                      RDD001070
      IF(DNAME.EQ.DONAME) IS=2                                          RDD001080
      IF(DNAME.EQ.DUNAME) IS=1                                          RDD001090
      INDV=1                                                            RDD001100
      INDR=1                                                            RDD001110
      GO TO 5                                                           RDD001120
 1000 IF(ICK.LE.0.OR.IDP.LE.0) GO TO 1030                               RDD001130
      IF (IDCNT(IDP).LT.10) GO TO 1025                                  RDD001140
      WRITE(6,170)                                                      RDD001150
  170 FORMAT(// ' TOO MANY DO OR DU FIELDS THESE IGNORED')              RDD001160
      GO TO 5                                                           RDD001170
 1025 CONTINUE                                                          RDD001180
       READ (RRUNIT,150) (CARD(I), I=1,20)                              RDD001190
       WRITE (6,150)  (CARD(I), I=1,20)                                 RDD001200
       REWIND RRUNIT                                                    RDD001210
       DVERT(INDV,IDP) = ARRAY(IPT + 1)                                 RDD001220
      IDLIM = DVERT(INDV,IDP)*2                                         RDD001230
      DO 1010 I=1,IDLIM                                                 RDD001240
      INDV=INDV+1                                                       RDD001250
      VERTEX(TOTVT2+I)=ARRAY(IPT+I+1)                                   RDD001260
 1010 DVERT(INDV,IDP) = ARRAY(IPT + I + 1)                              RDD001270
      INDV = INDV + 1                                                   RDD001280
      TOTVT2=TOTVT2+IDLIM                                               RDD001290
      DO 1020 I=1,6                                                     RDD001300
      DAFCT(INDR,IDP)=FLDINF(I)                                         RDD001310
 1020 INDR=INDR+1                                                       RDD001320
      IDCNT(IDP)=IDCNT(IDP)+1                                           RDD001330
      IDUM=IDCNT(IDP)+1                                                 RDD001340
      DVPNT(IDUM,IDP)=INDV                                              RDD001350
      GO TO 5                                                           RDD001360
C     FINISHED WITH DO/DU FIELD PROCESSING                              RDD001370
 1030   CONTINUE                                                        RDD001380
      IDP=0                                                             RDD001390
      IDPP=ITRIGO+ITRIGU                                                RDD001400
      IF(ICK.LT.0)GO TO 20                                              RDD001410
      IF(ICK.EQ.0)GO TO 30                                              RDD001420
      IF(NOCL.GT.0)GO TO 6                                              RDD001430
      WRITE(6,200)                                                      RDD001440
      CALL CVERR                                                        RDD001450
    6 CONTINUE                                                          RDD001460
      NV=ARRAY(IPT+1)                                                   RDD001470
      NVRT=NVRT+NV                                                      RDD001480
      NOFLD=NOFLD+1                                                     RDD001490
      NSAMP=(SAMEND-SAMSTR)/SAMINC+1                                    RDD001500
      FLDSAM=0                                                          RDD001510
      IB=IPT+2                                                          RDD001520
      NO=NV-1                                                           RDD001530
      NR=NO-5                                                           RDD001540
      IF(NO.GT.5)NO=5                                                   RDD001550
      IE=IB+NO*2 - 1                                                    RDD001560
      WRITE(6,600)NOFLD,ARRAY(IPT),SAMINC,LININC,                       RDD001570
     *      (LPRN,ARRAY(I),ARRAY(I+1),I=IB,IE,2)                        RDD001580
```

FILE: RDDATA FORTRAN A

```
      IF(NP.LE.0)GO TO 7                                                        RDD01590
      IB=IE+1                                                                   RDD01600
      IE=IB+NP*2 - 1                                                            RDD01610
      WRITE(A,650) (LPRN,ARRAY(I),ARRAY(I+1),I=IB,IE,2)                         RDD01620
    7 CONTINUE                                                                  RDD01630
      IF(NSAMP*NOFEAT.GT.IDIM)GO TO 90                                          RDD01640
C*                                                                              RDD01650
C*    POSITION TAPE FOR THIS FIELD                                              RDD01660
C*                                                                              RDD01670
      CALL FLDINT(FLDINF,FETVEC,NOFEAT)                                         RDD01680
      FLDSAM=0                                                                  RDD01690
      DO 10 LINE=LINSTR,LINEND,LININC                                           RDD01700
      LND(1)=0                                                                  RDD01710
      LND(2)=0                                                                  RDD01720
      IDAB = 2                                                                  RDD01730
      IDEE=1                                                                    RDD01740
      IF (IDPP.EQ.0) GO TO 1095                                                 RDD01750
C     BOTH DO AND DU TRIGGERS OFF --- SKIP AROUND                               RDD01760
      DO 1060 IND=1,IDPP                                                        RDD01770
      IDLIM=IDCNT(IND)                                                          RDD01780
      DO 1050 I=1,IDLIM                                                         RDD01790
      IDUM=(I-1)*4                                                              RDD01800
      LDSTR=DIRECT(IDUM+1,IND)                                                  RDD01810
      LDEND=DIRECT(IDUM+2,IND)                                                  RDD01820
      LDINC=DIRECT(IDUM+3,IND)                                                  RDD01830
      DO 1040 II = LDSTR,LDEND,LDINC                                            RDD01840
      IF(II.NE.LINE) GO TO 1040                                                 RDD01850
      LND(IND)=LND(IND)+1                                                       RDD01860
      IDUM=LND(IND)                                                             RDD01870
      LDOU(IDUM,IND) = I                                                        RDD01880
 1040 CONTINUE                                                                  RDD01890
 1050 CONTINUE                                                                  RDD01900
 1060 CONTINUE                                                                  RDD01910
      IF (LND(1).EQ.0.AND.LND(IDPP).EQ.0) GO TO 1095                            RDD01920
C    NO DO  OR   DU   FOR THIS LINE                                             RDD01930
      IF(LND(1).GT.0)IDAB=1                                                     RDD01940
      IF(IDPP.EQ.2.AND.LND(2).GT.0) IDEE=2                                      RDD01950
      DO 1090 IND=IDAB,IDEE                                                     RDD01960
      IDLIM=LND(IND)                                                            RDD01970
      IF (IDLIM.EQ.0) GO TO 1090                                                RDD01980
      IDUM=0                                                                    RDD01990
      DPINT(1,1) = 1                                                            RDD02000
      DPINT(1,2) = 1                                                            RDD02010
      DO 1080 I=1,IDLIM                                                         RDD02020
      IDF=LDOU(I,IND)                                                           RDD02030
      DVP=DVPNT(IDF,IND)                                                        RDD02040
      CALL FDLINT(DVERT(DVP+1,IND),DVERT(DVP,IND),FL,LINE,SAMPS,NI)             RDD02050
      NDINT(I,IND)=NI                                                           RDD02060
      IF (NI.EQ.0) GO TO 1080                                                   RDD02070
      DO 1070 II=1,NI                                                           RDD02080
 1070 DPINT(II+IDUM,IND)=FL(II)                                                 RDD02090
      IDUM=IDUM+NI                                                              RDD02100
      DPINT(I+1,IND) = IDUM + 1                                                 RDD02110
 1080 CONTINUE                                                                  RDD02120
 1090 CONTINUE                                                                  RDD02130
 1095 CONTINUE                                                                  RDD02140
      CALL LINERD(IDATA,ENDTAP)                                                 RDD02150
      IF(ENDTAP.EQ.-1)GO TO 80                                                  RDD02160
C*                                                                              RDD02170
C*    FIND SAMPLE INTERSECTS FOR THIS LINE - NI=NO. OF INTERSECTS               RDD02180
C*                                                                              RDD02190
      CALL FDLINT(ARRAY(IPT +2),NV,FL,LINE,SAMPS,NI)                            RDD02200
C*                                                                              RDD02210
C*    STORE DATA ON THIS LINE INTO OUTPUT BUFFER                                RDD02220
C*                                                                              RDD02230
      REAL RWRD                                                                 RDD02240
      EQUIVALENCE(RWRD,ITGWRD)                                                  RDD02250
      MODSS=MOD(SAMSTR,SAMINC)                                                  RDD02260
      DO 40 I=1,NI,2                                                            RDD02270
      IB=(FL(I)-SAMSTR)/SAMINC+1                                                RDD02280
      IE=(FL(I+1)-SAMSTR)/SAMINC+1                                              RDD02290
      IF(MODSS.NE.MOD(FL(I),SAMINC)) IB=IB+1                                    RDD02300
      IF(IB.GT.IE) GOTO 40                                                      RDD02310
      IF(IDPP.EQ.0) GOTO 2055                                                   RDD02320
      IF(LND(IDAB).EQ.0.AND.LND(IDEE).EQ.0) GOTO 2055                           RDD02330
      DO 2050 IND=IDAB,IDEE                                                     RDD02340
      IDLIM=LND(IND)                                                            RDD02350
      IF(IDLIM.EQ.0) GOTO 2050                                                  RDD02360
      IDSIT=1                                                                   RDD02370
```

FILE: RDDATA FORTRAN A

```
      MEANDD=MEANDO                                                          RDD02380
      IF(IDBB.EQ.IDEE) GOTO 2003                                             RDD02390
      IF(IND.EQ.2) IDSIT=2                                                   RDD02400
      IF(IND.EQ.2) MEANDD=MEANDU                                             RDD02410
      GOTO 2009                                                              RDD02420
 2003 IF(IDBB.EQ.1.AND.IDPP.EQ.2) GOTO 2009                                  RDD02430
      IF(ITRIGU.EQ.0) GOTO 2009                                              RDD02440
      IDSIT=2                                                                RDD02450
      MEANDD=MEANDU                                                          RDD02460
 2009 CONTINUE                                                               RDD02470
      DO 2040 K=1,IDLIM                                                      RDD02480
      NDIN=NDINT(K,IND)                                                      RDD02490
      IF(NDIN.EQ.0) GOTO 2040                                                RDD02500
      DPIN=DPINT(K,IND)                                                      RDD02510
      DO 2010 KK=1,NDIN                                                      RDD02520
 2010 DIN(KK)=DINT(DPIN+KK-1,IND)                                            RDD02530
      IDUM=0                                                                 RDD02540
      DO 2020 KK=1,NDIN,2                                                    RDD02550
      IDUM=IDUM+1                                                            RDD02560
      IDB(IDUM)=(DIN(KK)-SAMSTR)/SAMINC+1                                    RDD02570
      IDE(IDUM)=(DIN(KK+1)-SAMSTR)/SAMINC+1                                  RDD02580
      IF(MODSS.NE.MOD(DIN(KK),SAMINC)) IDB(IDUM)=IDB(IDUM)+1                 RDD02590
 2020 CONTINUE                                                               RDD02600
      DO 2030 KK=1,IDUM                                                      RDD02610
      IDS=IDB(KK)                                                            RDD02620
      IDF=IDE(KK)                                                            RDD02630
      IF(IDS.GT.IE.OR.IB.GT.IDF) GOTO 2030                                   RDD02640
      IF(IDS.LE.IB) IDS=IB                                                   RDD02650
      IF(IDF.GT.IE) IDF=IE                                                   RDD02660
      IF(IDS.GT.IDF) GOTO 2030                                               RDD02670
      DO 2025 KKK=IDS,IDF                                                    RDD02680
      NDOU(IND)=NDOU(IND)+1                                                  RDD02690
      DO 2023 KKKK=1,NOFEAT                                                  RDD02700
      DUMMY1=KKK+NSAMP*(KKKK-1)                                              RDD02710
 2023 IDATA(DUMMY1)=MEANDD                                                   RDD02720
 2025 CONTINUE                                                               RDD02730
 2030 CONTINUE                                                               RDD02740
 2040 CONTINUE                                                               RDD02750
      IF(IDSIT.EQ.1.AND.NDOU(IND).GT.0) DOFLAG=1                             RDD02760
      IF(IDSIT.EQ.2.AND.NDOU(IND).GT.0) DUFLAG=1                             RDD02770
 2050 CONTINUE                                                               RDD02780
 2055 NDOU=DOFLAG+DUFLAG                                                     RDD02790
      DO 50 J=IB,IE                                                          RDD02800
      FLDSAM=FLDSAM+1                                                        RDD02810
      DO 50 K=1,NOFEAT                                                       RDD02820
      IWRD=IWRD+1                                                            RDD02830
      DUMMY2=J+NSAMP*(K-1)                                                   RDD02840
      RWRD=IDATA(DUMMY2)                                                     RDD02850
      DUMMY3=HFINDX-1+IWRD+BUFSIZ*(IBUF-1)                                   RDD02860
      ARRAY(DUMMY3)=ITGWRD                                                   RDD02870
      IF(IWRD.LT.BUFSIZ) GOTO 50                                             RDD02880
      TOTWRD=TOTWRD+IWRD                                                     RDD02890
      IF(TOTWRD.GT.NWDS) GOTO 35                                             RDD02900
      DUMMY4=HFINDX+BUFSIZ*(IBUF-1)                                          RDD02910
      CALL RWWRITE(ADDRES,ARRAY(DUMMY4),BUFSIZ,LSTAT(IBUF))                  RDD02920
      IBUF=IBUF+1                                                            RDD02930
      ADDRES=ADDRES+BUFSIZ                                                   RDD02940
      IF(IBUF.GT.NBUFS) IBUF=1                                               RDD02950
   40 IF(LSTAT(IBUF).EQ.1) GOTO 40                                           RDD02960
      IWRD=0                                                                 RDD02970
   50 CONTINUE                                                               RDD02980
   60 CONTINUE                                                               RDD02990
   10 CONTINUE                                                               RDD03000
      IPT = IPT + NV*2 + 2                                                   RDD03010
      ARRAY(IPT)=FLDSAM                                                      RDD03020
      DO 15 I=1,6                                                            RDD03030
      IPT=IPT+1                                                              RDD03040
   15 ARRAY(IPT)=FLDINF(I)                                                   RDD03050
      IPT=IPT+1                                                              RDD03060
      IF(IPT+30 .GT. RESERV)GO TO 70                                         RDD03070
      GO TO 5                                                                RDD03080
C*                                                                           RDD03090
C*    CLASS NAME CARD ENCOUNTERED - REREAD PREVIOUS CARD TO GET NAME         RDD03100
C*                                                                           RDD03110
   20 NOCL=NOCL+1                                                            RDD03120
      IF(NOCL.GT.1)GO TO 25                                                  RDD03130
      READ(30,100)NXTCLS                                                     RDD03140
      REWIND 30                                                              RDD03150
      GO TO 4                                                                RDD03160
```

FILE: RDDATA FORTRAN A

```
   25 CLSNAM=NXTCLS                                                      RDD03170
      READ(30,100)NXTCLS                                                 RDD03180
      REWIND 30                                                          RDD03190
      GO TO 31                                                           RDD03200
C*    EMPTY LAST BUFFER AND RETURN TO PROCESS DATA FOR THIS CLASS.       RDD03210
   30 CLSNAM=NXTCLS                                                      RDD03220
      LAST=1                                                             RDD03230
C*                                                                       RDD03240
   31 TOTWRD=TOTWRD+IWRD                                                 RDD03250
      IF(TOTWRD.GT.NWDS) GOTO 35                                         RDD03260
      DUMMY=RFINDX+BUFSIZ*(IBUF-1)                                       RDD03270
      CALL RWRITE(ADDRES,ARRAY(DUMMY),IWRD,LSTAT(IBUF))                  RDD03280
      TOTPTS=TOTWRD/NOFEAT                                               RDD03290
      IF(TOTWRD+TOTPTS.LE.NWDS)RETURN                                    RDD03300
      WRITE(6,200)NWDS                                                   RDD03310
      RETURN                                                             RDD03320
   35 WRITE(6,200)NWDS                                                   RDD03330
      CALL CMERR                                                         RDD03340
   70 WRITE(6,300)RESERV                                                 RDD03350
      CALL CMERR                                                         RDD03360
   80 WRITE(6,400)                                                       RDD03370
      CALL CMERR                                                         RDD03380
   90 WRITE(6,700)IDIM                                                   RDD03390
      CALL CMERR                                                         RDD03400
  100 FORMAT(10X,A4)                                                     RDD03410
  140  FORMAT(/// ' DESIGNATED OTHER OR UNIDENTIFIABLE FIELDS INPUT')    RDD03420
  200 FORMAT(' TOO MUCH DATA REQUESTED--PIXELS*(CHANNELS+1) CANNOT EXCEERDD03430
     *D',I10)                                                            RDD03440
  300 FORMAT(' STORAGE REQUIRED FOR FIELD DEFINITION INFORMATION EXCEEDSRDD03450
     * THE DIMENSION LIMIT OF',I5)                                       RDD03460
  400 FORMAT(' END-OF-TAPE REACHED BEFORE END OF FIELD')                 RDD03470
  500 FORMAT(//40X,'FIELDS TO BE CLUSTERED FOR CLASS',1X,A4//            RDD03480
     *   T36,'SAMPLE',T45,'LINE'/T20,'FIELD NAME',T36,'INC.',           RDD03490
     *   T45,'INC.',T73,'VERTICES  (SAMPLE,LINE)'/)                     RDD03500
  600 FORMAT(1X,T16,I3,T22,A4,T36,I4,T45,I4,T60,                         RDD03510
     * 5(A1,I4,',',I4,')',1X))                                          RDD03520
  650 FORMAT(1X,T60,5(A1,I4,',',I4,')',1X))                              RDD03530
  700 FORMAT(' NO  OF PIXELS TO BE UNPACKED PER SCAN EXCEEDS THE DIMENSIRDD03540
     *ON LIMIT OF',I5)                                                   RDD03550
  800 FORMAT(// 'INPUT ERROR - A CLASSNAME CARD MUST BE INPUT BEFORE A GRDD03560
     *ROUP OF FIELDS'/)                                                  RDD03570
      RETURN                                                             RDD03580
      END                                                                RDD03590
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: RDMEAN FORTRAN A

```
      SUBROUTINE RDMEAN(MENS)                                          RDM00010
      IMPLICIT INTEGER(A-X)                                            RDM00020
C*                                                                     RDM00030
C*    THIS SUBROUTINE READS THE 'MEAN' CARD DECK OR FILE FOR ISOCLS.   RDM00040
C*                                                                     RDM00050
C     INCLUDE COMRK4.LIST                                              RDM00060
C     INCLUDE COMRK5.LIST                                              RDM00070
C     INCLUDE COMRK6.LIST                                              RDM00080
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                   COM00010
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                COM00020
     2           (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               COM00030
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                COM00010
     *                    MAP,SPTRIG,   IPD,  KPTS, NOPTS, PUNCH,      COM00020
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                     COM00030
     *             BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,       COM00040
     *             NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT         COM00050
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                     COM00060
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR     COM00070
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)     COM00080
     * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                             COM00090
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                               COM00100
     *  DUFLAG,DUDU,STOOTS(60),NSDOTS,SUNCOR(30),LLNCAT,               COM00110
     * DVERT(250,2),DRECT(60,2),DVRNT(11,2),IDCNT(2),NDOU(2)           COM00120
     *  ,MXFFT1,MAXPOP                                                 COM00130
      REAL SUNCOR                                                      COM00140
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,        COM00010
     *              HISFIL,HISKEY,TRFORM,ERPTP,ERPKEY,MAPUNT,NOFILE,   COM00020
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               COM00030
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                   COM00040
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,             COM00050
     *  CROUNT,PRTUNT,RANDIO                                           COM00060
CSEND                                                                  RDM00100
      COMMON/PASSA/NOFET1,FTVEC1(30)                                   RDM00110
      DIMENSION MENS(30,1)                                             RDM00120
      READ(CROUNT,500) LNCAT,NOFET1,(FTVEC1(I), I=1,NOFET1)            RDM00130
      DO 10 I=1,LNCAT                                                  RDM00140
 10   READ(CROUNT,510) (MENS(J,I), J=1,NOFET1)                         RDM00150
 500  FORMAT(5X,I5,15X,I5/5X,3012)                                     RDM00160
 510  FORMAT(5X,5F15.8)                                                RDM00170
      ADDRES=IBEGIN                                                    RDM00180
      CALL RWRITE(ADDRES,LNCAT,1,LSTAT)                                RDM00190
      ADDRES=ADDRES+1                                                  RDM00200
      CALL RWRITE(ADDRES,NOFET1,1,LSTAT)                               RDM00210
      ADDRES=ADDRES+1                                                  RDM00220
      CALL RWRITE(ADDRES,FTVEC1,NOFET1,LSTAT)                          RDM00230
      ADDRES=ADDRES+NOFET1                                             RDM00240
      KW=MXFFT1*LNCAT                                                  RDM00250
      CALL RWRITE(ADDRES,MENS,KW,JSTAT)                                RDM00260
   11 IF(JSTAT.EQ.1)GOTO 11                                            RDM00270
      RETURN                                                           RDM00280
C                                                                      RDM00290
C                                                                      RDM00300
      ENTRY RDFILE(MEANS,MENS)                                         RDM00310
      DIMENSION  MEANS(NOFEAT,MAXCLS)                                  RDM00320
      ADDRES=IBEGIN                                                    RDM00330
      CALL RREAD(ADDRES,LNCAT,1,LSTAT)                                 RDM00340
      ADDRES=ADDRES+1                                                  RDM00350
      CALL RREAD(ADDRES,NOFET1,1,LSTAT)                                RDM00360
      ADDRES=ADDRES+1                                                  RDM00370
      CALL RREAD(ADDRES,FTVEC1,NOFET1,LSTAT)                           RDM00380
      ADDRES=ADDRES+NOFET1                                             RDM00390
      KW=MXFFT1*LNCAT                                                  RDM00400
      CALL RREAD(ADDRES,MENS,KW,JSTAT)                                 RDM00410
   12 IF(JSTAT.EQ.1)GOTO 12                                            RDM00420
C*                                                                     RDM00430
C*    STORE ONLY CHANNELS REQUESTED IN FETVEC                          RDM00440
C*                                                                     RDM00450
      IF(LNCAT.GT.MAXCLS)LNCAT=MAXCLS                                  RDM00460
      DO 40 J=1,NOFEAT                                                 RDM00470
      DO 30 K=1,NOFET1                                                 RDM00480
      IF(FETVEC(J).NE.FTVEC1(K))GO TO 30                               RDM00490
      DO 20 I=1,LNCAT                                                  RDM00500
   20 MEANS(J,I) = MENS(K,I)                                           RDM00510
      GO TO 40                                                         RDM00520
   30 CONTINUE                                                         RDM00530
      WRITE(6,100)FETVEC(J)                                            RDM00540
      DO 35 I=1,LNCAT                                                  RDM00550
   35 MEANS(J,I) = 50. + I*10.                                         RDM00560
   40 CONTINUE                                                         RDM00570
```

FILE: RDMEAN FORTRAN A

```
C*                                                                      RDM00580
C*     PRINT INITIAL CLUSTER CENTERS                                    RDM00590
C*                                                                      RDM00600
      WRITE(6,200)                                                      RDM00610
      IB=1                                                              RDM00620
      IE=12                                                             RDM00630
   45 IF(NOFEAT.LT.IE)IE=NOFEAT                                         RDM00640
      WRITE(6,300)(FETVEC(J),J=IB,IE)                                   RDM00650
      DO 50 I=1,LNCAT                                                   RDM00660
   50 WRITE(6,400)I, (MEANS(J,I),J=IB,IE)                               RDM00670
      IF(IE.EQ.NOFEAT)GO TO 60                                          RDM00680
      IB=IB+12                                                          RDM00690
      IE=IE+12                                                          RDM00700
      GO TO 45                                                          RDM00710
   60 RETURN                                                            RDM00720
  100 FORMAT(' MEANS FOR CHANNEL',I4,' ARE NOT ON FILE--DUMMY VALVES WILRDM00730
     *L BE USED')                                                       RDM00740
  200 FORMAT(///15X,'INITIAL CLUSTER MEANS'/)                           RDM00750
  300 FORMAT(2X,'CLUSTER',2X,12(1X,'CH(',I2,')',1X))                    RDM00760
  400 FORMAT(5X,I2,5X,12(F6.2,2X))                                      RDM00770
      END                                                               RDM00780
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SETUP7 FORTRAN A

```
      SUBROUTINE SETUP7(ARRAY,TOP,ITIME)                                    SET00010
      IMPLICIT INTEGER (A-X)                                                SET00020
C*****************************************************************************SET00030
C                                                                           SET00040
C*    SETUP7 READS AND ANALYZES ALL CONTROL CARD INPUT FOR THE              SET00050
C*    ISOCLS PROCESSOR                                                      SET00060
C                                                                           SET00070
C     INCLUDE COMBK4.LIST                                                   SET00080
C     INCLUDE COMBK5.LIST                                                   SET00090
C     INCLUDE COMBK6.LIST                                                   SET00100
C     INCLUDE CMBK16.LIST                                                   SET00110
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                        SET00120
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                     SET00130
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                  SET00140
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                     SET00150
     *                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,          SET00160
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                          SET00170
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,           SET00180
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT             SET00190
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                          SET00200
     *,VARS(7,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR         SET00210
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)         SET00220
     * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                                  SET00230
     * SYMDO,SYMDU,ITRIGO,ITRIGU,OOFLAG,                                    SET00240
     *  OUFLAG,OODU,STOOTS(60),NSOOTS,SUNCOR(30),LLNCAT,                    SET00250
     * OVERT(250,2),OVECT(60,2),OVPNT(11,2),IOCNT(2),NOOU(2)                SET00260
     *  ,MXFET),MAXPOP                                                      SET00270
      REAL SUNCOR                                                           SET00280
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,             SET00290
     *              HISFIL,HISKEY,TRFORM,ERIPTP,EPPKEY,MAPUNT,NOFILE,       SET00300
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    SET00310
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                        SET00320
     *  ,DOTUNT,DOTFIL,NCHPAS,TPNSFL,BMTRFL,HISTFL,PCHUNT,                  SET00330
     *  CRDUNT,PRTUNT,RANDIO                                                SET00340
      COMMON/ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP          SET00350
CSEND                                                                       SET00360
C*****************************************************************************SET00370
C                                                                           SET00380
       REAL DLMIN,CHNTHS,SEP,STDMAX,PERCEN                                  SET00390
      DIMENSION ARRAY(1),CHAR(2)                                            SET00400
      DIMENSION COMVEC(2),EQUVEC(2)                                         SET00410
      DATA COMVEC/1,','/,EQUVEC/1,'='/                                      SET00420
      EQUIVALENCE(STOP,ISTOP)                                               SET00430
      DIMENSION INVEC(30),CARD(62),ACARD(20),SLASH(2)                       SET00440
       DATA SLASH/1,'/'/                                                    SET00450
      DATA INVEC /'CHAN', 'ISTO', 'NMIN', 'KRN ', 'STDM', 'DLMI',           SET00460
     *            'SEP ', 'HED1', 'HED2', 'DATE', '*END', 'COMM',           SET00470
     *      'SYMB', 'MEAN', 'MAP ', 'OPTI', 'CLAS', 'CHAI',                 SET00480
     *            'CLUS', 'FORM', 'DATA', 'STAT', 'SEQU', 'PERC',           SET00490
     *            'MODU', 'SUBC', 'PMIN', 'OOTF', 'SUNA', 'OOTS'/           SET00500
      DIMENSION SMBLS(60)                                                   SET00510
      DATA IRCD/'I'/,UBCD/'U'/,LBCD/'L'/,MBCD/'M'/                          SET00520
      DATA CHAR/1,'='/,  EQUAL/'='/                                         SET00530
      DATA PRCD/'P'/,ERCD/'E'/,DBCD/'D'/                                    SET00540
      DATA FRCD/'F'/,CHCD/'C'/                                              SET00550
      DATA TRCD/'T'/,SRCD/'S'/,OBCD/'O'/,BLANK/' '/,COMMA/','/              SET00560
      DATA SMBLS /'1','2','3','4','5','6','7','8','9','A','B','C','D',     SET00570
     1            'E','F','G','H','I','J','K','L','M','N','O','P','Q',     SET00580
     2            'R','S','T','U','V','W','X','Y','Z','+','|','-','/',     SET00590
     3            '-','*','.','$',' ','=',0,' ',')','(',' ',':',           SET00600
     4            '&','>','<','?',1H',',','.','.',' ' /                    SET00610
      IF(ITIME.NE.1)GO TO 5                                                 SET00620
C*                                                                          SET00630
C*    SET UP DEFAULT VALUES FOR INPUT PARAMETERS                            SET00640
C*                                                                          SET00650
       ERCOMP=0                                                             SET00660
       IORDER=0                                                             SET00670
      NOSEQ = 2                                                             SET00680
      NOCHAN = 0                                                            SET00690
       SEP=1                                                                SET00700
       NOSUB2=0                                                             SET00710
       PMIN=1                                                               SET00720
       INITM=0                                                              SET00730
      MAPKEY=1                                                              SET00740
      MAPFMT = 0                                                            SET00750
      ICNT=0                                                                SET00760
      PUNCH=0                                                               SET00770
      DLMIN=3.2                                                             SET00780
      STDMAX=4.5                                                            SET00790
```

FILE: SETUP7 FORTRAN A

```
      MAP=20                                                            SET00800
      STATKY=0                                                          SET00810
      ISTOP=10                                                          SET00820
      NMIN=30                                                           SET00830
      KRN=20                                                            SET00840
      NCLASS=1                                                          SET00850
      MAXCLS = 60                                                       SET00860
      ICHN=0                                                            SET00870
      PERCEN = .2                                                       SET00880
      IPCT = 80                                                         SET00890
      ISUNC=0                                                           SET00900
      ISUNT=0                                                           SET00910
      NSDOTS=0                                                          SET00920
      DO 1 I=1,MAXPOP                                                   SET00930
    1 SYMMTX(I) = SMALS(I)                                              SET00940
      NOFEAT = 0                                                        SET00950
      ISTART =0                                                         SET00960
      DO  2  I=1,19,2                                                   SET00970
    2 SEQUEN(I)=SRCD                                                    SET00980
      DO  3  I=2,18,2                                                   SET00990
    3 SEQUEN(I)=CRCD                                                    SET01000
      SEQUEN(20)=0                                                      SET01010
    5 CONTINUE                                                          SET01020
      IF(ITIME.NE.1) WRITE(6,HEAD)                                      SET01030
      WRITE(6,630)                                                      SET01040
C                                                                       SET01050
CC      PUT THE NEXT CARD IN THE REREAD BUFFER                          SET01060
C                                                                       SET01070
      RRUNIT=30                                                         SET01080
   10 READ(21,1000)(ACARD(I),I=1,20)                                    SET01090
 1000 FORMAT(20A4)                                                      SET01100
      WRITE(RRUNIT,1000)(ACARD(I),I=1,20)                               SET01110
      REWIND RRUNIT                                                     SET01120
C                                                                       SET01130
CC                                                                      SET01140
CC    READ IN CARD                                                      SET01150
C                                                                       SET01160
      READ(30,480) CODE,CARD                                            SET01170
      REWIND RRUNIT                                                     SET01180
      WRITE(6,550) CODE,CARD                                            SET01190
      COL = 0                                                           SET01200
      DO 20 I=1,30                                                      SET01210
      IF(CODE.EQ.INVEC(I)) GO TO                                        SET01220
     *(30,50,70,80,90,100,110,130,140,150,280,160,                      SET01230
     *  170,250,210,240,200,235,230,220,260,270,246,249,                SET01240
     *  256,283,284,300,340,380),I                                      SET01250
   20 CONTINUE                                                          SET01260
      WRITE(6,490) CODE,CARD                                            SET01270
      GO TO 10                                                          SET01280
C           CHANNEL CARD                                                SET01290
   30 J=NXTCHR(CARD,COL)                                                SET01300
      IF(J.EQ.BLANK) GO TO 10                                           SET01310
      IF(ITIME.EQ.1) GO TO 35                                           SET01320
      WRITE(6,640)                                                      SET01330
      GO TO 10                                                          SET01340
   35 CONTINUE                                                          SET01350
      IF(J.EQ.SRCD) GO TO 37                                            SET01360
      IF(J.EQ.DRCD) GO TO 43                                            SET01370
   36 WRITE(6,645)                                                      SET01380
  645 FORMAT(' ERROR ON CHANNEL CARD')                                  SET01390
      GO TO 10                                                          SET01400
   37 M=FIND12(CARD,COL,EQUVEC)                                         SET01410
      IF(M.EQ.-1) GO TO 36                                              SET01420
      NOCHAN=NUMBER(CARD,COL,CHNVC,NOCHAN)                              SET01430
      COL = COL-1                                                       SET01440
      CALL ORDER(CHNVC,NOCHAN)                                          SET01450
      GO TO 30                                                          SET01460
   43 M=FIND12(CARD,COL,EQUVEC)                                         SET01470
      IF (M .EQ. -1) GO TO 36                                           SET01480
      NOFEAT = NUMBER(CARD,COL,FETVEC,NOFEAT)                           SET01490
      COL = COL - 1                                                     SET01500
      CALL ORDER(FETVEC,NOFEAT)                                         SET01510
      GO TO 30                                                          SET01520
C*                                                                      SET01530
C*    ISTOP CARD (MAXIMUM NUMBER OF ITERATIONS)                         SET01540
C*                                                                      SET01550
   50 J = NXTCHR(CARD,COL)                                              SET01560
      IF (J.EQ.BLANK) GO TO 10                                          SET01570
      COL=COL-1                                                         SET01580
```

FILE: SETUP7 FORTRAN A

```
      J       = NUMBER(CARD,COL,ISTOP,ISTART)                                  SET01590
      GO TO 10                                                                 SET01600
C*                                                                             SET01610
C*    NMIN CARD (MINIMUM NUMBER OF POINTS PER CLUSTER)                         SET01620
C*                                                                             SET01630
   70 J = NXTCHR(CARD,COL)                                                     SET01640
      IF (J.EQ.BLANK) GO TO 10                                                 SET01650
      COL = COL-1                                                              SET01660
      J       = NUMBER(CARD,COL,NMIN ,ISTART)                                  SET01670
      GO TO 10                                                                 SET01680
C*                                                                             SET01690
C*    KRN CARD (NUMBER OF ITERATIONS PER FULL OUTPUT)                          SET01700
C*                                                                             SET01710
   80 J = NXTCHR(CARD,COL)                                                     SET01720
      IF (J.EQ.BLANK) GO TO 10                                                 SET01730
      COL = COL-1                                                              SET01740
      J       = NUMBER(CARD,COL,KRN  ,ISTART)                                  SET01750
      GO TO 10                                                                 SET01760
C*                                                                             SET01770
C*    STDMAX CARD (MAXIMUM STANDARD DEVIATION PER CLUSTER)                     SET01780
C*                                                                             SET01790
   90 J = FLTNUM(CARD,COL,STDMAX,1)                                            SET01800
      GO TO 10                                                                 SET01810
C*                                                                             SET01820
C*    DLMIN CARD (MINIMUM DISTANCE BETWEEN CLUSTER MEANS)                      SET01830
C*                                                                             SET01840
  100 J = FLTNUM(CARD,COL,DLMIN,1)                                             SET01850
      GO TO 10                                                                 SET01860
C*                                                                             SET01870
C*    SEP CARD  (DISTANCE FOR SPLITTING)                                       SET01880
C*                                                                             SET01890
  110 J = FLTNUM(CARD,COL,SEP,1)                                               SET01900
      SPTRIG=1                                                                 SET01910
      GO TO 10                                                                 SET01920
C*                                                                             SET01930
C*    HED1 CARD                                                                SET01940
C*                                                                             SET01950
  130 READ (30,500)HED1                                                        SET01960
      REWIND RRUNIT                                                            SET01970
      GO TO 10                                                                 SET01980
C*                                                                             SET01990
C*    HED2 CARD                                                                SET02000
C*                                                                             SET02010
  140 READ (30,500)HED2                                                        SET02020
      REWIND RRUNIT                                                            SET02030
      GO TO 10                                                                 SET02040
C*                                                                             SET02050
C*    DATE CARD                                                                SET02060
C*                                                                             SET02070
  150 READ(30,510) DATE                                                        SET02080
      REWIND RRUNIT                                                            SET02090
      GO TO 10                                                                 SET02100
C*                                                                             SET02110
C*    COMMENT CARD                                                             SET02120
C*                                                                             SET02130
  160 READ(30,500)COMENT                                                       SET02140
      REWIND RRUNIT                                                            SET02150
      GO TO 10                                                                 SET02160
C*                                                                             SET02170
C*    SYMBOLS CARD                                                             SET02180
C*                                                                             SET02190
  170 CONTINUE                                                                 SET02200
  180 ICNT=ICNT + 1                                                            SET02210
      IF (ICNT.GT.MAXPOP) GO TO 10                                             SET02220
      SYMMTX(ICNT)=BLANK                                                       SET02230
  190 M=NXTCHR(CARD,COL)                                                       SET02240
      IF (M.EQ.BLANK) GO TO 10                                                 SET02250
       IF(M.EQ.COMMA) GO TO 180                                                SET02260
      SYMMTX(ICNT) = M                                                         SET02270
  195 M=NXTCHR(CARD,COL)                                                       SET02280
      IF(M.EQ.BLANK)GO TO 10                                                   SET02290
      IF(M.NE.COMMA) GO TO 195                                                 SET02300
      GO TO 180                                                                SET02310
C*                                                                             SET02320
C*    MAXCLASS CARD (NO. CLASSES FOR THIS EXECUTION OF ISOCLS- STATISTIC       SET02330
C*                   FILE WILL BE WRITTEN AFTER 'NCLASS' CLASSES HAVE          SET02340
C*                   BEEN CLUSTERED)                                           SET02350
C*                                                                             SET02360
  200 IF(ITIME.EQ.1) GO TO 205                                                 SET02370
```

FILE: SETUP7 FORTRAN A

```
      WRITE(6,650)                                                          SET02380
      GO TO 10                                                              SET02390
  205 J=NUMBER(CARD,COL,NCLASS,ISTART)                                      SET02400
      GO TO 10                                                              SET02410
C*                                                                          SET02420
C*    MAP CARD  (NUMBER OF ITERATIONS TO OUTPUT MAP)                        SET02430
C*                                                                          SET02440
  210 J=NXTCHR(CARD,COL)                                                    SET02450
      IF (J.EQ.BLANK) GO TO 10                                              SET02460
      COL=COL-1                                                             SET02470
      J           = NUMBER(CARD,COL,MAP    ,ISTART)                     SET02480
      GO TO 10                                                              SET02490
C*                                                                          SET02500
C*    PUNCH CARD (PUNCH STATISTICS ON CARDS)                                SET02510
C*                                                                          SET02520
  215 M = FIND12(CARD,COL,CHAR)                                             SET02530
      IF (M .NE. 2)  PUNCH = 1                                              SET02540
      IF(M.NE.2)GO TO 245                                                   SET02550
      J           = NUMBER(CARD,COL,PUNCH ,ISTART)                        SET02560
      GO TO 245                                                             SET02570
C*                                                                          SET02580
C*    MAP FORMAT CARD                                                       SET02590
C*                                                                          SET02600
  220 M = NXTCHR(CARD,COL)                                                  SET02610
      IF (M .EQ. UBCD)  MAPFMT = 1                                          SET02620
      IF (M .EQ. LBCD)  MAPFMT = 2                                          SET02630
      IF(M.EQ.BLANK) MAPFMT=1                                               SET02640
      GO TO 10                                                              SET02650
C*                                                                          SET02660
C*    CLUSTERS CARD (MAX. NO. OF CLUSTERS PER CLASS)                        SET02670
C*                                                                          SET02680
  230 J=NXTCHR(CARD,COL)                                                    SET02690
      IF (J.EQ.BLANK) GO TO 10                                              SET02700
      COL=COL-1                                                             SET02710
      J           = NUMBER(CARD,COL,MAXCLS,ISTART)                        SET02720
      GO TO 10                                                              SET02730
C*                                                                          SET02740
C*    CHAIN CARD (CHAIN CLUSTERS WHICH ARE DLMIN UNITS APART)               SET02750
  235 ICHN=1                                                                SET02760
      J=FLTNUM(CARD,COL,CHNTHS,1)                                           SET02770
      GO TO 10                                                              SET02780
C                                                                           SET02790
C     OPTION CARD                                                           SET02800
C                                                                           SET02810
  240 J = NXTCHR(CARD,COL)                                                  SET02820
      IF (J .EQ. BLANK)  GO TO 10                                           SET02830
C                                                                           SET02840
C      ORDER COLOR KEYS                                                     SET02850
C                                                                           SET02860
      IF(J.EQ.OBCD) IORDER = 1                                              SET02870
C                                                                           SET02880
C      ERROR COMPUTATION                                                    SET02890
C                                                                           SET02900
      IF(J.NE.EBCD) GO TO 241                                               SET02910
      COL=COL+1                                                             SET02920
      J=NXTCHR(CARD,COL)                                                    SET02930
      IF(J.EQ.CBCD) ERCOMP=1                                                SET02940
C                                                                           SET02950
      GO TO 245                                                             SET02960
  241 CONTINUE                                                              SET02970
C                                                                           SET02980
C     PUNCH CARD                                                            SET02990
C                                                                           SET03000
      IF (J .EQ. PHCD)  GO TO 215                                           SET03010
C*                                                                          SET03020
C*    STATS                                                                 SET03030
C*                                                                          SET03040
      IF(J.NE.SBCD)  GO TO 242                                              SET03050
       J=NXTCHR(CARD,COL)                                                   SET03060
      IF(J.EQ.TBCD) STATKY=1                                                SET03070
  242 CONTINUE                                                              SET03080
C     CLUSTERS FOR MAPTAP                                                   SET03090
      IF(J.EQ.CBCD)MAPKEY=2                                                 SET03100
C*                                                                          SET03110
C*    FIND12 A COMMA                                                        SET03120
C*                                                                          SET03130
  245 J=FIND12(CARD,COL,COMVEC)                                             SET03140
      IF(J.LE.0)GO TO 10                                                    SET03150
      GO TO 240                                                             SET03160
```

FILE: SETUP7 FORTRAN A

```
C                                                                       SET03170
C     SEQUENCE CARD                                                     SET03180
C                                                                       SET03190
  244   I=1                                                           SET03200
  247   M=NXTCHR(CARD,COL)                                            SET03210
        IF(M.EQ.BLANK) GO TO 248                                      SET03220
         SEQUEN(I)=M                                                  SET03230
         I=I+1                                                        SET03240
         GO TO 247                                                    SET03250
  248 NOSEQ = I - 1                                                   SET03260
        GO TO 10                                                      SET03270
C                                                                     SET03280
C     PERCENT CARD                                                    SET03290
C                                                                     SET03300
  249   J=NXTCHR(CARD,COL)                                            SET03310
        IF(J.EQ.BLANK) GO TO 10                                       SET03320
         COL=COL-1                                                    SET03330
         J=NUMBER(CARD,COL,IPCT,J)                                    SET03340
        PERCEN=1.-FLOAT(IPCT)/100.                                    SET03350
         GO TO 10                                                     SET03360
C                                                                     SET03370
C     MEANS CARD                                                      SET03380
C                                                                     SET03390
  250 J = NXTCHR(CARD,COL)                                            SET03400
        IF(J .EQ. BLANK)     GO TO 10                                 SET03410
        IF(J .EQ. CRCD)      GO TO 255                                SET03420
        IF(J .NE. FRCD)      GO TO 10                                 SET03430
        ISOKEY=1                                                      SET03440
        GO TO 10                                                      SET03450
  255 ISOKEY=1                                                        SET03460
        CALL RDMEAN(ARRAY)                                            SET03470
        GO TO 10                                                      SET03480
C*                                                                    SET03490
C*     READ MODULE DECK AND WRITE TO INPUT STAT UNIT AND FILE.         SET03500
C*                                                                    SET03510
  256   SAVE1=SAVTAP                                                  SET03520
        SAVE2=STAFIL                                                  SET03530
        SAVTAP=INUNIT                                                 SET03540
        STAFIL=INFILE                                                 SET03550
        CALL CRDSTA(ARRAY,TOP)                                        SET03560
        SAVTAP=SAVE1                                                  SET03570
        STAFIL=SAVE2                                                  SET03580
        INITM=1                                                       SET03590
        GO TO 10                                                      SET03600
C                                                                     SET03610
C     DATA FILE CARD                                                  SET03620
C                                                                     SET03630
  260 M = NXTCHR(CARD,COL)                                            SET03640
        IF(M.EQ.BLANK) GO TO 10                                       SET03650
        IF (M.EQ.URCD) GO TO 265                                      SET03660
        IF(M.EQ.FRCD) GO TO 267                                       SET03670
  263 WRITE(6,750)                                                    SET03680
  750 FORMAT(' ERROR ON DATA FILE CARD')                              SET03690
        GO TO 10                                                      SET03700
  265 J = FIND12(CARD,COL,EQUVEC)                                     SET03710
        IF (J.EQ. -1)  GO TO 263                                      SET03720
        M = NUMBER(CARD,COL,DATAPE,ZERO)                              SET03730
        COL = COL - 1                                                 SET03740
        GO TO 260                                                     SET03750
  267 J = FIND12(CARD,COL,EQUVEC)                                     SET03760
        IF (J .EQ. -1)  GO TO 263                                     SET03770
        M = NUMBER(CARD,COL,DATFIL,ZERO)                              SET03780
        DATFIL = DATFIL - 1                                           SET03790
        IF (DATFIL .LT. 0)  DATFIL = 0                                SET03800
        COL = COL - 1                                                 SET03810
        GO TO 260                                                     SET03820
C                                                                     SET03830
C     STAT FILE CARD                                                  SET03840
C                                                                     SET03850
  270   M=NXTCHR(CARD,COL)                                            SET03860
  271   IF(M.EQ.IRCD) GO TO 278                                       SET03870
        IF(M.EQ.URCD) GO TO 275                                       SET03880
        IF(M.EQ.FRCD) GO TO 277                                       SET03890
        IF(M.EQ.ORCD) GO TO 272                                       SET03900
        IF(M.EQ.COMMA) GO TO 270                                      SET03910
        IF (M.EQ.BLANK) GO TO 10                                      SET03920
  273   WRITE(6,755)                                                  SET03930
  755   FORMAT(' ERROR ON STATFILE CARD')                             SET03940
        GO TO 10                                                      SET03950
```

FILE: SETUP7 FORTRAN A

```
 272    J=FIND12(CARD,COL,SLASH)                                          SET03960
        IF(J.EQ.-1)GO TO 273                                              SET03970
 274    M=NXTCHR(CARD,COL)                                                SET03980
       IF(M.EQ.COMMA) GO TO 274                                           SET03990
       IF(M.EQ.FRCD) GO TO 277                                            SET04000
       IF(M.NE.URCD) GO TO 271                                            SET04010
 275    J=FIND12(CARD,COL,EQUVEC)                                         SET04020
        IF(J.EQ.-1)GO TO 273                                              SET04030
        M=NUMBER(CARD,COL,SAVTAP,ZERO)                                    SET04040
        COL=COL-1                                                         SET04050
        GO TO 274                                                         SET04060
 277    J=FIND12(CARD,COL,EQUVEC)                                         SET04070
        IF(J.EQ.-1)GO TO 273                                              SET04080
        M=NUMBER(CARD,COL,STAFIL,ZERO)                                    SET04090
        COL=COL-1                                                         SET04100
        STAFIL=STAFIL-1                                                   SET04110
        IF(STAFIL.LT.0)STAFIL=0                                           SET04120
        GO TO 274                                                         SET04130
 278    J=FIND12(CARD,COL,SLASH)                                          SET04140
        IF(J.EQ.-1)GO TO 273                                              SET04150
       INITM = 1                                                          SET04160
 279    M=NXTCHR(CARD,COL)                                                SET04170
       IF(M.EQ.COMMA) GO TO 279                                           SET04180
       IF(M.EQ.FRCD) GO TO 282                                            SET04190
       IF(M.NE.URCD) GO TO 271                                            SET04200
 281    J=FIND12(CARD,COL,EQUVEC)                                         SET04210
        IF(J.EQ.-1)GO TO 273                                              SET04220
        M=NUMBER(CARD,COL,INUNIT,ZERO)                                    SET04230
        COL=COL-1                                                         SET04240
        GO TO 279                                                         SET04250
 282    J=FIND12(CARD,COL,EQUVEC)                                         SET04260
        IF(J.EQ.-1)GO TO 273                                              SET04270
        M=NUMBER(CARD,COL,INFILE,ZERO)                                    SET04280
        COL=COL-1                                                         SET04290
        GO TO 279                                                         SET04300
C*                                                                        SET04310
C*      SUBCLASSES CARD--USE THE MEANS FOR THESE SUBCLASSES FROM THE      SET04320
C*      STAT FILE FOR INITIAL MEANS                                       SET04330
C*                                                                        SET04340
 283    NOSUB2=NUMBER(CARD,COL,SUBVEC,NOSUB2)                             SET04350
        GO TO 10                                                          SET04360
C*                                                                        SET04370
C*      MINIMUM POPULATION FOR STATISTICS PASS.                           SET04380
C*                                                                        SET04390
 284    M=NUMBER(CARD,COL,PMIN,ZERO)                                      SET04400
        GO TO 10                                                          SET04410
C      DOTFIL INPUT/UNIT=N,FILE=M                                         SET04420
C      OR UNIT=N,FILE=M                                                   SET04430
 300   J=NXTCHR(CARD,COL)                                                 SET04440
 301    IF (J.EQ.BLANK) GO TO 320                                         SET04450
       IF(J.NE.URCD) GO TO 305                                            SET04460
       J=FIND12(CARD,COL,EQUVEC)                                          SET04470
       IF (J.NE.2) GO TO 320                                              SET04480
       ISTART=0                                                           SET04490
       J=NUMBER(CARD,COL,ARRAY(TOP-30),ISTART)                            SET04500
        DOTUNT = ARRAY(TOP - 30)                                          SET04510
       J=FIND12(CARD,COL,EQUVEC)                                          SET04520
       IF (J.NE.2) GO TO 320                                              SET04530
       ISTART=0                                                           SET04540
       J=NUMBER(CARD,COL,ARRAY(TOP-30),ISTART)                            SET04550
       DOTFIL=ARRAY(TOP-30)                                               SET04560
       DOTFIL = DOTFIL - 1                                                SET04570
       GO TO 10                                                           SET04580
 305   IF(J.NE.IRCD) GO TO 310                                            SET04590
       J=FIND12(CARD,COL,SLASH)                                           SET04600
       IF (J.NE.2) GO TO 320                                              SET04610
       J = NXTCHR(CARD,COL)                                               SET04620
       IF (J.EQ.FRCD) GO TO 315                                           SET04630
       IF (J.EQ.URCD) GO TO 301                                           SET04640
       GO TO 320                                                          SET04650
 310   IF (J.NE.FRCD) GO TO 320                                           SET04660
 315   J = FIND12(CARD,COL,EQUVEC)                                        SET04670
       IF (J.NE.2) GO TO 320                                              SET04680
       ISTART = 0                                                         SET04690
       J = NUMBER(CARD,COL,ARRAY(TOP - 30),ISTART)                        SET04700
       DOTFIL = ARRAY(TOP - 30) - 1                                       SET04710
       J = FIND12(CARD,COL,EQUVEC)                                        SET04720
       IF (J.NE.2) GO TO 320                                              SET04730
       ISTART = 0                                                         SET04740
```

FILE: SETUP7 FORTRAN A

```
      J = NUMBER(CARD,COL,ARRAY(TOP - 30),ISTART)                       SET04750
      DOTUNT = ARRAY(TOP - 30)                                          SET04760
      GO TO 10                                                          SET04770
  320 WRITE(6,760)                                                      SET04780
  760 FORMAT(' ERROR ON DOTFILE CARD')                                  SET04790
      GO TO 10                                                          SET04800
C     SUNANG                                                            SET04810
  340 J=NXTCHR(CARD,COL)                                                SET04820
      IF(J.NE.THCD) GO TO 345                                           SET04830
      ISUNT=1                                                           SET04840
      GO TO 10                                                          SET04850
  345 ISTART=0                                                          SET04860
      COL = COL - 1                                                     SET04870
      J=NUMBER(CARD,COL,ARRAY(TOP-30),ISTART)                           SET04880
       IF (J.GT.8) J = 8                                                SET04890
      IF(J.EQ.0) GO TO 10                                               SET04900
      ISUNC=0                                                           SET04910
      DO 350 JJ=1,J                                                     SET04920
      ISUNC=ISUNC+1                                                     SET04930
  350 SUNANG(JJ)=ARRAY(TOP-31+ISUNC)                                    SET04940
      GO TO 10                                                          SET04950
C     DOTS                                                              SET04960
  380 ISTART=0                                                          SET04970
      J = NUMBER(CARD,COL,STDOTS(NSDOTS + 1),ISTART)                    SET04980
      NSDOTS = NSDOTS + J                                               SET04990
      IF(NSDOTS.GT.60) NSDOTS=60                                        SET05000
      GO TO 10                                                          SET05010
C                                                                       SET05020
C     *END* CARD                                                        SET05030
C                                                                       SET05040
  280 CONTINUE                                                          SET05050
      IF (NOFEAT .GT. 0)  GO TO 285                                     SET05060
      NOFEAT=30                                                         SET05070
      DO 261 I=1,30                                                     SET05080
  261 FETVEC(I)=I                                                       SET05090
  285 CONTINUE                                                          SET05100
C*                                                                      SET05110
C*    PRINT USER REQUEST                                                SET05120
C*                                                                      SET05130
      WRITE(6,660)                                                      SET05140
      WRITE(6,670)ISTOP,NMIN,KRN,MAP,MAXCLS,NCLASS,                     SET05150
     *   (FETVEC(I),I=1,NOFEAT)                                         SET05160
      WRITE(6,680)DLMIN,STOMAX                                          SET05170
      WRITE (6,685) IPCT,PMIN,NSDOTS,ISUNC,ISUNT                        SET05180
      IF(SPTRIG.EQ.1)WRITE(6,690)SEP                                    SET05190
      IF(ICHN.EQ.1) WRITE(6,710)CHNTHS                                  SET05200
      IF(PUNCH.EQ.1) WRITE(6,700)                                       SET05210
       IF(IORDER.EQ.1)WRITE(6,715)                                      SET05220
       IF(MAPFMT.EQ.1)WRITE(6,720)                                      SET05230
       IF(MAPFMT.EQ.2)WRITE(6,725)                                      SET05240
      IF (NOFEAT .GT. NMIN)  WRITE(6,740)                               SET05250
      RETURN                                                            SET05260
C                                                                       SET05270
  480 FORMAT(A4,6X,62A1)                                                SET05280
  490 FORMAT(' INVALID INPUT CARD--IGNORED'/T5,A4,6X,62A1)               SET05290
  500 FORMAT(10X,15A4)                                                  SET05300
  510 FORMAT(10X,3A4)                                                   SET05310
  550 FORMAT(7X,A4,4X,62A1)                                             SET05320
  630 FORMAT(//' INPUT SUMMARY'//)                                      SET05330
  640 FORMAT(' CHANNELS CANNOT BE CHANGED UNTIL THIS EXECTUION OF ISOCLS SET05340
     * IS COMPLETED')                                                   SET05350
  650 FORMAT(' NO. OF CLASSES CANNOT BE CHANGED UNTIL THIS EXECUTION OF SET05360
     * ISOCLS IS COMPLETED')                                           SET05370
  660 FORMAT(///' YOU HAVE SELECTED THE FOLLOWING PARAMETER VALUES AND  SET05380
     * OPTIONS'//)                                                     SET05390
  670 FORMAT(' STOP AFTER',I5,' ITERATION(S)' /                         SET05400
     *       ' ALLOW A MINIMUM OF',I6,' PIXELS PER CLUSTER' /           SET05410
     *       ' PRINT A CLUSTER SUMMARY EVERY',I5,' ITERATION(S)'//      SET05420
     *       ' PRINT A CLUSTER MAP EVERY',I5,' ITERATION(S)' /          SET05430
     *       ' ALLOW A MAXIMUM OF',I5,' CLUSTERS PER CLASS' /           SET05440
     *       ' THE STATISTICS FILE WILL BE WRITTEN AFTER',I4,' CLASS(ES)SET05450
     * HAVE BEEN CLUSTERED' /                                           SET05460
     *       ' CHANNELS ARE---', 30I3 )                                 SET05470
  680 FORMAT(' DLMIN =', F7.3 /' STOMAX =',F7.3)                        SET05480
  685 FORMAT(' PERCENT =',I5   /' PMIN =',I5 /' NSDOTS =',I5 /          SET05490
     *' NO. SUN ANGLES FROM CARDS =',I5 /' SUN ANGLE TAPE SW =',I5/)    SET05500
  690 FORMAT(' SEP   =', F7.3)                                          SET05510
  700 FORMAT(' PUNCH THE MODULE STAT DECK')                             SET05520
  710 FORMAT(' CHAIN CLUSTERS WHICH ARE',F7.3,' UNITS APART')           SET05530
```

FILE: SETUP7 FORTRAN A

```
                                                                                SET05540
  715   FORMAT(1X,'ORDER COLOR KEYS')                                           SET05550
  720   FORMAT(1X,'WRITE A CLUSTER MAP OUTPUT TAPE IN UNIVERSAL FORMAT')        SET05560
  725   FORMAT(1X,'WRITE A CLUSTER MAP OUTPUT TAPE IN LARSYS II FORMAT')        SET05570
  740 FORMAT(//' **WARNING**  NMIN IS LESS THAN NO. OF CHANNELS,COVARIANCES WILSET05580
     *L NOT BE INVERTIBLE')                                                     SET05590
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

10. SELECT PROCESSOR

FILE: SELECT FORTRAN A

```
      SUBROUTINE SELECT(ARRAY,TOP)                                          SEL00010
C                                                                           SEL00020
      IMPLICIT INTEGER (A-H,O-Z)                                            SEL00030
C|-------------------------------------------------------------------------|SEL00040
C|-------------------------------------------------------------------------|SEL00050
C||                                                                        |SEL00060
C||    CALL..    CALL SELECT(ARRAY,TOP)                                    |SEL00070
C||    ARGS..    ARRAY -  SEE 'MONTOR'                                     |SEL00080
C||              TOP   -  SEE 'MONTOR'                                     |SEL00090
C||                                                                        |SEL00100
C||                                                                        |SEL00110
C||                                                                        |SEL00120
C||    PURPOSE.. COORDINATES THE VARIOUS ROUTINES                          |SEL00130
C||              FOR 'FEATURE SELECTION' STEP                              |SEL00140
C||                                                                        |SEL00150
C||    RETURNS.. NONE                                                      |SEL00160
C||                                                                        |SEL00170
C|-------------------------------------------------------------------------|SEL00180
C|-------------------------------------------------------------------------|SEL00190
C                                                                           SEL00200
C                                                                           SEL00210
C                                                                           SEL00220
C                                                                           SEL00230
C                                                                           SEL00240
C     ------------------------------------------------------------------    SEL00250
C                                                                           SEL00260
C                                                                           SEL00270
C     INCLUDE COMMA1.LIST                                                   SEL00280
C     ------------------------------------------------------------------    SEL00290
C                                                                           SEL00300
C     INCLUDE COMMAA.LIST                                                   SEL00310
      DIMENSION ARRAY(1),SUBRAY(12000)                                      SEL00320
      DATA SUBS1//12000/                                                    SEL00330
C*    THE ARRAY SUBRAY IS USED IN SELECT FOR VARIABLE DIMENSIONING          SEL00340
C     INCLUDE COMMA7.LIST                                                   SEL00350
      COMMON/INFO/NOCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFLD2,                SEL00360
     *            AVF32,COVAR2,CLSID2,SUBID2,SUBDS2,FLDSV2,VERTX2,          SEL00370
     *            FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),              SEL00380
     *            KEYPTS(50),NOGRP,GRPNAM(60),GRPDEX(61),                   SEL00390
     *            GRPCHR(61),GROUPS(124)                                    SEL00400
      COMMON/GLOBAL/HEAD(32),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             SEL00410
     *            HISFIL,HISKEY,TRFORM,ERIPTR,ERRKEY,MAPUNT,NOFILE,         SEL00420
     *    DRUMAD,DRMNOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    SEL00430
     *  ,NWSTUR,NWSTFI,SCTUUN,NAMFIL                                        SEL00440
     *  ,DOTUNT,DOTFIL,ACHPAS,TRNSFL,BMTHFL,HISTFL,PCHUNT,                  SEL00450
     *  CRUNIT,PRTUNT,MEND10                                                SEL00460
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                   COM00010
     *    INCVEC(30),ICOUNT,SETWGT,EVALRF(1100),FETVC4(30)                  COM00020
     *   ,DOFET,VARS74,CUMPAS,DTAB4,WGHS14,RESTVC(10),DIVSIZ                COM00030
     *   ,STATKY,ADRESD,ADWESR,ADWESF,ADRSM1,ADRSM2                         COM00040
      INTEGER ADWESD,ADWESR,ADWESF,ADRSM1,ADRSM2,STATKY                     COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                   COM00060
CEND                                                                        SEL00550
C*                                                                          SEL00560
C*    COMMON -LOC- FSL IS USED ONLY BY THE 'SELECT' PROCESSOR               SEL00570
C*                                                                          SEL00580
C*                                                                          SEL00590
C*     DEFINITIONS                                                          SEL00600
C*         PRCKEY - KEY INDICATING WHICH PROCEDURE TO EXECUTE               SEL00610
C*                  1 - EXHAUSTIVE SEARCH                                   SEL00620
C*                  2 - WITHOUT REPLACEMENT                                 SEL00630
C*                  3 - DAVIDON                                             SEL00640
C*                  4 - EVALUATE A USER INPUT B-MATRIX                      SEL00650
C*                  5 - EVALUATE SPECIFIC CHANNELS INPUT BY USER            SEL00660
C*         CRIKEY - KEY INDICATING WHICH CRITERIA IS TO BE USED             SEL00670
C*                  FOR MEASURING SEPARABILITY.                             SEL00680
C*                  1 - WEIGHTED AV. DIVERGENCE                             SEL00690
C*                  2 - WEIGHTED AV. TRANSFORMED DIVERGENCE                 SEL00700
C*                  3 - WEIGHTED AV. BHATTACHARYYA DISTANCE                 SEL00710
C*         INCFET - NO. OF CHANNELS TO INCLUDE IN THE 'BEST' SET. THIS      SEL00720
C*                  USER INPUT ON THE 'INCLUDE' CONTROL CARD.               SEL00730
C*         INCVEC - VECTOR CONTAINING THE CHANNELS TO BE INCLUDED. INPUT    SEL00740
C*                  ON THE 'INCLUDE' CARD. MEANINGFUL ONLY IF WITHOUT       SEL00750
C*                  REPLACEMENT PROCEDURE IS EXECUTED.                      SEL00760
C*         ICOUNT - MAX. NO. OF ITERATIONS IN DAVIDON PROCEDURE.            SEL00770
C*         SETWGT - TRIGGER INDICATING WHETHER OR NOT DEFAULT WEIGHTS       SEL00780
C*                  ARE TO BE SET.                                          SEL00790
C*         EVALRF - BUFFER CONTAINING ALL USER 'EVALUATE' REQUESTS.         SEL00800
```

FILE: SELECT FORTRAN A

```
C*                    EVALBF(1) = NO. OF CHANNELS FOR FIRST REQUEST          SEL00810
C*                          (2-N)= A SET OF CHANNELS TO BE EVALUATED         SEL00820
C*                          (N+1)=NO. OF CHANNELS ON SECOND REQUEST          SEL00830
C*                            -                                              SEL00840
C*                          ETC.                                             SEL00850
C*          CFAC    - CONSTANT FACTOR NEEDED IN COMPUTATION OF AVERAGE       SEL00860
C*                    DIVERGENCE. DOUBLE PRECISION VARIABLE COMPUTED IN      SEL00870
C*                    'PRELIM'.                                              SEL00880
C*          TOTMSR  - SEPARABILITY MEASURE COMPUTED USING ALL CHANNELS.      SEL00890
C*                    DOUBLE PRE. VARIABLE COMPUTED IN 'PRELIM'.             SEL00900
C*          FETVC4  - VECTOR CONTAINING THE SET OF CHANNELS SELECTED AS      SEL00910
C*                    THE 'BEST SET'.                                        SEL00920
C*          NOFET4  - DIMENSION OF FETVC4                                    SEL00930
C*          SEPMSR  - SEPARABILITY MEASURE COMPUTED USING THE 'BEST SET'     SEL00940
C*                    OF CHANNELS.                                           SEL00950
C*          VARSZ4  - SIZE OF COVARIANCE MATRIX FOR 'BEST' SET.              SEL00960
C*                    NOFET4 * (NOFET4+1)/2                                  SEL00970
C*          COMBAS  - NEXT AVAILABLE ADDRESS IN 'ARRAY' FOR STORAGE OF DATA  SEL00980
C*          DTABA   - BEGINNING ADDRESS IN 'ARRAY' FOR INTERCLASS SEP.       SEL00990
C*                    MEASURE TABLE.                                         SEL01000
C*          WGHS14  - BEGINNING ADDRESS IN 'ARRAY' FOR INTERCLASS WEIGHTS.   SEL01010
C*          BESTVC  - VECTOR CONTAINING THE 'BEST' REQUEST FROM USER INPUT   SEL01020
C*          DIVSIZ  - SIZE OF INTERCLASS SEP. MEASURE TABLE.                 SEL01030
C*          STATKY  - TRIGGER INDICATING WHETHER OR NOT TO PRINT STATS.      SEL01040
C*                                                                           SEL01050
C*     THE FOLLOWING VARIABLES ARE BEGINNING ADDRESSES FOR 'S' SCRATCH       SEL01060
C*     I/O AREAS ON THE RANDOM ACCESS DRUM FILE.                             SEL01070
C*                                                                           SEL01080
C*          ADRESD - DRUM ADDRESS WHERE INTERCLASS SEP. MEASURES COMPUTED    SEL01090
C*                   USING ALL CHANNELS ARE STORED.                          SEL01100
C*          ADRESP --                                                        SEL01110
C*          ADRESF -- THESE FOUR ADRESSES ARE FOR TEMPORARY STORAGE OF       SEL01120
C*          ADRSM1 -- DATA ARRAYS IN THE DAVIDON PROCEDURE ONLY.             SEL01130
C*          ADRSM2 --                                                        SEL01140
C*                                                                           SEL01150
C*                                                                           SEL01160
C     ------------------------------------------------------------------    SEL01170
C     ------------------------------------------------------------------    SEL01180
C                                                                            SEL01190
      COMMON/BESTSV/ KPRPTS(60), IPRIOR, KBEST, NCPASS                       SEL01200
      DIMENSION ILABLA(13),ILAHLY(13)                                        SEL01210
      DIMENSION FETVEC(30)                                                   SEL01220
      DIMENSION  PER(6),INDPER(6),PERM(24A)                                  SEL01230
      DATA INDPER/1,13,5A,13M,213,249/                                       SEL01240
      DATA PERM/1,2,1,3,1,4,1,5,1,6,1,7,                                     SEL01250
     *    1,2,3,1,2,4,1,2,5,1,2,6,1,2,7,1,3,4,1,3,5,1,3,6,1,3,7,            SEL01260
     *    1,4,5,1,4,6,1,4,7,1,5,6,1,5,7,1,6,7,                              SEL01270
     *    1,2,3,4,1,2,3,5,1,2,3,6,1,2,3,7,1,2,4,5,1,2,4,6,                  SEL01280
     *    1,2,4,7,1,2,5,6,1,2,5,7,1,2,6,7,1,3,4,5,1,3,4,6,                  SEL01290
     *    1,3,4,7,1,3,5,6,1,3,5,7,1,3,6,7,1,4,5,6,                          SEL01300
     *    1,4,5,7,1,4,6,7,1,5,6,7,                                          SEL01310
     *    1,2,3,4,5,1,2,3,4,6,1,2,3,4,7,1,2,3,5,6,1,2,3,5,7,                SEL01320
     *    1,2,3,6,7,1,2,4,5,6,1,2,4,5,7,1,2,4,6,7,1,2,5,6,7,                SEL01330
     *    1,3,4,5,6,1,3,4,5,7,1,3,4,6,7,1,3,5,6,7,1,4,5,6,7,                SEL01340
     *    1,2,3,4,5,6,1,2,3,4,5,7,1,2,3,4,6,7,1,2,3,5,6,7,                  SEL01350
     *    1,2,4,5,6,7,1,3,4,5,6,7/                                          SEL01360
      DIMENSION IPSCHK(A)                                                    SEL01370
      DIMENSION FETSAV(30),FTSAV(30)                                         SEL01380
      DOUBLE PRECISION SMALL                                                 SEL01390
      SMALL = 1.0E-35                                                        SEL01400
      JTIME = 1                                                              SEL01410
   10 CONTINUE                                                               SEL01420
      CALL SETUP4(ARRAY,TOP,STOPFG,JTIME,SUBRAY,SUMSIZ)                      SEL01430
      IF(STOPFG.NE.0) GO TO 9                                                SEL01440
      IF(PRCKEY.NE.A) GO TO 9                                                SEL01450
      DO A I=1,4                                                             SEL01460
    A IPSCHK(I) = 0                                                          SEL01470
      DO 7 I=1,NOFET2                                                        SEL01480
      K=(FETVC2(I)-1)/NCPASS +1                                              SEL01490
    7 IPSCHK(K) = 1                                                          SEL01500
      NPASS = 0                                                              SEL01510
      DO A I=1,4                                                             SEL01520
    A NPASS = NPASS + IPSCHK(I)                                              SEL01530
      NFPPS = NOFET2/NPASS                                                   SEL01540
      TDUM = NFPPS*NPASS                                                     SEL01550
      IF(NOFET2.NE.IDUM) GO TO 90                                            SEL01560
      IF(KBEST.LT.2.OR.KBEST.GE.NPASS) GO TO 90                              SEL01570
      NOFET4 = NFPPS*KBEST                                                   SEL01580
      NFSAVE = NOFET4                                                        SEL01590
```

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SELECT FORTRAN A

```
      KPASS = KREST                                                          SEL01600
    9 CONTINUE                                                               SEL01610
      JTIME = JTIME+1                                                        SEL01620
      IF(STOPFG.EQ.0)GO TO 5                                                 SEL01630
C*    SET FETVC2 FOR CLASSIFY                                                SEL01640
      NOFET4=NFSAVE                                                          SEL01650
      DO 4 I=1,NOFET4                                                        SEL01660
    4 FETVC2(I)=FETVC4(I)                                                    SEL01670
      CALL ORDER(FETVC2,NOFET4)                                              SEL01680
      NOFET2 = NOFET4                                                        SEL01690
      RETURN                                                                 SEL01700
    5 CONTINUE                                                               SEL01710
C*                                                                           SEL01720
C*    SET ADRESD FOR RANDOM ACCESS DRUM FILE                                 SEL01730
C*                                                                           SEL01740
      ADRESD=DRUMAD                                                          SEL01750
      JREST = 0                                                              SEL01760
C*    PRELIM--TAKE CARE OF PRELIMINARIES                                     SEL01770
C*    COMPUTE SEPARABILITY MEASURE AND INTERCLASS MEASURES USING ALL         SEL01780
C*    FEATURES, AND SAVE ON SCRATCH FILE FOR LATER PRINTING. IF CRIKEY=1     SEL01790
C*    SET DEFAULT WEIGHTS IF WEIGHTS NOT INPUT, ALSO COMPUTE 'S' MATRIX.     SEL01800
C*                                                                           SEL01810
      SBASE=1                                                                SEL01820
      S1=SBASE                                                               SEL01830
      SBASE=S1+NOCLS2*VARS22+1                                               SEL01840
      IF(CRIKEY.NE.1)SBASE=1                                                 SEL01850
      SLEFT=SUBSIZ-SBASE                                                     SEL01860
      CALL PRELIM(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(DTAB4),                   SEL01870
     *            ARRAY(WGHS14),SUBRAY(S1),SUBRAY(SBASE),SLEFT)              SEL01880
      IF(PRCKEY .EQ. 5) GO TO 60                                             SEL01890
      IF(PRCKEY.EQ.4)GO TO 12                                                SEL01900
      IF(PRCKEY.EQ.6) GO TO 15                                               SEL01910
   11 JREST=JREST+1                                                          SEL01920
      NFSAVE=NOFET4                                                          SEL01930
      NOFET4=FESTVC(JREST)                                                   SEL01940
      IF(NOFET4.LE.0)GO TO 60                                                SEL01950
C*                                                                           SEL01960
C*    IF DAVIDON PROCEDURE INDICATED, FIND BEST SET OF FEATURES BY           SEL01970
C*    WITHOUT REPLACEMENT. IF FIRST GUESS B-MATRIX WAS NOT INPUT.            SEL01980
C*                                                                           SEL01990
   12 IF(PRCKEY.NE.3)GO TO 15                                                SEL02000
C*                                                                           SEL02010
C*    SET ADDRESSES FOR RANDOM ACCESS DRUM FILE                              SEL02020
C*                                                                           SEL02030
      ADRESP=ADRESD+DIVSIZ*2                                                 SEL02040
      ADRESF=ADRESP+NOFET4*NOFET2*2                                          SEL02050
      ADRSH1=ADRESF+NOFET4*NOFET2*2                                          SEL02060
      ADRSH2=ADRSH1+(NOFET4*NOFET2*2)**2                                     SEL02070
C*                                                                           SEL02080
C*    WAS FIRST GUESS B-MATRIX INPUT                                         SEL02090
C*                                                                           SEL02100
      IF(BMKEY.EQ.1)GO TO 15                                                 SEL02110
      SAVPRC=3                                                               SEL02120
      PRCKEY=2                                                               SEL02130
C*                                                                           SEL02140
C*    COMPUTE BASES FOR ARRAYS OF 'NEST' SET OF FEATURES                     SEL02150
C*    TRANSFORMED COVARIANCES AND MEANS STORED IN DOUBLE PRECISION           SEL02160
C*                                                                           SEL02170
   15 VARS74=NOFET4*(NOFET4+1)/2                                             SEL02180
      COVAR4=CORBAS                                                          SEL02190
      AVAR4=COVAR4 + NOCLS2*VARS74*2                                         SEL02200
      CORBSS = AVAR4 + NOCLS2 * NOFET4 *2                                    SEL02210
      IF(CORBSS .LE. TOP) GO TO 20                                           SEL02220
      WRITE(6,200) CORBSS                                                    SEL02230
      CALL CMERR                                                             SEL02240
C*                                                                           SEL02250
C*    SUBRAY STORAGE - STORE 'S' ARRAYS ONLY IF CRIKEY=1. STORE PARTIALS     SEL02260
C*    ONLY IF PRCKEY=3. STORE B-MATRIX IF PRCKEY=3 OR 4.                     SEL02270
C*                                                                           SEL02280
   20 S2=S1 + NOCLS2*VARS22                                                  SEL02290
      P1=S2 + NOCLS2*VARS74*2                                                SEL02300
      IF(CRIKEY.NE.1)P1=1                                                    SEL02310
      P1=P1 + NOFET4*NOFET2*2                                                SEL02320
      SBASE=P1 + NOFET4*NOFET2*2                                             SEL02330
      IF(PRCKEY.NE.3)SBASE=P1                                                SEL02340
      IF(PRCKEY.LT.3)SBASE=H1                                                SEL02350
      SLEFT=SUBSIZ-SBASE                                                     SEL02360
      IF(SBASE.LE.SUBSIZ)GO TO 25                                            SEL02370
      WRITE(6,100)SBASE                                                      SEL02380
```

FILE: SELECT FORTRAN A

```
      CALL CMERR                                                        SEL02390
C*                                                                      SEL02400
C*    PERFORM THE OPTIMIZATION PROCEDURE INDICATED BY PRCKEY            SEL02410
C*                                                                      SEL02420
 25   GO TO (30,35,40,45,85,87),PRCKEY                                  SEL02430
C*                                                                      SEL02440
C*    EXHAUSTIVE SEARCH PROCEDURE                                       SEL02450
C*                                                                      SEL02460
 30   CALL EXSRCH(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(DTAB4),ARRAY(WGHS14) SEL02470
     *           ,ARRAY(COVAR4),ARRAY(AVAR4),SUBRAY(S1),SUBRAY(S2),     SEL02480
     *            SUBRAY(SBASE),SLEFT)                                  SEL02490
      GO TO 50                                                          SEL02500
C*                                                                      SEL02510
C*    WITHOUT REPLACEMENT PROCEDURE                                     SEL02520
C*                                                                      SEL02530
 35   CALL WHRPLC(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(DTAB4),ARRAY(WGHS14) SEL02540
     *           ,ARRAY(COVAR4),ARRAY(AVAR4),SUBRAY(S1),SUBRAY(S2),     SEL02550
     *            SUBRAY(SBASE),SLEFT)                                  SEL02560
      GO TO 50                                                          SEL02570
C*                                                                      SEL02580
C*    DAVIDON PROCEDURE                                                 SEL02590
C*                                                                      SEL02600
 40   CALL DAVIDN(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(DTAB4),ARRAY(WGHS14) SEL02610
     *           ,ARRAY(COVAR4),ARRAY(AVAR4),SUBRAY(S1),SUBRAY(S2)      SEL02620
     *           ,SUBRAY(B1),SUBRAY(P1),SUBRAY(SBASE),SLEFT)            SEL02630
      CALL WRTBMT(SUBRAY(B1),NOFET4,NOFET2,FETVC2)                      SEL02640
      GO TO 50                                                          SEL02650
C*                                                                      SEL02660
C*    USER INPUT H-MATRIX                                               SEL02670
 45   CALL USERIN(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(DTAB4),ARRAY(WGHS14) SEL02680
     *           ,ARRAY(COVAR4),ARRAY(AVAR4),SUBRAY(S1),SUBRAY(S2)      SEL02690
     *           ,SUBRAY(B1),SUBRAY(SBASE),SLEFT)                       SEL02700
C*                                                                      SEL02710
C*    GENERATE REPORTS                                                  SEL02720
C*                                                                      SEL02730
 50   CALL GENRPT(ARRAY(CLSID2),ARRAY(WGHS14),ARRAY(DTAB4),            SEL02740
     *    SUBRAY(SBASE),SLEFT,FETVEC)                                   SEL02750
      CALL PLOT(SUBRAY(SBASE),ARRAY(DTAB4),DIVSIZ,MAXX,ILABLX,ILABLY,   SEL02760
     *          ICODE,IOPT)                                           SEL02770
C*                                                                      SEL02780
      IF(SAVPRC.NE.3)GO TO 11                                           SEL02790
      SAVPRC=0                                                          SEL02800
      PRCKEY=3                                                          SEL02810
      GO TO 20                                                          SEL02820
C*                                                                      SEL02830
C*    PERFORM EVALUATE REQUEST                                          SEL02840
C*                                                                      SEL02850
 60   IV=1                                                              SEL02860
      ISAVE=PRCKEY                                                      SEL02870
      PRCKEY=5                                                          SEL02880
 70   NOFET4=EVALBF(IV)                                                 SEL02890
      IF(NOFET4.GT.0)GO TO 75                                           SEL02900
      PRCKEY=ISAVE                                                      SEL02910
      GO TO 10                                                          SEL02920
 75   DO 80 I=1,NOFET4                                                  SEL02930
      IV=IV+1                                                           SEL02940
 80   FETVEC(I) = EVALBF(IV)                                            SEL02950
C                                                                       SEL02960
C     RENUMBERING CHANNELS IN REFRENCE TO SUBSET OF CHANNELS            SEL02970
C                                                                       SEL02980
      DO 82 I=1,NOFET2                                                  SEL02990
      DO 82 J=1,NOFET4                                                  SEL03000
      IF (FETVEC(J) .NE. FETVC2(I)) GO TO 82                            SEL03010
      FETVC4(J) = I                                                     SEL03020
 82   CONTINUE                                                          SEL03030
      CALL ORDER(FETVC4,NOFET4)                                         SEL03040
C*                                                                      SEL03050
C*    GO  COMPUTE BASE ADDRESSES FOR REDUCED ARRAYS                     SEL03060
C*                                                                      SEL03070
      GO TO 15                                                          SEL03080
 85   CONTINUE                                                          SEL03090
      CALL EVLEFT(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(DTAB4),ARRAY(WGHS14) SEL03100
     *           ,ARRAY(COVAR4),ARRAY(AVAR4),SUBRAY(S1),SUBRAY(S2)      SEL03110
     *           ,SUBRAY(SBASE),SLEFT)                                  SEL03120
      CALL GENRPT(ARRAY(CLSID2),ARRAY(WGHS14),ARRAY(DTAB4)              SEL03130
     *  ,SUBRAY(SBASE),SLEFT,FETVEC)                                   SEL03140
      CALL PLOT(SUBRAY(SBASE),ARRAY(DTAB4),DIVSIZ,MAXX,ILABLX,ILABLY,   SEL03150
     *          ICODE,IOPT)                                           SEL03160
      IV=IV+1                                                           SEL03170
```

```
FILE: SELECT   FORTRAN  A

      ) TO 70                                                                   SEL03180
C      EST K OF N PASSES                                                        SEL03190
 87   CONTINUE                                                                  SEL03200
 91   NCNT = NPASS + 1 - KPASS                                                  SEL03210
      IDUM = INDPER(KPASS - 1)                                                  SEL03220
      NCNTP = (INDPER(KPASS) - IDUM)/KPASS                                      SEL03230
      DO 99 I=1,NCNT                                                            SEL03240
      DO 98 II=1,NCNTP                                                          SEL03250
      DO 92 III=1,KPASS                                                         SEL03260
      IDM=IDUM+(II-1)*KPASS+III-1                                               SEL03270
      PER(III) = PERM(IDM) + I -1                                               SEL03280
      IF (PER(III).GT.KPASS) GO TO 97                                           SEL03290
 92   CONTINUE                                                                  SEL03300
      DO 94 III=1,KPASS                                                         SEL03310
      IDM=PER(III)                                                              SEL03320
      DO 93 IIII = 1,NFPPS                                                      SEL03330
      IDMA = NFPPS*(III-1) + IIII                                               SEL03340
      IDMB = NFPPS*(IDM-1) + IIII                                               SEL03350
      FETVC4(IDMA) = IDMB                                                       SEL03360
 93   FETVEC(IDMA) = FETVC2(IDMB)                                               SEL03370
 94   CONTINUE                                                                  SEL03380
      CALL EVLFET(ARRAY(COVAR2),ARRAY(AVAR2),                                   SEL03390
     *   ARRAY(DTAR4),ARRAY(WGHS14),ARRAY(COVAR4),ARRAY(AVAR4),                 SEL03400
     *    SUBRAY(S1),SUBRAY(S2),SUBRAY(SBASE),SLEFT)                            SEL03410
      IF(SMALL.LT.SEPMSR) GO TO 96                                              SEL03420
      DO 95 III = 1,NOFET4                                                      SEL03430
      FTSAV(III) = FETVEC(III)                                                  SEL03440
 95   FETSAV(III) = FETVC4(III)                                                 SEL03450
      SMALL = SEPMSR                                                            SEL03460
 96   CONTINUE                                                                  SEL03470
      CALL GENPRT(ARRAY(CLSID2),ARRAY(WGHS14),                                  SEL03480
     *    ARRAY(DTAR4),SUBRAY(SBASE),SLEFT,FETVEC)                              SEL03490
 97   CONTINUE                                                                  SEL03500
 98   CONTINUE                                                                  SEL03510
 99   CONTINUE                                                                  SEL03520
      DO 101 I=1,NOFET4                                                         SEL03530
      FETVEC(I) = FTSAV(I)                                                      SEL03540
 101  FETVC4(I) = FTSAV(I)                                                      SEL03550
      WRITE(6,1010)                                                             SEL03560
1010 FORMAT(1H1,' BEST SEPARABILITY MEASURE',//)                                SEL03570
      WRITE(6,1000) SMALL                                                       SEL03580
      WRITE(6,1020)                                                             SEL03590
1020 FORMAT(//' CORRESPONDING FEATURES',//)                                     SEL03600
      WRITE(6,1000) (FETVEC(I), I=1,NOFET4)                                     SEL03610
1000 FORMAT( )                                                                  SEL03620
      GO TO 10                                                                  SEL03630
 90   WRITE(6,250) NOFET4,(FETVC2(I),I=1,NOFET2)                                SEL03640
 250  FORMAT(' ERROR IN INPUT CHANNELS'//,1X,I2,30(1X,I2))                      SEL03650
      CALL CMERR                                                                SEL03660
 100 FORMAT(' CORE OVERFLOW IN SUBWAY-',I6,' STORAGE LOCATIONS NECESSARSEL03670
     *Y FOR THIS PROBLEM')                                                      SEL03680
 200 FORMAT(' CORE OVERFLOW IN ARRAY-',I6,' STORAGE LOCATIONS NEEDED FOSEL03690
     *R THIS PROBLEM')                                                          SEL03700
      END                                                                       SEL03710
```

FILE: AVEDIV FORTRAN A

```
      SUBROUTINE AVEDIV(SMSR,COVMTX,S,COVMT2,S2,WRKRY,IWRKSZ,           AVE00010
     *                  IPART,PARTLS,BMAT,IFULL)                        AVE00020
C     INCLUDE COMBK1.LIST                                               AVE00030
C     INCLUDE COMBK7.LIST                                               AVE00040
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          AVE00050
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SURDS2,FLDSV2,VERTX2,    AVE00060
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        AVE00070
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             AVE00080
     *              GRPCHK(61),GROUPS(124)                              AVE00090
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,               COM00010
     *    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)               COM00020
     *     ,NOFET4,VARSZ4,CORMAS,DTAB4,WGHS14,RESTVC(10),DIVSIZ          COM00030
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                    COM00040
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                 COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                               COM00060
CSEND                                                                   AVE00170
      INTEGER VARSZ2,VARSZ4                                             AVE00180
C*                                                                      AVE00190
C*    SUBROUTINE TO COMPUTE WEIGHTED AVERAGE DIVERGENCE,AND PARTIALS    AVE00200
C*    WITH RESPECT TO H.                                                AVE00210
C*                                                                      AVE00220
C*    IF IFULL=1  COMPUTE AVERAGE DIVERGENCE FOR ALL 'NOFET' CHANNELS.  AVE00230
C*                PARTIALS CANNOT BE COMPUTED WHEN IFULL=1.             AVE00240
C*                                                                      AVE00250
C*    IF IPART=0  COMPUTE PARTIALS WITH RESPECT TO BMAT.                AVE00260
C*                                                                      AVE00270
C*                                                                      AVE00280
      DOUBLE PRECISION DET,SMSR,TRACE                                   AVE00290
      DOUBLE PRECISION BMAT,PARTLS                                      AVE00300
      DOUBLE PRECISION COVMT2,S2,WRKRY(1)                               AVE00310
      DIMENSION COVMTX(VARSZ2,NOCLS2), COVMT2(VARSZ4,NOCLS2),           AVE00320
     *          S(VARSZ2,NOCLS2),      S2(VARSZ4,NOCLS2)                AVE00330
      DIMENSION PARTLS(1),BMAT(1)                                       AVE00340
      ICV=1                                                             AVE00350
      ITEST=ICV+VARSZ4                                                  AVE00360
      IF(IFULL.NE.1)GO TO 4                                             AVE00370
      IS1=ICV+VARSZ2                                                    AVE00380
      ITEST=IS1+VARSZ2                                                  AVE00390
    4 CONTINUE                                                          AVE00400
      IF(IPART.LT.0)GO TO 1                                             AVE00410
      IW1=ITEST                                                         AVE00420
      IW2=IW1+NOFET4*NOFET2                                             AVE00430
C*    ZERO PARTLS                                                       AVE00440
      KK=NOFET2*NOFET4                                                  AVE00450
      DO 3 K=1,KK                                                       AVE00460
    3 PARTLS(K)=0.0                                                     AVE00470
      ITEST=IW2 + NOFET4*NOFET2                                         AVE00480
    1 IF(IWRKSZ/2.GE.ITEST)GO TO 2                                      AVE00490
      WRITE(6,200)IWRKSZ                                                AVE00500
      CALL CMERR                                                        AVE00510
    2 CONTINUE                                                          AVE00520
      SMSR=0.0                                                          AVE00530
      DO 30 I=1,NOCLS2                                                  AVE00540
      IF(IFULL.EQ.1)GO TO 15                                            AVE00550
      DO 10 J=1,VARSZ4                                                  AVE00560
   10 WRKRY(J)=COVMT2(J,I)                                              AVE00570
      NF=NOFET4                                                         AVE00580
      GO TO 17                                                          AVE00590
   15 DO 16 J=1,VARSZ2                                                  AVE00600
      WRKRY(J)=COVMTX(J,I)                                              AVE00610
   16 WRKRY(IS1+J-1)=S(J,I)                                             AVE00620
      NF=NOFET2                                                         AVE00630
   17 CALL COLINV(WRKRY(ICV),NF,IERR,3,DET)                             AVE00640
      IF(IERR.EQ.0)GO TO 20                                             AVE00650
      WRITE(6,100) I                                                    AVE00660
      GO TO 30                                                          AVE00670
   20 IF(IFULL.EQ.1)SMSR=SMSR+TRACE(WRKRY(ICV),WRKRY(IS1),NOFET2)       AVE00680
      IF(IFULL.NE.1)SMSR=SMSR+TRACE(WRKRY(ICV),S2(1,I),NOFET4)          AVE00690
C*                                                                      AVE00700
C*    COMPUTE PARTIALS ONLY IF IPART = 0.                               AVE00710
C*                                                                      AVE00720
      IF(IPART.LT.0)GO TO 30                                            AVE00730
C*                                                                      AVE00740
C*    COMPUTE PARTIAL DERIVATIVES WITH RESPECT TO BMAT                  AVE00750
C*                                                                      AVE00760
      CALL MT1(BMAT,COVMTX(1,I),WRKRY(IW1),NOFET4,NOFET2)               AVE00770
      CALL MT2(WRKRY(ICV),WRKRY(IW1),WRKRY(IW2),NOFET4,NOFET2)          AVE00780
      CALL MT2(S2(1,I),WRKRY(IW2),WRKRY(IW1),NOFET4,NOFET2)             AVE00790
      CALL MT1(BMAT,S(1,I),WRKRY(IW2),NOFET4,NOFET2)                    AVE00800
```

FILE: AVEDIV FORTRAN A

```
      DO 25 K=1,KK                                                          AVE00810
      L=K-1                                                                 AVE00820
   25 WRKRY(IW1+L)=WRKRY(IW2+L)-WRKRY(IW1+L)                                 AVE00830
      CALL MT2(WRKRY(ICV),WRKRY(IW1),WRKRY(IW2),NOFET4,NOFET2)              AVE00840
      DO 26 K=1,KK                                                          AVE00850
      L=K-1                                                                 AVE00860
   26 PARTLS(K)=WRKRY(IW2+L)+PARTLS(K)                                      AVE00870
   30 CONTINUE                                                              AVE00880
      SMSR = -(CFAC*SMSR/2. - NF)                                           AVE00890
      IF(IPART.LT.0)RETURN                                                  AVE00900
      DO 40 K=1,KK                                                          AVE00910
   40 PARTLS(K)=-CFAC*PARTLS(K)                                             AVE00920
      RETURN                                                                AVE00930
  100 FORMAT(' REDUCED COVARIANCE MATRIX FOR CLASS',I3,' IS NOT POSITIVE    AVE00940
     * DEFINITE')                                                          AVE00950
  200 FORMAT(' MORE STORAGE NEEDED IN SUB. AVEDIV FOR WORK ARRAY--WORK SAVE00960
     *IZE=',I7)                                                           AVE00970
      END                                                                   AVE00980
```

FILE: BHTCHR FORTRAN A

```
      SUBROUTINE BHTCHR(SMSR,COVMTX,AVEMTX,WEIGHT,DIVTAB,                BHT00010
     *                  COVMT2,AVEMT2,                                  BHT00020
     *                  WRKRY,IWRKSZ,IPART,PARTLS,BMAT,IFULL)           BHT00030
C     INCLUDE COMBK7.LIST                                               BHT00040
      INTEGER VARSZ4,DIVSIZ,VARSZ2                                      BHT00050
C     INCLUDE COMBK1.LIST                                               BHT00060
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          BHT00070
     *              AVAR2,COVAR2,CLS102,SUBNO2,SUBOS2,FLOSV2,VERTX2,    BHT00080
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        BHT00090
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             BHT00100
     *              GRPCHK(61),GROUPS(124)                              BHT00110
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,               BHT00120
     *    INCVEC(30),ICOUNT,SETWGT,EVALHF(100),FETVC4(30)               BHT00130
     *     ,NOFET4,VARSZ4,CORHAS,DTAH4,WGHS14,RESTVC(10),DIVSIZ          BHT00140
     *    ,STATKY,ADRFSD,ADRESP,ADRESF,ADRSH1,ADRSH2                    BHT00150
      INTEGER ADRFSD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                 BHT00160
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                               BHT00170
C$END                                                                   BHT00180
C*                                                                      BHT00190
C*    SUBROUTINE TO COMPUTE THE INTERCLASS BHATTACHRYYA DISTANCE,       BHT00200
C*    THE WEIGHTED AVERAGE DISTANCE, AND THE PARTIALS WITH RESPECT      BHT00210
C*    TO B.                                                             BHT00220
C*                                                                      BHT00230
C*    IF IFULL=1  COMPUTE B. DISTANCE FOR ALL 'NOFET' CHANNELS.         BHT00240
C*                PARTIALS CANNOT BE COMPUTED WHEN IFULL=1.             BHT00250
C*                                                                      BHT00260
C*                                                                      BHT00270
      DOUBLE PRECISION DIVTAB(DIVSIZ),DET1,DET2,DET3                    BHT00280
      DOUBLE PRECISION SMSR                                             BHT00290
      DOUBLE PRECISION BMAT,PARTLS(1)                                   BHT00300
      DOUBLE PRECISION COVMT2,AVEMT2,WRKRY(1),T(30),RNUM                BHT00310
      DIMENSION COVMTX(VARSZ2,NOCLS2),    COVMT2(VARSZ4,NOCLS2),        BHT00320
     *          AVEMTX(NOFET2,NOCLS2),    AVEMT2(NOFET4,NOCLS2),        BHT00330
     *          WEIGHT(DIVSIZ),BMAT(1)                                  BHT00340
      IVSZ=VARSZ4                                                       BHT00350
      NF=NOFET4                                                         BHT00360
      IF(IFULL.EQ.1)IVSZ=VARSZ2                                         BHT00370
      IF(IFULL.EQ.1)NF=NOFET2                                           BHT00380
      ICV1=1                                                            BHT00390
      ICV2=ICV1+IVSZ                                                    BHT00400
      IW1=ICV2 +IVSZ                                                    BHT00410
      IW2=IW1  +IVSZ                                                    BHT00420
      ITEST=IW2+NF                                                      BHT00430
      IF(IPART.LT.0)GO TO 3                                             BHT00440
C*    ZERO PARTIALS                                                     BHT00450
      IQ=NOFET2*NOFET4                                                  BHT00460
      DO 2 IK=1,IQ                                                      BHT00470
    2 PARTLS(IK)=0.0                                                    BHT00480
      IW3=IW2+NOFET2                                                    BHT00490
      IW4=IW3+NOFET2                                                    BHT00500
      IW5=IW4+MAX0(VARSZ2,IQ)                                           BHT00510
      IW6=IW5+IQ                                                        BHT00520
      IW7=IW6+IQ                                                        BHT00530
      ITEST=IW7+IQ                                                      BHT00540
    3 CONTINUE                                                          BHT00550
      IF(IWRKSZ/2.GE.ITEST)GO TO 1                                      BHT00560
      WRITE(6,300)IWRKSZ                                                BHT00570
      CALL CMERR                                                        BHT00580
    1 CONTINUE                                                          BHT00590
      SMSR=0                                                            BHT00600
      NM=0                                                              BHT00610
      IC=NOCLS2-1                                                       BHT00620
      DO 60 I=1,IC                                                      BHT00630
C*    FIND INVERSE AND DETERMINANT FOR CLASS I                          BHT00640
      DO 5 IK=1,IVSZ                                                    BHT00650
      IF(IFULL.EQ.1)WRKRY(IK)=COVMTX(IK,I)                              BHT00660
      IF(IFULL.NE.1)WRKRY(IK)=COVMT2(IK,I)                              BHT00670
    5 CONTINUE                                                          BHT00680
      CALL COLINV(WRKRY(ICV1),NF,IERR,3,DET1)                           BHT00690
      DET1=1./DET1                                                      BHT00700
      IF(IERR.EQ.0)GO TO 6                                              BHT00710
      WRITE(6,100)I                                                     BHT00720
      GO TO 60                                                          BHT00730
    6 IM=I+1                                                            BHT00740
      DO 50 J=IM,NOCLS2                                                 BHT00750
      NM=NM+1                                                           BHT00760
C*    COMPUTE INVERSE AND DETERMINANT FOR CLASS J - AND -               BHT00770
C*    COMPUTE INVERSE AND DETERMINANT FOR SUM OF CLASSES I AND J        BHT00780
      DO 10 IK=1,IVSZ                                                   BHT00790
```

FILE: BHTCHR FORTRAN A

```
      IF(IFULL.EQ.1)GO TO 8                                                  BHT00800
      WRKRY(ICV2+IK-1)=COVMT2(IK,J)                                          BHT00810
      WRKRY(IW1 +IK-1)=COVMT2(IK,J)+COVMT2(IK,I)                             BHT00820
      GO TO 10                                                               BHT00830
    8 WRKRY(ICV2+IK-1)=COVMTX(IK,J)                                          BHT00840
      WRKRY(IW1+IK-1)=COVMTX(IK,J)+COVMTX(IK,I)                              BHT00850
   10 CONTINUE                                                               BHT00860
      CALL COLINV(WRKRY(ICV2),NF,IERR,3,DET2)                                BHT00870
      DET2=1./DET2                                                           BHT00880
      IF(IERR.EQ.0)GO TO 15                                                  BHT00890
      WRITE(6,100)J                                                          BHT00900
      GO TO 50                                                               BHT00910
   15 CALL COLINV(WRKRY(IW1),NF,IERR,3,DET3)                                 BHT00920
      DET3=1./DET3                                                           BHT00930
      IF(IERR.EQ.0)GO TO 16                                                  BHT00940
      WRITE(6,200)I,J                                                        BHT00950
      GO TO 50                                                               BHT00960
   16 IF(IFULL.NE.1)GO TO 18                                                 BHT00970
      DO 17 IK=1,NOFET2                                                      BHT00980
   17 T(IK)=AVEMTX(IK,I)-AVEMTX(IK,J)                                        BHT00990
      GO TO 25                                                               BHT01000
   18 DO 20 IK=1,NOFET4                                                      BHT01010
   20 T(IK)=AVEMT2(IK,I) - AVEMT2(IK,J)                                      BHT01020
C     MULTIPLY T TRANSPOSE TIMES WRKRY1                                      BHT01030
   25 CONTINUE                                                               BHT01040
      CALL MT4(WRKRY(Iw1),T,WRKRY(Iw2),NF,NF,1,1)                            BHT01050
      RNUM=0.0                                                               BHT01060
      DO 26 IK=1,NF                                                          BHT01070
   26 RNUM=RNUM+ WRKRY(IW2+IK-1)*T(IK)                                       BHT01080
      DIVTAB(NM)=DEXP(-.25*RNUM -.5*DLOG(DET3/(2**NF                         BHT01090
     *              * DSQRT(DET1*DET2))))                                    BHT01100
      SMSR=SMSR+WEIGHT(NM)*DIVTAB(NM)                                        BHT01110
      IF(IPART.LT.0)GO TO 50                                                 BHT01120
C*                                                                           BHT01130
C*    COMPUTE PARTIALS                                                       BHT01140
C*                                                                           BHT01150
      CALL MT4(WRKRY(Iw2),BMAT,WRKRY(Iw3),1,NOFET4,NOFET2,0)                 BHT01160
      DO 30 IK=1,VARSZ2                                                      BHT01170
   30 WRKRY(IK+IW4-1)=COVMTX(IK,J)+COVMTX(IK,I)                              BHT01180
      CALL MT4(WRKRY(IW4),WRKRY(IW3),WRKRY(IW2),NOFET2,NOFET2,1,1)           BHT01190
      DO 35 IK=1,NOFET2                                                      BHT01200
   35 WRKRY(IW3+IK-1)=AVEMTX(IK,I)-AVEMTX(IK,J) - WRKRY(IW2+IK-1)            BHT01210
      CALL MT4(T,WRKRY(IW3),WRKRY(IW5),NOFET4,1,NOFET2,0)                    BHT01220
      CALL MT4(WRKRY(Iw1),WRKRY(IW5),WRKRY(IW6),NOFET4,NOFET4,NOFET2,1)      BHT01230
      DO 40 IK=1,IQ                                                          BHT01240
   40 WRKRY(IW6+IK-1)=WRKRY(IW6+IK-1)/2                                      BHT01250
      M=I                                                                    BHT01260
      IC=ICV1                                                                BHT01270
   41 DO 45 IK=1,VARSZ4                                                      BHT01280
      L=IK-1                                                                 BHT01290
   45 WRKRY(IW4+L)=WRKRY(IW1+L)-WRKRY(IC+L)/2                                BHT01300
      CALL MT4(WRKRY(Iw4),BMAT,WRKRY(IW7),NOFET4,NOFET4,NOFET2,1)            BHT01310
      CALL MT1(WRKRY(IW7),COVMTX(1,M),WRKRY(IW5),NOFET4,NOFET2)              BHT01320
      DO 43 IK=1,IQ                                                          BHT01330
      L=IK-1                                                                 BHT01340
   43 WRKRY(IW6+L)=WRKRY(IW6+L)+WRKRY(IW5+L)                                 BHT01350
      IF(M.EQ.J)GO TO 46                                                     BHT01360
      M=J                                                                    BHT01370
      IC=ICV2                                                                BHT01380
      GO TO 41                                                               BHT01390
   46 DO 47 IK=1,IQ                                                          BHT01400
      L=IK-1                                                                 BHT01410
      PARTLS(IK)=PARTLS(IK)-WEIGHT(NM)*DIVTAB(NM)*WRKRY(IW6+L)/NOCLS2        BHT01420
   47 CONTINUE                                                               BHT01430
   50 CONTINUE                                                               BHT01440
   60 CONTINUE                                                               BHT01450
      SMSR=SMSR/NOCLS2                                                       BHT01460
      RETURN                                                                 BHT01470
  100 FORMAT(' COVAR FOR CLASS',I3,' IS NOT POSITIVE DEFINITE')              BHT01480
  200 FORMAT(' COVAR FOR SUM OF CLASSES',2I4,' IS NOT POSITIVE DEF.')        BHT01490
  300 FORMAT(' NOT ENOUGH WORK AREA AVAILABLE IN BHTCHR -- IWRKSZ=',I5)      BHT01500
      END                                                                    BHT01510
```

FILE: BSTCHK FORTRAN A

```
      SUBROUTINE BSTCHK(NOBEST)                                          BST00010
C                                                                        BST00020
      IMPLICIT INTEGER (A-H,O-Z)                                         BST00030
CI----------------------------------------------------------------------IBST00040
CI----------------------------------------------------------------------IBST00050
CCI                                                                      IBST00060
CCI   CALL..    CALL BSTCHK(NOBEST)                                      IBST00070
CCI                                                                      IBST00080
CCI   ARGS..    NOBEST - NO OF FEATURES-TUPLES TO ANALYZE                 IBST00090
CCI                                                                      IBST00100
CCI   REQUIRES. COMMON   /INFORM/                                        IBST00110
CCI                                                                      IBST00120
CCI   PURPOSE.. CHECKS VALIDITY OF REQUESTED FEATURE-TUPLES              IBST00130
CCI                                                                      IBST00140
CCI   RETURNS.. CORRECT FEATURE-TUPLE QUE                                IBST00150
CCI                                                                      IBST00160
CCI----------------------------------------------------------------------IBST00170
CCI                                                                      IBST00180
CCI   CALL..    CALL SHWCHK(COMBUF,CPTR)                                 IBST00190
CCI                                                                      IBST00200
CCI   ARGS..    COMBUF - SHOW REQUEST QUE                                IBST00210
CCI                                                                      IBST00220
CCI                                                                      IBST00230
CCI   PURPOSE.. CHECKS VALIDITY OF SHOW REQUESTS                         IBST00240
CCI                                                                      IBST00250
CCI   RETURNS.. CORRECT SHOW REQUEST QUE                                 IBST00260
CCI                                                                      IBST00270
CCI----------------------------------------------------------------------IBST00280
CI----------------------------------------------------------------------IBST00290
C                                                                        BST00300
      DIMENSION COMBUF(1)                                                BST00310
C                                                                        BST00320
CCCCC                                                                    BST00330
      INCLUDE COMBK1.LIST                                                BST00340
CCCCC                                                                    BST00350
CCCCC ----------------------------------------------------------------- BST00360
C     ----------------------------------------------------------------- BST00370
      DIMENSION INVERT(30)                                               BST00380
C     INCLUDE COMBK7.LIST                                                BST00390
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           BST00400
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     BST00410
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         BST00420
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              BST00430
     *              GRPCHK(61),GROUPS(124)                               BST00440
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CHIKEY,INCFET,                COM00010
     *    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)                COM00020
     *     ,NOFET4,VARS74,COMBAS,DTAR4,WGHS14,BESTVC(10),DIVSIZ          COM00030
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                     COM00040
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                  COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                COM00060
CSEND                                                                    BST00520
      II = 0                                                             BST00530
      DO 30 I=1,NOBEST                                                   BST00540
      J = BESTVC(I)                                                      BST00550
      IF(J.GT.NOFET2)GO TO 10                                            BST00560
      II = II+1                                                          BST00570
      BESTVC(II) = J                                                     BST00580
      GO TO 30                                                           BST00590
   10 WRITE (6,20) I                                                     BST00600
   20 FORMAT('0A0760 **BEST***,I3,' IS GREATER THAN OR EQUAL TO NO. OF F BST00610
     *EATURES IN GIVEN DATA...IGNORED')                                  BST00620
   30 CONTINUE                                                           BST00630
      NOBEST = II                                                        BST00640
C                                                                        BST00650
      RETURN                                                             BST00660
C                                                                        BST00670
CCCCC                                                                    BST00680
CCCCC ----------------------------------------------------------------- BST00690
C     ----------------------------------------------------------------- BST00700
C                                                                        BST00710
C                                                                        BST00720
      ENTRY EVLCHK(COMBUF,CPTR)                                          BST00730
C                                                                        BST00740
      DO 32 I=1,NOFET2                                                   BST00750
   32 INVERT(I)=0                                                        BST00760
      DO 33 I=1,NOFET2                                                   BST00770
      K = FETVC2(I)                                                      BST00780
   33 INVERT(K)=I                                                        BST00790
      K = 0                                                              BST00800
```

FILE: BSTCHK FORTRAN A

```
      I = 1                                                                  BST00810
   40 IF (I.GE.CPTR) GO TO 100                                               BST00820
      N = COMBUF(I)                                                          BST00830
      J = I+N                                                                BST00840
      I = I+1                                                                BST00850
      LAST = 0                                                               BST00860
      DO 50 L=I,J                                                            BST00870
      LL = COMBUF(L)                                                         BST00880
      IF (LL.LE.LAST.OR.INVERT(LL).EQ.0.OR.LL.GT.FETVC2(NOFET2)) GO TO 80    BST00890
     10                                                                      BST00900
   50 LAST = LL                                                              BST00910
      K = K+1                                                                BST00920
      COMBUF(K) = N                                                          BST00930
      DO 60 L=I,J                                                            BST00940
      K = K+1                                                                BST00950
   60 COMBUF(K) = COMBUF(L)                                                  BST00960
   70 I = J+1                                                                BST00970
      GO TO 40                                                               BST00980
C                                                                            BST00990
   80 WRITE (6,90)(COMBUF(LL),LL=I,J)                                        BST01000
   90 FORMAT(' INVALID EVALUATE REQUEST....',18I4)                           BST01010
      GO TO 70                                                               BST01020
C                                                                            BST01030
  100 CPTR = K                                                               BST01040
C                                                                            BST01050
      RETURN                                                                 BST01060
       END                                                                   BST01070
```

FILE: COLINV FORTRAN A

```
      SUBROUTINE COLINV(S,N,IERR,IND,DET)                                   COL00010
      DOUBLE PRECISION S,SUM,DET                                            COL00020
C                                                                           COL00030
C     PURPOSE                                                               COL00040
C        INVERT A GIVEN SYMMETRIC POSITIVE DEFINITE MATRIX (S) BY           COL00050
C        COMPUTING A TRIANGULAR FACTORIZATION (R), INVERTING R TO           COL00060
C        OBTAIN A, AND THEN FINDING THE INVERSE OF S.                       COL00070
C        S=R*(TRANSPOSE OF R), A=(INVERSE OF R),                            COL00080
C        (INVERSE OF S)=(TRANSPOSE OF A)*A .                                COL00090
C                                                                           COL00100
C     ARGUMENTS                                                             COL00110
C        S    -LOWER TRIANGULAR PART OF THE GIVEN SYMMETRIC MATRIX          COL00120
C              STORED ROWWISE IN N*(N+1)/2 SUCCESSIVE STORAGE LOC-TIONS.    COL00130
C              ON RETURN S CONTAINS THE LOWER TRIANGULAR MATRIX R,          COL00140
C              THE LOWER TRIANGULAR MATRIX A, OR THE LOWER TRIANGULAR       COL00150
C              PART OF THE INVERSE OF S, DEPENDING ON VALUE                 COL00160
C              ASSIGNED TO IND.                                             COL00170
C        N    -THE NUMBER OF ROWS OR COLUMNS IN GIVEN MATRIX.               COL00180
C        IERR -RESULTING ERROR PARAMETER CODED AS FOLLOWS                   COL00190
C              IERR=0  -NO ERROR                                            COL00200
C              IERR=-1 -MATRIX S IS NOT POSITIVE DEFINITE                   COL00210
C        IND  -PARAMETER INDICATING WHICH MATRIX IS RETURNED --             COL00220
C              IND=1   -MATRIX R                                            COL00230
C              IND=2   -MATRIX A                                            COL00240
C              IND=3   -MATRIX (INVERSE OF S)                               COL00250
C        DET  -DETERMINANT OF RETURNED MATRIX.                              COL00260
C                                                                           COL00270
C     REFERENCE                                                             COL00280
C        "INVERSION OF SYMMETRIC POSITIVE DEFINITE MATRICES," BY            COL00290
C        J.K. BRYAN AND D.L. TEBBE, E.E. DEPT., UNIV. OF MISSOURI, 1971.    COL00300
C                                                                           COL00310
      DIMENSION S(1)                                                        COL00320
      IERR = 0                                                              COL00330
      LS = 0                                                                COL00340
      N2 = 1                                                                COL00350
      DET = 1.0                                                             COL00360
      DO 70 I=1,N                                                           COL00370
      N2 = N2 + I - 1                                                       COL00380
      M = I*(I+1)/2                                                         COL00390
      L = N2                                                                COL00400
      DO 40 J=I,N                                                           COL00410
      L = L + J - 1                                                         COL00420
      IF( I .EQ. 1 ) GO TO 20                                               COL00430
      M1 = J*(J-1)/2 + 1                                                    COL00440
      L1 = L - 1                                                            COL00450
      N1 = N2                                                               COL00460
      DO 10 K=M1,L1                                                         COL00470
      S(L) = S(L) - S(N1)*S(K)                                              COL00480
   10 N1 = N1 + 1                                                           COL00490
   20 IF( J .EQ. I ) GO TO 30                                               COL00500
      S(L) = S(L)/S(M)                                                      COL00510
      GO TO 40                                                              COL00520
   30 IF( S(L) .LE. 0.0 ) GO TO 100                                         COL00530
      S(L) = DSQRT(S(L))                                                    COL00540
      DET = DET*S(L)                                                        COL00550
   40 CONTINUE                                                              COL00560
      IF( IND.EQ.1 ) GO TO 70                                               COL00570
      S(M) = 1./S(M)                                                        COL00580
      IF( I .EQ. 1 ) GO TO 70                                               COL00590
      I1 = I - 1                                                            COL00600
      DO 60 J=1,I1                                                          COL00610
      K1 = I - J                                                            COL00620
      L1 = LS                                                               COL00630
      LS = LS + 1                                                           COL00640
      M1 = J*(J-1)/2 + 1                                                    COL00650
      J1 = J - 1                                                            COL00660
      SUM = 0.0                                                             COL00670
      DO 50 K=1,K1                                                          COL00680
      L1 = L1 + 1                                                           COL00690
      J1 = J1 + 1                                                           COL00700
      M1 = M1 + J1 - 1                                                      COL00710
   50 SUM = SUM + S(L1)*S(M1)                                               COL00720
   60 S(LS) = -S(M)*SUM                                                     COL00730
   70 LS = LS + 1                                                           COL00740
      IF( IND.EQ.2 .OR. IND.EQ.3 ) DET=1./DET                               COL00750
      IF( IND.EQ.1 .OR. IND.EQ.2 ) RETURN                                   COL00760
      DO 90 I=1,N                                                           COL00770
      M = I*(I-1)/2 + 1                                                     COL00780
      L = M                                                                 COL00790
```


ORIGINAL PAGE IS
OF POOR QUALITY

FILE: COLINV FORTRAN A

```
      DO 90 J=I+1,N                                                  COL00800
      L2 = L                                                         COL00810
      L = L + J - I                                                  COL00820
      SUB = 0.0                                                      COL00830
      DO 80 K=I+1,N                                                  COL00840
      L2 = L2 + K - I                                                COL00850
      M1 = L2 + J - I                                                COL00860
   80 SUB = SUB + S(L2)*S(M1)                                        COL00870
   90 S(L) = SUM                                                     COL00880
      DET = DET*DET                                                  COL00890
      RETURN                                                         COL00900
  100 IERR = -1                                                      COL00910
      RETURN                                                         COL00920
      END                                                            COL00930
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: CONVRT FORTRAN A

```
      SUBROUTINE CONVRT(NOSPAC,IFLG)                                    CON00010
C                                                                       CON00020
C  THE PURPOSE OF THIS ROUTINE IS TO CONVERT EBCDIC CHARS TO COMPUTATIONCON00030
C  OR TO CONVERT COMPUTATIONAL CHARS TO EBCDIC CHARS                    CON00040
C                                                                       CON00050
      LOGICAL*1 LTEST(4),LSPAC(4)                                       CON00060
C                                                                       CON00070
      EQUIVALENCE (ITEST,LTEST(4)), (ICHAR,LSPAC(4))                    CON00080
      DATA BCD0/ZF0/,BCD9/ZF9/                                          CON00090
C                                                                       CON00100
      ITEST = 0                                                         CON00110
      ICHAR = NOSPAC                                                    CON00120
      LTEST(4) = LSPAC(4)                                               CON00130
C                                                                       CON00140
C  CK FOR FLAG NOT 0, IMPLYING CHAR IS EBCDIC                           CON00150
      IF (IFLG .NE. 0) GO TO 1040                                       CON00160
C                                                                       CON00170
C  CK FOR COMPUTATIONAL NUMBER & CONVERT TO EBCDIC                      CON00180
      IF (ITEST .LT. 0  .OR.  ITEST .GT. 9) GO TO 9999                  CON00190
      ITEST = ITEST + BCD0                                              CON00200
      LSPAC(4) = LTEST(4)                                               CON00210
      NOSPAC = ICHAR                                                    CON00220
      RETURN                                                            CON00230
C                                                                       CON00240
C  CK FOR EBCDIC NUMBER TO CONVERT TO COMPUTATIONAL NUMBER              CON00250
 1040 IF (ITEST .LT. BCD0  .OR.  ITEST .GT. BCD9) GO TO 9999            CON00260
      ITEST = ITEST -BCD0                                               CON00270
      LSPAC(4) = LTEST(4)                                               CON00280
      NOSPAC = ICHAR                                                    CON00290
      RETURN                                                            CON00300
C  ERROR                                                                CON00310
 9999 WRITE (6,9998)                                                    CON00320
 9998 FORMAT (1X,'ERROR - NON-NUMERIC CHARACTER INPUT')                 CON00330
      RETURN                                                            CON00340
      END                                                               CON00350
```

FILE: CUBIC FORTRAN A

```
      SUBROUTINE CUBIC(XX,YY,XMIN2)                                          CUB00010
      DOUBLE PRECISION                                                       CUB00020
     1XX(4)          ,YY(4)          ,ARRAY(4,4)   ,COEFF(4)     ,CONST(4)  , CUB00030
     2TEMP           ,ADD            ,F            ,XMIN2                   CUB00040
      DO 14 I = 1,4                                                          CUB00050
      ARRAY(I,1) = 1.000                                                     CUB00060
      ARRAY(I,2) = XX(I)                                                     CUB00070
      ARRAY(I,3) = XX(I)**2                                                  CUB00080
      ARRAY(I,4) = XX(I)**3                                                  CUB00090
   14 CONST(I) = YY(I)                                                       CUB00100
   15 DO 6 K2 = 1,3                                                          CUB00110
      KK = K2 + 1                                                            CUB00120
      L = K2                                                                 CUB00130
      DO 7 I = KK,4                                                          CUB00140
      IF(DABS(ARRAY(I,K2)) - DABS(ARRAY(L,K2)))7,7,20                        CUB00150
   20 CONTINUE                                                               CUB00160
      L = I                                                                  CUB00170
    7 CONTINUE                                                               CUB00180
      IF(L - K2)8,8,21                                                       CUB00190
   21 CONTINUE                                                               CUB00200
      DO 9 J = K2,4                                                          CUB00210
      TEMP = ARRAY(K2,J)                                                     CUB00220
      ARRAY(K2,J) = ARRAY(L,J)                                               CUB00230
    9 ARRAY(L,J) = TEMP                                                      CUB00240
      TEMP = CONST(K2)                                                       CUB00250
      CONST(K2) = CONST(L)                                                   CUB00260
      CONST(L) = TEMP                                                        CUB00270
    8 DO 6 I = KK,4                                                          CUB00280
      F = ARRAY(I,K2)/ARRAY(K2,K2)                                           CUB00290
      ARRAY(I,K2) = 0.000                                                    CUB00300
      DO 10 J = KK,4                                                         CUB00310
   10 ARRAY(I,J) = ARRAY(I,J) - F*ARRAY(K2,J)                                CUB00320
    6 CONST(I) = CONST(I) - F*CONST(K2)                                      CUB00330
      COEFF(4) = CONST(4)/ARRAY(4,4)                                         CUB00340
      I = 3                                                                  CUB00350
   11 II = I + 1                                                             CUB00360
      ADD = 0.000                                                            CUB00370
      DO 12 J = II,4                                                         CUB00380
   12 ADD = ADD + ARRAY(I,J)*COEFF(J)                                        CUB00390
      COEFF(I) = (CONST(I) - ADD)/ARRAY(I,I)                                 CUB00400
      I = I - 1                                                              CUB00410
      IF(I)22,22,11                                                          CUB00420
   22 CONTINUE                                                               CUB00430
      F = COEFF(3)**2 - 3.0D0*COEFF(2)*COEFF(4)                              CUB00440
      IF(F)16,23,23                                                          CUB00450
   23 CONTINUE                                                               CUB00460
      IF(DABS(COEFF(4)) - 1.0D-15)17,17,24                                   CUB00470
   24 CONTINUE                                                               CUB00480
      XMIN2 = (-COEFF(3) + DSQRT(F))/(3.0D0*COEFF(4))                        CUB00490
      RETURN                                                                 CUB00500
   17 IF(DABS(COEFF(3)) - 1.0D-15)16,16,25                                   CUB00510
   25 CONTINUE                                                               CUB00520
      XMIN2 = -COEFF(2)/(COEFF(3) + DSQRT(F))                                CUB00530
      RETURN                                                                 CUB00540
   16 XMIN2 = 0.000                                                          CUB00550
      RETURN                                                                 CUB00560
      END                                                                    CUB00570
```

FILE: DAVDN1 FORTRAN A

```
      SUBROUTINE DAVDN1(FX,P,H)                                          DAV00010
C*                                                                       DAV00020
C*    DAVDN1 INITIALIZES THE H AND P ARRAYS USED IN DAVIDON.PROCEDURE    DAV00030
C*                                                                       DAV00040
C*     H***** MATRIX IS AN N*N MATRIX APPROXIMATING THE INVERSE          DAV00050
C*            MATRIX OF PARTIAL DERIVATIVES. IT IS INITIALIZED           DAV00060
C*            HERE AS THE IDENTITY MATRIX AND STORED ON SCRATCH          DAV00070
C*            FILE. THE H MATRIX IS UPDATED AFTER EACH CYCLE THRU        DAV00080
C*            THE SEARCH PROCEDURE OF SUBROUTINE DAVDN2.                 DAV00090
C*                                                                       DAV00100
C*     P***** VECTOR CORRESPONDING TO SEARCH DIRECTION                   DAV00110
C*            (H-MATRIX TIMES THE GRADIENT)                              DAV00120
C*     DFDK** MINUS THE NORM OF THE GRADIENT WITH RESPECT TO THE         DAV00130
C*            H-MATRIX SQUARED.                                          DAV00140
C*     FX**** GRADIENT OF FUNCTION TO BE MINIMIZED                       DAV00150
C*                                                                       DAV00160
C*     * H * AND * FX* SAVED ON SCRATCH FILE TO BE USED IN DAVDN3        DAV00170
C*                                                                       DAV00180
C     INCLUDE COMEF7.LIST                                                DAV00190
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                COM00010
     *    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)                COM00020
     *     ,NOFET4,VARSZ4,CORBAS,UTAB4,WGHS14,RESTVC(10),DIVSIZ          COM00030
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                     COM00040
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                  COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                COM00060
C*END                                                                    DAV00210
      DOUBLE PRECISION FX(1),P(1),H(1),DFDK                              DAV00220
      DOUBLE PRECISION CAYMIN,FII,CCAY                                   DAV00230
      INTEGER AD                                                         DAV00240
      COMMON/DVNBLK/DFDK,CAYMIN,FII,CCAY,I1D,I1DMEN,ITT,ICNT,N           DAV00250
      N2=N**2                                                            DAV00260
      DFDK=0.0                                                           DAV00270
      AD=ADRSH1                                                          DAV00280
      DO 30 I=1,N                                                        DAV00290
      P(I)=-FX(I)                                                        DAV00300
      DFDK=DFDK+P(I)*FX(I)                                               DAV00310
      DO 20 J=1,N                                                        DAV00320
      H(J)=0.0                                                           DAV00330
   20 IF(I.EQ.J)H(J)=1.0                                                 DAV00340
      CALL RWRITE(AD,H,N2,ISTAT)                                         DAV00350
      AD=AD+N2                                                           DAV00360
   21 IF(ISTAT.EQ.1)GO TO 21                                             DAV00370
      IF(ISTAT.EQ.0)GO TO 30                                             DAV00380
      WRITE(6,100)ISTAT                                                  DAV00390
   30 CONTINUE                                                           DAV00400
C*                                                                       DAV00410
C*    SAVE FX ON SCRATCH FILE                                            DAV00420
C*                                                                       DAV00430
      CALL RWRITE(ADRESF,FX,N2,ISTAT)                                    DAV00440
   40 IF(ISTAT.EQ.1)GO TO 40                                             DAV00450
      IF(ISTAT.EQ.0)RETURN                                               DAV00460
      WRITE(6,100)ISTAT                                                  DAV00470
      CALL CMERR                                                         DAV00480
  100 FORMAT(/' ERROR ON DRUM FILE - SUBROUTINE DAVDN1---ISTAT=',I3)     DAV00490
      END                                                                DAV00500
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: DAVDN2 FORTRAN A

```
      SUBROUTINE DAVDN2(XBAR,XBARS,P,WRK,IWKSZ,*)                          DAV00010
      DOUBLE PRECISION XBAR(N),XBARS(N),P(N),WRK(IWKSZ)                    DAV00020
      DOUBLE PRECISION FHAT(4),CAY(4),X(10),Y(10),                         DAV00030
     * CAYMIN,CCAY,DFDK,EPSCF,EPSGS,EX,F,FII,                              DAV00040
     * G1,G2,TEMP,XMIN1,XMIN2,XNEW                                         DAV00050
      COMMON/DVNBLK/DFDK,CAYMIN,FII,CCAY,IID,IIDMEN,ITT,ICNT,N             DAV00060
C     DFDK****** MINUS  THE  NORM  OF  THE GRADIENT  WITH  RESPECT  TO     DAV00070
C                 THE  H-MATRIX  SQUARED                                   DAV00080
C     FII******* INITIAL  VALUE  OF  FUNCTION  TO  BE  MINIMIZED           DAV00090
C*    IPART***** SET TO -1 SO PARTIALS WILL NOT BE COMPUTED                DAV00100
C*                DURING SEARCH                                            DAV00110
C*    N********* NUMBER OF VARIABLES                                       DAV00120
C     ITT******* COUNTER  DENOTING  THE  NUMBER  OF  EVALUATIONS           DAV00130
C                 OF  THE  FUNCTION  TO  BE  MINIMIZED                     DAV00140
C     P********* VECTOR  CORRESPONDING  TO  SEARCH  DIRECTION              DAV00150
C                 (G=MATRIX  TIMES  THE  GRADIENT )                        DAV00160
C     XBAR****** NOMINAL  VECTOR  OF  CONTROL  PARAMETERS  AT              DAV00170
C                 THE  START  OF  EACH  CYCLE                              DAV00180
C     XBARS***** PERTURBED  VECTOR  OF  CONTROL  PARAMETERS  IN            DAV00190
C                 ACCORDANCE  WITH  THE  DAVIDON  ALGORITHM  AND           DAV00200
C                 SEARCH  PROCEDURES                                       DAV00210
      G1 = 1.500 - DSQRT(1.2500)                                           DAV00220
      G2 = 1.000 - G1                                                      DAV00230
C                                                                          DAV00240
C  THIS STARTS ONE DIMENSIONAL SEARCH ON K                                 DAV00250
C                                                                          DAV00260
C     INITIALIZE 1D SEARCH VARIABLES                                       DAV00270
CCC   EPSCF IS RELATIVE EPSILON FOR CUBIC FIT.                             DAV00280
CCC   EPSGS IS RELATIVE EPSILON FOR GOLDEN SECTION.                        DAV00290
CCC   KUBIC = 1 CUBIC FIT METHOD                                           DAV00300
CCC   KUBIC = 0 GOLDEN SECTION METHOD                                      DAV00310
CCC   KUBIC = -1 GOLDEN SECTION AND CUBIC FIT METHODS COMBINED             DAV00320
C                                                                          DAV00330
 9038 IPART=-1                                                             DAV00340
      Y(10) = FII                                                          DAV00350
      JSW = -1                                                             DAV00360
      KUBIC=1                                                              DAV00370
      EPSCF=1.D-3                                                          DAV00380
      CAY(1)=0.D+0                                                         DAV00390
      FHAT(1) = FII                                                        DAV00400
      CAY(2)=CCAY                                                          DAV00410
      DO 9039 I=1,N                                                        DAV00420
 9039 XBARS(I)=XBAR(I)+CAY(2)*P(I)                                         DAV00430
      CALL FINT2(FHAT(2),IPART,XBARS,WRK,IWKSZ,&9999)                      DAV00440
      Y(8) = FHAT(2)                                                       DAV00450
      X(8)=CAY(2)                                                          DAV00460
      ITT=ITT+1                                                            DAV00470
      Y(1)=2.D+0*(FHAT(2)-FHAT(1)-DFDK*CAY(2))/(CAY(2)*CAY(2))             DAV00480
      IF(Y(1) - 1.0D-15)9040,1,1                                           DAV00490
    1 CONTINUE                                                             DAV00500
      XMIN1=-DFDK/Y(1)                                                     DAV00510
      CAY(3)=XMIN1                                                         DAV00520
 9042 DO 9041 I=1,N                                                        DAV00530
 9041 XBARS(I)=XBAR(I)+CAY(3)*P(I)                                         DAV00540
      CALL FINT2(FHAT(3),IPART,XBARS,WRK,IWKSZ,&9999)                      DAV00550
      ITT=ITT+1                                                            DAV00560
      GO TO 9043                                                           DAV00570
 9040 CAY(3)=CCAY*1.20+0                                                   DAV00580
      XMIN1=CAY(3)                                                         DAV00590
      GO TO 9042                                                           DAV00600
 9043 Y(1)=FHAT(2)-FHAT(1)-DFDK*CAY(2)                                     DAV00610
      Y(2)=FHAT(3)-FHAT(1)-DFDK*CAY(3)                                     DAV00620
      X(1)=CAY(2)*CAY(2)/2.D+0                                             DAV00630
      X(2)=X(1)*CAY(2)/3.D+0                                               DAV00640
      X(3)=CAY(3)*CAY(3)/2.D+0                                             DAV00650
      X(4)=X(3)*CAY(3)/3.D+0                                               DAV00660
      Y(3)=X(3)*X(2)-X(1)*X(4)                                             DAV00670
      Y(6)=Y(3)                                                            DAV00680
      IF(Y(3).LT.0.D+0)Y(6)=-Y(3)                                          DAV00690
      XNEW = Y(5)                                                          DAV00700
      Y(4)=(X(2)*Y(2)-X(4)*Y(1))/Y(3)                                      DAV00710
      Y(5)=(X(3)*Y(1)-X(1)*Y(2))/Y(3)                                      DAV00720
      Y(6)=Y(4)*Y(4)-2.D+0*DFDK*Y(5)                                       DAV00730
      IF(Y(6))9044,2,2                                                     DAV00740
    2 CONTINUE                                                             DAV00750
      XMIN2 = -2.000*DFDK/(Y(4) + DSQRT(Y(6)))                             DAV00760
      IF(XMIN2 - 1.0D-15)9044,3,3                                          DAV00770
    3 CONTINUE                                                             DAV00780
      IF(DABS(XMIN2-XMIN1) - EPSCF*XMIN1)4,4,9045                          DAV00790
```

FILE: DAVDN2 FORTRAN A

```
    4 CONTINUE                                                          DAV00800
      CAYMIN=XMIN2                                                      DAV00810
      GO TO 9999                                                        DAV00820
 9045 CAY(4)=XMIN2                                                      DAV00830
      GO TO 9048                                                        DAV00840
 9044 CAY(4)=CAY(3)*1.5D+0                                              DAV00850
 9048 DO 9047 I=1,N                                                     DAV00860
 9047 XBARS(I)=XBAR(I)+CAY(4)*P(I)                                      DAV00870
      CALL FINT2(FHAT(4),IPART,XBARS,WRK,IWKSZ,&9999)                   DAV00880
      ITT=ITT+1                                                         DAV00890
      DO 9050 J = 1,3                                                   DAV00900
      DO 9050 I = 1,3                                                   DAV00910
      IF(CAY(I) - CAY(I+1))9050,5,5                                     DAV00920
    5 CONTINUE                                                          DAV00930
      TEMP = CAY(I)                                                     DAV00940
      CAY(I) = CAY(I+1)                                                 DAV00950
      CAY(I+1) = TEMP                                                   DAV00960
      TEMP = FHAT(I)                                                    DAV00970
      FHAT(I) = FHAT(I+1)                                               DAV00980
      FHAT(I+1) = TEMP                                                  DAV00990
 9050 CONTINUE                                                          DAV01000
      GO TO 542                                                         DAV01010
 9066 FII = Y(10)                                                       DAV01020
      CAY(4)=X(9)                                                       DAV01030
      FHAT(4)=Y(9)                                                      DAV01040
      I10MEN = -1                                                       DAV01050
  801 IPART=-1                                                          DAV01060
       KURIC = 1                                                        DAV01070
      IF(I10)888,888,6                                                  DAV01080
    6 CONTINUE                                                          DAV01090
      KURIC = 0                                                         DAV01100
  888 XMIN1 = 0.000                                                     DAV01110
      CAY(1)=0.D+0                                                      DAV01120
      FHAT(1) = FII                                                     DAV01130
       IF(DFDK .GT. -1.D-12)GO TO 340                                   DAV01140
       IF(DFDK .GT. -1.D-9) GO TO 330                                   DAV01150
       IF(DFDK .GT. -1.D-5) GO TO 320                                   DAV01160
      IF(DFDK.GT.-1.D+1)GO TO 350                                       DAV01170
      EPSGS=1.D-2                                                       DAV01180
      EPSCF=1.D-2                                                       DAV01190
      GO TO 410                                                         DAV01200
  350 EPSCF=1.D-3                                                       DAV01210
      EPSGS=1.D-3                                                       DAV01220
       GO TO 410                                                        DAV01230
  320 EPSCF=1.D-5                                                       DAV01240
      EPSGS=1.D-5                                                       DAV01250
       GO TO 410                                                        DAV01260
  330  EPSCF = 1.D-2                                                    DAV01270
        EPSGS = 1.D-2                                                   DAV01280
       GO TO 410                                                        DAV01290
  340  EPSCF = 1.D-1                                                    DAV01300
        EPSGS = 1.D-1                                                   DAV01310
C                                                                       DAV01320
C  PART 1                                                               DAV01330
C     ESTABLISH GOLDEN SECTION IN WHICH FUNCTION IS UNIMODAL            DAV01340
C                                                                       DAV01350
  410 IF(I1D)7,7,9067                                                   DAV01360
    7 CONTINUE                                                          DAV01370
      CAY(4)=CCAY                                                       DAV01380
  420 DO 430 I=1,N                                                      DAV01390
  430 XBARS(I)=XBAR(I)+CAY(4)*P(I)                                      DAV01400
      CALL FINT2(FHAT(4),IPART,XBARS,WRK,IWKSZ,&9999)                   DAV01410
       ITT = ITT + 1                                                    DAV01420
 9067 CONTINUE                                                          DAV01430
      IF(FHAT(4) - FHAT(1))8,490,490                                    DAV01440
    8 CONTINUE                                                          DAV01450
  440 CAY(2)=CAY(4)                                                     DAV01460
      FHAT(2)=FHAT(4)                                                   DAV01470
      CAY(4) = CAY(2)/G1                                                DAV01480
      DO 450 I=1,N                                                      DAV01490
  450 XBARS(I)=XBAR(I)+CAY(4)*P(I)                                      DAV01500
      CALL FINT2(FHAT(4),IPART,XBARS,WRK,IWKSZ,&9999)                   DAV01510
       ITT = ITT + 1                                                    DAV01520
      IF(FHAT(4) - FHAT(2))440,9,9                                      DAV01530
    9 CONTINUE                                                          DAV01540
  460 CAY(3) = G2*(CAY(4) - CAY(1)) + CAY(1)                            DAV01550
      DO 470 I=1,N                                                      DAV01560
  470 XBARS(I)=XBAR(I)+CAY(3)*P(I)                                      DAV01570
      CALL FINT2(FHAT(3),IPART,XBARS,WRK,IWKSZ,&9999)                   DAV01580
```

FILE: DAVDN2 FORTRAN A

```
      ITT = ITT + 1                                                     DAV01590
      IF(FHAT(4) - FHAT(3))10,10,510                                    DAV01600
   10 CONTINUE                                                          DAV01610
  475 FHAT(4)=FHAT(3)                                                   DAV01620
      CAY(4)=CAY(3)                                                     DAV01630
      CAY(3)=CAY(2)                                                     DAV01640
      FHAT(3)=FHAT(2)                                                   DAV01650
      CAY(2)=CAY(4) - CAY(3) + CAY(1)                                   DAV01660
      DO 480 I=1,N                                                      DAV01670
  480 XBARS(I)=XBAR(I)+CAY(2)*P(I)                                      DAV01680
      CALL FINT2(FHAT(2),IPART,XBARS,WRK,IWKSZ,&9999)                   DAV01690
      ITT = ITT + 1                                                     DAV01700
      IF(FHAT(2) - FHAT(1))510,11,11                                    DAV01710
   11 CONTINUE                                                          DAV01720
  485  FHAT(4) = FHAT(2)                                                DAV01730
       CAY(4) = CAY(2)                                                  DAV01740
  490 IF(IDAVN.EQ.2) GO TO 8007                                         DAV01750
      CAY(2) = .61*(CAY(4) - CAY(1)) + CAY(1)                           DAV01760
      IF(CAY(2) - 1.0D-15)12,12,495                                     DAV01770
   12 CONTINUE                                                          DAV01780
 8007 CAYMIN=0.0+0                                                      DAV01790
       WRITE(  6  ,492)                                                 DAV01800
  492  FORMAT(1X,46HMINIMUM IS AT ORIGIN - PROGRAM CANNOT CONTINUE)     DAV01810
      RETURN 1                                                          DAV01820
  495  DO 500 I=1,N                                                     DAV01830
  500 XBARS(I)=XBAR(I)+CAY(2)*P(I)                                      DAV01840
      CALL FINT2(FHAT(2),IPART,XBARS,WRK,IWKSZ,&9999)                   DAV01850
       ITT = ITT + 1                                                    DAV01860
       IF(FHAT(2) .LT. FHAT(1))GO TO 460                                DAV01870
       GO TO 485                                                        DAV01880
C                                                                       DAV01890
C  PART 2                                                               DAV01900
C     SHRINK THE GOLDEN SECTION CONTAINING THE MINIMUM                  DAV01910
C                                                                       DAV01920
  510 IF(IIOMEN)13,13,9066                                              DAV01930
   13 CONTINUE                                                          DAV01940
  514 IF(FHAT(3) - FHAT(2))14,525,520                                   DAV01950
   14 CONTINUE                                                          DAV01960
      CAY(1)=CAY(2)                                                     DAV01970
      FHAT(1)=FHAT(2)                                                   DAV01980
      CAY(2)=CAY(3)                                                     DAV01990
      FHAT(2)=FHAT(3)                                                   DAV02000
      CAY(3)=CAY(1)+CAY(4)-CAY(2)                                       DAV02010
      FX = CAY(3)                                                       DAV02020
       JSW = 1                                                          DAV02030
       GO TO 612                                                        DAV02040
  520 CAY(4) = CAY(3)                                                   DAV02050
      CAY(4)=CAY(3)                                                     DAV02060
      FHAT(4)=FHAT(3)                                                   DAV02070
      CAY(3)=CAY(2)                                                     DAV02080
      FHAT(3)=FHAT(2)                                                   DAV02090
      CAY(2)=CAY(1)+CAY(4)-CAY(3)                                       DAV02100
      FX = CAY(2)                                                       DAV02110
      JSW = -1                                                          DAV02120
       GO TO 612                                                        DAV02130
  525 FHAT(1) = FHAT(2)                                                 DAV02140
      FHAT(1)=FHAT(2)                                                   DAV02150
      CAY(4)=CAY(3)                                                     DAV02160
      FHAT(4)=FHAT(3)                                                   DAV02170
       CAY(3) = 3.0D+0*CAY(2) - CAY(3) - CAY(1)                         DAV02180
      CAY(1)=CAY(2)                                                     DAV02190
       CAY(2) = CAY(4) + CAY(1) - CAY(3)                                DAV02200
      FX = CAY(2)                                                       DAV02210
      JSW = 0                                                           DAV02220
       GO TO 612                                                        DAV02230
  536 FHAT(3)=F                                                         DAV02240
C                                                                       DAV02250
C    CHECK ON THE UNIMODALITY OF THE FUNCTION IN THE NEW INTERVAL       DAV02260
C                                                                       DAV02270
  530 IF(FHAT(2) - FHAT(1))15,15,485                                    DAV02280
   15 CONTINUE                                                          DAV02290
      IF(FHAT(3) - FHAT(4))16,16,475                                    DAV02300
   16 CONTINUE                                                          DAV02310
C                                                                       DAV02320
C  PART 3                                                               DAV02330
C   FIND MINIMUM BY EITHER GOLDEN SECTION OR CUBIC FIT TECHNIQUE        DAV02340
C                                                                       DAV02350
  540 IF(CAY(3) - CAY(2)*(1.000+EPSGS))585,17,17                        DAV02360
   17 CONTINUE                                                          DAV02370
```

FILE: DAVDN2 FORTRAN A

```
      IF(KUBIC.EQ.0) GO TO 514                                          DAV02380
      IF(CAY(4) - 1.0D-5)510,542,542                                    DAV02390
  542 CALL CUBIC(CAY,FHAT,XMIN2)                                        DAV02400
      IF(XMIN2)510,510,18                                               DAV02410
   18 CONTINUE                                                          DAV02420
      IF(KUBIC.GT.0) GO TO 2000                                         DAV02430
      IF(XMIN1)580,580,19                                               DAV02440
   19 CONTINUE                                                          DAV02450
      IF(XMIN2*(1.0D0-EPSCF) - XMIN1)20,20,580                          DAV02460
   20 CONTINUE                                                          DAV02470
  545  CAYMIN = (XMIN2 + XMIN1)/2.0D+0                                  DAV02480
       GO TO 9999                                                       DAV02490
  580  XMIN1 = XMIN2                                                    DAV02500
       GO TO 510                                                        DAV02510
  585 CAYMIN=(CAY(3)+CAY(2))/2.0+0                                      DAV02520
       GO TO 9999                                                       DAV02530
C                                                                       DAV02540
  612  DO 613  I=1,N                                                    DAV02550
  613 XBARS(I) = XBAR(I) + FX*P(I)                                      DAV02560
      CALL FINT2(F,IPART,XBARS,WRK,IWKSZ,&9999)                         DAV02570
       ITT = ITT + 1                                                    DAV02580
      IF(JSW)21,21,536                                                  DAV02590
   21 CONTINUE                                                          DAV02600
       FHAT(2) = F                                                      DAV02610
      IF(JSW)540,22,22                                                  DAV02620
   22 CONTINUE                                                          DAV02630
      FX = CAY(3)                                                       DAV02640
       JSW = 1                                                          DAV02650
       GO TO 512                                                        DAV02660
C                                                                       DAV02670
C      REARRANGE Y(I) SO THEY ARE IN ASCENDING ORDER                    DAV02680
C                                                                       DAV02690
 2000 DO 2020 I = 1,4                                                   DAV02700
      X(I) = CAY(I)                                                     DAV02710
 2020 Y(I) = FHAT(I)                                                    DAV02720
      DO 2040 J = 1,3                                                   DAV02730
      DO 2040 I = 1,3                                                   DAV02740
      IF(Y(I) - Y(I+1))2040,23,23                                       DAV02750
   23 CONTINUE                                                          DAV02760
      TEMP = Y(I)                                                       DAV02770
      Y(I) = Y(I+1)                                                     DAV02780
      Y(I+1) = TEMP                                                     DAV02790
      TEMP = X(I)                                                       DAV02800
      X(I) = X(I+1)                                                     DAV02810
      X(I+1) = TEMP                                                     DAV02820
 2040 CONTINUE                                                          DAV02830
 3000  DO 3001  I=1,N                                                   DAV02840
 3001  XBARS(I) = XBAR(I) + XMIN2 * P(I)                                DAV02850
      CALL FINT2(F,IPART,XBARS,WRK,IWKSZ,&9999)                         DAV02860
       ITT = ITT + 1                                                    DAV02870
 3005 DO 3010 K = 1,4                                                   DAV02880
       J = K                                                            DAV02890
      IF(F.LT.Y(K))GO TO 3020                                           DAV02900
 3010 CONTINUE                                                          DAV02910
       GO TO 4000                                                       DAV02920
 3020  I = 4                                                           DAV02930
 3025  IF(I .EQ. J)GO TO 3030                                          DAV02940
       Y(I) = Y(I-1)                                                    DAV02950
       X(I) = X(I-1)                                                    DAV02960
       I = I-1                                                          DAV02970
       GO TO 3025                                                       DAV02980
 3030 Y(J) = F                                                          DAV02990
       X(J) = XMIN2                                                     DAV03000
 3045 CALL CUBIC(X,Y,XMIN2)                                             DAV03010
 3050 IF(XMIN2)24,4000,24                                               DAV03020
   24 CONTINUE                                                          DAV03030
      IF(DABS(XMIN2-X(1)) - EPSCF*XMIN2)25,25,3000                      DAV03040
   25 CONTINUE                                                          DAV03050
       CAYMIN = XMIN2                                                   DAV03060
       GO TO 9999                                                       DAV03070
 4000 IF(IIOMEN)26,26,9066                                              DAV03080
   26 CONTINUE                                                          DAV03090
      KUBIC = -1                                                        DAV03100
       XMIN1 = 0.0D+0                                                   DAV03110
       XMIN2 = 0.0D+0                                                   DAV03120
       GO TO 514                                                        DAV03130
 9999 CCAY=CAYMIN                                                       DAV03140
      IIOMEN=II0                                                        DAV03150
C                                                                       DAV03160
```

FILE: DAVDN2 FORTRAN A

```
C  THIS ENDS  ONE DIMENSIONAL SEARCH ON K                          DAV03170
C                                                                  DAV03180
      30 DO 31 I=1,N                                               DAV03190
         P(I)=CAYMIN*P(I)                                          DAV03200
         XBAR(I)=XBAR(I)+P(I)                                      DAV03210
      31 CONTINUE                                                  DAV03220
         RETURN                                                    DAV03230
         END                                                       DAV03240
```

FILE: DAVDN3 FORTRAN A

```
      SUBROUTINE DAVDN3(FX,FX1,P,H,HY)                                  DAV00010
C*                                                                      DAV00020
C*    DAVDN3 - UPDATES THE H MATRIX AND THE P MATRIX FOR THE NEXT       DAV00030
C*             CYCLE THROUGH THE SEARCH.                                DAV00040
C*           - THE H MATRIX IS BROUGHT IN TO CORE ONE ROW AT A TIME.    DAV00050
C*             SCRATCH FILES 7 AND 28 ARE USED TO STORE AND UPDATE      DAV00060
C*             THIS MATRIX.                                             DAV00070
C*    N***** - NUMBER OF VARIABLES                                      DAV00080
C*    FX**** - GRADIENT OF FUNCTION BEING MINIMIZED                     DAV00090
C*             (PARTIAL DER. WITH RESPECT TO NEW H-MATRIX)              DAV00100
C*    FX1*** - VALUE OF GRADIENT AT THE BEGINNING OF THE CYCLE.         DAV00110
C*             (PARTIAL DER. WITH RESPECT TO OLD H-MATRIX)              DAV00120
C*    H****** - THE H MATRIX APPROXIMATES THE INVERSE MATRIX OF         DAV00130
C*             PARTIAL DERIVATIVES.                                     DAV00140
C*    HY**** - VECTOR USED IN UPDATING THE H-MATRIX                     DAV00150
C*    P***** - VECTOR CORRESPONDING TO SEARCH DIRECTION                 DAV00160
C*             (H-MATRIX TIMES THE GRADIENT)                            DAV00170
C*                                                                      DAV00180
C*                                                                      DAV00190
C     INCLUDE COMBK7.LIST                                               DAV00200
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,               DAV00210
     *     INCVEC(30),ICOUNT,SETWGT,EVALPF(100),FETVC4(30)              DAV00220
     *     ,NOFET4,VARS74,CORMAS,DTAR4,WGHS14,BESTVC(10),DIVSIZ         DAV00230
     *     ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                   DAV00240
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                 DAV00250
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                               DAV00260
C$END                                                                   DAV00270
      DOUBLE PRECISION DFDK,CAYMIN,CCAY,FII                             DAV00280
      DOUBLE PRECISION SIGYI,YHY,DELXFX                                 DAV00290
      DOUBLE PRECISION FX(N),FX1(N),P(N),H(N),HY(N)                     DAV00300
      COMMON/DVNBLK/DFDK,CAYMIN,FII,CCAY,I1D,I1DMEN,ITT,ICNT,N          DAV00310
      N2=N*2                                                            DAV00320
      KAD=ADRSH1                                                        DAV00330
      LAD=ADRSH2                                                        DAV00340
      IF(MOD(ICNT,2).EQ.1)GO TO 10                                      DAV00350
      KAD=ADRSH2                                                        DAV00360
      LAD=ADRSH1                                                        DAV00370
   10 CONTINUE                                                          DAV00380
      CALL RREAD(ADRESF,FX1,N2,ISTAT)                                   DAV00390
C     UNIVAC CHECK DRUM STATUS                                          DAV00400
      DELXFX=0.0                                                        DAV00410
      JAD=KAD                                                           DAV00420
      DO 40 I=1,N                                                       DAV00430
      DELXFX=DELXFX + P(I)*FX(I)/CAYMIN                                 DAV00440
      HY(I)=0.0                                                         DAV00450
C*    READ ONE ROW OF THE H MATRIX                                      DAV00460
      CALL RREAD(JAD,H,N2,ISTAT)                                        DAV00470
C     UNIVAC CHECK DRUM STATUS                                          DAV00480
      JAD=JAD+N2                                                        DAV00490
   20 DO 30 J=1,N                                                       DAV00500
   30 HY(I)=HY(I) + H(J)*(FX(J)-FX1(J))                                 DAV00510
   40 CONTINUE                                                          DAV00520
      YHY=0.0                                                           DAV00530
      SIGYI=0.0                                                         DAV00540
      DO 50 I=1,N                                                       DAV00550
      SIGYI=SIGYI + P(I)*(FX(I)-FX1(I))                                 DAV00560
   50 YHY=YHY + (FX(I)-FX1(I))*HY(I)                                    DAV00570
      IF(SIGYI.LT.1.0-36 .OR. YHY .LT. 1.D-36)GO TO 80                  DAV00580
      DO 65 I=1,N                                                       DAV00590
      FX1(I)=0.0                                                        DAV00600
      CALL RREAD(KAD,H,N2,ISTAT)                                        DAV00610
C     UNIVAC CHECK DRUM STATUS                                          DAV00620
      KAD=KAD+N2                                                        DAV00630
C*    UPDATE THIS ROW OF THE H MATRIX                                   DAV00640
C*                                                                      DAV00650
   55 DO 60 J=1,N                                                       DAV00660
      H(J) = H(J) + P(I)*P(J)/SIGYI - HY(I)*HY(J)/YHY                   DAV00670
   60 CONTINUE                                                          DAV00680
      DO 61 J=1,N                                                       DAV00690
C*                                                                      DAV00700
C*    USE FX1 STORAGE TO UPDATE P ARRAY                                 DAV00710
C*                                                                      DAV00720
      FX1(I)=FX1(I) - H(J)*FX(J)                                        DAV00730
   61 CONTINUE                                                          DAV00740
C*    WRITE UPDATED H                                                   DAV00750
      CALL RWRITE(LAD,H,N2,ISTAT)                                       DAV00760
      LAD=LAD+N2                                                        DAV00770
C     UNIVAC CHECK DRUM STATUS                                          DAV00780
   65 CONTINUE                                                          DAV00790
```

10-22

114

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: DAVDN3 FORTRAN A

```
      DFDK=0.0                                                          DAV00800
      DO 70 I=1,N                                                       DAV00810
      P(I)=FX1(I)                                                       DAV00820
   70 DFDK=DFDK+FX(I)*P(I)                                              DAV00830
C*                                                                      DAV00840
C*    SAVE LATEST PARTIALS ON SCRATCH FILE                              DAV00850
C*                                                                      DAV00860
      CALL RWRITE(ADRESF,FX,N2,ISTAT)                                   DAV00870
C     UNIVAC CHECK DRUM STATUS                                          DAV00880
      RETURN                                                            DAV00890
   80 WRITE(6,300)SIGYI,YHY                                             DAV00900
      RETURN                                                            DAV00910
C*                                                                      DAV00920
  300 FORMAT(' *DAVDN3*--EITHER SIGYI OR YHY TOO CLOSE TO ZERO TO UPDATEDAV00930
     * H MATRIX--SIGYI=',E15.7,' YHY=',E15.7)                           DAV00940
C     UNIVAC CHECK DRUM STATUS                                          DAV00950
      END                                                               DAV00960
```

FILE: DAVIDN FORTRAN A

```
      SUBROUTINE DAVIDN(COVMTX,AVEMTX,DIVTAB,WEIGHT,COVMT2,AVEMT2,S,S2,  DAV00010
     *                  AMAT,PARTLS,WRKRY,IWRKSZ)                        DAV00020
C*****************************************************************     DAV00030
C*                                                                      DAV00040
C*      DAVIDN IS THE DRIVER ROUTINE FOR THE DAVIDON-FEETCHER-POWELL PRODAV00050
C*                                                                      DAV00060
C       INCLUDE COMBK1.LIST                                             DAV00070
C       INCLUDE COMBK7.LIST                                             DAV00080
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          DAV00090
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    DAV00100
     *              FETVC2(30),SUMVC2(75),SUMPTR(75),CLSVC2(60),        DAV00110
     *              KEYPTS(50),NOGRP,GRPNAM(60),GRPDEX(61),             DAV00120
     *              GWPCHR(61),GROUPS(124)                              DAV00130
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,               DAV00140
     *     INVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)               DAV00150
     *              ,NOFET4,VARSZ4,CORBAS,DTAB4,WGHS14,MESTVC(10),DIVSIZ DAV00160
     *     ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                   DAV00170
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                 DAV00180
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                               DAV00190
C$END                                                                   DAV00200
C     EQUIVALENCE (IBLOCK(2),PCHKEY)                                    DAV00210
      DOUBLE PRECISION AMAT(NOFET4,NOFET2)                              DAV00220
      DIMENSION WRKRY(1)                                                DAV00230
      DIMENSION COVMTX(1), AVEMTX(1), WEIGHT(1), S(1)                   DAV00240
      INTEGER FETVC2,FETVC4,BMKEY,CRIKEY,PCHKEY                         DAV00250
      INTEGER SETWGT                                                    DAV00260
      COMMON/DVNALK/DFDK,CAYMIN,FII,CCAY,I1D,I1DMEN,ITT,ICNT,N          DAV00270
      DOUBLE PRECISION DELF,FII,CAYMIN,DFDK,CCAY,PARTLS(1),SMSR         DAV00280
      DOUBLE PRECISION DIVTAB(1),COVMT2(1),AVEMT2(1),S2(1)              DAV00290
      CCAY=1.D-6                                                        DAV00300
      I1D=1                                                             DAV00310
      I1DMEN=1                                                          DAV00320
C*    SET CONVERGENCE TOLERANCE                                         DAV00330
                                                                        DAV00340
      DELF=1.D-6                                                        DAV00350
      IF(CRIKEY.EQ.1.AND.SETWGT.EQ.2)DELF=1.D-1                         DAV00360
      IF(CRIKEY.EQ.3)DELF=1.D-5                                         DAV00370
      ICNT=0                                                            DAV00380
C*                                                                      DAV00390
C*    INITIALIZE B-MATRIX (XBAR)                                        DAV00400
C*                                                                      DAV00410
      IF(BMKEY.LE.0)GO TO 10                                            DAV00420
C*    READ B-MATRIX IN SINGLE PRECISION, THEN STORE IN DOUBLE PRECISION DAV00430
C*                                                                      DAV00440
      CALL BMFIL(WRKRY,NOFET4,NOFET2,FETVC2,2)                          DAV00450
      IK=0                                                              DAV00460
      DO 5 I=1,NOFET2                                                   DAV00470
      DO 5 J=1,NOFET4                                                   DAV00480
      IK=IK+1                                                           DAV00490
    5 AMAT(J,I)=WRKRY(IK)                                               DAV00500
      GO TO 20                                                          DAV00510
C*                                                                      DAV00520
C*    INITIALIZE B-MATRIX FROM 'REST' SET FOUND IN WITHOUT REPLACEMENT PDAV00530
C*                                                                      DAV00540
   10 CALL ORDER(FETVC4,NOFET4)                                         DAV00550
      DO 15 I=1,NOFET4                                                  DAV00560
      DO 15 J=1,NOFET2                                                  DAV00570
      AMAT(I,J)=0.0                                                     DAV00580
   15 IF(FETVC4(I).EQ.FETVC2(J))AMAT(I,J)=1.0                           DAV00590
C*                                                                      DAV00600
C*    INITIALIZE ROUTINE FOR EVALUATING PARTIALS AND SEP.MEASURE        DAV00610
C*                                                                      DAV00620
   20 CALL FINT1(COVMTX,AVEMTX,DIVTAB,WEIGHT,S,S2,COVMT2,AVEMT2,PARTLS) DAV00630
      ITT=0                                                             DAV00640
C*                                                                      DAV00650
C*    COMPUTE PARTIALS FOR INITIAL B-MATRIX                             DAV00660
C*                                                                      DAV00670
      IPART=1                                                           DAV00680
      CALL FINT2(SMSR,IPART,AMAT,WRKRY,IWRKSZ,&25)                      DAV00690
   25 CONTINUE                                                          DAV00700
      ITT=ITT+1                                                         DAV00710
      FII=SMSR                                                          DAV00720
      N=NOFET2*NOFET4                                                   DAV00730
      LEFT=IWRKSZ-N*2                                                   DAV00740
C*                                                                      DAV00750
C*    IP=BASE ADDRESS FOR P ARRAY                                       DAV00760
C*    IF=BASE ADDRESS FOR H ARRAY (ONLY ONE ROW OF H IN CORE AT A TIME  DAV00770
C*    IHY=BASE ADDRESS FOR HY ARRAY                                     DAV00780
C*                                                                      DAV00790
```

FILE: DAVIDN FORTRAN A

```
      IP=1                                                                  DAV00800
      IH=IP+N*2                                                             DAV00810
      IHY=IH+N*2                                                            DAV00820
      IF1=IHY+N*2                                                           DAV00830
      ITEST=IF1+N*2                                                         DAV00840
      IF(ITEST.LE.IWRKSZ)GO TO 30                                           DAV00850
      WRITE(6,100)IWRKSZ                                                    DAV00860
      CALL CMERR                                                            DAV00870
   30 CONTINUE                                                              DAV00880
C*                                                                          DAV00890
C*    INITIALIZE H AND P ARRAYS - SAVE H AND PARTLS ON SCRATCH FILE         DAV00900
C*    ALSO COMPUTE INITIAL DFDK                                             DAV00910
      CALL DAVDN1(PARTLS,WRKRY(IP),WRKRY(IH))                               DAV00920
C*    PARTLS STORAGE CAN NOW BE USED IN DAVIDN2 FOR TEMP. STORAGE OF XRA    DAV00930
C*    DAVDN2 RETURNS A NEW H-MATRIX AND P ARRAY AND CAYMIN                  DAV00940
   40 CALL DAVDN2(HMAT,PARTLS,WRKRY(IP),WRKRY(IH),LEFT,&50)                 DAV00950
      N2=2*N                                                                DAV00960
C*       SAVE P ARRAY ON SCRATCH FILE                                       DAV00970
      CALL RWRITE(ADRESP,WRKRY(IP),N2,LSTAT)                                DAV00980
C*    WAIT FOR I/O                                                          DAV00990
      IF(LSTAT.EQ.1)GO TO 41                                                DAV01000
   41 IF(LSTAT.NE.0)GO TO 70                                                DAV01010
C*    COMPUTE NEW PARTIALS                                                  DAV01020
      CALL FINT2(SMSR,IPART,BMAT,WRKRY,IWRKSZ,&45)                          DAV01030
   45 CONTINUE                                                              DAV01040
      ITT=ITT+1                                                             DAV01050
      ICNT=ICNT+1                                                           DAV01060
      SFRMSR=SMSR                                                           DAV01070
      IF(DABS(SMSR-FII    ).LT.DELF)GO TO 50                                DAV01080
      FII=SMSR                                                              DAV01090
      IF(ITT.GE.ICOUNT)GO TO 50                                             DAV01100
      SMSR=SFRMSR                                                           DAV01110
C*    READ BACK P ARRAY                                                     DAV01120
      CALL RREAD(ADRESP,WRKRY(IP),N2,LSTAT)                                 DAV01130
   46 IF(LSTAT.EQ.1)GO TO 46                                                DAV01140
      IF(LSTAT.NE.0)GO TO 70                                                DAV01150
C*                                                                          DAV01160
C*    SUBROUTINE DAVDN3 UPDATES THE H AND P ARRAYS AND SAVES NEW            DAV01170
C*    H AND NEW PARTIALS ON SCRATCH FILE                                    DAV01180
C*                                                                          DAV01190
      CALL DAVDN3(PARTLS,WRKRY(IF1),WRKRY(IP),WRKRY(IH),WRKRY(IHY))         DAV01200
      GO TO 40                                                              DAV01210
C*                                                                          DAV01220
C*    IF CRITERIA = AV. DIVERGENCE - COMPUTE INTERCLASS DIVERGENCES         DAV01230
C*                                                                          DAV01240
   50 IF(CRIKEY.NE.1)GO TO 60                                               DAV01250
      CALL DIVRG1(COVMT2,VARSZ4,AVEMT2,DIVT18,NOCLS2,NOFET4,                DAV01260
     *            WRKRY,IWRKSZ)                                             DAV01270
   60 CONTINUE                                                              DAV01280
C*                                                                          DAV01290
C*    STORE IN SINGLE PRECISION ARRAY AND                                   DAV01300
C*    WRITE B-MATRIX ON FILE AND PUNCH ON CARDS IF REQUESTED                DAV01310
C*                                                                          DAV01320
      IK=0                                                                  DAV01330
      DO 65 I=1,NOFET2                                                      DAV01340
      DO 65 J=1,NOFET4                                                      DAV01350
      IK=IK+1                                                               DAV01360
      WRKRY(IK)=BMAT(J,I)                                                   DAV01370
   65 CONTINUE                                                              DAV01380
      WMKEY=1                                                               DAV01390
      CALL WMFIL(WRKRY,NOFET4,NOFET2,FETVC2,5)                              DAV01400
C     IF(PCHKEY.EQ.1)CALL WMFIL(WRKRY,NOFET4,NOFET2,FETVC2,4)               DAV01410
      RETURN                                                                DAV01420
   70 WRITE(6,200)LSTAT                                                     DAV01430
  100 FORMAT(' NOT ENOUGH WORK AREA AVAILABLE IN DAVIDN--IWRKSZ=',I6)       DAV01440
  200 FORMAT(/' ERROR ON DUMM FILE - SUBROUTINE DAVIDN---LSTAT=',I3)        DAV01450
      END                                                                   DAV01460
```

FILE: DIVERG FORTRAN A

```
      SUBROUTINE DIVERG(COVMTX,VARSIZ,AVEMTX,DIVTAB,NOCLS,
     *                  NOFET,WRKRY,IWRKSZ)
C*
C*    SUBROUTINE TO COMPUTE INTERCLASS DIVERGENCES
C*
      INTEGER VARSIZ
      DOUBLE PRECISION WRKRY(1),T
      DOUBLE PRECISION DIVTAB,DET,TRACE
      DIMENSION  COVMTX(VARSIZ,NOCLS),  AVEMTX(NOFET,NOCLS),
     *           DIVTAB(1),T(30)
      IDP=1
      GO TO 3
      ENTRY DIVRG1(COVMT2,VARSIZ,AVEMT2,DIVTAB,NOCLS,NOFET,WRKRY,IWRKSZ)
      DOUBLE PRECISION COVMT2(VARSIZ,NOCLS),AVEMT2(NOFET,NOCLS)
      IDP=0
    3 CONTINUE
      ICV1=1
      ICV2=ICV1+VARSIZ
      IW1=ICV2+VARSIZ
      IW2=IW1+VARSIZ
      IF(IWRKSZ/2.GE.IW2+VARSIZ)GO TO 4
      WRITE(6,200)IWRKSZ
      CALL CMERR
    4 CONTINUE
      MN=0
      IC=NOCLS-1
      DO 30 I=1,IC
C*    FIND INVERSE FOR CLASS I COVAR. MATRIX
      DO 1 IK=1,VARSIZ
      IF(IDP.EQ.1)WRKRY(IK)=COVMTX(IK,I)
      IF(IDP.EQ.0)WRKRY(IK)=COVMT2(IK,I)
    1 CONTINUE
      CALL COLINV(WRKRY(ICV1),NOFET,IERR,3,DET)
      IF(IERR.EQ.0)GO TO 2
      WRITE(6,100) I
      GO TO 30
    2 IM=I+1
      DO 20 J=IM,NOCLS
      MN=MN+1
      DO 5 II=1,NOFET
      IF(IDP.EQ.1)T(II)=AVEMTX(II,I)-AVEMTX(II,J)
      IF(IDP.EQ.0)T(II)=AVEMT2(II,I)-AVEMT2(II,J)
    5 CONTINUE
      K=0
      DO 10 II=1,NOFET
      DO 10 JJ=1,II
      K=K+1
      IF(IDP.EQ.1)GO TO 6
      WRKRY(ICV2+K-1)=COVMT2(K,J)
      WRKRY(IW1+K-1)  =COVMT2(K,J)  + T(II)*T(JJ)
      WRKRY(IW2+K-1)  =COVMT2(K,I)  + T(II)*T(JJ)
      GO TO 10
    6 CONTINUE
      WRKRY(ICV2+K-1)=COVMTX(K,J)
      WRKRY(IW1+K-1)  =COVMTX(K,J)  + T(II)*T(JJ)
      WRKRY(IW2+K-1)=COVMTX(K,I)+T(II)*T(JJ)
   10 CONTINUE
C*    FIND INVERSE FOR CLASS J COVAR. MATRIX
      CALL COLINV(WRKRY(ICV2),NOFET,IERR,3,DET)
      IF(IERR.EQ.0)GO TO 15
      WRITE(6,100)J
      GO TO 20
   15 DIVTAB(MN)=.5*(TRACE(WRKRY(ICV1),WRKRY(IW1),NOFET)
     *     + TRACE(WRKRY(ICV2),WRKRY(IW2),NOFET)) - NOFET
   20 CONTINUE
   30 CONTINUE
      RETURN
  100 FORMAT(' COVAR FOR CLASS',I4,' IS NOT POSITIVE DEFINITE')
  200 FORMAT(' NOT ENOUGH WORK AREA AVAILABLE IN DIVERG -- IWRKSZ=',I5)
      END
```

FILE: EVALSP FORTRAN A

```
      SUBROUTINE EVALSP( SMSR,COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,          EVA00010
     *                   DIVTAB,WEIGHT,IPART,PARTLS,AMAT,WRKRY,IWRKSZ)  EVA00020
C*                                                                       EVA00030
C*    THIS SUBROUTINE COORDINATES THE ROUTINES FOR COMPUTING THE         EVA00040
C*    SEPARABILITY MEASURE FOR A PARTICULAR LINEAR COMBINATION OR        EVA00050
C*    SET OF FEATURES                                                    EVA00060
C*                                                                       EVA00070
      DOUBLE PRECISION SMSR,COVMT2(1),AVEMT2(1),S2(1),DIVTAB(1)          EVA00080
     *   ,PARTLS(1),AMAT(1)                                              EVA00090
      INTEGER CRIKEY                                                     EVA00100
C     INCLUDE COMMK1,LIST                                                EVA00110
C     INCLUDE COMMK7,LIST                                                EVA00120
      COMMON/INFO2M/NOCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFL02,           EVA00130
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUMOS2,FLUSV2,VEMTX2,     EVA00140
     *              FETVC2(30),SUMVC2(75),SUMPTM(75),CLSVC2(60),         EVA00150
     *              REMPTS(40),NOGMP,GMPNAM(60),GRPDEX(61),              EVA00160
     *              GRPCHR(61),GROUPS(124)                               EVA00170
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                COM00010
     *    INCVEC(30),ICOUNT,SFTWGT,EVALMF(100),FETVC4(30)                COM00020
     *     ,NOFET4,VARS74,COVMAS,DIAM4,WGHS14,REFTVC(10),DIVSIZ          COM00030
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSM1,ADRSM2                     COM00040
      INTEGER ADRESD,ADRESP,ADRESF,ADRSM1,ADRSM2,STATKY                  COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                COM00060
CSEND                                                                  EVA00240
      DIMENSION COVMTX(1), AVEMTX(1), S(1),                            EVA00250
     2   WEIGHT(1),IPART(1),WRKRY(1)                                     EVA00260
      IFULL=0                                                          EVA00270
      GO TO(10,20,30,40),CRIKEY                                        EVA00280
C*    WEIGHTED AVERAGE DIVERGENCE                                      EVA00290
   10 CALL AVEDIV(SMSR,COVMTX,S,COVMT2,S2,WRKRY,IWRKSZ,                EVA00300
     *            IPART,PARTLS,AMAT,IFULL)                            EVA00310
      RETURN                                                           EVA00320
C*                                                                     EVA00330
C*    WEIGHTED AVERAGE TRANSFORMED DIVERGENCE                          EVA00340
C*                                                                     EVA00350
   20 CALL TRNDIV(SMSR,COVMTX,AVEMTX,COVMT2,AVEMT2,                    EVA00360
     *            WEIGHT,DIVTAB,WRKRY,                                EVA00370
     *            IWRKSZ,IPART,PARTLS,AMAT,IFULL)                     EVA00380
      RETURN                                                           EVA00390
C*                                                                     EVA00400
C*    WEIGHTED AVERAGE BHATTACHARYYA DISTANCE                          EVA00410
C*                                                                     EVA00420
   30 CALL BHTCHR(SMSR,COVMTX,AVEMTX,WEIGHT,DIVTAB,                    EVA00430
     *            COVMT2,AVEMT2,WRKRY,                                EVA00440
     *            IWRKSZ,IPART,PARTLS,AMAT,IFULL)                     EVA00450
C*                                                                     EVA00460
   40 CONTINUE                                                         EVA00470
      RETURN                                                           EVA00480
      END                                                              EVA00490
```

FILE: EVLFET FORTRAN A

```
      SUBROUTINE EVLFET(COVMTX,AVEMTX,DIVTAB,WEIGHT,COVMT2,AVEMT2,
     *                  S,S2,WRKRY,IWRKSZ)
C     INCLUDE COMMK1.LIST
C     INCLUDE COMMA7.LIST
      COMMON/INFOUP/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLO2,
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBIS2,FLOSV2,VERTX2,
     *              FETVC2(30),SUBVC2(75),SUMPTR(75),CLSVC2(60),
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),
     *              GRPCHR(61),GROUPS(124)
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,
     *    INCVEC(30),ICOUNT,SETWGT,EVALRF(100),FETVC4(30)
     *    ,NOFET4,VARSZ4,COMMAS,DTAM4,WGHS14,BESTVC(10),DIVSIZ
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSM1,ADRSM2
      INTEGER ADRESD,ADRESP,ADRESF,ADRSM1,ADRSM2,STATKY
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR
CSEND
      DOUBLE PRECISION COVMT2(1),AVEMT2(1),S2(1),DUM(1),DIVTAB(1)
      INTEGER VARSZ4,FETVC4
      INTEGER CRIKEY
      DIMENSION AVEMTX(1),COVMTX(1),  WEIGHT(1),
     2    S(1),  WRKRY(1)
      IPART=-1
      CALL GTSTAT(COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,FETVC4,
     *  DUM,WRKRY,IWRKSZ)
      CALL EVALSP(SEPMSR,COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,DIVTAB,
     *            WEIGHT,IPART,DUM,DUM,WRKRY,IWRKSZ)
      IF(CRIKEY.NE.1)RETURN
      CALL DIVAG1(COVMT2,VARSZ4,AVEMT2,DIVTAB,NOCLS2,NOFET4,WRKRY,
     *            IWRKSZ)
      RETURN
      END
```

EVL00010
EVL00020
EVL00030
EVL00040
EVL00050
EVL00060
EVL00070
EVL00080
EVL00090
EVL00100
EVL00110
EVL00120
EVL00130
EVL00140
EVL00150
EVL00160
EVL00170
EVL00180
EVL00190
EVL00200
EVL00210
EVL00220
EVL00230
EVL00240
EVL00250
EVL00260
EVL00270
EVL00280
EVL00290
EVL00300
EVL00310

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: EXSRCH FORTRAN A

```
      SUBROUTINE EXSRCH(COVMTX,AVEMTX,DIVTAB,WEIGHT,COVMT2,                 EXS00010
     *                  AVEMT2,S,S2,WRKRY,IWRKSZ)                           EXS00020
C*                                                                          EXS00030
CC*   THIS SUBROUTINE USES THE EXHAUSTIVE SEARCH PROCEDURE TO FIND          EXS00040
CC*   THE BEST 'NOFET4' OUT OF 'NOFET2' FEATURES. BY MAXIMIZING THE         EXS00050
CC*   SEPARABILITY MEASURE INDICATED BY 'CRIKEY'.                           EXS00060
CC*                                                                         EXS00070
CCC   INCLUDE COMMK1.LIST                                                   EXS00080
CC    INCLUDE COMMK7.LIST                                                   EXS00090
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,              EXS00100
     *              AVAR2,COVAR2,CLS1D2,SUBNO2,SUBDS2,FLDSV2,VERTX2,        EXS00110
     *              FETVC2(30),SUMVC2(75),SUMPTM(75),CLSVC2(60),            EXS00120
     *              KEPPTS(50),NOGRP,GRPNAM(60),GRPDEX(61),                 EXS00130
     *              GRPCHK(41),GROUPS(124)                                  EXS00140
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                   COM00010
     *     INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)                  COM00020
     *     ,NOFET4,VARSZ4,CORMAS,DTAH4,WGHS14,RESTVC(10),DIVSIZ             COM00030
     *    ,STATKY,ADRESD,ADRESP,ADRESP,ADRESF,ADRSM1,ADRSM2                 COM00040
      INTEGER ADRESD,ADRESP,ADRESF,ADRSM1,ADRSM2,STATKY                     COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                   COM00060
CSEND                                                                       EXS00220
      INTEGER FETVC2,FETVC4,TVEC,VARSZ2,VARSZ4                              EXS00230
      INTEGER DIVSIZ,SVEC(30),CRIKEY                                        EXS00240
      DOUBLE PRECISION TMSR,DIVTAB(DIVSIZ),DUM(1),DM                        EXS00250
      DOUBLE PRECISION COVMT2(VARSZ4,1),AVEMT2(NOFET4,1),S2(VARSZ4,1)       EXS00260
      DIMENSION COVMTX(VARSZ2,NOCLS2) ,                                     EXS00270
     *          AVEMTX(NOFET2,NOCLS2) ,                                     EXS00280
     *          S(VARSZ2,NOCLS2)                                            EXS00290
      DIMENSION TVEC(30)                                                    EXS00300
      DIMENSION WEIGHT(1),WRKRY(1)                                          EXS00310
C*                                                                          EXS00320
CC*   INITIALIZE TVEC FOR GETSET ROUTINE                                    EXS00330
CC*                                                                         EXS00340
      DO 1 I=1,NOFET4                                                       EXS00350
    1 TVEC(I)=I                                                             EXS00360
      TVEC(NOFET4)=NOFET4-1                                                 EXS00370
      SEPMSW=1.E-36                                                         EXS00380
C*    GET NEXT SET OF FEATURES                                              EXS00390
CC*                                                                         EXS00400
    4 CALL GETSET(TVEC,NOFET4,NOFET2,LAST)                                  EXS00410
      IF(LAST.EQ.0)GO TO 10                                                 EXS00420
C*                                                                          EXS00430
CC*   GET SUBSET OF STATISTICS FOR THIS SET OF FEATURES                     EXS00440
CC*                                                                         EXS00450
      CALL GTSTAT(COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,TVEC,DM,WRKRY,IWRKSZ)    EXS00460
C*                                                                          EXS00470
CC*   EVALUATE SEPARABILITY MEASURE FOR THIS SET OF FEATURES.               EXS00480
CC*   SET IPART SO PARTIAL DERIVATIVES WILL NOT BE CALCULATED.              EXS00490
CC*                                                                         EXS00500
      IPART=-1                                                              EXS00510
      CALL EVALSP(TMSR,COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,DIVTAB,             EXS00520
     *            WEIGHT,IPART,DUM,DUM,WRKRY,IWRKSZ)                        EXS00530
      IF(SEPMSW.LT.TMSR)GO TO 4                                             EXS00540
      DO 5 I=1,NOFET4                                                       EXS00550
      K=TVEC(I)                                                             EXS00560
      SVEC(I)=K                                                             EXS00570
    5 FETVC4(I)=FETVC2(K)                                                   EXS00580
      SEPMSW=TMSR                                                           EXS00590
      GO TO 4                                                               EXS00600
C*    FINISHED                                                              EXS00610
CC*                                                                         EXS00620
CCC*                                                                        EXS00630
CC*   COMPUTE INTERCLASS MEASURES FOR FEATURES CHOSEN                       EXS00640
CC*                                                                         EXS00650
   10 CONTINUE                                                              EXS00660
      CALL GTSTAT(COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,SVEC,DM,WRKRY,IWRKSZ)    EXS00670
      CALL EVALSP(SEPMSW,COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,DIVTAB,           EXS00680
     *            WEIGHT,IPART,DUM,DUM,WRKRY,IWRKSZ)                        EXS00690
      IF(CRIKEY.NE.1)RETURN                                                 EXS00700
      CALL DIVHG1(COVMT2,VARSZ4,AVEMT2,DIVTAB,NOCLS2,                       EXS00710
     *            NOFET4,WRKRY,IWRKSZ)                                      EXS00720
   20 RETURN                                                                EXS00730
      END                                                                   EXS00740
```

```
FILE: FINT1    FORTRAN  A

      SUBROUTINE FINT1(COVMTX,AVEMTX,DIVTAB,WEIGHT,S,S2,COVMT2,          FIN00010
     *                 AVEMT2,PARTLS)                                    FIN00020
C*                                                                       FIN00030
C*    THIS SUBROUTINE IS A DRIVER FOR OBTAINING THE PARTIALS AND/        FIN00040
C*    OR SEPARABILITY MEASURE FOR THE DAVIDON PROCEDURE.                 FIN00050
C*                                                                       FIN00060
C*    --THE FIRST ENTRY POINT (FINT1) IS FOR INITIALIZING ADDRESSES FOR  FIN00070
C*      LONG ARGUMENT LIST, AND PRINTING THE HEADER FOR CONVERGENCE      FIN00080
C*      CHARACTERISTIC SUMMARY.                                          FIN00090
C*    --ENTRY POINT FINT2 MUST BE CALLED FOR EACH EVALUATION OF          FIN00100
C*      THE SEPARABILITY MEASURE OR PARTIALS                             FIN00110
C*                                                                       FIN00120
C*                                                                       FIN00130
C*    COVMTX - (INPUT) COVARIANCE MATRIX FOR EACH CLASS.                 FIN00140
C     INCLUDE COMMK6.LIST                                                FIN00150
C     INCLUDE COMMK1.LIST                                                FIN00160
C     INCLUDE COMMK7.LIST                                                FIN00170
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           FIN00180
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     FIN00190
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         FIN00200
     *              KEEPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              FIN00210
     *              GRPCHK(61),GROUPS(124)                               FIN00220
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          FIN00230
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILF,    FIN00240
     *    DPUMAD,DRMMDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 FIN00250
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    FIN00260
     *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              FIN00270
     *   CROUNT,PRTUNT,RANDIO                                            FIN00280
      COMMON/ESL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                FIN00290
     *    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)                FIN00300
     *     ,NOFET4,VARSZ4,CORBAS,DTAB4,WGHS14,RESTVC(10),DIVSIZ           FIN00310
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                      FIN00320
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                  FIN00330
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                FIN00340
CSEND                                                                    FIN00350
      INTEGER VARSZ2,VARSZ4                                              FIN00360
      DIMENSION COVMTX(VARSZ2,NOCLS2)  ,  AVEMTX(NOFET2,NOCLS2),         FIN00370
     *  S(VARSZ2,NOCLS2), WEIGHT(1)                                      FIN00380
      DOUBLE PRECISION   COVMT2(VARSZ4,NOCLS2)  ,  AVEMT2(NOFET4,NOCLS2)FIN00390
     *                 ,  S2(VARSZ4,NOCLS2),                             FIN00400
     *            DIVTAB(1),PARTLS(1)                                    FIN00410
      DOUBLE PRECISION DUM,RATIO,SPSR                                    FIN00420
      INTEGER CRIKEY                                                     FIN00430
      DIMENSION FRM(46),VFRM(3)                                          FIN00440
      COMMON/FNTDUM/ ITT, ICYCLF                                         FIN00450
      DIMENSION DUM(1)                                                   FIN00460
      DATA VFRM/'.1HD','.1HT','.1HH'/                                    FIN00470
      DATA ADATA/'2H/ '/, ADATA2/'.T98'/                                 FIN00480
      DATA FRM/'(///','/45X','.34H','CONV','ERGE','NCE ','CHAR','ACTE',  FIN00490
     1  'RIST','IC S','UMMA','RY/3','5X,1','0HFU','NCTI','ONAL','.T59',  FIN00500
     2  '.16H','FUNC','TION','AL V','ALUE','.T94','.5HR=','ATIO','/20X',  FIN00510
     3  '.5HC=','YCLE','.T35','.11H','EVAL','UATI','ONS,','T66 ','    ',  FIN00520
     4  '.T95','    ','    ','2H /','    ','/1X ','.T67','.1HR','.T96',  FIN00530
     5  '.1HR','/)  */                                                  FIN00540
      ITT=0                                                              FIN00550
      ICYCLF=0                                                           FIN00560
      WRITE(6,HEAD)                                                      FIN00570
      WRITE(6,100)                                                       FIN00580
      WRITE(6,200)NOFET4,ICOUNT                                          FIN00590
      GO TO (10,20,30),CRIKEY                                            FIN00600
   10 WRITE(6,300)TOTMSR                                                 FIN00610
      GO TO 40                                                           FIN00620
   20 WRITE(6,400)TOTMSR                                                 FIN00630
      GO TO 40                                                           FIN00640
   30 WRITE(6,500)TOTMSR                                                 FIN00650
   40 FRM(35)=VFRM(CRIKEY)                                               FIN00660
      FRM(37)=VFRM(CRIKEY)                                               FIN00670
      FRM(40)=VFRM(CRIKEY)                                               FIN00680
      IF(CRIKEY.NE.1)FRM(39)=ADATA                                       FIN00690
      IF(CRIKEY.NE.1)FRM(44)=ADATA2                                      FIN00700
      WRITE(6,FRM)                                                       FIN00710
      RETURN                                                             FIN00720
      ENTRY FINT2(SPSR,IPART,BMAT,WRKRY,IWRKSZ,*)                        FIN00730
C*                                                                       FIN00740
C*      SPSR - SEPARABILITY MEASURE RETURNED FROM EVALSP                 FIN00750
C*      IPART- TRIGGER TO COMPUTE PARTIALS OR NOT                        FIN00760
C*           - LESS THAN 0 - COMPUTE PARTIALS                            FIN00770
C*           - GREATER THAN 0 - DO NOT COMPUTE PARTIALS                  FIN00780
C*      BMAT - H-MATRIX                                                  FIN00790
```

```
FILE: FINT1    FORTRAN  A

C*       WRKRY- WORKING STORAGE                                                FIN00800
C*       IWRKZ- SIZE OF WRKRY IN COMPUTER WORDS                                FIN00810
C*                                                                             FIN00820
C*************************************************************************** FIN00830
         DIMENSION BMAT(1),WRKRY(1)                                            FIN00840
         DOUBLE PRECISION BMAT,SMSR                                            FIN00850
C*                                                                             FIN00860
C*       GET TRANSFORMED STATISTICS FOR THIS B-MATRIX                          FIN00870
C*                                                                             FIN00880
         CALL GTSTAT(COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,DUM,BMAT,WRKRY,          FIN00890
     *                IWRKSZ)                                                  FIN00900
         CALL EVALSP(SPSP,COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,DIVTAB,             FIN00910
     *               WEIGHT,IPART,PARTLS,BMAT,WRKRY,IWRKSZ)                    FIN00920
          SMSR=DABS(SPSP)                                                      FIN00930
          ITT=ITT+1                                                            FIN00940
          ICNT=ICNT+1                                                          FIN00950
          IF(ICYCLE.LE.1)GO TO 50                                              FIN00960
          IF(ICNT.LE.30)GO TO 50                                               FIN00970
          WRITE(6,700)                                                         FIN00980
          ICNT=0                                                               FIN00990
          RETURN 1                                                             FIN01000
     50   CONTINUE                                                             FIN01010
          IF(IPART.LT.0)RETURN                                                 FIN01020
          ICNT=0                                                               FIN01030
          ICYCLE=ICYCLE+1                                                      FIN01040
          IF(CRITEY.EQ.1)RATIO=SMSR/TOTMSR                                     FIN01050
          IF(CRITEY.NE.1)RATIO=TOTMSR/SMSR                                     FIN01060
          WRITE(6,600)ICYCLE,ITT,SMSR,RATIO                                    FIN01070
          RETURN                                                               FIN01080
    100   FORMAT(///25X,'CONVERGENCE CHARACTERISTIC SUMMARY FOR THE DAVIDON-FFIN01090
         *LETCHER-POWELL PROCEDURE'/25X,76('-')/)                              FIN01100
    200   FORMAT(//35X,'NUMBER OF LINEAR COMBINATIONS',10X,'=',I10/            FIN01110
         *    35X,'DESIRED NO. OF FUNCTIONAL EVAL.',8X,'=',I10)                FIN01120
    300   FORMAT(35X,'MAX. WEIGHTED AVERAGE DIVERGENCE (D)',3X,'=',E10.5)      FIN01130
    400   FORMAT(35X,'MIN. WEIGHTED AV.TRANS. DIVERGENCE (T)=',E10.5)          FIN01140
    500   FORMAT(35X,'MIN. WEIGHTED AV.BHATTACHARYYA DIS. (M)=',E10.5)         FIN01150
    600   FORMAT(20X,I4,T37,I4,T61,E12.7,T90,E12.7)                            FIN01160
    700   FORMAT(' MAX. ITERATIONS PER CYCLE - BEGIN NEW CYCLE')               FIN01170
          END                                                                  FIN01180
```

FILE: GENRPT FORTRAN A

```
      SUBROUTINE GENRPT(CLSNAM,WEIGHT,DIVTAB,WRKRY,IWRKSZ,FETVEC)          GEN00010
      DIMENSION CLSNAM(NOCLS2),WEIGHT(1)                                   GEN00020
C     INCLUDE COMBK6.LIST                                                  GEN00030
      DOUBLE PRECISION RATIO                                               GEN00040
      DOUBLE PRECISION WRKRY(1),DIVTAB(1)                                  GEN00050
      DIMENSION FETVEC(30)                                                 GEN00060
C     INCLUDE COMBK1.LIST                                                  GEN00070
C     INCLUDE COMBK7.LIST                                                  GEN00080
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,             GEN00090
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,       GEN00100
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),           GEN00110
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                GEN00120
     *              GRPCHK(61),GROUPS(124)                                 GEN00130
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,            GEN00140
     *             HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILF,       GEN00150
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                   GEN00160
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                       GEN00170
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                 GEN00180
     *  CHUNT,PRTUNT,RANDIO                                                GEN00190
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                  GEN00200
     *    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)                  GEN00210
     *     ,NOFET4,VARS74,CORBAS,DTAR4,WGHS14,RESTVC(10),DIVSIZ            GEN00220
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                       GEN00230
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                    GEN00240
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                  GEN00250
C$END                                                                      GEN00260
      INTEGER CRIKEY,PRCKEY                                                GEN00270
      DIMENSION PRC(6,5),CRI(13,4)                                         GEN00280
      DATA PRC/'EXMA','USTI','VE S','EARC','H   ','    ',                 GEN00290
     *         'WITH','OUT ','REPL','ACEM','ENT ','    ',                 GEN00300
     *         'DAVI','DON-','FLET','CHER','-POW','ELL ',                 GEN00310
     *         'EVAL','. B-','MATR','IX R','EQUE','ST  ',                 GEN00320
     *         'EVAL','. FE','ATUR','ES R','EQUE','ST  ',                 GEN00330
     *         'REST','  K ','  OF','  N ','  PA','SSES'/                 GEN00340
      DATA CRI/'WEIG','HTED',' AVE','RAGE',' DIV','ERGE','NCE ',         GEN00350
     *         '    ','    ','    ','    ','    ','    ','    ',         GEN00360
     *         'WEIG','HTED',' AV.',' TRA','NSFO','RMED',' DIV',         GEN00370
     *         'ERGE','NCE ','    ','    ','    ','    ','    ',         GEN00380
     *         'WEIG','HTED',' AV.',' BHA','TTAC','HARY','YA D',         GEN00390
     *         'ISTA','NCE ','    ','    ','    ','    ','    ',         GEN00400
     *         'PROB','ABIL','ITY ',' OF M','ISCL','ASSI','FICA',         GEN00410
     *         'TION',' OF ','TRAI','NING',' DAT','A   '/                 GEN00420
      INTEGER DIVSIZ,PAGSIZ                                                GEN00430
      INTEGER FETVC4                                                       GEN00440
      IPCNT=27                                                             GEN00450
      WRITE(6,HEAD)                                                        GEN00460
      WRITE(6,100)(PRC(I,PRCKEY),I=1,6),(CRI(I,CRIKEY),I=1,13)             GEN00470
      IB=1                                                                 GEN00480
      IE=15                                                                GEN00490
    5 IF(IE.GT.NOFET2)IE=NOFET2                                            GEN00500
      WRITE(6,105)(FETVC2(I),I=IB,IE)                                      GEN00510
      IF(IE.EQ.NOFET2)GO TO 10                                             GEN00520
      IB=IE+1                                                              GEN00530
      IE=IE+15                                                             GEN00540
      GO TO 5                                                              GEN00550
   10 CONTINUE                                                             GEN00560
      WRITE(6,125)                                                         GEN00570
      IF(PRCKEY.EQ.3.OR.PRCKEY.EQ.4)WRITE(6,110)NOFET4                     GEN00580
      IF(PRCKEY.EQ.3.OR.PRCKEY.EQ.4)GO TO 18                               GEN00590
      IB=1                                                                 GEN00600
      IE=16                                                                GEN00610
   12 IF(IE.GT.NOFET4)IE=NOFET4                                            GEN00620
      IF(PRCKEY.LT.3)WRITE(6,120)(FETVC4(I),I=IB,IE)                       GEN00630
      IF (PRCKEY.GE.5) WRITE(6,115) (FETVEC(I), I=IB,IE)                   GEN00640
      IF(IE.EQ.NOFET4)GO TO 18                                             GEN00650
      IB=IE+1                                                              GEN00660
      IE=IE+16                                                             GEN00670
      GO TO 12                                                             GEN00680
   18 CONTINUE                                                             GEN00690
      IF(CRIKEY.EQ.1)SEPMSR=DABS(SEPMSR)                                   GEN00700
      IF(PRCKEY .EQ.3)WRITE(6,130)SEPMSR                                   GEN00710
      IF(PRCKEY .EQ.4)WRITE(6,130)SEPMSR                                   GEN00720
      IF (PRCKEY.GE.5) WRITE(6,145) SEPMSR                                 GEN00730
      IF(PRCKEY.LT.3)WRITE(6,140)SEPMSR                                    GEN00740
      IF(CRIKEY.EQ.1)RATIO=SEPMSR/TOTMSR                                   GEN00750
      IF(CRIKEY.NE.1)RATIO=TOTMSR/SEPMSR                                   GEN00760
      IF(CRIKEY.NE.1)WRITE(6,150)TOTMSR,RATIO                              GEN00770
      IF(CRIKEY.EQ.1)WRITE(6,155)TOTMSR,RATIO                              GEN00780
      WRITE(6,160)                                                         GEN00790
```

FILE: GENRPT FORTRAN A

```
      IF(CRIKEY.EQ.1)WRITE(6,170)                                          GEN00800
      IF(CRIKEY.EQ.2)WRITE(6,175)                                          GEN00810
      IF(CRIKEY.EQ.3)WRITE(6,180)                                          GEN00820
      IF(PRCKEY.LE.2)WRITE(6,190)                                          GEN00830
      IF(PRCKEY.EQ.3)WRITE(6,200)                                          GEN00840
      IF(PRCKEY.EQ.4)WRITE(6,200)                                          GEN00850
      IF (PRCKEY.GE.5) WRITE(6,205)                                        GEN00860
      WRITE(6,125)                                                         GEN00870
C*                                                                         GEN00880
C*    READ INTERCLASS MEASURE FOR ALL FEATURES - COMPUTED AND SAVED IN     GEN00890
C*    PRELIM.                                                              GEN00900
C*                                                                         GEN00910
      NW=DIVSIZ*2                                                          GEN00920
      CALL RREAD(ADDRESD,WRKRY,NW,ISTAT)                                   GEN00930
   19 IF(ISTAT.EQ.1)GO TO 19                                               GEN00940
      NC=NOCLS2-1                                                          GEN00950
      IK=0                                                                 GEN00960
      DO 20 I=1,NC                                                         GEN00970
      K=I+1                                                                GEN00980
      DO 20 J=K,NOCLS2                                                     GEN00990
      IK=IK+1                                                              GEN01000
      IF(CRIKEY.EQ.1)RATIO=DIVTAB(IK)/WRKRY(IK)                            GEN01010
      IF(CRIKEY.NE.1)RATIO=WRKRY(IK)/DIVTAB(IK)                            GEN01020
      WRITE(6,210)CLSNAM(I),CLSNAM(J),                                     GEN01030
     *            WEIGHT(IK),DIVTAB(IK),WRKRY(IK),RATIO                    GEN01040
      IPCNT=IPCNT+1                                                        GEN01050
      IF(IPCNT.LT.PAGSIZ)GO TO 20                                          GEN01060
      IPCNT=17                                                             GEN01070
      IF(IK.EQ.DIVSIZ)GO TO 20                                             GEN01080
      WRITE(6,HEAD)                                                        GEN01090
      WRITE(6,160)                                                         GEN01100
      IF(CRIKEY.EQ.1)WRITE(6,170)                                          GEN01110
      IF(CRIKEY.EQ.2)WRITE(6,175)                                          GEN01120
      IF(CRIKEY.EQ.3)WRITE(6,180)                                          GEN01130
      IF(PRCKEY.LE.2)WRITE(6,190)                                          GEN01140
      IF(PRCKEY.EQ.3)WRITE(6,200)                                          GEN01150
      IF(PRCKEY.EQ.4)WRITE(6,200)                                          GEN01160
      IF (PRCKEY.GE.5) WRITE(6,205)                                        GEN01170
      WRITE(6,125)                                                         GEN01180
   20 CONTINUE                                                             GEN01190
      IF(CRIKEY.NE.2)RETURN                                                GEN01200
C*                                                                         GEN01210
C*    GET DIVERGENCE BACK FOR PLOTS                                        GEN01220
C*                                                                         GEN01230
      DO 30 I=1,DIVSIZ                                                     GEN01240
      WRKRY(I)=-16.*DLOG(WRKRY(I))                                         GEN01250
      DIVTAB(I)=-16.*DLOG(DIVTAB(I))                                       GEN01260
   30 CONTINUE                                                             GEN01270
  100 FORMAT(1X,T35,'RESULTS FOR CHANNEL SELECTION ACTIVITY USING:'/       GEN01280
     *       1X,T40,'OPTIMIZATION PROCEDURE - ',6A4/                       GEN01290
     *       1X,T40,'SEPARABILITY MEASURE   - ',13A4)                      GEN01300
  105 FORMAT(1X,T40,'CHANNELS CONSIDERED     - ',15(I2,','))               GEN01310
  110 FORMAT(1X,T35,'NO. OF LINEAR COMBINATIONS                  - ',I3)   GEN01320
  115 FORMAT(1X,T35,'CHANNELS EVALUATED                          - ',      GEN01330
     * 16(I2,','))                                                         GEN01340
  120 FORMAT(1X,T35,'CHANNELS SELECTED                           - ',      GEN01350
     *  I2,15(','I2))                                                     GEN01360
  125 FORMAT(/)                                                            GEN01370
  130 FORMAT(1X,T35,'SEPARABILITY MEASURE FOR LINEAR COMB.       - ',      GEN01380
     *  F14.8)                                                            GEN01390
  140 FORMAT(1X,T35,'SEPARABILITY MEASURE FOR SELECTED CHANNELS - ',       GEN01400
     *  F14.8)                                                            GEN01410
  145 FORMAT(1X,T35,'SEPARABILITY MEASURE FOR EVALUATE REQUEST  - ',       GEN01420
     * E14.8)                                                            GEN01430
  150 FORMAT(1X,T35,'MINIMUM SEP.MEASURE (USING ALL CHANNELS)    - ',      GEN01440
     * E14.8/1X,T35,'RATIO',T78,'- ',F14.8)                              GEN01450
  155 FORMAT(1X,T35,'MAXIMUM SEP.MEASURE (USING ALL CHANNELS)    - ',      GEN01460
     *F14.8/1X,T35,'RATIO',T78,'- ',F14.8)                              GEN01470
  160 FORMAT(////   1X,T50,'INTERCLASS SEPARABILITY TABLE' //)            GEN01480
  170 FORMAT(1X,T73,'INTERCLASS DIVERGENCE')                               GEN01490
  175 FORMAT(1X,T65,'INTERCLASS TRANSFORMED DIVERGENCE')                   GEN01500
  180 FORMAT(1X,T65,'INTERCLASS BHATTACHARYYA DISTANCE')                   GEN01510
  190 FORMAT(1X,T22,'SUBCLASS PAIR',T41,'WEIGHT',T58,'SELECTED CHANNELS'   GEN01520
     *, T80,'ALL CHANNELS',T99,'RATIO')                                    GEN01530
  200 FORMAT(1X,T22,'SUBCLASS PAIR',T41,'WEIGHT',T58,'LINEAR COMBINATION'  GEN01540
     *, T80,'ALL CHANNELS',T99,'RATIO')                                    GEN01550
  205 FORMAT(1X,T24,'SUBCLASS PAIR',T41,'WEIGHT',T58,'EVALUATED CHANNELS'  GEN01560
     *, T80,'ALL CHANNELS',T99,'RATIO')                                    GEN01570
  210 FORMAT(20X,A4,7X,A4,4X,E8.3,T60,D14.8,T78,D14.8,T96,D14.8)           GEN01580
```

FILE: GENRPT FORTRAN A

```
      RETURN                                                    GEN01590
      END                                                       GEN01600
```

```
FILE: GETSET   FORTRAN  A

      SUBROUTINE GETSET(VEC,NOFET4,NOFET2,LAST)
      INTEGER VEC
C*
C*    SUBROUTINE TO RETURN A UNIQUE SET OF INDICES FOR NOFET4 FEATURES
C*    IN VEC.  VEC SHOULD BE INITIALIZED TO 1,2,3...(NOFET4-1) BEFORE
C*    ENTERING THIS ROUTINE FOR THE FIRST TIME.
C*
      DIMENSION VEC(1)
      DO 30 JJ=1,NOFET4
      J2=NOFET4-JJ+1
      J3=J2+1
      VEC(J2) = VEC(J2)+1
      IF(J3.GT.NOFET4)GO TO 20
      DO 10 JJJ=J3,NOFET4
   10 VEC(JJJ)=VEC(JJJ-1)+1
   20 IF(VEC(NOFET4).LE.NOFET2)GO TO 40
   30 CONTINUE
      LAST=0
      RETURN
   40 LAST=1
      RETURN
      END
```

FILE: GTSTAT FORTRAN A

```
      SUBROUTINE GTSTAT(COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,                GTS00010
     *                  VEC,BMAT,WRKRY,IWRKSZ)                           GTS00020
C*                                                                       GTS00030
C*    THIS SUBROUTINE SELECTS THE SUBSETS OF THE STATISTICAL             GTS00040
C*    PARAMETERS COVMTX,AVEMTX AND S DEFINED BY VEC OR BMAT AND          GTS00050
C*    STORES THE SUBSETS INTO COVMT2,AVEMT2, AND S2 RESPECTIVELY.        GTS00060
C*                                                                       GTS00070
C     INCLUDE COMMN1,LIST                                                GTS00080
C     INCLUDE COMMK7,LIST                                                GTS00090
      COMMON/INFOMM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           GTS00100
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLOSV2,VERTX2,     GTS00110
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         GTS00120
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              GTS00130
     *              GRPCHK(61),GROUPS(124)                               GTS00140
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                COM00010
     *    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)                COM00020
     *     ,NOFET4,VARSZ4,CORHAS,DTAR4,WGHS14,RESTVC(10),DIVSIZ           COM00030
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                     COM00040
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                  COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                COM00060
CSEND                                                                    GTS00220
      INTEGER PRCKEY,CRIKEY,VEC,VARSZ2,VARSZ4                            GTS00230
      DOUBLE PRECISION COVMT2(VARSZ4,1),AVEMT2(NOFET4,1),                GTS00240
     *   S2(VARSZ4,1),BMAT(NOFET4,1),WRKRY(1),SUM                        GTS00250
      DIMENSION  COVMTX(VARSZ2,NOCLS2),                                  GTS00260
     *           AVEMTX(NOFET2,NOCLS2),                                  GTS00270
     *           S(VARSZ2,NOCLS2),VEC(1)                                 GTS00280
C*                                                                       GTS00290
C*    IF WITHOUT REPLACEMENT OR EX. SEARCH PROCEDURE, SELECT ELEMENTS    GTS00300
C*    DETERMINED BY 'VEC'.                                               GTS00310
      GO TO (5,5,20,20,5,5),PRCKEY                                       GTS00320
    5 CONTINUE                                                           GTS00330
      DO 10 I=1,NOCLS2                                                   GTS00340
      IK=0                                                               GTS00350
      DO 10 J=1,NOFET4                                                   GTS00360
      K=VEC(J)                                                           GTS00370
      LOC=K*(K-1)/2                                                      GTS00380
      AVEMT2(J,I)=AVEMTX(K,I)                                            GTS00390
      DO 10 L=1,J                                                        GTS00400
      IK=IK+1                                                            GTS00410
      IW=VEC(L)+LOC                                                      GTS00420
      COVMT2(IK,I)=COVMTX(IW,I)                                          GTS00430
      IF(CRIKEY.NE.1)GO TO 10                                            GTS00440
      S2(IK,I)=S(IW,I)                                                   GTS00450
   10 CONTINUE                                                           GTS00460
      RETURN                                                             GTS00470
C*                                                                       GTS00480
C*    DAVIDON OR USER INPUT PROCEDURE, MULTIPLY B-MATRIX                 GTS00490
C*                                                                       GTS00500
   20 CONTINUE                                                           GTS00510
      IW1=1                                                              GTS00520
      ITEST= IW1 + NOFET2*NOFET4                                         GTS00530
      IF(IWRKSZ/2.GE.ITEST)GO TO 30                                      GTS00540
      WRITE(6,100)IWRKSZ                                                 GTS00550
      CALL CMERR                                                         GTS00560
   30 CONTINUE                                                           GTS00570
      DO 80 I=1,NOCLS2                                                   GTS00580
      DO 60 J=1,NOFET4                                                   GTS00590
      SUM=0.0                                                            GTS00600
      DO 50 K=1,NOFET2                                                   GTS00610
   50 SUM=SUM+AVEMTX(K,I)*BMAT(J,K)                                      GTS00620
   60 AVEMT2(J,I)=SUM                                                    GTS00630
   80 CONTINUE                                                           GTS00640
      CALL TRNSFR(COVMTX,COVMT2,WRKRY(IW1),BMAT)                         GTS00650
      IF(CRIKEY.EQ.1)CALL TRNSFR(S,S2,WRKRY(IW1),BMAT)                   GTS00660
      RETURN                                                             GTS00670
  100 FORMAT(' NOT ENOUGH WORK AREA IN GTSTAT -- IWRKSZ=',I5)            GTS00680
      END                                                                GTS00690
```


ORIGINAL PAGE IS
OF POOR QUALITY

FILE: MT1 FORTRAN A

```
C*
C*
C*       MT1 - MATRIX B IS STORED IN SYMMETRIC NOTATION AND SINGLE PRE.
C*
C*
C*
      SUBROUTINE MT1(A,B,C,M,N)
      DOUBLE PRECISION A(M,N),C(M,N),SUM
      DIMENSION B(1)
C*    MATRIX B IS STORED IN SYMMETRIC NOTATION
      DO 60 II=1,M
      DO 60 JJ=1,N
      SUM=0.0
      DO 55 KK=1,N
      IF(KK.GE.JJ) IK=KK*(KK-1)/2 + JJ
      IF(KK.LT.JJ) IK=JJ*(JJ-1)/2 + KK
   55 SUM = SUM + A(II,KK)*B(IK)
   60 C(II,JJ)=SUM
      RETURN
      END
```

```
FILE: MT2      FORTRAN  A

C*     SUBROUTINE TO FORM PRODUCT OF MATRICES A*B AND STORE IN C.
C*
C*        MT2 - MATRIX A IS STORED IN SYMMETRIC NOTATION
C*
          SUBROUTINE MT2(A,B,C,M,N)
          DOUBLE PRECISION A(1),B(M,N),C(M,N),SUM1
          DO 90 II=1,M
          DO 90 JJ=1,N
          SUM1=0.0
          DO 85 IJ=1,M
       IF(IJ.GE.II)IK=IJ*(IJ-1)/2+II
       IF(IJ.LT.II)IK=II*(II-1)/2+IJ
    85    SUM1=SUM1 + A(IK)*B(IJ,JJ)
    90    C(II,JJ)=SUM1
          RETURN
          END
```

FILE: MT3 FORTRAN A

```
C*
C*      MATRICES A AND/OR B MAY BE STORED IN SYMMETRIC NOTATION
C*        L=M IF A IS SYMMETRIC ---- N=M IF B IS SYMMETRIC
        SUBROUTINE MT3(A,B,C,L,M,N,ISYMA,ISYMB)
        DOUBLE PRECISION A(1),B(1),C(L,N),SUM
        DO 70 II=1,L
        DO 70 JJ=1,N
        SUM=0.0
        DO 65 KK=1,M
        IF(ISYMA.EQ.1)GO TO 61
        IK=L*(KK-1) + II
        GO TO 62
   61   IF(KK.GE.II)IK=KK*(KK-1)/2 + II
        IF(KK.LT.II)IK=II*(II-1)/2 + KK
   62   IF(ISYMB.EQ.1)GO TO 63
        JK=M*(JJ-1) + KK
        GO TO 65
   63   IF(KK.GE.JJ)JK=KK*(KK-1)/2 + JJ
        IF(KK.LT.JJ)JK=JJ*(JJ-1)/2 + KK
   65   SUM=SUM + A(IK)*B(JK)
   70   C(II,JJ)=SUM
        RETURN
        END
```

```
FILE: MT4      FORTRAN  A

C*
      SUBROUTINE MT4(A,B,C,L,M,N,ISYM)
C*
C*    MATRIX A CAN BE STORED FULL OR IN SYMMETRIC NOTATION
C*
C*       ISYM=1  IF A IS SYMMETRIC
C*       ISYM=0  IF A IS FULL
C*
      DOUBLE PRECISION A(1),B(M,N),C(L,N),SUM
        DO 95 II=1,L
        DO 95 JJ=1,N
        SUM=0.0
        DO 90 IJ=1,M
      IF(ISYM.EQ.1)GO TO 85
      IK=L*(IJ-1) + II
      GO TO 90
   85 IF(IJ.GE.II)IK=IJ*(IJ-1)/2 + II
      IF(IJ.LT.II)IK=II*(II-1)/2 + IJ
   90   SUM = SUM + A(IK)*B(IJ,JJ)
   95   C(II,JJ)=SUM
        RETURN
      END
```

FILE: PLOT FORTRAN A

```
      SUBROUTINE PLOT(X,Y,NOX,MAXX,ILABLX,ILABLY,ICODE,IOPT)                    PL0000010
C*****************************************************************************PL0000020
C*                                                                           *PL0000030
C*    THIS SUBROUTINE WAS WRITTEN BY J.K.DALY OF TRW FOR THE ASTEP           *PL0000040
C*    PROGRAM.  IT WAS MODIFIED SLIGHTLY FOR USE IN THIS PROGRAM BY          *PL0000050
C*    R. MINTER                                                              *PL0000060
C*                                                                           *PL0000070
C*    --ICODE AND IOPT ARE SET TO 0, AND ILABX,ILABY ARE BLANK.              *PL0000080
C*      WRITE STATEMENTS WERE ADDED FOR LABELING.                            *PL0000090
C*                                                                           *PL0000100
C*****************************************************************************PL0000110
CCC   PROGRAMMER - J.K. DALY                                                  PL0000120
CCC   DATE       - FEBRUARY, 1973                                             PL0000130
CCC MODIFIED FOR IBM 370-148 R HANSEN,C HORTON  DEC.1977                      PL0000140
CCC   INPUT VARIABLES                                                         PL0000150
CCCCC       X       - ARRAY CONTAINING X COORDINATES (TYPE REAL)              PL0000160
CCCCC       Y       - ARRAY CONTAINING Y COORDINATES (TYPE REAL)              PL0000170
CCCCC       NOX     - NO. OF X COORDINATES INPUT (MAX. 39)                    PL0000180
CCCCC       MAXX    - MAXIMUM VALUE OF POINT ON SCALE TO BE INPUT (INTEGER)   PL0000190
CCCCC       ILABLX  - LABLE OF THE X-AXIS (MAX. OF 78 CHARACTERS ALLOWED)     PL0000200
CCCCC       ILABLY  - LABLE OF THE Y-AXIS (MAX. OF 78 CHARACTERS ALLOWED)     PL0000210
CCCCC      ICODE    - 1 = RUN IS MADE ON TSS.INVALID 12-77                    PL0000220
CCCCC               0 = RUN IS MADE ON 1108                                   PL0000230
CCCCC       IOPT    - 0 = 45 DEGREE ANGLE LINE TO BE DRAWN ON PLOT            PL0000240
CCCCC               1 = NO 45 DEG ANGLE LINE.INVALID 12-77                    PL0000250
CCC   INTERNAL VARIABLES                                                      PL0000260
CCCCC       FMOX    - WHICH WORD OF THE FORMAT ARRAY(FMTARY) TO START         PL0000270
CCCCC                 LOADING THE LABELS.                                     PL0000280
CCCCC       FMTARY  - ARRAY TO BE BUILT FOR EACH LINE OF THE PLOT             PL0000290
CCCCC       I       - INDEX FOR INNER LOOP FOR DETERMINING THE LOCATION       PL0000300
CCCCC                 OF THE POINT ON THE X-AXIS                              PL0000310
CCCCC       INDEX   - MAXIMUM NO. OF WORDS OF FMTARY TO BE WRITTEN (TSS=11,   PL0000320
CCCCC                 1108=15)                                                PL0000330
CCCCC       INITLN  - INITIAL LINE TO BEGIN PRINTING THE LABELS               PL0000340
CCCCC       INUM    - NUMBER OF POINTS TO FALL IN SAME PLACE ON PLOT.         PL0000350
CCCCC       J       - INDEX FOR LOOP TO FIND NEXT HIGHEST Y-COORDINATE.       PL0000360
CCCCC       K       - INDEX FOR OUTER LOOP BUILDING AND PRINTING EACH         PL0000370
CCCCC                 LINE OF PLOT.                                           PL0000380
CCCCC       LABNDX  - INDEX FOR LABARY                                        PL0000390
CCCCC       LASTLN  - LAST LINE TO PRINT LABEL ON PLOT                        PL0000400
CCCCC       M       - INDEX OF SCLARY                                         PL0000410
CCCCC       MAXWIZ  - NO. OF CHARACTERS/PLOT                                  PL0000420
CCCCC       MLNCT   - NO. OF LINES/PLOT                                       PL0000430
CCCCC       NDX     - INDEX TO 'WICLAR'                                       PL0000440
CCCCC       NOXPT   - NO. OF POINTS TO FALL BETWEEN LABELED POINTS            PL0000450
CCCCC                 ON X-AXIS                                               PL0000460
CCCCC       NOYPT   - NO. OF POINTS TO FALL BETWEEN LABELED POINTS            PL0000470
CCCCC                 ON Y-AXIS                                               PL0000480
CCCCC       SCLARY  - ARRAY CONTAINING POINTS TO BE PRINTED AS LABELS         PL0000490
CCCCC                 ON THE X- AND Y-AXIS                                    PL0000500
CCCCC       WICLAR  - ARRAY CONTAINING LINE NO.S OF BOTH KINDS OF PLOTS.      PL0000510
CCCCC                 DEPENDING ON THE LINE NO. DETERMINES WHICH LABLE        PL0000520
CCCCC                 WILL BE PRINTED.                                        PL0000530
CCCCC       XLNVLU  - VALUE OF EACH POINT ON THE X-AXIS.  USED TO CALCULATE   PL0000540
CCCCC                 EACH POINT OF XSCLAR (X SCALE ARRAY) IN SUBROUTINE      PL0000550
CCCCC                 SCALE.                                                  PL0000560
CCCCC       XSCLAR  - ACTUAL VALUE OF EACH POINT ON X-AXIS (TYPE REAL)        PL0000570
CCCCC       YLNVLU  - VALUE OF EACH POINT ON Y-AXIS. USED TO CALCULATE EACH   PL0000580
CCCCC                 POINT OF YSCLAR (Y SCALE ARRAY)                         PL0000590
CCC * * * * *                                                                 PL0000600
CC    INCLUDE COMMA.LIST                                                      PL0000610
C     INCLUDE COMMB.LIST                                                      PL0000620
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,               PL0000630
     *              HISFIL,HISKEY,TRFORM,ERRTP,ERRKEY,MAPUNT,NOFILE,         PL0000640
     *     DRUMAD,DRUMOS,MAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    PL0000650
     *   ,NMSTUN,NMSTFI,SCTHUN,MAPFIL                                        PL0000660
     *   ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                  PL0000670
     *  CRDUNT,PRTUNT,HANDIO                                                 PL0000680
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CHIKEY,INCFET,                     PL0000690
     *     INCVEC(30),ICOUNT,SFTWGT,FVALHF(100),FFTVC4(30)                  PL0000700
     *      ,NOFET4,VARS74,COMMAS,DTAH4,WORMS14,MESTVC(10),DIVSI7           PL0000710
     *     ,STATKY,ADMESD,ADRESP,ADMESF,ADRSM1,ADRSM2                        PL0000720
      INTEGER ADMESD,ADRESP,ADMESF,ADRSM1,ADRSM2,STATKY                       PL0000730
      INTEGER HCDU,PCDX,HCORLK,HCDSTR                                         PL0000740
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                     PL0000750
CSEND                                                                         PL0000760
C                                                                             PL0000770
      DIMENSION           FMTARY(23)                                          PL0000780
C                                                                             PL0000790
```

```
FILE: PLOT     FORTRAN  A

      REAL                                YSCLAR(57),XSCLAR(85)                PL000800
      DOUBLE PRECISION X(NOX),Y(NOX)                                           PL000810
      REAL SCLARY(4)                                                           PL000820
      INTEGER CRIKEY,PRCKEY                                                    PL000830
      INTEGER LABARY(40),LABNOX,HICLAB(10),FMOX                                PL000840
      INTEGER ILAPLX(20), ILABLY(20),FMTARY,ASTRIC,O                           PL000850
C                                                                              PL000860
      LOGICAL*1 FMTA(92), LTEST(4),LSTAR,LOG1                                  PL000870
C                                                                              PL000880
      DATA LOG1//F1/, LBLANK/Z40/,LSTAR/Z5C/                                   PL000890
      DATA FMTARY/23*4H    /,ASTRIC/4H****/,IBLANK/4H    /                     PL000900
      DATA IPOINT/4H   1/                                                      PL000910
      DATA ACCBLK/Z40/,ACUSTR/Z5C/,HCD9/ZF9/,RCDX/ZE7/                         PL000920
      DATA HICLAB /55,54,52,50,49,42,41,40,39,38/                              PL000930
C                                                                              PL000940
      EQUIVALENCE (FMTA(1),FMTARY(1)),(ITEST,LTEST(1))                         PL000950
C                                                                              PL000960
C                                                                              PL000970
C  BLANK X AND Y LABELS                                                        PL000980
      DO 100 I= 1,20                                                           PL000990
      ILABLX(I)=IBLANK                                                         PL001000
  100 ILABLY(I)=IBLANK                                                         PL001010
C                                                                              PL001020
C*    FIND MAXX                                                                PL001030
      MAXX=1                                                                   PL001040
      AMAX=0.                                                                  PL001050
      IF(CRIKEY.EQ.3)GO TO 900                                                 PL001060
      IKCT=0                                                                   PL001070
C                                                                              PL001080
C  COUNT NUMBER OF X(I) GREATER THAN 700. SET AMAX = LARGEST X(I)              PL001090
C  SET MAXX=700 UNLESS MORE THAN 1/5 OF ELEMS GRTR 70.THEN MAXX=1400           PL001100
      DO 800 I=1,NOX                                                           PL001110
      IF(X(I).GT.700)IKCT=IKCT+1                                               PL001120
  800 CONTINUE                                                                 PL001130
      MAXX= 700                                                                PL001140
      IF(FLOAT(IKCT)/FLOAT(NOX) .GT. .2)MAXX=1400                              PL001150
  900 CONTINUE                                                                 PL001160
C                                                                              PL001170
C  WRITE MESSAGE 'SEPARABILITY TO BE GAINED MAP'                               PL001180
      CALL SETMRG(66,2,64)                                                     PL001190
      WRITE(6,1495)                                                            PL001200
 1495 FORMAT(50X,'SEPARABILITY-TO-BE-GAINED MAP')                              PL001210
C                                                                              PL001220
C  BLANK LABARY                                                                PL001230
      DO 1000 II=1,40                                                          PL001240
      LABARY(II) = IBLANK                                                      PL001250
 1000 CONTINUE                                                                 PL001260
C                                                                              PL001270
C  BLANK FMTARY                                                                PL001280
      DO 1001 II=1,23                                                          PL001290
 1001 FMTARY(II) = IBLANK                                                      PL001300
C                                                                              PL001310
C  STORE  X- AND Y-AXIS LABELS INTO LABEL ARRAY(LABARY)                        PL001320
      DO 1020 II=1,20                                                          PL001330
      IF (ILABLY(II).EQ.0) GO TO 1020                                          PL001340
      LABARY(II) = ILABLY(II)                                                  PL001350
      IF (ILABLX(II).EQ.0) GO TO 1020                                          PL001360
      LABARY(II+20) = ILABLX(II)                                               PL001370
 1020 CONTINUE                                                                 PL001380
C                                                                              PL001390
C                                                                              PL001400
C* * * * SET VARIABLES FOR PRINT PLOT ON 1108                                  PL001410
      LABNOX = 1                                                               PL001420
      S = 7                                                                    PL001430
      MLNCT  = 56                                                              PL001440
      MMORIZ = 45                                                              PL001450
      NOXPT  = 12                                                              PL001460
      NOYPT  = 8                                                               PL001470
      NOSPAC = 46                                                              PL001480
      INITIN = 55                                                              PL001490
      LASTLN = 49                                                              PL001500
      NOX    = 1                                                               PL001510
      FMOX   = 1                                                               PL001520
      INDEX = 23                                                               PL001530
C                                                                              PL001540
C  CALL SCALE TO FIND SCALE FOR PLOT                                           PL001550
 1040 CONTINUE                                                                 PL001560
C                                                                              PL001570
C                                                                              PL001580
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: PLOT FORTRAN A

```
      CALL SCALE(MAXX,MLNCT,INCRF,YSCLAR,XSCLAR,SCLARY,XLNVLU,YLNVLU,    PL001590
     *           MHORIZ,NOXPT,NOYPT)                                     PL001600
      XLNVLU = XLNVLU/2.0                                               PL001610
      YLNVLU = YLNVLU/2.0                                               PL001620
 1080 CONTINUE                                                          PL001630
C                                                                       PL001640
C                                                                       PL001650
C  FIND Y COORDINATE                                                    PL001660
      DO 1180 II80 = 1,MLNCT                                            PL001670
      K = MLNCT - II80 + 1                                              PL001680
      DO 1180 J=1,NOX                                                   PL001690
      YY = Y(J)                                                         PL001700
      IF(YY.GT.YSCLAR(K))GO TO 1180                                     PL001710
      IF(YY.LE.YSCLAR(K-1))GO TO 1180                                   PL001720
      IF (YY.GT.FLOAT(MAXX)) GO TO 1180                                 PL001730
 1100 CONTINUE                                                          PL001740
C                                                                       PL001750
CC                                                                      PL001760
C* * * * Y COORDINATE FOUND. NOW FIND POS. OF X COOD.                   PL001770
      DO 1160 II60 = 1,MHORIZ                                           PL001780
      I = MHORIZ - II60 + 1                                             PL001790
C  CHECK FOR X VALUE IN RANGE                                           PL001800
      XX = X(J)                                                         PL001810
      IF (XX.GT.XSCLAR(I)) GO TO 1160                                   PL001820
      IF(XX.LE.XSCLAR(I-1)) GO TO 1160                                  PL001830
      IF (XX.GT.FLOAT(MAXX)) GO TO 1180                                 PL001840
C  MOVE ITH CHAR TO ITEST, RIGHT JUSTIFIED, 0 FILLED                    PL001850
      ITEST = 0                                                         PL001860
      LTEST(4) = FMTA(I)                                                PL001870
C  CHECK FOR BLANK OR STAR IMPLYING NO PREVIOUS OCCURANCES              PL001880
      IF (ITEST .EQ. ECDBLK .OR. ITEST .EQ. ACDSTR) GO TO 1140          PL001890
C  CHECK FOR X IMPLYING MAXIMUM COUNT ALREADY REACHED                   PL001900
      IF (ITEST .EQ. HCDX) GO TO 1180                                   PL001910
C  ADD 1 TO COUNT. IF MORE THAN 9 OCCURANCES, CHG COUNT TO X            PL001920
      ITEST = ITEST + 1                                                 PL001930
      IF (ITEST .GT. ACD9) ITEST =HCDX                                  PL001940
C  STORE UPDATED COUNT BACK IN FMT ARRAY                                PL001950
      FMTA(I) = LTEST(4)                                                PL001960
      GO TO 1180                                                        PL001970
C  BLANK OR STAR, STORE 1 FOR FIRST OCCURANCE                           PL001980
 1140 FMTA(I) = LOG1                                                    PL001990
      GO TO 1180                                                        PL002000
 1160 CONTINUE                                                          PL002010
 1180 CONTINUE                                                          PL002020
C                                                                       PL002030
CC                                                                      PL002040
C  BUILD 45 DEGREE ANGLE LINE OF PLOT                                   PL002050
      IF (MOD(K,2).EQ.0) GO TO 1220                                     PL002060
      NOSPAC = NOSPAC - 3                                               PL002070
      LTEST(4) = FMTA(NOSPAC)                                           PL002080
      IF (ITEST .NE. LBLANK) GO TO 1220                                 PL002090
      FMTA(NOSPAC) = LSTAR                                              PL002100
C                                                                       PL002110
CC                                                                      PL002120
C  SET UP LABEL                                                         PL002130
 1220 CONTINUE                                                          PL002140
C  DETERMINE WHETHER OR NOT LABEL IS PRINTED                            PL002150
      IF (K.LT.LASTLN) GO TO 1340                                       PL002160
      IF (K.GT.INITLN) GO TO 1340                                       PL002170
      IF (K.NE.MICLAR(NOX)) GO TO 1340                                  PL002180
      IF (K.EQ.52.OR.K.EQ.40) GO TO 1320                                PL002190
C                                                                       PL002200
C  MOVE LABARY TO FMTARY                                                PL002210
      IJ = FMOX + 4                                                     PL002220
      IF (MOD(K,2) .EQ. 0) IJ =FMOX + 5                                 PL002230
      DO 1240 II=FMOX,IJ                                                PL002240
      FMTARY(II) = LABARY(LABNOX)                                       PL002250
      LABNOX = LABNOX + 1                                               PL002260
 1240 CONTINUE                                                          PL002270
      NOX = NOX + 1                                                     PL002280
      GO TO 1340                                                        PL002290
 1320 CONTINUE                                                          PL002300
      FMTARY(FMOX+3)=IBLANK                                             PL002310
      NOX = NOX + 1                                                     PL002320
 1340 CONTINUE                                                          PL002330
      IF (K.EQ.MLNCT) GO TO 1380                                        PL002340
      IF (MOD(K+7,8).EQ.0) GO TO 1380                                   PL002350
      WRITE (6,1540)FMTARY                                              PL002360
      GO TO 1440                                                        PL002370
```

FILE: PLOT FORTRAN A

```
 1560 FORMAT (1H ,22X,1H*,23A4)                                          PL002380
 1350 CONTINUE                                                           PL002390
C                                                                        PL002400
C  PRINT PLOT FOR 1108                                                  PL002410
      IF (M.LE.1) GO TO 1480                                             PL002420
      IF (K.EQ.MLNCT) WRITE (6,1660)FMTARY                               PL002430
 1660 FORMAT(1H ,21X,2H *,23A4)                                          PL002440
      IF (K.EQ.MLNCT) GO TO 1440                                         PL002450
      WRITE (6,1540)SCLARY(M),FMTARY                                     PL002460
 1540 FORMAT(1H ,14X,E7.2,2H *,23A4)                                     PL002470
      M = M - 1                                                          PL002480
 1440 CONTINUE                                                           PL002490
C                                                                        PL002500
C                                                                        PL002510
C  WRITE X AND Y LABELS AS SELECTED                                      PL002520
      DO 1460 II=1,INDEX                                                 PL002530
      FMTARY(II4+1)=IBLANK                                               PL002540
 1460 CONTINUE                                                           PL002550
      IF(K.NE.40)GO TO 1030                                              PL002560
      IF(PRCKEY.EQ.3 .OR. PRCKEY.EQ.4)WRITE(6,1710)                      PL002570
 1710 FORMAT(1H+,'LINEAR')                                               PL002580
      IF(PRCKEY.EQ.1 .OR. PRCKEY.EQ.2)WRITE(6,1720)                      PL002590
 1720 FORMAT(1H+,'SELECTED')                                             PL002600
 1030 IF(K.NE.30)GO TO 1035                                              PL002610
      IF(PRCKEY.EQ.3 .OR. PRCKEY.EQ.4) WRITE(6,1730)                     PL002620
 1730 FORMAT(1H+,'COMB.')                                                PL002630
      IF(PRCKEY.EQ.1 .OR. PRCKEY.EQ.2) WRITE(6,1740)                     PL002640
 1740 FORMAT(1H+,'CHANNELS')                                             PL002650
 1035 CONTINUE                                                           PL002660
 1480 CONTINUE                                                           PL002670
C                                                                        PL002680
C* * * * PRINT X-AXIS LINE AND LABELS                                    PL002690
C* * * * 110A SIZE PLOT                                                  PL002700
      WRITE (6,1580)(SCLARY(L),L=1,8)                                    PL002710
 1580 FORMAT(1H ,22X,7(12H* * * * * *  ),1H*/15X,E7.2,3X,7(5X,E7.2))     PL002720
      WRITE(6,1750)                                                      PL002730
 1750 FORMAT(/60X,'ALL CHANNELS')                                        PL002740
      IF(CRIKEY.EQ.3)WRITE(6,1690)                                       PL002750
 1690 FORMAT(/50X,   'INTERCLASS BHATTACHARYYA DISTANCE')                PL002760
      IF(CRIKEY.NE.3)WRITE(6,1700)                                       PL002770
 1700 FORMAT(/55X,   'PAIRWISE DIVERGENCE')                              PL002780
      RETURN                                                             PL002790
      END                                                                PL002800
```

FILE: PRELIM FORTRAN A

```
      SUBROUTINE PRELIM(COVMTX,AVEMTX,DIVTAB,WEIGHT,S,                  PRE00010
     *                  WRKRY,WRKSIZ)                                  PRE00020
C*                                                                      PRE00030
C*    THIS SUBROUTINE PERFORMS SOME OF THE PRELIMANARY TASKS FOR        PRE00040
C*    FEATURE SELECTION.  THE INTERCLASS MEASURES USING ALL FEATURES    PRE00050
C*    ARE COMPUTED AND STORED ON A SCRATCH FILE FOR LATER PRINTING.     PRE00060
C*                                                                      PRE00070
C*    IN ADDITION, THIS SUBROUTINE COMPUTES THE 'S' MATRICES USED IN    PRE00080
C*    COMPUTING WEIGHTED AVERAGE DIVERGENCE IF CRIKEY=1.  IF WEIGHTS    PRE00090
C*    ARE TO BE SET BY DEFAULT, THE SUBROUTINE ALSO PERFORMS THIS TASK. PRE00100
C*                                                                      PRE00110
      IMPLICIT INTEGER(A-Z)                                             PRE00120
C     INCLUDE COMMK1.LIST                                               PRE00130
C     INCLUDE COMMK7.LIST                                               PRE00140
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          PRE00150
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    PRE00160
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        PRE00170
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             PRE00180
     *              GRPCHK(61),GROUPS(124)                              PRE00190
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,               COM00010
     *    INCVEC(30),ICOUNT,SETWGT,EVALRF(100),FFIVC4(30)               COM00020
     *     ,NOFET4,VARSZ4,CORBAS,DTAB4,WGHS14,RESTVC(10),DIVSIZ         COM00030
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                    COM00040
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                 COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                               COM00060
C$END                                                                   PRE00270
      COMMON/RESTKN/ KPPPTS(60), IPRIOR, KREST, NCPASS                  PRE00280
      REAL   COVMTX(VARSZ2,NOCLS2),  AVEMTX(NOFET2,NOCLS2),            PRE00290
     *       WEIGHT(DIVSIZ), S(VARSZ2,NOCLS2)                           PRE00300
      REAL T(30),APRWGT(780),ANUMPX                                     PRE00310
      DOUBLE PRECISION DIVTAB(DIVSIZ),DUM                               PRE00320
      DOUBLE PRECISION WRKRY(1)                                         PRE00330
      REAL TW(60)                                                       PRE00340
      ANUMPX = 0.                                                       PRE00350
      DO 1 I=1,NOCLS2                                                   PRE00360
    1 ANUMPX = ANUMPX + FLOAT(KPPPTS(I))                                PRE00370
      K = 0                                                             PRE00380
      NC = NOCLS2 - 1                                                   PRE00390
      DO 2 I=1,NC                                                       PRE00400
      IK = I + 1                                                        PRE00410
      DO 2 II = IK,NOCLS2                                               PRE00420
      K = K + 1                                                         PRE00430
      APRWGT(K)=FLOAT(KPPPTS(I)*KPPPTS(II))/(ANUMPX**2)                 PRE00440
    2 APRWGT(K)=SQRT(APRWGT(K))                                         PRE00450
      IF(IPRIOR.NE.0) WRITE(6,990)                                      PRE00460
  990 FORMAT(1H1,'  APRIORI WEIGHT MULTIPLIERS AND TOTAL NO PIXELS')    PRE00470
      IF(IPRIOR.NE.0) WRITE(6,1000) (APRWGT(I),I=1,DIVSIZ),ANUMPX       PRE00480
 1000 FORMAT( )                                                         PRE00490
      IF (SETWGT.EQ.2) GO TO 6                                          PRE00500
      DO 5 I=1,DIVSIZ                                                   PRE00510
    5 WEIGHT(I) = 1.0                                                   PRE00520
    6 IF (IPRIOR.EQ.0) GO TO 9                                          PRE00530
      DO 7 I=1,DIVSIZ                                                   PRE00540
    7 WEIGHT(I) = WEIGHT(I)*APRWGT(I)                                   PRE00550
    9 CONTINUE                                                          PRE00560
C*    SET IPART SO PARTIALS WILL NOT BE COMPUTED.                       PRE00570
      IPART=-1                                                          PRE00580
      IFULL=1                                                           PRE00590
      GO TO(10,70,80,90),CRIKEY                                         PRE00600
C*                                                                      PRE00610
C*    CRITERIA - WEIGHTED AVERAGE DIVERGENCE                            PRE00620
C*    --COMPUTE INTERCLASS DIVERGENCES                                  PRE00630
C*    --SET WEIGHTS IF SETWGT=0                                         PRE00640
C*    --COMPUTE S-MATRICES                                              PRE00650
C*    --COMPUTE WEIGHTED AVERAGE DIVERGENCE FOR ALL FEATURES            PRE00660
C*                                                                      PRE00670
   10 CALL DIVERG(COVMTX,VARSZ2,AVEMTX,DIVTAB,NOCLS2,NOFET2,            PRE00680
     *            WRKRY,WRKSIZ)                                        PRE00690
      IF(SETWGT.NE.0)GO TO 25                                           PRE00700
      DO 20 K=1,DIVSIZ                                                  PRE00710
      WEIGHT(K) = DEXP(-DIVTAB(K)/16.)                                  PRE00720
   20   IF(IPRIOR.NE.0) WEIGHT(K) = WEIGHT(K)*APRWGT(K)                 PRE00730
C*    COMPUTE S-MATRICES                                                PRE00740
   25 CONTINUE                                                          PRE00750
      DO 30 J=1,NOCLS2                                                  PRE00760
      DO 30 I=1,VARSZ2                                                  PRE00770
   30 S(I,J)=0.0                                                        PRE00780
      NC=NOCLS2-1                                                       PRE00790
      DO 60 N=1,NOCLS2                                                  PRE00800
```

FILE: PRELIM FORTRAN A

```
C*
C*     SELECT ALL WEIGHTS FOR CLASS N                                    PRE00810
C*                                                                       PRE00820
      KT=0                                                              PRE00830
      K=0                                                               PRE00840
      MN=0                                                              PRE00850
      DO 35 J=1,NC                                                      PRE00860
      IJ=J+1                                                            PRE00870
      DO 35 I=IJ,NOCLS2                                                 PRE00880
      K=K+1                                                             PRE00890
      IF(J.NE.N.AND.I.NE.N)GO TO 35                                     PRE00900
      KT=KT+1                                                           PRE00910
      TW(KT)=WEIGHT(K)                                                  PRE00920
   35 CONTINUE                                                          PRE00930
      DO 50 M=1,NOCLS2                                                  PRE00940
      IF(M.EQ.N)GO TO 50                                                PRE00950
      MN=MN+1                                                           PRE00960
      DO 40 I=1,NOFET2                                                  PRE00970
   40 T(I)=AVEMTX(I,N)-AVEMTX(I,M)                                      PRE00980
      J=0                                                               PRE00990
      DO 45 I=1,NOFET2                                                  PRE01000
      DO 45 K=1,I                                                       PRE01010
      J=J+1                                                             PRE01020
   45 S(J,N)=S(J,N)+TW(MN)*(COVMTX(J,M)+T(I)*T(K))                      PRE01030
   50 CONTINUE                                                          PRE01040
   60 CONTINUE                                                          PRE01050
C*    COMPUTE CFAC                                                      PRE01060
      CFAC=0                                                            PRE01070
      DO 65 I=1,DIVSIZ                                                  PRE01080
   65 CFAC = CFAC + WEIGHT(I)                                           PRE01090
      CFAC=1./CFAC                                                      PRE01100
C*    COMPUTE AVERAGE WEIGHTED DIVERGENCE                               PRE01110
      CALL AVEDIV(TOTMSR,COVMTX,S,DUM,DUM,WRKRY,WRKSIZ,                 PRE01120
     *                   IPART,DUM,DUM,IFULL)                           PRE01130
      TOTMSR=DABS(TOTMSR)                                               PRE01140
      GO TO 95                                                          PRE01150
C*                                                                      PRE01160
C*    CRITERIA - WEIGHTED AVERAGE TRANSFORMED DIVERGENCE                PRE01170
C*                                                                      PRE01180
C*                                                                      PRE01190
   70 CALL TRNDIV(TOTMSR,COVMTX,AVEMTX,DUM,DUM,WEIGHT,DIVTAB,           PRE01200
     *                   WRKRY,WRKSIZ,IPART,DUM,DUM,IFULL)              PRE01210
      GO TO 85                                                          PRE01220
C*                                                                      PRE01230
C*    CRITERIA - BHATTACHARYYA DISTANCE                                 PRE01240
C*                                                                      PRE01250
   80 CALL BHTCHR(TOTMSR,COVMTX,AVEMTX,WEIGHT,DIVTAB,DUM,DUM,           PRE01260
     *                   WRKRY,WRKSIZ,IPART,DUM,DUM,IFULL)              PRE01270
C*    SAVE INTERCLASS WEIGHTS ON DRUM                                   PRE01280
   85 IQ=DIVSIZ*2                                                       PRE01290
      CALL RWRITE(ADRESD,DIVTAB,IQ,ISTAT)                               PRE01300
   86 IF(ISTAT.EQ.1)GO TO 86                                            PRE01310
   90 RETURN                                                            PRE01320
      END                                                               PRE01330
                                                                        PRE01340
```

FILE: PRTFLD FORTRAN A

```
      SUBROUTINE PRTFLD(COVMTX,AVEMTX,FLDMTX,VERTEX,                    PRT00010
     *                  CLSNAM,SUBNAM)                                  PRT00020
C                                                                       PRT00030
      IMPLICIT INTEGER (A-H,O-Z)                                        PRT00040
CCCCC                                                                   PRT00050
CCCCC PRINT TRAINING FIELDS AND CLASS STATISTICS                        PRT00060
CCCCC                                                                   PRT00070
CCCCC                                                                   PRT00080
CCC                                                                     PRT00090
CCC   INCLUDE COMBK1,LIST                                               PRT00100
CCC                                                                     PRT00110
      DATA LPRN/'CH('/,RPRN/')'/                                        PRT00120
CC    INCLUDE COMBK6.LIST                                               PRT00130
CC    INCLUDE COMBK7.LIST                                               PRT00140
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          PRT00150
     *              AVAR2,COVAR2,CLSID2,SUBN02,SUBOS2,FLDSV2,VERTX2,    PRT00160
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        PRT00170
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             PRT00180
     *              GRPCHK(61),GROUPS(124)                              PRT00190
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         PRT00200
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,   PRT00210
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                PRT00220
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    PRT00230
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,BMSTFL,PCHUNT,              PRT00240
     *  CRDUNT,PRTUNT,RANDIO                                            COM00010
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,               COM00020
     *    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)               COM00030
     *     ,NOFET4,VARSZ4,CORMAS,DTAB4,WGHS14,RESTVC(10),DIVSIZ          COM00040
     *    ,STATKY,ADRESO,ADRESP,ADRESF,ADRSH1,ADRSH2                    COM00050
      INTEGER ADRESO,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                 COM00060
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                               PPT00320
C$END                                                                   PRT00330
      DIMENSION COVMTX(VARSZ2,NOSUB2),AVEMTX(NOFET2,NOSUB2),            PRT00340
     *          FLDMTX(4,NOFLD2),VERTEX(2,TOTVT2),                      PRT00350
     *          CLSNAM(NOCLS2),SUBNAM(NOSUB2)                           PRT00360
      DATA ONE/1/,SCHSZ2/3600/,RCDTWO/'2'/                              PRT00370
C                                                                       PRT00380
CCCCC                                                                   PRT00390
CCCCC WRITE OUT TRAINING FIELDS                                         PRT00400
CCCCC ---------------------------------------------------------------   PRT00410
C                                                                       PRT00420
      CALL WRTFLD(FLDMTX,VERTEX,NOFLD2,1,CLSNAM,SUBNAM)                 PRT00430
C                                                                       PRT00440
CCCCC                                                                   PRT00450
CCCCC PRINT THE COVARIANCE AND MEAN                                     PRT00460
CCCCC ---------------------------------------------------------------   PRT00470
C                                                                       PRT00480
  210 IF (STATKY.EQ.0) GO TO 300                                        PRT00490
C                                                                       PRT00500
      CNT = 7+(5+3+2*NOFET2)*((NOFET2+11)/12)                           PRT00510
      CNT = PAGSIZ/CNT                                                  PRT00520
      INC = CNT                                                         PRT00530
C                                                                       PRT00540
      DO 290 ICLAS = 1,NOSUB2                                           PRT00550
      IF (INC.LT.CNT) GO TO 220                                         PRT00560
      WRITE (6,HEAD)                                                    PRT00570
      INC = 0                                                           PRT00580
  220 WRITE(6,230) SUBNAM(ICLAS)                                        PRT00590
  230 FORMAT(/1X,'SUBCLASS ',A4 )                                       PRT00600
      DO 240 LOC=1,NOFET2,12                                            PRT00610
      STOP = LOC+11                                                     PRT00620
      IF( STOP .GT. NOFET2 ) STOP = NOFET2                              PRT00630
      WRITE(6,250)(LPRN,FETVC2(I),RPRN,I=LOC,STOP)                      PRT00640
  240 WRITE (6,260)(AVEMTX(I,ICLAS),I=LOC,STOP)                         PRT00650
  250 FORMAT(10X,12(A3,I2,A1,3X))                                       PRT00660
  260 FORMAT('0MEAN',3X,12F9.2)                                         PRT00670
      WRITE (6,280)                                                     PRT00680
  280 FORMAT('0 COVARIANCE MATRIX')                                     PRT00690
      CALL WRTMTX(COVMTX(1,ICLAS),NOFET2,RCDTWO)                        PRT00700
      INC = INC+1                                                       PRT00710
  290 CONTINUE                                                          PRT00720
C                                                                       PRT00730
  300 CONTINUE                                                          PRT00740
      RETURN                                                            PRT00750
       END
```

FILE: SCALE FORTRAN A

```
      SUBROUTINE SCALE(MAXX,MLNCT,INCRE,YSCLAR,XSCLAR,SCLARY,XLNVLU,      SCA00010
     *          YLNVLU,MHORIZ,NOXPT,NOYPT)                                SCA00020
C     INCLUDE COMMON7.LIST                                                SCA00030
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                 SCA00040
     *    INCVFC(30),ICOUNT,SETWGT,EVALRF(100),FETVC4(30)                 SCA00050
     *     ,NOFET4,VARS74,CORRAS,DTAH4,WGHS14,RESTVC(10),DIVSIZ           SCA00060
     *     ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                     SCA00070
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                   SCA00080
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                 SCA00090
C*END                                                                     SCA00100
      INTEGER CRIKEY                                                      SCA00110
      REAL YSCLAR(MLNCT), XSCLAR(MHORIZ),XLNVLU,YLNVLU,RSAVE,XPTS,YPTS    SCA00120
      DIMENSION SCLARY(8)                                                 SCA00130
C* * * * ZERO OUT SCALE LABLE ARRAY - SCLARY                              SCA00140
      DO 1000 I=1,8                                                       SCA00150
      SCLARY(I) = 0                                                       SCA00160
 1000 CONTINUE                                                            SCA00170
      IF(CRIKEY.EQ.3 .OR. CRIKEY.EQ.4)GO TO 1070                          SCA00180
C* * * * DETERMINE LABELS FOR X AND Y AXIS                                SCA00190
      INCRE = MAAX/7                                                      SCA00200
      IF (MOD(INCRE,5) .NE. 0) INCRE = 5 - MOD(INCRE,5) + INCRE           SCA00210
      ISAVE = INCRE                                                       SCA00220
      DO 1020 I=2,8                                                       SCA00230
      SCLARY(I) = ISAVE                                                   SCA00240
      ISAVE = INCRE + ISAVE                                               SCA00250
 1020 CONTINUE                                                            SCA00260
C* * * * DETERMINE THE VALUE OF EACH POINT ON THE X AND Y AXIS            SCA00270
      XPTS   = NOXPT                                                      SCA00280
      YPTS   = NOYPT                                                      SCA00290
      RSAVE  = INCRE                                                      SCA00300
      XLNVLU = RSAVE/XPTS                                                 SCA00310
      YLNVLU = RSAVE/YPTS                                                 SCA00320
      YSCLAR(1) = 0.0                                                     SCA00330
      XSCLAR(1) = 0.0                                                     SCA00340
      RSAVE = YLNVLU                                                      SCA00350
      DO 1040 I=2,MLNCT                                                   SCA00360
      YSCLAR(I) = RSAVE                                                   SCA00370
      RSAVE = RSAVE + YLNVLU                                              SCA00380
 1040 CONTINUE                                                            SCA00390
      RSAVE = XLNVLU                                                      SCA00400
      DO 1060 I=2,MHORIZ                                                  SCA00410
      XSCLAR(I) = RSAVE                                                   SCA00420
      RSAVE = RSAVE + XLNVLU                                              SCA00430
 1060 CONTINUE                                                            SCA00440
      RETURN                                                              SCA00450
 1070 CONTINUE                                                            SCA00460
      XPTS=NOXPT                                                          SCA00470
      YPTS=NOYPT                                                          SCA00480
      IEXP=7                                                              SCA00490
      DO 1075 I=1,8                                                       SCA00500
      SCLARY(I) = 1. / 10.**IEXP                                          SCA00510
 1075 IEXP = IEXP-1                                                       SCA00520
      K=0                                                                 SCA00530
      DO 1080 I=1,7                                                       SCA00540
      K=K+1                                                               SCA00550
      RNCRE = (SCLARY(I+1) - SCLARY(I))/YPTS                              SCA00560
      YSCLAR(K)=SCLARY(I)                                                 SCA00570
      DO 1080 J=2,NOYPT                                                   SCA00580
      K=K+1                                                               SCA00590
 1080 YSCLAR(K)=YSCLAR(K-1) + RNCRE                                       SCA00600
      K=0                                                                 SCA00610
      DO 1090 I=1,7                                                       SCA00620
      K=K+1                                                               SCA00630
      XSCLAR(K)=SCLARY(I)                                                 SCA00640
      RNCRE = (SCLARY(I+1)-SCLARY(I))/XPTS                                SCA00650
      DO 1090 J=2,NOXPT                                                   SCA00660
      K=K+1                                                               SCA00670
      XSCLAR(K)=XSCLAR(K-1)+RNCRE                                         SCA00680
 1090 CONTINUE                                                            SCA00690
      RETURN                                                              SCA00700
      END                                                                 SCA00710
```

FILE: SETUP4 FORTRAN A

```
      SUBROUTINE SETUP4(ARRAY,TOP,STOPFG,JTIME,SUBRAY,SUBSIZ)                SET00010
C                                                                            SET00020
      IMPLICIT INTEGER (A-H,O-Z)                                             SET00030
CI-----------------------------------------------------------------------ISET00040
CCI----------------------------------------------------------------------ISET00050
CCCI                                                                      ISET00060
CCCI                                                                      ISET00070
CCCI    PURPOSE.. COORDINATES ROUTINES TO ANALYZE SUPERVISOR              ISET00080
CCCI              CONTROL CARDS FOR 'SELECT' PROCESSOR                    ISET00090
CCCI                                                                      ISET00100
CCCI    RETURNS.. SUPERVISOR INFORMATION  AND REDUCED STATISTICS          ISET00110
CCCI----------------------------------------------------------------------ISET00120
CCI-----------------------------------------------------------------------ISET00130
CI------------------------------------------------------------------------ISET00140
CCCCCCC                                                                   SET00150
CCCCCCC                                                                   SET00160
CCCCCCC                                                                   SET00170
CCCCCCC     ------------------------------------------------------------- SET00180
CCCCCCC                                                                   SET00190
CCCCCCC                                                                   SET00200
CCCCCCC     INCLUDE COMBK1.LIST                                           SET00210
C           INCLUDE COMBK4.LIST                                           SET00220
      DIMENSION CODVEC(23),ACARD(20)                                      SET00230
      DIMENSION WGHBUF(400),NUMVEC(30),COMVEC(2)                          SET00240
      DIMENSION EQUVEC(2)                                                 SET00250
      DATA EQUVEC/1,'='/                                                  SET00260
      DATA CODMAX /23/                                                    SET00270
      DATA WSIZ/400/,ESIZ/100/,BLANK/1H /,COMVEC/1,','/                   SET00280
      DATA SYMMAX/60/                                                     SET00290
      DATA CODVEC/ 'CHAN','SUBC','REST','OPTI','WEIG','EVAL','MODU',      SET00300
     2     'GROU','PROC','CRIT','B-MA','INCL','ICOU','DATE','HED1',      SET00310
     3     'HED2','COMM','*END','$END','STAT','APRI','BSPA','NCPA'/      SET00320
C     INCLUDE COMBK6.LIST                                                 SET00330
C     INCLUDE COMBK7.LIST                                                 SET00340
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,            SET00350
     *              AVAR2,COVAR2,CLSID2,SUBN02,SUBOS2,FLDSV2,VERTX2,      SET00360
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),          SET00370
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),               SET00380
     *              GRPCHK(61),GROUPS(124)                                SET00390
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                      SET00400
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                   SET00410
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               SET00420
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,           SET00430
     *              HISFIL,HISKEY,TRFORM,SRIPTP,ERPKEY,MAPUNT,NOFILE,     SET00440
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  SET00450
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                      SET00460
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                SET00470
     *  CRDUNT,PRTUNT,RANDIO                                              SET00480
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                 SET00490
     *    INCVEC(30),ICOUNT,SETWGT,EVALAF(100),FETVC4(30)                 SET00500
     *     ,NOFET4,VARSZ4,CORHAS,DTAB4,WGHS14,BESTVC(10),DIVSIZ           SET00510
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                      SET00520
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                   SET00530
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                 SET00540
CSEND                                                                     SET00550
      COMMON/RESTKN/ KPPPTS(60), IPRIOR, KBEST, NCPASS                    SET00560
      DIMENSION PROC(3,6),CRI(3,3)                                        SET00570
      DATA CRI/'DIVE','RGEN','CE  ','TRAN','S. D','IV. ','RHAT','T. D',  SET00580
     *'IST.'/                                                             SET00590
      DATA PROC/'EX. ',' SEA','RCH ','WITH','OUT ','RPLC',               SET00600
     * 'DAVI','DON ','    ','EVL ','B MA','TRIX','EVAL','UATE','    ',  SET00610
     * 'REST',' K  ','PASS'/                                             SET00620
      DIMENSION ARRAY(1),CARD(62)                                         SET00630
      DIMENSION SUBRAY(1)                                                 SET00640
      REAL SUBRAY                                                         SET00650
      DATA CRCD/'C'/, SBCD/'S'/, PBCD/'P'/, UBCD/'U'/, FBCD/'F'/          SET00660
C                                                                         SET00670
CCCCCCC                                                                   SET00680
CCCCCCC                                                                   SET00690
CCCCCCC     ------------------------------------------------------------- SET00700
CCCCCCC                                                                   SET00710
CCCCCCC                                                                   SET00720
C           ------------------------------------------------------------- SET00730
      NOGRP=0                                                             SET00740
C     ------------------------------------------------------------------- SET00750
      INCFET=0                                                            SET00760
      PRCKEY=2                                                            SET00770
      NOFET2=0                                                            SET00780
      WPTR=0                                                              SET00790
```

FILE: SETUP4 FORTRAN A

```
      EPTR=0                                                                    SET00800
      GRPTR=0                                                                   SET00810
      ICOUNT=300                                                                SET00820
      STOPFG=0                                                                  SET00830
      NOCLS2=0                                                                  SET00840
      NOSUB2=0                                                                  SET00850
      DATSWT=0                                                                  SET00860
      SETWGT=0                                                                  SET00870
      WTKEY = 0                                                                 SET00880
      NOGRP=0                                                                   SET00890
      IPRIOR = 0                                                                SET00900
      NCPASS=4                                                                  SET00910
      DO 2 I=1,61                                                               SET00920
      GRPCHK(I)=0                                                               SET00930
    2 CONTINUE                                                                  SET00940
      DO 5 I=1,10                                                               SET00950
    5 RESTVC(I)=0                                                               SET00960
      CRTKEY=1                                                                  SET00970
      NOFET2=0                                                                  SET00980
      PRUNIT = 30                                                               SET00990
      IF(JTIME.EQ.1)GO TO 10                                                    SET01000
      WRITE(6,HEAD)                                                             SET01010
      WRITE(6,1000)                                                             SET01020
C*                                                                              SET01030
C*    SETUP REREAD BUFFER                                                       SET01040
C*                                                                              SET01050
      CALL REREAD(PRUNIT,80)                                                    SET01060
C*    NOW READ CARD INTO THE BUFFER                                             SET01070
   10 READ (2,15)(ACARD(I),I=1,20)                                              SET01080
   15 FORMAT(20A4)                                                              SET01090
      WRITE(PRUNIT,15)(ACARD(I),I=1,20)                                         SET01100
      REWIND PRUNIT                                                             SET01110
      READ(PRUNIT,2000)CODE,CARD                                                SET01120
      REWIND PRUNIT                                                             SET01130
      WRITE(6,3000)CODE,CARD                                                    SET01140
      COL=0                                                                     SET01150
      DO 20 I=1,CODMAX                                                          SET01160
      IF(CODVEC(I).EQ.CODE) GO TO (30,40,50,60,70,80,100,120,130,140,           SET01170
     *150,160,170,180,190,200,210,220,350,215,205,207,208),I                    SET01180
   20 CONTINUE                                                                  SET01190
      WRITE(6,8000)                                                             SET01200
      WRITE(6,8001)ACARD                                                        SET01210
 8001 FORMAT(10A4)                                                              SET01220
      GO TO 10                                                                  SET01230
C*                                                                              SET01240
C*    CHANNELS CARD                                                             SET01250
C*    FEATURES CARD - IF B-MATRIX IS INPUT IGNORE THIS CARD                     SET01260
   30 IF(BMSWT.EQ.1)GO TO 10                                                    SET01270
      NOFET2=NUMBER(CARD,COL,FETVC2,NOFET2)                                     SET01280
      CALL ORDER(FETVC2,NOFET2)                                                 SET01290
      GO TO 10                                                                  SET01300
C*                                                                              SET01310
C*    SUBCLASSES CARD                                                           SET01320
C*                                                                              SET01330
   40 NOSUB2=NUMBER(CARD,COL,SUBVC2,NOSUB2)                                     SET01340
      CALL ORDER(SUBVC2,NOSUB2)                                                 SET01350
      GO TO 10                                                                  SET01360
C*                                                                              SET01370
C*    REST CARD                                                                 SET01380
C*                                                                              SET01390
   50 NOREST=NUMBER(CARD,COL,RESTVC,NOREST)                                     SET01400
      GO TO 10                                                                  SET01410
C*                                                                              SET01420
C*    OPTION CARD - OPTIONS ARE STATS OR PUNCH                                  SET01430
C*                                                                              SET01440
   60 J=NXTCHR(CARD,COL)                                                        SET01450
      IF(J.EQ.CRCD) WTKEY=1                                                     SET01460
      IF(J.EQ.SRCD) STATKY=1                                                    SET01470
      IF(J.EQ.PRCD) PCHKEY=1                                                    SET01480
      IF(J.EQ.BLANK)GO TO 10                                                    SET01490
      J=FIND12(CARD,COL,COMVEC)                                                 SET01500
      IF(J.EQ.-1)GOTO 10                                                        SET01510
      GO TO 60                                                                  SET01520
C*                                                                              SET01530
C*    WEIGHTS CARD                                                              SET01540
C*                                                                              SET01550
   70 CALL WGTSCN(CARD,COL,SUBRAY,WGHBUF,WSIZ,WPTR)                             SET01560
      SETWGT=2                                                                  SET01570
      GO TO 10                                                                  SET01580
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SETUP4 FORTRAN A

```
C*                                                                      SET01590
C*     EVALUATE CARD - A SET OF FEATURES TO BE EVALUATED ACCORDING TO   SET01600
C*                     REQUESTED CRITERIA                               SET01610
C*                                                                      SET01620
C*                                                                      SET01630
   80 IF(EPTR.GE.ESIZ)GO TO 10                                          SET01640
      J=NUMBER(CARD,COL,NUMVEC,0)                                       SET01650
      EPTR=EPTR+1                                                       SET01660
      EVALRF(EPTR)=J                                                    SET01670
      IF(EPTR+2+J  .LE.ESIZ)GO TO 85                                    SET01680
      WRITE(6,4000)                                                     SET01690
      EPTR=ESIZ                                                         SET01700
      GO TO 10                                                          SET01710
   85 DO 90 L=1,J                                                       SET01720
      EPTR = EPTR+1                                                     SET01730
      EVALRF(EPTR)=NUMVEC(L)                                            SET01740
   90 CONTINUE                                                          SET01750
      GO TO 10                                                          SET01760
C*                                                                      SET01770
C*     MODULE CARD DECK - READ CARD DECK AND STORE ON STAT FILE.         SET01780
C*                                                                      SET01790
  100 CALL CRDSTA(ARRAY,TOP)                                            SET01800
      GO TO 10                                                          SET01810
C*                                                                      SET01820
C*     GROUP CARD                                                       SET01830
C*                                                                      SET01840
  120 J=GRPSCN(CARD,SYMMAX,GRPTR)                                       SET01850
      IF(J.EQ.0)GO TO 10                                                SET01860
      WRITE(6,5000)                                                     SET01870
      GO TO 10                                                          SET01880
C*     PROCEDURE CARD   1=EXHAUSTIVE SEARCH                             SET01890
C*                      2=WITHOUT REPLACEMENT (DEFAULT)                 SET01900
C*                      3=DAVIDON                                       SET01910
C*                      4=USER INPUT                                    SET01920
C*                      5=EVALUATE FEATURE CHANNELS                     SET01930
C*                                                                      SET01940
  130 J=NUMBER(CARD,COL,NUMVEC,0)                                       SET01950
      PROKEY=NUMVEC(1)                                                  SET01960
      GO TO 10                                                          SET01970
C*     CRITERIA CARD    1=AVERAGE WEIGHTED DIVERGENCE                   SET01980
C*                      2=TRANSFORMED DIVERGENCE                        SET01990
C*                      3=BHATTACHARYYA                                 SET02000
C*                                                                      SET02010
C*                                                                      SET02020
  140 J=NUMBER(CARD,COL,NUMVEC,0)                                       SET02030
      CRIKEY=NUMVEC(1)                                                  SET02040
      IF (CRIKEY .LT. 1 .OR. CRIKEY .GT. 3) CRIKEY = 1                  SET02050
      GO TO 10                                                          SET02060
C*                                                                      SET02070
C*     B-MATRIX - CARDS OR FILE                                         SET02080
C*              - BMSWT=1 MEANS B-MATRIX INPUT AND ON FILE,NOT IN CORE  SET02090
C*                                                                      SET02100
  150 J=NXTCHR(CARD,COL)                                                SET02110
      BMSWT=1                                                           SET02120
      BMKEY=1                                                           SET02130
      IF(J.NE.CHCO) GO TO 10                                            SET02140
      CALL BMFIL (ARRAY,NOFET4,NOFET2,FETVC2,1)                         SET02150
C*    SET DATSWT BACK TO ZERO TO INDICATE STATS MAY HAVE BEEN OVERWRITEN SET02160
      DATSWT=0                                                          SET02170
      GO TO 10                                                          SET02180
C*                                                                      SET02190
C*     INCLUDE CARD - FEATURES TO BE INCLUDED IN 'REST' SET,            SET02200
C*                    WITHOUT REPLACEMENT PROCEDURE                     SET02210
C*                                                                      SET02220
  160 INCFET= NUMBER(CARD,COL,INCVEC,INCFET)                            SET02230
      GO TO 10                                                          SET02240
C*                                                                      SET02250
C*     ICOUNT CARD - NO. OF ITERATIONS FOR DAVIDON PROCEDURE            SET02260
C*                                                                      SET02270
  170 J=NUMBER(CARD,COL,NUMVEC,0)                                       SET02280
      ICOUNT=NUMVEC(1)                                                  SET02290
      GO TO 10                                                          SET02300
C*                                                                      SET02310
C*     DATE CARD                                                        SET02320
  180 READ(30,6000)DATE                                                 SET02330
      REWIND RRUNIT                                                     SET02340
      GO TO 10                                                          SET02350
C*                                                                      SET02360
C*     HED1 CARD                                                        SET02370
  190 READ(30,6000)HED1
```

FILE: SETUP4 FORTRAN A

```
      REWIND RRUNIT                                                     SET02380
      GO TO 10                                                          SET02390
C*                                                                      SET02400
C*    HED2 CARD                                                         SET02410
  200 READ(30,6000)HED2                                                 SET02420
      REWIND RRUNIT                                                     SET02430
      GO TO 10                                                          SET02440
C     APRIORI CARD                                                      SET02450
  205 IPRIOR = 1                                                        SET02460
      GO TO 10                                                          SET02470
  207 J = NUMBER(CARD,COL,NUMVEC,0)                                     SET02480
      KBEST = NUMVEC(1)                                                 SET02490
      GO TO 10                                                          SET02500
  208 J=NUMBER(CARD,COL,NUMVEC,0)                                       SET02510
      NCPASS=NUMVEC(1)                                                  SET02520
      GO TO 10                                                          SET02530
C*                                                                      SET02540
C*    COMMENT CARD                                                      SET02550
  210 READ(30,6000)COMENT                                               SET02560
      REWIND RRUNIT                                                     SET02570
      GO TO 10                                                          SET02580
C*                                                                      SET02590
C*    STAT FILE NO.                                                     SET02600
C*                                                                      SET02610
  215 M = NXTCHR(CARD ,COL)                                             SET02620
      IF(M.EQ.BLANK) GO TO 10                                           SET02630
      IF(M.EQ.UPCD) GO TO 1702                                          SET02640
      IF(M.EQ.FRCD)) GO TO 1703                                         SET02650
 1723 WRITE(6,755)                                                      SET02660
  755 FORMAT(' ERROR ON STAT FILE CARD ')                               SET02670
      GO TO 10                                                          SET02680
 1702 J=FIND12(CARD ,COL,EQUVEC)                                        SET02690
      IF(J.EQ.-1) GO TO 1723                                            SET02700
      M=NUMBER(CARD ,COL,SAVTAP,ZERO)                                   SET02710
      COL=COL-1                                                         SET02720
      GO TO 215                                                         SET02730
 1703 J=FIND12(CARD ,COL,EQUVEC)                                        SET02740
      IF(J.EQ.-1) GO TO 1723                                            SET02750
      FILNO = NUMBER(CARD ,COL,STAFIL,FILNO)                            SET02760
      STAFIL = STAFIL - 1                                               SET02770
      COL=COL-1                                                         SET02780
      GO TO 215                                                         SET02790
C*                                                                      SET02800
C*    *END*  -  END OF THIS SET OF CONTROL CARDS                        SET02810
C*           -  GET STATS AND FETVEC INTO CORE                          SET02820
  220 CONTINUE                                                          SET02830
C*                                                                      SET02840
C*    IF B-MATRIX IS INPUT, OBTAIN DIMENSIONING INFORMATION AND         SET02850
C*       FETVC2 FROM BMFILE.                                            SET02860
  225 IF(BMSWT.EQ.0)GO TO 230                                           SET02870
      CALL BMFIL (DUMMY,NOFET4,NOFET2,FETVC2,3)                         SET02880
C*                                                                      SET02890
C*    READ AND REDUCE STATS                                             SET02900
C*                                                                      SET02910
  230 CALL REDSAV(ARRAY,TOP,BMSWT)                                      SET02920
C*                                                                      SET02930
C*    CHECK 'BEST' REQUESTS                                             SET02940
C*                                                                      SET02950
  260 IF(NOBEST.EQ.0)GO TO 270                                          SET02960
      CALL BSTCHK(NOBEST)                                               SET02970
      IF(NOBEST.GT.0)GO TO 280                                          SET02980
  270 NOBEST=1                                                          SET02990
      BESTVC(NOBEST)=0                                                  SET03000
C*                                                                      SET03010
C*    CHECK ON EVALUATE REQUESTS                                        SET03020
C*                                                                      SET03030
  280 CONTINUE                                                          SET03040
      EPTR=EPTR+1                                                       SET03050
      IF(PRCKEY.EQ.6) BESTVC(1) = KBEST                                 SET03060
      EVALBF(EPTR)=0                                                    SET03070
C*                                                                      SET03080
C*    PRINT USER REQUESTS                                               SET03090
      WRITE(6,9300)(PROC(M,PRCKEY),M=1,3),                              SET03100
     * (CRI(M,CRIKEY),M=1,3)                                           SET03110
      WRITE(6,9310)(BESTVC(I),I=1,NOBEST)                               SET03120
      WRITE(6,9320)(FETVC2(I),I=1,NOFET2)                               SET03130
      IF(INCFET.GT.0)WRITE(6,9330)(INCVEC(I),I=1,INCFET)                SET03140
      IF(SETWGT.EQ.2)WRITE(6,9340)                                      SET03150
      IF(WTKEY.EQ.1) WRITE(6,9360)                                      SET03160
```

FILE: SETUP4 FORTRAN A

```
      IF(SETWGT.NE.2.AND.WTKEY.NE.1)WRITE(6,9350)                       SET03170
      IF (IPRIOR.NE.0) WRITE(6,9370)                                    SET03180
      WRITE (6,9340) NCPASS                                             SET03190
C*                                                                      SET03200
C*    PRINT OUT SAVED TRAINING FIELDS AND REDUCED COVARIANCES.          SET03210
C*                                                                      SET03220
      CALL PRTFLD(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(FLDSV2),             SET03230
     *            ARRAY(VERTX2),ARRAY(CLSID2),ARRAY(SUBDS2))            SET03240
C*                                                                      SET03250
C*                                                                      SET03260
C*    IF CLSWT OPTION IS INPUT, SET UP WEIGHT ARRAY FOR INTERCLASS      SET03270
C*     SUBCLASS WEIGHTS.                                                SET03280
C*                                                                      SET03290
      W1 = 1 + WPTR * 4                                                 SET03300
  285 IF(WTKEY.EQ.1) GO TO 9500                                         SET03310
C*    STORAGE FOR FIELD INFORMATION NO LONGER NEEDED.                   SET03320
C*    MOVE CLASS ID INFORMATION                                         SET03330
C*    AND MEANS AND COVARIANCES.                                        SET03340
C*                                                                      SET03350
C*     ---REDSAV STORES INTO ARRAY IN THE FOLLOWING ORDER               SET03360
C*               1.CLASS DESCRIPTIONS                                   SET03370
C*               2.NO. OF SUBCLASSES IN EACH CLASS                      SET03380
C*               3.SUBCLASS DESCRIPTIONS                                SET03390
C*               4.TRAINING FIELD INFO.                                 SET03400
C*               5.TRAINING FIELD VERTICES                              SET03410
C*               6.COVARIANCE MATRICES                                  SET03420
C*               7.MEAN VECTORS                                         SET03430
C*                                                                      SET03440
C*     ITEMS 1,2,4,5 ARE NOT NEEDED AFTER PRINTING, SO THE STORAGE      SET03450
C*     IS REASSIGNED AS FOLLOWS. ADDING STORAGE FOR OTHER ARRAYS.       SET03460
C*               1. SUBCLASS DESCRIPTIONS                               SET03470
C*               2. COVARIANCE MATRICES                                 SET03480
C*               3. MEAN VECTORS                                        SET03490
C*               4. INTER(SUB)CLASS WEIGHTS                             SET03500
C*               5. INTER(SUB)CLASS SEPARABILITY MEASURE TABLE          SET03510
C*     IN SELECT DRIVER 'ARRAY' IS ALSO USED TO STORE ADDITIONAL INFO.   SET03520
C*                                                                      SET03530
  287 CONTINUE                                                          SET03540
      DO 290 I=1,NOSUB2                                                 SET03550
  290 ARRAY(CLSID2+I-1) = ARRAY(SUBDS2+I-1)                             SET03560
      SUBDS2=CLSID2                                                     SET03570
      NA=SUBDS2 + NOSUB2                                                SET03580
      IWPOS = (VARSZ2+NOFET2)*NOSUB2                                    SET03590
      DO 295 I=1,IWPOS                                                  SET03600
  295 ARRAY(NA+I-1) = ARRAY(COVAR2+I-1)                                 SET03610
      COVAR2=NA                                                         SET03620
      AVAR2=COVAR2 + NOSUB2*VARSZ2                                      SET03630
C*                                                                      SET03640
C*    FROM HTER ON THROUGH SELECT SUBLCASSES ARE REFERRED TO AS CLASSES SET03650
C*                                                                      SET03660
      NOCLS2=NOSUB2                                                     SET03670
C*                                                                      SET03680
C*    COMPUTE BASES FOR OTHER ARRAYS.                                   SET03690
C*                                                                      SET03700
      DIVSIZ=NOCLS2*(NOCLS2-1)/2                                        SET03710
      WGHS14=AVAR2  + NOFET2*NOCLS2                                     SET03720
      DTAB4 =WGHS14 + DIVSIZ                                            SET03730
      CORRAS = DTAB4 + DIVSIZ*2                                         SET03740
      IC=CORRAS                                                         SET03750
      IF(CORRAS .LT.TOP)GO TO 300                                       SET03760
      WRITE(6,9100)IC                                                   SET03770
      CALL CMERR                                                        SET03780
C*                                                                      SET03790
C*    SET UP ARRAY OF INTERCLASS WEIGHTS IF INPUT - IF DEFAULT IS TAKEN SET03800
C*    WEIGHTS ARE COMPUTED IN PRELIM.                                   SET03810
C*                                                                      SET03820
  300 IF(SETWGT.NE.2.AND.WTKEY.NE.1) GO TO 310                          SET03830
C*    PASS KEYS TO SUBROUTINE IN ALREADY EXISTING STORAGE               SET03840
      ARRAY(WGHS14) = SETWGT                                            SET03850
      NT = WGHS14 + 1                                                   SET03860
      ARRAY(NT) = WTKEY                                                 SET03870
      CALL WGTCHK(ARRAY(WGHS14),ARRAY(CLSID2),SUBRAY,WGHBUF,WPTR,       SET03880
     *            SUBRAY(W1),NOCLS2)                                    SET03890
      SETWGT = 2                                                        SET03900
  310 CONTINUE                                                          SET03910
      RETURN                                                            SET03920
C*                                                                      SET03930
C*    SEND*  CARD                                                       SET03940
  350 STOPFG=1                                                          SET03950
```

FILE: SETUP4 FORTRAN A

```
      RETURN                                                                    SET03960
 1000 FORMAT(' $SELECT')                                                        SET03970
 2000 FORMAT(A4,6X,62A1)                                                        SET03980
 3000 FORMAT(5X,A4,6X,62A1)                                                     SET03990
 4000 FORMAT(' TOO MANY EVALUATE REQUESTS--REMAINDER IGNORED')                  SET04000
 5000 FORMAT(' GROUP CARD IN ERROR - IGNORED')                                  SET04010
 6000 FORMAT(10X,15A4)                                                          SET04020
 7000 FORMAT(' PROGRAM CANNOT PROCESS LESS THAN 2 CHANNELS')                    SET04030
 8000 FORMAT(' INVALID CONTROL CARD - IGNORED')                                 SET04040
 9000 FORMAT(' PROGRAM CANNOT PROCESS LESS THAN 2 CLASSES')                     SET04050
 9100 FORMAT(' CORE NEEDED IN ARRAY FOR THIS PROBLEM IS',I6,' WORDS')           SET04060
 9200 FORMAT(' ERROR IN ATTEMPT TO READ STATISTICS FILE-EXECUTION TERMINSET04070
     *ATED FROM SETUP4')                                                        SET04080
 9300 FORMAT(///' YOU HAVE SELECTED THE FOLLOWING OPTIONS:'/                    SET04090
     *    5X, 'PROCEDURE',T35,3A4/ 5X,'CRITERIA',T35,3A4)                       SET04100
 9310 FORMAT(5X,'SELECT THE BEST SET(S) OF',T35,10(I2,','))                     SET04110
 9320 FORMAT(5X,'FROM CHANNELS',T35,30(I2,','))                                 SET04120
 9340 FORMAT(5X,'USE INPUT WEIGHTS')                                            SET04130
 9350 FORMAT(5X,'USE DEFAULT WEIGHTS')                                          SET04140
 9370 FORMAT(5X,'INCLUDE IN THE BEST SET, CHANNELS',30(I2,','))                 SET04150
 9360 FORMAT(5X,'USE AUTOMATIC INTERCLASS SUBCLASS WEIGHTS')                    SET04160
 9370 FORMAT(5X,'USE APRIORI WEIGHTING TO MODIFY INTERSUBCLASS WTS')            SET04170
 9380 FORMAT(5X, 'NUMBER CHANNELS PER PASS IS',T35,I5)                          SET04180
C*    INITIALIZE ALL SUBCLASS WEIGHT PAIRS TO 0.0 IN WORKING ARRAY             SET04190
 9500 CONTINUE                                                                  SET04200
      DO  2100  IK=1,NOSUB2                                                     SET04210
      DO  2100  JK=1,NOSUB2                                                     SET04220
      IOUM=(W1*IK-1*(JK-1)*NOSUB2)                                              SET04230
      IIOUM=(W1*JK-1*(IK-1)*NOSUB2)                                             SET04240
      SUBRAY(IOUM)=0.0                                                          SET04250
 2100 SUBRAY(IIOUM)=0.0                                                         SET04260
C*                                                                             SET04270
C*    REPLACE INTERCLASS SUBCLASS PAIRS WITH WEIGHT = 1.0                      SET04280
C*                                                                             SET04290
      END1 = 0                                                                  SET04300
      NK = NOCLS2 - 1                                                           SET04310
      DO  2200  KI=1,NK                                                         SET04320
      JJ1=ARRAY(SUBNO2-1*KI)                                                    SET04330
      START1 = END1 + 1                                                         SET04340
      END1 = START1 + JJ1 - 1                                                   SET04350
      DO  2300  I=START1,END1                                                   SET04360
      END2 = END1                                                               SET04370
      N = KI + 1                                                                SET04380
      DO 400 K2 = N,NOCLS2                                                      SET04390
      JJ2=ARRAY(SUBNO2-1*K2)                                                    SET04400
      START2 = END2 + 1                                                         SET04410
      END2 = START2 + JJ2 - 1                                                   SET04420
      DO 500 N = START2,END2                                                    SET04430
      KOUM=(W1-1*I+(N-1)*NOSUB2)                                                SET04440
      KKOUM=(W1-1*N+(I-1)*NOSUB2)                                               SET04450
      SUBRAY(KOUM)=1.0                                                          SET04460
      SUBRAY(KKOUM)=1.0                                                         SET04470
  500 CONTINUE                                                                  SET04480
  400 CONTINUE                                                                  SET04490
 2300 CONTINUE                                                                  SET04500
 2200 CONTINUE                                                                  SET04510
      GO TO 287                                                                 SET04520
       END                                                                      SET04530
```

FILE: TRACE FORTRAN A

```
      FUNCTION TRACE(A,B,N)
C*
      DOUBLE PRECISION TRACE
C*    FUNCTION ROUTINE TO COMPUTE THE TRACE OF THE PRODUCT OF TWO
C*    SYMMETRIC MATRICES,STORED IN SYMMETRIC NOTATION. THE DIMENSIONS
C*    OF A AND B ARE  N*(N+1)/2
C*
      DOUBLE PRECISION A,B,SUM,SUM1
      DIMENSION A(1),B(1)
      K=0
      SUM1=0.0
      DO 20 I=1,N
      M=I-1
      SUM=0.0
      IF(M.EQ.0)GO TO 15
      SUM=0.0
      DO 10 J=1,M
      K=K+1
   10 SUM = SUM + A(K)*B(K)
   15 K=K+1
      SUM1 = SUM1 + A(K)*B(K) + SUM*2.
   20 CONTINUE
      TRACE = SUM1
      RETURN
      END
```

10-55

147

FILE: TRNDIV FORTRAN A

```
      SUBROUTINE TRNDIV(SPMSR,COVMTX,AVEMTX,COVMT2,AVEMT2,                TRN00010
     *                  WEIGHT,DIVTAB,                                    TRN00020
     *                  WRKRY,IWRKSZ,IPART,PARTLS,AMAT,IFULL)             TRN00030
C                                                                         TRN00040
C*    SUBROUTINE TO COMPUTE THE AVERAGE WEIGHTED TRANSFORMED              TRN00050
C*    DIVERGENCE, AND PARTIALS WITH RESPECT TO A.                         TRN00060
C*                                                                        TRN00070
C*    IF IFULL=1  COMPUTE TRANSFORMAED DIVERGENCE FOR ALL 'NOFET' CHANNFTRN00080
C*                PARTIALS CANNOT BE COMPUTED WHEN IFULL=1.               TRN00090
C*                                                                        TRN00100
C*                                                                        TRN00110
C                                                                         TRN00120
      INCLUDE COMBK7.LIST                                                 TRN00130
      DOUBLE PRECISION SPMSR                                              TRN00140
C     INCLUDE COMBK1.LIST                                                 TRN00150
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,            TRN00160
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,      TRN00170
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),          TRN00180
     *              KEPTS(40),NOGRP,GRPNAM(60),GRPDEX(61),                TRN00190
     *              GRPCHR(61),GROUPS(124)                                COM00010
      COMMON/FSL/CFAC,TOTMSH,SEPMSH,PRCKEY,CRIKEY,INCFET,                 COM00020
     *    INCVFC(30),ICOUNT,SETWGT,EVALAF(100),FETVC4(30)                 COM00030
     *     ,NOFET4,VARSZ4,COPMAS,UTAB4,WGHS14,RESTVC(10),DIVSIZ           COM00040
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSM1,ADRSM2                      COM00050
      INTEGER ADRESD,ADRESP,ADRESF,ADRSM1,ADRSM2,STATKY                   COM00060
      DOUBLE PRECISION CFAC,TOTMSH,SEPMSH                                 TRN00270
C$END                                                                     TRN00280
      INTEGER VARSZ4                                                      TRN00290
      INTEGER VARSZ2,DIVSIZ                                               TRN00300
      DOUBLE PRECISION AMAT,PARTLS                                        TRN00310
      DOUBLE PRECISION DIVTAB,DET,DET2,CON,TRACE                          TRN00320
      DOUBLE PRECISION COVMT2,AVEMT2,WRKRY(1),T(30)                       TRN00330
      DIMENSION COVMTX(VARSZ2,NOCLS2),   COVMT2(VARSZ4,NOCLS2),           TRN00340
     *          AVEMTX(NOFET2,NOCLS2),   AVEMT2(NOFET4,NOCLS2),         TRN00350
     *          WEIGHT(DIVSIZ), DIVTAB(DIVSIZ), PARTLS(1), AMAT(1)      TRN00360
      IVSZ=VARSZ4                                                         TRN00370
      NF=NOFET4                                                           TRN00380
      IF(IFULL.EQ.1)IVSZ=VARSZ2                                           TRN00390
      IF(IFULL.EQ.1)NF=NOFET2                                             TRN00400
      ICOV1=1                                                             TRN00410
      ICOV2=ICOV1+IVSZ                                                    TRN00420
      IS2=ICOV2+IVSZ                                                      TRN00430
      IW1=IS2 + IVSZ                                                      TRN00440
      IS1=IW1+IVSZ                                                        TRN00450
      ITEST=IS1                                                           TRN00460
      IF(IPART.LT.0)GO TO 3                                               TRN00470
C*    ZERO PARTIALS                                                       TRN00480
      IQ=NOFET2*NOFET4                                                    TRN00490
      DO 2 IK=1,IQ                                                        TRN00500
    2 PARTLS(IK)=0.0                                                      TRN00510
      IW2=IS1+VARSZ2                                                      TRN00520
      IW3=IW2+IQ                                                          TRN00530
      IW4=IW3+IQ                                                          TRN00540
      ITEST=IW4+IQ                                                        TRN00550
    3 CONTINUE                                                            TRN00560
      IF(IWRKSZ/2.GE.ITEST)GO TO 1                                        TRN00570
      WRITE(6,600)IWRKSZ                                                  TRN00580
      CALL CMERR                                                          TRN00590
    1 CONTINUE                                                            TRN00600
      SPMSR=0.0                                                           TRN00610
      MN=0                                                                TRN00620
      NC=NOCLS2-1                                                         TRN00630
      DO 100 I=1,NC                                                       TRN00640
      NS=I+1                                                              TRN00650
C*    FIND INVERSE COVAR FOR CLASS I                                      TRN00660
      DO 5 II=1,IVSZ                                                      TRN00670
      IF(IFULL.EQ.1)WRKRY(II)=COVMTX(II,I)                                TRN00680
      IF(IFULL.NE.1)WRKRY(II)=COVMT2(II,I)                                TRN00690
    5 CONTINUE                                                            TRN00700
      CALL COLINV(WRKRY(ICOV1),NF,IERR,3,DET)                             TRN00710
      IF(IERR.EQ.0)GO TO 6                                                TRN00720
      WRITE(6,500)I                                                       TRN00730
      GO TO 100                                                           TRN00740
    6 DO 90 J=NS,NOCLS2                                                   TRN00750
      DO 7 II=1,NF                                                        TRN00760
      IF(IFULL.NE.1)T(II)=AVEMTX(II,I)-AVEMTX(II,J)                       TRN00770
      IF(IFULL.NE.1)T(II)=AVEMT2(II,I)-AVEMT2(II,J)                       TRN00780
    7 CONTINUE                                                            TRN00790
      MN=MN+1                                                             TRN00800
      K=0
```


ORIGINAL PAGE IS
OF POOR QUALITY

FILE: TRNDIV FORTRAN A

```
      DO 12 II=1,NF                                                     TRN00810
      DO 12 IJ=1,II                                                     TRN00820
      K=K+1                                                             TRN00830
      IF(IFULL.NE.1)GO TO 10                                            TRN00840
      WRKRY(IS2+K-1)=COVMTX(K,I)+COVMTX(K,J)+T(II)*T(IJ)                TRN00850
      GO TO 12                                                          TRN00860
   10 WRKRY(IS2+K-1)=COVMT2(K,I) + COVMT2(K,J) + T(II)*T(IJ)            TRN00870
   12 CONTINUE                                                          TRN00880
C*    IF PARTIALS ARE TO BE CALCULATED COMPUTE FULL 'S' MATRIX FOR      TRN00890
C*    CLASSES I AND J                                                   TRN00900
C*                                                                      TRN00910
      IF(IPART.LT.0)GO TO 25                                            TRN00920
      DO 15 II=1,NOFET2                                                 TRN00930
   15 T(II)=AVEMTX(II,I) - AVEMTX(II,J)                                 TRN00940
      K=0                                                               TRN00950
      DO 20 II=1,NOFET2                                                 TRN00960
      DO 20 IJ=1,II                                                     TRN00970
      K=K+1                                                             TRN00980
   20 WRKRY(IS2+K-1)=COVMTX(K,I) + COVMTX(K,J) + T(II)*T(IJ)            TRN00990
C*    FIND INVERSE FOR CLASS J                                          TRN01000
   25 DO 30 II=1,IVSZ                                                   TRN01010
      IF(IFULL.EQ.1)WRKRY(ICOV2+II-1)=COVMTX(II,J)                      TRN01020
      IF(IFULL.NE.1)WRKRY(ICOV2+II-1)=COVMT2(II,J)                      TRN01030
   30 CONTINUE                                                          TRN01040
      CALL COLINV(WRKRY(ICOV2),NF,IERR,3,DET2)                          TRN01050
      IF(IERR.EQ.0)GO TO 35                                             TRN01060
      WRITE(6,500)J                                                     TRN01070
      GO TO 90                                                          TRN01080
C*    SUM INVERSES AND COMPUTE TRACE OF SUM  *  S2                      TRN01090
   35 DO 40 II=1,IVSZ                                                   TRN01100
   40 WRKRY(IW1+II-1)=WRKRY(ICOV1+II-1) + WRKRY(ICOV2+II-1)             TRN01110
      DIVTAB(MN)=TRACE(WRKRY(IW1),WRKRY(IS2),NF)/2. - 2.*NF             TRN01120
      DIVTAB(MN)=DEXP(-DIVTAB(MN)/15.)                                  TRN01130
       SPMSP = SPMSP + DIVTAB(MN)*WEIGHT(MN)                            TRN01140
      IF(IPART.LT.0)GO TO 90                                            TRN01150
C*                                                                      TRN01160
C*    COMPUTE PARTIALS                                                  TRN01170
C*                                                                      TRN01180
      CALL MT1(AMAT,COVMTX(1,I),WRKRY(IW2),NOFET4,NOFET2)               TRN01190
      CALL MT3(WRKRY(ICOV1),WRKRY(IW2),WRKRY(IW3),NOFET4,NOFET4,NOFET2, TRN01200
     *          1,0)                                                    TRN01210
      CALL MT3(WRKRY(IS2),WRKRY(IW3),WRKRY(IW2),NOFET4,NOFET4,NOFET2,   TRN01220
     *          1,0)                                                    TRN01230
      CALL MT3(AMAT,WRKRY(IS1),WRKRY(IW4),NOFET4,NOFET2,NOFET2,0,1)     TRN01240
      DO 42 IK=1,IQ                                                     TRN01250
      L=IK-1                                                            TRN01260
   42 WRKRY(IW4+L)=WRKRY(IW4+L)-WRKRY(IW2+L)                            TRN01270
      CALL MT3(WRKRY(ICOV1),WRKRY(IW4),WRKRY(IW2),NOFET4,NOFET4,        TRN01280
     *          NOFET2,1,0)                                           TRN01290
      CALL MT1(AMAT,COVMTX(1,J),WRKRY(IW3),NOFET4,NOFET2)               TRN01300
      CALL MT3(WRKRY(ICOV2),WRKRY(IW3),WRKRY(IW4),NOFET4,NOFET4,        TRN01310
     *          NOFET2,1,0)                                           TRN01320
      CALL MT3(WRKRY(IS2),WRKRY(IW4),WRKRY(IW3),NOFET4,NOFET4,          TRN01330
     *          NOFET2,1,0)                                           TRN01340
      CALL MT3(AMAT,WRKRY(IS1),WRKRY(IW4),NOFET4,NOFET2,NOFET2,0,1)     TRN01350
      DO 43 IK=1,IQ                                                     TRN01360
      L=IK-1                                                            TRN01370
   43 WRKRY(IW4+L)=WRKRY(IW4+L)-WRKRY(IW3+L)                            TRN01380
      CALL MT3(WRKRY(ICOV2),WRKRY(IW4),WRKRY(IW3),NOFET4,NOFET4,        TRN01390
     *          NOFET2,1,0)                                           TRN01400
      DO 44 IK=1,IQ                                                     TRN01410
      L=IK-1                                                            TRN01420
   44 WRKRY(IW2+L)=WRKRY(IW2+L) + WRKRY(IW3+L)                          TRN01430
      CON = WEIGHT(MN)*DIVTAB(MN)/(16.*NOCLS2)                          TRN01440
      DO 50 IK=1,IQ                                                     TRN01450
      PARTLS(IK)=PARTLS(IK) - CON*WRKRY(IW2+IK-1)                       TRN01460
   50 CONTINUE                                                          TRN01470
   90 CONTINUE                                                          TRN01480
  100 CONTINUE                                                          TRN01490
      SPMSP = SPMSP/NOCLS2                                              TRN01500
      RETURN                                                            TRN01510
  500 FORMAT(' REDUCED COVARIANCE MATRIX FOR CLASS',I3,' IS NOT POSITIVETRN01520
     * DEFINITE')                                                       TRN01530
  600 FORMAT(' NOT ENOUGH WORK AREA IN TRNDIV -- LWRKSZ=',I5)           TRN01540
      END                                                               TRN01550
```

FILE: TRNSFR FORTRAN A

```
      SUBROUTINE TRNSFR(A,A2,W,BMAT)                                    TRN00010
C     INCLUDE COMMK7.LIST                                               TRN00020
C     INCLUDE COMBK1.LIST                                               TRN00030
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          TRN00040
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLDSV2,VERTX2,    TRN00050
     *              FFTVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        TRN00060
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             TRN00070
     *              GRPCHK(61),GROUPS(124)                              TRN00080
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PWCKEY,CRIKEY,INCFET,               COM00010
     *    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FFTVC4(30)               COM00020
     *     ,NOFET4,VARSZ4,CORBAS,DTAR4,WGHS14,RESTVC(10),DIVSIZ          COM00030
     *    ,STATKY,ADRESD,ADRESR,ADRESF,ADRSH1,ADRSH2                    COM00040
      INTEGER ADRESD,ADRESR,ADRESF,ADRSH1,ADRSH2,STATKY                 COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                               COM00060
CSEND                                                                   TRN00160
      INTEGER VARSZ4,VARSZ2                                             TRN00170
      DOUBLE PRECISION SUM                                              TRN00180
      DOUBLE PRECISION BMAT(NOFET4,NOFET2)                              TRN00190
      DOUBLE PRECISION A2(VARSZ4,NOCLS2),W (NOFET4,NOFET2)              TRN00200
      DIMENSION A(VARSZ2,NOCLS2)                                        TRN00210
C*                                                                      TRN00220
C*    MULTIPLY BMAT * A * BMAT(TRANPOSE) AND STORE IN A2                TRN00230
C*                                                                      TRN00240
      DO 190 JJ=1,NOCLS2                                                TRN00250
      DO 150 I=1,NOFET4                                                 TRN00260
      DO 150 J=1,NOFET2                                                 TRN00270
      SUM=0.0                                                           TRN00280
      DO 140 K=1,NOFET2                                                 TRN00290
      IF(K.GE.J)IP=K*(K-1)/2 + J                                        TRN00300
      IF(K.LT.J)IP=J*(J-1)/2 + K                                        TRN00310
  140 SUM=SUM + BMAT(I,K)* A(IP,JJ)                                     TRN00320
      W(I,J)=SUM                                                        TRN00330
  150 CONTINUE                                                          TRN00340
C*                                                                      TRN00350
      DO 180 I=1,NOFET4                                                 TRN00360
      DO 170 J=1,NOFET4                                                 TRN00370
      SUM=0.0                                                           TRN00380
      DO 160 K=1,NOFET2                                                 TRN00390
  160 SUM=SUM + W(J,K) * BMAT(I,K)                                      TRN00400
      IF(J.GE.I)IP=J*(J-1)/2 + I                                        TRN00410
      IF(J.LT.I)IP=I*(I-1)/2 + J                                        TRN00420
      A2(IP,JJ)=SUM                                                     TRN00430
  170 CONTINUE                                                          TRN00440
  180 CONTINUE                                                          TRN00450
  190 CONTINUE                                                          TRN00460
      RETURN                                                            TRN00470
      END                                                               TRN00480
```

10-58

150

FILE: USERIN FORTRAN A

```
      SUBROUTINE USERIN(COVMTX,AVEMTX,DIVTAB,WEIGHT,COVMT2,AVEMT2,S,S2,  USE00010
     *                  BMAT,WRKRY,IWRKSZ)                               USE00020
C*                                                                       USE00030
C*    SUBROUTINE USERIN COORDINATES THE NECESSARY ROUTINES TO COMPUTE     USE00040
C*    THE REQUESTED SEPARABILITY MEASURE FOR THE INPUT B-MATRIX.         USE00050
C*                                                                       USE00060
C*                                                                       USE00070
C     INCLUDE COMRK1.LIST                                                USE00080
C     INCLUDE COMRK7.LIST                                                COM00010
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           COM00020
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     COM00030
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         COM00040
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              COM00050
     *              GRPCHK(61),GROUPS(124)                               COM00010
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,                COM00020
     *    INCVEC(30),ICOUNT,SETWGT,EVALBF(100),FETVC4(30)                COM00030
     *     ,NOFET4,VARSZ4,CORMAS,DTAR4,WGHS14,RESTVC(10),DIVSIZ           COM00040
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                     COM00050
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY                  COM00060
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                                USE00100
C$END                                                                    USE00110
      INTEGER CRIKEY,VARSZ2,VARSZ4                                       USE00120
      DIMENSION  COVMTX(VARSZ2,NOCLS2) , AVEMTX(NOFET2,NOCLS2),          USE00130
     * S(VARSZ2,NOCLS2),                                                USE00140
     *           WEIGHT(1),WRKRY(1)                                      USE00150
      DOUBLE PRECISION COVMT2(VARSZ4,1),AVEMT2(NOFET4,1),                USE00160
     * S2(VARSZ4,1),BMAT(1)                                             USE00170
      DOUBLE PRECISION DIVTAB(1)                                         USE00180
      DIMENSION DUM(1)                                                   USE00190
C*                                                                       USE00200
C*    GET B-MATRIX FROM FILE IN SINGLE PRECISION THEN STORE IN D.P.      USE00210
C*                                                                       USE00220
      CALL BMFIL(WRKRY,NOFET4,NOFET2,FETVC2,2)                           USE00230
      IK=NOFET4*NOFET2                                                   USE00240
      DO 10 I=1,IK                                                       USE00250
   10 BMAT(I)=WRKRY(I)                                                   USE00260
C*                                                                       USE00270
C*    GET TRANSFORMED STATISTICS                                         USE00280
C*                                                                       USE00290
      CALL GTSTAT(COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,DUM,BMAT,WRKRY,       USE00300
     *            IWRKSZ)                                                USE00310
C*    EVALUATE SEPARABILITY MEASURE                                      USE00320
      IPART=-1                                                           USE00330
      CALL EVALSP(SEPMSR,COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,DIVTAB,        USE00340
     *            WEIGHT,IPART,DUM,BMAT,WRKRY,IWRKSZ)                    USE00350
      IF(CRIKEY.NE.1)RETURN                                              USE00360
C*                                                                       USE00370
C*    EVALUATE INTERCLASS DIVERGENCES                                    USE00380
C*                                                                       USE00390
      CALL DIVRG1(COVMT2,VARSZ4,AVEMT2,DIVTAB,NOCLS2,NOFET4,             USE00400
     *            WRKRY,IWRKSZ)                                          USE00410
      RETURN                                                             USE00420
      END
```

FILE: WGTCHK FORTRAN A

```
      SUBROUTINE WGTCHK(WEIGHT,CLSNAM,NAMPR,WGHT,WPTR,WRKRY,NOCLS2)      WGT00010
      IMPLICIT INTEGER(A-Z)                                              WGT00020
      REAL WEIGHT(1),WGHT(1),WRKRY(NOCLS2,NOCLS2),WHT                    WGT00030
      DIMENSION CLSNAM(NOCLS2),NAMPR(2,WPTR)                             WGT00040
      REAL RWGT,RKEY                                                     WGT00050
      LOGICAL*1 LTEMP(4),LSTOR(4),NSTOR(4)                               WGT00060
      EQUIVALENCE (RWGT,SETWGT),(RKEY,WTKEY),(LSTORE,LSTOR(1)),(NSTOPE,  WGT00070
     *NSTOR(1)),(ITEMP,LTEMP(1))                                         WGT00080
      DATA BLANK/' '/,OTHERS/'OTHE'/,BLANKS/'    '/                      WGT00090
C*    PASS KEYS IN ALREADY EXISTING STORAGE                              WGT00100
      RWGT = WEIGHT(1)                                                   WGT00110
      RKEY = WEIGHT(2)                                                   WGT00120
C*    DELETE BLANKS FOR TESTING                                          WGT00130
      DO 20 I=1,NOCLS2                                                   WGT00140
      LSTORE = CLSNAM(I)                                                 WGT00150
      NSTORE = BLANKS                                                    WGT00160
      IZ = 1                                                             WGT00170
      DO 10 IY=1,4                                                       WGT00180
      ITEMP=BLANKS                                                       WGT00190
      LTEMP(4)=LSTOR(IY)                                                 WGT00200
      IF((ITEMP.EQ.BLANK).AND.(IZ.EQ.1))GO TO 10                         WGT00210
      NSTOR(IZ) = LSTOR(IY)                                              WGT00220
      IZ = IZ + 1                                                        WGT00230
   10 CONTINUE                                                           WGT00240
   20 CLSNAM(I) = NSTORE                                                 WGT00250
      IF(WTKEY.EQ.1) GO TO 27                                            WGT00260
C*                                                                       WGT00270
C*    SET ALL CLASS PAIR WEIGHTS TO 1.0 IF USER HAS NOT INPUT WEIGHT FOR WGT00280
C*      'OTHERS' OR 'CLSWT'.  SET TO 'OTHERS' VALUE IF INPUT.            WGT00290
C*                                                                       WGT00300
      WHT=1.0                                                            WGT00310
      DO 22 I=1,WPTR                                                     WGT00320
      IF(NAMPR(1,I) .NE. OTHERS)GO TO 22                                 WGT00330
      WHT=WGHT(I)                                                        WGT00340
      GO TO 24                                                           WGT00350
   22 CONTINUE                                                           WGT00360
   24 DO 25 IK=1,NOCLS2                                                  WGT00370
      DO 25 JK=1,NOCLS2                                                  WGT00380
      WRKRY(IK,JK)=WHT                                                   WGT00390
   25 WRKRY(JK,IK)=WHT                                                   WGT00400
C*                                                                       WGT00410
C*     PLACEMENT OF INPUTED WEIGHT VALUES                                WGT00420
C*                                                                       WGT00430
   27 IF(SETWGT.NE.2) GO TO 55                                           WGT00440
      DO 50 I = 1,WPTR                                                   WGT00450
      DO 40 J=1,NOCLS2                                                   WGT00460
      IF(CLSNAM(J) .NE. NAMPR(1,I))GO TO 40                              WGT00470
C*                                                                       WGT00480
C*    FOUND MATCH ON FIRST NAME IN NAMPR - INDEX J                       WGT00490
C*    NOW SEE ABOUT SECOND NAME                                          WGT00500
C*                                                                       WGT00510
      IF(NAMPR(2,I).EQ. BLANK) GO TO 35                                  WGT00520
      DO 30 K=1,NOCLS2                                                   WGT00530
      IF(CLSNAM(K) .NE. NAMPR(2,I))GO TO 30                              WGT00540
C*                                                                       WGT00550
C*    FOUND SECOND MATCH - INDEX K                                       WGT00560
C*                                                                       WGT00570
      WRKRY(J,K)=WGHT(I)                                                 WGT00580
      WRKRY(K,J)=WGHT(I)                                                 WGT00590
      GO TO 50                                                           WGT00600
   30 CONTINUE                                                           WGT00610
      WRITE(6,100) NAMPR(2,I)                                            WGT00620
      GO TO 50                                                           WGT00630
C*                                                                       WGT00640
C*    ALL PAIRS FOR CLASS J SET TO SAME WEIGHT                           WGT00650
C*                                                                       WGT00660
   35 DO 36 IK=1,NOCLS2                                                  WGT00670
      WRKRY(IK,J)=WGHT(I)                                                WGT00680
   36 WRKRY(J,IK)=WGHT(I)                                                WGT00690
      GO TO 50                                                           WGT00700
   40 CONTINUE                                                           WGT00710
      IF(NAMPR(1,I) .EQ. OTHERS)GO TO 50                                 WGT00720
   46 WRITE(6,100) NAMPR(1,I)                                            WGT00730
   50 CONTINUE                                                           WGT00740
   55 CONTINUE                                                           WGT00750
C*                                                                       WGT00760
C*    REDUCE WRKRY MATRIX AND STORE IN WEIGHTS                           WGT00770
C*                                                                       WGT00780
      K=0                                                                WGT00790
```

FILE: WGTCHK FORTRAN A

```
      NC=NOCLS2-1                                                       WGT00800
      DO 60 I=1,NC                                                      WGT00810
      IK=I+1                                                            WGT00820
      DO 60 J=IK,NOCLS2                                                 WGT00830
      K=K+1                                                             WGT00840
      WEIGHT(K)=WPKRY(I,J)                                              WGT00850
   60 CONTINUE                                                          WGT00860
      RETURN                                                            WGT00870
  100 FORMAT(' SUBCLASS ',A6,' IS NOT AMONG INPUT SUBCLASSES - WEIGHT INWGT00880
     *PUT IGNORED')                                                     WGT00890
      END                                                               WGT00900
```

10-61

FILE: WGTSCN FORTRAN A

```
      SUBROUTINE WGTSCN(CARD,COL,NAMPR,WGHT,WSIZ,NCNT)                  WGT00010
      IMPLICIT INTEGER(A-Z)                                             WGT00020
C|-----------------------------------------------------------------------|WGT00030
C|                                                                       |WGT00040
C|     CALL... J=WGTSCN(CARD,COL,NAMPR,WGHT,WSIZ)                        |WGT00050
C|                                                                       |WGT00060
C|     ARGS... CARD - ARRAY OF CHARACTERS TO BE SCANNED.                 |WGT00070
C|                    ONE CHARACTER PER COMPUTER WORD.                   |WGT00080
C|              COL  - COLUMN IN CARD TO BEGIN SCAN. ON OUTPUT           |WGT00090
C|                    COL IS LAST COLUMN OF CARD SCANNED.                |WGT00100
C|              NAMPR- ARRAY CONTAINING,ON OUTPUT, THE PAIRS OF CLASS    |WGT00110
C|                    NAMES SCANNED FROM CARD.                           |WGT00120
C|              WGHT - ARRAY CONTAINING WEIGHT FOR CORRESPONDING         |WGT00130
C|                    CLASS PAIR                                         |WGT00140
C|              WSIZ - SIZE OF WGHT BUFFER                               |WGT00150
C|                                                                       |WGT00160
C|     NCNT - RUNNING COUNT OF NAME  PAIRS SCANNED                       |WGT00170
C|     PURPOSE.SCANS THE WEIGHTS CONTROL CARD, SAVING THE CLASS NAME     |WGT00180
C|              PAIRS AND ASSOCIATED WEIGHT FOR LATER VERIFICATION IN    |WGT00190
C|              SUBROUTINE WGTCHK.  THE WEIGHT CARD MAY TAKE THE FOLLOWIN|WGT00200
C|              FORMS:                                                   |WGT00210
C|                 WEIGHT    CLASS 1=10.5,CLASS 2=12.0, OTHERS=20.0      |WGT00220
C|              OR:                                                      |WGT00230
C|                 WEIGHTS    (CLASS 1,CLASS 2)=15.0,CLASS 3=1.0,OTHERS=5|WGT00240
C|                                                                       |WGT00250
C|-----------------------------------------------------------------------|WGT00260
      DIMENSION COMVEC(2),EQUVEC(2),RPNVEC(2)                           WGT00270
      DATA BLANK/' '/,LFTPRN/'('/,COMMA/','/,EQUAL/'='/,RHTPRN/')'/     WGT00280
      DATA BLANKS/'    '/                                               WGT00290
      DATA COMVEC/1,','/,EQUVEC/1,'='/,RPNVEC/1,')'/                    WGT00300
      REAL WGHT                                                         WGT00310
      DIMENSION CARD(1),NAMPR(2,1),WGHT(1)                              WGT00320
      LOGICAL*1 LSTOR(4),LCARD(4)                                       WGT00330
      EQUIVALENCE (MCARD,LCARD(1)),(STOR,LSTOR(1))                      WGT00340
C*                                                                      WGT00350
C*                                                                      WGT00360
    1 J=NXTCHR(CARD,COL)                                                WGT00370
      IF(J.EQ.BLANK)GO TO 60                                            WGT00380
      IF(J.NE.LFTPRN)GO TO 2                                            WGT00390
      COL=COL+1                                                         WGT00400
    2 CONTINUE                                                          WGT00410
C*    BLANK  OUT  NAMPR  FOR  THIS  PAIR                                WGT00420
      NCNT=NCNT+1                                                       WGT00430
      DO 3 I=1,2                                                        WGT00440
    3 NAMPR(I,NCNT)=BLANKS                                              WGT00450
      IR=WSIZ-NCNT                                                      WGT00460
      IF(IR.GT.0)GO TO 4                                                WGT00470
      WRITE(4,200)WSIZ                                                  WGT00480
      GO TO 60                                                          WGT00490
    4 CONTINUE                                                          WGT00500
      STOR=BLANKS                                                       WGT00510
      WCNT = 0                                                          WGT00520
    6 WCNT = WCNT + 1                                                   WGT00530
      IF(CARD(COL).EQ.BLANK)GO TO 10                                    WGT00540
      IF(CARD(COL).EQ.EQUAL)GO TO 45                                    WGT00550
      IF(CARD(COL).EQ.COMMA)GO TO 20                                    WGT00560
      MCARD = CARD(COL)                                                 WGT00570
      LSTOR(WCNT)=LCARD(1)                                              WGT00580
      NAMPR(1,NCNT)=STOR                                                WGT00590
   10 COL=COL+1                                                         WGT00600
      IF(WCNT.EQ.4)GO TO 15                                             WGT00610
      GO TO 6                                                           WGT00620
C*                                                                      WGT00630
C*    ONLY FOUR CHARACTERS PER NAME ALLOWED - IGNORE REMAINDER          WGT00640
C*    FIND = OR , - ERROR OTHERWISE                                     WGT00650
C*                                                                      WGT00660
   15 CONTINUE                                                          WGT00670
      J=FIND12(CARD,COL,COMVEC)                                         WGT00680
      IF(J.EQ.-1)GO TO 40                                               WGT00690
      GO TO 19                                                          WGT00700
C*                                                                      WGT00710
C*    COMMA FOUND - ANOTHER NAME SHOULD FOLLOW                          WGT00720
C*                                                                      WGT00730
   19 WCNT = 0                                                          WGT00740
   20 COL=COL+1                                                         WGT00750
   21 WCNT = WCNT + 1                                                   WGT00760
      STOR = BLANKS                                                     WGT00770
      IF(CARD(COL).EQ.RHTPRN)GO TO 40                                   WGT00780
      IF(CARD(COL).EQ.BLANK)GO TO 25                                    WGT00790
```

FILE: WGTSCN FORTRAN A

```
      MCARD = CARD(COL)                                                 WGT00800
      LSTOR(NCNT) = LCARD(1)                                            WGT00810
      NAMPR(2,NCNT)=STOR                                                WGT00820
   25 COL=COL+1                                                         WGT00830
      IF(NCNT.EQ.4)GO TO 30                                             WGT00840
      GO TO 21                                                          WGT00850
   30 CONTINUE                                                          WGT00860
      J=FIND12(CARD,COL,RPNVEC)                                         WGT00870
      IF(J.NE.-1)GO TO 40                                               WGT00880
      WRITE(6,100)                                                      WGT00890
      NCNT=NCNT-1                                                       WGT00900
      GO TO 60                                                          WGT00910
C*                                                                      WGT00920
C*                                                                      WGT00930
C*                                                                      WGT00940
   40 J=FIND12(CARD,COL,EQUVEC)                                         WGT00950
      IF(J.NE.-1)GO TO 45                                               WGT00960
      WRITE(6,100)                                                      WGT00970
      NCNT=NCNT-1                                                       WGT00980
      GO TO 60                                                          WGT00990
   45 J=FLTNUM(CARD,COL,WGHT(NCNT),IR)                                  WGT01000
      COL=COL-1                                                         WGT01010
      J=NXTCHR(CARD,COL)                                                WGT01020
      IF(J.EQ.COMMA)GO TO 1                                             WGT01030
      IF(J.EQ.BLANK)GO TO 60                                            WGT01040
      IF(J.NE.LFTPRN)GO TO 2                                            WGT01050
      COL=COL+1                                                         WGT01060
      GO TO 2                                                           WGT01070
   60 CONTINUE                                                          WGT01080
      RETURN                                                            WGT01090
  100 FORMAT(' SYNTAX ERROR ON WEIGHT CARD-REMAINDER OF CARD IGNORED')  WGT01100
  200 FORMAT(' WEIGHT BUFFER IS FILLED-ONLY',I5,' CLASS NAME PAIRS ALLOWWGT01110
     *ED')                                                              WGT01120
      END                                                               WGT01130
```

FILE: WHRPLC FORTRAN A

```
      SUBROUTINE WHRPLC(COVMTX,AVEMTX,DIVTAB,WEIGHT,COVMT2,          WHR00010
     *                  AVEMT2,S,S2,WRKRY,IWRKSZ)                    WHR00020
C*                                                                   WHR00030
C*    SUBROUTINE TO FIND THE BEST SET OF NOFET4 FEATURES             WHR00040
C*    USING THE 'WITHOUT REPLACEMENT' PROCEDURE.                     WHR00050
C*                                                                   WHR00060
      INTEGER FETVC2,FETVC4,TVEC,TRYVEC,KEEP                         WHR00070
C     INCLUDE COMBK7.LIST                                            WHR00080
C     INCLUDE COMBK1.LIST                                            WHR00090
      COMMON/INFOCM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,       WHR00100
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2, WHR00110
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),     WHR00120
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),          WHR00130
     *              GRPCHK(61),GROUPS(124)                           WHR00140
      COMMON/FSL/CFAC,TOTMSR,SEPMSR,PRCKEY,CRIKEY,INCFET,            COM00010
     *    INCVEC(30),ICOUNT,SETWGT,EVALHF(100),FETVC4(30)            COM00020
     *     ,NOFET4,VARSZ4,CORMAS,DTAB4,WGMS14,MESTVC(10),DIVSIZ       COM00030
     *    ,STATKY,ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2                 COM00040
      INTEGER ADRESD,ADRESP,ADRESF,ADRSH1,ADRSH2,STATKY              COM00050
      DOUBLE PRECISION CFAC,TOTMSR,SEPMSR                            COM00060
CSEND                                                                WHR00220
      DOUBLE PRECISION COVMT2(1),AVEMT2(1),S2(1)                     WHR00230
      DOUBLE PRECISION DIVTAB(1),TMSR,DUM(1),DM                      WHR00240
      INTEGER CRIKEY                                                 WHR00250
      DIMENSION  COVMTX(1), AVEMTX(1),  WEIGHT(1),                   WHR00260
     :S(1),WRKRY(1)                                                  WHR00270
      IPART=-1                                                       WHR00280
      DIMENSION TVEC(30),NBEST(30),TRYVEC(30)                        WHR00290
C*                                                                   WHR00300
C*    SAVE THE VALUE OF NOFET4                                       WHR00310
      NFSAVE=NOFET4                                                  WHR00320
      IF(NBST.GT.0)GO TO 15                                          WHR00330
      IF(INCFET.LE.0)GO TO 15                                        WHR00340
      DO 10 I=1,INCFET                                               WHR00350
      DO 5 J=1,NOFET2                                                WHR00360
      IF(INCVEC(I).EQ.FETVC2(J))GO TO 6                              WHR00370
    5 CONTINUE                                                       WHR00380
      WRITE(6,100)INCVEC(I)                                          WHR00390
      GO TO 10                                                       WHR00400
    6 NBST=NBST+1                                                    WHR00410
      NBEST(NBST)=J                                                  WHR00420
   10 CONTINUE                                                       WHR00430
C*                                                                   WHR00440
C*    SET UP VECTOR OF FEATURES TO TRY WITH NBEST                    WHR00450
C*                                                                   WHR00460
   15 IF(NBST.GE.NFSAVE)GO TO 50                                     WHR00470
      SEPMSR=1.E+35                                                  WHR00480
      NTRY=0                                                         WHR00490
      DO 25 I=1,NOFET2                                               WHR00500
      IF(NBST.EQ.0)GO TO 24                                          WHR00510
      DO 20 J=1,NBST                                                 WHR00520
      IF(I.EQ.NBEST(J))GO TO 25                                      WHR00530
   20 CONTINUE                                                       WHR00540
   24 NTRY=NTRY+1                                                    WHR00550
      TRYVEC(NTRY)=I                                                 WHR00560
   25 CONTINUE                                                       WHR00570
C*                                                                   WHR00580
C*    TRY EACH FEATURE IN TRYVEC WITH THE 'BEST' SO FAR AND KEEP     WHR00590
C*    THE ONE WHICH GIVES MAXIMUM SEPARABILITY MEASURE.              WHR00600
C*                                                                   WHR00610
      NF=NBST+1                                                      WHR00620
      NOFET4=NF                                                      WHR00630
      DO 40 I=1,NTRY                                                 WHR00640
      IF(NBST.EQ.0)GO TO 35                                          WHR00650
      DO 30 J=1,NBST                                                 WHR00660
   30 TVEC(J)=NBEST(J)                                               WHR00670
   35 TVEC(NF)=TRYVEC(I)                                             WHR00680
      CALL ORDER(TVEC,NF)                                            WHR00690
      CALL GTSTAT(COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,TVEC,DM,WRKRY,IWRKSZ)WHR00700
      CALL EVALSP(TMSR,COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,DIVTAB,      WHR00710
     *            WEIGHT,IPART,DUM,DUM,WRKRY,IWRKSZ)                 WHR00720
      IF(SEPMSR.LT.TMSR)GO TO 40                                     WHR00730
      KEEP=TRYVEC(I)                                                 WHR00740
      SEPMSR=TMSR                                                    WHR00750
   40 CONTINUE                                                       WHR00760
      NBST=NBST+1                                                    WHR00770
      NBEST(NBST)=KEEP                                               WHR00780
      GO TO 15                                                       WHR00790
   50 NOFET4=NFSAVE                                                  WHR00800
```

FILE: WHPPLC FORTRAN A

```
      DO 60 I=1,NOFET4                                                  WHR00810
      K=NHEST(I)                                                        WHR00820
      TVEC(I)=K                                                         WHR00830
   60 FETVC4(I)=FETVC2(K)                                               WHR00840
C*                                                                      WHR00850
C*    COMPUTE INTERCLASS MEASURES FOR FEATURES CHOSEN.                  WHR00860
C*                                                                      WHR00870
      CALL ORDER(TVEC,NOFET4)                                           WHR00880
      CALL GTSTAT(COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,TVEC,DUM,WRKRY,      WHR00890
     *            IWRKSZ)                                               WHR00900
      CALL FVALSP(SPONSO,COVMTX,AVEMTX,S,COVMT2,AVEMT2,S2,DIVTAB,       WHR00910
     *            WEIGHT,IPART,DUM,DUM,WRKRY,IWRKSZ)                    WHR00920
      IF(CRITKEY.NE.1)RETURN                                            WHR00930
      CALL DIVRG1(COVMT2,VARSZ4,AVEMT2,DIVTAB,NOCLS2,                   WHR00940
     *            NOFET4,WRKRY,IWRKSZ)                                  WHR00950
      RETURN                                                            WHR00960
  100 FORMAT(' THE INCLUDE REQUEST FOR FEATURE',I4,                     WHR00970
     *   ' IS NOT A LEGITIMATE REQUEST--IGNORED')                       WHR00980
      END                                                               WHR00990
```

ORIGINAL PAGE IS
OF POOR QUALITY

11. CLASSIFY PROCESSOR

FILE: CLSFY FORTRAN A

```
      SUBROUTINE CLSFY(ARRAY,TOP)                                          CLS00010
      IMPLICIT INTEGER (A-H,O-Z)                                           CLS00020
      DIMENSION ARRAY(3000)                                                CLS00030
C                                                                          CLS00040
CCCI-----------------------------------------------------------------------ICLS00050
CCCI-----------------------------------------------------------------------ICLS00060
CCCI                                                                       ICLS00070
CCCI    CALL..    CALL CLSFY (ARRAY,TOP)                                   CLS00080
CCCI                                                                       ICLS00090
CCCI    ARGS..    ARRAY -  SFE 'MONTOR'                                    ICLS00100
CCCI              TOP   -  SFE 'MONTOR'                                    ICLS00110
CCCI                                                                       ICLS00120
CCCI    REQUIRES. COMMONS  /INFORM/CLASS/GLOBAL/BMTRX/SCRACH/              CLS00130
CCCI              ROUTINES SETUP2    CLSFY1    CLSFY2                      ICLS00140
CCCI                                                                       ICLS00150
CCCI    PURPOSE.. COORDINATES THE VARIOUS ROUTINES                         ICLS00160
CCCI              FOR 'CLASSIFICATION' STEP                                ICLS00170
CCCI                                                                       ICLS00180
CCCI    RETURNS.. NONE                                                     ICLS00190
CCCI                                                                       ICLS00200
CCCI-----------------------------------------------------------------------ICLS00210
CCCI-----------------------------------------------------------------------ICLS00220
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLS00230
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLS00240
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLS00250
      INCLUDE COMBK1.LIST                                                  CLS00260
      ---------------------------------------------------------------------CLS00280
      INCLUDE COMBK2.LIST                                                  CLS00290
CC    INCLUDE COMBK6.LIST                                                  CLS00300
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFLD2,             CLS00310
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,       COM00010
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),           COM00020
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                COM00030
     *              GRPCHK(61),GROUPS(124)                                 COM00040
C$END                                                                      COM00050
      COMMON /CLASS/ APRFLG,BMCOMB,BMFEAT,BMFLG,NOCAT,THIJ1,IDATA1,        CLS00330
     *               NFILE,STATKY,CATNAM(60),                              CLS00340
     3               CLSSYM(60),CON(60),DET(60),FLDESC,FLDINF(6),          CLS00350
     4               KCLSNA(60),NOCTCL(60),SUBCAT(60)                      CLS00360
     *    ,NOCHAN,CHNVEC(30)                                               CLS00370
                                                                           CLS00380
C*    CLASS COMMON IS USED BY THE CLASSIFY PROCESSOR. IT IS IN CORE        CLS00390
C*    ONLY WHEN THIS PROCESSOR IS.                                         CLS00400
C*       APRFLG - SET TO THE NUMBER OF APRIORI VALUES READ IN FROM         CLS00410
C*                INPUT CARDS                                              CLS00420
C*       BMCOMB - NUMBER OF LINEAR COMBINATIONS IN B-MATRIX                CLS00430
C*       BMFEAT - NUMBER OF CHANNELS USED IN COMPUTING THE B-MATRIX        CLS00440
C*       BMFLG  - INDICATES WHETHER A B-MATRIX HAS BEEN INPUT              CLS00450
C*       NOCAT  - NUMBER OF CATEGORIES                                     CLS00460
C*       THIJ1  - BEGINNING ADDRESS FOR STORING THE CLASS-PAIR THRES-      CLS00470
C*                HOLD TABLE                                               CLS00480
C*       IDATA1 - BEGINNING ADDRESS FOR STORING THE DATA                   CLS00490
C*       NFILE  - NUMBER OF THE NEXT FILE TO BE WRITTEN ON MAPTAP          CLS00500
C*       STATKY - FLAG CONTROLLING STATISTICS PRINT OUT                    CLS00510
C*       CATNAM - CONTAINS THE CATEGOTY NAMES                              CLS00520
C*       CLSSYM - DEFAULT SYMBOLS USED IN PRINTING CLASSIFICATION MAP      CLS00530
C*       CON    - CONTAINS THE SUBCLASS CONSTANTS                          CLS00540
C*       DET    - CONTAINS THE SUBCLASS DETERMINANTS                       CLS00550
C*       FLDESC - FIELD NAME                                               CLS00560
C*       FLDINF - CONTAINS THE RECTANGULAR COORDINATES SURROUNDING         CLS00570
C*                THE NON-RECTANGLE FIELD                                  CLS00580
C*       KCLSNA - CONTAINS THE CLASS NAMES IN THE ORDER TAKEN OFF THE      CLS00590
C*                CATEGORY CONTROL CARD                                    CLS00600
C*       NOCTCL - NUMBER OF CLASSES IN EACH CATEGORY                       CLS00610
C*       SUBCAT - CONTAINS THE CATEGORY NUMBER TO WHICH THE CORRESPONDIN   CLS00620
C*                SUBCLASS BELONGS                                         CLS00630
C*                                                                        CLS00640
C                                                                          CLS00650
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPF,SAVTAP,BMFILE,BMKEY,            CLS00660
     *             HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,       CLS00670
     *    ,DRUMAD,DRMLOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  CLS00680
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                       CLS00690
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTPFL,HISTFL,PCHUNT,                 CLS00700
     *  CRDUNT,PRTUNT,RANDIO                                               CLS00710
C                                                                          CLS00720
      COMMON /BMTRX/BMATRX(450)                                            CLS00730
C                                                                          CLS00740
C     ---------------------------------------------------------------------CLS00750
```

FILE: CLSFY   FORTRAN A

```
C
C CCCCC CCC CCCCCCCCC
      DIMENSION KATNO(60)                                                    CLS00760
                                                                             CLS00770
      ------------------------------------------------------------------     CLS00780
                                                                             CLS00790
      REAL APRIOR(60)                                                        CLS00800
      ------------------------------------------------------------------     CLS00810
                                                                             CLS00820
      COMMON /SCRACH/ IDATA(12500)                                           CLS00830
      ------------------------------------------------------------------     CLS00840
                                                                             CLS00850
      *** NOTE:  THE IDATA ARRAY IS USED  EXTENSIVELY AS A SCRATCH            CLS00860
      AREA IN CLSFY1, FOR OUTPUT OF CLASS-PAIR THRESHOLDS IN                 CLS00870
      SUBR. THRESH ( FOR USE IN CONTEX) , INPUT OF SCAN LINE TO BE           CLS00880
      CLASSIFIED IN CLSFY2.                                                  CLS00890
C                                                                            CLS00900
      FLDFLG=0                                                               CLS00910
      CALL SETUP2(ARRAY,TOP,FLDFLG,APRIOR,BMATRX,KATNO)                      CLS00920
      IF (FLDFLG.EQ.1) GO TO 10                                              CLS00930
C                                                                            CLS00940
      CALL CLSFY1(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(FLDSV2),                  CLS00950
     *     ARRAY(CLSID2),APRIOR,BMATRX,ARRAY(VERTX2),ARRAY(SUBDS2),          CLS00960
     *  ARRAY(SUBNO2),ARRAY(COVAR2),ARRAY(AVAR2),KATNO)                       CLS00970
C                                                                            CLS00980
      CALL CLSFY2(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(FLDSV2),                  CLS00990
     *           ARRAY(CLSID2),ARRAY(SUBDS2),ARRAY(SUBNO2),KATNO,BMATRX)     CLS01000
C                                                                            CLS01010
      END FILE MAPTAP                                                        CLS01020
C                                                                            CLS01030
C                                                                            CLS01040
   10 WRITE (6,20)                                                           CLS01050
   20 FORMAT(//// 1X,'***  $CLASSIFY - COMPLETED  ***' ////)                 CLS01060
      RETURN                                                                 CLS01070
      END                                                                    CLS01080
                                                                             CLS01090
                                                                             CLS01100
                                                                             CLS01110
                                                                             CLS01120
                                                                             CLS01130
                                                                             CLS01140
                                                                             CLS01150
```

FILE: CATGRY FORTRAN A

```
      SUBROUTINE CATGRY(NCHAN,NPTS,AVE,COR,IR,VR,BMATR,IDATA,NLINE,      CAT00010
     *     VERTCS,NV,PTSTHS)                                             CAT00020
C                                                                        CAT00030
CCCCC---------------------------------------------------------------------CAT00040
CCCCC                                                                    CAT00050
CCCCC                                                                    CAT00060
CCCCC                                                                    CAT00070
CCCCC     NCHAN  : NO. OF CHANNELS                                       CAT00080
CCCCC     NPTS   : NO. OF PTS. IN RECTANGLE FIELD                        CAT00090
CCCCC     AVE    : MEANS ARRAY                                           CAT00100
CCCCC     COR    : COVARIANCE ARRAY                                      CAT00110
CCCCC     IR     : CLASSIFIED DATA                                       CAT00120
CCCCC     VR     : CORRESPONDING PDFS OF IR ARRAY                        CAT00130
CCCCC     CON    : SUBCLASS CONSTANTS                                    CAT00140
CCCCC     BMATR  : B-TRANSFORMATION MATRIX, IF AVAILABLE                 CAT00150
CCCCC     IDATA  : SCAN LINE OF DATA TO BE CLASSIFIED                    CAT00160
CCCCC     NLINE  : LINE NUMBER CORRESPONDING TO DATA TAPE                CAT00170
CCCCC     VERTCS : VERTICES OF FIELD TO BE CLASSIFIED                    CAT00180
CCCCC     NV     : NO. OF VERTICES                                       CAT00190
CCCCC                                                                    CAT00200
CCCCC PURPOSE : EACH PIXEL IS ASSIGNED TO A CATEGORY, THEN ASSIGNED      CAT00210
CCCCC           TO A SUBCLASS WITHIN THAT CHOSEN CATEGORY. ON THE        CAT00220
CCCCC           MAPTAP THE CHOSEN SUBCLASS NUMBER AND ITS CORRESPOND-    CAT00230
CCCCC           ING PDF IS OUTPUT.                                       CAT00240
CCCCC                                                                    CAT00250
CCCCC RETURNS : IR ARRAY RETURNS THE SUBCLASS NUMBER EACH PIXEL WAS      CAT00260
CCCCC           ASSIGNED TO                                              CAT00270
CCCCC         : VR ARRAY RETURNS THE CORRESPONDING PDF                   CAT00280
CCCCC---------------------------------------------------------------------CAT00290
C                                                                        CAT00300
      IMPLICIT INTEGER (A-Z)                                             CAT00310
      LOGICAL BMFLAG,KD1                                                 CAT00320
      DIMENSION IDATA(1),IR(1000),FL(22),SUBNUM(60),VERTCS(1)            CAT00330
      REAL VR(1000),AVE(1),COR(1),DATA(30),DM(30),                       CAT00340
     *     BMATR(BMCOMB,BMFEAT),S,P(60),TF,PK                            CAT00350
      REAL  FDATA, SUM                                                   CAT00360
      REAL TEMAX(60),E,CON                                               CAT00370
C     INCLUDE COMBK.LIST                                                 CAT00380
C     INCLUDE COMBK2.LIST                                                CAT00390
      COMMON/INFORM/NOCL32,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           CAT00400
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     CAT00410
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         CAT00420
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              CAT00430
     *              GRPCHK(61),GROUPS(124)                               CAT00440
      COMMON /CLASS/ APPFLG,BMCOMB,BMFEAT,BMFLG,NOCAT,THIJ1,IDATA1,      CAT00450
     *               NFILE,STATKY,CATNAM(60),                            CAT00460
     3               CLSSYM(60),CON(60),DET(60),FLDESC,FLDINF(6),        CAT00470
     4               KCLSNA(60),NOCTCL(60),SUBCAT(60)                    CAT00480
     *     ,NOCHAN,CHNVEC(30)                                            CAT00490
CSEND                                                                    CAT00500
      EQUIVALENCE (FLDINF(1),LINSTR) , (FLDINF(2),LINEND),               CAT00510
     *            (FLDINF(3),LININC),  (FLDINF(4),SAMSTR) ,              CAT00520
     *            (FLDINF(5),SAMEND),  (FLDINF(6),SAMINC)                CAT00530
C                                                                        CAT00540
CC                                                                       CAT00550
      BMFLAG = BMFLG  .GT. 0                                             CAT00560
      IF (BMFLAG) GO TO 10                                               CAT00570
      NF = NCHAN                                                         CAT00580
      GO TO 20                                                           CAT00590
   10 NF = BMCOMB                                                        CAT00600
   20 KD1 = NF .EQ. 1                                                    CAT00610
C                                                                        CAT00620
CCC   ZERO OUT IR AND VR                                                 CAT00630
C                                                                        CAT00640
      DO 55     K=1,NPTS                                                 CAT00650
      IR(K) = 0                                                          CAT00660
   55 VR(K) = 0.0                                                        CAT00670
C                                                                        CAT00680
      CALL FDLINT(VERTCS,NV,FL,NLINE,IPTS,NI)                            CAT00690
      DO 350   JJ=1,NI,2                                                 CAT00700
      IA = (FL(JJ) - SAMSTR) / SAMINC + 1                                CAT00710
      IE = (FL(JJ+1) - SAMSTR) / SAMINC + 1                              CAT00720
      IF (MOD(SAMSTR,SAMINC) .NE. MOD(FL(JJ),SAMINC))  IB = IB + 1       CAT00730
      IF (IA .GT. IE )   GO TO 350                                       CAT00740
   23 DO 250   II=IA,IE                                                  CAT00750
C                                                                        CAT00760
CCC   FLOAT DATA SAMPLE, AND APPLY THE B-MATRIX, IF AVAILABLE            CAT00770
CC                                                                       CAT00780
      IF (BMFLAG)  GO TO 30                                              CAT00790
```

FILE: CATGRY FORTRAN A

```
      DO 25  I=1,NCHAN                                                  CAT00800
      IDUM = NPTS * (I - 1) + II                                        CAT00810
   25 DATA(I) = IDATA(IDUM)                                             CAT00820
      GO TO 45                                                          CAT00830
C                                                                       CAT00840
   30 DO  35   I=1,BMCOMB                                               CAT00850
      SUM = 0.0                                                         CAT00860
      DO 40   K=1,NCHAN                                                 CAT00870
      INDUM = NPTS * (K - 1) + I                                        CAT00880
      FDATA = IDATA(INDUM)                                              CAT00890
   40 SUM = SUM + BMATR(I,K) * FDATA                                    CAT00900
   35 DATA(I) = SUM                                                     CAT00910
   45 CONTINUE                                                          CAT00920
C                                                                       CAT00930
   60 DO 65 LL=1,NOCAT                                                  CAT00940
   65 P(LL) = 0.0                                                       CAT00950
CC                                                                      CAT00960
                                                                        CAT00970
      IBM = - NF                                                        CAT00980
      LC  = 0                                                           CAT00990
      DO 130   KL=1,NOCAT                                               CAT01000
  130 TFMAX(KL) = -1.0E35                                               CAT01010
CCC                                                                     CAT01020
                                                                        CAT01030
      DO 150  KK=1,NOSUB2                                               CAT01040
      IBM = IBM + NF                                                    CAT01050
      KM = IBM + 1                                                      CAT01060
      S  = DATA(1) - AVE(KM)                                            CAT01070
      DM(1) = S                                                         CAT01080
      LC = LC + 1                                                       CAT01090
      TF = COR(KK) * (S*S) / COR(LC)                                    CAT01100
      IF (KD1) GO TO 146                                                CAT01110
CCCC  LOOP FOR COMPUTING THE KD-TH ELEMENT OF Y ( L**-1 * (X-M)), WHICH CAT01120
                                                 IS STORED IN S         CAT01130
CCCC                                                                    CAT01140
      DO 145 KD=2,NF                                                    CAT01150
      KM = IBM + KD                                                     CAT01160
      S  = DATA(KD) - AVE(KM)                                           CAT01170
      J1 = KD - 1                                                       CAT01180
      DO 140  LD=1,J1                                                   CAT01190
      LC = LC + 1                                                       CAT01200
  140 S = S - COR(LC) * DM(LD)                                          CAT01210
C                                                                       CAT01220
      DM(KD) = S                                                        CAT01230
C                                                                       CAT01240
      LC = LC + 1                                                       CAT01250
CCC   COMPUTE THE KD-TH TERM IN :1/2*Y * D**-1 * Y                      CAT01260
                                = 1/2 * (X-M) * K**-1 * (X-M)           CAT01270
CCC                                                                     CAT01280
  145 TF = TF + ( S*S )/COR(LC)                                         CAT01290
C                                                                       CAT01300
  146 TF = -.5 * TF                                                     CAT01310
CCC   SUM THE EXP(P(KK))                                                CAT01320
CCC   IF PDF IS SMALLER THAN -88.  DO NOT EXP.                          CAT01330
      IF ( TF .LE. -88)   F = 0.0                                       CAT01340
      IF ( TF .LE. -88)   GO TO 148                                     CAT01350
      F  = EXP(TF)                                                      CAT01360
  148 CONTINUE                                                          CAT01370
      CTGORY = SUBCAT(KK)                                               CAT01380
      P(CTGORY) = F + P(CTGORY)                                         CAT01390
C                                                                       CAT01400
CCCC  FIND MAX. VALUE OF PDF OVER ALL SUBCLASSES WITHIN A CATEGORY      CAT01410
CCCC  AND SAVE THE SUBCLASS NUMBER OF THE LARGEST PDF                   CAT01420
CCCC                                                                    CAT01430
      IF (TF .LE. TFMAX(CTGORY)) GO TO 150                              CAT01440
      TFMAX(CTGORY) = TF                                                CAT01450
      SUBNUM(CTGORY) = KK                                               CAT01460
  150 CONTINUE                                                          CAT01470
CCCC  FIND THE MAX. CATEGORY PDF (THIS PDF IS THE SUMMATION OF ALL      CAT01480
                                  THE PDF'S OF THE SUBCLASS WITHIN      CAT01490
                                  THE CATEGORY)                         CAT01500
CCCC                                                                    CAT01510
      PK = 0.0                                                          CAT01520
      DO 175  LL=1,NOCAT                                                CAT01530
                                                                        CAT01540
                                                                        CAT01550
                                                                        CAT01560
                                                                        CAT01570
                                                                        CAT01580
```

FILE: CATGRY FORTRAN A

```
      IF ( P(LL) .LE. PK)  GO TO 175                                          CAT01590
      PK = P(LL)                                                              CAT01600
      IC = LL                                                                 CAT01610
  175 CONTINUE                                                                CAT01620
C                                                                             CAT01630
CCCCC                                                                         CAT01640
CCCCC ALL OF THE SUBCLASS PDF'S WERE TOO SMALL TO EXP., THEREFORE             CAT01650
CCCCC       THIS PIXEL WILL NOT BE CLASSIFIED.                                CAT01660
C                                                                             CAT01670
      IF (PK .NE. 0.0)  GO TO 180                                             CAT01680
      IR(III) = NOSUB2 + 1                                                    CAT01690
      VR(III) = 0.0                                                           CAT01700
      PTSTHS = PTSTHS + 1                                                     CAT01710
      GO TO 250                                                               CAT01720
  180 CONTINUE                                                                CAT01730
C                                                                             CAT01740
CCCCC STORE THE LARGEST SUBCLASS PDF AND SUBCLASS NO. OF THE CATEGORY         CAT01750
CCCCC WITH THE  MAX. PDF                                                      CAT01760
C                                                                             CAT01770
      IR(III) = SUBNUM(IC)                                                    CAT01780
      VR(III) = TFMAX(IC)                                                     CAT01790
  250 CONTINUE                                                                CAT01800
  340 CONTINUE                                                                CAT01810
      RETURN                                                                  CAT01820
      END                                                                     CAT01830
```

FILE: CATSCN FORTRAN A

```
      FUNCTION CATSCN(CARD,KCLSNA,CATNME,KK,NOCLSS,NOCAT)                    CAT00010
C                                                                            CAT00020
      IMPLICIT INTEGER (A-Z)                                                 CAT00030
      DIMENSION KCLSNA(1),CARD(42),IBUFF(6)                                  CAT00040
C|-----------------------------------------------------------------------    CAT00050
C|                                                                           CAT00060
C|    CATSCN SCANS THE CATEGORY CARD FROM CLASSIFY AND STORES THE            CAT00070
C|    CATEGORY NAME IN CATNAM AND STORE THE CLASS NAMES IN KCLSNA            CAT00080
C|                                                                           CAT00090
C|-----------------------------------------------------------------------    CAT00100
      DATA BLANK/' '/,SLASH/'/'/,STAR/'*'/,COMMA/','/                        CAT00110
      LOGICAL*1 LCHAR(4),LLCHAR(4)                                           CAT00120
      DIMENSION ICHAR(1),IICHAR(1)                                           CAT00130
      EQUIVALENCE(LCHAR(1),ICHAR(1)),(LLCHAR(1),IICHAR(1))                   CAT00140
      K = 1                                                                  CAT00150
    4 COL = 0                                                                CAT00160
    8 KK = KK + 1                                                            CAT00170
C                                                                            CAT00180
   10 J = NXTCHR(CARD,COL)                                                   CAT00190
      IF(J .EQ. BLANK)GO TO 110                                              CAT00200
      IF(J.EQ.COMMA)GO TO 10                                                 CAT00210
      IF(J.EQ.SLASH)GO TO 10                                                 CAT00220
      IF(J.EQ.STAR)GO TO 100                                                 CAT00230
C                                                                            CAT00240
C     PICK CHARACTERS OFF CARD ONE AT A TIME                                 CAT00250
      LL = 1                                                                 CAT00260
      DO 20    I=1,6                                                         CAT00270
      J2 = CARD(COL)                                                         CAT00280
      IF(J2.EQ.SLASH)GO TO 30                                                CAT00290
      IF(J2.EQ.STAR)GO TO 30                                                 CAT00300
      IF(J2.EQ.COMMA)GO TO 30                                                CAT00310
      IF(J2.EQ.BLANK)GO TO 20                                                CAT00320
      IBUFF(LL) = J2                                                         CAT00330
      LL = LL + 1                                                            CAT00340
      COL = COL + 1                                                          CAT00350
   20 CONTINUE                                                               CAT00360
   30 COL = COL - 1                                                          CAT00370
      IF (LL .GE. 7)  GO TO 35                                               CAT00380
      GO TO 50                                                               CAT00390
C                                                                            CAT00400
   35 DO 40    JJ=LL,6                                                       CAT00410
   40 IBUFF(JJ) = BLANK                                                      CAT00420
C                                                                            CAT00430
   50 CONTINUE                                                               CAT00440
      DO 60    I=1,4                                                         CAT00450
      IICHAR(I)=IBUFF(I)                                                     CAT00460
   60 LCHAR(I)=LLCHAR(I)                                                     CAT00470
      WRD1=ICHAR(1)                                                          CAT00480
C                                                                            CAT00490
      GO TO (70,80),K                                                        CAT00500
C                                                                            CAT00510
C     WRD1 CONTAINS CATEGORY NAME                                            CAT00520
C                                                                            CAT00530
   70 CATNME = WRD1                                                          CAT00540
      K = 2                                                                  CAT00550
      GO TO 10                                                               CAT00560
C                                                                            CAT00570
C     WRD1 CONTAINS CLASS NAME                                               CAT00580
C                                                                            CAT00590
   80 KCLSNA(KK) = WRD1                                                      CAT00600
      NOCLSS = NOCLSS + 1                                                    CAT00610
      GO TO 8                                                                CAT00620
C                                                                            CAT00630
C     NEXT CARD IS A CONTINUATION CARD                                       CAT00640
C                                                                            CAT00650
  100 READ(2,500)CARD                                                        CAT00660
      COL = 0                                                                CAT00670
      WRITE(6,550)CARD                                                       CAT00680
  500 FORMAT(10X,42A1)                                                       CAT00690
  550 FORMAT( T15, 42A1)                                                     CAT00700
      GO TO 10                                                               CAT00710
C                                                                            CAT00720
C     FINISHED SCANNING CARD                                                 CAT00730
C                                                                            CAT00740
  110 KK = KK - 1                                                            CAT00750
      CATSCN = KK                                                            CAT00760
C                                                                            CAT00770
      RETURN                                                                 CAT00780
      END                                                                    CAT00790
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: CLSFY1 FORTRAN A

```
      SUBROUTINE CLSFY1(COVMTX,AVEMTX,FLDMTX,CLSMTX,APRIOR,                CLS00010
     *  BMATR,VERTEX,SUBDES,SUBNO,COVNEW,AVENEW,KATNO)                     CLS00020
C                                                                          CLS00030
C                                                                          CLS00040
      IMPLICIT INTEGER (A-H,O-Z)                                           CLS00050
C                                                                          CLS00060
CCI---------------------------------------------------------------------ICLS00070
CCI---------------------------------------------------------------------ICLS00080
CCI                                                                      ICLS00090
CCI     CALL...   CALL CLSFY1(COVMTX,AVEMTX,FLDMTX,RUNMTX,CLSMTX,APRIOR,  CLS00100
CCI                     BMATR,COVNEW,AVENEW)                             CLS00110
CCI                                                                      ICLS00120
CCI     ARGS...   COVMTX : LOCATION OF COVARIANCE MATRICES ( SYMETTRIC     CLS00130
CCI                     STORAGE ) FOR NOCLS2 TRAINING CLASSES.           CLS00140
CCI                                                                      CLS00150
CCI               AVEMTX : LOCATION OF NOCLS2 TRAINING CLASS MEAN VECTORS CLS00160
CCI                     ( NOFET2 MEANS PER CLASS )                       CLS00170
CCI                                                                      CLS00180
CCI               FLDMTX : LOCATION OF TRAINING FIELD(S) INFORMATION     CLS00190
CCI                                                                      CLS00200
CCI               CLSMTX : LOCATION OF NAME FOR EACH CLASS               CLS00210
CCI                                                                      CLS00220
CCI               APRIOR : LOCATION OF APRIORI PROBABILITY VALUES FOR    CLS00230
CCI                     EACH CLASS                                       CLS00240
CCI                                                                      CLS00250
CCI     VERTEX : LOCATION OF VERTICES OF SAVED TRAINING FIELDS           CLS00260
CCI                                                                      CLS00270
CCI     SUBDES : LOCATION OF SUBCLASS NAMES                              CLS00280
CCI                                                                      CLS00290
CCI     SUBNO  : LOCATION OF ARRAY CONTAINING NO. OF SUBCLASSES IN       CLS00300
CCI              EACH CLASS                                              CLS00310
CCI               COVNEW : LOCATION USED TO STORE ''B''-TRANSFORMED      CLS00320
CCI                     COVARIANCE MATRICES.                             CLS00330
CCI                                                                      CLS00340
      CONTINUE                                                           CLS00350
CCI               AVENEW : LOCATION USED TO STORE THE ''B''-TRANSFORMED  CLS00360
CCI                     MEAN VECTORS.                                    CLS00370
CCI                                                                      CLS00380
CCI               BMATR : LOCATION OF THE ''B''-TRANSFORMATION MATRIX,   CLS00390
CCI                     IF AVAILABLE, FOR APPLICATION TO THE CLASS       CLS00400
CCI                     MEANS AND COVARIANCE MATRICES.                   CLS00410
CCI                                                                      ICLS00420
CCI     KATNO  : CATEGORY - CLASS CORRESPONDENCE                         CLS00430
CCI                                                                      ICLS00440
CCI                                                                      CLS00450
CCI                                                                      CLS00460
CCI     PURPOSE...                                                       CLS00470
CCI                                                                      CLS00480
CCI               IF AVAILABLE, THE ''B''-TRANSFORMATION MATRIX IS APPLIEDCLS00490
CCI                                                                      CLS00500
CCI     TO THE SUBCLASS MEAN VECTORS AND COVARIANCE MATRICES.            CLS00510
CCI               OBTAINS THE (MODIFIED) CHOLESKY FACTORIZATION OF THE   CLS00520
CCI     SUBCLASS COVARIANCE MATRICES. PROVIDES THE 'CONSTANT' OF         CLS00530
CCI               THE PROBABILITY DENSITY FUNCTION AND DETERMINANT FOR   CLS00540
CCI     EACH SUBCLASS. AND OBTAINS THE SUBCLASS-PAIR THRESHOLDS FOR      CLS00550
CCI               USE BY SUBR. CONTEX IN CLASSIFICATION OF INPUT SCAN     CLS00560
CCI               LINES. PUBLISHES AND OUTPUTS ON MAPTAP THE TRAINING    CLS00570
CCI               FIELD(S) INFORMATION AND THE STATISTICS FOR EACH OF THE CLS00580
CCI               TRAINING CLASSES.                                      CLS00590
CCI                                                                      ICLS00600
CCI     RETURNS...CHOLESKY FACTORIZATION OF THE INPUT COVARIANCE MATRICES CLS00610
CCI     (AFTER 'B'-TRANSFORMATION', IF APPLICABLE). SUBCLASSPAIR         CLS00620
CCI     THRESHOLDS. AND SUBCLASS STATISTICS OUTPUT ON MAPTAP.            CLS00630
      CONTINUE                                                           CLS00640
CCI                                                                      ICLS00650
CCI---------------------------------------------------------------------ICLS00660
CCI---------------------------------------------------------------------ICLS00670
C                                                                          CLS00680
C                                                                          CLS00690
C                                                                          CLS00700
      INCLUDE COMBK1,LIST                                                  CLS00710
C                                                                          CLS00720
C                                                                          CLS00730
      ----------------------------------------------------------------     CLS00740
C                                                                          CLS00750
      INCLUDE COMBK2,LIST                                                  CLS00760
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,             CLS00770
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,       CLS00780
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),           CLS00790
```

FILE: CLSFY1 FORTRAN A

```
      *               KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                  CLS00800
      *               GRPCHK(61),GROUPS(124)                                   CLS00810
       COMMON /CLASS/ APRFLG,BMCOMB,BMFEAT,BMFLG,NOCAT,THIJ1,IDATA1,           CLS00820
      *               NFILE,STATKY,CATNAM(60),                                 CLS00830
      3               CLSSYM(60),CON(60),DET(60),FLDESC,FLDINF(6),             CLS00840
      4               KCLSNA(60),NOCTCL(60),SUBCAT(60)                         CLS00850
      *     ,NOCHAN,CHNVEC(30)                                                 CLS00860
       COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,               CLS00870
      *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,         CLS00880
      *     DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                     CLS00890
      *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                         CLS00900
      *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                   CLS00910
      *   CROUNT,PRTUNT,RANDIO                                                 CLS00920
C$END                                                                         CLS00930
C                                                                             CLS00940
C                                                                             CLS00950
C     ------------------------------------------------------------------      CLS00960
C                                                                             CLS00970
      REAL CON,DET,COVMTX(VARSZ2,NOSUB2)                                      CLS00980
      REAL AVEMTX(NOFET2,NOSUB2),APRIOR(1),VEC(30),BXKXBT(120)                CLS00990
C                                                                             CLS01000
      REAL SUM,SUMTR,DUM(60),RERROR(60),APR,COV(465)                          CLS01010
C                                                                             CLS01020
      REAL  RELERR                                                            CLS01030
C                                                                             CLS01040
C     ------------------------------------------------------------------      CLS01050
C                                                                             CLS01060
      COMMON /SCRACH/SCR1(2000),SCR2(10500)                                   CLS01070
C                                                                             CLS01080
C     ------------------------------------------------------------------      CLS01090
C                                                                             CLS01100
C                                                                             CLS01110
      EQUIVALENCE  ( VEC(1),  SCR1(1)) , ( BXKXBT(1),  SCR1(31) ) ,           CLS01120
C                                                                             CLS01130
     1             ( SUM,  SCR1(151) ) , ( SUMTR,  SCR1(152) )                CLS01140
C                                                                             CLS01150
     2        , ( DUM(1),  SCR1(153)), ( APR,  SCR1(215) )                    CLS01160
C                                                                             CLS01170
     3 ,(RERROR(1),SCR1(216)),(COV(1),SCR1(280))                              CLS01180
C     ------------------------------------------------------------------      CLS01190
C                                                                             CLS01200
      DIMENSION CLSMTX(1),FLDMTX(4,NOFLD2),SUBNO(1),SUBDES(1),                CLS01210
     *  VERTEX(2,TOTVT2),KATNO(1)                                             CLS01220
C                                                                             CLS01230
      DATA BCDTWO/'2'/,BCDFOR/'4'/,DASH/'----'/                                CLS01240
      DATA LPRN/'(   '/                                                       CLS01250
C                                                                             CLS01260
C     ------------------------------------------------------------------      CLS01270
C                                                                             CLS01280
C                                                                             CLS01290
C                                                                             CLS01300
C                                                                             CLS01310
      REAL  BMATR(BMCOMB,BMFEAT)                                              CLS01320
C                                                                             CLS01330
      REAL AVENEW(BMCOMB,BMFEAT),COVNEW(BMFLG,NOSUB2)                         CLS01340
C                                                                             CLS01350
C     ------------------------------------------------------------------      CLS01360
C                                                                             CLS01370
C                                                                             CLS01380
C                                                                             CLS01390
      INCLUDE  COMBK6,LIST                                                    CLS01400
C                                                                             CLS01410
C     ------------------------------------------------------------------      CLS01420
C                                                                             CLS01430
C                                                                             CLS01440
C                                                                             CLS01450
C     HEADER RECORD  NO. 2 FOR MAPTAP                                         CLS01460
C                                                                             CLS01470
      IF (NOCAT .GT. 0)  NOCAT1 = NOCAT                                       CLS01480
      IF ( NOCAT .LE. 0  ) NOCAT1 = NOCLS2                                    CLS01490
      WRITE(MAPTAP)(CATNAM(I),I=1,NOCAT1),(CLSMTX(I),I=1,NOCLS2),             CLS01500
     1             (SUBNO(I),I=1,NOCLS2),(SUBDES(I),I=1,NOSUB2),            CLS01510
     2            ((FLDMTX(I,J),I=1,4),J=1,NOFLD2),((VERTEX(I,J),I=1,2),    CLS01520
     3             J=1,TOTVT2),(SUBCAT(I),I=1,NOSUB2),(CLSVC2(I),           CLS01530
     4                  I=1,NOSUB2),(KATNO(I),I=1,NOCLS2),(KEPPTS(I),       CLS01540
     5                  I=1,NOSUB2)                                         CLS01550
C                                                                             CLS01560
C     WRITE OUT TRAINING FIELD INFORMATION                                    CLS01570
C     ------------------------------------------------------------------      CLS01580
```

FILE: CLSFY1 FORTRAN A

```
C                                                                               CLS01590
  65  FORMAT(3X,'***  CLASSIFICATION STUDY  ***   MAPTAP FILE ',5X,I2//)      CLS01600
  10  CALL WRTFLD(FLDMTX,VERTEX,NOFLD2,1,CLSMTX,SUBDES)                        CLS01610
C                                                                               CLS01620
CCCC                                                                            CLS01630
CCCC    PUBLISH THE CLASSES AND CHANNELS (WITH SPECTRAL BAND) TO BE             CLS01640
CCCC    CONSIDERED IN CLASSIFICATION.                                           CLS01650
C                                                                               CLS01660
      WRITE(6,MEAN)                                                             CLS01670
      WRITE(6,65) NFILE                                                         CLS01680
      WRITE(6,20)                                                               CLS01690
   20 FORMAT( ////T27,'SUBCLASSES CONSIDERED',T90,'CHANNELS CONSIDERED'        CLS01700
     *//T21,'SYMBOL',T32,'SUBCLASS',T45,'A PRIOR',T88,'TRAINING   RECOGNCLS01710
     *ITION')                                                                   CLS01720
      II = NOSUB2                                                               CLS01730
      IF ( II .LT. NOFET2 ) II = NOFET2                                         CLS01740
      DO 30 I=1,II                                                              CLS01750
      WRITE (6,40)                                                              CLS01760
      IF (I .LE. NOSUB2) WRITE(6,50)CLSSYM(I),SUBDES(I),APRIOR(I)               CLS01770
      IF (I .LE. NOFET2) WRITE(6,60)FETVC2(I),CHNVEC(I)                         CLS01780
   30 CONTINUE                                                                  CLS01790
   40 FORMAT(' ')                                                               CLS01800
   50 FORMAT('+',T23,A1,T33,A4,T45,F7.4)                                        CLS01810
   60 FORMAT('+',T91,I2,T103,I2)                                                CLS01820
C                                                                               CLS01830
CCCC    SAVE AND PUBLISH THE MEAN AND COVARIANCE                                CLS01840
CCCC    ----------------------------------------------------------------------CLS01850
CCCC                                                                            CLS01860
CCCC                                                                            CLS01870
CCCC    IF R-MATRIX IS AVAILABLE, TRANSFORM THE COVARIANCE MATRIX AND           CLS01880
CCCC    MEAN VECTOR                                                             CLS01890
C                                                                               CLS01900
                                                                                CLS01910
      TEMPF2 = NOFET2                                                           CLS01920
      IF( RMFLG .LE. 0)  GO TO 611                                              CLS01930
      DO 610 NCLS=1,NOSUB2                                                      CLS01940
      DO  606  K=1,RMCOMB                                                       CLS01950
      RRM102 = ( R * (R-1) )/2                                                  CLS01960
      DO  603  C=1,R                                                            CLS01970
      I = RRM102 + C                                                            CLS01980
      SUMTR = 0.0                                                               CLS01990
      DO  602  J=1,NOFET2                                                       CLS02000
      SUM = 0.0                                                                 CLS02010
      DO  601  K=1,NOFET2                                                       CLS02020
      II = ( K * (K-1) )/2 + J                                                  CLS02030
      IF( J .GT. K)  II= ( J * (J-1) )/2 + K                                    CLS02040
  601 SUM = SUM + RMATR(R,K) * COVMTX(II,NCLS)                                  CLS02050
  602 SUMTR = SUMTR + RMATR(C,J) * SUM                                          CLS02060
  603 RXKXRT(I) = SUMTR                                                         CLS02070
      SUM = 0.0                                                                 CLS02080
      DO  604  J=1,NOFET2                                                       CLS02090
  604 SUM = SUM + RMATR(R,J) * AVEMTX(J,NCLS)                                   CLS02100
      VEC(R) = SUM                                                              CLS02110
  605 CONTINUE                                                                  CLS02120
      DO  606  J=1,I                                                            CLS02130
  606 COVNEW(J,NCLS) = RXKXRT(J)                                                CLS02140
      DO  607  J=1,RMCOMB                                                       CLS02150
  607 AVENEW(J,NCLS) = VEC(J)                                                   CLS02160
  610 CONTINUE                                                                  CLS02170
      VARS72 = RMFLG                                                            CLS02180
      NOFET2 = RMCOMB                                                           CLS02190
  611 CONTINUE                                                                  CLS02200
      GO TO 80                                                                  CLS02210
   70 CONTINUE                                                                  CLS02220
C                                                                               CLS02230
CCCC    OBTAIN THE SUBCLASS-PAIR THRESHOLDS, FOR USE BY CLSFY2/CONTEX           CLS02240
C                                                                               CLS02250
C                                                                               CLS02260
      IF (NOCAT .GT. 0)  GO TO 612                                              CLS02270
      NPL1 = NOFET2 + 1                                                         CLS02280
      CALL THRESH(NOSUB2,NOFET2,NPL1,APRIOR,AVEMTX,COVMTX,DET,VARS72,           CLS02290
     1            SCR2(1),SCR2(901),SCR2(1801),SCR2(1831),SCR2(1861),         CLS02300
     2      SCR2(2791),SCR1(THIJ1))                                           CLS02310
  612 CONTINUE                                                                  CLS02320
C                                                                               CLS02330
C                                                                               CLS02340
      NOFET2 = TEMPF2                                                           CLS02350
      RETURN                                                                    CLS02360
   80 III = 0                                                                   CLS02370
```

FILE: CLSFY1 FORTRAN A

```
C                                                                       CLS02380
C                                                                       CLS02390
C                                                                       CLS02400
C     ------------------------------------------------------------------CLS02410
C                                                                       CLS02420
C     OUTPUT THE ORIGINAL COVARIANCE AND MEAN MATRIX( AFTER B-TRANSFORM CLS02430
C     IF  B-MATRIX AVAILABLE) FOR ALL SUBCLASSES, ON THE CLASSIFICATION CLS02440
C     RESULTS OUTPUT FILE,  MAPTAP                                      CLS02450
C                                                                       CLS02460
C     ------------------------------------------------------------------CLS02470
C                                                                       CLS02480
C     HEADER RECORD  NO. 3 FOR MAPTAP                                   CLS02490
C                                                                       CLS02500
      WRITE(MAPTAP) ((COVMTX(I,J),I=1,VARSZ2),J=1,NOSUB2),             CLS02510
     *              ((AVEMTX(I,J),I=1,NOFET2),J=1,NOSUB2)               CLS02520
C                                                                       CLS02530
      BADFLG = 0                                                        CLS02540
C                                                                       CLS02550
      IF (STATKY.EQ.0) GO TO 180                                        CLS02560
      CNT = 7+(5+3+2*NOFET2)*((NOFET2+11)/12)                           CLS02570
      CNT = PAGSIZ/CNT                                                  CLS02580
      INC = CNT                                                         CLS02590
      DO 170 ICLAS=1,NOCLS2                                             CLS02600
      NUMSUB = SUBNO(ICLAS)                                             CLS02610
      DO 120  II=1,NUMSUB                                               CLS02620
      III = III + 1                                                     CLS02630
      IF (INC.LT.CNT) GO TO 100                                         CLS02640
      WRITE (6,HEAD)                                                    CLS02650
      WRITE(6,65)NFILE                                                  CLS02660
      INC = 0                                                           CLS02670
  100 CONTINUE                                                          CLS02680
      WRITE(6,110) CLSMTX(ICLAS),SUBDES(III)                            CLS02690
  110 FORMAT(//'  CLASS    : ',A4/'  SUBCLASS: ',A4)                    CLS02700
      DO 140 LOC=1,NOFET2,12                                            CLS02710
      STOP = LOC+11                                                     CLS02720
      IF( STOP .GT. NOFET2 ) STOP = NOFET2                              CLS02730
  140 WRITE(6,150)( AVEMTX(I,III),I=LOC,STOP)                           CLS02740
  150 FORMAT('0 MEAN:',3X,12F9.2)                                       CLS02750
C                                                                       CLS02760
      IF( BMFLG .GT. 0)  GO TO 161                                      CLS02770
      WRITE(6,160)                                                      CLS02780
  160 FORMAT(//'  COVARIANCE MATRIX:')                                  CLS02790
      GO TO 162                                                         CLS02800
  161 WRITE(6,1611)                                                     CLS02810
 1611 FORMAT(//'  COVARIANCE MATRIX (B-TRANSFORMED) :')                 CLS02820
  162 CALL WRTMTX(COVMTX(1,III),NOFET2,       BCDTWO)                   CLS02830
      INC = INC+1                                                       CLS02840
  120   CONTINUE                                                        CLS02850
  170 CONTINUE                                                          CLS02860
C                                                                       CLS02870
C                                                                       CLS02880
CCC   OBTAIN THE ''MODIFIED'' CHOLESKY DECOMPOSITION OF THE COVARIANCE  CLS02890
CCC     MATRIX FOR EACH SUBCLASS, THE DETERMINANT, AND COMPUTE THE      CLS02900
CCC   '' CONSTANT ''  TERM OF THE PROBABILITY DENSITY FUNCTION          CLS02910
CCC   =  -2 * LOG Q(I)  + LOG DETERMINANT(I)  , WHERE Q(I) IS THE       CLS02920
CCC     APRIORI PROBABILITY VALUE FOR SUBCLASS I                        CLS02930
CCC                                                                     CLS02940
CCC     PDF = Q(I) * ( DET(I)**-1/2 * E**-1/2 * (X-M)*  * K**-1 * (X-M) ) CLS02950
CCC                                                                     CLS02960
CCC     LOG PDF = -1/2 * ( CON + (X-M)*  * K**-1  * (X-M) )             CLS02970
CCC                                                                     CLS02980
CCC                                                                     CLS02990
C                                                                       CLS03000
  180  DO 195  NCLS=1,NOSUB2                                           CLS03010
C                                                                       CLS03020
CCC     TRANSFER ORIGINAL COVARIANCE MATRIX TO TEMPORARY STORAGE ( COV ) CLS03030
C                                                                       CLS03040
      DO  181  I=1,VARSZ2                                               CLS03050
  181 COV(I) = COVMTX(I,NCLS)                                           CLS03060
C                                                                       CLS03070
CCC     OBTAIN THE ''MODIFIED'' CHOLESKY FACTORIZATION OF THE           CLS03080
CCC     COVARIANCE MATRIX                                               CLS03090
C                                                                       CLS03100
      CALL MCHLSK( COVMTX(1,NCLS), NOFET2, DUM, DET(NCLS) )             CLS03110
C                                                                       CLS03120
      IF( DET(NCLS) .GT. 0.0)  GO TO 183                                CLS03130
      WRITE(6,1900)  NCLS, DET(NCLS)                                    CLS03140
 1900 FORMAT(//// 5X,'*****  CLSFY/CLSFY1/------ ---  THE COVARIANCE MATCLS03150
     1RIX,FOR SUBCLASS NO',I4,' IS EITHER SINGULAR OR NOT POSITIVE DEFINCLS03160
```

FILE: CLSFY1 FORTRAN A

```
     2ITE(//// 35X,'DETERMINANT =',F20.4///5X,'*****  TERMINATING PROGRAM CLS03170
     3 EXECUTION  *****' )                                                  CLS03180
C                                                                           CLS03190
      BADFLG = 99                                                           CLS03200
      GO TO 195                                                             CLS03210
  183 APR = APRIOR(NCLS)**(-2)                                              CLS03220
      CON(NCLS) = ALOG( APR * DET(NCLS) )                                   CLS03230
C                                                                           CLS03240
CCC                                                                         CLS03250
CCC    OBTAIN THE RELATIVE ERROR OF FACTORED MATRIX ( RATIO OF              CLS03260
CCC    EUCLIDEAN NORM OF DIFFERENCE,  K-LDL*  , TO EUCLIDEAN NORM OF  K)CLS03270
C                                                                           CLS03280
      RERROR(NCLS) = RELERR( COVMTX(1,NCLS),COV,NOFET2,VARSZ2)              CLS03290
C                                                                           CLS03300
C                                                                           CLS03310
  195 CONTINUE                                                              CLS03320
C                                                                           CLS03330
CCC   ERROR TERMINATION, IF ONE OR MORE SUBCLASSES HAVE AN INVALID          CLS03340
CCC   (SINGULAR, OR NON-POSITIVE DEFINITE) COVARIANCE MATRIX                CLS03350
C                                                                           CLS03360
      IF( BADFLG .GT. 0)  CALL EXIT                                         CLS03370
CCC                                                                         CLS03380
CCC                                                                         CLS03390
CCC                                                                         CLS03400
CCCCCC--------------------------------------------------------------------CLS03410
CCCCCC OUTPUT THE ( MODIFIED ) CHOLESKY FACTORIZATION OF THE                CLS03420
CCCCCC COVARIANCE MATRIX, ON THE CLASSIFICATION OUTPUT FILE, MAPTAP         CLS03430
CCCCCC--------------------------------------------------------------------CLS03440
CCCCCC                                                                      CLS03450
CCCCCC                                                                      CLS03460
CCC   HEADER RECORD  NO. 4 FOR MAPTAP                                       CLS03470
C                                                                           CLS03480
      WRITE(MAPTAP) ((COVMTX(I,J),I=1,VARSZ2),J=1,NOSUB2),                  CLS03490
     *              (CON(I),I=1,NOSUB2),(DET(I),I=1,NOSUB2)                 CLS03500
      IF (STATKY.EQ.0) GO TO 230                                            CLS03510
      CNT = 13 + (5+2*NOFET2) * ((NOFET2+11)/12)                            CLS03520
      CNT = PAGSIZ/CNT                                                      CLS03530
      INC = CNT                                                             CLS03540
C                                                                           CLS03550
      III = 0                                                               CLS03560
      DO 210   L=1,NOCLS2                                                   CLS03570
      NSUBCL = SUBNO(L)                                                     CLS03580
      DO 210   I=1,NSUBCL                                                   CLS03590
      III = III + 1                                                         CLS03600
      IF (INC.LT.CNT) GO TO 200                                             CLS03610
      WRITE (6,HEAD)                                                        CLS03620
      WRITE(6,65)NFILE                                                      CLS03630
      INC =0                                                                CLS03640
  200 WRITE(6,220)CLSMTX(L),L,SUBDES(III),III,DET(III),CON(III)             CLS03650
      CALL WRTMTX(COVMTX(1,III),NOFET2,RCDFOR)                              CLS03660
C                                                                           CLS03670
      WRITE(6,205)RERROR(III)                                               CLS03680
  205 FORMAT(1X,'** RELATIVE ERROR  ( EUCLIDEAN NORM (K-LDL*)/EUCLIDEAN CLS03690
     1NORM  K )  =' , F15.8 // )                                            CLS03700
C                                                                           CLS03710
      INC = INC+1                                                           CLS03720
  210 CONTINUE                                                              CLS03730
  220 FORMAT(1H0// T50,'MULTISPECTRAL CHARACTERISTICS FOR'//T57,A4,         CLS03740
     1' ( CLASS',I3,' )',/T56,A4,2X,' ( SUBCLASS',I3,' )'//                 CLS03750
     2 1H0,'DETERMINANT =',F25.4 / 1H0,          'PROB. DENSITY FUCLS03760
     2NCTION - CONSTANT TERM=',F10.4// 1H0,'COVARIANCE MATRIX  (CHOLESKYCLS03770
     3 FACTORIZATION) :' / )                                                CLS03780
C                                                                           CLS03790
C                                                                           CLS03800
  230 CONTINUE                                                              CLS03810
C                                                                           CLS03820
      GO TO 70                                                              CLS03830
       END                                                                  CLS03840
```

```
FILE: CLSFY2   FORTRAN  A

      SUBROUTINE CLSFY2(COVMTX,AVEMTX,FLDMTX,CLSMTX,SUBDES,SUBNO,        CLS00010
     *           KATNO,BMATRX)                                           CLS00020
C                                                                        CLS00030
C                                                                        CLS00040
      IMPLICIT INTEGER (A-H,O-Z)                                         CLS00050
C                                                                        CLS00060
CI-----------------------------------------------------------------------ICLS00070
CI-----------------------------------------------------------------------ICLS00080
CI                                                                       ICLS00090
CI    CALL...   CALL CLSFY2(COVMTX,AVEMTX,FLDMTX,RUNMTX,CLSMTX,BMATRX)   CLS00100
CI                                                                      ICLS00110
CI    ARGS...   COVMTX : LOCATION OF THE COVARIANCE MATRICES            CLS00120
CI                       ( IN ''SYMETTRIC'' STORAGE) FOR NOCLS2 TRAININGCLS00130
CI                       CLASSES                                        CLS00140
CI                                                                      CLS00150
CI              AVEMTX : LOCATION OF THE MATRIX OF TRAINING CLASS MEAN  CLS00160
CI                       VECTORS (NOFET2 MEANS PER CLASS)               CLS00170
CI                                                                      CLS00180
CI              FLDMTX : LOCATION OF MATRIX OF TRAINING FIELD(S)        CLS00190
CI                       INFORMATION                                    CLS00200
CI                                                                      CLS00210
CI              CLSMTX : LOCATION OF MATRIX OF TRAINING CLASS NAMES     CLS00220
CI    BMATRX : B-TRANSFORMATION MATRIX,IF AVAILABLE                     CLS00230
CI                                                                      CLS00240
CI    SUBDES : LOCATION OF SUBCLASS NAMES                               CLS00250
CI                                                                      CLS00260
CI    SUBNO  : LOCATION OF NO. OF SUBCLASSES IN EACH CLASS              CLS00270
CI                                                                      CLS00280
CI    KATNO  : CATEGORY - CLASS CORRESPONDENCE  ARRAY                   CLS00290
CI                                                                      CLS00300
CI              BMATRX : LOCATION OF THE ''B''-TRANSFORMATION MATRIX,   CLS00310
CI                       IF AVAILABLE, FOR TRANSFORMATION OF INPUT      CLS00320
CI                       SAMPLE VECTOR IN SUBR. CONTEX                  CLS00330
CI                                                                     ICLS00340
CI                                                                     ICLS00350
CI    PURPOSE...  CLASSIFIES THE SET OF SAMPLES ( MULTI-CHANNEL DATA    CLS00360
CI                POINTS) ON EACH SCAN LINE OF THE SET OF SCAN LINES   CLS00370
CI                PRESCRIBED BY THE ''FIELD DEFINITION'' CARD INPUT TO CLS00380
CI                THE CLASSIFICATION PROCESSOR. CLASSIFICATION IS      CLS00390
CI                PERFORMED BY THE METHOD OF MAXIMUM LIKELIHOOD        CLS00400
CI                (MINIMUM PROBABILITY OF MIS-CLASSIFICATION) IN       CLS00410
CI                SUBROUTINE  CONTEX . THE DIMENSIONALITY OF THE SAMPLE CLS00420
CI                IS PRESCRIBED BY CONTROL CARD INPUT TO THE PROCESSOR CLS00430
CI                (BMCOMB) OF CHANNELS (BMFEAT) IN THE ''B'' - MATRIX, CLS00440
CI                ( ''CHANNELS'' ) OR BY THE NO. OF LINEAR COMBINATIONS CLS00450
CI                IF AVAILABLE .                                       CLS00460
CI                                                                     CLS00470
CI                                                                     CLS00480
CI                                                                     ICLS00490
CI    RETURNS... 1. IF STANDARD CLASSIFIER IS USED, THE SUBCLASS       CLS00500
CI               NUMBER AND PROBABILITY DENSITY FUNCTION               CLS00510
CI                VALUE  FOR EACH POINT OF EVERY SCAN LINE OF THE FIELD CLS00520
CI                IS OUTPUT ON THE CLASSIFICATION OUTPUT FILE , MAPTAP. CLS00530
CI               2. IF CATEGORY CLASSIFIER IS USED, THE SUBCLASS       CLS00540
CI               NUMBER OF THE SUBCLASS WITH THE LARGEST PROBABILITY   CLS00550
CI               DENSITY FUNCTION WITHIN THE CHOSEN CATEGORY AND THE   CLS00560
CI               PROBABILITY DENSITY FUNCTION VALUE OF THE CHOSEN      CLS00570
CI               CATEGORY FOR EACH POINT OF EVERY SCAN LINE OF THE     CLS00580
CI               FIELD IS OUTPUT ON THE CLASSIFICATION OUTPUT FILE.    CLS00590
CI                                                                     CLS00600
CI                                                                     ICLS00610
CI-----------------------------------------------------------------------ICLS00620
CI-----------------------------------------------------------------------ICLS00630
C                                                                        CLS00640
C                                                                        CLS00650
      INCLUDE  COMBK6,LIST                                               CLS00660
C                                                                        CLS00670
      INCLUDE COMBK1,LIST                                                CLS00680
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           CLS00690
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLDSV2,VERTX2,     CLS00700
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         CLS00710
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              CLS00720
     *              GRPCHK(61),GROUPS(124)                               CLS00730
      COMMON /CLASS/ APPFLG,BMCOMB,BMFEAT,BMFLG,NOCAT,THIJ1,IDATA1,      CLS00740
     *               NFILE,STATKY,CATNAM(60),                            CLS00750
     3               CLSSYM(60),CON(60),DET(60),FLDESC,FLDINF(6),        CLS00760
     4               KCLSNA(60),NOCTCL(60),SUBCAT(60)                    CLS00770
     *     ,NOCHAN,CHNVEC(30)                                            CLS00780
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          CLS00790
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: CLSFY2 FORTRAN A

```
      *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,     CLS00800
      *     DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 CLS00810
      *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                     CLS00820
      *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,               CLS00830
      *   CHOUNT,PRTUNT,HANDIO                                             CLS00840
C$END                                                                      CLS00850
C                                                                          CLS00860
      REAL CON,DET,      VR(1000),COVMTX(VARSZ2,NOSUB2),AVEMTX(NOFET2,    CLS00870
     *       NOSUB2),BMATRX                                              CLS00880
C                                                                          CLS00890
C                                                                          CLS00900
C                                                                          CLS00910
      DIMENSION FLDMTX(4,NOFLD2),CLSMTX(1),VERTCS(22)                      CLS00920
     1   ,COL(3,110),OUT(110),IR(1000)                                     CLS00930
     *   ,SUBNO(1),SUBDES(1),KATNO(1)                                      CLS00940
C                                                                          CLS00950
C                                                                          CLS00960
C     ***  NOTE : THE IDATA ARRAY IS USED FOR INPUT OF THE SCAN LINE TO    CLS00970
C                 BE CLASSIFIED TO SUBR. CONTEX, AND ALSO TO STORE THE     CLS00980
C                 CLASSIFIED SCAN LINE, BY CONTEX. THE ''IR'' AND ''VR''   CLS00990
C                 ARRAYS, EQUIVALENCED TO THE IDATA ARRAY, ARE             CLS01000
C                 LOCATIONS USED BY CONTEX TO OUTPUT THE CLASSIFIED        CLS01010
C                 SCAN LINE AND THE PROBABILITY DENSITY FUNCTION           CLS01020
C                 VALUES, RESPECTIVELY.                                    CLS01030
C                                                                          CLS01040
C                                                                          CLS01050
C                                                                          CLS01060
C                                                                          CLS01070
C     ---------------------------------------------------------------------CLS01080
C                                                                          CLS01090
      INCLUDE COMPK2,LIST                                                  CLS01100
C                                                                          CLS01110
      EQUIVALENCE      (FLDINF(1), LINSTR), (FLDINF(2),LINEND),            CLS01120
     1                 (FLDINF(3),LININC), (FLDINF(4), SAMSTR),            CLS01130
     2                 (FLDINF(5),SAMEND),(FLDINF(6),SAMINC),              CLS01140
     3                 (FLDINF(7),FLDTYP)                                  CLS01150
C                                                                          CLS01160
C     ---------------------------------------------------------------------CLS01170
C                                                                          CLS01180
C                                                                          CLS01190
C     ---------------------------------------------------------------------CLS01200
C                                                                          CLS01210
      COMMON /SCRACH/ IDATA(12500)                                         CLS01220
C                                                                          CLS01230
C                                                                          CLS01240
C     ---------------------------------------------------------------------CLS01250
C                                                                          CLS01260
C                                                                          CLS01270
C                                                                          CLS01280
C                                                                          CLS01290
C     ---------------------------------------------------------------------CLS01300
C                                                                          CLS01310
C                                                                          CLS01320
      DATA  LINMAX/1000/, ENDBCD/'SEND'/, DASH/'----'/                     CLS01330
C                                                                          CLS01340
C                                                                          CLS01350
C                                                                          CLS01360
      EQUIVALENCE (IR,COL)                                                 CLS01370
      DATA LPRN/'(   '/,BLANK/'    '/                                      CLS01380
C                                                                          CLS01390
C     ---------------------------------------------------------------------CLS01400
C     ---------------------------------------------------------------------CLS01410
C                                                                          CLS01420
C                                                                          CLS01430
C                                                                          CLS01440
C                                                                          CLS01450
C     READ THE FIELD DESCRIPTION CARD,CONTAINING LINE-SAMPLE COORDINATES   CLS01460
C     OF THE FIELD TO BE CLASSIFIED                                        CLS01470
C     ---------------------------------------------------------------------CLS01480
C                                                                          CLS01490
      CALL TAPHDR(DATAPE,DATFIL)                                           CLS01500
C                                                                          CLS01510
   10 CONTINUE                                                             CLS01520
      PTSTHS = 0                                                           CLS01530
      ICK = LAREAD(FLDESC,VERTCS,FLDINF,NC)                                CLS01540
      IF ( ICK .EQ. 0 ) GO TO 160                                          CLS01550
      IF (ICK .EQ. -1 .OR. ICK .EQ. -2)  GO TO 10                          CLS01560
C                                                                          CLS01570
   30 CONTINUE                                                             CLS01580
```

```
FILE: CLSFY2   FORTRAN  A

      WRITE(6,HEAD)                                                          CLS01590
      NC1 = 2*NC                                                             CLS01600
      NV = NC - 1                                                            CLS01610
      WRITE(6,35)FLDESC,NV,FLDINF(6),FLDINF(3),(LPRN,VERTCS(I),              CLS01620
     *    VERTCS(I+1),I=1,NC1,2)                                             CLS01630
   35 FORMAT('//// 156,'AREA OF CLASSIFICATION'///  36X,'SAMPLE',3X,        CLS01640
     1       'LINE'. / 5X,'FIELD NAME',3X,'NO. OF VERTICES',3X,'INC.  ',    CLS01650
     2       3X,'INC.',30X,'VERTICES' / 7X,A4,14X,I2,10X,I2,6X,I2,5X,       CLS01660
     3       5(A1,I4,',',I4,')',2X) / (52X,5(A1,I4,',',I4,')',2X)/))        CLS01670
C                                                                            CLS01680
CCC   SAVE CLASSIFIED FIELD INFORMATION ON MAPTAP                            CLS01690
CC    ---------------------------------------------------------------------CLS01700
C                                                                            CLS01710
   50 CONTINUE                                                               CLS01720
C*    INITIALIZE TAPE HEADING FOR THIS FIELD                                 CLS01730
      CALL FLDINT(FLDINF,CHNVEC,NUCHAN)                                      CLS01740
C                                                                            CLS01750
CC                                                                           CLS01760
C                                                                            CLS01770
      LINES = ( LINEND-LINSTR)/LININC + 1                                    CLS01780
C                                                                            CLS01790
      PTS = ( SAMEND-SAMSTR)/SAMINC + 1                                      CLS01800
C                                                                            CLS01810
      IF (PTS .GT. LINMAX) WRITE(6,55)                                       CLS01820
   55 FORMAT( // 5X,'WIDTH OF RECTANGULAR FIELD SURROUNDING CLASSIFICATI     CLS01830
     *ON FIELD CONNOT EXCEED 1000 POINTS.')                                  CLS01840
      IF (PTS .GT. LINMAX)  CALL CMERR                                       CLS01850
C                                                                            CLS01860
      SCANLN = PTS * NOFET2                                                  CLS01870
      IF (SCANLN .GT. (12500-IDATA1+1)) GO TO 170                            CLS01880
C                                                                            CLS01890
CCC                                                                          CLS01900
CCC   FIELD RECORD  FOR MAPTAP                                               CLS01910
C                                                                            CLS01920
      WRITE(MAPTAP)(FLDINF(I),I=1,6),PTS,LINES,FLDESC,NC,                    CLS01930
     *             (VERTCS(I),I=1,NC),(VERTCS(I+NC),I=1,NC)                  CLS01940
C                                                                            CLS01950
CCC   CLASSIFY THE FIELD                                                     CLS01960
CC    ---------------------------------------------------------------------CLS01970
      CALL SETMRG(66,0,66)                                                   CLS01980
C                                                                            CLS01990
      WRITE (6,HEAD)                                                         CLS02000
      CALL MAPHDG(NOCAT,CLSSYM,CATNAM,KATNO,CLSMTX,SUBNO,SUBDES)             CLS02010
      WRITE(6,800)                                                           CLS02020
  800 FORMAT (///)                                                           CLS02030
      J = 0                                                                  CLS02040
      DO 80 I=SAMSTR,SAMEND,SAMINC                                           CLS02050
      J = J+1                                                                CLS02060
      COL(1,J) = I/100                                                       CLS02070
      COL(2,J) = MOD(I,100)/10                                               CLS02080
      COL(3,J)=MOD(I,10)                                                     CLS02090
      IF (J.EQ.110) GO TO 90                                                 CLS02100
   80 CONTINUE                                                               CLS02110
   90 DO 100 I=1,3                                                           CLS02120
  100 WRITE (6,110)(COL(I,K),K=1,J)                                          CLS02130
  110 FORMAT('  ',9X,110I1)                                                  CLS02140
      WRITE(6,115)                                                           CLS02150
  115 FORMAT( / )                                                            CLS02160
C                                                                            CLS02170
C                                                                            CLS02180
      ILINE=LINSTR-LININC                                                    CLS02190
      DO 140 JLINE=1,LINES                                                   CLS02200
      CALL LINERD(IDATA(IDATA1),ENDTAP)                                      CLS02210
      IF ( ENDTAP .EQ. -1)  GO TO 150                                        CLS02220
      ILINE=ILINE+LININC                                                     CLS02230
C                                                                            CLS02240
C     ***  CLASSIFY THE SCAN LINE IN IDATA                                   CLS02250
CCCCC                                                                        CLS02260
CCCCC                                                                        CLS02270
CCCCC                                                                        CLS02280
CC     CALL STANDARD CLASSIFIER                                              CLS02290
C                                                                            CLS02300
      IF (NOCAT .LE. 0) CALL CONTEX(NOFET2,NOSUB2,PTS,AVEMTX,COVMTX,         CLS02310
     *    BMATRX,IDATA(IDATA1),VERTCS,NC,IR,VR,ILINE,IDATA(THIJ1))           CLS02320
C                                                                            CLS02330
CCC   CALL CATEGORY CLASSIFIER                                               CLS02340
CCC                                                                          CLS02350
      IF (NOCAT .GT. 0)  CALL CATGRY(NOFET2,       PTS,AVEMTX,COVMTX,        CLS02360
     *   IR,VR,BMATRX,IDATA(IDATA1),ILINE,VERTCS,NC,PTSTHS)                  CLS02370
```

FILE: CLSFY2 FORTRAN A

```
C                                                                       CLS02380
                                                                        CLS02390
      DO 120 JKL=1,J                                                    CLS02400
      ISYMB = IR(JKL)                                                   CLS02410
      IF ( ISYMB .EQ. 0 ) OUT(JKL) = BLANK                              CLS02420
      IF ( ISYMB .NE. 0 ) OUT(JKL) = CLSSYM(ISYMB)                      CLS02430
  120 CONTINUE                                                          CLS02440
      WRITE (6,130)ILINE,(OUT(I),I=1,J)                                 CLS02450
  130 FORMAT(' ',I5,4X,110A1)                                           CLS02460
C                                                                       CLS02470
CCCCC SAVE CLASSIFIED INFORMATION ON MAPTAP ---------------------------CLS02480
CCCCC                                                                   CLS02490
CCCCC                                                                   CLS02500
CCCCC DATA  RECORD  FOR MAPTAP                                          CLS02510
CCCCC                                                                   CLS02520
      WRITE(MAPTAP)ILINE,(IR(I),I=1,PTS),(VR(I),I=1,PTS)                CLS02530
  140 CONTINUE                                                          CLS02540
      IF (PTSTHS .GT. 0)  WRITE(6,145) PTSTHS                           CLS02550
  145 FORMAT(////9X,'AS THE COMPUTER CANNOT EXPONENTIATE A NUMBER SMALLECLS02560
     *R THAN EXP(-88),',I6,' PTS WERE NOT CLASSIFIED IN THIS FIELD')    CLS02570
  150 ILINE = 0                                                         CLS02580
C                                                                       CLS02590
CCC   END OF  FIELD RECORD FOR MAPTAP                                   CLS02600
CCC                                                                     CLS02610
CCC   WRITE(MAPTAP)ILINE,(IR(I),I=1,PTS),(VR(I),I=1,PTS)               CLS02620
C                                                                       CLS02630
C     GO TO 10                                                          CLS02640
C                                                                       CLS02650
CCCCC GO HOME ---------------------------------------------------------CLS02660
CCCCC                                                                   CLS02670
CCCCC                                                                   CLS02680
  160 PTS = 0                                                           CLS02690
C                                                                       CLS02700
CCC   END OF RUN RECORD FOR MAPTAP                                      CLS02710
CCC                                                                     CLS02720
      WRITE(MAPTAP)(FLDINF(I),I=1,6),PTS,LINES,FLDESC,NC,               CLS02730
     *             (VERTCS(I),I=1,NC),(VERTCS(I+NC),I=1,NC)             CLS02740
C                                                                       CLS02750
      CALL SFTMRG(66,4,62)                                              CLS02760
C                                                                       CLS02770
      RETURN                                                            CLS02780
C                                                                       CLS02790
  170 WRITE(6,175)                                                      CLS02800
  175 FORMAT(//5X,'TOO MUCH DATA REQUESTED.  DO ONE OF THE FOLLOWING:'  CLS02810
     *,/7X,'1)  FOR STANDARD CLASSIFIER -',/11X,'REDUCE PARAMETERS SUCH THCLS02820
     *AT',/11X,'(NO. OF SUBCLASSFS-1)*(NO. OF SUBCLASSFS-2)/2 * NO. OF SUCLS02830
     *BCLASSFS * (PTS PER SCAN LINE)*(NO. OF CHANNELS) < 12500',/7X,     CLS02840
     *'2)  FOR CATEGORY CLASSIFIER -',/11X, 'REDUCE DATA REQUESTED SUCH TCLS02850
     *HAT',/ 11X, '(PTS PER SCAN LINE)*(NO. OF CHANNELS) < 12500.')      CLS02860
      CALL CMERR                                                        CLS02870
      END
```

FILE: CONTEX FORTRAN A

```
      SUBROUTINE CONTEX(NCHAN,NC,NPTS,AVE,COV,BMATR,IDATA,VERTCS,VT,     CON00010
     *   IR,VR,ILINE,TH)                                                 CON00020
C                                                                        CON00030
CCCCC ------------------------------------------------------------------CON00040
C                                                                        CON00050
C     INCLUDE COMBK2.LIST                                                CON00060
      COMMON /CLASS/ APRFLG,BMCOMB,BMFEAT,BMFLG,NOCAT,THIJ1,IDATA1,      CON00070
     *               NFILE,STATKY,CATNAM(60),                            CON00080
     3               CLSSYM(60),CON(60),DET(60),FLDESC,FLDINF(6),        CON00090
     4               KCLSNA(60),NOCTCL(60),SUBCAT(60)                    CON00100
     *     ,NOCHAN,CHNVEC(30)                                            CON00110
CSEND                                                                    CON00120
C                                                                        CON00130
C                                                                        CON00140
      INTEGER BMFLG,BMCOMB,BMFEAT                                        CON00150
     *   ,SAMSTR,SAMINC,SAMEND                                           CON00160
      INTEGER VERTCS,VT,FL                                               CON00170
C                                                                        CON00180
CCCCC ------------------------------------------------------------------CON00190
C                                                                        CON00200
C                                                                        CON00210
CCCCC ------------------------------------------------------------------CON00220
C                                                                        CON00230
CCCCC ------------------------------------------------------------------CON00240
      LOGICAL  BMFLAG  ,  KD1                                            CON00250
C                                                                        CON00260
C     DIMENSION IDATA(1),AVE(1),COV(1),DATA(30)                          CON00270
     1,JORDER(60),BMATR(BMCOMB,BMFEAT),JTEST(60),IR(1),VR(1),            CON00280
     *   NCNT(60),DM(30),FL(22),VERTCS(1)                                CON00290
C                                                                        CON00300
C     DIMENSION TH(1)                                                    CON00310
CCCCC ------------------------------------------------------------------CON00320
C                                                                        CON00330
C  USING THE **MODIFIED** CHOLESKY DECOMPOSITION OF THE COVAR. MATRIX ,  CON00340
C  THE ROUTINE COMPUTES THE PROBABILITY DENSITY FUNCTION AND OBTAINS THE  CON00350
C  MAXIMUM PROBABILITY ( **MAXIMUM LIKELIHOOD** ) OVER ALL CLASSES OF    CON00360
C  THE SET OF TRAINING CLASSES, FOR ASSIGNING A CLASS TO EACH RESOLUTION  CON00370
C  ELEMENT ( **PIXEL** ) IN THE INPUT SAMPLE VECTOR (INPUT SCAN LINE ).   CON00380
C                                                                        CON00390
C THE PRE-COMPUTED CLASS-PAIR THRESHOLDS, IN  TH , ARE USED TO MINIMIZE  CON00400
C THE NUMBER OF CLASS PROBABILITY DENSITY FUNCTIONS (PDF) COMPUTED       CON00410
C TO OBTAIN THE MAXIMUM PDF FOR A GIVEN SAMPLE BEING CLASSIFIED .        CON00420
C                                                                        CON00430
C THE  IRVR  ARRAY IS USED TO RETURN THE CLASS NUMBER AND PDF VALUE      CON00440
C FOR EACH SAMPLE ON THE INPUT SCAN LINE                                 CON00450
C                                                                        CON00460
C                                                                        CON00470
C                                                                        CON00480
C                                                                        CON00490
C                                                                        CON00500
C                                                                        CON00510
C!-----------------------------------------------------------------------!CON00520
C!-----------------------------------------------------------------------!CON00530
C!                                                                       !CON00540
C!    CALL...    CALL CONTEX(NCHAN,NC,NPTS,AVE,COV,BMATR,IDATA,VERTCS,   !CON00550
C!                           VT,IR,VR,ILINE)                           !CON00560
C!                                                                       !CON00570
C!    ARGS..     NCHAN : THE NUMBER OF CHANNELS TO BE USED IN            !CON00580
C!                       CLASSIFICATION OF EACH DATA SAMPLE.            !CON00590
C!                                                                       !CON00600
C!       NC : THE NUMBER OF SUBCLASSES ( TRAINING CLASSES,               !CON00610
C!     CONTINUE                                                          !CON00620
C!                       FOR WHICH COVARIANCE MATRICES, MEAN VECTORS,    !CON00630
C!                       AND CLASS-PAIR THRESHOLDS ARE AVAILABLE -       !CON00640
C!                       REPRESENTS THE MAXIMUM NUMBER OF POSSIBILITIES  !CON00650
C!                       FOR CLASSIFICATION OF EACH DATA SAMPLE.)        !CON00660
C!                                                                       !CON00670
C!                 NPTS : THE NUMBER OF INPUT DATA POINTS (PER CHANNEL)  !CON00680
C!              ON THE RECTANGULAR FIELD                                 !CON00690
C!                                                                       !CON00700
C!    VERTCS : VERTICES OF FIELD TO BE CLASSIFIED                        !CON00710
C!                                                                       !CON00720
C!        VT : NO. OF VERTICES OF FIELD TO BE CLASSIFIED                 !CON00730
C!                                                                       !CON00740
C!        IR : WILL CONTAIN THE CLASSIFIED DATA                          !CON00750
C!                                                                       !CON00760
C!        VR : WILL CONTAIN THE CORRESPONDING PDF OF IR ARRAY            !CON00770
C!                                                                       !CON00780
C                                                                        CON00790
```

FILE: CONTEX FORTRAN A

```
C      INLINE : SCAN LINE NUMBER FROM DATA TAPE                          CON00800
C                    TO BE CLASSIFIED ON EACH INPUT SCAN LINE.           CON00810
C                                                                        CON00820
C                 AVE : THE VECTOR OF ''NCHAN'' MEANS( OR ''BMCOMB''     CON00830
C                       MEANS IF THE ''B-TRANSFORMATION'' MATRIX         CON00840
C                       HAS BEEN APPLIED).                               CON00850
C                                                                        CON00860
C                 COV : COVARIANCE MATRIX ( AFTER CHOLESKY               CON00870
C                       FACTORIZATION) - ACTUAL DIMENSION OF COV. MATRIX CON00880
C                       IS DEPENDENT ON WHETHER IT HAS BEEN              CON00890
C                       ''B-TRANSFORMED'' .                              CON00900
C                                                                        CON00910
C      CONTINUE                                                          CON00920
C                 CON :  THE ''CONSTANT'' OF THE PROB. DENSITY FUNCTION  CON00930
C                        FOR EACH CLASS                                  CON00940
C                                                                        CON00950
C               BMATR : THE  ''B'' -TRANSFORMATION MATRIX                CON00960
C                                                                        CON00970
C               IDATA : THE SCAN LINE TO BE CLASSIFIED                   CON00980
C                                                                        CON00990
C                                                                        CON01000
C    OTHER...                                                            CON01010
C                                                                        CON01020
C               BMFLG :  A FLAG TO INDICATE THE PRESENCE OR ABSENCE OF   CON01030
C                        ''B-TRANSFORMATION''  MATRIX .                  CON01040
C                                                                        CON01050
C              BMCOMB : THE NUMBER OF LINEAR COMBINATIONS OF  BMFEAT     CON01060
C                       CHANNELS, IN THE ''B''-MATRIX.                   CON01070
C                                                                        CON01080
C-------------------------------------------------------------------------CON01090
C-------------------------------------------------------------------------CON01100
C                                                                        CON01110
C                                                                        CON01120
      EQUIVALENCE (FLDINF(1),LINSTR) , (FLDINF(2),LINEND),              CON01130
     *            (FLDINF(3),LININC),  (FLDINF(4),SAMSTR) ,            CON01140
     *            (FLDINF(5),SAMEND),  (FLDINF(6),SAMINC)              CON01150
C                                                                        CON01160
      BMFLAG = BMFLG .GT. 0                                             CON01170
      IF( BMFLAG )  GO TO 1000                                          CON01180
      NV = NCHAN                                                        CON01190
      GO TO 1001                                                        CON01200
 1000 NV = BMCOMB                                                       CON01210
 1001 KD1 = NV .EQ. 1                                                   CON01220
C                                                                        CON01230
      NPC = ( NV * (NV+1))/2                                            CON01240
C                                                                        CON01250
      NM1 = NC - 1                                                      CON01260
      DO  1  I=1,NC                                                     CON01270
    1 JORDER(I) = I                                                     CON01280
C                                                                        CON01290
      IPT1 = 1                                                          CON01300
C                                                                        CON01310
    2 JJ = IPT1                                                         CON01320
      DO  100  I=1,NC                                                   CON01330
  100 NCNT(I) = 0                                                       CON01340
C                                                                        CON01350
      DO 110  I=1,NPTS                                                  CON01360
      IR(I) = 0                                                         CON01370
  110 VR(I) = 0.0                                                       CON01380
      CALL FDLINT(VERTCS,VT,FL,ILINE,IPTS,NI)                           CON01390
      DO 250  LL=1,NI,2                                                 CON01400
      IB = (FL(LL) - SAMSTR) / SAMINC + 1                               CON01410
      IE = (FL(LL+1) - SAMSTR) / SAMINC + 1                             CON01420
      IF (MOD(SAMSTR,SAMINC) .NE. MOD(FL(LL),SAMINC)) IB = IB + 1       CON01430
      IF (IB .GT. IE)  GO TO 250                                        CON01440
  115 DO 200  II=IB,IE                                                  CON01450
C                                                                        CON01460
      DO  3  I=1,NC                                                     CON01470
    3 JTEST(I) = I                                                      CON01480
C                                                                        CON01490
CC     FLOAT THE DATA SAMPLE, AND APPLY THE ''B''-MATRIX,IF AVAILABLE    CON01500
CC                                                                       CON01510
      IF ( BMFLAG )  GO TO 5                                            CON01520
C                                                                        CON01530
      DO  4  I=1,NCHAN                                                  CON01540
      IDUM = NPTS * (I - 1) + II                                        CON01550
    4 DATA(I) = IDATA(IDUM)                                             CON01560
      GO TO 15                                                          CON01570
C                                                                        CON01580
```

FILE: CONTEX FORTRAN A

```
  5   DO  7  I=1,RMCOMB                                                    CON01590
      SUM = 0.0                                                            CON01600
      DO  6  K=1,NCHAN                                                     CON01610
      INDUM = NPTS * (K - 1) + I                                           CON01620
      FDATA = IDATA(INDUM)                                                 CON01630
  6   SUM = SUM + BMATR(I,K) * FDATA                                       CON01640
C                                                                          CON01650
  7   DATA(I) = SUM                                                        CON01660
C                                                                          CON01670
CCC CALC. THE LIKLIHOOD VALUES (PROBABILITIES, IF YOU WILL)                CON01680
C                                                                          CON01690
  15  TFMAX = -1.0E35                                                      CON01700
      JI = 0                                                               CON01710
CCC                                                                        CON01720
CCC   COMPUTE THE PDF FOR CLASS JJ                                         CON01730
CCC                                                                        CON01740
  20  JJM1 = JJ - 1                                                        CON01750
C                                                                          CON01760
      LC = NPC * JJM1                                                      CON01770
CC    LOCATION (-1) OF COV. MATRIX, CLASS JJ                               CON01780
CC                                                                         CON01790
      IMN = JJM1 * NV                                                      CON01800
CC    LOCATION (-1) OF MEAN VECTOR, CLASS JJ                               CON01810
CC                                                                         CON01820
      KM = IMN + 1                                                         CON01830
      S = DATA(1) - AVE(KM)                                                CON01840
      DM(1) = S                                                            CON01850
      LC = LC + 1                                                          CON01860
      TF = CON(JJ) + ( S * S )/COV(LC)                                     CON01870
      IF(KD1)  GO TO 146                                                   CON01880
CCC                                                                        CON01890
CCC   LOOP FOR COMPUTING THE KD-TH ELEMENT OF Y (=L**-1 * (X-M) ) , WHICH  CON01900
CCC                                                  IS STORED IN S        CON01910
CC                                                                         CON01920
      DO  145  KD=2,NV                                                     CON01930
      KM = IMN + KD                                                        CON01940
      S = DATA(KD) - AVE(KM)                                               CON01950
      J1 = KD - 1                                                          CON01960
      DO  140  LD = 1,J1                                                   CON01970
      LC = LC + 1                                                          CON01980
 140  S = S - COV(LC) * DM(LD)                                             CON01990
C                                                                          CON02000
      DM(KD) = S                                                           CON02010
C                                                                          CON02020
      LC = LC + 1                                                          CON02030
C                                                                          CON02040
CCC   COMPUTE THE KD-TH TERM IN :  1/2 * Y * D**-1 * Y                     CON02050
CCC                              = 1/2 * (X-M) * K**-1 * (X-M)             CON02060
CCC                                                                        CON02070
 145  TF = TF + ( S * S )/COV(LC)                                          CON02080
C                                                                          CON02090
 146  TF = -.5 * TF                                                        CON02100
CCC                                                                        CON02110
CCC                                                                        CON02120
CCC   TEST THIS SAMPLE  PDF FOR CLASS JJ - IF GREATER THAN THE PDF FOR     CON02130
CCC   CURRENT CLASS IC, SET IC = JJ, TEST THE CLASS-PAIR THRESHOLDS FOR    CON02140
CCC   OTHER POSSIBLE CLASSES FOR THIS SAMPLE - IF THE PDF FOR CURRENT      CON02150
CCC   CLASS IS EXCEEDED BY ANY CLASS-PAIR THRESHOLD,EVALUATE THE PDF       CON02160
CCC   FOR THE OTHER CLASS OF THE CLASS-PAIR, AND REPEAT THE TEST FOR       CON02170
CCC   MAX. PDF. IF ALL PDF'S FOR WHICH CLASS-PAIR THRESHOLDS HAVE          CON02180
CCC   DICTATED TO BE TESTED HAVE BEEN EVALUATED, AND THE CURRENT PDF       CON02190
CCC   FOR CLASS IC IS THE MAX. PDF OF ALL PDF'S EVALUATED,                 CON02200
CCC   CLASSIFY THE SAMPLE AS CLASS IC                                      CON02210
CCC                                                                        CON02220
      IF( TF .LE. TFMAX)  GO TO 149                                        CON02230
      TFMAX = TF                                                           CON02240
      IC = JJ                                                              CON02250
 149  JTEST(JJ) = 0                                                        CON02260
 150  JI = JI + 1                                                          CON02270
      IF( JI .GT. NC )  GO TO 152                                          CON02280
      J = JORDER(JI)                                                       CON02290
      IF( JTEST(J) .EQ. 0)  GO TO 150                                      CON02300
      IF ( IC .GT. J )  GO TO 1501                                         CON02310
      NTH = ( (J-1) * (J-2) )/2 + IC                                       CON02320
      GO TO 1502                                                           CON02330
1501  NTH = ( (IC-1) * (IC-2) )/2 + J                                      CON02340
1502  IF( TFMAX .LT. TH(NTH) )  GO TO 151                                  CON02350
      JTEST(J) = 0                                                         CON02360
      GO TO 150                                                            CON02370
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: CONTEX FORTRAN A

```
  151  JJ = J                                                                    CON02380
       GO TO 20                                                                  CON02390
  152  JJ = IC                                                                   CON02400
C                                                                               CON02410
CCCC   STORE THE CLASS NO. FOR THE SAMPLE, IN IR , AND THE VALUE OF THE         CON02420
       PDF FOR THE SAMPLE, IN VR .                                              CON02430
C                                                                               CON02440
       IR(II) = IC                                                              CON02450
       VR(II) = TFMAX                                                           CON02460
C                                                                               CON02470
CC                                                                              CON02480
       NCNT(IC) = NCNT(IC) + 1                                                  CON02490
C                                                                               CON02500
  200  CONTINUE                                                                 CON02510
       IF (LL .EQ. 1) IPT1 = IR(IR)                                             CON02520
  250  CONTINUE                                                                 CON02530
C                                                                               CON02540
CCCCC  SET PROBABLE CLASS FOR FIRST SAMPLE,NEXT SCAN LINE                       CON02550
C                                                                               CON02560
C                                                                               CON02570
CCCCCC ORDER THE CLASSES,ACCORDING TO FREQUENCY OF OCCURENCE ON CURRENT         CON02580
       SCAN LINE                                                                CON02590
C                                                                               CON02600
       DO  201  I=1,NC                                                          CON02610
  201  JORDER(I) = I                                                            CON02620
       DO  210  I=1,NM1                                                         CON02630
       IPL1 = I + 1                                                             CON02640
       DO  205  K=IPL1,NC                                                       CON02650
       IF ( NCNT(I) .GT. NCNT(K) )  GO TO 205                                   CON02660
       ITEMP = NCNT(I)                                                          CON02670
       NCNT(I) = NCNT(K)                                                        CON02680
       NCNT(K) = ITEMP                                                          CON02690
       ITEMP = JORDER(I)                                                        CON02700
       JORDER(I) = JORDER(K)                                                    CON02710
       JORDER(K) = ITEMP                                                        CON02720
  205  CONTINUE                                                                 CON02730
  210  CONTINUE                                                                 CON02740
C                                                                               CON02750
C                                                                               CON02760
       RETURN                                                                   CON02770
        END                                                                     CON02780
```

FILE: FALSY FORTRAN A

```
      SUBROUTINE FALSY(XL,XU,C,FXL,FXU,KC,XN,KT,T,K,KP1,S1,S2,U1,U2,BB)  FAL00010
      DIMENSION S1(K,K),S2(K,K),U1(K),U2(K),BB(K,KP1)                    FAL00020
      RE=.1E-05*C                                                        FAL00030
      IF(RE.LT..000001) RE=.000001                                       FAL00040
      IF(KC.EQ.0) GO TO 7                                                FAL00050
      FXL = G(XL,S1,S2,U1,U2,BB,KT,T,K,KP1)                              FAL00060
      FXU = G(XU,S1,S2,U1,U2,BB,KT,T,K,KP1)                              FAL00070
    7 R=(C-FXL)/(FXU-FXL)                                                FAL00080
      X=XL+R*(XU-XL)                                                     FAL00090
      FX = G(X,S1,S2,U1,U2,BB,KT,T,K,KP1)                                FAL00100
      FM=ABS(FX-C)                                                       FAL00110
      I = 0                                                              FAL00120
C                                                                        FAL00130
      GO TO 8                                                            FAL00140
    9 FXN = G(XN,S1,S2,U1,U2,BB,KT,T,K,KP1)                              FAL00150
      I=I+1                                                              FAL00160
      F=ABS(FXN-C)                                                       FAL00170
C                                                                        FAL00180
      IF(F.GT.FM) GO TO 25                                               FAL00190
      IF(F.LT.RE) RETURN                                                 FAL00200
      FM=F                                                               FAL00210
      IF(XN.LT.X) GO TO 12                                               FAL00220
      XH=.5*(XU+XN)                                                      FAL00230
      FXH = G(XH,S1,S2,U1,U2,BB,KT,T,K,KP1)                              FAL00240
      IF(ABS(FXH-C).LT.RE) GO TO 30                                      FAL00250
      IF(FXH.GT.C) GO TO 14                                              FAL00260
      XL=XN                                                              FAL00270
      FXL=FXN                                                            FAL00280
      X=XH                                                               FAL00290
      FX=FXH                                                             FAL00300
      GO TO 15                                                           FAL00310
   14 XL=X                                                               FAL00320
      FXL=FX                                                             FAL00330
      X=XN                                                               FAL00340
      FX=FXN                                                             FAL00350
      XU=XH                                                              FAL00360
      FXU=FXH                                                            FAL00370
      GO TO 15                                                           FAL00380
   12 XH=.5*(XL+XN)                                                      FAL00390
      FXH = G(XH,S1,S2,U1,U2,BB,KT,T,K,KP1)                              FAL00400
      IF(ABS(FXH-C).LT.RE) GO TO 30                                      FAL00410
      IF(FXH.GT.C) GO TO 13                                              FAL00420
      XL=XH                                                              FAL00430
      FXL=FXH                                                            FAL00440
      XU=X                                                               FAL00450
      FXU=FX                                                             FAL00460
      X=XN                                                               FAL00470
      FX=FXN                                                             FAL00480
      GO TO 15                                                           FAL00490
   13 XU=XH                                                              FAL00500
      FXU=FXN                                                            FAL00510
      X=XH                                                               FAL00520
      FX=FXH                                                             FAL00530
   15 IF( I .EQ. 25)  RETURN                                             FAL00540
      GO TO 8                                                            FAL00550
   25 XN=X                                                               FAL00560
      RETURN                                                             FAL00570
   30 XN=XH                                                              FAL00580
      RETURN                                                             FAL00590
C     FOLLOWING CODE FITS A QUADRATIC TO THE THREE POINTS                FAL00600
C     XL , X , XU , WHICH IS AN APPROXIMATION OF THE FUNCTION  G( H(X) )  FAL00610
C     WITHIN THE DEFINED INTERVAL. A ROOT , XN , OF THE APPROXIMATING    FAL00620
C     QUADRATIC IS RETURNED TO BE USED AS A TRIAL SOLUTION OF            FAL00630
C     G( H(X) ) = C2 - C1                                                FAL00640
C                                                                        FAL00650
    8 W1 = X - XL                                                        FAL00660
      W2 = X*X - XL*XL                                                   FAL00670
      W3=FX-FXL                                                          FAL00680
      W4=XU-XL                                                           FAL00690
      W5=XU*XU-XL*XL                                                     FAL00700
      W6=FXU-FXL                                                         FAL00710
      W7=C-FXL                                                           FAL00720
      A=W1*W6-.5*W4                                                      FAL00730
      IF(ABS(A).LT.1.E-7)GO TO 44                                        FAL00740
      B=W3*W5-W2*W6                                                      FAL00750
      E=-XL*XL*W3-XL*W4*W7*(W2*W4-W1*W5)                                 FAL00760
      D=B*B-4.0*A*E                                                      FAL00770
      IF(D.LT.0.)GO TO 44                                                FAL00780
      D=SQRT(D)                                                          FAL00790
```

FILE: FALSY FORTRAN A

```
      XN=(-B+D)/(2.0*A)                                          FAL00800
      IF(XN.GT.XU) XN=(-B-D)/(2.0*A)                             FAL00810
      GO TO 9                                                    FAL00820
   44 XN=X                                                       FAL00830
      GO TO 9                                                    FAL00840
      END                                                        FAL00850
```

FILE: G FORTRAN A

```
      FUNCTION G(A,S1,S2,U1,U2,BB,KT,T,K,KP1)                          G  00010
C                                                                      G  00020
CCCCC ---------------------------------------------------------------- G  00030
CCCCC                                                                  G  00040
CCCCC                                                                  G  00050
C     ---------------------------------------------------------------- G  00060
C                                                                      G  00070
      DIMENSION  U1(K), U2(K), S1(K,K), S2(K,K), BB(K,KP1)             G  00080
C                                                                      G  00090
      DIMENSION WJ(2),JC(60)                                           G  00100
C                                                                      G  00110
C                                                                      G  00120
      WJ(1) = 4.0                                                      G  00130
C                                                                      G  00140
C                                                                      G  00150
      KZ=0                                                             G  00160
      IF( A .LE. 1.0E-8)  GO TO 25                                     G  00170
      P=(1.0-A)/A                                                      G  00180
      DO 5 I=1,K                                                       G  00190
      BB(I,KP1) = U2(I) - U1(I)                                        G  00200
      DO 5 J=1,K                                                       G  00210
    5 BB(I,J) = S1(I,J) + P*S2(I,J)                                    G  00220
    6 CALL GJR( BB, KP1, K, K, KP1, &8, JC, WJ )                       G  00230
    8 WS2W=0.                                                          G  00240
      WU=0.                                                            G  00250
      DO 17 I=1,K                                                      G  00260
      WU = WU + BB(I,KP1) * ( U2(I) - U1(I) )                          G  00270
      DO 14 J=1,K                                                      G  00280
   14 WS2W = WS2W + BB(I,KP1) * BB(J,KP1) * S2(I,J)                    G  00290
   17 CONTINUE                                                         G  00300
      IF(KZ.EQ.1) GO TO 32                                             G  00310
      G=WU-P*WS2W/A                                                    G  00320
      IF(KT.EQ.1) GO TO 35                                             G  00330
      RETURN                                                           G  00340
   25 DO 27 I=1,K                                                      G  00350
      BB(I,KP1) = U2(I) - U1(I)                                        G  00360
      DO 27 J=1,K                                                      G  00370
   27 BB(I,J) = S2(I,J)                                                G  00380
      KZ=1                                                             G  00390
      GO TO 6                                                          G  00400
   32 G=-WU                                                            G  00410
      RETURN                                                           G  00420
   35 T=P*P*WS2W                                                       G  00430
      RETURN                                                           G  00440
      END                                                              G  00450
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: GJR FORTRAN A

```
      SUBROUTINE GJR(A,NC,NR,N,MC,*,JC,V)                              GJR00010
      DIMENSION A(NR,NC),JC(1),V(2)                                    GJR00020
      CALL OVERFL(IJK)                                                 GJR00030
      IW=V(1)                                                          GJR00040
      M=1                                                              GJR00050
      S=1.                                                             GJR00060
      L=N+(MC-N)*(IW/4)                                                GJR00070
      KD=2-MOD(IW/2,2)                                                 GJR00080
      IF(KD.EQ. 1)V(1) = 0.                                            GJR00090
C     JLS111069                                                        GJR00100
      KI=2-MOD(IW,2)                                                   GJR00110
      GO TO (5,20),KI                                                  GJR00120
5     DO 10  I=1,N                                                     GJR00130
10    JC(I)=I                                                          GJR00140
20    DO 91  I=1,N                                                     GJR00150
      GO TO (22,21),KI                                                 GJR00160
21    M=I                                                              GJR00170
22    IF (I.EQ.N)  GO TO 60                                            GJR00180
      X=-1.                                                            GJR00190
      DO 30  J=I,N                                                     GJR00200
      IF (X.GT.ABS(A(J,I)))  GO TO 30                                  GJR00210
      X=ABS(A(J,I))                                                    GJR00220
      K=J                                                              GJR00230
30    CONTINUE                                                         GJR00240
      IF(K.EQ.I) GO TO 60                                              GJR00250
      S=-S                                                             GJR00260
      V(2) = -V(2)                                                     GJR00270
C     JLS111069                                                        GJR00280
      GO TO (35,40),KI                                                 GJR00290
 35   MU=JC(I)                                                         GJR00300
      JC(I)=JC(K)                                                      GJR00310
      JC(K)=MU                                                         GJR00320
40    DO 50  J=M,L                                                     GJR00330
      X=A(I,J)                                                         GJR00340
      A(I,J)=A(K,J)                                                    GJR00350
50    A(K,J)=X                                                         GJR00360
60    IF (ABS(A(I,I)).GT.0.)  GO TO 70                                 GJR00370
      IF(KD.EQ.1) V(1)=0.                                              GJR00380
      JC(1)=I-1                                                        GJR00390
      RETURN 1                                                         GJR00400
70    GO TO (71,72),KD                                                 GJR00410
71    IF(A(I,I).LT.0.) S=-S                                            GJR00420
      V(1) = V(1) + ALOG(ABS(A(I,I)))                                  GJR00430
C     JLS111069                                                        GJR00440
72    X=A(I,I)                                                         GJR00450
      A(I,I)=1.                                                        GJR00460
      DO 80  J=M,L                                                     GJR00470
      A(I,J)=A(I,J)/X                                                  GJR00480
      CALL OVERFL (IFL)                                                GJR00490
      IF(IFL.EQ.1) GO TO 150                                           GJR00500
80    CONTINUE                                                         GJR00510
      DO 91  K=1,N                                                     GJR00520
      IF (K.EQ.I)  GO TO 91                                            GJR00530
      X=A(K,I)                                                         GJR00540
      A(K,I)=0.                                                        GJR00550
      DO 90  J =M,L                                                    GJR00560
      A(K,J)=A(K,J)-X*A(I,J)                                           GJR00570
      CALL OVERFL (IFL)                                                GJR00580
      IF(IFL.EQ.1) GO TO 150                                           GJR00590
90    CONTINUE                                                         GJR00600
91    CONTINUE                                                         GJR00610
      GO TO (95,140),KI                                                GJR00620
95    DO 130  J=1,N                                                    GJR00630
      IF (JC(J).EQ.J)  GO TO 130                                       GJR00640
      JJ=J+1                                                           GJR00650
      DO 100  I=JJ,N                                                   GJR00660
      IF (JC(I).EQ.J)  GO TO 110                                       GJR00670
100   CONTINUE                                                         GJR00680
110   JC(I)=JC(J)                                                      GJR00690
      DO 120  K=1,N                                                    GJR00700
      X=A(K,I)                                                         GJR00710
      A(K,I)=A(K,J)                                                    GJR00720
120   A(K,J)=X                                                         GJR00730
130   CONTINUE                                                         GJR00740
140    JC(I)=N                                                         GJR00750
      IF(KD.EQ.1) V(2)=S                                               GJR00760
      RETURN                                                           GJR00770
150   JC(I)=1-I                                                        GJR00780
      IF(KD.EQ.1) V(2)=S                                               GJR00790
```

```
FILE: GJR       FORTRAN  A                                          GJR00800
                                                                    GJR00810
      RETURN 1
      END
```

FILE: MAPHDG FORTRAN A

```
      SUBROUTINE MAPHDG(NOCAT,CLSSYM,CATNAM,KATNO,CLSMTX,SUBNO,SUBDES)  MAP00010
C                                                                       MAP00020
CCC                                                                     MAP00030
CCC   THIS ROUTINE PRINTS THE HEADER INFORMATION FOR THE CLASSIFICATION MAP00040
CCC   MAP IN CLASSIFY AND DISPLAY                                       MAP00050
CCC                                                                     MAP00060
CCC                                                                     MAP00070
CCC                                                                     MAP00080
CCC            NOCAT  -- NO. OF CATEGORIES                              MAP00090
CCC            CLSSYM -- SYMBOLS FOR CATEGORIES OR SUBCLASSES           MAP00100
CCC            CATNAM -- CATEGORY NAMES                                 MAP00110
CCC            KATNO  -- CATEGORY EACH CLASS WAS ASSIGNED TO            MAP00120
CCC            CLSMTX -- CLASS NAMES                                    MAP00130
CCC            SUBNO  -- NO. OF SUBCLASSES IN EACH CLASS                MAP00140
CCC            SUBDES -- SUBCLASS NAMES                                 MAP00150
CCC            CLSVC2 -- CLASS EACH SUBCLASS WAS ASSIGNED TO (IN COMMON MAP00160
CCC                      BLOCK INFORM)                                  MAP00170
C                                                                       MAP00180
      IMPLICIT INTEGER (A-Z)                                            MAP00190
C                                                                       MAP00200
C     INCLUDE COMBK1.LIST                                               MAP00210
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    COM00020
     *              FEIVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        COM00030
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             COM00040
     *              GRPCHK(61),GROUPS(124)                              COM00050
CSEND                                                                   MAP00230
C                                                                       MAP00240
      LOGICAL ISWTH                                                     MAP00250
      DIMENSION CLSSYM(1),CATNAM(1),KATNO(1),CLSMTX(1),SUBNO(1),        MAP00260
     *          SUBDES(1)                                               MAP00270
C                                                                       MAP00280
CC    PRINTS CATEGORY CLASSIFIER INFORMATION                            MAP00290
C                                                                       MAP00300
      IF (NOCAT .LE. 0) GO TO 82                                        MAP00310
      WRITE(6,200)                                                      MAP00320
  200 FORMAT(//  T42,'MAP OF CATEGORY CLASSIFIER CLASSIFICATION RESULTS'MAP00330
     *   /// T32,'CATEGORY',T62,'CLASS',T93,'SUBCLASS'/ T31,           MAP00340
     *    'NO.',T37,'NAME',              T60,'NO.',T66,'NAME',         MAP00350
     *     T88,'NO.',T94, 'NAME',T101,'SYMBOL')                        MAP00360
C                                                                       MAP00370
      DO 68   I=1,NOCAT                                                 MAP00380
C                                                                       MAP00390
      WRITE(6,210)I,CATNAM(I)                                           MAP00400
  210 FORMAT(/T31,I2,T37,A4)                                            MAP00410
      ISWTH = .TRUE.                                                    MAP00420
      DO 63   J=1,NOCLS2                                                MAP00430
      IF (KATNO(J) .EQ. I) GO TO 64                                     MAP00440
      GO TO 63                                                          MAP00450
   64 IF (ISWTH) GO TO 65                                               MAP00460
      WRITE(6,220)J,CLSMTX(J)                                           MAP00470
  220 FORMAT(/T60,I2,T66,A4)                                            MAP00480
      ISWTH = .TRUE.                                                    MAP00490
      GO TO 66                                                          MAP00500
   65 WRITE(6,230)J,CLSMTX(J)                                           MAP00510
  230 FORMAT(1H+,T60,I2,T66,A4)                                         MAP00520
   66 DO 67   K=J,NOSUB2                                                MAP00530
      IF ( CLSVC2(K) .EQ. J ) GO TO 70                                  MAP00540
      GO TO 67                                                          MAP00550
   70 NSUBCL = SUBNO(J)                                                 MAP00560
      KK = 0                                                            MAP00570
      DO 75   L=1,NSUBCL                                                MAP00580
      KK = K + L - 1                                                    MAP00590
      IF (ISWTH) GO TO 72                                               MAP00600
      WRITE(6,250)KK,SUBDES(KK),CLSSYM(KK)                              MAP00610
  250 FORMAT(T85,I2,T94,A4,T103,A1)                                     MAP00620
      GO TO 75                                                          MAP00630
   72 WRITE(6,240)KK,SUBDES(KK),CLSSYM(KK)                              MAP00640
  240 FORMAT(1H+,T88,I2,T94,A4,T103,A1)                                 MAP00650
      ISWTH = .FALSE.                                                   MAP00660
   75 CONTINUE                                                          MAP00670
      GO TO 63                                                          MAP00680
   67 CONTINUE                                                          MAP00690
   63 CONTINUE                                                          MAP00700
   68 CONTINUE                                                          MAP00710
      RETURN                                                            MAP00720
C                                                                       MAP00730
CC    PRINTS STANDARD CLASSIFIER INFORMATION                            MAP00740
C                                                                       MAP00750
```

FILE: MAPHDG FORTRAN A

```
 82   CONTINUE                                                                  MAP00760
                                                                                MAP00770
C                                                                               MAP00780
      WRITE(6,260)                                                              MAP00790
 260  FORMAT( //  T42, 'MAP OF STANDARD CLASSIFIER CLASSIFICATION RESULT       MAP00800
     *S' ///T45,'CLASS',T77,'SURCLASS'/ T42,'NO.',T50,'NAME',T72,'NO.',        MAP00810
     *T78, 'NAME',T85,'SYMBOL')                                                 MAP00820
C                                                                               MAP00830
      CLSNUM = 1                                                                MAP00840
      ISWTH = .TRUE.                                                            MAP00850
      DO 89   I=1,NOSUB2                                                        MAP00860
      IF (CLSNUM.EQ. CLSVC2(I))  GO TO 85                                       MAP00870
      CLSNUM = CLSNUM + 1                                                       MAP00880
      GO TO 87                                                                  MAP00890
 85   IF (ISWTH)  GO TO 87                                                      MAP00900
      WRITE(6,270)I,SUBDES(I),CLSSYM(I)                                         MAP00910
 270  FORMAT(T72,I2,T78,A4,T87,A1)                                              MAP00920
      GO TO 89                                                                  MAP00930
 87   WRITE(6,280)CLSNUM,CLSMTX(CLSNUM),I,SUBDES(I),CLSSYM(I)                   MAP00940
 280  FORMAT(/T42,I2,T50,A4,T72,I2,T78,A4,T87,A1)                               MAP00950
      ISWTH = .FALSE.                                                           MAP00960
 89   CONTINUE                                                                  MAP00970
      RETURN                                                                    MAP00980
      END
```

FILE: MCHLSK FORTRAN A

```
      SUBROUTINE MCHLSK( KK,NV,DUM,DET)
C
CCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCC
CCCCCCCCCCCCCCCCCCCCC
CCCCC
      THIS ROUTINE COMPUTES THE MODIFIED CHOLESKY DECOMPOSITION OF
      THE COVARIANCE MATRIX. THE DECOMPOSITIONS OVERLAY THE ELEMENTS
      OF THE COVARIANCE MATRIX.

      KK = L  D L*

      KK =  COVARIANCE MATRIX STORED IN SYMMETRIC STORAGE

      NV = NO. OF  CHANNELS

      DUM =  A WORK AREA OF SISE  NV-1

      DET  = THE DETERMINANT OF THE COVARIANCE MATRIX

C
      REAL  KK
      LOGICAL JE1
      DIMENSION  KK(1), DUM(1)
C
      DOUBLE PRECISION  TF, R, R1, DUM, T1
C
      JE1 = .TRUE.
      J1 = 0
      JO = 0
      DET = 1.0
C
CCC
      LOOP OVER ALL CHANNELS
C
      DO  10  J=1,NV
      KL = J-1
      L = J+1
      JO = J1
      J1 = J1 + J
      TF = KK(J1)
      IF(JE1) GO TO 12
      K1 = 0
C
CCCCC
      COMPUTE  THE DIAGONAL ELEMENTS OF D AND STORE IN KK
CCCCC
      TEMPORARILY STORE THE PRODUCT  KK(I,I)*KK(J,I) IN DUM(I)
C
      DO  15  I=1,KL
      R = KK(JO + I)
      K1 = K1 + I
      R1 = KK(K1) * R
      TF = TF - R1 * R
      DUM(I) = R1
 15   CONTINUE
      KK(J1) = TF
 12   CONTINUE
      DET = DET * TF
      IF (L .GT. NV) GO TO 10
      IRO = J1 - L + 1
C
CCC
      COMPUTE THE  R, J-TH ELEMENT OF L , USING T1
C
      DO  20  IR= L,NV
      IRO = IRO + IR - 1
      T1 = KK(IRO + J)
      IF(JE1) GO TO 16
      DO 25  I= 1,KL
      T1 = T1 - DUM(I) * KK(IRO + I)
 25   CONTINUE
 14   KK(IRO + J) = T1/TF
 20   CONTINUE
      JE1 = .FALSE.
 10   CONTINUE
C
CCCCC
CCCCC
CCCCC
CCCCC
      KK CONTAINS , IN "SYMETRIC" STORAGE, THE MODIFIED CHOLESKY
      FACTORIZATION OF THE INPUT MATRIX. THE LOWER TRIANGULAR MATRIX, L,
      OCCUPIES THE OFF-DIAGONAL ELEMENTS OF KK , AND THE DIAGONAL
      MATRIX, D , IS STORED IN THE DIAGONAL ELEMENTS IN KK.
C
      RETURN
      END
```

FILE: REDIF2 FORTRAN A

```
      SUBROUTINE REDIF2(ARRAY,TOP,APRIOR,KATNO,BMATRX,PRIORI)                  REDO0010
      IMPLICIT INTEGER (A-H,O-Z)                                               REDO0020
      DIMENSION ARRAY(1)                                                       REDO0030
C                                                                              REDO0040
CCI---------------------------------------------------------------------------IREDO0050
CCI---------------------------------------------------------------------------IREDO0060
CCI                                                                           IREDO0070
CCI     CALL...  CALL REDIF2(ARRAY,TOP,APRIOR,KATNO,BMATRX,PRIORI)            REDO0080
CCI                                                                           IREDO0090
CCI     ARGS...  ARRAY  - SEE MONTOR                                          REDO0100
CCI              TOP    - SEE MONTOR                                          REDO0110
CCI     APRIOR - APRIORI VALUES FOR EACH SUBCLASS                             REDO0120
CCI     KATNO  - CATEGORY - CLASS CORRESPONDENCE                              REDO0130
CCI     BMATRX - B-TRANSFORMATION, IF AVAILABLE                               REDO0140
CCI     PRIORI - TEMPORARY STORAGE FOR A PRIORI VALUES                        REDO0150
CCI                                                                           IREDO0160
CCI     REQUIRES. COMMONS /INFORM/GLOBAL/CLASS/                               REDO0170
CCI                                                                           REDO0180
CCI               ROUTINES FIND12,CROSTA,GRPSCN,FLTNUM,CATSCN                 REDO0190
CCI                                                                           IREDO0200
CCI     PURPOSE.. READS AND ANALYZES SUPER CONTROL CARDS                      IREDO0210
CCI               FOR 'CLASSIFY'                                              IREDO0220
CCI                                                                           IREDO0230
CCI     RETURNS.. SUPERVISOR INFORMATION AND STATISTICS                       IREDO0240
CCI                                                                           IREDO0250
CCI---------------------------------------------------------------------------IREDO0260
CCI---------------------------------------------------------------------------IREDO0270
CCCCCC                                                                        REDO0280
CCCCCC                                                                        REDO0290
CCCCCC                                                                        REDO0300
CCCCCC                                                                        REDO0310
C     ------------------------------------------------------------------------REDO0320
       CONTINUE                                                               REDO0330
C                                                                             REDO0340
CC    INCLUDE COMBK1,LIST                                                     REDO0350
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,                REDO0360
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,          REDO0370
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),              REDO0380
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                   REDO0390
     *              GRPCHK(61),GROUPS(124)                                    REDO0400
      COMMON /CLASS/ APRFLG,BMCOMB,BMFEAT,BMFLG,NOCAT,THIJ1,IDATA1,           REDO0410
     *               NFILE,STATKY,CATNAM(60),                                 REDO0420
     3               CLSSYM(60),CON(60),DET(60),FLDESC,FLDINF(6),             REDO0430
     4               KCLSNA(60),NOCTCL(60),SUBCAT(60)                         REDO0440
     *    ,NOCHAN,CHNVEC(30)                                                  REDO0450
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,               REDO0460
     *              HISFIL,HISKEY,TRFORM,ERPTP,ERPKEY,MAPUNT,NOFILE,          REDO0470
     *    DRUMAD,DRM*OS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                      REDO0480
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                          REDO0490
     *  ,DOTUNT,DOTFIL,NCHPAS,TPNSFL,BMTRFL,HISTFL,PCHUNT,                    REDO0500
     *  CROUNT,PRTUNT,RANDIO                                                  REDO0510
C$END                                                                         REDO0520
CC                                                                            REDO0530
                                                                              REDO0540
      EQUIVALENCE      (HED1(1),HEAD(4)), (DATE(1),HEAD(22)),                 REDO0550
     1                 (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                REDO0560
                                                                              REDO0570
C     ------------------------------------------------------------------------REDO0580
CCCCCC                                                                        REDO0590
CCCCCC                                                                        REDO0600
C     ------------------------------------------------------------------------REDO0610
CCCCCC                                                                        REDO0620
      INCLUDE  COMBK6,LIST                                                    REDO0630
C     ------------------------------------------------------------------------REDO0640
CCCCCCCCC                                                                     REDO0650
CCCCCCCCC                                                                     REDO0660
CCCCCCCCC                                                                     REDO0670
C     ------------------------------------------------------------------------REDO0680
CCCCCCCCC                                                                     REDO0690
CCCCCCCCC                                                                     REDO0700
C     ------------------------------------------------------------------------REDO0710
CCCCCCCCC                                                                     REDO0720
CCCCCCCCC                                                                     REDO0730
C     ------------------------------------------------------------------------REDO0740
C                                                                             REDO0750
      DIMENSION CODTAB(16),KATNO(60)                                          REDO0760
     1 ,  HED1(10), HED2(10), DATE(2), COMENT(10), BMCOF(3)                   REDO0770
     2          , APRIOR(1), CARD2(62), COMVEC(2)                            REDO0780
      DIMENSION EQUVEC(2),ACARD(20)                                           REDO0790
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: REDIF2 FORTRAN A

```
C                                                                       RED00800
C                                                                       RED00810
C     INCLUDE COMBK2,LIST                                               RED00820
      DATA CODTAB/'SUBC','CHAN','DATE','*END','COMM',                   RED00830
     1'HED1','HED2','OPTI','MODU','GROU','B-MA',                        RED00840
     2'APRI','CATE','DATA','STAT','MAPT'/                               RED00850
C                                                                       RED00860
C                                                                       RED00870
      DATA  BLANK/' '/, NBCD/'N'/, OBCD/'O'/, SBCD/'S'/, CBCD/'C'/      RED00880
     1  , COMVEC/1, ','  /  ,  BMCOF/2, 'C' , 'F' /                     RED00690
      DATA IBCD/'I'/                                                    RED00900
      DIMENSION SLASH(2)                                                RED00910
      DATA SLASH/1,'/'/                                                 RED00920
C                                                                       RED00930
C                                                                       RED00940
      DATA ENDCRD/'*END'/,EQUVEC/1,'='/                                 RED00950
C                                                                       RED00960
      DATA ORCD/'O'/,FBCD/'F'/,TESTB/'ILE '/,UBCD/'U'/                  RED00970
      REAL BMATRX(1),APRIOR,PRIORI(60)                                  RED00980
      LOGICAL*1 LCHAR(4),LLCHAR(4)                                      RED00990
      DIMENSION ICHAR(1),IICHAR(1)                                      RED01000
      DATA BLANKS/'    '/                                               RED01010
      EQUIVALENCE (LCHAR(1),ICHAR(1)),(LLCHAR(1),IICHAR(1))             RED01020
C                                                                       RED01030
CCCCCCCCC                                                               RED01040
CCCCCCCCC                                                               RED01050
CCCCCCCCC  ------------------------------------------------------------ RED01060
CCCCCCCCC  ------------------------------------------------------------ RED01070
CCCCCCCCC                                                               RED01080
CCCCCCCCC                                                               RED01090
CCCCCCCCC  INITZ                                                        RED01100
CCCCCCCCC  ------------------------------------------------------------ RED01110
C                                                                       RED01120
      SYMMAX = 60                                                       RED01130
      NF=0                                                              RED01140
      NOSUR2=0                                                          RED01150
      I = 0                                                             RED01160
      NOCAT = 0                                                         RED01170
      NOCLS2 = 0                                                        RED01180
      DATSWT = 0                                                        RED01190
      NOGRP = 0                                                         RED01200
      GRPTR = 0                                                         RED01210
      BMFLG = 0                                                         RED01220
      APRFLG = 0                                                        RED01230
      APRKEY = 0                                                        RED01240
      NOCHAN = 0                                                        RED01250
      DO 10   I=1,60                                                    RED01260
      CATNAM(I) = BLANK                                                 RED01270
   10 NOCTCL(I) = 0                                                     RED01280
      DO 1 I=1,SYMMAX                                                   RED01290
      APRIOR(I) = 0.0                                                   RED01300
    1 GRPCHK(I) = 0                                                     RED01310
C                                                                       RED01320
CCCC  READ SUPERVISOR CONTROL CARDS                                     RED01330
CCCC  ------------------------------------------------------------     RED01340
C                                                                       RED01350
  225 CONTINUE                                                          RED01360
C                                                                       RED01370
CCC     PUT THE NEXT CARD IN THE REREAD BUFF-ER>-RQ+X)                  RED01380
C                                                                       RED01390
      READ(21,230)(ACARD(I),I=1,20)                                     RED01400
  230 FORMAT(20A4)                                                      RED01410
      WRITE(30,230)(ACARD(I),I=1,20)                                    RED01420
      REWIND 30                                                         RED01430
      READ(30,2204) CODE,CARD2                                          RED01440
      REWIND 30                                                         RED01450
 2204 FORMAT(A4,6X,62A1)                                                RED01460
      COL = 0                                                           RED01470
      WRITE(6,2206)   CODE,CARD2                                        RED01480
 2206 FORMAT(15,A4,6X,62A1)                                             RED01490
      DO 237 I = 1,14                                                   RED01500
  235 IF ( CODE .EQ. CODTAB(I) )                                        RED01510
     1  GO TO (240,250,260,270,500,510,520,530,540,560,580,590,600,610, RED01520
     * 620,630),I                                                      RED01530
  237 CONTINUE                                                          RED01540
      GO TO 1000                                                        RED01550
C                                                                       RED01560
CC    SUBCLASS                                                          RED01570
C     ------------------------                                          RED01580
```

FILE: REDIF2 FORTRAN A

```
C                                                                       RED01590
  240 M = NXTCHR(CARD2,COL)                                             RED01600
      IF ( M .EQ. BLANK ) GO TO 225                                     RED01610
      COL = COL-1                                                       RED01620
      NOSUB2 = NUMBER(CARD2,COL,SUBVC2,NOSUB2)                          RED01630
      CALL ORDER(SUBVC2,NOSUB2)                                         RED01640
C                                                                       RED01650
      GO TO 225                                                         RED01660
C                                                                       RED01670
C      CHANNELS                                                         RED01680
C      --------                                                         RED01690
C     ------------------------                                          RED01700
C                                                                       RED01710
  250 IF( RMFLG .GT. 0)  GO TO 225                                      RED01720
      M = NXTCHR( CARD2, COL )                                          RED01730
      IF ( M .EQ. BLANK ) GO TO 225                                     RED01740
      J = FIND12(CARD2,COL,EQUVEC)                                      RED01750
      IF (J .EQ. -1)  GO TO 251                                         RED01760
      IF(M .EQ. SPCD)GO TO 253                                          RED01770
      IF(M .EQ. DPCD)GO TO 255                                          RED01780
  251 WRITE(4,252)                                                      RED01790
  252 FORMAT(' ERROR ON CHANNELS CARD')                                 RED01800
      GO TO 225                                                         RED01810
  253 NF = NUMBER(CARD2,COL,FETVC2,NF)                                  RED01820
      NOFET2 = NF                                                       RED01830
      CALL ORDER(FETVC2,NOFET2)                                         RED01840
      COL = COL - 1                                                     RED01850
      GO TO 250                                                         RED01860
  255 NOCHAN = NUMBER(CARD2,COL,CHNVEC,NOCHAN)                          RED01870
      CALL ORDER(CHNVEC,NOCHAN)                                         RED01880
      COL = COL - 1                                                     RED01890
      GO TO 250                                                         RED01900
C                                                                       RED01910
C      DATE                                                             RED01920
C     ------------------------                                          RED01930
C                                                                       RED01940
  260 M = NXTCHR(CARD2,COL)                                             RED01950
      IF ( M .EQ. BLANK ) GO TO 225                                     RED01960
      READ (30,9999) DATE                                               RED01970
      REWIND 30                                                         RED01980
 9999 FORMAT(10X,11A4)                                                  RED01990
      GO TO 225                                                         RED02000
C                                                                       RED02010
C      COMMENT                                                          RED02020
C     ------------------------                                          RED02030
C                                                                       RED02040
  500 CONTINUE                                                          RED02050
      READ (30,9999) COMENT                                             RED02060
      REWIND 30                                                         RED02070
      GOTO 225                                                          RED02080
C                                                                       RED02090
C      HED1                                                             RED02100
C     ------------------------                                          RED02110
C                                                                       RED02120
  510 CONTINUE                                                          RED02130
      READ (30,9999) HED1                                               RED02140
      REWIND 30                                                         RED02150
      GOTO 225                                                          RED02160
C                                                                       RED02170
C      HED2                                                             RED02180
C     ------------------------                                          RED02190
C                                                                       RED02200
  520 CONTINUE                                                          RED02210
      READ (30,9999) HED2                                               RED02220
      REWIND 30                                                         RED02230
      GOTO 225                                                          RED02240
C                                                                       RED02250
C      OPTION CARD                                                      RED02260
C     ------------------------------------------------------------------RED02270
C                                                                       RED02280
  530 M = NXTCHR(CARD2,COL)                                             RED02290
      IF ( M .EQ. BLANK ) GO TO 225                                     RED02300
      IF ( M .NE. SPCD  ) GO TO 1000                                    RED02310
      STATKY =1                                                         RED02320
      M = FIND12( CARD2,COL,COMVEC)                                     RED02330
      IF ( M .LE. 0) GO TO 225                                          RED02340
      GO TO 530                                                         RED02350
C      MODULE                                                           RED02360
C      ------------------------                                         RED02370
```

```
FILE: REDIF2   FORTRAN  A

C                                                                               RED023A0
  540 CONTINUE                                                                  RED02340
      CALL CROSTA(ARRAY,TOP)                                                    RED02400
      DATSWT = 1                                                                RED02410
      GOTO 225                                                                  RED02420
C                                                                               RED02430
CCC   GROUPS                                                                    RED02440
CCC   ------------------------                                                  RED02450
C                                                                               RED02460
  560 CONTINUE                                                                  RED02470
      I = GRPSCN(CARD2,SYMMAX,GRPTR)                                            RED024A0
      GOTO 225                                                                  RED02490
C                                                                               RED02500
CCC    B-MATRIX INPUT                                                           RED02510
C                                                                               RED02520
  580 M= FIND12(CARD2,COL,BMCOF)                                                RED02530
C                                                                               RED02540
      IF(M .EQ. 2)GO TO 581                                                     RED02550
C     B-MATRIX INPUT FROM CARDS                                                 RED02560
C                                                                               RED02570
      IF(M .EQ. 3)GO TO 582                                                     RED02580
C     B-MATRIX INPUT FROM FILE                                                  RED02590
C                                                                               RED02600
      WRITE(6,5801)  CODE , CARD2                                               RED02610
 5801 FORMAT(//// 5X,'****  CLSFY/REDIF2  ---  B-MATRIX INPUT STIPULATED      RED02620
     1 BY CONTROL CARD ...'//5X,2H''',A4,6X,62A1,2H''' //5X,                    RED02630
     2 '**** UNABLE TO INTERPRET TYPE OF B-MATRIX INPUT --- PROGRAM EXEC       RED02640
     3UTION TERMINATED FROM REDIF2 ****'/1H1)                                   RED02650
C                                                                               RED02660
      CALL EXIT                                                                 RED02670
C                                                                               RED026A0
CCC                                                                             RED02690
CCC   B-MATRIX INPUT ON CARDS                                                   RED02700
C                                                                               RED02710
  581 CALL BMFIL(BMATRX,BMCOMB,BMFEAT,FETVC2,1)                                 RED02720
      NOFET2 = BMFEAT                                                           RED02730
      BMKEY = 1                                                                 RED02740
      BMFLG = (BMCOMB*(BMCOMB+1)) / 2                                           RED02750
      IF( BMFEAT .LE. 0 .OR. BMCOMB .LE. 0)  GO TO 586                          RED027A0
      NBM = BMCOMB * BMFEAT                                                     RED02770
      IF( NBM .GT. 450)  GO TO 586                                              RED027A0
      BMFLG = (BMCOMB * (BMCOMB+1))/2                                           RED02790
C                                                                               RED02800
      END FILE  BMFILE                                                          RED02810
      BMKEY = 1                                                                 RED02820
C                                                                               RED02830
      GO TO 225                                                                 RED02840
C                                                                               RED02850
CCC                                                                             RED02860
CCC    B-MATRIX INPUT FROM B-MATRIX FILE                                        RED02870
C                                                                               RED028A0
  582 CALL BMFIL(BMATRX,BMCOMB,BMFEAT,FETVC2,2)                                 RED02890
      NOFET2 = BMFEAT                                                           RED02900
C                                                                               RED02910
      IF( BMFEAT .LE. 0 .OR. BMCOMB .LE. 0)  GO TO 587                          RED02920
      NBM = BMCOMB * BMFEAT                                                     RED02930
      IF( NBM .GT. 450)  GO TO 587                                              RED02940
C                                                                               RED02950
      BMFLG = ( BMCOMB * (BMCOMB+1))/2                                          RED02960
C                                                                               RED02970
      GO TO 225                                                                 RED02980
C                                                                               RED02990
CCC                                                                             RED03000
CCC   ERROR RETURNS ---  B-MATRIX INPUT                                         RED03010
C                                                                               RED03020
 584  WRITE(6,5841)  CODE, CARD2                                                RED03030
 5841 FORMAT(//// 5X,'**** CLSFY/REDIF2 --- BAD CARD INPUT DETECTED ON AN      RED03040
     1TTEMPT TO READ B-MATRIX INFORMATION AS DIRECTED BY THE CONTROL CAR        RED03050
     2D ...'//5X,2H''',A4,6X,62A1,2H'''//5X,'***  TERMINATING PROGRAM EXEC       RED03060
     3UTION FROM REDIF2 ****' /1H1)                                             RED03070
C                                                                               RED03080
      CALL EXIT                                                                 RED03090
C                                                                               RED03100
  587 WRITE(6,5871)BMCOMB,BMFEAT,(FETVC2(I),I=1,BMFEAT)                         RED03110
 5871 FORMAT(////5X,'***** CLSFY/REDIF2 --- B-MATRIX INPUT FROM BMFILE-'       RED03120
     2 //2X,'BAD INPUT VALUES DETECTED: NO. COMBINATIONS(BMCOMB) =',I5,        RED03130
     31X,' ,NO. CHANNELS (BMFEAT) =',I5,1X,', CHANNEL VECTOR (BMVEC) = ,       RED03140
     4,' // ( 5X,30I4) )                                                        RED03150
      WRITE(6,5872)                                                             RED03160
```

```
FILE: REDIF2   FORTHAN  A

 5872 FORMAT(/// 5X,'***** TERMINATING PROGRAM EXECUTION FROM REDIF2 **'REDO3170
     1***'/1H1)                                                        REDO3180
      CALL EXIT                                                        REDO3190
C                                                                      REDO3200
CCC   APRIORI  CARD  ---  TRAINING CLASS APRIORI PROBABILITY VALUES    REDO3210
C                                                                      REDO3220
      M = NXTCHR(CARD2,COL)                                            REDO3230
  590 IF(M .NE. FRCD)GO TO 592                                         REDO3240
      APRKEY = -777777                                                 REDO3250
C     FLAG FOR COMPUTING  APRIOR VALUES FROM STATFILE                  REDO3260
      GO TO 596                                                        REDO3270
  592 M=SYMMAX - APRKEY                                                REDO3280
      COL = 0                                                          REDO3290
      NAPR = APRKEY + 1                                                REDO3300
      APRKEY = APRKEY + FLTNUM(CARD2,COL,PRIORI(NAPR),M)               REDO3310
      IF (APRKEY.NE.1) GO TO 596                                       REDO3320
      WRITE(6,594)                                                     REDO3330
  594 FORMAT(/T5,'*** CLSFY/REDIF2 - BAD CARD INPUT ON APRIORI CARD - DE'REDO3340
     *FAULT APRIORI PROBABILITY VALUES WILL BE USED.',/)                REDO3350
      APRKEY = 0                                                       REDO3360
C                                                                      REDO3370
C 596 APRFLG = APRKEY                                                  REDO3380
C                                                                      REDO3390
      GO TO 225                                                        REDO3400
CCC                                                                    REDO3410
      CATEGORY CARD                                                    REDO3420
      -------------                                                    REDO3430
CCC                                                                    REDO3440
  600 IF(NOCAT.EQ.-7654321)GO TO 225                                   REDO3450
      LL = NXTCHR(CARD2,COL)                                           REDO3460
       IF (LL.NE.FRCD) GO TO 605                                       REDO3470
       LL = NXTCHR(CARD2,COL)                                          REDO3480
       IF (LL.NE.FRCD) GO TO 605                                       REDO3490
      NOCAT = -7654321                                                 REDO3500
      GO TO 225                                                        REDO3510
  605 NOCAT = NOCAT + 1                                                REDO3520
      L = CATSCN(CARD2,KCLSNA,CATNAM(NOCAT),L,NOCTCL(NOCAT),NOCAT)     REDO3530
      GO TO 225                                                        REDO3540
CCC                                                                    REDO3550
      DATA FILE CARD                                                   REDO3560
CCC                                                                    REDO3570
  610 M = NXTCHR(CARD2,COL)                                            REDO3580
      IF(M .EQ. BLANK)GO TO 225                                        REDO3590
      IF(M .EQ. URCD)GO TO 616                                         REDO3600
      IF(M .EQ. FRCD) GO TO 617                                        REDO3610
  613 WRITE(6,753)                                                     REDO3620
  753 FORMAT(' ERROR ON DATA FILE CARD')                               REDO3630
      GO TO 225                                                        REDO3640
  615 J = FIND12(CARD2,COL,EQUVEC)                                     REDO3650
      IF ( J .EQ. -1)  GO TO 613                                       REDO3660
      M = NUMBER(CARD2,COL,DATAPE,ZERO)                                REDO3670
      COL = COL - 1                                                    REDO3680
      GO TO 610                                                        REDO3690
  617 J = FIND12(CARD2,COL,EQUVEC)                                     REDO3700
      IF (J.EQ. -1) GO TO 613                                          REDO3710
      M = NUMBER(CARD2,COL,DATFIL,ZERO)                                REDO3720
      DATFIL = DATFIL - 1                                              REDO3730
      IF (DATFIL .LT. 0)  DATFIL = 0                                   REDO3740
      COL = COL - 1                                                    REDO3750
      GO TO 610                                                        REDO3760
C                                                                      REDO3770
CCC   STAT FILE CARD                                                   REDO3780
CCC                                                                    REDO3790
  620 M = NXTCHR(CARD2,COL)                                            REDO3800
      IF(M .EQ. BLANK)GO TO 225                                        REDO3810
      IF(M .EQ. URCD)GO TO 625                                         REDO3820
      IF(M .EQ. FRCD)GO TO 627                                         REDO3830
  623 WRITE(6,755)                                                     REDO3840
  755 FORMAT(' ERROR ON STAT FILE CARD')                               REDO3850
      GO TO 225                                                        REDO3860
  625 J = FIND12(CARD2,COL,EQUVEC)                                     REDO3870
      IF ( J .EQ. -1)  GO TO 623                                       REDO3880
      M = NUMBER(CARD2,COL,SAVTAP,ZERO)                                REDO3890
      COL = COL - 1                                                    REDO3900
      GO TO 620                                                        REDO3910
  627 J = FIND12(CARD2,COL,EQUVEC)                                     REDO3920
      IF (J .EQ. -1)  GO TO 623                                        REDO3930
      M = NUMBER(CARD2,COL,STAFIL,ZERO)                                REDO3940
      STAFIL = STAFIL - 1                                              REDO3950
```

```
FILE: REDIF2   FORTRAN  A

      IF (STAFIL .LT. 0) STAFIL = 0                                          RED03960
      COL = COL - 1                                                          RED03970
      GO TO 620                                                              RED03980
C                                                                            RED03990
C                                                                            RED04000
C     -----                                                                  RED04010
C     ------------------------                                               RED04020
C                                                                            RED04030
C                                                                            RED04040
 630    CONTINUE                                                             RED04050
        ERRKEY = 1                                                           RED04060
        J = NXTCHR(CARD2,COL)                                                RED04070
 631      IF (J.EQ.BLANK) GO TO 650                                          RED04080
        IF ( J.NE.UBCD) GO TO 635                                            RED04090
        J = FIND12(CARD2,COL,EQUVEC)                                         RED04100
        IF (J.NE.2) GO TO 650                                                RED04110
        ISTART = 0                                                           RED04120
        J = NUMBER(CARD2,COL,MAPTAP,ISTART)                                  RED04130
        J = FIND12(CARD2,COL,EQUVEC)                                         RED04140
        IF (J.NE.2) GO TO 650                                                RED04150
        ISTART = 0                                                           RED04160
        J = NUMBER(CARD2,COL,ERRKEY,ISTART)                                  RED04170
        GO TO 225                                                            RED04180
 635    IF (J.NE.ONECD) GO TO 640                                            RED04190
        J = FIND12(CARD2,COL,SLASH)                                          RED04200
        IF (J.NE.2) GO TO 650                                                RED04210
        J = NXTCHR(CARD2,COL)                                                RED04220
        IF (J.EQ.FBCD) GO TO 645                                             RED04230
        IF ( J.EQ.UBCD) GO TO 631                                            RED04240
        GO TO 650                                                            RED04250
 640    IF (J.NE.FBCD) GO TO 650                                             RED04260
 645    J = FIND12(CARD2,COL,EQUVEC)                                         RED04270
        IF (J.NE.2) GO TO 650                                                RED04280
        ISTART = 0                                                           RED04290
        J = NUMBER(CARD2,COL,ERRKEY,ISTART)                                  RED04300
        J = FIND12(CARD2,COL,EQUVEC)                                         RED04310
        IF (J.NE.2) GO TO 650                                                RED04320
        ISTART = 0                                                           RED04330
        J = NUMBER(CARD2,COL,MAPTAP,ISTART)                                  RED04340
        GO TO 225                                                            RED04350
 650    WRITE (6,655)                                                        RED04360
 655    FORMAT( '  ERROR ON MAPTAP CONTROL CARD')                            RED04370
        GO TO 225                                                            RED04380
 270    CONTINUE                                                             RED04390
 290      CONTINUE                                                           RED04400
          IF (NOCAT.EQ.1) WRITE(6,615)                                       RED04410
          IF (NOCAT.EQ.1) CALL CMERR                                         RED04420
  615     FORMAT(//// 5X,'MUST HAVE AT LEAST TWO CATEGORIES')               RED04430
      RETURN                                                                 RED04440
C                                                                            RED04450
C     ERROR ROUTINES                                                         RED04460
C     ------------------------------------------------------------------     RED04470
C                                                                            RED04480
 1000 WRITE (6,10002) CODE, CARD2                                            RED04490
10002 FORMAT(//// 5X,'****  CLSFY/REDIF2  ---  BAD PROCESSOR CONTROL CARRED04500
     1D  ',A4,6X,42A1,2H'//5X,'**** TERMINATING PROGRAM  FXRED04510
     2ECUTION FROM REDIF2  ****'/1H1)                                        RED04520
      GO TO 225                                                              RED04530
C                                                                            RED04540
       END                                                                   RED04550
```

FILE: RELERR FORTRAN A

```
      FUNCTION  RELERR(COVMTX, COV, NOFET2, VARSZ2)
      INTEGER  VARSZ2
      COMMON/SCR/SCR1(2000),SCR2(10500)
      DIMENSION  D(30), COV(VARSZ2), COVMTX(VARSZ2)
      REAL   LDL(465)
      EQUIVALENCE  ( ENORMO,  SCR2(1)), ( ENORMD,  SCR2(2)) ,
C
     1  ( D(1),  SCR2(3)), ( LDL(1),  SCR2(33)),
C
     2  ( SUM,  SCR2(963)), ( II,  SCR2(964)), ( I,  SCR2(965)),
C
     3  ( J,  SCR2(966)), ( L,  SCR2(967)), ( JJ,  SCR2(968)),
C
     4  ( KK,  SCR2(969)), ( JK, SCR2(970)), ( KP,  SCR2(971)),
C
     5  (  III,  SCR2(972)), ( IJ,  SCR2(973)), ( JP,  SCR2(974)),
C
     6  ( IP,  SCR2(975))
C
C     COMPUTE THE EUCLIDEAN NORM OF THE COVARIANCE MATRIX, BEFORE
C     CHOLESKY FACTORIZATION
C
      ENORMO = 0.0
      DO  181  I=1,VARSZ2
  181 ENORMO = ENORMO + 2.0 * COV(I) * COV(I)
C
      II = 0
      DO  182  I=1,NOFET2
      II = II + I
      ENORMO = ENORMO - ( COV(II) * COV(II) )
      D(I) = COVMTX(II)
  182 COVMTX(II) = 1.0
C
      ENORMD = SQRT(ENORMO)
C
      IJ = 0
      DO  187  I=1,NOFET2
      IK = I
      II = ( IK * (IK-1) )/2
      DO  186  J=1,IK
      JK = J
      SUM = 0.0
      JJ = ( JK * (JK-1) )/2
      DO  185  KP=1,JK
      KK = KP
      JP = JJ + KP
      IP = II + KP
  185 SUM = SUM + ( COVMTX(JP) * COVMTX(IP) * D(KP) )
      IJ = IJ + 1
  186 LDL(IJ) = SUM
C     =L * D * L*
  187 CONTINUE
C
      II = 0
      DO  188  L=1,NOFET2
      II = II + L
  188 COVMTX(II) = D(L)
      ENORMD = 0.0
      III = 0
      II = 0
      DO  190  I=1,NOFET2
      II = II + I
      DO  189  J=1,I
      III = III + 1
      SUM = COV(III) - LDL(III)
  189 ENORMD = ENORMD + 2.0 * SUM * SUM
      SUM = COV(III) - LDL(III)
  190 ENORMD = ENORMD - ( SUM * SUM )
C
      IF ( ENORMD .LE. 1.0E-8)  GO TO 191
      ENORMD = SQRT(ENORMD)
C
  191 RELERR = ENORMD/ENORMO
      RETURN
      END
```

FILE: SETUP2 FORTRAN A

```
      SUBROUTINE SETUP2(ARRAY,TOP,FLDFLG,APRIOR,BMATRX,KATNO)                SET00010
      IMPLICIT INTEGER (A-H,O-Z)                                             SET00020
      REAL CON,DET,BMATRX,APRIOR,NORM                                        SET00030
     *  ,APRI,APRIO,PRIORI                                                  SET00040
C                                                                            SET00050
      DIMENSION ARRAY(1),PRIORI(60)                                          SET00060
C                                                                            SET00070
CCI|-------------------------------------------------------------------------ISET00080
CCI|-------------------------------------------------------------------------ISET00090
CCI|                                                                        ISET00100
CCI     CALL..     CALL SETUP2(ARRAY,TOP,FLDFLG,APRIOR,BMATRX,KATNO)       SET00110
CCI                                                                         SET00120
CCI     ARGS..     ARRAY  - SEE MONTOR                                      SET00130
CCI                TOP    - SEE MONTOR                                     SET00140
CCI                FLDFLG -                                                SET00150
CCI                APRIOR - APRIORI VALUES FOR EACH SUBCLASS               SET00160
CCI                BMATRX - B-TRANSFORMATION MATRIX, IF AVAILABLE          SET00170
CCI             KATNO  - CATEGORY - CLASS CORPESPONDENCE                    SET00180
CCI                                                                        ISET00190
CCI                                                                        ISET00200
CCI     REQUIRES. COMMON /INFORM/CLASS/GLOBAL/                              SET00210
CCI                                                                         SET00220
CCI                ROUTINES FIND12,RED1F2,REDSAV                            SET00230
CCI                                                                        ISET00240
CCI     PURPOSE    ANALYSIZE SUPERVISOR INFORMATION                        ISET00250
CCI                                                                        ISET00260
CCI     RETURNS.. SUPERVISOR INFORMATION AND REDUCED STATISTICS            ISET00270
CCI                FOR PROCESSING                                          ISET00280
CCI                                                                        ISET00290
CCI|-------------------------------------------------------------------------ISET00300
CCI|-------------------------------------------------------------------------ISET00310
CCCC                                                                        SET00320
C                                                                           SET00330
C                                                                           SET00340
       CONTINUE                                                             SET00350
C     INCLUDE COMBK1.LIST                                                   SET00360
CCC                                                                         SET00370
C                                                                           SET00380
      EQUIVALENCE      (DATE(1), HEAD(22))                                  SET00390
C                                                                           SET00400
CCCCC ---------------------------------------------------------------------SET00410
CCCCC                                                                       SET00420
      INCLUDE COMBK2.LIST                                                   SET00430
CCCCC                                                                       SET00440
CCCCC ---------------------------------------------------------------------SET00450
CCCCC                                                                       SET00460
CCCCC                                                                       SET00470
CCCCC ---------------------------------------------------------------------SET00480
CCCCC                                                                       SET00490
CCCCC                                                                       SET00500
CCCCC ---------------------------------------------------------------------SET00510
CCCCC                                                                       SET00520
CCCCC                                                                       SET00530
C     INCLUDE COMBK6.LIST                                                   SET00540
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,              SET00550
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLUSV2,VERTX2,        SET00560
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),            SET00570
     *              KEEPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                 SET00580
     *              GRPCHK(61),GROUPS(124)                                  SET00590
      COMMON /CLASS/ APRFLG,BMCOMB,BMFEAT,BMFLG,NOCAT,THIJ1,IDATA1,         SET00600
     *               NFILE,STATKY,CATNAM(60),                               SET00610
     3               CLSSYM(60),CON(60),DET(60),FLDESC,FLDINF(6),           SET00620
     4               KCLSNA(60),NOCTCL(60),SUBCAT(60)                       SET00630
     *    ,NOCHAN,CHNVEC(30)                                                SET00640
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             SET00650
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,       SET00660
     *    DRUMAD,DRM,DS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    SET00670
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                        SET00680
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                  SET00690
     *  CRDUNT,PRTUNT,RANDIO                                                SET00700
C$END                                                                       SET00710
      DIMENSION            DATE(2),FILVEC(2),APRIOR(1),KATNO(60)            SET00720
      DIMENSION CARD(62),CLSSY(60)                                          SET00730
C                                                                           SET00740
      DATA       YRCD/'Y'/, NBCD/'N'/, MODBCD/'MODU'/, BLANK/' '/,          SET00750
     1           FILVEC/  1 ,  'F'  /                                       SET00760
      DATA ENDCD/'$END'/                                                    SET00770
      DATA CLSSY/'1','2','3','4','5','6','7','8','9','A','B','C','D',      SET00780
     *           'E','F','G','H','I','J','K','L','M','N','O','P','Q',      SET00790
```

FILE: SETUP2 FORTRAN A

```
     2             'R','S','T','U','V','W','X','Y','Z','_','|','¬','/',  SET00800
     3             '-','0','1','2','3','4','5','6','7','8','9','(',':',  SET00810
     4             '&','>','?',' ','.','*',' '/                           SET00820
      DO 5 I=1,60                                                         SET00630
    5 CLSSYM(I)=CLSSY(I)                                                  SET00840
C                                                                         SET00850
CCC   ---------------------------------------------------------------------SET00860
CCC                                                                       SET00870
C                                                                         SET00880
C                                                                         SET00890
      DIMENSION  BMATRX(BMCOMB,BMFEAT)                                    SET00900
C                                                                         SET00910
CCC                                                                       SET00920
CCC   ---------------------------------------------------------------------SET00930
CCC                                                                       SET00940
CCC                                                                       SET00950
CCC   ---------------------------------------------------------------------SET00960
CCC                                                                       SET00970
CCC                                                                       SET00980
CCC   ---------------------------------------------------------------------SET00990
C                                                                         SET01000
      NOCLSS=0                                                            SET01010
      FILEOP = 0                                                          SET01020
C                                                                         SET01030
CCC   GO READ SUPERVISOR INFORMATION                                      SET01040
CCC   ---------------------------------------------------------------------SET01050
C                                                                         SET01060
      CALL REDIF2(ARRAY,TOP,APRIOR,KATNO,BMATRX,PRIORI)                   SET01070
C                                                                         SET01080
CC                                                                        SET01090
C     GET TAPE READY                                                      SET01100
C                                                                         SET01110
  310     REWIND MAPTAP                                                   SET01120
          NFILE = ERRKEY                                                  SET01130
          IF (NFILE.LE.0) NFILE = 1                                       SET01140
          ISKIP = NFILE - 1                                               SET01150
          IF (ISKIP.LE.0) GO TO 400                                       SET01160
          CALL FSRSFL(MAPTAP,ISKIP,ISTAT)                                 SET01170
          IF (ISTAT.NE.0) CALL EXIT                                       SET01180
  400      CONTINUE                                                       SET01190
C                                                                         SET01200
CC                                                                        SET01210
CCC   CALL ROUTINE TO REDUCE THE ARRAYS                                   SET01220
CCC   ---------------------------------------------------------------------SET01230
C                                                                         SET01240
      CALL REDSAV(ARRAY,TOP,BMFLG)                                        SET01250
C                                                                         SET01260
CCC   TEST FOR CATEGORY 'FILE' INPUT -IF SO FORM CATEGORY/CLASS/SUBCLASS  SET01270
C        ASSOCIATION USING CLASS/SUBCLASS INFORMATION.                    SET01280
      IDUM = -765321                                                      SET01290
      IF(NOCAT.NE.IDUM)GO TO 406                                          SET01300
      FILEOP = 1                                                          SET01310
      NOCAT = NOCLS2                                                      SET01320
      DO 405 I=1,NOCLS2                                                   SET01330
      P=CLSID2 + I -1                                                     SET01340
      CATNAM(I) = ARRAY(P)                                                SET01350
      NOCTCL(I) = 1                                                       SET01360
      KCLSNA(I) = ARRAY(P)                                                SET01370
  405 CONTINUE                                                            SET01380
  406 CONTINUE                                                            SET01390
C*                                                                        SET01400
C*    DEFAULT -- SAME CHANNELS AS SELECTED FROM STAT FILE                 SET01410
C*                                                                        SET01420
      IF (NOCHAN .NE. 0)  GO TO 335                                       SET01430
      DO 330   I=1,NOFET2                                                 SET01440
  330 CHNVEC(I) = FETVC2(I)                                               SET01450
      NOCHAN = NOFET2                                                     SET01460
  335 CONTINUE                                                            SET01470
C                                                                         SET01480
CCC    COMPUTE BASE ADDRESSES FOR SCRACH ARRAY                            SET01490
CCC   THIJ1  -  BASE ADDRESS FOR TABLE COMPUTED IN THRESH CONTAINING      SET01500
CCC             CLASS-PAIR THRESHOLDS                                     SET01510
CCC   IDATA1-  BASE ADDRESS FOR DATA PASSED BACK FROM TAPERD              SET01520
C                                                                         SET01530
      THIJ1 = 1                                                           SET01540
      IDATA1 = THIJ1 + (NOSUB2-1)*(NOSUB2-2)/2 + NOSUB2                   SET01550
      IF (NOCAT .GT. 0)  IDATA1 = 1                                       SET01560
C                                                                         SET01570
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SETUP2 FORTRAN A

```
C                                                                       SET01590
C     STORE A BLANK IN DEFAULT SYMBOLS.  THIS WILL BE USED IN           SET01600
C     PRINTING THE MAP CLASSIFICATION FOR THE UNCLASSIFIED PIXEL        SET01610
C                                                                       SET01620
      CLSSYM(NOSUB2 + 1) = BLANK                                        SET01630
      IF (NOCAT .LE. 0)  GO TO 465                                      SET01640
C                                                                       SET01650
CC                                                                      SET01660
CC    ALL CLASSES MUST BE ASSIGNED  TO A CATEGORY                       SET01670
C                                                                       SET01680
      DO 410  II=1,NOCAT                                                SET01690
  410 NOCLSS = NOCTCL(II) + NOCLSS                                      SET01700
      IF ( NOCLSS .EQ. NOCLS2 ) GO TO  415                              SET01710
      WRITE(6,450)(ARRAY(CLSID2+I-1),I=1,NOCLS2)                        SET01720
      WRITE(6,460)(KCLSNA(I),I=1,NOCLSS)                                SET01730
      CALL CMERR                                                        SET01740
C                                                                       SET01750
C     SET UP KATNO ARRAY TO CONTAIN THE CATEGORY EACH CLASS             SET01760
C     BELONGS TO                                                        SET01770
C                                                                       SET01780
  415 CONTINUE                                                          SET01790
      NOCLAS = 0                                                        SET01800
      DO 435  L=1,NOCAT                                                 SET01810
      NOCLS1 = NOCLAS + 1                                               SET01820
      NOCLAS = NOCTCL(L) + NOCLS1 - 1                                   SET01830
      DO 425   J=NOCLS1,NOCLAS                                          SET01840
      CLSNAM = KCLSNA(J)                                                SET01850
      DO 420   K=1,NOCLS2                                               SET01860
      IF (CLSNAM .EQ. ARRAY(CLSID2-1+K)) GO TO 430                      SET01870
  420 CONTINUE                                                          SET01880
  440 WRITE(6,450)(ARRAY(CLSID2+I-1),I=1,NOCLS2)                        SET01890
  450 FORMAT(// ' AN ERROR HAS OCCURRED IN GROUPING CLASSES INTO CATEGORSET01900
     *IES.CHECK THE FOLLOWING :'// 5X,'1. NOT ALL OF THE CLASSES HAVE BESET01910
     *EN ASSIGNED TO A CATEGORY.'/ 5X, '2. A CLASS NAME ON THE CATEGORY SET01920
     *CARD HAS BEEN MISSPELLED.'// 10X, 'CLASS NAMES FROM SAVTAP FILE ARSET01930
     *E : ' / 5X, (10(A6,2X)))                                          SET01940
      WRITE(6,460)(KCLSNA(I),I=1,NOCLSS)                                SET01950
  460 FORMAT( / 10X, 'CLASS NAMES FROM CATEGORY CARDS ARE : '/ 5X,      SET01960
     *(10(A4,4X)))                                                     SET01970
      CALL CMERR                                                        SET01980
  430 KATNO(K) = L                                                      SET01990
  425 CONTINUE                                                          SET02000
  435 CONTINUE                                                          SET02010
C                                                                       SET02020
C     SET UP SUBCAT ARRAY TO CONTAIN THE CATEGORY EACH SUBCLASS         SET02030
C     BELONGS TO                                                        SET02040
C                                                                       SET02050
      DO 437  II = 1,NOSUB2                                             SET02060
      CLSNUM = CLSVC2(II)                                               SET02070
  437 SUBCAT(II) = KATNO(CLSNUM)                                        SET02080
  465 CONTINUE                                                          SET02090
C                                                                       SET02100
C     PRINT OUT THE SUPERVISOR INFORMATION                              SET02110
C     ------------------------------------------------------------------SET02120
C                                                                       SET02130
      WRITE(6,HEAD)                                                     SET02140
      WRITE(6,5012)                                                     SET02150
      IF ( STATKY .EQ. 1 ) WRITE(6,5014)                                SET02160
      IF ( NOCAT .GT. 0 ) WRITE(6,5018)                                 SET02170
      IF ( NOCAT .LE. 0 ) WRITE(6,5020)                                 SET02180
      IF (FILEOP.EQ.1) WRITE(6,5022)                                    SET02190
 5012 FORMAT(T2,'THE FOLLOWING OPTIONS HAVE BEEN SELECTED'/)            SET02200
 5014 FORMAT(T5,'PRINT MULTISPECTRAL STATISTICS.')                      SET02210
 5018 FORMAT(T5,'CATEGORY CLASSIFIER OPTION HAS BEEN SELECTED.')        SET02220
 5020 FORMAT(T5,'STANDARD CLASSIFIER OPTION HAS BEEN SELECTED.')        SET02230
 5022 FORMAT(T7,'ALSO CLASSES FROM STATFILE WILL BE CONSIDERED THE CATEGSET02240
     *ORIES FOR CLASSIFICATION')                                       SET02250
  500 CONTINUE                                                          SET02260
  502 FORMAT( // 1X,'SUPERVISOR INFORMATION : '//T5,'FILE NUMBER ....'SET02270
     *  T10/ T5,'NO. OF FIELDS.....',                                  SET02280
     2 I4/T5,'NO. OF CLASSES.....',I4/T5,'NO. OF SUBCLASSES.....'     SET02290
     3 I4/T5,'NO. OF CHANNELS.....',I4//////)                         SET02300
C                                                                       SET02310
      IF (APRFLG .LE. 0 )  GO TO 6045                                   SET02320
C                                                                       SET02330
C     NORMALIZE APRIORI VALUES THAT WERE INPUT BY USER                  SET02340
C                                                                       SET02350
      NORM = 0.0                                                        SET02360
      DO  600  I=1,APRFLG                                               SET02370
```

```
FILE: SETUP2   FORTRAN  A

  600 NORM = NORM + ABS(PRIORI(I))                                          SET02380
      IF ( NORM .LT. .999999 .OR. NORM .GT. 1.000001) WRITE(6,6041)         SET02390
 6041 FORMAT(TS,'** USER INPUT A PRIORI VALUES DO NOT SUM TO 1.0.  INPUT    SET02400
     * VALUES WERE NORMALIZED **')                                         SET02410
      DO 603   I=1,APRFLG                                                  SET02420
  603 PRIORI(I) = ABS(PRIORI(I))/NORM                                       SET02430
      IF (APRFLG .EQ. NOSUB2)  GO TO 610                                    SET02440
      IF (APRFLG .EQ. NOCLS2)  GO TO 620                                    SET02450
      IF (APRFLG .EQ. NOCAT)  GO TO 630                                     SET02460
C                                                                           SET02470
      WRITE(6,6010)                                                         SET02480
 6010 FORMAT(TS,'** ERROR IN A PRIORI CONTROL CARD. USER INPUT VALUES IG    SET02490
     *NORED. **')                                                          SET02500
      GO TO 6045                                                            SET02510
C                                                                           SET02520
C     APRIORI VLAUES INPUT BY SUBCLASSES                                    SET02530
C                                                                           SET02540
  610 DO 615  I=1,NOSUB2                                                    SET02550
  615 APRIOR(I) = PRIORI(I)                                                 SET02560
      WRITE(6,6015)                                                         SET02570
 6015 FORMAT(TS,  'APRIORI VALUES INPUT BY SUBCLASSES     ')               SET02580
      GO TO 6070                                                            SET02590
C                                                                           SET02600
C     APRIORI VALUES INPUT BY CLASSES                                       SET02610
C                                                                           SET02620
  620 JJ = 0                                                                SET02630
      DO 625  I=1,NOCLS2                                                    SET02640
      NOCL = ARRAY(SUBNO2-1+I)                                              SET02650
      APRI = PRIORI(I) / FLOAT(NOCL)                                        SET02660
      DO 625  J=1,NOCL                                                      SET02670
      JJ = JJ + 1                                                           SET02680
  625 APRIOR(JJ) = APRI                                                     SET02690
      WRITE(6,6025)                                                         SET02700
 6025 FORMAT(TS,  'APRIORI VALUES INPUT BY CLASSES.  APRIOR(I) = CLASS(J   SET02710
     *) APRIOR / (NO. SUBCLASSES IN CLASS(J)) ')                           SET02720
      GO TO 6070                                                            SET02730
C                                                                           SET02740
C     APRIORI VALUES INPUT BY CATEGORIES                                    SET02750
C                                                                           SET02760
  630 CLSNM = 0                                                             SET02770
      DO 660   I=1,NOCAT                                                    SET02780
      APRI = PRIORI(I)                                                      SET02790
      II = NOCTCL(I)                                                        SET02800
      M = 1                                                                 SET02810
      NOSBCL = 0                                                            SET02820
      DO 645   K=1,II                                                       SET02830
      DO 640   KK=1,NOCLS2                                                  SET02840
      IF (KCLSNA(K+CLSNM) .EQ. ARRAY(CLSID2-1+KK))  GO TO 645               SET02850
  640 CONTINUE                                                              SET02860
  645 NOSBCL = ARRAY(SUBNO2-1+KK) + NOSBCL                                  SET02870
      APRIO = APRI / FLOAT(NOSBCL)                                          SET02880
      DO 650   KKK=1,NOSUB2                                                 SET02890
      IF ( I .EQ. SUBCAT(KKK))  APRIOR(KKK) = APRIO                         SET02900
      IF ( I .EQ. SUBCAT(KKK))  M = M + 1                                   SET02910
      IF (M .GT. NOSBCL)  GO TO 655                                         SET02920
  650 CONTINUE                                                              SET02930
  655 CLSNM = CLSNM + II                                                    SET02940
  660 CONTINUE                                                              SET02950
      WRITE(6,6060)                                                         SET02960
 6060 FORMAT(TS,  'APRIORI VALUES INPUT BY CATEGORY.  APRIOR(I) = CATEGOSET02970
     *RY(J) APRIORI / (NO. SUBCLASSES IN CATEGORY(J)) ')                   SET02980
      GO TO 6070                                                            SET02990
C                                                                           SET03000
 6045 IF(APRFLG.EQ.-777777)GO TO 760                                        SET03010
C     FLAG FOR SAVTAP COMPUTATION                                           SET03020
      IF (NOCAT.LE. 0) GO TO 605                                            SET03030
C                                                                           SET03040
C     COMPUTE DEFAULT APRIORI VALUSE FOR CATEGORY CLASSIFIER                SET03050
C                                                                           SET03060
      CLSNM = 0                                                             SET03070
      APRI = 1.0 / FLOAT(NOCAT)                                             SET03080
      DO 750  I=1,NOCAT                                                     SET03090
      II = NOCTCL(I)                                                        SET03100
      M = 1                                                                 SET03110
      NOSBCL = 0                                                            SET03120
      DO 715  K=1,II                                                        SET03130
      DO 705  KK=1,NOCLS2                                                   SET03140
      IF (KCLSNA(K+CLSNM) .EQ. ARRAY(CLSID2-1+KK))  GO TO 715               SET03150
  705 CONTINUE                                                              SET03160
```

FILE: SETUP2 FORTRAN A

```
  715 NOSBCL = ARRAY(SUBNO2-1+KK) + NOSBCL                                  SET03170
      APRIO = APRI * 1.0 / FLOAT(NOSBCL)                                    SET03180
      DO 720  KKK=1,NOSUB2                                                  SET03190
      IF ( I .EQ. SUBCAT(KKK) ) APRIOR(KKK) = APRIO                         SET03200
      IF ( I .EQ. SUBCAT(KKK) ) M = M + 1                                   SET03210
      IF( M .GT. NOSBCL)  GO TO 740                                         SET03220
  720 CONTINUE                                                              SET03230
  740 CLSNM  = CLSNM + II                                                   SET03240
  750 CONTINUE                                                              SET03250
      GO TO 6053                                                            SET03260
C                                                                           SET03270
C     COMPUTE APRIORI VALUES FROM STATFILE                                  SET03280
C                                                                           SET03290
  760 TKEPTS = 0                                                            SET03300
      DO 765 I=1,NOSUB2                                                     SET03310
  765 TKEPTS = TKEPTS + KEPPTS(I)                                           SET03320
C     TOTAL ALL SUBCLASS PIXELS                                             SET03330
      DO 770 I = 1,NOSUB2                                                   SET03340
      APRIOR(I) = FLOAT(KEPPTS(I))/FLOAT(TKEPTS)                            SET03350
  770 CONTINUE                                                              SET03360
       IPAT = STAFIL + 1                                                    SET03370
       WRITE (6,775) IPAT                                                   SET03380
  775 FORMAT(T5,'APRIORI VALUES FROM STATFILE',I3,'   APRIORI= NO. PI       SET03390
     *XELS IN SUBCLASS/TOTAL NO. PIXELS IN ALL SUBCLASSES ***')             SET03400
      GO TO 6070                                                            SET03410
C                                                                           SET03420
C     COMPUTE DEFAULT APRIOR VALUES FOR STANDARD CLASSIFIER                 SET03430
C                                                                           SET03440
  605 NORM = 1.0 / FLOAT(NOSUB2)                                            SET03450
      DO 606 I=1,NOSUB2                                                     SET03460
  606 APRIOR(I) = NORM                                                      SET03470
C                                                                           SET03480
      WRITE(6,6050)                                                         SET03490
C                                                                           SET03500
 6050 FORMAT(T5,'DEFAULT APRIORI PROBABILITY VALUES WILL BE USED.  SUBCL    SET03510
     *ASS(I) = 1.0/(NO. OF SUBCLASSES)')                                    SET03520
      GO TO 6070                                                            SET03530
 6053 CONTINUE                                                              SET03540
      WRITE(6,6055)                                                         SET03550
 6055 FORMAT(T5,'DEFAULT APRIORI PROBABILITY VALUES FOR SUBCLASS(I) = 1.    SET03560
     *0/(NO OF CATEGORIES)*(NO. OF SUBCLASSES IN CATEGORY(J)')              SET03570
C                                                                           SET03580
 6070 APRFLG = NOSUB2                                                       SET03590
      WRITE(6,602)NFILE,NOFLD2,NOCLS2,NOSUB2,NOFET2                         SET03600
  607 IF( BMFLG .LE. 0)  GO TO 700                                          SET03610
C                                                                           SET03620
      CALL WRTBM(BMATRX,BMCOMB,BMFEAT,FETVC2)                               SET03630
C                                                                           SET03640
  700 CONTINUE                                                              SET03650
C                                                                           SET03660
      IF (NOCHAN .NE. NOFET2) WRITE(6,800)                                  SET03670
  800 FORMAT(' NO. OF CHANNELS REQUESTED FOR DATA TAPE AND NO. OF CHANNE    SET03680
     *LS ON STAT'/' FIEL MUST BE EQUAL')                                    SET03690
      IF (NOCHAN .NE. NOFET2) CALL CMERR                                    SET03700
C     ----------------------------------------------------------------      SET03710
C                                                                           SET03720
C     WRITE FIRST RECORD ON THE CLASSIFICATION RESULTS OUTPUT FILE.         SET03730
C     MAPTAP                                                                SET03740
C                                                                           SET03750
C     ----------------------------------------------------------------      SET03760
C                                                                           SET03770
C                                                                           SET03780
C     HEADER RECORD  NO. 1 FOR MAPTAP                                       SET03790
C                                                                           SET03800
      WRITE(MAPTAP)(DATE(I),I=1,2),BMFLG,BMCOMB,BMFEAT,NOCLS2,              SET03810
     *     NOFLD2,NOSUB2,NOFET2,TOTVT2,NOCAT,VARS72,(FETVC2(I),I=1,         SET03820
     *     NOFET2)                                                          SET03830
C                                                                           SET03840
C                                                                           SET03850
      RETURN                                                                SET03860
       END                                                                  SET03870
```

FILE: SETUS FORTRAN A

```
      SUBROUTINE SETUS(AVEMTX,COVMTX,DIAG,MEAN,COV,NCHAN,NOCLS2,
     *VARSZ2,JJ)
C
C     USING THE LOWER TRIANGULAR MATRIX AND DIAGONAL MATRIX FOR
C     CLASS(JJ) IN COVMTX, THE FULL COVARIANCE MATRIX FOR CLASS(JJ) IS
C     COMPUTED, AS L X D X L* , AND RETURNED IN COV
C     ALSO, THE MEAN VECTOR FOR CLASS(JJ) IS EXTRACTED FROM AVEMTX,
C     AND IS RETURNED IN MEAN
C
      INTEGER ROW,COL,VARSZ2
      REAL MEAN(NCHAN)
      DIMENSION COV(NCHAN,NCHAN)
      DIMENSION AVEMTX(NCHAN,NOCLS2),COVMTX(VARSZ2,NOCLS2),DIAG(NCHAN)
C
      N1 = 0
      N2 = 0
      DO 2 ROW = 1,NCHAN
      N1 = N1 + ROW
      DIAG(ROW) = COVMTX(N1,JJ)
      MEAN(ROW) = AVEMTX(ROW,JJ)
    2 COVMTX(N1,JJ) = 1.0
C
      DO 5 ROW = 1,NCHAN
      II = (ROW * (ROW - 1)) / 2
      DO 4 COL = 1,ROW
      SUM = 0.0
      JK = (COL * (COL - 1)) / 2
      DO 3 K = 1,COL
      JP = JK + K
      IP = II + K
    3 SUM = SUM + COVMTX(JP,JJ) * COVMTX(IP,JJ) * DIAG(K)
      COV(ROW,COL) = SUM
    4 COV(COL,ROW) = SUM
    5 CONTINUE
C
      II = 0
      DO 6 COL = 1,NCHAN
      II = II + COL
    6 COVMTX(II,JJ) = DIAG(COL)
      RETURN
      END
```

FILE: THRESH FORTRAN A

```
      SUBROUTINE THRESH(NOCLS2,NOFET2,NPL1,   APRIOR,AVEMTX,COVMTX,DET,  THR00010
     1   VARSZ2,S1,S2,U1,U2,BB,DIAG,THIJ)                                  THR00020
C                                                                         THR00030
C                                                                         THR00040
C                                                                         THR00050
C     SUBROUTINE THRESH COMPUTES THE CLASS-PAIR THRESHOLDS , AND RETURNS  THR00060
C     THEM IN ''SYMETTRIC'' STORAGE ( THIJ) IN THE FOLLOWING MANNER:      THR00070
C                                                                         THR00080
C                 2,1                                                     THR00090
C                 3,1  3,2                                                THR00100
C                 4,1  4,2  4,3                                           THR00110
C                 5,1  5,2  5,3  5,4                                      THR00120
C                 6,1  6,2  6,3  6,4  6,5                                 THR00130
C                       .                                                 THR00140
C                       .                                                 THR00150
C                       .                                                 THR00160
C                                                                         THR00170
C                                                                         THR00180
C     THRESH REQUIRES THE FUNCTION SUBPROGRAM, G ( WITH ALTERNATE ENTRY   THR00190
C     POINT, GG ), AND THE SUBROUTINE  FALSY .                            THR00200
C                                                                         THR00210
C     ------------------------------------------------------              THR00220
C               COMPUTATION OF THE CLASS-PAIR THRESHOLDS                  THR00230
C     ------------------------------------------------------              THR00240
C                                                                         THR00250
C     FOR     U1 = MEAN VECTOR, CLASS(1) OF THE CLASS-PAIR                THR00260
C             U2 =  ''    ''   , CLASS(2) ''  ''   ''   ''                THR00270
C             COV1 =  COVARIANCE MATRIX, CLASS(1) OF THE CLASS-PAIR       THR00280
C             COV2 =   ''        ''    , CLASS(2)  ''  ''   ''   ''       THR00290
C             C1 =  2 * LOG APRIORI(CLASS 1) - LOG DETERMINANT(CLASS 1)   THR00300
C             C2 =  2 * LOG APRIORI(CLASS 2) - LOG DETERMINANT(CLASS 2)   THR00310
C             AND                                                         THR00320
C                                                                         THR00330
C             C = C2 - C1                                                 THR00340
C                                                                         THR00350
C       (1)   IF C1  .LE. C2 - (U1-U2)* X COV2**-1 X (U1-U2), THIJ= C1    THR00360
C                                                                         THR00370
C       (2)   IF  C2  .LE. C1 - (U1-U2)* X COV1**-1 X (U1-U2), THIJ= C2   THR00380
C                                                                         THR00390
C       (3)   IF NEITHER (1) NOR (2), COMPUTE THE CLASS-PAIR THRESHOLD    THR00400
C             ITERATIVELY AS FOLLOWS:                                     THR00410
C     FIND A NUMBER,  0 .GE. X .LE. 1  , SO THAT THE SOLUTION VECTOR,     THR00420
C     H(X), OF THE SYSTEM OF EQUATIONS,                                   THR00430
C                                                                         THR00440
C     (3A)    (  (1-X)* COV1**-1 + X * COV2**-1) * H(X)                   THR00450
C               = (1-X) * COV1**-1 * U1 + X * COV2**-1 * U2               THR00460
C                                                                         THR00470
C     ALSO SATISFIES   G( H(X) ) = C2 - C1 , WHERE                        THR00480
C                                                                         THR00490
C     (3B)      G( H(X) ) = ( H(X) - U2)*  * COV2**-1  * ( H(X) - U2)     THR00500
C                         - ( H(X) - U1)*  *  COV1**-1 * ( H(X) - U1 )    THR00510
C                                                                         THR00520
C                                                                         THR00530
C     SUBROUTINE FALSY DETERMINES 3-POINT INTERVALS IN THE RANGE,         THR00540
C      0 .GE. X .LE. 1  , FITS A QUADRATIC Q(X) TO THE THREE POINTS AND   THR00550
C      OBTAINS THE ROOT, X , FOR  Q(X)= C2-C1. THE ROOT,X , OF THIS       THR00560
C      QUADRATIC  APPROXIMATION OF  G( H(X) ) IS SENT TO FUNCTION G.      THR00570
C                                                                         THR00580
C      FUNCTION G  OBTAINS THE SOLUTION VECTOR , H(X), FOR THE SYSTEM     THR00590
C      OF EQUATIONS  (3A) ,  AND APPLIES THE  H(X) TO OBTAIN THE VALUE    THR00600
C      OF  G( H(X) )   (EQN. 3B),  FALSY TESTS THE VALUE OF  G( H(X) )    THR00610
C      IN RELATION TO  C2-C1 .WITHIN A PRE-SET TOLERANCE ON THE RESIDUAL. THR00620
C      IF   G( H(X) ) .NE.  C2-C1 + TOLERANCE, ITERATION CONTINUES IN     THR00630
C      FALSY WITH NEW  INTERVALS, QUADRATIC FIT OF THE INTERVALS, ROOT    THR00640
C      OF THE APPROXIMATING  QUADRATIC  SENT TO  FUNCTION G.  FALSY       THR00650
C      RETURNS TO THRESH WHEN A ROOT, X , OF THE APPROXIMATING QUADRATIC  THR00660
C      IS FOUND WHICH YIELDS AN H(X) THAT SATISFIES G( H(X) ) = C2-C1     THR00670
C      WITHIN THE PRE-SET TOLERANCE.                                      THR00680
C                                                                         THR00690
C      THRESH COMPUTES THE CLASS-PAIR THRESHOLD, THIJ, BY OBTAINING FROM  THR00700
C      G  THE EVALUATION OF  G( H(X) ) FOR THE  X  RETURNED BY FALSY :    THR00710
C                                                                         THR00720
C      THIJ = .5 * ( C1 - ( H(X)- U1)*  *  COV1**-1  *  ( H(X)-U1)  )     THR00730
C                                                                         THR00740
C                                                                         THR00750
C                                                                         THR00760
C                                                                         THR00770
      INTEGER  VARSZ2                                                     THR00780
      ------------------------------------------------------------------  THR00790
```

FILE: THRESH FORTRAN A

```
C     ------------------------------------------------------------------THR00800
CC                                                                      THR00810
C                                                                       THR00820
      DIMENSION THIJ(1)                                                 THR00830
      DIMENSION  AVEMTX(NOFET2,NOCLS2), COVMTX(VARSZ2,NOCLS2),          THR00840
     1 S1(NOFET2,NOFET2),S2(NOFET2,NOFET2), DIAG(NOFET2) ,              THR00850
     2  APRIOR(NOCLS2), U1(NOFET2), U2(NOFET2), DET(NOCLS2) ,           THR00860
     3   BB(NOFET2,NPL1)                                               THR00870
C                                                                       THR00880
CC                                                                      THR00890
C                                                                       THR00900
      K = NOFET2                                                        THR00910
      KP1 = NPL1                                                        THR00920
      DO  66  I=1,NOCLS2                                                THR00930
      TAI = 2.0 * ALOG( APRIOR(I) )                                     THR00940
      CALL SETUS(AVEMTX,COVMTX,DIAG,U1,S1,NOFET2,NOCLS2,VARSZ2,I)       THR00950
      D1 = ALOG( DET(I) )                                               THR00960
      DO  65  J=1,NOCLS2                                                THR00970
      TAJ = 2.0 * ALOG( APRIOR(J) )                                     THR00980
      CALL SETUS(AVEMTX,COVMTX,DIAG,U2,S2,NOFET2,NOCLS2,VARSZ2,J)       THR00990
      D2 = ALOG( DET(J) )                                               THR01000
      C = D2 - D1 + 2.0 * ALOG( APRIOR(I)/APRIOR(J) )                   THR01010
      KT=0                                                              THR01020
      IF(J.EQ.I) GO TO 65                                               THR01030
C                                                                       THR01040
CCC   COMPUTE THE SYMMETRIC MATRIX STORAGE LOCATION                     THR01050
      CLASS I-J PAIR THRESHOLD VALUE                                    THR01060
CCC                                                                     THR01070
      JM1 = J - 1                                                       THR01080
      JM2 = J - 2                                                       THR01090
      NTH = ( JM1 * JM2)/2 + I                                          THR01100
      XL=0.                                                             THR01110
      XU=1.0                                                            THR01120
      G0 = G(XL,S1,S2,U1,U2,BB,KT,T,K,KP1)                              THR01130
      G1 = G(XU,S1,S2,U1,U2,BB,KT,T,K,KP1)                              THR01140
      IF(G0.GT.C) GO TO 14                                              THR01150
      IF(G1.LT.C) GO TO 15                                              THR01160
      KC=1                                                              THR01170
       CALL FALSY(XL,XU,C,FXL,FXU,KC,XN,KT,T,K,KP1,S1,S2,U1,U2,BB)      THR01180
      KT=1                                                              THR01190
      T = G(XU,S1,S2,U1,U2,BB,KT,T,K,KP1)                               THR01200
      THIJ(NTH) = TAJ - D2 - T                                          THR01210
      GO TO 65                                                          THR01220
   14 THIJ(NTH) = TAI - D1                                              THR01230
      GO TO 65                                                          THR01240
   15 THIJ(NTH) = TAJ - D2                                              THR01250
   65 CONTINUE                                                          THR01260
   66 CONTINUE                                                          THR01270
      NUMTH = ( NOCLS2 * (NOCLS2-1) )/2                                 THR01280
      NTH = NOCLS2 - 1                                                  THR01290
C     WRITE(6,9065)                                                     THR01300
CCC   JO = 0                                                            THR01310
CCC   DO  67  I=1,NTH                                                   THR01320
CCC   IO = ( I * (I-1))/2 + 1                                           THR01330
CCC   JO = JO + I                                                       THR01340
C  67 WRITE(6,9050)  ( THIJ(J), J=IO,JO)                                THR01350
C9065 FORMAT(//// 2X,'THRESHOLD ARRAY  ---  SYMETTRIC STORAGE ...' )    THR01360
C9050 FORMAT(// 2X,10F12.5//4X,10F12.5//6X,10F12.5//)                   THR01370
C                                                                       THR01380
      DO  690  I=1,NUMTH                                                THR01390
  690 THIJ(I) = THIJ(I) * .5                                            THR01400
C     CLASS - PAIR THRESHOLDS                                           THR01410
CC                                                                      THR01420
   70 RETURN                                                            THR01430
      END                                                               THR01440
```

12. DISPLAY PROCESSOR

FILE: DSPLAY FORTRAN A

```
      SUBROUTINE DSPLAY(ARRAY,TOP)                                          DSP00010
C                                                                           DSP00020
      IMPLICIT INTEGER (A-H,O-Z)                                            DSP00030
C                                                                           DSP00040
C                                                                           DSP00050
      DIMENSION ARRAY(1)                                                    DSP00060
C                                                                           DSP00070
CCCC|-------------------------------------------------------------------|   DSP00080
CCCC|-------------------------------------------------------------------|   DSP00090
CCCC|                                                                   |   DSP00100
CCCC|     CALL..    CALL DSPLAY(ARRAY,TOP)                             |   DSP00110
CCCC|                                                                   |   DSP00120
CCCC|     ARGS..    ARRAY -  SEE 'MONTOR'                              |   DSP00130
CCCC|                TOP   -  SEE 'MONTOR'                              |   DSP00140
CCCC|                                                                   |   DSP00150
CCCC|     PURPOSE..COORDINATES ROUTINES FOR DISPLAYING CLASSIFICATION   |   DSP00160
CCCC|              MAP AND PERFORMANCE TABLES.                          |   DSP00170
CCCC|                                                                   |   DSP00180
CCCC|-------------------------------------------------------------------|   DSP00190
CCCC|-------------------------------------------------------------------|   DSP00200
CCCC                                                                        DSP00210
      INCLUDE CMHK10.LIST                                                   DSP00220
C     INCLUDE COMT10                                                        DSP00230
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNU(60),       DSP00240
     *             SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,             DSP00250
     *             PCFOKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,      DSP00260
     *             PLTKEY,RMFLG,RMCOMB,RMFEAT,CDATE(2),                   DSP00270
     *             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,      DSP00280
     *             THRES(60),SYMMTX(66),HIGH(60),CON(60)                  DSP00290
     *             ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)                DSP00300
     *             ,NOSUB2,NOTHED,TOTVT2,NOCLS2                           DSP00310
     *             ,KATNO(60),NOCAT,FILTER,MAPFMT                         DSP00320
     *             ,DESKEY,DESUNI,DESOTH,CROP ,ACROP,AOTHER,ATOTAL        DSP00330
     *             ,SITE(6),ANALYS(5),CAM(15),CRPKEY,KEPPTS(60)           DSP00340
     *             ,OOTKEY,OUTERR                                         DSP00350
C*                                                                          DSP00360
CC*   COMMON BLOCK DISPL IS USED ONLY IN THE DISPLAY PROCESSOR              DSP00370
CC*                                                                         DSP00380
CC*                                                                         DSP00390
CC*   DEFINITIONS                                                           DSP00400
CC*                                                                         DSP00410
CC*     CATFLG - FLAG INDICATING WHETHER OR NOT CATEGORY PERFORMANCE        DSP00420
CC*              REPORTS MUST BE GENERATED.                                 DSP00430
CC*     CATNAM - NAMES OF CATEGORIES. READ FROM MAPTAP.                     DSP00440
CC*     CLSNAM - NAMES OF CLASSES. READ FROM MAPTAP.                        DSP00450
CC*     SUBNAM - NAMES OF SUBCLASSES. READ FROM MAPTAP.                     DSP00460
CC*     SUBCAT - SUBCLASS-CATEGORY CORRESPONDENCE VECTOR                    DSP00470
CC*              (SUBCAT(I)=M  MEANS SUBCLASS I BELONGS TO CATEGORY M)      DSP00480
CC*     CLSSUB - SUBCLASS-CLASS CORRESPONDENCE VECTOR.                      DSP00490
CC*              (CLSSUB(I)=M  MEANS SUBCLASS I BELONGS TO CLASS M)         DSP00500
CC*     NOMAP  - TRIGGER INDICATING WHETHER OR NOT A MAP IS TO BE PRINTED   DSP00510
CC*     TOTVT3 - TOTAL NO. OF VERTICES IN INPUT TEST FIELDS.                DSP00520
CC*     NOSUB3 - NO. OF SUBCLASSES USED IN CLASSIFY PLUS ONE,FOR THE        DSP00530
CC*              THRESHOLD CLASS.                                           DSP00540
CC*     PCFOKY - KEY INDICATING WHETHER OR NOT GROUND TRUTH PERFORMANCE     DSP00550
CC*              REPORTS ARE TO BE PRINTED ON A PER FIELD BASIS.            DSP00560
CC*     TSTKEY - KEY INDICATING WHETHER OR NOT TEST FIELDS WERE INPUT.      DSP00570
CC*     TRNKEY - KEY INDICATING WHETHER OR NOT TRAINING FIELDS ARE TO       DSP00580
C*               BE OUTLINED.                                              DSP00590
C      CONTINUE                                                             DSP00600
CC*     THRSKY - THRESHOLD KEY                                              DSP00610
CC*              =1 APPLY CHI-SQUARE THRESHOLDS                             DSP00620
CC*              =2 APPLY EMPIRICAL  THRESHOLDS                             DSP00630
CC*              =3 APPLY USER-INPUT THRESHOLDS                             DSP00640
CC*              =4 APPLY FISHER DISTRIBUTION THRESHOLD                     DSP00650
CC*              =0 NO THRESHOLDING                                         DSP00660
CC*     STATKY - KEY FOR PRINTING STATS FROM MAPTAP                         DSP00670
CC*     EMPTRS - EMPIRICAL THRESHOLDING FLAG                                DSP00680
CC*     THRSVA - USER-INPUT THRESHOLD VALUE FLAG                            DSP00690
CC*     PLTKEY - FLAG FOR PRINTING CUMMULATIVE HISTOGRAMS OF QUADRATIC      DSP00700
CC*              FORM.                                                      DSP00710
CC*     RMFLG  - FLAG INDICATING WHETHER OR NOT A B-MATRIX WAS              DSP00720
CC*              APPLIED IN CLASSIFY.                                       DSP00730
CC*     RMCOMB - NO. OF LINEAR COMBINATIONS IN B-MATRIX                     DSP00740
CC*     RMFEAT - NO. OF CHANNELS USED IN COMPUTING B-MATRIX                 DSP00750
CC*     CDATE  - DATE OF CLASSIFICATION                                     DSP00760
CC*     FLDSV2 - ADDRESS IN 'ARRAY' FOR TRAINING FIELD INFORMATION.         DSP00770
CC*              FOR EACH TRAINING FIED 4 PIECES OF INFORMATION ARE         DSP00780
```

FILE: DSPLAY FORTRAN A

```
C*                STORED - 1=FIELD NAME                                          DSP00800
C*                         2=CLASS NO.                                           DSP00810
C*                         3=SUBCLASS NO.                                        DSP00820
C*                         4=NO. OF VERTICES                                     DSP00830
C*       FIELD2 - ADDRESS IN 'ARRAY' FOR RECTANGULAR AREA SURROUNDING            DSP00840
C*                EACH TRAINING FIELD. FOR EACH TRAINING FIELD 5 PIECES          DSP00850
C*                OF INFORMATION ARE STORED.                                     DSP00860
C*                         1=LINE START                                          DSP00870
C*                         2=LINE END                                            DSP00880
      CONTINUE                                                                  DSP00890
C*                       3=SAMPLE START                                         DSP00900
C*                       4=SAMPLE END                                           DSP00910
C*                       5=POINTER INTO VERTEX ARRAY FOR VERTICES               DSP00920
C*                         OF THIS FIELD.                                       DSP00930
C*       VERTX2 - ADRESS IN 'ARRAY' FOR TRAINING FIELD VERTICES.                 DSP00940
C*       FLDSV3 - SAME AS FLDSV2 FOR TEST FIELDS                                 DSP00950
C*       FIELD3 - SAME AS FIELD2 FOR TEST FIELDS                                 DSP00960
C*       VERTX3 - SAME AS VERTX2 FOR TEST FIELDS                                 DSP00970
C*       PCTIN3 - ADDRESS IN 'ARRAY' FOR PERFORMANCE TABLE.                      DSP00980
C*       THRES  - THRESHOLD VALUES                                               DSP00990
C*       SYMMTX - SYMBOLS FOR EACH SUBCLASS. PLUS THRESHOLD SYMBOL               DSP01000
C*                AND OUTLINE SYMBOLS.                                           DSP01010
C*       HIGH   - THRESHOLD REJECTION PERCENTAGE - EMPIRICAL OPTION              DSP01020
C*       CON    - CONSTANT FACTOR FROM PROBABILITY DENSITY FUNCTION              DSP01030
C*                FROM CLASSIFY. ONE FOR EACH SUBCLASS.                          DSP01040
C*       FLDKEY - KEY INDICATING WHETHER GROUND TRUTH FIELDS ARE                 DSP01050
C*                ASSOCIATED WITH CLASSES OR SUBCLASSES.                         DSP01060
C*       NOFLD2 - NO. OF TRAINING FIELDS                                         DSP01070
C*       NOFLD3 - NO. OF TEST FIELDS                                             DSP01080
C*       NOFET2 - NO. OF CHANNELS USED IN CLASSIFICATION.                        DSP01090
C*       FETVCT - CHANNELS USED IN CLASSIFICATION.                               DSP01100
C*       NOSUB2 - NO. OF SUBCLASSES USED IN CLASSIFICATION.                      DSP01110
C*       NOTRFD - NO. OF GROUND TRUTH FIELDS FOR WHICH PERFORMANCE               DSP01120
C*                TABLES WILL BE MADE. EQUALS NOFLD2 OR NOFLD3.                  DSP01130
C*       TOTVT2 - TOTAL NO. OF VERTICES FOR TRAINING FIELDS.                     DSP01140
C*       NOCLS2 - NO. OF CLASSES USED IN CLASSIFICATION.                         DSP01150
C*       KATNO  - CLASS - CATEGORY CORRESPONDENCE VECTOR                         DSP01160
      CONTINUE                                                                  DSP01170
C*                (KATNO(I)=M  MEANS CLASS I IS IN CATEGORY M)                   DSP01180
C*       NOCAT  - NO. OF CATEGORIES.                                             DSP01190
C*       FILTER - FLAG FOR SPATIAL FILTERING OPTION.                             DSP01200
C*     MAPFMT - FORMAT FOR OUTPUT MAP TAPE                                       DSP01210
C*       DESKEY - KEY INDICATING WHETHER OR NOT DESIGNATED FIELDS WERE IN        DSP01220
C*       DESUNI - NO. FOR DESIGNATED UNIDENTIABLE  (NOSUB2+5)                    DSP01230
C*       DESOTH - NO. FOR DESIGNATED OTHER  (NOSUB2+6)                           DSP01240
C*       CROP   - NAME OF CROP FOR WHICH INTENSIVE TEST SITE SUMMARY             DSP01250
C*                REPORT IS TO BE PRINTED. CROP IS TO BE COMPARED WITH OT        DSP01260
C*       ACROP  - ACRES OF 'CROP'  - USER INPUT                                  DSP01270
C*       AOTHER - ACRES OF 'OTHER' - USER INPUT                                  DSP01280
C*       ATOTAL - TOTAL ACRES IN CLASSIFIED SEGMENT                              DSP01290
C*       SITE   - NAME OF SITE (CLASSIFIED SEGMENT)                              DSP01300
C*       ANALYS - NAME OF ANALYST PERFORMING STUDY                               DSP01310
C*       CAMS   - NAME OF PROCEDURE CONFIGURATION USED IN STUDY                  DSP01320
C*       CRPKEY - KEY FOR GENERATING INTENSIVE TEST SITE SUMMARY REPORT          DSP01330
C*-      KEPPTS - TOTAL NUMBER PIXELS IN EACH SUBCLASS                           DSP01340
C*                                                                               DSP01350
C*       DOTKEY - KEY INDICATING WHETHER OR NOT DOT DATA CLASSIFICATION          DSP01360
C*     PERFORMANCE SUMMARIES ARE TO BE PROCESSED:  DOTKEY = 0 , NO DOT           DSP01370
C*     DATA PROCESSING ;  DOTKEY .GT. 0 , DOT PERFORMANCE SUMMARIES              DSP01380
C*     ARE PROVIDED ( IF DOTERR = 0 ) .                                          DSP01390
C*                                                                               DSP01400
C*       DOTERR - FLAG INDICATING THAT THE DOT DATA PROCESSING WILL BE           DSP01410
C*     DISCONTINUED DUE EITHER TO DISAGREEMENT OF LABELED  VS                    DSP01420
C*     CLASSIFICATION CATEGORY NAMES. OR ERRONEOUS PARAMETERS ON                 DSP01430
C*     DOTFILE CONTROL CARD .                                                    DSP01440
      CONTINUE                                                                  DSP01450
C*                                                                              DSP01460
C*                                                                              DSP01470
C$END                                                                           DSP01480
C*                                                                              DSP01490
C*     SETUP3 WILL READ FIRST 2 RECORDS FROM MAPTAP, AND CALL REDIF3            DSP01500
C*     TO READ IN CONTROL CARDS.  ALL OF THE PARAMETERS IN COMMON BLOCK         DSP01510
C*     DISPL ARE INITIALIZE BEFORE RETURNING TO THIS ROUTINE IN ADDITION        DSP01520
C*     TRAINING AND/OR TEST FIELD DEFINITIONS WILL BE STORED IN 'ARRAY'         DSP01530
C*                                                                              DSP01540
C                                                                               DSP01550
C***       THIS ADDED OR CHANGED NOV.13.1978 TO INCLUDE LIST PROS.              DSP01560
C                                                                               DSP01570
            REAL  ALP(2)                                                        DSP01580
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: DSPLAY FORTRAN A

```
      CALL SETUP3(ARRAY,TOP,LISTSW,GTUNIT,GTFILE,AIUNIT,AIFILE,      DSP01590
     *    PPUNIT,PPFILE,NAMECT,ALP)                                  DSP01600
C•                                                                   DSP01610
C••   DSPLY1 WILL READ NEXT 2 RECORDS FROM MAPTAP AND PRINT THE      DSP01620
C••   STATISTICS IF REQUESTED.                                       DSP01630
C•                                                                   DSP01640
      CALL DSPLY1                                                    DSP01650
C                                                                    DSP01660
C•••        CODE ADDED NOV. 13,1978 TO INCLUDE LIST PROCESSING       DSP01670
C                                                                    DSP01680
      IF (LISTSW.NE.0) GO TO 30                                      DSP01690
      IF(EMPTRS.NE.2.AND. PLTKEY.NE.1)GO TO 30                       DSP01700
C•                                                                   DSP01710
C••   EMTHRS COMPUTES AND PLOTS THE HISTOGRAM OF THE QUADRATIC FORM  DSP01720
C••   FOR THE CORRECTLY CLASSIFIED PIXELS WITHIN THE TRAINING OR TEST DSP01730
C••   FIELDS.                                                        DSP01740
C•                                                                   DSP01750
      IF ( DOTKEY .GT. 0 .OR. DOTERR .GT. 0 )  GO TO 30              DSP01760
      IF(PLOKEY.EQ.1)GO TO 10                                        DSP01770
      WRITE(6,100)                                                   DSP01780
      GO TO 30                                                       DSP01790
   10 CONTINUE                                                       DSP01800
      IF(TSTKEY.EQ.1)CALL EMTHRS(ARRAY(FLDSV3),ARRAY(FIELD3),        DSP01810
     *                           ARRAY(VERTX3),NOFLD3)               DSP01820
      IF(TSTKEY.NE.1)CALL EMTHRS(ARRAY(FLDSV2),ARRAY(FIELD2),        DSP01830
     *                           ARRAY(VERTX2),NOFLD2)               DSP01840
   30 CONTINUE                                                       DSP01850
C                                                                    DSP01860
C--   TEST THRSKY = 4  FOR FISHER F-DISTRIBUTION THRESHOLDS          DSP01870
C--   CALL FDIST TO COMPUTE AND STORE THRESHOLDS                     DSP01880
C                                                                    DSP01890
      IF(THRSKY.EQ.4) CALL FDIST                                     DSP01900
C--                                                                  DSP01910
C••                                                                  DSP01920
C••   DSPLY2 PRINTS THE MAP AND CALLS PCT TO BUILD PERFORMANCE TABLES. DSP01930
C•                                                                   DSP01940
      CALL DSPLY2(ARRAY(FLDSV2),ARRAY(FIELD2),ARRAY(VERTX2),         DSP01950
     *            ARRAY(FLDSV3),ARRAY(FIELD3),ARRAY(VERTX3),         DSP01960
     *            ARRAY(PCTID3),LISTSW,GTUNIT,GTFILE,                DSP01970
     *            AIUNIT,AIFILE,PPUNIT,PPFILE,NAMECT,ALP)            DSP01980
C                                                                    DSP01990
C••   IF DOT DATA PROCESSING WAS REQUESTED, THE PERFORMANCE          DSP02000
C••   TABLES WERE PERFORMED IN DSPLY2                                DSP02010
C•                                                                   DSP02020
C                                                                    DSP02030
C•••        CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING        DSP02040
C                                                                    DSP02050
      IF (LISTSW.NE.0) GO TO 99                                      DSP02060
      IF (DOTERR.EQ.1) GO TO 99                                      DSP02070
      IF (DOTKEY.GT.0) GO TO 99                                      DSP02080
C•                                                                   DSP02090
C••   PRTPCT PRINTS THE PERFORMANCE TABLES                           DSP02100
C•                                                                   DSP02110
      IF(TSTKEY.EQ.1)CALL PRTPCT(ARRAY(FLDSV3),ARRAY(PCTID3),NOFLD3) DSP02120
      IF(TSTKEY.NE.1)CALL PRTPCT(ARRAY(FLDSV2),ARRAY(PCTID3),NOFLD2) DSP02130
   99 WRITE(6,6000)                                                  DSP02140
 6000 FORMAT(1H1,5X,' ***** DISPLAY COMPLETED *****'///)             DSP02150
  100 FORMAT(/'  **DISPLAY** - FIELDS MUST BE DEFINED FOR SUBCLASSES FOR DSP02160
     * EMPIRICAL THRESHOLDS'/)                                       DSP02170
      RETURN                                                         DSP02180
       END                                                           DSP02190
```

FILE: CHI FORTRAN A

```
      REAL FUNCTION CHI(X,N,IFLAG)                                      CHI 00010
C-----------------------------------------------------------------------CHI 00020
C     TO COMPUTE THE VALUE OF THE CHI-SQUARED DISTRIBUTION WITH N-D.F   CHI 00030
C-----------------------------------------------------------------------CHI 00040
      IF(X.GT.0.0) GO TO 5                                              CHI 00050
      CHI=0.0                                                           CHI 00060
      RETURN                                                            CHI 00070
C-----------------------------------------------------------------------CHI 00080
C     CHECK TO SEE IF THE DEGREES OF FREEDOM IS EVEN                    CHI 00090
C-----------------------------------------------------------------------CHI 00100
    5 IF(MOD(N,2).EQ.0) GO TO 1                                         CHI 00110
C-----------------------------------------------------------------------CHI 00120
C     CALCULATION OF CHI FOR 1 DEGREE OF FREEDOM                        CHI 00130
C-----------------------------------------------------------------------CHI 00140
      G=SQRT(X)                                                         CHI 00150
      CHI=2.0*RNORM(G)-1.0                                              CHI 00160
      G=G/1.2533141                                                     CHI 00170
      IN=3                                                              CHI 00180
      GO TO 2                                                           CHI 00190
C-----------------------------------------------------------------------CHI 00200
C     CALCULATION OF CHI FOR 2 DEGREES OF FREEDOM                       CHI 00210
C-----------------------------------------------------------------------CHI 00220
    1 IN=4                                                              CHI 00230
      G=X/2.0                                                           CHI 00240
      IF(ABS(G).GT.88.027) GOTO 4                                       CHI 00250
      CHI=1.0-EXP(-G)                                                   CHI 00260
    2 IF(N.LT.3) RETURN                                                 CHI 00270
      IF(ABS(X/2.0).GT.88.027) GOTO 4                                   CHI 00280
      G=G*EXP(-X/2.0)                                                   CHI 00290
C-----------------------------------------------------------------------CHI 00300
C     CALCULATION OF CHI FOR N-GT-2 DEGREES OF FREEDOM                  CHI 00310
C-----------------------------------------------------------------------CHI 00320
      DO 3 I=IN,N,2                                                     CHI 00330
      CHI=CHI-G                                                         CHI 00340
      G=G*X/I                                                           CHI 00350
      CALL OVERFL(INOCT)                                                CHI 00360
      IF(INOCT.EQ.1)GOTO 4                                              CHI 00370
    3 CONTINUE                                                          CHI 00380
      RETURN                                                            CHI 00390
    4 IFLAG=1                                                           CHI 00400
      RETURN                                                            CHI 00410
      END                                                               CHI 00420
```

FILE: CHIN FORTRAN A

```
      FUNCTION CHIN(ALPHA,N,IFLAG)                                                  CHI00010
      DIMENSION H(7), D(15)                                                         CHI00020
      EQUIVALENCE (H(7), D(1))                                                      CHI00030
      DATA D/-.7080,-.4020,-.1980,-.0600,.0060,.0360,.0360,                         CHI00040
     1 .0120,-.0140,-.0360,-.0300,.0120,.1020,.2580,.4920/                          CHI00050
      IFLAG=0                                                                       CHI00060
      IF(N.EQ.1)  CHIN=(1.-NORM(1.-ALPHA/2.,IFLAG))                                 CHI00070
      IF(IFLAG.EQ.1) GOTO 10                                                        CHI00080
      CHIN=CHIN**2                                                                  CHI00090
      IF(N.EQ.2)  CHIN = -2. * ALOG(ALPHA)                                          CHI00100
      IF(N.LE.2)  RETURN                                                            CHI00110
      X=TINORM(1.-ALPHA,IFLAG)                                                      CHI00120
      IF(IFLAG.EQ.1) GOTO 10                                                        CHI00130
      I=2.*X                                                                        CHI00140
      IF(I.LT.-7) I=-7                                                              CHI00150
      IF(I.GT.6) I=6                                                                CHI00160
      Y=(H(I)*(2.*X-I)*(H(I+1)-H(I)))/N                                             CHI00170
      CHIN=2./(9.*N)                                                                CHI00180
      X=N*(1.-CHIN*(X-Y)*SQRT(CHIN))**3                                             CHI00190
      IF((N.LE.55.AND.X.LT.0.)                ) X=N*(1.-CHIN*X*SQRT(CHIN))**3CHI00200
      IF(N.LE.55) GO TO 1                                                           CHI00210
      IF(X.GT.176.16)  GOTO 10                                                      CHI00220
      CHIN=X                                                                        CHI00230
    4 RETURN                                                                        CHI00240
    1 IC=0                                                                          CHI00250
      IE=N-2                                                                        CHI00260
      IR=3                                                                          CHI00270
      G=2.5066283                                                                   CHI00280
      IF(MOD(N,2).EQ.1) GO TO 2                                                     CHI00290
      IR=2                                                                          CHI00300
      G=2.                                                                          CHI00310
    2 IF(IE.LE.1)GO TO 11                                                           CHI00320
      DO 3  I=IR,IE,2                                                               CHI00330
    3 G=G*I                                                                         CHI00340
   11 N2 = (N-2)/2                                                                  CHI00350
      N3 = N-2-N2                                                                   CHI00360
      SQX = SQRT(X)                                                                 CHI00370
      CHA = ((1.-CHI(X,N,IFLAG)-ALPHA)*G)/(SQX**N2)                                 CHI00380
      IF(IFLAG.EQ.1) GOTO 10                                                        CHI00390
      CHB = EXP(X/2.)/(SQX**N3)                                                     CHI00400
      CHIN = X + CHA*CHB                                                            CHI00410
      IF(ABS(X-CHIN)/AMAX1(X,CHIN).LT.5.E-06) GO TO 4                               CHI00420
      IF(IC.GT.200)  RETURN                                                         CHI00430
      IC=IC+1                                                                       CHI00440
      X=CHIN                                                                        CHI00450
      IF(X.GT.176.16)  GOTO 10                                                      CHI00460
      GO TO 11                                                                      CHI00470
   10 IFLAG=1                                                                       CHI00480
      RETURN                                                                        CHI00490
      END                                                                           CHI00500
```

FILE: DESIG FORTRAN A

```
      SUBROUTINE DESIG(LINE,IR,FLDSAV,FIELD ,VERTEX,NOFLD,
     *                 SAMSTR,SAMEND,SAMINC)
      IMPLICIT INTEGER(A-Z)
      DIMENSION IR(1),FIELD(5,NOFLD), FLDSAV(4,NOFLD),VERTEX(1),FL(22)
C*
C*    THIS ROUTINE SETS THE IR ARRAY FOR DESIGNATED FIELDS
C*
      DO 50 I=1,NOFLD
      IF(LINE .LT.FIELD(1,I))GO TO 50
      IF(LINE .GT.FIELD(2,I))GO TO 50
      IF(FIELD(3,I).GT.SAMEND)GO TO 50
      IF(FIELD(4,I).LT.SAMSTR)GO TO 50
C*
C*    FOUND A DESIGNATED FIELD ON THIS LINE
C*
      NV=FLDSAV(4,I)
      IPT=FIELD(5,I)
      ID = FLDSAV(2,I)
      CALL FDLINT(VERTEX(IPT),NV,FL,LINE,SAMPS,NI)
      DO 20 J=1,NI,2
      IB = (FL(J)-SAMSTR)/SAMINC+1
      IE = (FL(J+1)-SAMSTR)/SAMINC+1
      IF(MOD(SAMSTR,SAMINC) .NE. MOD(FL(J),SAMINC)) IB=IB+1
      IF(IB.GT.IE)GO TO 20
      DO 10 K=IB,IE
   10 IR(K)=ID
   20 CONTINUE
   50 CONTINUE
      RETURN
      END
```

FILE: DISTCV FORTRAN A

```
      SUBROUTINE DISTCV(DSFUNC,TOTPTS,RANGE)                                DIS00010
      IMPLICIT INTEGER(A-Z)                                                 DIS00020
CI*****************************************************************CIDIS00030
CI                                                                    DIS00040
CI                                                                    DIS00050
CI    DISTCV PLOTS THE DISTRIBUTION AND CHI SQUARE CURVES AND COMPUTES DIS00060
CI                  THE EMPIRICAL THRESHOLD VALUES                    DIS00070
CI                                                                    DIS00080
CI                                                                    DIS00090
CI*****************************************************************CIDIS00100
      REAL Q, DSFUNC,DISTVL,THRESH,REJECT,PCTREJ,THRES,CHISQ1         DIS00110
      REAL REJPCT,CHISQ,INC,CHIN                                      DIS00120
      DIMENSION DSFUNC(RANGE,60), TOTPTS(1), THRESH(60), MINM(60),    DIS00130
     *          SYMBLS(100), FIELDS(2),FIELD1(2), PCTREJ(60)          DIS00140
      DIMENSION CHISQ(100)                                            DIS00150
C     INCLUDE CMBK10,LIST                                             DIS00160
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60), DIS00170
     *             SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,         DIS00180
     *             PCFOKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,  DIS00190
     *             PLTKEY,AMFLG,AMCOMB,AMFEAT,CDATE(2),               DIS00200
     *             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,  DIS00210
     *             THRES(60),SYMMTX(66),HIGH(60),CON(60)              DIS00220
     *             ,FLOKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)            DIS00230
     *             ,NOSUB2,NOTHRO,TOTVT2,NOCLS2                       DIS00240
     *             ,KATNO(60),NOCAT,FILTER,MAPFMT                     DIS00250
     *             ,DESKEY,DESUNI,DESOTH,CROP  ,ACROP,AOTHER,ATOTAL   DIS00260
     *             ,SITE(6),ANALYS(5),CAM(15),CRPKEY,KEPPTS(60)       DIS00270
     *             ,DOTKEY,DOTERR                                     DIS00280
CSEND                                                                 DIS00290
      DATA FIELD /'TEST'/, FIELDS/'TRAI','NING'/                      DIS00300
      DATA BLANK/'    '/,MXQUAD/20/                                   DIS00310
      DATA ASTK /'*'/,DOLLAR /'$'/                                    DIS00320
      EQUIVALENCE ( HIGH(1),PCTREJ(1) )                               DIS00330
      DO 5    I=1,2                                                   DIS00340
    5 FIELD1(I) = BLANK                                               DIS00350
      DO 10   L=1,99                                                  DIS00360
      REJPCT = 1. -  FLOAT(L) / 100.0                                 DIS00370
      CHISQ1 = CHIN(REJPCT,NOFET2,FLAG)                               DIS00380
      IF(FLAG.EQ.1) GOTO 13                                           DIS00390
      CHISQ(L) = CHISQ1       + 0.05                                  DIS00400
      GO TO 10                                                        DIS00410
   13 WRITE(6,140) REJPCT,CHISQ1                                      DIS00420
  140 FORMAT(' OVERFLOW',2X,F5.2,3X,F15.5)                            DIS00430
   10 CONTINUE                                                        DIS00440
      CALL SETMRG(66,0,66)                                            DIS00450
      DO 200 J=1,NOSUB2                                               DIS00460
      TOTPTS(J)=0                                                     DIS00470
      DO 20   M=1,RANGE                                               DIS00480
      TOTPTS(J) = DSFUNC(M,J) + TOTPTS(J)                             DIS00490
   20 CONTINUE                                                        DIS00500
      IF (TOTPTS(J) .EQ. 0)  GO TO 200                                DIS00510
      DO 40   M=1,RANGE                                               DIS00520
      MM = M - 1                                                      DIS00530
      IF (MM .EQ. 0)  DSFUNC(MM+1,J) =(DSFUNC(MM+1,J) / TOTPTS(J)) * 100DIS00540
      IF (MM .EQ. 0)  GO TO 40                                        DIS00550
      DSFUNC(M,J) = DSFUNC(MM,J) +(DSFUNC(M,J) / TOTPTS(J)) * 100     DIS00560
   40 CONTINUE                                                        DIS00570
C                                                                     DIS00580
CC    FIND MINIMUM M FOR WHICH DSFUNC(M,J) .GT. 1 - 100*PCTREJ        DIS00590
C                                                                     DIS00600
      DISTVL =(1. -  PCTREJ(J))* 100                                  DIS00610
      MXREJT = DISTVL + 0.5                                           DIS00620
      DO 50   M=1,RANGE                                               DIS00630
      IF (DISTVL .LE. DSFUNC(M,J) )  MINM(J) = M                      DIS00640
      IF (DISTVL .LE. DSFUNC(M,J))  GO TO 55                          DIS00650
   50 CONTINUE                                                        DIS00660
   55 CONTINUE                                                        DIS00670
C                                                                     DIS00680
CC    THRESHOLD                                                       DIS00690
C                                                                     DIS00700
      REJECT = PCTREJ(J) * 100                                        DIS00710
      IF (TSTKEY .EQ. 1)  FIELD1(1) = FIELD                           DIS00720
      IF (TSTKEY .EQ. 0)  FIELD1(1) = FIELDS(1)                       DIS00730
      IF (TSTKEY .EQ. 0)  FIELD1(2) = FIELDS(2)                       DIS00740
      THRESH(J) =(0.1 * MINM(J)) * 2                                  DIS00750
C                                                                     DIS00760
CC     PRINT HEADING                                                  DIS00770
C                                                                     DIS00780
      CLASNO=J                                                        DIS00790
```

FILE: DISTCV FORTRAN A

```
      WRITE(6,100)SURNAM(J)                                                  DIS00800
  100 FORMAT(1H1,////T52,'DISTRIBTION CURVE FOR SUBCLASS ',A4)               DIS00810
      WRITE(6,105) THRES(CLASNO),REJECT,(FIELD1(I),I=1,2),THRESH(J)          DIS00820
  105 FORMAT( /// T10, 'CHI SQUARE THRESHOLD = ',F5.2, T53, 'EMPIRICAL TDIS00830
     *HRESHOLD FROM', T91, 'USER REJECTION PERCENTAGE = ', F4.1/ T53,     DIS00840
     *  2A4, ' FIELDS = ',F5.2)                                           DIS00850
      WRITE(6,110)                                                           DIS00860
  110 FORMAT(///T15, 10(1H0,9X),1H1,/ T15,'0',9X,'1',9X,'2',9X,'3',9X,      DIS00870
     *       '4',9X,'5',9X,'6',9X,'7',9X,'8',9X,'9',9X,'0'/                 DIS00880
     *  T15,11(1H0,9X)/                                                    DIS00890
     *       T11,'0.0',1X,101(1H+))                                         DIS00900
      INC = FLOAT(MXQUAD)/ FLOAT(RANGE)                                      DIS00910
      II = 1                                                                 DIS00920
      N = 0                                                                  DIS00930
      DO 90  L=1,RANGE                                                       DIS00940
      Q = FLOAT(MXQUAD*L) / RANGE                                            DIS00950
      N = N + 1                                                              DIS00960
      DO 60   M= 1,100                                                       DIS00970
   60 SYMBLS(M) = BLANK                                                      DIS00980
                                                                             DIS00990
C                                                                            DIS01000
CCC     CHI SQUARE CURVE                .                                    DIS01010
C                                                                            DIS01020
   65 IF ( II .GT. MXPEJT )   GO TO 75                                       DIS01030
      IF ( CHISQ(II) .GE. (Q+INC) ) GO TO 75                                 DIS01040
      SYMBLS(II) = ASTK                                                      DIS01050
      II = II + 1                                                            DIS01060
      GO TO 65                                                               DIS01070
   75 CONTINUE                                                               DIS01080
C                                                                            DIS01090
CCC     DISTRIBUTION CURVE                                                   DIS01100
C                                                                            DIS01110
      PERCNT = DSFUNC(L,J)                                                   DIS01120
      IF (PERCNT .EQ. 0)  GO TO 77                                           DIS01130
      IF (SYMBLS(PERCNT) .NE. BLANK)  SYMBLS(PERCNT) = DOLLAR                DIS01140
      IF (SYMBLS(PERCNT) .NE. BLANK)   GO TO 77                              DIS01150
   76 SYMB=J                                                                 DIS01160
      SYMBLS(PERCNT) = SYMMTX(SYMB)                                          DIS01170
   77 CONTINUE                                                               DIS01180
      IF (N .EQ. 5)  GO TO 80                                                DIS01190
      WRITE(6,120) (SYMBLS(K),K=1,100)                                       DIS01200
  120 FORMAT(T15,'+',100A1)                                                  DIS01210
      GO TO 90                                                               DIS01220
   80 WRITE(6,130) Q, (SYMBLS(K),K=1,100)                                    DIS01230
  130 FORMAT(T10,F4.1,1X,'+',100A1)                                          DIS01240
      N = 0                                                                  DIS01250
   90 CONTINUE                                                               DIS01260
      WRITE(6,150) SYMMTX(SYMB)                                              DIS01270
  150 FORMAT(///T10, 'NOTE :  ',A1,' - CLASS DISTRIBUTION CURVE'//T18,      DIS01280
     *'* - CHI SQUARE DISTRIBUTION CURVE'//T18,'$ - INTERSECTION OF CURVDIS01290
     *ES')                                                                   DIS01300
      IF (EMPTRS .EQ. 0)  GO TO 200                                          DIS01310
      THRES(J)=THRESH(J)                                                     DIS01320
  200 CONTINUE                                                               DIS01330
      CALL SETMRG(66,4,62)                                                   DIS01340
      RETURN                                                                 DIS01350
      END
```

FILE: DSPLY1 FORTRAN A

```
      SUBROUTINE DSPLY1                                                      DSP00010
C                                                                            DSP00020
      IMPLICIT INTEGER (A-H,O-Z)                                             DSP00030
CI-----------------------------------------------------------------------IDSP00040
CI-----------------------------------------------------------------------IDSP00050
CCI                                                                        IDSP00060
CCI                                                                        IDSP00070
CCI    PURPOSE.. READS THE STATISTICS FROM 'MAPTAP'                        IDSP00080
CCI                                                                        DSP00090
CCI-----------------------------------------------------------------------IDSP00100
CCI-----------------------------------------------------------------------IDSP00110
CCCC                                                                       DSP00120
CCCC                                                                       DSP00130
CCCC                                                                       DSP00140
CCCC                                                                       DSP00150
C     INCLUDE COMBK6.LIST                                                  DSP00160
      REAL COV,DET(60)                                                     DSP00170
C     INCLUDE COMBK10.LIST                                                 DSP00180
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,            DSP00190
     *              HISFIL,HISKEY,IPFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,      DSP00200
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                   DSP00210
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                       DSP00220
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                 DSP00230
     *  CRDUNT,PRTUNT,RANDIO                                               DSP00240
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60),      DSP00250
     *             SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,            DSP00260
     *             PCFOKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,     DSP00270
     *             PLTKFY,BMFLG,BMCOMB,BMFEAT,CDATE(2),                  DSP00280
     *             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,     DSP00290
     *             THRES(60),SYMMTX(66),HIGH(60),CON(60)                 DSP00300
     *             ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)               DSP00310
      DATA  BCDFOR/'4'/, BCDTWO/'2'/, DASH/'----'/                          DSP00400
C                                                                          DSP00410
CCCC  ---------------------------------------------------------------------DSP00420
CCCC                                                                       DSP00430
CCCC  RETRIEVE AND PRINT THE COVARIANCE AND MEAN                           DSP00440
CCCC  ---------------------------------------------------------------------DSP00450
C                                                                          DSP00460
  200 IF( BMFLG .LE. 0)  GO TO 202                                         DSP00470
C                                                                          DSP00480
      NOFET2 = BMCOMB                                                      DSP00490
  202 CV=1                                                                 DSP00500
      VARSZ2=NOFET2*(NOFET2+1)/2                                           DSP00510
      MN=CV + VARSZ2*NOSUB2                                                DSP00520
      IF(MN + NOSUB2*NOFET2 .GT. SIZE)GO TO 180                            DSP00530
      GO TO 190                                                            DSP00540
  170 CONTINUE                                                             DSP00550
C*    CALL DSPLIA(STORAGE(CV),STORAGE(MN),VARSIZ2,NOFET2,NOSUB2)           DSP00560
      RETURN                                                               DSP00570
  180 WRITTE(6,181)                                                        DSP00580
  181 FORMAT(' NOT ENOUGH STORAGE FOR COVARIANCE MATRICES - DSPLY1')       DSP00590
      CALL CMERR                                                           DSP00600
      CONTINUE                                                             DSP00610
      RETURN                                                               DSP00620
C                                                                          DSP00630
CCCC  END  ---  SUBROUTINE  DSPLY1                                         DSP00640
CCCC  *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-DSP00650
CCCC                                                                       DSP00660
CCCC                                                                       DSP00670
CCCC  INTERNAL SUBROUTINE  DSPLIA                                          DSP00680
CCCC                                                                       DSP00690
C*    SUBROUTINE DSPLIA(COVMTX,AVEMTTX,VARSZ2,NOFET2,NOCLS2)               DSP00700
C*    COVMTX(I,1)=STORAGE(I)                                               DSP00710
C*    AVEMTX(I,J)=STORAGE(MN-1+I+(J-1)*NOFET2                              DSP00720
  190 CONTINUE                                                             DSP00730
C                                                                          DSP00740
CCCC                                                                       DSP00750
CCCC  READ ORIGINAL COVARIANCE AND MEANS MATRIX FOR EACH CLASS             DSP00760
CCCC  ( B-TRANSFORMED IF B-MATRIX WAS APPLIED IN $CLASSIFY -               DSP00770
CCCC  BMFLG .GT. 0 IF  SO )                                                DSP00780
C                                                                          DSP00790
```

FILE: DSPLY1 FORTRAN A

```
      DUMMY=NOCLS2                                                          DSP00800
      NOCLS2=NOSUB2                                                         DSP00810
      NS = VARSZ2 * NOCLS2                                                  DSP00820
      NSS = NOFET2 * NOCLS2                                                 DSP00830
      READ(MAPTAP)(STORAG(CV+I-1),I=1,NS),(STORAG(MN+I-1),I=1,NSS)          DSP00840
C                                                                           DSP00850
      IF (STATKY.EQ.0) GO TO 290                                            DSP00860
C                                                                           DSP00870
      CNT = 7+(5+3+2*NOFET2)*((NOFET2+11)/12)                               DSP00880
      CNT = PAGSIZ/CNT                                                      DSP00890
      INC = CNT                                                             DSP00900
C                                                                           DSP00910
      DO 280 ICLAS=1,NOCLS2                                                 DSP00920
      IF (INC.LT.CNT) GO TO 210                                             DSP00930
      WRITE (6,HEAD)                                                        DSP00940
      INC = 0                                                               DSP00950
  210 WRITE(6,220) SUBNAM(ICLAS),SYMMTX(ICLAS),(DASH ,I=1,5)                DSP00960
  220 FORMAT(//' SUBCLASS',A4,' REPRESENTED BY SYMBOL -    ',A1 /           DSP00970
     *  1X,3A4,A1,T44,A3/)                                                 DSP00980
      DO 230 LOC=1,NOFET2,12                                                DSP00990
      STOP = LOC+11                                                         DSP01000
      IF( STOP .GT. NOFET2 ) STOP = NOFET2                                  DSP01010
      NS=MN-1+(ICLAS-1)*NOFET2                                              DSP01020
      WRITE(6,260)(STORAG(NS+I),I=LOC,STOP)                                 DSP01030
  230 CONTINUE                                                              DSP01040
  260 FORMAT('0 MEAN:',3X,12F9.2)                                           DSP01050
      WRITE(6,2601)    DASH                                                 DSP01060
 2601 FORMAT(1X,A4/)                                                        DSP01070
C                                                                           DSP01080
      IF( PMFLG .GT. 0)  GO TO 271                                          DSP01090
      WRITE(6,270) (DASH,I=1,5)                                             DSP01100
      GO TO 272                                                             DSP01110
  270 FORMAT('0COVARIANCE MATRIX:' / 1X,5A4)                                DSP01120
  271 WRITE(6,2711)   ( DASH, I=1,9)                                        DSP01130
 2711 FORMAT( '0COVARIANCE MATRIX (B-TRANSFORMED) :'/1X,8A4,A3)             DSP01140
  272 NS=1+(ICLAS-1)*VARSZ2                                                 DSP01150
      CALL WPTMTX(STORAG(NS),NOFET2,BCDTWO)                                 DSP01160
      INC = INC+1                                                           DSP01170
  290 CONTINUE                                                              DSP01180
C                                                                           DSP01190
CCC   READ COVARIANCE MATRIX ( AFTER CHOLESKY FACTORIZATION),               DSP01200
CCC   PROBABILITY DENSITY FUNCTION CONSTANTS, CON , AND COVARIANCE          DSP01210
CCC   MATRIX DETERMINANT, DET , FOR EACH CLASS                              DSP01220
CCC                                                                         DSP01230
CCC   ---------------------------------------------------------------------DSP01240
C                                                                           DSP01250
  290 CONTINUE                                                              DSP01260
      NS = VARSZ2 * NOCLS2                                                  DSP01270
      READ(MAPTAP)(STORAG(CV+I-1),I=1,NS),(CON(I),I=1,NOCLS2),(DET(I),      DSP01280
     *I=1,NOCLS2)                                                           DSP01290
      IF (STATKY.EQ.0) GO TO 330                                            DSP01300
C                                                                           DSP01310
      CNT = 11+(3+2*NOFET2)*((NOFET2+11)/12)                                DSP01320
      CNT = PAGSIZ/CNT                                                      DSP01330
      INC = CNT                                                             DSP01340
      DO 310 I=1,NOCLS2                                                     DSP01350
      IF (INC.LT.CNT) GO TO 300                                             DSP01360
      WRITE (6,HEAD)                                                        DSP01370
      INC =0                                                                DSP01380
  300 WRITE(6,320)SUBNAM(I),SYMMTX(I),DET(I),CON(I)                         DSP01390
      NS = 1+(I-1)*VARSZ2                                                   DSP01400
      CALL WPTMTX(STORAG(NS),NOFET2,BCDFOR)                                 DSP01410
      INC = INC+1                                                           DSP01420
  310 CONTINUE                                                              DSP01430
  320 FORMAT(//1X,'MULTSPECTRAL CHARACTERISTICS FOR SUBCLASS ',A4,'         DSP01440
     1REPRESENTED BY SYMBOL ',A1/1H0,'DETERMINANT = ',F18.9/1H0,'CONSTANT   DSP01450
     2T TERM = ', F10.4//1H0,'COVARIANCE MATRIX (CHOLESKY DECOMPOSITION)    DSP01460
     1 :' )                                                                 DSP01470
C                                                                           DSP01480
CCC   GO HOME                                                               DSP01490
CCC   ---------------------------------------------------------------------DSP01500
C                                                                           DSP01510
  330 CONTINUE                                                              DSP01520
      NOCLS2=DUMMY                                                          DSP01530
      GO TO 170                                                             DSP01540
       END                                                                  DSP01550
```

FILE: DSPLY2 FORTRAN A

```
      SUBROUTINE DSPLY2(TRNSAV,TRNFLD,TRNVER,TSTSAV,TSTFLD,                 DSP00010
     *TSTVER,PCTAB,LISTSW,GTUNIT,GTFILE,AIUNIT,AIFILE,                      DSP00020
     *PPUNIT,PPFILE,NAMECT,ALP)                                             DSP00030
            IMPLICIT INTEGER(A-Z)                                           DSP00040
C                                                                           DSP00050
C***  CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING                     DSP00060
C                                                                           DSP00070
            REAL ALP(2)                                                     DSP00080
CI-----------------------------------------------------------------------IDSP00090
CI-----------------------------------------------------------------------IDSP00100
CI                                                                        IDSP00110
CI     DSPLY2 CAN  PERFORM SPATIAL FILTERING ON THE CLASSIFIED DATA.      DSP00120
CI        (I.E. - THE FOUR NEAREST NEIGHBORS OF EACH PIXEL ARE            DSP00130
CI                TESTED FOR 'SAMENESS'. IF THE FOUR NEIGHBORS ARE        DSP00140
CI                CLASSIFIED AS ONE TYPE AND THE PIXEL IN QUESTION        DSP00150
CI                WAS CLASSIFIED AS A DIFFERENT TYPE, THE CLASSIFICATION  DSP00160
CI                IS CHANGED TO THE SAME AS THE NEIGHBORS.)               DSP00170
CI                EXAMPLE:                                                DSP00180
CI                   LINE N    -     C     -                            DSP00190
CI                        N+1  -    CXC    -  X IS CHANGED TO C          DSP00200
CI                        N+2  -     C     -                            DSP00210
CI                                                                        DSP00220
CI     DSPLY2 ALSO PERFORMS THRESHOLDING, PRINTS THE CLASSIFICATION       DSP00230
CI     MAP, AND CALLS THE APPROPIATE ROUTINES TO BUILD AND THEN PRINT     DSP00240
CI     THE CLASSIFICATION PERFORMANCE TABLES.                             DSP00250
C*                                                                        DSP00260
C*     THE DOT DATA PERFORMANCE SUMMARIES ARE PROVIDED FROM THE INTERNAL  DSP00270
C*     SUBROUTINE,DOTSUM                                                  DSP00280
C*                                                                        DSP00290
C*     LIST PROCESSING ADDED NOV 13, 1978 LISTSW IS THE                   DSP00300
C*     NAME OF THE SWITCH                                                 DSP00310
CI                                                                        DSP00320
CI                                                                      IDSP00330
CI-----------------------------------------------------------------------IDSP00340
CI-----------------------------------------------------------------------IDSP00350
CC                                                                        DSP00360
C                                                                         DSP00370
      REAL CON                                                            DSP00380
      DIMENSION TRNSAV(4,NOFLD2),TRNFLD(5,NOFLD2),TRNVER(2,TOTVT2)        DSP00390
     *         ,TSTSAV(4,NOFLD3),TSTFLD(5,NOFLD3),TSTVER(2,TOTVT3)        DSP00400
     *    ,COL(3,110),SCRAT(330),FLDINF(6)                                DSP00410
     *         ,IR(1000,3),VR(1000),OUT(1000),ILINE(3),BUF(110,20)        DSP00420
     *         ,JSTAT(20),PCTAB(NOTRFD,NOSUB3)                            DSP00430
     2, CLTOT(61) , DOTLBL(61,2),PUNC(61)                                 DSP00440
     3 , LABCLS(60,61), LBLTOT(61), TOTLBL(60), PCORR(61) ,               DSP00450
     4  ALPHA(60,60), TOTTHR(60),DOTORD(61,2)                            DSP00460
C     INCLUDE CMRK10,LIST                                                 DSP00470
C     INCLUDE COMRK6                                                      DSP00480
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,           DSP00490
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,     DSP00500
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  DSP00510
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                      DSP00520
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                DSP00530
     *  ,CRDUNT,PRTUNT,RANDIO                                             DSP00540
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60),     DSP00550
     *             SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,             DSP00560
     *             PCFOKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,      DSP00570
     *             PLTKEY,BMFLG,BMCOMP,BMFEAT,CDATE(2),                   DSP00580
     *             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,      DSP00590
     *             THRES(60),SYMMTX(66),HIGH(60),CON(60)                  DSP00600
     *             ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)                DSP00610
     *             ,NOSUB2,NOTRFD,TOTVT2,NOCLS2                           DSP00620
     *             ,KATNO(60),NOCAT,FILTER,MAPFMT                         DSP00630
     *             ,DESKEY,DESUNI,DESOTH,CROP  ,ACROP,AOTHER,ATOTAL       DSP00640
     *             ,SITE(6),ANALYS(5),CAM(15),CRPKEY,KEPPTS(60)           DSP00650
     *             ,DOTKEY,DOTERR                                         DSP00660
C$END                                                                     DSP00670
      LOGICAL START,FULL                                                  DSP00680
      DATA AST/'****'/                                                    DSP00690
      DATA THRESH/'THRE'/,BLANK/'    '/                                   DSP00700
      EQUIVALENCE      (FLDINF(1), LINSTR), (FLDINF(2),LINEND),           DSP00710
     1                 (FLDINF(3),LININC), (FLDINF(4), SAMSTR),           DSP00720
     2                 (FLDINF(5),SAMEND),(FLDINF(6),SAMINC),             DSP00730
     3                 (COL,SCRAT),(IR,BUF)                               DSP00740
C-----------------------------------------------------------------------   DSP00750
C     FOLLOWING EQUIVALENCED ARRAYS WILL ONLY BE ACCESSED IN INTERNAL     DSP00760
C        SUBROUTINE DOTSUM                                                DSP00770
      EQUIVALENCE (CLTOT(1),IR(1,1)),(PUNC(1),IR(62,1)),                  DSP00780
     *(DOTLBL(1,1),IR(124,1)),                                            DSP00790
```

FILE: DSPLY2 FORTRAN A

```
     *   ( PCORR(1) , IR(246,1) ) , ( TOTLBL(1) , IR(307,1) ) ,                DSP00800
     *    ( TOTHR(1) , IR(367,1) ) ,  ( LBLTOT(1) , IR(427,1) )  ,             DSP00810
     *    ( DOTORD(1,1) , IR(488,1) )                                          DSP00820
      REAL ALPHA,PCORR,PUNC                                                   DSP00830
       DIMENSION FMT1(19),FMT2(24),FMT3(8),FMT4(8),FMT5(9),TEMP(5,3)          DSP00840
C     DATA FMT1/'(TA','     ','4   ','(8HC','ATEG','ORY.','25X)','/T2,',     DSP00850
CC    *    '   ','4   ','(26H','----','----','----','----','----','----',    DSP00860
C     *    '--,7','X)) '/                                                     DSP00870
      DATA FMT2/'(T2,','     ','4   ','(26H','LABE','LED ',' CLA','SSIF',     DSP00880
     *   'IED(','SUBC','L),7','X)/T','2,  ','4   ','(7H-','----','--,2',     DSP00890
     *   'X,17','H---','----','----','----','----','--,7','X)) '/             DSP00900
      DATA FMT3/'(T90','.   ','3   ','(1H(','.F7.','3.1H','),4X',')) '/     DSP00910
      DATA FMT4/'(/T2',',4.  ','8   ','(1H(','.F7.','3.1H','),4X',')) '/    DSP00920
C     DATA FMT5/'(//T','8,  ','4   ','(1H(','I3,','1H,','I3.1','H),2',     DSP00930
CC    *   '4X)) '/                                                           DSP00940
      DIMENSION BLANKS(4)                                                     DSP00950
      DATA BLANKS/'    ','    ','    ','    '/                               DSP00960
       REAL PCC,PCTOT,PIXTOT                                                  DSP00970
C-----------------------------------------------------------------------     DSP00980
C                                                                             DSP00990
CCC                                                                           DSP01000
CCCC  ---------------------------------------------------------------------   DSP01010
CCC                                                                           DSP01020
C                                                                             DSP01030
      DIMENSION FORMAT(3,2)                                                   DSP01040
      DATA FORMAT/'UNIV','ERSA','L   ','LARS','YS I','I   '/                  DSP01050
      REAL     TOTALS(66),VR,THRES                                            DSP01060
      DIMENSION TTOL(66)                                                      DSP01070
C                                                                             DSP01080
CCC   ---------------------------------------------------------------------   DSP01090
CCC   ---------------------------------------------------------------------   DSP01100
C                                                                             DSP01110
      TRNNO=NOSUB3+1                                                          DSP01120
      TSTNO=NOSUB3+2                                                          DSP01130
C*                                                                            DSP01140
C*    SET POINTERS FOR SYMBOLS ARRAY AND TOTALS ARRAY                         DSP01150
C*                                                                            DSP01160
      DUPNO = NOSUB3+3                                                        DSP01170
      DESUNI= NOSUB3+4                                                        DSP01180
      DESOTH= NOSUB3+5                                                        DSP01190
C*                                                                            DSP01200
C*    FLAG USED IN DOTPCT TO INITIALIZE PCTAB=0                               DSP01210
C*                                                                            DSP01220
      PCTKEY=0                                                                DSP01230
C                                                                             DSP01240
C***  CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING                       DSP01250
C                                                                             DSP01260
            FLDCNT = 0                                                        DSP01270
            IF (LISTSW.EQ.0) GO TO 17                                         DSP01280
            DO 16 I = 1,209                                                   DSP01290
            LR = (I - 1)/19                                                   DSP01300
            LR = (LR + 1)*10                                                  DSP01310
            LS = (LR - 1)/10                                                  DSP01320
            LS = 10*(I - (LS*19))                                             DSP01330
            TRNVER(1,I) = LS                                                  DSP01340
            TRNVER(2,I) = LR                                                  DSP01350
 16         CONTINUE                                                          DSP01360
            GO TO 20                                                          DSP01370
 17         CONTINUE                                                          DSP01380
      IF ( DOTKEY .GT. 0  .OR.  DOTERR .GT. 0 )  GO TO 11                     DSP01390
      DO 10 I=1,NOSUB3                                                        DSP01400
      DO 10 J=1,NOTRFD                                                        DSP01410
   10 PCTAB(J,I) = 0                                                          DSP01420
C*                                                                            DSP01430
      GO TO 20                                                                DSP01440
C*                                                                            DSP01450
   11 IF( DOTERR .GT. 0 )  GO TO 20                                           DSP01460
C*                                                                            DSP01470
C*    INITIALIZE  TRNSAV FOR OUTLINING DOTS IN SUBR. FLDBOR                   DSP01480
C*                                                                            DSP01490
      DO  12  I=1,NOFLD2                                                      DSP01500
   12 TRNSAV(4,I) = 2                                                         DSP01510
C*                                                                            DSP01520
CCC                                                                           DSP01530
CCCC                                                                          DSP01540
CCCC  PRINT OUT HEADING                                                       DSP01550
CCC   ---------------------------------------------------------------------   DSP01560
CCC                                                                           DSP01570
                                                                              DSP01580
```

FILE: DSPLY2 FORTRAN A

```
   20 READ(MAPTAP)FLDINF,PTS,LINES,FLDESC                                  DSP01590
      IF (PTS .GT. 1000) WRITE(6,22)                                       DSP01600
      IF (PTS .GT. 1000) STOP                                              DSP01610
   22 FORMAT(' DISPLAY WILL ACCEPT ONLY 1000 PTS/SCAN LINE')               DSP01620
      ISTRT=SAMSTR                                                         DSP01630
      IEND=SAMEND                                                          DSP01640
      IF (PTS.EQ.0) GO TO 310                                              DSP01650
      HDREC = 1                                                            DSP01660
      DO  25  I=1,DESOTH                                                   DSP01670
      TTOL(I)=0                                                            DSP01680
   25 TOTALS(I)=0.0                                                        DSP01690
C                                                                          DSP01700
C     PRINT OUT THE COLUMN NUMBERS                                         DSP01710
C     ----------------------------------------------------------------------DSP01720
C                                                                          DSP01730
   30 J = 0                                                                DSP01740
      CALL SETMRG(6R,0,68)                                                 DSP01750
      IF(NOMAP.EQ.0)GO TO 85                                               DSP01760
       WRTFG=1                                                             DSP01770
       GO TO 370                                                           DSP01780
 31    PRTFG=1                                                             DSP01790
       GO TO 510                                                          DSP01800
   85 CONTINUE                                                             DSP01810
      SPKNT=0                                                              DSP01820
      COUNTR=0                                                             DSP01830
      DRUMLN=0                                                             DSP01840
      LAST=0                                                               DSP01850
      START=.FALSE.                                                        DSP01860
      FULL =.FALSE.                                                        DSP01870
      I1=1                                                                 DSP01880
      I2=2                                                                 DSP01890
      I3=3                                                                 DSP01900
      J=1                                                                  DSP01910
      ADRES=DRUMAD                                                         DSP01920
   91 READ(MAPTAP) ILINE(J),(IR(I,J),I=1,PTS),(VR(I),I=1,PTS)              DSP01930
      IF(ILINE(J).EQ.0)GO TO 105                                           DSP01940
C*                                                                         DSP01950
C*    IF DESIGNATED FIELDS HAVE BEEN INPUT, SET THE IR ARRAY FOR PIXELS    DSP01960
C*    IN THOSE FIELDS.                                                     DSP01970
      IF(DESKEY.EQ.1)CALL DESIG(ILINE(J),IR(1,J),TSTSAV,TSTFLD,TSTVER,     DSP01980
     *                  NOFLD3,SAMSTR,SAMEND,SAMINC)                       DSP01990
      DO 100 I=1,PTS                                                       DSP02000
      L = IR(I,J)                                                          DSP02010
      IF(L.EQ.0)GO TO 100                                                  DSP02020
      IF(L.EQ.NOSUB3)GO TO 92                                              DSP02030
      IF(L.GT.NOSUB3)GO TO 95                                              DSP02040
C*                                                                         DSP02050
C*    THRESHOLDING                                                         DSP02060
C*                                                                         DSP02070
C*    THE VALUE OF THE QUADRATIC FORM MUST BE EXTRACTED FROM VR FOR THRE   DSP02080
C*                                                                         DSP02090
C*           Q = -2*VR - CON                                               DSP02100
C*                                                                         DSP02110
C*                                                                         DSP02120
      IF(THRSKY.EQ.0)GO TO 95                                              DSP02130
      IF((-2.*VR(I)-CON(L)).LT. THRES(L))GO TO 95                          DSP02140
   92 CONTINUE                                                             DSP02150
C*                                                                         DSP02160
C*    TTOL = TOTAL PIXELS THRESHOLDED, BY SUBCLASS                         DSP02170
C*                                                                         DSP02180
      TTOL(L) = TTOL(L) + 1                                                DSP02190
      IR(I,J) = NOSUB3                                                     DSP02200
      L = NOSUB3                                                           DSP02210
C*                                                                         DSP02220
C*   TOTALS = TOTAL NO. PIXELS CLASSIFIED INTO EACH SUBCLASS,              DSP02230
C*    INCLUDING THRESHOLDED AND DESIGNATED OTHER , DESIGNATED UNIDENT      DSP02240
C*                                                                         DSP02250
   95 TOTALS(L) = TOTALS(L) + 1                                            DSP02260
  100 CONTINUE                                                             DSP02270
C*                                                                         DSP02280
C*    HAVE 3 LINES BEEN READ                                               DSP02290
C*                                                                         DSP02300
      IF(START)GO TO 105                                                   DSP02310
      J=J+1                                                                DSP02320
      IF(J.LT.3)GO TO 91                                                   DSP02330
      START=.TRUE.                                                         DSP02340
      GO TO 91                                                             DSP02350
C*                                                                         DSP02360
C*    SPATIAL FILTERING                                                    DSP02370
```

12-13

2/2

FILE: DSPLY2 FORTRAN A

```
C*                                                                      DSP02380
  105 IF(FILTER.EQ.0)GO TO 115                                          DSP02390
      I=2                                                               DSP02400
  106 IF((IR(I-1,I2) .NE. IR(I+1,I2)) GO TO 110                         DSP02410
      IF((IR(I,I1)   .NE. IR(I,I3))   GO TO 110                         DSP02420
      IF((IR(I,I1)   .NE. IR(I-1,I2)) GO TO 110                         DSP02430
      IF(IR(I,I2) .EQ. IR(I,I1))GO TO 110                               DSP02440
      IF(IR(I,I1) .EQ. NOSUB3)GO TO 110                                 DSP02450
      ICC=IR(I,I1)                                                      DSP02460
      ICK=IR(I,I2)                                                      DSP02470
      IF(ICK.EQ.NOSUB3)GO TO 110                                        DSP02480
      TOTALS(ICC)=TOTALS(ICC)+1                                         DSP02490
      TOTALS(ICK)=TOTALS(ICK)-1                                         DSP02500
      SPKNT=SPKNT+1                                                     DSP02510
      IR(I,I2) = IR(I,I1)                                               DSP02520
  110 I=I+1                                                             DSP02530
      IF(I.LE.PTS-1)GO TO 106                                           DSP02540
C*                                                                      DSP02550
CC*   GET PERFORMANCE FOR LINE I1                                       DSP02560
CC*                                                                     DSP02570
C                                                                       DSP02580
CC*** CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING                 DSP02590
C                                                                       DSP02600
  115         IF (LISTSW.NE.0) GO TO 116                                DSP02610
      IF (DOTERR.EQ.1) GO TO 114                                        DSP02620
      IF (DOTKEY.EQ.0) GO TO 40                                         DSP02630
  116         CONTINUE                                                  DSP02640
C*                                                                      DSP02650
CC*   TEST TO SEE IF THE CURRENT LINE CONTAINS ANY DOTS                 DSP02660
C*                                                                      DSP02670
      IF (ILINE(I)).GT.TRNVER(2,NOFLD2)) GO TO 114                      DSP02680
      ACNT = 0                                                          DSP02690
      ECNT = 0                                                          DSP02700
      DO 41 I=1,NOFLD2                                                  DSP02710
      IF (ILINE(I)).NE.TRNVER(2,I)) GO TO 42                            DSP02720
      IF (ACNT.EQ.0) BCNT = I                                           DSP02730
      GO TO 41                                                          DSP02740
   42 IF (ACNT.EQ.0) GO TO 41                                           DSP02750
      ECNT = I-1                                                        DSP02760
      GO TO 43                                                          DSP02770
   41 CONTINUE                                                          DSP02780
      IF (ILINE(I)).NE.TRNVER(2,NOFLD2)) GO TO 114                      DSP02790
      ECNT = NOFLD2                                                     DSP02800
C*                                                                      DSP02810
CC*   FOR DOT DATA PROCESSING, CALL THE INTERNAL SUBROUTINE DOTPCT TO   DSP02820
CC*   BUILD THE CLASSIFICATION PERFORMANCE TABLE (PCTAB)                DSP02830
C*                                                                      DSP02840
   43  GO TO 432                                                        DSP02850
   40 IF (TSTKEY.EQ.0)CALL PCT(ILINE(I1),IR(1,I1),TRNFLD,TRNVER,        DSP02860
     *                         TRNSAV,PCTAB,NOFLD2,SAMSTR,SAMEND,SAMINC) DSP02870
      IF(TSTKEY.EQ.1)CALL PCT(ILINE(I1),IR(1,I1),TSTFLD,TSTVER,         DSP02880
     *                         TSTSAV,PCTAB,NOFLD3,SAMSTR,SAMEND,SAMINC) DSP02890
  114 IF (NOMAP.EQ.0) GO TO 135                                         DSP02900
C*                                                                      DSP02910
C                                                                       DSP02920
CC*** CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING                 DSP02930
C                                                                       DSP02940
         IF (LISTSW.NE.0) GO TO 117                                     DSP02950
         IF (DOTERR.GT.0) GO TO 117                                     DSP02960
C*                                                                      DSP02970
CC*                                                                     DSP02980
CC*   OUTLINE TRAINING AND/OR TEST FIELDS                               DSP02990
C*                                                                      DSP03000
      IF(TRNKEY.EQ.1)CALL FLDBOR(TRNNO,ILINE(I1),IR(1,I1),NOFLD2,       DSP03010
     *                   TRNFLD,TRNSAV,TRNVER,NOSUB3,SAMSTR,SAMEND,     DSP03020
     *                   SAMINC,LININC)                                 DSP03030
      IF(TSTKEY.EQ.1)CALL FLDBOR(TSTNO,ILINE(I1),IR(1,I1),NOFLD3,       DSP03040
     *                   TSTFLD,TSTSAV,TSTVER,NOSUB3,SAMSTR,SAMEND,     DSP03050
     *                   SAMINC,LININC)                                 DSP03060
C*    SET UP SYMBOLS FOR THIS LINE. FIRST MAKE SURE I/O FROM LAST LINE  DSP03070
C*    IS COMPLETED.                                                     DSP03080
  117         CONTINUE                                                  DSP03090
      DO 120 I=1,PTS                                                    DSP03100
      L = IR(I,I1)                                                      DSP03110
      IF(L.EQ.0)OUT(I)=BLANK                                            DSP03120
      IF(L.NE.0)OUT(I)=SYMMTX(L)                                        DSP03130
  120 CONTINUE                                                          DSP03140
C*                                                                      DSP03150
C*    WRITE FIRST 110 SAMPLES ON LINE PRINTER AND THE REST ON DRUM      DSP03160
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: DSPLY2 FORTRAN A

```
C*                                                                      DSP03170
      IPTS=PTS                                                          DSP03180
      IF(IPTS.GT.110)IPTS=110                                           DSP03190
      IF(PTS.LE.110)GO TO 125                                           DSP03200
      IPD=PTS-110                                                       DSP03210
      IF(FULL)GO TO 125                                                 DSP03220
      CALL KWRITE(ADRES,OUT(111),IPD,LSTAT)                             DSP03230
      ADRES=ADRES+IPD                                                   DSP03240
      DRUMLN=DRUMLN+1                                                   DSP03250
      IF(ADRES+IPD .LE. DRUMAD+DRMWDS)GO TO 125                         DSP03260
      FULL=.TRUE.                                                       DSP03270
  125 WRITE(6,240)ILINE(I1),(OUT(I),I=1,IPTS)                           DSP03280
C*                                                                      DSP03290
C*    IS CLASSIFICATION MAP TO BE OUTPUT IN UNIVERSAL OR LARSYS         DSP03300
C*    FORMAT                                                            DSP03310
C*                                                                      DSP03320
  135 IF (MAPFMT .LE. 0)  GO TO 200                                     DSP03330
C*                                                                      DSP03340
C*    CHECK TO SEE IF LAST WRITE IS COMPLETED                           DSP03350
C*                                                                      DSP03360
      GO TO (155,160), HDREC                                            DSP03370
C*                                                                      DSP03380
C*    WRITE HEADER RECORD                                               DSP03390
C*                                                                      DSP03400
  155 NC = 1                                                            DSP03410
      LNES = 0                                                          DSP03420
      FEAT = 1                                                          DSP03430
      LSTLIN = 0                                                        DSP03440
      HDREC = 2                                                         DSP03450
      NOFILE = NOFILE + 1                                               DSP03460
      CALL WRTHED(NC,FEAT,PTS,MAPFMT,MAPUNT)                            DSP03470
C*                                                                      DSP03480
C*    WRITE DATA RECORD                                                 DSP03490
C*                                                                      DSP03500
  160 LNES = LNES + 1                                                   DSP03510
      IF (LNES .EQ. LINES)  LSTLIN = -1                                 DSP03520
      CALL WRTLN(IR(1,I1),LSTLIN)                                       DSP03530
  200 CONTINUE                                                          DSP03540
      IF(ILINE(I3).EQ.0)GO TO 201                                       DSP03550
C*                                                                      DSP03560
C*    SET INDICES AND GO READ NEXT LINE                                 DSP03570
C*                                                                      DSP03580
      J=I1                                                              DSP03590
      I1=I2                                                             DSP03600
      I2=I3                                                             DSP03610
      I3=J                                                              DSP03620
      GO TO 91                                                          DSP03630
C*                                                                      DSP03640
C*    LAST LINE IN THIS FIELD HAS BEEN READ. MAKE SURE LAST 2 LINES     DSP03650
C*    ARE PRINTED.                                                      DSP03660
C*                                                                      DSP03670
  201 LAST=LAST+1                                                       DSP03680
      IF(LAST.EQ.2)GO TO 203                                            DSP03690
      I1=I2                                                             DSP03700
      I2=I3                                                             DSP03710
      GO TO 115                                                         DSP03720
C*                                                                      DSP03730
C*    NOW FINISH PRINTING MAP FOR THIS FIELD.                           DSP03740
C*                                                                      DSP03750
  203 CONTINUE                                                          DSP03760
      IF (MAPFMT .GT. 0) WRITE(6,2200)NOFILE,FLDESC,(FORMAT(I,MAPFMT),  DSP03770
     *    I=1,3),LNES                                                   DSP03780
 2200 FORMAT(///T55,'FILE NO.     - ',I6,/T55,'FIELD NAME  - ',A4,/      DSP03790
     *  T55,'FORMAT      - ',3A4,/T55,'NO. RECORDS - ',I6)              DSP03800
      IF (NOMAP .LE. 0 ) GO TO 230                                      DSP03810
C                                                                       DSP03820
      IF(PTS.LE.110)GO TO 230                                           DSP03830
      AD=0                                                              DSP03840
      NRUFS=20                                                          DSP03850
      IF(NRUFS.GT.LINES)NRUFS=LINES                                     DSP03860
      KPTS=PTS-110                                                      DSP03870
      LPTS=110                                                          DSP03880
      NWDTHS=KPTS/110                                                   DSP03890
      IF(MOD(KPTS,110).NE.0)NWDTHS=NWDTHS+1                             DSP03900
      LASTRC=MOD(KPTS,110)                                              DSP03910
      IF(LASTRC.EQ.0)LASTRC=110                                         DSP03920
      I=0                                                               DSP03930
  210 I=I+1                                                             DSP03940
      IF (I.GT.NWDTHS) GO TO 220                                        DSP03950
```

FILE: DSPLY2 FORTRAN A

```
      LINCNT=0                                                          DSP03960
        WRTFG=2                                                         DSP03970
        GO TO 370                                                       DSP03980
 221    PRTFG=2                                                         DSP03990
        GO TO 510                                                       DSP04000
 222    CONTINUE                                                        DSP04010
      IF(I.EQ.NWDTHS)LPTS=LASTRC                                        DSP04020
      ADRES=DRUMAD + AD                                                 DSP04030
      DO 205 J=1,NBUFS                                                  DSP04040
      CALL RREAD(ADRES,BUF(1,J),LPTS,JSTAT(J))                          DSP04050
      ADRES=ADRES+KPTS                                                  DSP04060
      LINCNT=LINCNT + 1                                                 DSP04070
  205 CONTINUE                                                          DSP04080
      LINE=LINSTR                                                       DSP04090
      IBUF=1                                                            DSP04100
C*    FINISHED READING                                                  DSP04110
  210 IF(JSTAT(IBUF).EQ.1)GO TO 210                                     DSP04120
      WRITE(6,240)LINE,(BUF(IK,IBUF),IK=1,LPTS)                         DSP04130
      LINE = LINE + LININC                                              DSP04140
      LINCNT=LINCNT+1                                                   DSP04150
      IF(LINCNT .GT. DRUMLN)GO TO 215                                   DSP04160
      CALL RREAD(ADRES,BUF(1,IBUF),LPTS,JSTAT(IBUF))                    DSP04170
      ADRES=ADRES + KPTS                                                DSP04180
  215 IBUF=IBUF+1                                                       DSP04190
      IF(IBUF.GT.NBUFS)IBUF=1                                           DSP04200
      IF(LINE.LE.LINEND)GO TO 210                                       DSP04210
      AD=AD + LPTS                                                      DSP04220
      GO TO 219                                                         DSP04230
  220 CONTINUE                                                          DSP04240
  240 FORMAT(' ',I5,2X,110A1)                                           DSP04250
      IF(FILTER.EQ.0)GO TO 230                                          DSP04260
      WRITE(6,305)SPKNT                                                 DSP04270
  305 FORMAT(/' THE CLASSIFICATION OF',I7,' PIXELS WAS CHANGED AS A RESUDSP04280
     *LT OF SPATIAL FILTERING'/)                                        DSP04290
  230 CONTINUE                                                          DSP04300
  250 CONTINUE                                                          DSP04310
C                                                                       DSP04320
C*    PRINT CLASSIFICATION SUMMARY FOR THIS FIELD                       DSP04330
C*                                                                      DSP04340
      CALL SETMPG(66,4,62)                                              DSP04350
      CALL PRTSUM(TOTALS,TTOL,FLDESC)                                   DSP04360
            IF (LISTSW.NE.0) GO TO 255                                  DSP04370
      IF ( DOTKEY .EQ. 0  .OR.  DOTERR .GT. 0 )  GO TO 500              DSP04380
C*                                                                      DSP04390
C*    DOTS IN THE DESIGNATED AREA OR NOT IN THE CLASSIFIED AREA WILL    DSP04400
C*    HAVE PCTAR=0                                                      DSP04410
C                                                                       DSP04420
C*** CODE ADDED NOV 13 ,1978 TO INCLUDE LIST PROCESSING                 DSP04430
C                                                                       DSP04440
 255          CONTINUE                                                  DSP04450
C*                                                                      DSP04460
C                                                                       DSP04470
      WRITE (6,5)                                                       DSP04480
 5    FORMAT (1H1)                                                      DSP04490
      DO 7 CHPCT=1,NOFLD2                                               DSP04500
      IF (PCTAB(CHPCT,1).LE.NOSUB3) GO TO 8                             DSP04510
      WRITE (6,4) TRNVER(1,CHPCT),TRNVER(2,CHPCT)                       DSP04520
   4  FORMAT(//3X,'DOT (',I4,',',I4,')  IS IN THE DESIGNATED AREDSP04530
     *EA' )                                                             DSP04540
      PCTAR(CHPCT,1) = 0                                                DSP04550
      GO TO 7                                                           DSP04560
 8    IF (PCTAB(CHPCT,1).GT.0) GO TO 7                                  DSP04570
      WRITE (6,2) TRNVER(1,CHPCT),TRNVER(2,CHPCT)                       DSP04580
   2  FORMAT(//3X,'DOT (',I4,',',I4,')  IS NOT IN THE CLASSIFIDSP04590
     *ED AREA' )                                                        DSP04600
 7    CONTINUE                                                          DSP04610
      WRITE (6,5)                                                       DSP04620
C                                                                       DSP04630
C*** CODE ADDED NOV 13, 1978 TO INCLUDE LIST PROCESSING                 DSP04640
C                                                                       DSP04650
            IF (LISTSW.EQ.0) GO TO 600                                  DSP04660
           FLDCNT = FLDCNT + 1                                          DSP04670
                          CALL LISTSM(TOTALS,TTOL,                      DSP04680
     *       PCTAB,LISTSW,GTUNIT,GTFILE,AIUNIT,AIFILE,                  DSP04690
     *       PRUNIT,PRFILE,NAMECL,ALP,FLDCNT,                           DSP04700
     *       NOCAT,CATNAM,SUBCAT,NOFLD2,NOSUB2)                         DSP04710
C                                                                       DSP04720
C                                                                       DSP04730
C                                                                       DSP04740
```

FILE: DSPLY2 FORTRAN A

```
  500 CONTINUE                                                                  DSP04750
      GO TO 20                                                                  DSP04760
  310 CONTINUE                                                                  DSP04770
      RETURN                                                                    DSP04780
C                                                                               DSP04790
C     SELF-CONTAINED SUBROUTINE TO PRINT HEADERS                                DSP04800
C     ------------------------------------------                                DSP04810
C                                                                               DSP04820
  370 WRITE (6,HEAD)                                                            DSP04830
      WRITE(6,390)FLDESC,CDATE                                                  DSP04840
  390 FORMAT(' DISPLAY OF CLASSIFIED FIELD.....',A4/                            DSP04850
     *       '                    CLASSIFICATION  DATE.....',2A4)               DSP04860
      IF(AMFLG.GT.0)WRITE(6,390)AMCOMB,(FETVC2(L),L=1,AMFEAT)                   DSP04870
      IF(AMFLG.EQ.0)WRITE(6,400) (FETVC2(L),L=1,NOFET2)                         DSP04880
      CALL MAPMD (NOCAT,SYMMTX,CATNAM,KATNO,CLSNAM,SUBNO,SUBNAM,                DSP04890
     *            CLSSUM,NOCLS2,NOSUM2,THRSKY,THRES)                            DSP04900
      IF(DESKEY.EQ.1)WRITE(6,410) SYMMTX(DESUN)                                 DSP04910
  410 FORMAT(/5X,'** DESIGNATED FIELDS SYMBOL IS ',A1/)                         DSP04920
  380 FORMAT(12X,'CLASSIFICATION CHANNELS...',I2,' LINEAR COMBINATIONS O        DSP04930
     *F CHANNELS'/33X, 30I3)                                                    DSP04940
  400 FORMAT(12X,'CLASSIFICATION CHANNELS...',30I3)                             DSP04950
      GO TO (31,221),PRTFG                                                      DSP04960
C                                                                               DSP04970
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*  DSP04980
C*                                                                              DSP04990
C*    INTERNAL ROUTINE TO PRINT COLUMN NUMBERS                                  DSP05000
C*                                                                              DSP05010
  510 J=0                                                                       DSP05020
      DO 50 IJ=ISTRT,IEND,SAMINC                                                DSP05030
      J=J+1                                                                     DSP05040
      COL(1,J) = IJ/100                                                         DSP05050
      COL(2,J)=MOD(IJ,100)/10                                                   DSP05060
      COL(3,J)=MOD(IJ,10)                                                       DSP05070
      IF(IJ.EQ.110)GO TO 60                                                     DSP05080
   50 CONTINUE                                                                  DSP05090
   60 SAMEN=IJ                                                                  DSP05100
      ISTRT=SAMEN + SAMINC                                                      DSP05110
      IPTS=J                                                                    DSP05120
      WRITE(6,96)                                                               DSP05130
      DO 80 IJ=1,3                                                              DSP05140
   80 WRITE(6,90)(COL(IJ,J),J=1,IPTS)                                           DSP05150
   90 FORMAT(9X,110I1)                                                          DSP05160
      WRITE(6,96)                                                               DSP05170
   96 FORMAT(/)                                                                 DSP05180
      GO TO (85,222),PRTFG                                                      DSP05190
C                                                                               DSP05200
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*  DSP05210
C                                                                               DSP05220
C     INTERNAL SUBROUTINE DOTPCT                                                DSP05230
C                                                                               DSP05240
C*    PURPOSE - TO SET PCTAB TO EACH DOT'S RESPECTIVE SUBCLASS                  DSP05250
C*      (OR THRESHOLD) NUMBER WHEN THE DOTKEY FLAG IS ON                        DSP05260
C                                                                               DSP05270
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*  DSP05280
C*                                                                              DSP05290
C*    TEST FOR THE POSITION OF THE DOT IN THE CLASSIFICATION                    DSP05300
C*    RECORD                                                                    DSP05310
C*                                                                              DSP05320
C*    IF THE DOT'S POSITION IS NOT COMPATIBLE WITH THE CLASSIFICATION           DSP05330
C*    RECORD, PCTAB(DOT,1) = 0                                                  DSP05340
C*                                                                              DSP05350
  432 IF (PCTKEY.NE.1) GO TO 430                                                DSP05360
      DO 431 I=1,NOFLD2                                                         DSP05370
  431 PCTAB(I,1)=0                                                              DSP05380
      PCTKEY = 1                                                                DSP05390
  430 DO 411 K=FCNT,ECNT                                                        DSP05400
      SAMDF=TUNVER(1,K) - SAMSTR                                                DSP05410
      SAMPS=SAMDF/SAMINC + 1                                                    DSP05420
      CHSPS=(SAMPS-1)*SAMINC                                                    DSP05430
      SAMDF = SAMDF + SAMSTR                                                    DSP05440
      CHSPS = CHSPS + SAMSTR                                                    DSP05450
      IF (CHSPS.NE.SAMDF) GO TO 421                                             DSP05460
      PCTAB(K,1) = IR(SAMPS,I1)                                                 DSP05470
      GO TO 411                                                                 DSP05480
  421 PCTAB(K,1) = 0                                                            DSP05490
  411 CONTINUE                                                                  DSP05500
      GO TO 114                                                                 DSP05510
C                                                                               DSP05520
C                                                                               DSP05530
```

12-17
216

FILE: DSPLY2 FORTRAN A

```
C*****************************************************************************DSP05540
CC                                                                             DSP05550
CC                                                                             DSP05560
CC     INTERNAL SUBROUTINE DOTSUM                                              DSP05570
CC                                                                             DSP05580
C* PURPOSE - TO COMPUTE BOTH UNCORRECTED PROPORTIONS AND BIAS-CORRECTED         DSP05590
C*  PROPORTIONS OF THE ANALYST-LABELED CATEGORIES IN THE CLASSIFIED            DSP05600
C*  AREA.  ALSO COMPUTES THE ALPHA TABLE AND OUTPUTS TWO CATEGORY              DSP05610
C*  CLASSIFICATION PERFORMANCE SUMMARIES -                                     DSP05620
C*        (1)  SUMMARY OF ANALYST-LABELED CATEGORIES VERSUS TOTAL              DSP05630
C*             NUMBER OF DOTS CLASSIFIED INTO EACH CATEGORY.                   DSP05640
C*        (2)  CLASSIFICATION SUMMARY FOR EACH INDIVIDUAL DOT.                 DSP05650
CC                                                                             DSP05660
C*****************************************************************************DSP05670
C*                                                                             DSP05680
C*     INITIALIZE ARRAYS                                                       DSP05690
C*                                                                             DSP05700
  600   TYPE=TRNSAV(3,1)                                                       DSP05710
C*     ERROR FLAG FOR INVALID SUBCLASS                                         DSP05720
       NOGOOD=-54321                                                          DSP05730
C*     NUMBER OF DOT LABLED CATAGORIES                                         DSP05740
       NOCAT=DOTKEY                                                           DSP05750
C*     NUMBER OF CLASSIFICATION DOT CATAGORIES                                 DSP05760
       NCLAT=NOCAT                                                            DSP05770
C*     NO. OF CLASSIFIED CAT. + 1 FOR THRESHOLDED DOTS                         DSP05780
       NCLCAT = NOCAT + 1                                                     DSP05790
C*                                                                             DSP05800
C*     ON INPUT TO DOTSUM,                                                     DSP05810
C*     TRNSAV(1,I) =  SCRATCH USAGE                                            DSP05820
C*     TRNSAV(2,I) = MAPTAP CATEGORY NO. CORRESPONDING TO THE DOT'S            DSP05830
C*     TRNSAV(3,I) = DOT TYPE  =1  OR  =2                                      DSP05840
C*     TRNSAV(4,I) = NO. OF VERTICES FOR THE DOT  =2                           DSP05850
C*                                                                             DSP05860
C*     TRNVER(1,I) = DOT SAMPLE NO.                                            DSP05870
C*     TRNVER(2,I) = DOT LINE NO.                                              DSP05880
C*                                                                             DSP05890
C*     FOR I = 1,2, ... , NO. OF DOTS                                          DSP05900
C*                                                                             DSP05910
C*                                                                             DSP05920
       DO 610 I=1,NOFLD2                                                       DSP05930
C*     LABLED DOT CATAGORY                                                     DSP05940
       TRNSAV(3,I)=TRNSAV(2,I)                                                 DSP05950
C*     VERTEX SAMPLE OF DOT                                                    DSP05960
       TRNSAV(1,I)=TRNVER(1,I)                                                 DSP05970
C*     LINE NUMBER OF DOT                                                      DSP05980
       TRNSAV(2,I)=TRNVER(2,I)                                                 DSP05990
  610   CONTINUE                                                               DSP06000
       DO 630 I=1,NCLCAT                                                       DSP06010
       CLTOT(I)=0                                                              DSP06020
  630   CONTINUE                                                               DSP06030
       DO 130 I=1,NOCAT                                                        DSP06040
       LBLTOT(I) = 0                                                           DSP06050
       TOTLBL(I)=0                                                             DSP06060
       TOTTHR(I)=0                                                             DSP06070
       DO 620 J=1,NCLCAT                                                       DSP06080
       LABCLS(I,J) = 0                                                         DSP06090
  620   CONTINUE                                                               DSP06100
  130  CONTINUE                                                                DSP06110
C*                                                                             DSP06120
C*     COMPUTE TOTAL NUMBER OF PIXELS CLASSIFIED IN TOTAL CLASSIFIED           DSP06130
C*     AREA WITHOUT THRESHOLDED PIXELS                                         DSP06140
C*                                                                             DSP06150
       PIXTOT=0.0                                                              DSP06160
       DO 620 I=1,NOSUB2                                                       DSP06170
       PIXTOT=PIXTOT + TOTALS(I)                                               DSP06180
  620   CONTINUE                                                               DSP06190
C*                                                                             DSP06200
C*     CALCULATE TOTAL NO. PIXELS FOR EACH ANALYST-LABELED CATEGORY            DSP06210
C*                                                                             DSP06220
       DO 631 I=1,NOSUB2                                                       DSP06230
       CAT = SUBCAT(I)                                                         DSP06240
       LBLTOT(CAT) = LBLTOT(CAT) + TOTALS(I)                                   DSP06250
  631   CONTINUE                                                               DSP06260
C*                                                                             DSP06270
C*     COMPUTE DOT TOTALS CLASSIFIED BY CATEGORY                               DSP06280
C*                                                                             DSP06290
       DO 640 I=1,NOFLD2                                                       DSP06300
       SUBCL=PCTAB(I,1)                                                        DSP06310
       IF (SUBCL.LE.0.OR.SUBCL.GT.NOSUB3) GO TO 641                            DSP06320
```

FILE: DSPLY2 FORTRAN A

```
      CAT=SURCAT(SURCL)                                                 DSP06330
      CLTOT(CAT)=CLTOT(CAT) + 1                                         DSP06340
        GO TO 643                                                       DSP06350
  641  CONTINUE                                                         DSP06360
      PCTAB(I,1) = NOGOOD                                               DSP06370
  643  CONTINUE                                                         DSP06380
  640  CONTINUE                                                         DSP06390
C                                                                       DSP06400
C*    BUILD AN ORDERED SET OF DOT CATEGORY LABELS FOR ANALYST-LABELED   DSP06410
C*     CATEGORIES                                                       DSP06420
C                                                                       DSP06430
      J=1                                                               DSP06440
      LBL=TRNSAV(3,1)                                                   DSP06450
      DOTORD(J,1) = CATNAM(LBL)                                         DSP06460
      DOTORD(J,2) = LBL                                                 DSP06470
        DO 660 I=2,NOFLD2                                               DSP06480
      LBL=TRNSAV(3,I)                                                   DSP06490
        DO 660 K=1,J                                                    DSP06500
      IF(LBL.EQ.DOTORD(K,2)) GO TO 49                                   DSP06510
  660  CONTINUE                                                         DSP06520
      J=J+1                                                             DSP06530
      DOTORD(J,1) = CATNAM(LBL)                                         DSP06540
      DOTORD(J,2) = LBL                                                 DSP06550
   49  CONTINUE                                                         DSP06560
  650  CONTINUE                                                         DSP06570
C                                                                       DSP06580
      IF ( J .NE. NDCAT )  WRITE(6,3003)  J , NDCAT                     DSP06590
C                                                                       DSP06600
 3003 FORMAT(//// 5X, '**** DSPLY2/DOTSUM -- DISCREPANCY IN DOTFILE INFODSP06610
     *RMATION ****' // 27X,'NO. OF DOT CATEGORY LABELS MATCHING MAPTAP  DSP06620
     *CATEGORY NAMES =' , I6// 27X, 'NO. OF DOT CATEGORIES IS GIVEN AS =DSP06630
     * ' , I6 //// )                                                    DSP06640
C                                                                       DSP06650
      DO 53 I=1,NDCAT                                                   DSP06660
      K = I                                                             DSP06670
      DO 52 J=1,NDCAT                                                   DSP06680
      JK = DOTORD(J,2)                                                  DSP06690
      IF(JK.NE.K) GO TO 51                                              DSP06700
      DOTLBL(K,1) = DOTORD(J,1)                                         DSP06710
      DOTLBL(K,2) = DOTORD(J,2)                                         DSP06720
   51 CONTINUE                                                          DSP06730
   52 CONTINUE                                                          DSP06740
   53 CONTINUE                                                          DSP06750
C                                                                       DSP06760
C*    COMPUTE UNCORRECTED PROPORTIONS OF DOT CATEGORIES OVER THE TOTAL  DSP06770
C*     AREA CLASSIFIED                                                  DSP06780
C                                                                       DSP06790
      DO 70 I=1,NDCAT                                                   DSP06800
      J=DOTLBL(I,2)                                                     DSP06810
      PUNC(I) =  FLOAT(LBLTOT(J))/PIXTOT                                DSP06820
   70  CONTINUE                                                         DSP06830
C                                                                       DSP06840
C*    INITIALIZE TRNSAV(4,I),I=1,NOFLD2 (NO. OF DOTS) TO CONTAIN        DSP06850
C*    CLASSIFICATION CATEGORY NUMBER OF DOT I.                          DSP06860
C                                                                       DSP06870
        DO 680 I=1,NOFLD2                                               DSP06880
      SURCL = PCTAB(I,1)                                                DSP06890
      IF(SURCL.EQ.NOGOOD) CAT = NOGOOD                                  DSP06900
      IF (SURCL.GT.NOSUM3) CAT=NDCAT + 1                                DSP06910
      IF(SURCL.GE.1.AND.SURCL.LE.NOSUM2)CAT=SURCAT(SURCL)               DSP06920
      TRNSAV(4,I)=CAT                                                   DSP06930
  680  CONTINUE                                                         DSP06940
C                                                                       DSP06950
C                                                                       DSP06960
C*    TABLE OF TOTALS OF ANALYST-LABELED DOT CATEGORIES VERSUS          DSP06970
C*     CLASSIFIED DOT CATEGORIES.  LABCLS(LABELED CAT,CLASSIFIED CAT)   DSP06980
C*                                                                      DSP06990
C*                                                                      DSP07000
        DO 615 J=1,NOFLD2                                               DSP07010
      CLS=TRNSAV(4,J)                                                   DSP07020
        IF (CLS.EQ.NOGOOD) GO TO 616                                    DSP07030
      LB=TRNSAV(3,J)                                                    DSP07040
      LABCLS(LB,CLS) = LABCLS(LB,CLS) + 1                               DSP07050
  615  CONTINUE                                                         DSP07060
  616  CONTINUE                                                         DSP07070
C                                                                       DSP07080
C*    COMPUTE ALPHA TABLE FOR LABELED VERSUS CLASSIFIED DOTS            DSP07090
C                                                                       DSP07100
        DO 141 I=1,NDCAT                                                DSP07110
```

~~12-19~~
218

FILE: DSPLY2 FORTRAN A

```
       DO 156 J=1,NDCAT                                                  DSP07120
       IF(CLTOT(I).LE.0) GO TO 153                                       DSP07130
       ALPHA(I,J) =  FLOAT(LABCLS(J,I))/ FLOAT(CLTOT(I))                 DSP07140
       GO TO 154                                                         DSP07150
  153 ALPHA(I,J) = 0.0                                                   DSP07160
  154 CONTINUE                                                           DSP07170
  156   CONTINUE                                                         DSP07180
  161   CONTINUE                                                         DSP07190
C*                                                                       DSP07200
C*     COMPUTE CORRECTED PROPORTIONS FOR LABELED DOT CATEGORIES          DSP07210
C*                                                                       DSP07220
       DO 180 I=1,NDCAT                                                  DSP07230
       PCTOT=0.0                                                         DSP07240
       DO 170 J=1,NDCAT                                                  DSP07250
       PCTOT=PCTOT + ALPHA(J,I)*PUNC(J)                                  DSP07260
  170  CONTINUE                                                          DSP07270
       PCORR(I)=PCTOT*100                                                DSP07280
  180  CONTINUE                                                          DSP07290
C*                                                                       DSP07300
C*     MAKE PUNC AND ALPHA TO REPRESENT PERCENTAGES                      DSP07310
C*                                                                       DSP07320
       DO 185 J=1,NDCAT                                                  DSP07330
       PUNC(J)=PUNC(J) * 100.0                                           DSP07340
       DO 175 I=1,NDCAT                                                  DSP07350
       ALPHA(I,J) = ALPHA(I,J) * 100.0                                   DSP07360
  175 CONTINUE                                                           DSP07370
  185 CONTINUE                                                           DSP07380
C*                                                                       DSP07390
C*     CALCULATE TOTAL NO. LABELED DOTS IN EACH LABELED DOT CATEGORY     DSP07400
C*                                                                       DSP07410
       DO 199 J=1,NDCAT                                                  DSP07420
       LBL=DOTLBL(J,2)                                                   DSP07430
       DO 190 I=1,NOFLD2                                                 DSP07440
       IF (TRNSAV(3,I).EQ.LBL) TOTLBL(J)=TOTLBL(J)+1                     DSP07450
  190  CONTINUE                                                          DSP07460
  199   CONTINUE                                                         DSP07470
C*                                                                       DSP07480
C*     TOTTHR WILL CONTAIN TOTAL THRESHOLDED DOTS IN EACH ANALYST-LABELEDSP07490
C*      CATEGORY                                                         DSP07500
C*                                                                       DSP07510
       DO 251   I=1,NDCAT                                                DSP07520
       TOTTHR(I) = LABCLS(I,NCLCAT)                                      DSP07530
  251   CONTINUE                                                         DSP07540
C      PRINT SUMMARY TABLES                                              DSP07550
       K1=1                                                              DSP07560
       K2=NOCAT                                                          DSP07570
       IF(NOCAT.GT.3)K2=3                                                DSP07580
       WX=1                                                              DSP07590
       NN = NDCAT                                                        DSP07600
       IF(NDCAT.GT.30)NN=30                                              DSP07610
  900  WRITE(6,1000)                                                     DSP07620
 1000  FORMAT(1H1,T35,'DOT DATA PERFORMANCE SUMMARY')                    DSP07630
       WRITE(6,1002)                                                     DSP07640
 1002  FORMAT(/ T25,'(SUMMARY OF ANALYST LABELED VERSUS CLASSIFIED CATEGODSP07650
      *RIES)')                                                          DSP07660
       WRITE(6,1004)                                                     DSP07670
 1004  FORMAT(//,T2,'ANALYST',T14,'TOTAL NUMBER',T31,'PERCENT',T46,'NUMBEDSP07680
      *R',T58,'UNCORRECTED',T76,'BIAS',T92,'DOT CLASSIFICATION SUMMARY')  DSP07690
       WRITE(6,1005)                                                     DSP07700
 1005  FORMAT(T2,'LABELED',T15,'ANALYST',T30,'CORRECTLY',T43,'THRESHOLDEDDSP07710
      *',T59,'CATEGORY',T74,'CORRECTED',T102,'(ALPHA)')                  DSP07720
       WRITE(6,1006)                                                     DSP07730
 1006  FORMAT(T4,'DOT',T14,'LABELED DOTS',T29,'CLASSIFIED',T58,'PROPORTIODSP07740
      *N',T74,'CATEGORY')                                               DSP07750
       WRITE(6,1008)(CATNAM(I),I=1,K2)                                   DSP07760
 1008  FORMAT(T2, 'CATEGORY',T62,'%',T73,'PROPORTION',1X,3(9X,A4)/T77,'%'DSP07770
      *  // )                                                           DSP07780
       DO 1010 I=WX,NN                                                   DSP07790
       PCC = (FLOAT(LABCLS(I,I))/ FLOAT(TOTLBL(I))) * 100.0              DSP07800
       WRITE(6,2001)DOTLBL(I,1),TOTLBL(I),PCC,TOTTHR(I),PUNC(I),PCORR(I),DSP07810
      *(LABCLS(I,J),J=K1,K2)                                             DSP07820
 2001  FORMAT(/T4,A4,T16,I4,T30,F7.3,T45,I4,T60,F7.3,T74,F7.3,T91,3(I6,7XDSP07830
      *))/)                                                             DSP07840
       CALL RNI4A1(FMT3(3),1,K2)                                         DSP07850
       WRITE(6,FMT3)(ALPHA(J,I),J=K1,K2)                                 DSP07860
 1010  CONTINUE                                                          DSP07870
       IF(NOCAT.LE.3)GO TO 2005                                          DSP07880
       K3=K2                                                             DSP07890
 1200  WRITE(6,3000)                                                     DSP07900
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: DSPLY2 FORTRAN A

```
 3000 FORMAT(1H1,T2,'ANALYST LABELED',T50,'DOT CLASSIFICATION SUMMARY')  DSP07910
      WRITE(6,3001)                                                     DSP07920
 3001 FORMAT(T3,'DOT CATEGORY',T59,'(ALPHA)')                           DSP07930
      K3=K3 + 1                                                         DSP07940
      KNN = K3 + 8- 1                                                   DSP07950
      IF(NOCAT.LE.KNN)KNN=NOCAT                                         DSP07960
      WRITE(6,3002)(CATNAM(I),I=K3,KNN)                                 DSP07970
 3002 FORMAT(/T25,8(A4,9X))                                             DSP07980
      DO 4000 II=WX,NN                                                  DSP07990
      WRITE(6,3004) DOTLBL(II,1), (LABCLS(II,J),J=K3,KNN)               DSP08000
 3004 FORMAT(//TH,A4,T25,8(I6,7X))                                      DSP08010
      KN=KNN-K3+1                                                       DSP08020
      CALL BNI4A1(FMT4(3),1,KN)                                         DSP08030
      WRITE(6,FMT4)(ALPHA(J,II),J=K3,KNN)                               DSP08040
 4000 CONTINUE                                                          DSP08050
      IF(KNN.LE.NOCAT)GO TO 2005                                        DSP08060
      K3=KNN                                                            DSP08070
      GO TO 1200                                                        DSP08080
 2005 CONTINUE                                                          DSP08090
      IF(NN.GE.NDCAT) GO TO 4999                                        DSP08100
      WX=NN +1                                                          DSP08110
      NN= WX + 7                                                        DSP08120
      IF(NN.GT.NDCAT) NN=NDCAT                                          DSP08130
      GO TO 800                                                         DSP08140
 4999 SUBNAM(NOSUB3)=THRESH                                             DSP08150
      PG = 0                                                            DSP08160
      WRITE(6,5001)                                                     DSP08170
 5001 FORMAT(1H1,T39,'DOT DATA PERFORMANCE SUMMARY')                    DSP08180
      WRITE(6,5003)                                                     DSP08190
 5003 FORMAT(/T39,'(SUMMARY OF INDIVIDUAL DOTS)')                       DSP08200
      WRITE(6,5004)                                                     DSP08210
 5004 FORMAT(T43,'DOT ID = (SAMPLE,LINE)',/)                            DSP08220
      SAMP2 = 0                                                         DSP08230
      NWRITE = NOFLD2/4                                                 DSP08240
      IF(MOD(NOFLD2,4).GT.0)NWRITE=NWRITE+1                             DSP08250
      DO 7000 I=1,NWRITE                                                DSP08260
      PG = PG + 1                                                       DSP08270
      SAMP1 = SAMP2 + 1                                                 DSP08280
      SAMP2=SAMP1 + 3                                                   DSP08290
      IF ( SAMP2 .GT. NOFLD2)  SAMP2= NOFLD2                            DSP08300
      NSAMP = SAMP2 - SAMP1 + 1                                         DSP08310
      CALL BNI4A1(FMT5(3),1,NSAMP)                                      DSP08320
      WRITE(6,7005)(BLANKS(1),TRNSAV(1,S),TRNSAV(2,S),S=SAMP1,SAMP2)    DSP08330
 7005 FORMAT(//T7,3(A1,1H(,I3,1H,,I3,1H),23X),A1,1H(,I3,1H,,I3,1H))     DSP08340
      CALL BNI4A1(FMT1(3),1,NSAMP)                                      DSP08350
      CALL BNI4A1(FMT1(10),1,NSAMP)                                     DSP08360
      CALL BNI4A1(FMT2(3),1,NSAMP)                                      DSP08370
      CALL BNI4A1(FMT2(14),1,NSAMP)                                     DSP08380
C     WRITE(6,FMT1)                                                     DSP08390
      WRITE(6,7001) (BLANKS(Z),Z=1,NSAMP)                               DSP08400
 7001 FORMAT (T7,3(A1,8HCATEGORY,24X),A1,8HCATEGORY)                    DSP08410
      WRITE (6,8001) (BLANKS(Z),Z=1,NSAMP)                              DSP08420
 8001 FORMAT (T1,3(A1,26H-------  ----------------,6X),A1,              DSP08430
     *  26H-------  ----------------)                                   DSP08440
      WRITE(6,FMT2)                                                     DSP08450
      J=0                                                               DSP08460
      DO 6000 S=SAMP1,SAMP2                                             DSP08470
      J=J+1                                                             DSP08480
      XYZ=TRNSAV(3,S)                                                   DSP08490
      ABC=TRNSAV(4,S)                                                   DSP08500
      DEF = PCTSAV(S,1)                                                 DSP08510
      TEMP(J,1)=CATNAM(XYZ)                                             DSP08520
      IF(DEF.EQ.000000) GO TO 5998                                      DSP08530
      TEMP(J,2)=CATNAM(ABC)                                             DSP08540
      IF(DEF.EQ.NOSUB3)TEMP(J,2)=AST                                    DSP08550
      TEMP(J,3)=SUBNAM(DEF)                                             DSP08560
      GO TO 5999                                                        DSP08570
 5998 TEMP(J,3)=AST                                                     DSP08580
      TEMP(J,2)=AST                                                     DSP08590
 5999 CONTINUE                                                          DSP08600
 6000 CONTINUE                                                          DSP08610
      WRITE(6,5020)(TEMP(P,1),TEMP(P,2),TEMP(P,3),P=1,J)                DSP08620
 5020 FORMAT(/T4,3(A4,5X,A4,6X,'(',A4,')',8X),A4,5X,A4,6X,'(',A4,')')   DSP08630
      IF(PG.LT.5)GO TO 6001                                             DSP08640
      PG = 0                                                            DSP08650
      WRITE(6,6002)                                                     DSP08660
 6002 FORMAT(1H1)                                                       DSP08670
 6001 CONTINUE                                                          DSP08680
      W = I                                                             DSP08690
```

FILE: DSPLY2 FORTRAN A

```
 7000 CONTINUE                                                          DSP08700
      GO TO 500                                                         DSP08710
      END                                                               DSP08720
```

FILE: EMTHRS FORTRAN A

```
      SUBROUTINE EMTHRS(FLDSAV,FIELD,VERTEX,NOFLD)                          EMT00010
CI----------------------------------------------------------------------    EMT00020
CI                                                                          EMT00030
CI      THIS SUBROUTINE IS USED ONLY WHEN THE EMPIRICAL THRESHOLD OPTION    EMT00040
CI     REQUESTED                                                            EMT00050
CI    OR IF HISTOGRAMS OF THE QUADRATIC FORM WERE REQUESTED WITH SOME       EMT00060
CI    OTHER THRESHOLDS                                                      EMT00070
CI                                                                          EMT00080
CI----------------------------------------------------------------------    EMT00090
      IMPLICIT INTEGER (A-Z)                                                EMT00100
      REAL VR,DSFUNC,THRES,CON                                              EMT00110
C     INCLUDE COMMKS.LIST                                                   EMT00120
C     INCLUDE CMMS10.LIST                                                   EMT00130
      COMMON/GLOBAL/READ(63),MAPTAP,DATAPE,SAVTAP,AMFILE,AMKEY,            EMT00140
     *              HISFIL,HISKEY,TRFORM,ERRPTP,ERRKEY,MAPUNT,NOFILE,      EMT00150
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    EMT00160
     *  ,NMSTUN,NMSTFI,SCTHUN,MAPFIL                                        EMT00170
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                  EMT00180
     *  CRDUNT,PRTUNT,HANDIO                                                EMT00190
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60),       EMT00200
     *             SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,             EMT00210
     *             PCFDKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,      EMT00220
     *             PLTKEY,AMFLG,AMCOMB,AMFEAT,CDATE(2),                   EMT00230
     *             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,      EMT00240
     *             THRES(60),SYMMTX(66),HIGH(60),CON(60)                  EMT00250
     *             ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)                EMT00260
     *             ,NOSUB2,NOTHRD,TOTVT2,NOCLS2                           EMT00270
     *             ,KATNO(60),NOCAT,FILTER,MAPFMT                         EMT00280
     *             ,DESKEY,DESUNI,DESOTH,CROP ,ACROP,AOTHER,ATOTAL        EMT00290
     *             ,SITE(6),ANALYS(5),CAM(15),CRPKEY,KEPPTS(60)           EMT00300
     *             ,DOTKEY,DOTERR                                         EMT00310
C$END                                                                       EMT00320
      DATA RANGE/100/                                                       EMT00330
      DIMENSION DSFUNC(100,60),IR(1000),VR(1000),FLDINF(6),SCRAT(500)       EMT00340
      EQUIVALENCE (SCRAT,IR)                                                EMT00350
   10 READ(MAPTAP)FLDINF    ,PTS,LINES,FLDESC                             EMT00360
      IF(PTS.EQ.0)GO TO 40                                                  EMT00370
      DO 30 I=1,LINES                                                       EMT00380
      READ(MAPTAP)ILINE,(IR(J),J=1,PTS),(VR(J),J=1,PTS)                     EMT00390
C*    HISTOGRAM VR WITHIN GROUND TRUTH FIELDS                               EMT00400
      CALL PCTT(ILINE,IR,VR,FIELD,VERTEX,FLDSAV,DSFUNC,NOFLD,               EMT00410
     *          FLDINF(4),FLDINF(5),FLDINF(6),CON,RANGE)                    EMT00420
   30 CONTINUE                                                              EMT00430
C*    READ END OF FIELD RECORD AND GO SEE IF THERE IS ANOTHER FIELD         EMT00440
      READ(MAPTAP)ILINE                                                     EMT00450
      GO TO 10                                                              EMT00460
C*                                                                          EMT00470
C*    ALL FIELDS ON THIS FILE HAVE BEEN PROCESSED NOW PLOT THE HISTOGRAM    EMT00480
C*                                                                          EMT00490
   40 CONTINUE                                                              EMT00500
      CALL DISTCV(DSFUNC,SCRAT,RANGE)                                       EMT00510
C*                                                                          EMT00520
C*    NOW GO BACK TO BEGINNING OF THIS FILE AND POSITION TAPE OVER THE      EMT00530
C*    FOUR HEADER RECORDS - GETTING IT READY DSPLY2                         EMT00540
C*                                                                          EMT00550
      REWIND MAPTAP                                                         EMT00560
      CALL FSRSFL(MAPTAP,4,ISTAT)                                           EMT00570
      IF (ISTAT .GT. 0)  WRITE(6,500) ISTAT                                 EMT00580
  500 FORMAT(' ERROR BACK SPACING MAPTAP','ISTAT = ',I5)                    EMT00590
      IF (ISTAT .GT. 0)  CALL CMERR                                         EMT00600
      DO 210    J=1,4                                                       EMT00610
  210 READ(MAPTAP)                                                          EMT00620
      RETURN                                                                EMT00630
      END                                                                   EMT00640
```

FILE: FDIST FORTRAN A

```
      SUBROUTINE FDIST                                                      FDI00010
C-                                                                          FDI00020
C-    ROUTINE TO USE FISHIN TO GET THRESHOLD VALUES                         FDI00030
C-                                                                          FDI00040
      DIMENSION F(60)                                                       FDI00050
C-                                                                          FDI00060
C     INCLUDE CMBK10:LIST                                                   FDI00070
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60),       CMB00010
     *             SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,               CMB00020
     *             PCFUKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,        CMB00030
     *             PLTKEY,AMFLG,AMCOMM,AMFEAT,CDATE(2),                     CMB00040
     *             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,        CMB00050
     *             THRES(60),SYMMTX(66),HIGH(60),CON(60)                    CMB00060
     *             ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)                  CMB00070
     *             ,NOSUB2,NOTHED,TOTVT2,NOCLS2                             CMB00080
     *             ,KATNO(60),NOCAT,FILTER,MAPFMT                           CMB00090
     *             ,DESKEY,DESUN1,DESOTH,CROP  ,ACROP,AOTHER,ATOTAL         CMB00100
     *             ,SITE(6),ANALYS(5),CAM(15),CMPKEY,KEPPTS(60)             CMB00110
     *             ,DOTKEY,DOTERR                                           CMB00120
C$END                                                                       FDI00090
C-                                                                          FDI00100
C-     NOSUB2 = NUMBER OF SUBCLASSES                                        FDI00110
C-    THRES(I) CONTAINS INPUT CONFIDENCE LEVELS                             FDI00120
C-                                                                          FDI00130
C-                                                                          FDI00140
      DO 10 I=1,NOSUB2                                                      FDI00150
   10 F(I)=1-THRES(I)                                                       FDI00160
C-                                                                          FDI00170
C-      NX = NUMBER OF SAMPLES                                              FDI00180
C- NOFET2 = NUMBER OF CHANNELS                                              FDI00190
C-                                                                          FDI00200
C-    COMPUTE THRESHOLD VALUES USING FISHER F-DISTRIBUTION FUNTION          FDI00210
C-                                                                          FDI00220
      DO 20 I=1,NOSUB2                                                      FDI00230
      FX = KEPPTS(I)                                                        FDI00240
      NS=KEPPTS(I)-NOFET2                                                   FDI00250
C-                                                                          FDI00260
      IFLAG=0                                                               FDI00270
      VAR=FISHIN(F(I),NOFET2,NS,IFLAG)                                      FDI00280
      IF(IFLAG.EQ.1) GOTO 15                                                FDI00290
C-                                                                          FDI00300
      FK =(NOFET2*(FX-1)*(FX+1))/(NS*FX)                                    FDI00310
C-                                                                          FDI00320
      THRES(I) = FK*VAR                                                     FDI00330
C-                                                                          FDI00340
      GO TO 20                                                              FDI00350
C-                                                                          FDI00360
C-                                                                          FDI00370
C-                                                                          FDI00380
C-    THRESHOLD WILL BE SET TO 999.999 IF OVERFLOW OCCURS IN FISHIN.        FDI00390
C-                                                                          FDI00400
   15 THRES(I)=999.999                                                      FDI00410
      WRITE(6,11)I                                                          FDI00420
   11 FORMAT(6X,'FDIST- OVERFLOW CONDITION IN FISHIN ROUTINE FOR SUBCLASS   FDI00430
     1S=',I4,'. THRESHOLD SET TO 999.999')                                  FDI00440
C-                                                                          FDI00450
C-                                                                          FDI00460
   20 CONTINUE                                                              FDI00470
C-                                                                          FDI00480
      RETURN                                                                FDI00490
      END                                                                   FDI00500
```

FILE: FISH FORTRAN A

```
      FUNCTION FISH(F,N1,N2)                                            FIS00010
      LOGICAL E1,E2,E3                                                  FIS00020
      E1=.FALSE.                                                        FIS00030
      E2=.FALSE.                                                        FIS00040
      E3=.FALSE.                                                        FIS00050
      IF(MOD(N1,2).EQ.0) E1=.TRUE.                                      FIS00060
      IF(MOD(N2,2) .EQ.0) E2=.TRUE.                                     FIS00070
      X=N2/(N2+N1*F)                                                    FIS00080
      IF(.NOT.(E1.OR.E2)) GO TO 5                                       FIS00090
      IF(E1.AND..NOT.E2) GO TO 1                                        FIS00100
      IF(.NOT.E1.AND.E2) GO TO 2                                        FIS00110
      IF(N1.LE.N2) GO TO 1                                              FIS00120
    2 I=N1                                                              FIS00130
      N1=N2                                                             FIS00140
      N2=I                                                              FIS00150
      X=1.0-X                                                           FIS00160
      E3=.TRUE.                                                         FIS00170
    1 Y=1.0-X                                                           FIS00180
      FISH=0.0                                                          FIS00190
      H=SQRT(X**N2)                                                     FIS00200
      M=N1/2-1                                                          FIS00210
      I=-1                                                              FIS00220
    3 I=I+1                                                             FIS00230
      IF (I.GT.M) GO TO 10                                              FIS00240
      FISH=FISH+H                                                       FIS00250
      H=(H*Y*(N2+2.*I))/(2.*(I+1.))                                     FIS00260
      GO TO 3                                                           FIS00270
   10 IF(E3) GO TO 4                                                    FIS00280
      FISH=1.0-FISH                                                     FIS00290
      RETURN                                                            FIS00300
    4 I=N1                                                              FIS00310
      N1=N2                                                             FIS00320
      N2=I                                                              FIS00330
      RETURN                                                            FIS00340
    5 Y=1.0-X                                                           FIS00350
      H=.63661977*SQRT(X*Y)                                             FIS00360
      FISH=.63661977*ARCOS(SQRT(X))                                     FIS00370
      IF(N2.EQ.1) GO TO 8                                               FIS00380
      M=N2-2                                                            FIS00390
      DO  6 I=1,M,2                                                     FIS00400
      FISH=FISH+H                                                       FIS00410
    6 H=H*X*(I+1)/(I+2)                                                 FIS00420
    8 IF(N1.EQ.1) RETURN                                                FIS00430
      H=H*N2                                                            FIS00440
      M=N1-2                                                            FIS00450
      DO  7  I=1,M,2                                                    FIS00460
      FISH=FISH+H                                                       FIS00470
    7 H=H*Y*(N2+I)/(I+2)                                                FIS00480
      RETURN                                                            FIS00490
      END                                                               FIS00500
```

FILE: FISHIN FORTRAN A

```
      FUNCTION FISHIN(ALPHA,N1,N2,IFLAG)                                FIS00010
      Y1=N1                                                             FIS00020
      Y2=N2                                                             FIS00030
      IF(N1.EQ.1) Y1=2                                                  FIS00040
      IF(N2.EQ.1) Y2=2                                                  FIS00050
      X=TINORM(1.-ALPHA,IFLAG)                                          FIS00060
      IF(IFLAG.EQ.1) GOTO 6                                             FIS00070
      Y=(X**2-3.)/6.                                                    FIS00080
      IC=0                                                              FIS00090
      Y1=1./(Y1-1.)                                                     FIS00100
      Y2=1./(Y2-1.)                                                     FIS00110
      H=2./(Y1+Y2)                                                      FIS00120
      X=X*SQRT(H+Y)/H-(Y1-Y2)*(Y+5./6.-2./(3.*H))                       FIS00130
      X=EXP(2.*X)                                                       FIS00140
      G=1.                                                              FIS00150
      IB1=2                                                             FIS00160
      IF(MOD(N1,2).EQ.0) GO TO 1                                        FIS00170
      G=1.7724539                                                       FIS00180
      IB1=1                                                             FIS00190
1     IB2=2                                                             FIS00200
      IF(MOD(N2,2).EQ.0) GO TO 2                                        FIS00210
      G=G*1.7724539                                                     FIS00220
      IB2=1                                                             FIS00230
2     IB3=2                                                             FIS00240
      IF(MOD(N1+N2,2).EQ.0) GO TO 3                                     FIS00250
      G=G/1.7724539                                                     FIS00260
      IB3=1                                                             FIS00270
3     IF((IB1+IB2).NE.2)  G=2.*G                                        FIS00280
      IF((N1+N2).LE.3) GO TO 5                                          FIS00290
      ND=N1+N2-2-IB3                                                    FIS00300
       I=-2                                                             FIS00310
4      I=I+2                                                           FIS00320
       IF (I.GT.ND) GO TO 5                                            FIS00330
      IF((IB1+I).LE.(N1-2))  G=G*(IB1+I)                                FIS00340
      IF((IB2+I).LE.(N2-2))  G=G*(IB2+I)                                FIS00350
       G=G/(IB3 + I)                                                    FIS00360
       GO TO 4                                                          FIS00370
5     Y2=N2/(N2+N1*X)                                                   FIS00380
      Y1=1.-Y2                                                          FIS00390
      Y=1.+(G*(1.-ALPHA-FISH(X,N1,N2)))/SQRT(Y1**N1*Y2**N2)             FIS00400
      FISHIN=X*Y                                                        FIS00410
      IF(Y.LT.0.)  FISHIN=.5*X                                          FIS00420
      IF(ABS(X/FISHIN-1.).LT.(.5E-6))  GO TO 7                          FIS00430
      IF(ABS(X-FISHIN).LT.(.5E-6))  GO TO 7                             FIS00440
      IC=IC+1                                                           FIS00450
.     IF(IC.GT.100)  RETURN                                             FIS00460
      X=FISHIN                                                          FIS00470
      GO TO 5                                                           FIS00480
6     IFLAG=1                                                           FIS00490
7     RETURN                                                            FIS00500
      END                                                               FIS00510
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: FLDHOR FORTRAN A

```
      SUBROUTINE FLDHOR(ISYM,LINUM,IR,NOFLD,FIELD,FLDSAV,VERTEX,
     *                  NOSUB3,SAMSTR,SAMEND,SAMINC,LININC)
C*
C*    THIS SUBROUTINE SETS THE SYMBOL INDEX IN THE CLASSIFIED LINE (IR)
C*    ARRAY TO OUTLINE TRAINING OR TEST FIELDS IN THE MAP.
C*
      IMPLICIT INTEGER(A-Z)
      DIMENSION IR(1),FIELD(5,NOFLD),FLDSAV(4,NOFLD),VERTEX(1)
      DIMENSION FL(22)
      DO 50 I=1,NOFLD
      IF(LINUM+LININC .LT. FIELD(1,I))GO TO 50
      IF(LINUM-LININC .GT. FIELD(2,I))GO TO 50
      IF(FIELD(3,I) .GT. SAMEND)GO TO 50
      IF(FIELD(4,I) .LT. SAMSTR)GO TO 50
C*
C*    FOUND A FIELD THAT NEEDS BORDER ON THIS LINE. NOW FIND FIELD
C*    INTERSECTIONS ON THIS LINE.
C*
      NV=FLDSAV(4,I)
      IPT=FIELD(5,I)
C*
C*     TOP OR BOTTOM
C*
      IF(LINUM.GT.FIELD(2,I))GO TO 40
      IF(LINUM.LT.FIELD(1,I))GO TO 30
      CALL FDLINT(VERTEX(IPT),NV,FL,LINUM,SAMPS,NI)
      DO 20 J=1,NI,2
      IB =(FL(J)-SAMSTR)/SAMINC
      IE=(FL(J+1)-SAMSTR)/SAMINC + 2
      IF(MOD(SAMSTR,SAMINC).NE.MOD(FL(J),SAMINC))IB=IB+1
      BORNUM=ISYM
      IF(IR(IB).GT.NOSUB3)BORNUM = NOSUB3 + 3
      IR(IB)=BORNUM
      IF(IR(IE).GT.NOSUB3)BORNUM = NOSUB3 + 3
   20 IR(IE)=BORNUM
      GO TO 50
   30 CONTINUE
C*
C*    GET INTERCEPTS FOR TOP LINE IN FIELD
C*
      CALL FDLINT(VERTEX(IPT),NV,FL,FIELD(1,I),SAMPS,NI)
      GO TO 45
C*
C*    GET INTERCEPTS FOR BOTTOM LINE IN FIELD
C*
   40 CALL FDLINT(VERTEX(IPT),NV,FL,FIELD(2,I),SAMPS,NI)
   45 DO 47 J=1,NI,2
      IB = (FL(J)-SAMSTR)/SAMINC
      IE = (FL(J+1)-SAMSTR)/SAMINC + 2
      IF(MOD(SAMSTR,SAMINC).NE.MOD(FL(J),SAMINC))IB=IB+1
      DO 46 IJ=IB,IE
      BORNUM=ISYM
      IF(IR(IJ).GT.NOSUB3)BORNUM=NOSUB3+3
   46 IR(IJ)=BORNUM
   47 CONTINUE
   50 CONTINUE
      RETURN                                                    FLDR0720
      END
```

FILE: LISTSM FORTRAN A

```
      SUBROUTINE LISTSM(TOTALS,TTOL,PCTAB,LISTSW,GTUNIT,                LIS00010
     * GTFILE,AIUNIT,AIFILE,PPUNIT,PPFILE,NAMECT,ALP,                   LIS00020
     * FLDCNT,NCAT,CATNM,SUBCAT,NFLD2,NSUB2)                           LIS00030
      IMPLICIT INTEGER (A-Z)                                            LIS00040
C                                                                       LIS00050
C*****     SUBROUTINE WRITTEN NOV 1978 TO INCLUDE LIST PROCESSING       LIS00060
C****      M AND R ARE ROUNDARY DOT NAMES                               LIS00070
C****      D IS THE DESIGNATED NAME                                     LIS00080
C****      NAMECT IS THE NAME OF THE 'SMALL GRAINS ' CATEGORY           LIS00090
C****      THE CATEGORY CLASSIFIER MUST HAVE BEEN INVOKED               LIS00100
C                                                                       LIS00110
      COMMON /LISTMM/ NPGA(3,2),NAMPGA(209,3,2),LINPGA(209,3,2),        LIS00120
     * SAMPGA(209,3,2),DOTLAB(209,4,2),VPGA(3),IPGA                    LIS00130
C234567A                                                                LIS00140
C     INCLUDE CMARK14                                                   LIS00150
C     INCLUDE COMMK1                                                    LIS00160
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          LIS00170
     * AVAR2,COVAR2,CLSID2,SUBNO2,SURDS2,FLDSV2,VERTX2,                 LIS00180
     * FFTVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),                     LIS00190
     * KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                          LIS00200
     * GRPCHK(61),GROUPS(124)                                           LIS00210
      COMMON/DOTVEC/TYPE,CATNAM(60),NOCAT,TOTVEC,FLDINF(6),             LIS00220
     * PRTKEY,SIZE,LACIE                                                LIS00230
CSEND                                                                   LIS00240
      DIMENSION ALPMSG(3)                                               LIS00250
        DATA SYMTHR/'*'/,SYMDES/' '/, SYMOUT/' '/,                      LIS00260
     * MRCD/'M'/,RRCD/'R'/,DRCD/'D'/,BLANK/' '/,                        LIS00270
     * ALPMSG/'PPC ','GT  ','AI  '/                                     LIS00280
C                                                                       LIS00290
C                                                                       LIS00300
      DIMENSION TTOL(1),PCTAB(1),CATNM(1),SUBCAT(1),                    LIS00310
     * FIELDS(4,250),VERTEX(1000),INFUNT(3),INFFIL(3)                   LIS00320
      REAL TOTALS(1),ALP(2)                                             LIS00330
C                                                                       LIS00340
C                                                                       LIS00350
      REAL PMACH(2),PIXTOT,P11,P12,PB1,PB2,P,CATTOT(61)                 LIS00360
      CONTINUE                                                          LIS00370
      IF (FLDCNT.GT.1) GO TO 400                                        LIS00380
C                                                                       LIS00390
C*****     INITIALIZE IF FIRST FIELD TO BE SUMMARIZED                   LIS00400
C                                                                       LIS00410
      DO 10 I = 1,3                                                     LIS00420
      DO 10 II = 1,2                                                    LIS00430
10    NPGA(I,II) = 0                                                    LIS00440
      DO 20 I = 1,2                                                     LIS00450
      DO 20 II = 1,3                                                    LIS00460
      DO 20 III = 1,209                                                 LIS00470
      NAMPGA(III,II,I) = BLANK                                          LIS00480
      DOTLAB(III,II,I) = BLANK                                          LIS00490
20    CONTINUE                                                          LIS00500
C                                                                       LIS00510
C***    READ IN PPC GT AI FILES                                         LIS00520
C*** ASSUME TYPE 1 AND 2 ON SAME UNIT   BACK-TO-BACK                    LIS00530
C                                                                       LIS00540
      IPGA = 0                                                          LIS00550
      IF (PPUNIT.EQ.0) GO TO 25                                         LIS00560
      IPGA = IPGA + 1                                                   LIS00570
      VPGA(IPGA) = 1                                                    LIS00580
25    IF (GTUNIT.EQ.0) GO TO 30                                         LIS00590
      IPGA = IPGA + 1                                                   LIS00600
      VPGA(IPGA) = 2                                                    LIS00610
30    IF (AIUNIT.EQ.0) GO TO 35                                         LIS00620
      IPGA = IPGA + 1                                                   LIS00630
      VPGA(IPGA) = 3                                                    LIS00640
35    IF (IPGA.EQ.0) CALL CMERR                                         LIS00650
C                                                                       LIS00660
C                                                                       LIS00670
      INFUNT(1) = PPUNIT                                                LIS00680
      INFFIL(1) = PPFILE - 1                                            LIS00690
      INFUNT(2) = GTUNIT                                                LIS00700
      INFFIL(2) = GTFILE - 1                                            LIS00710
      INFUNT(3) = AIUNIT                                                LIS00720
      INFFIL(3) = AIFILE - 1                                            LIS00730
C                                                                       LIS00740
C                                                                       LIS00750
      DO 100 I = 1,IPGA                                                 LIS00760
        ISIT = VPGA(I)                                                  LIS00770
      IPT =0                                                            LIS00780
      STAMNT = 1                                                        LIS00790
```

FILE: LISTSM FORTRAN A

```
      TYPE = 1                                                          LIS00800
      NOCAT = 0                                                         LIS00810
      NOFLD2 = 0                                                        LIS00820
      TOTVT2 = 0                                                        LIS00830
      TOTVEC = 0                                                        LIS00840
      SWCHG = 0                                                         LIS00850
      INIT = 0                                                          LIS00860
  110  CALL LISTLC(FIELDS,STAMNT,6130,6140,6150,SWCHG,                   LIS00870
     *  INIT,INFUNT(ISIT),INFFIL(ISIT),IPT,VERTEX)                      LIS00880
C                                                                       LIS00890
C                                                                       LIS00900
  130 NPGA(ISIT,TYPE) = NOFLD2                                          LIS00910
      NAMPGA(NOFLD2,ISIT,TYPE) = FIELDS(1,NOFLD2)                       LIS00920
      LINPGA(NOFLD2,ISIT,TYPE) = FLDINF(1)                              LIS00930
      SAMPGA(NOFLD2,ISIT,TYPE) = FLDINF(4)                              LIS00940
      I10 = FLDINF(1)/10                                                LIS00950
      II10 = FLDINF(4)/10                                               LIS00960
      J = (I10 - 1)*19 + II10                                           LIS00970
      DOTLAB(J,ISIT,TYPE) = FIELDS(1,NOFLD2)                            LIS00980
      GO TO 110                                                         LIS00990
C                                                                       LIS01000
C***      DOT TYPE CHANGE                                               LIS01010
C                                                                       LIS01020
  140 NOFLD2 = 0                                                        LIS01030
      GO TO 110                                                         LIS01040
C                                                                       LIS01050
C***    SEND CARD IMAGE DETECTED                                        LIS01060
C                                                                       LIS01070
  150 CONTINUE                                                          LIS01080
  100   CONTINUE                                                        LIS01090
C                                                                       LIS01100
C***    PCTAB STORED IN ORDER OF FIRST LINE---SECOND LINE ETC.          LIS01110
C                                                                       LIS01120
  400 NSUB3 = NSUB2 + 1                                                 LIS01130
      NSUB6 = NSUB2 + 4                                                 LIS01140
      NSUB7 = NSUB2 + 5                                                 LIS01150
      DO 430 I = 1,NFLD2                                                LIS01160
      SUBCL = PCTAB(I)                                                  LIS01170
      IF (SUBCL.GT.0) GO TO 405                                         LIS01180
      DOTLAB(I,4,1) = SYMOUT                                            LIS01190
      GO TO 420                                                         LIS01200
  405 IF (SUBCL.LE.NSUB2) GO TO 410                                     LIS01210
      IF (SUBCL.EQ.NSUB3)  DOTLAB(I,4,1) = SYMTHR                       LIS01220
      IF (SUBCL.EQ.NSUB6.OR.SUBCL.EQ.NSUB7) DOTLAB(I,4,1) =SYMDES       LIS01230
      GO TO 420                                                         LIS01240
  410 CAT = SUBCAT(SUBCL)                                               LIS01250
      DOTLAB(I,4,1) = CATNM(CAT)                                        LIS01260
  420 DOTLAB(I,4,2) = DOTLAB(I,4,1)                                     LIS01270
  430 CONTINUE                                                          LIS01280
C                                                                       LIS01290
C***    COMPUTE TOTAL NUMBER OF CLASSIFIED DOTS                         LIS01300
C                                                                       LIS01310
      PIXTOT = 0.                                                       LIS01320
      DO 440 I = 1,NSUB2                                                LIS01330
  440 PIXTOT = PIXTOT + TOTALS(I)                                       LIS01340
C                                                                       LIS01350
C***      COMPUTE TOTAL NO. OF PIXELS IN EACH MACHINE CATEGORY          LIS01360
C***      COUNT UP PIXELS OF CHOSEN AND OTHER CATEGORIES                LIS01370
C                                                                       LIS01380
      PMACH(1) = 0.                                                     LIS01390
      PMACH(2) = 0.                                                     LIS01400
      DO 445 I = 1,61                                                   LIS01410
  445 CATTOT(I) = 0.                                                    LIS01420
      DO 450  I = 1,NSUB2                                               LIS01430
      CAT = SUBCAT(I)                                                   LIS01440
      CATTOT(CAT) = CATTOT(CAT) + TOTALS(I)                             LIS01450
      DUM = CATNM(CAT)                                                  LIS01460
      IF ( DUM.NE.NAMECT) GO TO 448                                     LIS01470
      ICAT = CAT                                                        LIS01480
      PMACH(1) =              CATTOT(CAT)                               LIS01490
      GO TO 450                                                         LIS01500
  448 PMACH(2) =              CATTOT(CAT)                               LIS01510
  450 CONTINUE                                                          LIS01520
C23456789                                                               LIS01530
C***    NCAT = NO. OF MACHINE CLASSIFIED CATEGORIES                     LIS01540
C***    ICAT = CATEGORY NUMBER OF PREFERRED  CATEGORY                   LIS01550
C23456789                                                               LIS01560
      PMACH(1) = PMACH(1)/PIXTOT                                        LIS01570
      PMACH(2) = PMACH(2)/PIXTOT                                        LIS01580
```

FILE: LISTSM FORTRAN A

```
C                                                                       LIS01590
C*** MAJOR LOOP                                                         LIS01600
C                                                                       LIS01610
      DO 600 I = 1,IPGA                                                 LIS01620
      ISIT = VPGA(I)                                                    LIS01630
      IF (NPGA(ISIT,1).EQ.0) GO TO 505                                  LIS01640
      ITYPE = 1                                                         LIS01650
      CALL LISTPR(ISIT,DOTLAB,ITYPE)                                    LIS01660
  505 IF (NPGA(ISIT,2).EQ.0) GO TO 600                                  LIS01670
      ITYPE = 2                                                         LIS01680
      CALL LISTPR(ISIT,DOTLAB(1,1,2),ITYPE)                             LIS01690
C                                                                       LIS01700
C*** COMPUTE N11,N12,N21,N22,NB1,NB2 FOR TYPE 2 DOTS                    LIS01710
C                                                                       LIS01720
      N11 = 0                                                           LIS01730
      N12 = 0                                                           LIS01740
      N21 = 0                                                           LIS01750
      N22 = 0                                                           LIS01760
      NB1 = 0                                                           LIS01770
      NB2 = 0                                                           LIS01780
      DO 550 II = 1,209                                                 LIS01790
      DUM = NAMPGA(II,ISIT,2)                                           LIS01800
      IF (DUM.NE.NAMECT) GO TO 530                                      LIS01810
C                                                                       LIS01820
C*** DOT LABEL IS PREFERRED CATEGORY                                    LIS01830
C                                                                       LIS01840
      DUMS = SAMPGA(II,ISIT,2)/10                                       LIS01850
      DUML = LINPGA(II,ISIT,2)/10                                       LIS01860
      DO 515 III = 1,11                                                 LIS01870
      DO 514 IIII = 1,19                                                LIS01880
      IF(IIII.NE.DUMS) GO TO 514                                        LIS01890
      IF(III.NE.DUML)    GO TO 514                                      LIS01900
C*** FOUND MACHINE CLASSIFIED DOT                                       LIS01910
      J = (III - 1)*19 + IIII                                           LIS01920
      DUMA = DOTLAB(J,4,2)                                              LIS01930
      IF (DUM.EQ.DUMA)  N11 = N11 + 1                                   LIS01940
      IF (DUMA.EQ.SYMDES) GO TO 514                                     LIS01950
C234567890                                                              LIS01960
      IF (DUMA.EQ.SYMTHR) GO TO 514                                     LIS01970
      IF (DUMA.EQ.SYMOUT) GO TO 514                                     LIS01980
      IF (DUM.NE.DUMA) N12 = N12 + 1                                    LIS01990
  514 CONTINUE                                                          LIS02000
  515  CONTINUE                                                         LIS02010
      GO TO 550                                                         LIS02020
C                                                                       LIS02030
C***                                                                    LIS02040
C                                                                       LIS02050
  530 IF (DUM.NE.NBCD.AND.DUM.NE.PBCD) GO TO 550                        LIS02060
C                                                                       LIS02070
C***    THIS PIXEL WAS LABELED BOUNDARY                                 LIS02080
C                                                                       LIS02090
      DUMS = SAMPGA(II,ISIT,2)/10                                       LIS02100
      DUML = LINPGA(II,ISIT,2)/10                                       LIS02110
      DO 535 III = 1,11                                                 LIS02120
      DO 534  IIII = 1,19                                               LIS02130
      IF(IIII.NE.DUMS) GO TO 534                                        LIS02140
      IF (III.NE.DUML) GO TO 534                                        LIS02150
      J = (III - 1)*19 + IIII                                           LIS02160
      DUMA = DOTLAB(J,4,2)                                              LIS02170
      IF (DUMA.EQ.NAMECT) NB1 = NB1 + 1                                 LIS02180
      IF (DUMA.EQ.SYMDES)  GO TO 534                                    LIS02190
      IF (DUMA.EQ.SYMTHR) GO TO 534                                     LIS02200
      IF (DUMA.EQ.SYMOUT) GO TO 534                                     LIS02210
      IF (DUMA.NE.NAMECT) NB2 = NB2 + 1                                 LIS02220
  534 CONTINUE                                                          LIS02230
  535 CONTINUE                                                          LIS02240
      GO TO 550                                                         LIS02250
C                                                                       LIS02260
C***    THIS PIXEL IS LABELED DESIGNATED OR IS IN THE OTHER CATEGORY    LIS02270
C                                                                       LIS02280
  550 IF (DUM.EQ.DBCD) GO TO 580                                        LIS02290
C*** IT IS IN THE OTHER CATEGORY                                        LIS02300
      DUMS = SAMPGA(II,ISIT,2)/10                                       LIS02310
      DUML = LINPGA(II,ISIT,2)/10                                       LIS02320
      DO 555 III = 1,11                                                 LIS02330
      DO 554 IIII = 1,19                                                LIS02340
      IF (IIII.NE.DUMS) GO TO 554                                       LIS02350
      IF (III.NE.DUML)  GO TO 554                                       LIS02360
      J = (III - 1)*19 + IIII                                           LIS02370
```

FILE: LISTSM FORTRAN A

```
      DUMA = OUTLAB(J,4,2)                                                  FIS02360
      IF (DUMA.EQ.NAMECT) N21 = N21 + 1                                     FIS02390
      IF (DUMA.EQ.SYMOES) GO TO 554                                         FIS02400
      IF (DUMA.EQ.SYMTHR) GO TO 554                                         FIS02410
      IF (DUMA.EQ.SYMOUT) GO TO 554                                         FIS02420
      IF (DUMA.NE.NAMECT) N22 = N22 + 1                                     FIS02430
  554 CONTINUE                                                              FIS02440
  555 CONTINUE                                                              FIS02450
  600 CONTINUE                                                              FIS02460
C                                                                           FIS02470
C*** COMPUTE PROPORTION OF PREFERRED CLASS                                  FIS02480
C                                                                           FIS02490
C234567890                                                                  FIS02500
      P11 = FLOAT(N11)/FLOAT(N11 + N21 + NB1)                               FIS02510
      P12 = FLOAT(N12)/FLOAT(N12 + N22 + NB2)                               FIS02520
      PB1 = FLOAT(NB1)/FLOAT(N11 + N21 + NB1)                               FIS02530
      PB2 = FLOAT(NB2)/FLOAT(N12 + N22 + NB2)                               FIS02540
      P = PMACH(1)*(P11 + ALP(1)*PB1)                                       FIS02550
      P = P + PMACH(2)*(P12 + ALP(2)*PB2)                                   FIS02560
C                                                                           FIS02570
      WRITE(6,990)  ALPMSG(ISIT)                                            FIS02580
      WRITE(6,1000)                                                         FIS02590
      WRITE(6,1050) PMACH(1),PMACH(2)                                       FIS02600
      WRITE(6,1010)                                                         FIS02610
      WRITE(6,1020) N11,N12,N21,N22,NB1,NB2                                 FIS02620
      WRITE(6,1030)                                                         FIS02630
      WRITE(6,1040) P11,P12,PB1,PB2,P,ALP(1),ALP(2)                         FIS02640
C                                                                           FIS02650
  990 FORMAT('1',' FOR LIST PROCESSING',' ',A4,'  VS MACHINE CLASS')        FIS02660
 1000 FORMAT('0',' PROPORTION SUMMARY  ')                                   FIS02670
 1010  FORMAT('0','  N11      N12      N21      N22      NB1      NB2')     FIS02680
 1020 FORMAT('0',6(2X,I5))                                                  FIS02690
 1030  FORMAT ('0','  P11      P12      PB1      PB2      P    ALP  ')      FIS02700
 1040 FORMAT('0',7(2X,F6.4))                                                FIS02710
 1050 FORMAT('0','  P(1) = ',F7.4,'     P(2) = ',F7.4)                      FIS02720
  600 CONTINUE                                                              FIS02730
      RETURN                                                                FIS02740
      END                                                                   FIS02750
```

FILE: LISTPR FORTRAN A

```
C THIS SUBROUTINE PRINTS LABEL TABLES                                   LIS00010
      SUBROUTINE LISTPR(ISIT,DOTLAB,ITYPE)                              LIS00020
      IMPLICIT INTEGER (A-Z)                                            LIS00030
      DATA SLASH/'/'/,BLANK/' '/                                        LIS00040
      DIMENSION LINE(57)                                                LIS00050
           DIMENSION DOTLAB(19,11,4)                                    LIS00060
           I1 = ISIT                                                    LIS00070
           I2 = 4                                                       LIS00080
        NPRT = 6                                                        LIS00090
      WRITE(NPRT,10)                                                    LIS00100
   10 FORMAT(1H1)                                                       LIS00110
           IF (ISIT.EQ.1) WRITE(NPRT,120)                               LIS00120
  120          FORMAT(//,50X,'PPTC  VS  CLASSIFIED LABELS ')           LIS00130
           IF (ISIT.EQ.2) WRITE(NPRT,130)                               LIS00140
  130 FORMAT(//,50X,'GROUND TRUTH VS CLASSIFIED LABELS')                LIS00150
           IF (ISIT.EQ.3) WRITE(NPRT,230)                               LIS00160
  230 FORMAT(//,50X,'A, I, LABELS VS CLASSIFIED LABELS')                LIS00170
      WRITE(NPRT,20) ITYPE                                              LIS00180
   20 FORMAT(//,50X,'TYPE ',I,' DOT CLASSIFICATION')                    LIS00190
      WRITE(NPRT,30) (I,I=10,160,10)                                    LIS00200
   30 FORMAT(//,12X,16(5                                                LIS00210
      DO 200 I=1,11                                                     LIS00220
         DO 215 J = 1,57                                                LIS00230
  215    LINE(J) = BLANK                                                LIS00240
      DO 220 J=1,19                                                     LIS00250
      LL=3*(J-1)                                                        LIS00260
      LINE(LL+1)=DOTLAB(J,I,I1)                                         LIS00270
      LINE(LL+3)=DOTLAB(J,I,I2)                                         LIS00280
         IF (DOTLAB(J,I,I1).NE.BLANK)  LINE(LL + 2) = SLASH             LIS00290
       LL2 = LINE(LL + 2)                                               LIS00300
       IF(LL2.NE.SLASH) LINE(LL + 3) = BLANK                            LIS00310
  220  CONTINUE                                                          LIS00320
      I10=I*10                                                          LIS00330
      WRITE(NPRT,300)I10,(LINE(KK),KK=1,57)                             LIS00340
  300 FORMAT(//,3X,I3,19(2X,3A1))                                       LIS00350
  200  CONTINUE                                                         LIS00360
  100  CONTINUE                                                         LIS00370
      RETURN                                                            LIS00380
      END                                                               LIS00390
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: MAPMD FORTRAN A

```
      SUBROUTINE MAPMD(NOCAT,CLSSYM,CATNAM,KATNO,CLSMTX,SUBNO,          MAP00010
     *           SUBDES,CLSVC2,NOCLS2,NOSUM2,THRSKY,THRES)              MAP00020
C                                                                       MAP00030
C                                                                       MAP00040
C     THIS ROUTINE PRINTS THE HEADER INFORMATION FOR THE CLASSIFICATION MAP00050
C     MAP IN DISPLAY                                                    MAP00060
C                                                                       MAP00070
C                                                                       MAP00080
C                                                                       MAP00090
C              NOCAT  -- NO. OF CATEGORIES                              MAP00100
C              CLSSYM -- SYMBOLS FOR CATEGORIES OR SUBCLASSES           MAP00110
C              CATNAM -- CATEGORY NAMES                                 MAP00120
C              KATNO  -- CATEGORY EACH CLASS WAS ASSIGNED TO            MAP00130
C              CLSMTX -- CLASS NAMES                                    MAP00140
C              SUBNO  -- NO. OF SUBCLASSES IN EACH CLASS                MAP00150
C              SUBDES -- SUBCLASS NAMES                                 MAP00160
C              CLSVC2 -- CLASS EACH SUBCLASS WAS ASSIGNED TO (IN COMMON MAP00170
C                        BLOCK INFORM)                                  MAP00180
C                                                                       MAP00190
      IMPLICIT INTEGER (A-Z)                                            MAP00200
C                                                                       MAP00210
      DIMENSION CLSVC2(1)                                               MAP00220
      REAL THRES(1)                                                     MAP00230
C                                                                       MAP00240
      LOGICAL ISWTH                                                     MAP00250
      DIMENSION CLSSYM(1),CATNAM(1),KATNO(1),CLSMTX(1),SUBNO(1),        MAP00260
     *          SUBDES(1)                                               MAP00270
C                                                                       MAP00280
C     PRINTS CATEGORY CLASSIFIER INFORMATION                            MAP00290
C                                                                       MAP00300
      IF (NOCAT .LE. 0) GO TO 82                                        MAP00310
      WRITE(6,200)                                                      MAP00320
  200 FORMAT(//  T42,'MAP OF CATEGORY CLASSIFIER CLASSIFICATION RESULTS'MAP00330
     *   /// T32,'CATEGORY',T82,'CLASS',T93,'SUBCLASS'/ T31,          MAP00340
     *     'NO.',T37,'NAME',              T87,'NO.',T96,'NAME',       MAP00350
     *      T86,'NO.',T94, 'NAME',T101,'SYMBOL')                      MAP00360
      IF(THRESKY.NE.0)WRITE(6,205)                                      MAP00370
  205 FORMAT(1H+,T100,'THRES.')                                         MAP00380
C                                                                       MAP00390
      DO 80   I=1,NOCAT                                                 MAP00400
C                                                                       MAP00410
      WRITE(6,210)I,CATNAM(I)                                           MAP00420
  210 FORMAT(/T31,I2,T37,A4)                                            MAP00430
      ISWTH = .TRUE.                                                    MAP00440
      DO 43   J=1,NOCLS2                                                MAP00450
      IF (KATNO(J) .EQ. I) GO TO 44                                     MAP00460
      GO TO 43                                                          MAP00470
   44 IF (ISWTH) GO TO 45                                               MAP00480
      WRITE(6,220)J,CLSMTX(J)                                           MAP00490
  220 FORMAT(/ T80,I2,T86,A4)                                           MAP00500
      ISWTH = .TRUE.                                                    MAP00510
      GO TO 46                                                          MAP00520
   45 WRITE(6,230)J,CLSMTX(J)                                           MAP00530
  230 FORMAT(1H+,T80,I2,T86,A4)                                         MAP00540
   46 DO 47   K=1,NOSUM2                                                MAP00550
      IF ( CLSVC2(K) .EQ. J ) GO TO 70                                  MAP00560
      GO TO 47                                                          MAP00570
   70 NSUBCL = SUBNO(J)                                                 MAP00580
      KK = 0                                                            MAP00590
      DO 75   L=1,NSUBCL                                                MAP00600
      KK = K + L - 1                                                    MAP00610
      IF (ISWTH) GO TO 72                                               MAP00620
      WRITE(6,250)KK,SUBDES(KK),CLSSYM(KK)                              MAP00630
  250 FORMAT(1H+,T2,T94 ,A4,T103,A1)                                    MAP00640
      GO TO 74                                                          MAP00650
   72 WRITE(6,260)KK,SUBDES(KK),CLSSYM(KK)                              MAP00660
  260 FORMAT(1H+,T86,I2,T94,A4,T103,A1)                                 MAP00670
      ISWTH = .FALSE.                                                   MAP00680
   74 IF(THRESKY.NE.0) WRITE(6,265)THRES(KK)                            MAP00690
  265 FORMAT(1H+,T100,F4.3)                                             MAP00700
   75 CONTINUE                                                          MAP00710
      GO TO 47                                                          MAP00720
   47 CONTINUE                                                          MAP00730
   43 CONTINUE                                                          MAP00740
   80 CONTINUE                                                          MAP00750
   82 RETURN                                                            MAP00760
C                                                                       MAP00770
C     PRINTS STANDARD CLASSIFIER INFORMATION                            MAP00780
C                                                                       MAP00790
```

FILE: MAPHD FORTRAN A

```
 82   CONTINUE                                                                  MAP00800
C                                                                              MAP00810
      WRITE(6,260)                                                             MAP00820
 260  FORMAT( //  T42, 'MAP OF STANDARD CLASSIFIER CLASSIFICATION RESULT'MAP00830
     *5' ///T45,'CLASS',T77,'SUBCLASS'/ T42,'NO.',T50,'NAME',T72,'NO.',      MAP00840
     *T79, 'NAME',T85,'SYMBOL')                                               MAP00850
      IF(THRSKY.NE.0)WRITE(6,265)                                              MAP00860
 265  FORMAT(1H+,T93,'THRES.')                                                 MAP00870
C                                                                              MAP00880
      CLSNUM = 1                                                               MAP00890
      ISWTH = .TRUE.                                                           MAP00900
      DO 89   I=1,NOSUB2                                                       MAP00910
      IF (CLSNUM.EQ. CLSVC2(I))  GO TO 85                                      MAP00920
      CLSNUM = CLSNUM + 1                                                      MAP00930
      GO TO 87                                                                 MAP00940
 85   IF (ISWTH  GO TO 87                                                      MAP00950
      WRITE(6,27 ,SUBDES(I),CLSSYM(I)                                          MAP00960
 270  FORMAT(T72,I2,T77,A4,T87,A1)                                             MAP00970
      GO TO 88                                                                 MAP00980
 87   WRITE(6,280)CLSNUM,CLSMTX(CLSNUM),I,SUBDES(I),CLSSYM(I)                  MAP00990
 280  FORMAT(/ T42,I2,T50,A4,T72,I2,T79,A4,T87,A1)                             MAP01000
      ISWTH = .FALSE.                                                          MAP01010
 88   IF(THRSKY.NE.0)WRITE(6,285)THRES(I)                                      MAP01020
 285  FORMAT(1H+,T93,F6.3)                                                     MAP01030
 89   CONTINUE                                                                 MAP01040
      RETURN                                                                   MAP01050
      END                                                                      MAP01060
```

```
FILE: PCT      FORTRAN  A

      SUBROUTINE PCT(LINUM,IR ,FIELD,VERTEX,FLDSAV,PCTAB,NOFLD,
     *               SAMSTR,SAMEND,SAMINC)
      IMPLICIT INTEGER(A-Z)
C*
C*    THIS SUBROUTINE BUILDS THE PERFORMANCE TABLE FOR DISPLAY, OR THE
C*    HISTOGRAM OF THE QUADRATIC FORM FOR EMPIRICAL THRESHOLDS.
C*       ARGUMENTS:
C*            LINUM  - LINE NO. BEING TESTED.
C*            PTS    - NO. OF POINTS IN IR ARRAY
C*            FIELD  - RECTANGULAR COORDIANTES OF FIELDS (TRAINING OR TEST)
C*                     (5,NOFLD)
C*                     1-LINE START
C*                     2-LINE END
C*                     3-SAMPLE START
C*                     4-SAMPLE END
C*                     5-POINTER TO VERTEX ARRAY FOR VERTICES OF THIS FIELD
C*            VERTEX - ARRAY CONTAINING VERTICES FOR ALL FIELDS
C*            FLDSAV - FIELD INFORMATION
C*                     (4,NOFLD)
C*                     1-FIELD NAME
C*                     2-CLASS NO.
C*                     3-SUBCLASS NO.
C*                     4-NO. OF VERTICES
C*            PCTAB  - PERFORMANCE TABLE
C*                     (NOFLD,NOSUB3))
C*            NOFLD  - NO. OF FIELDS TO TEST
C*            IR     - ARRAY CONTAINING THE SUBCLASS NUMBERS FOR POINTS
C*                     ON THIS LINE.
C*            SAMSTR - BEGINNING SAMPLE NO. OF CLASSIFIED FIELD.
C*            SAMEND - LAST SAMPLE NO. OF CLASSIFIED FILED.
C*            SAMINC - SAMPLE INCREMENT USED IN CLASSIFY
C*
      DIMENSION  FIELD(5,NOFLD),VERTEX(1),FLDSAV(4,NOFLD),
     *           PCTAB(NOFLD,1),IR(1)
      DIMENSION VEC(50),FL(22)
      OPT=1
      GO TO 5
      ENTRY PCTT(LINUM,IR,VR,FIELD,VERTEX,FLDSAV,DSFUNC,NOFLD,
     *           SAMSTR,SAMEND,SAMINC,CON,RANGE)
      REAL DSFUNC(RANGE,50),VR(1),CON(1)
      OPT=2
    5 CONTINUE
C*
C*    FIND NUMBER OF FIELDS THAT THIS LINE INTERSECTS ------------------PCTT0620
C*    ------------------------------------------------------------------PCTT0630
C
      II=0                                                               PCTT0650
      DO 10 I=1,NOFLD                                                    PCTT0660
      IF (LINUM.LT.FIELD(1,I)) GO TO 10                                  PCTT0670
      IF (LINUM.GT.FIELD(2,I)) GO TO 10                                  PCTT0680
      IF (FIELD(3,I).GT.SAMEND) GO TO 10                                 PCTT0690
      IF (FIELD(4,I).LT.SAMSTR) GO TO 10                                 PCTT0700
      II = II+1                                                          PCTT0710
      VEC(II) = I                                                        PCTT0720
   10 CONTINUE                                                           PCTT0730
C     NOW CHECK THE FIELDS OF INTEREST  ( GIVEN BY 'VEC' )               PCTT0740
C     ------------------------------------------------------------------PCTT0750
C                                                                        PCTT0760
      IF (II .EQ. 0)  GO TO 35                                           PCTT0770
      DO 30 J=1,II                                                       PCTT0790
      JJ = VEC (J)                                                       PCTT0800
      NV=FLDSAV(4,JJ)
      IPT=FIELD(5,JJ)
C*    FIND INTERCEPTS FOR THIS FIELD
      CALL FDLINT(VERTEX(IPT),NV,FL,LINUM,SAMPS,NI)
      DO 30 I=1,NI,2
      JB = (FL(I)-SAMSTR)/SAMINC + 1
      JE = (FL(I+1)-SAMSTR)/SAMINC + 1
      IF(MOD(SAMSTR,SAMINC).NE.MOD(FL(I),SAMINC))JB=JB+1
      IF(JB.GT.JE)GO TO 30                                               PCTT0870
      DO 20 K=JB,JE
      K2 = IR(K)
      IF(K2.EQ.0)GO TO 20
      GO TO (15,18),OPT
   15 PCTAB(JJ,K2) = PCTAB(JJ,K2) + 1
      GO TO 20
   18 CONTINUE
```

```
FILE: PCT      FORTRAN  A

      DIVSN = FLDSAV(3,JJ)
C*
C*    WAS THIS PIXEL CLASSIFIED CORRECTLY
C*
      IF(DIVSN.NE.K2)GO TO 20
   21 L=10*(-.5*CON(K2)-VR(K)) + 0.5
      IF (L .LE. 0)  L = 1
      IF ( L .GT. RANGE)  L = RANGE
      DSFUNC(L,DIVSN) = DSFUNC(L,DIVSN) + 1.0
   20 CONTINUE                                                          PCTT0900
   30 CONTINUE                                                          PCTT0910
   35 CONTINUE                                                          PCTT0920
      RETURN                                                            PCTT0930
C                                                                       PCTT0940
CC    GO HOME                                                           PCTT0950
C     ------------------------------------------------------------------PCTT0970
C
       END
```

FILE: PRTPCT FORTRAN A

```
      SUBROUTINE PRTPCT(FLDSAV,PCTAB,NOFLD)                                  PRT00010
C*                                                                           PRT00020
C*    PRTPCT PRINTS THE FOLLOWING CLASSIFICATION PERFORMANCE TABLES          PRT00030
C*                                                                           PRT00040
C*        1. FIELD  BY SUBCLASS     - IF PCFDKY=1                            PRT00050
C*        2. FIELD  BY CLASS        - IF PCFDKY=1                            PRT00060
C*        3. FIELD  BY CATEGROY     - IF PCFDKY=1 AND CATFLG=1               PRT00070
C*        4. CLASS BY SUBCLASS                                               PRT00080
C*        5. CLASS BY CLASS                                                  PRT00090
C*        6. CLASS BY CATEGORY      - IF CATFLG=1                            PRT00100
C*                                                                           PRT00110
      IMPLICIT INTEGER(A-Z)                                                  PRT00120
      REAL PCTT                                                              PRT00130
      REAL PCT                                                               PRT00140
      DIMENSION FLDSAV(4,NOFLD),PCTAB(NOFLD,NOSUB3),TOTSAM(200),BUF(60)      PRT00150
C     INCLUDE COMMON.LIST                                                    PRT00160
C     INCLUDE COMMON.LIST                                                    PRT00170
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,              PRT00180
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,        PRT00190
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                     PRT00200
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                         PRT00210
     *  ,DOTUNT,DOTFIL,MCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                   PRT00220
     *  CRDUNT,PRTUNT,RANDIO                                                 PRT00230
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60),        PRT00240
     *              SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,               PRT00250
     *              PCFDKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,        PRT00260
     *              PLTKEY,BMFLG,RMCOMB,BMFEAT,CDATE(2),                     PRT00270
     *              FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,        PRT00280
     *              THRES(60),SYMMTX(66),HIGH(60),CON(60)                    PRT00290
     *              ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)                  PRT00300
     *              ,NOSUB2,NOTHED,TOTVT2,NOCLS2                             PRT00310
     *              ,KATNO(60),NOCAT,FILTER,MAPFMT                           PRT00320
     *              ,DESKEY,DESUNI,DESOTH,CROP ,ACROP,AOTHER,ATOTAL          PRT00330
     *              ,SITE(6),ANALYS(5),CAM(15),CRPKEY,KEPPTS(60)             PRT00340
     *              ,DOTKEY,DOTERR                                           PRT00350
CSEND                                                                        PRT00360
      DIMENSION TABLE(60,61)                                                 PRT00370
       DATA THR/'THRE'/                                                      PRT00380
       SUBNAM(NOSUB3) = THR                                                  PRT00390
       CLSNAM(NOCLS2+1) = THR                                                PRT00400
       CATNAM(NOCAT+1)=THR                                                   PRT00410
      IF(PCFDKY.NE.1)GO TO 81                                                PRT00420
C*    CLASSIFICATION SUMMARIES BY FIELD                                      PRT00430
C*                                                                           PRT00440
C*    SUBCLASS PERFORMANCE                                                   PRT00450
C*                                                                           PRT00460
      IB=1                                                                   PRT00470
      IE=14                                                                  PRT00480
    5 IF(IE.GT.NOSUB3)IE=NOSUB3                                              PRT00490
       WRITE(6,HEAD)                                                         PRT00500
      IF(TSTKEY.EQ.1)WRITE(6,100)                                            PRT00510
      IF(TSTKEY.NE.1)WRITE(6,200)                                            PRT00520
      IF(FLDKEY.EQ.1)WRITE(6,350)                                            PRT00530
      IF(FLDKEY.NE.1)WRITE(6,351)                                            PRT00540
      WRITE(6,300) (SUBNAM(I),I=IB,IE)                                       PRT00550
      WRITE(6,351)                                                           PRT00560
      PCTT=0.0                                                               PRT00570
      DO 20 J=1,NOFLD                                                        PRT00580
      IS=FLDSAV(3,J)                                                         PRT00590
      IC=FLDSAV(2,J)                                                         PRT00600
      IF(IS.NE.0)NAME=SUBNAM(IS)                                             PRT00610
      IF(IS.EQ.0)NAME=CLSNAM(IC)                                             PRT00620
      TOTSAM(J)=0                                                            PRT00630
      DO 10 K=1,NOSUB3                                                       PRT00640
   10 TOTSAM(J) = TOTSAM(J) + PCTAB(J,K)                                     PRT00650
      IF(FLDKEY.NE.1)GO TO 15                                                PRT00660
      PCT=FLOAT(PCTAB(J,IS))/FLOAT(TOTSAM(J))*100.                           PRT00670
      PCTT=PCTT+PCT                                                          PRT00680
      WRITE(6,400) FLDSAV(1,J),NAME,TOTSAM(J),PCT,(PCTAB(J,K),K=IB,IE)       PRT00690
      GO TO 20                                                               PRT00700
   15 WRITE(6,500) FLDSAV(1,J),NAME,TOTSAM(J),(PCTAB(J,K),K=IB,IE)           PRT00710
   20 CONTINUE                                                               PRT00720
      IF(IE.EQ.NOSUB3)GO TO 30                                               PRT00730
      IB=IE+1                                                                PRT00740
      IE=IE+14                                                               PRT00750
      PCTT=0.0                                                               PRT00760
      GO TO 5                                                                PRT00770
   30 CONTINUE                                                               PRT00780
      PCTT=PCTT/NOFLD                                                        PRT00790
```

FILE: PRTPCT FORTRAN A

```
      IF(FLOKEY.EQ.1)WRITE(6,860)PCTT                                   PRT00800
      PCTT=0.0                                                          PRT00810
      IB=1                                                              PRT00820
      IE=14                                                             PRT00830
C*    NOW FIELD BY CLASS                                                PRT00840
      IN = NOSUB2+1                                                     PRT00850
   32 IF(IE.GT.NOCLS2+1)IE=NOCLS2+1                                     PRT00860
      WRITE(6,HEAD)                                                     PRT00870
      IF(TSTKEY.NE.1)WRITE(6,600)                                       PRT00880
      IF(TSTKEY.EQ.1)WRITE(6,650)                                       PRT00890
      WRITE(6,351)                                                      PRT00900
      WRITE(6,300)(CLSNAM(I),I=IB,IE)                                   PRT00910
      WRITE(6,351)                                                      PRT00920
      DO 50 J=1,NOFLD                                                   PRT00930
C*    CONDENSE A LINE                                                   PRT00940
      DO 33 K=1,IN                                                      PRT00950
   33 BUF(K)=0                                                          PRT00960
      DO 35 K=1,NOSUB2                                                  PRT00970
      IK=CLSSUB(K)                                                      PRT00980
   35 BUF(IK)= BUF(IK) + PCTAB(J,K)                                     PRT00990
      IC=FLDSAV(2,J)                                                    PRT01000
      BUF(NOCLS2+1) =PCTAB(J,NOSUB3)                                    PRT01010
      PCT = FLOAT(BUF(IC))/FLOAT(TOTSAM(J)) * 100.                      PRT01020
      PCTT=PCTT+PCT                                                     PRT01030
      WRITE(6,400) FLDSAV(1,J),CLSNAM(IC),TOTSAM(J),PCT,                PRT01040
     *                  (BUF(K),K=IB,IE)                                PRT01050
   50 CONTINUE                                                          PRT01060
      IF(IE.EQ.NOCLS2+1)GO TO 60                                        PRT01070
      IB=IE+1                                                           PRT01080
      IE=IE+14                                                          PRT01090
      PCTT=0.0                                                          PRT01100
      GO TO 32                                                          PRT01110
   60 CONTINUE                                                          PRT01120
      PCTT=PCTT/NOFLD                                                   PRT01130
      WRITE(6,860)PCTT                                                  PRT01140
      PCTT=0.0                                                          PRT01150
C*                                                                      PRT01160
C*    NOW FIELD BY CATEGORY                                             PRT01170
C*                                                                      PRT01180
      IF(CATFLG.EQ.0)GO TO 81                                           PRT01190
      IB=1                                                              PRT01200
      IE=IE+14                                                          PRT01210
   62 IF(IE.GT.NOCAT+1)IE=NOCAT + 1                                     PRT01220
      WRITE(6,HEAD)                                                     PRT01230
      IF(TSTKEY.NE.1)WRITE(6,700)                                       PRT01240
      IF(TSTKEY.EQ.1)WRITE(6,750)                                       PRT01250
      WRITE(6,352)(CATNAM(I),I=IB,IE)                                   PRT01260
      WRITE(6,351)                                                      PRT01270
      DO 70 J=1,NOFLD                                                   PRT01280
      DO 63 K=1,IN                                                      PRT01290
   63 BUF(K)=0                                                          PRT01300
      DO 65 K=1,NOSUB2                                                  PRT01310
      IK = SUBCAT(K)                                                    PRT01320
   65 BUF(IK) = BUF(IK) + PCTAB(J,K)                                    PRT01330
      IC=FLDSAV(2,J)                                                    PRT01340
      ICAT=KATNO(IC)                                                    PRT01350
      PCT = FLOAT(BUF(ICAT))/FLOAT(TOTSAM(J)) * 100.                    PRT01360
      PCTT=PCTT +PCT                                                    PRT01370
      BUF(NOCAT+1) = PCTAB(J,NOSUB3)                                    PRT01380
      WRITE(6,400) FLDSAV(1,J),CATNAM(ICAT),TOTSAM(J),PCT,              PRT01390
     *                  (BUF(K),K=IB,IE)                                PRT01400
   70 CONTINUE                                                          PRT01410
      IF(IE.EQ.NOCAT+1)GO TO 80                                         PRT01420
      IB=IE+1                                                           PRT01430
      IE=IE+14                                                          PRT01440
      PCTT=0.0                                                          PRT01450
      GO TO 62                                                          PRT01460
   80 CONTINUE                                                          PRT01470
      PCTT=PCTT/NOFLD                                                   PRT01480
      WRITE(6,850)PCTT                                                  PRT01490
C*                                                                      PRT01500
C*    NOW COMPRESS PCTAB TO CLASS BY SUBCLASS                           PRT01510
C*                                                                      PRT01520
C*    ZERO TABLE                                                        PRT01530
   81 CONTINUE                                                          PRT01540
      DO 85 I=1,NOCLS2                                                  PRT01550
      TOTSAM(I)=0                                                       PRT01560
      DO 85 J=1,NOSUB3                                                  PRT01570
   85 TABLE(I,J)=0                                                      PRT01580
```

FILE: PRTPCT FORTRAN A

```
      DO 90 I=1,NOFLD                                                   PRT01590
      IC=FLDSAV(2,I)                                                    PRT01600
      DO 90 J=1,NOSUB3                                                  PRT01610
      TABLE(IC,J)=TABLE(IC,J)+PCTAB(I,J)                                PRT01620
      TOTSAM(IC)=TOTSAM(IC)+PCTAB(I,J)                                  PRT01630
   90 CONTINUE                                                          PRT01640
C*    CLASS BY SUBCLASS                                                 PRT01650
      IB=1                                                              PRT01660
      IE=14                                                             PRT01670
   91 IF(IE.GT.NOSUB3)IE=NOSUB3                                         PRT01680
       WRITE(6,HEAD)                                                    PRT01690
      IF(TSTKEY.NE.1)WRITE(6,800)                                       PRT01700
      IF(TSTKEY.EQ.1)WRITE(6,845)                                       PRT01710
      WRITE(6,810)(SUBNAM(I),I=IB,IE)                                   PRT01720
      WRITE(6,351)                                                      PRT01730
      DO 92 I=1,NOCLS2                                                  PRT01740
      IF(TOTSAM(I).EQ.0)GO TO 92                                        PRT01750
      WRITE(6,820)CLSNAM(I),TOTSAM(I),(TABLE(I,J),J=IB,IE)              PRT01760
   92 CONTINUE                                                          PRT01770
      IB=IB+14                                                          PRT01780
      IE=IE+14                                                          PRT01790
      IF(IE-14 .LT. NOSUB3)GO TO 91                                     PRT01800
C*                                                                      PRT01810
C*    CLASS BY CLASS                                                    PRT01820
C*                                                                      PRT01830
      DO 93 I=1,NOCLS2                                                  PRT01840
      DO 97 J=1,NOCLS2                                                  PRT01850
   97 BUF(J)=0                                                          PRT01860
      DO 94 J=1,NOSUB2                                                  PRT01870
      IC=CLSSUB(J)                                                      PRT01880
   94 BUF(IC)=BUF(IC) + TABLE(I,J)                                      PRT01890
      DO 95 J=1,NOCLS2                                                  PRT01900
   95 TABLE(I,J)=BUF(J)                                                 PRT01910
      TABLE(I,NOCLS2+1)=TABLE(I,NOSUB3)                                 PRT01920
   93 CONTINUE                                                          PRT01930
      PCTT=0                                                            PRT01940
      IB=1                                                              PRT01950
      IE=14                                                             PRT01960
   96 IF(IE.GT.NOCLS2+1)IE=NOCLS2+1                                     PRT01970
       WRITE(6,HEAD)                                                    PRT01980
      IF(TSTKEY.EQ.1)WRITE(6,830)                                       PRT01990
      IF(TSTKEY.NE.1)WRITE(6,835)                                       PRT02000
      WRITE(6,810)(CLSNAM(I),I=IB,IE)                                   PRT02010
      WRITE(6,351)                                                      PRT02020
      NC=0                                                              PRT02030
      DO 107 I=1,NOCLS2                                                 PRT02040
      IF(TOTSAM(I).EQ.0)GO TO 107                                       PRT02050
      NC=NC+1                                                           PRT02060
      PCT=(FLOAT(TABLE(I,I))/FLOAT(TOTSAM(I)))*100.                     PRT02070
      PCTT=PCTT + PCT                                                   PRT02080
      WRITE(6,850)CLSNAM(I),TOTSAM(I),PCT,(TABLE(I,J),J=IB,IE)          PRT02090
  107 CONTINUE                                                          PRT02100
      IF(IE.EQ.NOCLS2+1)GO TO 108                                       PRT02110
      IB=IB+14                                                          PRT02120
      IE=IE+14                                                          PRT02130
      PCTT=0.0                                                          PRT02140
      GO TO 96                                                          PRT02150
  108 CONTINUE                                                          PRT02160
      PCTT=PCTT/NC                                                      PRT02170
      WRITE(6,860) PCTT                                                 PRT02180
C*                                                                      PRT02190
C*    NOW CLASS BY CATEGORY                                             PRT02200
C*                                                                      PRT02210
      IF(CATFLG.EQ.0)RETURN                                             PRT02220
      DO 116 J=1,NOCLS2                                                 PRT02230
      DO 112 I=1,NOCAT                                                  PRT02240
  112 BUF(I)=0                                                          PRT02250
      DO 114 K=1,NOCLS2                                                 PRT02260
      IC=KATNO(K)                                                       PRT02270
  114 BUF(IC)=BUF(IC) + TABLE(J,K)                                      PRT02280
      DO 115 K=1,NOCAT                                                  PRT02290
  115 TABLE(J,K)= BUF(K)                                                PRT02300
      TABLE(J,NOCAT+1)=TABLE(J,NOCLS2+1)                                PRT02310
  116 CONTINUE                                                          PRT02320
      PCTT=0.0                                                          PRT02330
      IB=1                                                              PRT02340
      IE=14                                                             PRT02350
  117 IF(IE.GT.NOCAT+1)IE=NOCAT+1                                       PRT02360
       WRITE(6,HEAD)                                                    PRT02370
```

FILE: PRTPCT FORTRAN A

```
      IF(TSTKEY.EQ.1)WRITE(6,870)                                          PRT02380
      IF(TSTKEY.NE.1)WRITE(6,875)                                          PRT02390
      WRITE(6,810)(CATNAM(I),I=IB ,IE)                                     PRT02400
      WRITE(6,351)                                                         PRT02410
      NC=0                                                                 PRT02420
      DO 118 I=1,NOCLS2                                                    PRT02430
      IF(TOTSAM(I).EQ.0)GO TO 118                                          PRT02440
      NC=NC+1                                                              PRT02450
      IC=KATNO(I)                                                          PRT02460
      PCT=(FLOAT(TABLE(I,IC))/FLOAT(TOTSAM(I)))*100.                       PRT02470
      PCTT=PCTT + PCT                                                      PRT02480
      WRITE(6,850)CLSNAM(I),TOTSAM(I),PCT,(TABLE(I,J),J=IB,IE)             PRT02490
  118 CONTINUE                                                             PRT02500
      IF(IE.GE.NOCAT+1)GO TO 119                                           PRT02510
      IB=IB+14                                                             PRT02520
      IE=IE+14                                                             PRT02530
      PCTT=0.0                                                             PRT02540
      GO TO 117                                                            PRT02550
  119 CONTINUE                                                             PRT02560
      PCTT=PCTT/NC                                                         PRT02570
      WRITE(6,860)PCTT                                                     PRT02580
  100 FORMAT(10X,'CLASSIFICATION SUMMARY - TEST FIELDS BY SUBCLASS')       PRT02590
  200 FORMAT(10X,'CLASSIFICATION SUMMARY - TRAINING FIELDS BY SUBCLASS')   PRT02600
  300 FORMAT(1H+,'FIELD',T17,'TOTAL',T24,'PCT.'/                           PRT02610
     * 1X,'NAME',T10,'CLASS',T17,'PTS.',T24,'CORCT.',2X,14(1X,A6)/)        PRT02620
  351 FORMAT(/)                                                            PRT02630
  352 FORMAT(1X,'FIELD',T17,'TOTAL',T24,'PCT.'/                            PRT02640
     * 1X,'NAME',T10,'CAT.',T17,'PTS.',T24,'CORCT.',2X,14(1X,A6))          PRT02650
  350 FORMAT(/T10,'SUB')                                                   PRT02660
  400 FORMAT(3X,A4,3X,A4,I5,2X,F7.2,2X,14(I5,2X))                          PRT02670
  500 FORMAT(3X,A4,3X,A4,I5,12X,14(I5,2X))                                 PRT02680
  600 FORMAT(10X,'CLASSIFICATION SUMMARY - TRAINING FIELDS BY CLASS')      PRT02690
  650 FORMAT(10X,'CLASSIFICATION SUMMARY - TEST FIELDS BY CLASS')          PRT02700
  700 FORMAT(10X,'CLASSIFICATION SUMMARY - TRAINING FIELDS BY CATEGORY')   PRT02710
  750 FORMAT(10X,'CLASSIFICATION SUMMARY - TEST FIELDS BY CATEGORY')       PRT02720
  800 FORMAT(10X,'CLASSIFICATION SUMMARY - TRAINING CLASS BY SUBCLASS')    PRT02730
  810 FORMAT(/10X,'TOTAL    PCT.'/1X,'CLASS     PTS.    CORCT.'           PRT02740
     * ,5X,14(3X,A4)/)                                                    PRT02750
  820 FORMAT(3X,A4,3X,I5,15X,14(I5,2X))                                    PRT02760
  830 FORMAT(10X,'CLASSIFICATION SUMMARY - TEST CLASS BY CLASS')           PRT02770
  835 FORMAT(10X,'CLASSIFICATION SUMMARY - TRAINING CLASS BY CLASS')       PRT02780
  850 FORMAT(3X,A4,3X,I5,3X,F7.2,5X,14(I5,2X))                             PRT02790
  851 FORMAT(10X,'CLASSIFICATION SUMMARY - TEST CLASS BY SUBCLASS')        PRT02800
  860 FORMAT(//' OVERALL PERFORMANCE =',F10.2)                             PRT02810
  870 FORMAT(10X,'CLASSIFICATION SUMMARY - TEST CLASS BY CATEGORY')        PRT02820
  875 FORMAT(10X,'CLASSIFICATION SUMMARY - TRAINING CLASS BY CATEGORY')    PRT02830
      RETURN                                                               PRT02840
      END                                                                  PRT02850
```

FILE: PRTSUM FORTRAN A

```
      SUBROUTINE PRTSUM(TOTALS,TTOL,FLDESC)                                  PRT00010
C     INCLUDE CMBK10,LIST                                                    PRT00020
C     INCLUDE COMBK6,LIST                                                    PRT00030
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,              PRT00040
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,        PRT00050
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                     PRT00060
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                        PRT00070
     *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                  PRT00080
     *   CROUNT,PRTUNT,RANDIO                                                PRT00090
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60),        PRT00100
     *             SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,                PRT00110
     *             PCFDKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,         PRT00120
     *             PLTKEY,BMFLG,BMCOMB,BMFEAT,CDATE(2),                      PRT00130
     *             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,         PRT00140
     *             THRES(60),SYMMTX(66),HIGH(60),CON(60)                     PRT00150
     *             ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)                   PRT00160
     *             ,NOSUB2,NOTRFD,TOTVT2,NOCLS2                              PRT00170
     *             ,KATNO(60),NOCAT,FILTER,MAPFMT                            PRT00180
     *             ,DESKEY,DESUNI,DESOTH,CROP ,ACROP,AOTHER,ATOTAL           PRT00190
     *             ,SITE(6),ANALYS(5),CAM(15),CRPKEY,KEPPTS(60)              PRT00200
     *             ,DOTKEY,DOTERR                                            PRT00210
C$END                                                                        PRT00220
      DIMENSION TOTALS(66),TTOL(66)                                          PRT00230
      INTEGER G.  CRPTYP,F,H,Z,Y                                             PRT00240
      INTEGER TTOL,DESUNI,DESOTH                                             PRT00250
      INTEGER TIME,CLSSUB,CATFLG,SUBCAT                                      PRT00260
      INTEGER CRPKEY,CROP,CATNAM,CLSNAM,SUBNAM                               PRT00270
      PRT00280
C*                                                                           PRT00290
C*    IF INTENSIVE TEST SITE SUMMARY REPORT IS TO BE PRINTED, FIND A         PRT00300
C*    MATCH ON 'CROP' NAME.                                                  PRT00310
C*                                                                           PRT00320
      CRPTYP=0                                                               PRT00330
      IF(CRPKEY.NE.1)GO TO 10                                                PRT00340
      DO 2 I=1,NOCAT                                                         PRT00350
      IF(CROP.EQ.CATNAM(I))GO TO 6                                           PRT00360
    2 CONTINUE                                                               PRT00370
      DO 3 I=1,NOCLS2                                                        PRT00380
      IF(CROP.EQ.CLSNAM(I))GO TO 7                                           PRT00390
    3 CONTINUE                                                               PRT00400
      DO 4 I=1,NOSUB2                                                        PRT00410
      IF(CROP.EQ.SUBNAM(I))GO TO 8                                           PRT00420
    4 CONTINUE                                                               PRT00430
      WRITE(6,490)CROP                                                       PRT00440
  490 FORMAT(' THE CROP NAME ',A4,' DOES NOT MATCH A CATEGORY,CLASS OR SPRT00450
     *UBCLASS NAME'/' THE INTENSIVE TEST SITE SUMMARY REPORT CANNOT BE PPRT00460
     *RINTED')                                                               PRT00470
      CRPKEY=0                                                               PRT00480
      GO TO 10                                                               PRT00490
C*                                                                           PRT00500
C*    CROP IS A CATEGORY      CRPTYP=1                                       PRT00510
C*    CROP IS A CLASS         CRPTYP=2                                       PRT00520
C*    CROP IS A SUBCLASS      CRPTYP=3                                       PRT00530
C*                                                                           PRT00540
    6 INDEX=I                                                                PRT00550
      CRPTYP=1                                                               PRT00560
      GO TO 10                                                               PRT00570
    7 INDEX=I                                                                PRT00580
      CRPTYP=2                                                               PRT00590
      GO TO 10                                                               PRT00600
    8 INDEX=I                                                                PRT00610
      CRPTYP=3                                                               PRT00620
   10 CONTINUE                                                               PRT00630
      G=0                                                                    PRT00640
      H=0                                                                    PRT00650
      II=0                                                                   PRT00660
      JJ=0                                                                   PRT00670
      JJ=0                                                                   PRT00680
      DO 20 I=1,66                                                           PRT00690
   20 JJ=TOTALS(I) + JJ                                                      PRT00700
C*                                                                           PRT00710
C*    PRINT CLASSIFICATION SUMMARY FOR THIS FIELD                            PRT00720
C*                                                                           PRT00730
      CALL SETMRG(68,4,62)                                                   PRT00740
      WRITE(6,HEAD)                                                          PRT00750
      WRITE(6,260)FLDESC,JJ                                                  PRT00760
      J= JJ - TOTALS(DESUNI)                                                 PRT00770
      KT = TOTALS(DESUNI)                                                    PRT00780
      IF(J .LT. JJ)WRITE(6,265) KT, J                                        PRT00790
      MT=TOTALS(DESOTH)
```

FILE: PRTSUM FORTRAN A

```
      IF(MT.GT.0)WRITE(6,266)MT                                          PRT00800
      WRITE(6,270)                                                       PRT00810
      WRITE(6,275)                                                       PRT00820
      WRITE(6,288)                                                       PRT00830
      WRITE(6,276)                                                       PRT00840
      DO 290 I=1,NOSUB2                                                  PRT00850
      ITTL=TOTALS(I)+TTOL(I)                                             PRT00860
      PCTT=FLOAT(ITTL)/FLOAT(J)*100.                                     PRT00870
      IT=TOTALS(I)                                                       PRT00880
      PIT=TOTALS(I)/FLOAT(J)*100.                                        PRT00890
      PT=TTOL(I)                                                         PRT00900
      PC=PT/FLOAT(ITTL)*100.                                             PRT00910
      PPIT=100. - PC                                                     PRT00920
      PT=PT/FLOAT(J)*100.                                                PRT00930
      WRITE(6,280)SUBNAM(I),ITTL,PCTT,IT,PIT,PPIT,TTOL(I),PT,PC          PRT00940
      IF(CRPTYP.NE.3)GO TO 290                                           PRT00950
      IF(INDEX.EQ.I)GO TO 80                                             PRT00960
      H=H+ITTL                                                           PRT00970
      J1=J1+IT                                                           PRT00980
      GO TO 290                                                          PRT00990
   80 G=G+ITTL                                                           PRT01000
      I1=I1+IT                                                           PRT01010
  290 CONTINUE                                                           PRT01020
      KTTL=TOTALS(NOSUB3)                                                PRT01030
      JTTL = TOTALS(NOSUB3)-TTOL(NOSUB3)                                 PRT01040
      PPCTT=TOTALS(NOSUB3)/FLOAT(J)*100.                                 PRT01050
      WRITE(6,285)JTTL,TTOL(NOSUB3),KTTL,PPCTT                           PRT01060
      TIME=1                                                             PRT01070
      IC=NOCLS2                                                          PRT01080
  295 CONTINUE                                                           PRT01090
      WRITE(6,HEAD)                                                      PRT01100
      WRITE(6,260)FLOFSC,JJ                                              PRT01110
      IF(J.LT.JJ)WRITE(6,265)KT,J                                        PRT01120
      MT=TOTALS(DESOTH)                                                  PRT01130
      IF(MT.GT.0)WRITE(6,266)  MT                                        PRT01140
      WRITE(6,270)                                                       PRT01150
      IF(TIME.EQ.1)WRITE(6,286)                                          PRT01160
      IF(TIME.EQ.2)WRITE(6,287)                                          PRT01170
      WRITE(6,288)                                                       PRT01180
      WRITE(6,276)                                                       PRT01190
      DO 330 IJ=1,IC                                                     PRT01200
      IT=0                                                               PRT01210
      ITT=0                                                              PRT01220
      ITTL=0                                                             PRT01230
      DO 320 I=1,NOSUB2                                                  PRT01240
      IF(TIME.EQ.1)GO TO 315                                             PRT01250
      IF(SUBCAT(I).NE.IJ)GO TO 320                                       PRT01260
      GO TO 316                                                          PRT01270
  315 IF(CLSSUB(I).NE.IJ)GO TO 320                                       PRT01280
  316 IT=TOTALS(I)+IT                                                    PRT01290
      ITTL=TTOL(I)+TOTALS(I)+ITTL                                        PRT01300
      ITT=TTOL(I)+ITT                                                    PRT01310
  320 CONTINUE                                                           PRT01320
      PCTT=FLOAT(ITTL)/FLOAT(J)*100.                                     PRT01330
      PIT=FLOAT(IT)/FLOAT(J)*100.                                        PRT01340
      PT=FLOAT(ITT)/FLOAT(J)*100.                                        PRT01350
      PPIT=FLOAT(IT)/FLOAT(ITTL)*100.                                    PRT01360
      PC=100.-PPIT                                                       PRT01370
      IF(TIME.EQ.2)GO TO 325                                             PRT01380
      WRITE(6,290)CLSNAM(IJ),ITTL,PCTT,IT,PIT,PPIT,ITT,PT,PC             PRT01390
      IF(CRPTYP.NE.2)GO TO 326                                           PRT01400
      IF(INDEX.EQ.IJ)GO TO 323                                           PRT01410
      H=H+ITTL                                                           PRT01420
      J1=J1+IT                                                           PRT01430
      GO TO 326                                                          PRT01440
  323 G=G+ITTL                                                           PRT01450
      I1=I1+IT                                                           PRT01460
      GO TO 326                                                          PRT01470
  325  WRITE(6,290)CATNAM(IJ),ITTL,PCTT,IT,PIT,PPIT,ITT,PT,PC            PRT01480
      IF(CRPTYP.NE.1)GO TO 326                                           PRT01490
      IF(INDEX.EQ.IJ)GO TO 324                                           PRT01500
      H=H+ITTL                                                           PRT01510
      J1=J1+IT                                                           PRT01520
      GO TO 326                                                          PRT01530
  324 G=G+ITTL                                                           PRT01540
      I1=I1+IT                                                           PRT01550
  326 CONTINUE                                                           PRT01560
  330 CONTINUE                                                           PRT01570
      WRITE(6,285)JTTL,TTOL(NOSUB3),KTTL,PPCTT                           PRT01580
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: PRTSUM FORTRAN A

```
      IF(TIME.EQ.2)GO TO 340                                                    PRT01590
      IF(CATFLG.EQ.0)GO TO 340                                                  PRT01600
      IC=NOCAT                                                                  PRT01610
      TIME=2                                                                    PRT01620
      GO TO 295                                                                 PRT01630
  340 CONTINUE                                                                  PRT01640
      IF(CRPKEY.NE.1)RETURN                                                     PRT01650
      WRITE(6,500)CROP,SITE,ANALYS                                              PRT01660
       WRITE(6,505) CAM                                                         PRT01670
      D= ACROP/ATOTAL                                                           PRT01680
      E= AOTHER/ATOTAL                                                          PRT01690
      WRITE(6,510)CROP,ACROP,CROP,D, AOTHER,E, ATOTAL                           PRT01700
      H=H+TOTALS(DESOTH)                                                        PRT01710
      J1=J1+TOTALS(DESOTH)                                                      PRT01720
      Z=TOTALS(DESUN+I)                                                         PRT01730
      Y=JJ                                                                      PRT01740
      F=Y-Z                                                                     PRT01750
      WRITE(6,520) Y,Z,F,CROP,G,H,CROP,I1,J1                                    PRT01760
      K=G-I1                                                                    PRT01770
      L=H-J1                                                                    PRT01780
      M=K+L                                                                     PRT01790
      WRITE(6,530)CROP,K,L,M                                                    PRT01800
      RN=FLOAT(K)/FLOAT(G)                                                      PRT01810
      O =FLOAT(L)/FLOAT(H)                                                      PRT01820
      P =FLOAT(K)/FLOAT(F)                                                      PRT01830
      Q =FLOAT(L)/FLOAT(F)                                                      PRT01840
      WRITE(6,540)CROP,RN,O,CROP,P,Q                                            PRT01850
      R=FLOAT(K+L)/FLOAT(F)                                                     PRT01860
      S=FLOAT(G)/FLOAT(F)                                                       PRT01870
      T=FLOAT(H)/FLOAT(F)                                                       PRT01880
      U=FLOAT(I1)/FLOAT(F)                                                      PRT01890
      V=FLOAT(J1)/FLOAT(F)                                                      PRT01900
      W=FLOAT(H+K)/FLOAT(F)                                                     PRT01910
      WRITE(6,550)R,CROP,S,T,CROP,U,V,W                                         PRT01920
      WRITE(6,610)                                                              PRT01930
      WRITE(6,610)                                                              PRT01940
      WRITE(6,560)                                                              PRT01950
      WRITE(6,575)                                                              PRT01960
      WRITE(6,570)                                                              PRT01970
      WRITE(6,580)                                                              PRT01980
      WRITE(6,570)                                                              PRT01990
      WRITE(6,585)                                                              PRT02000
      WRITE(6,570)                                                              PRT02010
      WRITE(6,590)                                                              PRT02020
      WRITE(6,570)                                                              PRT02030
      WRITE(6,560)                                                              PRT02040
      WRITE(6,595)                                                              PRT02050
      WRITE(6,570)                                                              PRT02060
      WRITE(6,611)                                                              PRT02070
      WRITE(6,570)                                                              PRT02080
      T1=S-D                                                                    PRT02090
      T2=U-D                                                                    PRT02100
      WRITE(6,600) CROP,G,S,T1,I1,U,P,T2                                        PRT02110
      WRITE(6,570)                                                              PRT02120
      WRITE(6,560)                                                              PRT02130
      WRITE(6,605)                                                              PRT02140
      WRITE(6,570)                                                              PRT02150
      WRITE(6,615)                                                              PRT02160
      WRITE(6,570)                                                              PRT02170
      DATA NAME /'CROP'/                                                        PRT02180
      T1=T-E                                                                    PRT02190
      T2=W-E                                                                    PRT02200
      KH=H+K                                                                    PRT02210
      WRITE(6,600)NAME,H,T,T1,KH,W,O,T2                                         PRT02220
      WRITE(6,570)                                                              PRT02230
      WRITE(6,570)                                                              PRT02240
      WRITE(6,560)                                                              PRT02250
  240 FORMAT(//' CLASSIFICATION SUMMARY FOR FIELD ',A6//                        PRT02260
     *  ' TOTAL NUMBER OF SAMPLED POINTS ',I10)                                PRT02270
  245 FORMAT(' LESS DESIGNATED UNIDENTIFIABLE ',I10/T36,7('-')/                 PRT02280
     *   T33,I10)                                                             PRT02290
  246 FORMAT(//' NO. OF PIXELS DESIGNATED OTHER',T33,I10//)                     PRT02300
  500 FORMAT(1H1,T44,'INTENSIVE TEST SITE SUMMARY REPORT FOR ',A4 ///,          PRT02310
     *  T20,'NAME OF INTENSIVE TEST SITE ',6A4,T71,'NAME OF ANALYST ',         PRT02320
     *  5A4/T42,22('-'),T86,18('-'))                                           PRT02330
  505 FORMAT(T20,'PROCEDURE CONFIGURATION',T48 , 15A4/T48,56('-')/)            PRT02340
  510 FORMAT(T15,'GROUND TRUTH FOR INTENSIVE TEST SITE' /T15,36('-')/          PRT02350
     * T20,'ACREAGE OF ',A4, ' A = ',F6.1,T62,                                PRT02360
     * 'TRUE PROPORTION IN ',A4,5X,'A/C = D = ',F4.3 /                        PRT02370
```

```
FILE: PRTSUM   FORTRAN  A

     * T20,'ACREAGE OF OTHER   B = ', F6.1,T62 ,                          PRT02380
     * 'TRUE PROPORTION IN OTHER',4X,'H/C = E = ',F4.3 /                  PRT02390
     * T20,'TOTAL ACREAGE',T39,'C = ',F6.1///)                            PRT02400
  520 FORMAT(T15,'RESULTS OF COMPUTATION FOR INTENSIVE TEST SITE'/         PRT02410
     *   T15,46('-')/                                                     PRT02420
     * T20,'TOTAL NUMBER OF PIXELS IN INTENSIVE TEST SITE',T96,'Y=',I6/   PRT02430
     * T20,'TOTAL NO. OF PIXELS IN EXCLUSION (UNIDENTIFIABLE) AREA',T96,  PRT02440
     *     'Z=',I6 /                                                      PRT02450
     * T20,'TOTAL NUMBER OF PIXELS LESS PIXELS IN EXCLUSION AREA' ,T92,   PRT02460
     *     'Y-Z=F=',I6/                                                   PRT02470
     * T20,'NUMBER OF PIXELS CLASSIFIED AS ',A4,' BEFORE THRESHOLDING',   PRT02480
     * T96,'G=',I6 /                                                      PRT02490
     * T20,'NUMBER OF PIXELS CLASSIFIED AS OTHER BEFORE THRESHOLDING',    PRT02500
     * T96,'H=',I6 /                                                      PRT02510
     * T20,'NUMBER OF PIXELS CLASSIFIED AS ',A4,' AFTER THRESHOLDING',    PRT02520
     * T96,'I=',I6 /                                                      PRT02530
     * T20,'NUMBER OF PIXELS CLASSIFIED AS OTHER AFTER THRESHOLDING',     PRT02540
     * T96,'J=',I6 )                                                      PRT02550
  530 FORMAT(T20,'NUMBER OF PIXELS CLASSIFIED AS ',A4,' WHICH WERE THRES PRT02560
    *HOLDED',T92,'G-I=K=',I6 /                                           PRT02570
     * T20,'NUMBER OF PIXELS CLASSIFIED AS OTHER WHICH WERE THRESHOLDED' PRT02580
    *,T92,'H-J=L=',I6 /                                                  PRT02590
     * T20,'TOTAL NUMBER OF PIXELS THRESHOLDED',T92,'K+L=M=',I6)          PRT02600
  540 FORMAT(T20,'PROPORTION OF ',A4,' PIXELS THRESHOLDED',T92,'K/G=N=',  PRT02610
     * F6.3 /T20,'PROPORTION OF OTHER PIXELS THRESHOLDED',T92,'L/H=O=',   PRT02620
     * F6.3 /T20,'PROPORTION OF ',A4,' PIXELS THRESHOLDED (OF THE TEST S  PRT02630
    *ITE)', T92,'K/F=P=',F6.3 /                                          PRT02640
     *  T20,'PROPORTION OF OTHER PIXELS THRESHOLDED (OF THE TEST SITE)',  PRT02650
     *  T92,'L/F=Q=',F6.3 )                                               PRT02660
  550 FORMAT(T20,'PROPORTION OF PIXELS THRESHOLDED',T88,'(K+L)/F=R=',     PRT02670
     * F6.3 /T20,'PROPORTION OF ',A4,' BEFORE THRESHOLDING',T92,'G/F=S=' PRT02680
    *,F6.3 /T20,'PROPORTION OF OTHER BEFORE THRESHOLDING', T92,'H/F=T=' PRT02690
    *,F6.3 /T20,'PROPORTION OF ',A4,' AFTER THRESHOLDING', T92,'I/F=U=' PRT02700
    *,F6.3 /T20,'PROPORTION OF OTHER AFTER THRESHOLDING', T92, 'J/F=V=' PRT02710
    *,F6.3 /T20,'PROPORTION OF PIXELS TO BE CONSIDERED AS OTHER',T88,     PRT02720
    *'(G+K)/F=W=', F6.3 )                                                PRT02730
  560 FORMAT(T10,101('-') )                                              PRT02740
  570 FORMAT(1H0,T10,'-',T19,'-',T31,'-',T42,'-',T57,'-',T72,'-',T82,    PRT02750
     * '-',T94,'-',T110,'-')                                             PRT02760
  575 FORMAT(T44,'COMPUTED',T96,'COMPUTED')                              PRT02770
  580 FORMAT(T33,'PROPOR.',T44,'PROPOR. LESS',T59,'NO. OF PX.',T74,      PRT02780
     * 'PROPOR.', T96,'PROPOR. LESS' )                                   PRT02790
  585 FORMAT(T2,'NUMBER',T33,'BEFORE',T44,'TRUE PROPOR.', T59,           PRT02800
     * 'CLSFY. AFTER',T74,'AFTER', T84,'PROPOR.', T96,'TRUE PROPOR.')    PRT02810
  590 FORMAT(T2,'CROP',T21,'OF PIXELS',T33,'THRSH.', T44,                PRT02820
     * 'BEFORE THRSH.', T59,'THRESHOLD', T74,'THRSH.', T84,'THRSH.',T96, PRT02830
     * 'AFTER THRSH.')                                                   PRT02840
  595 FORMAT(T25,'(G)', T35,'(S)',T47,'(S-D)', T62,'(I)',                PRT02850
     *  T76,'(U)' ,T86,'(P)',T99,'(U-D)' )                               PRT02860
  600 FORMAT(T2,14,T23,I6,T33,F6.3, T46,F6.3, T61,I6,T74,F6.3,T84,F6.3,  PRT02870
     *  T98,F6.3 )                                                       PRT02880
  605 FORMAT(T25,'(H)', T35,'(T)',T47,'(T-E)', T61,'(H+K)',              PRT02890
     *  T76,'(W)', T86,'(O)', T99,'(W-E)' )                              PRT02900
  610 FORMAT(/)                                                          PRT02910
  611 FORMAT( )                                                          PRT02920
  615 FORMAT(    T2,'OTHER' )                                            PRT02930
  270 FORMAT(/T15,'PTS. BEFORE',T29,'PCT. OF',T44,'PTS. AFTER',          PRT02940
     *   T58,'PCT.OF',T70,'PCT.OF',T87,'PTS.',T99,'PCT.OF',             PRT02950
     *   T111,'PCT. OF'/T17,'THRES.', T29,'TOTAL',T46,'THRES.',         PRT02960
     *   T58,'TOTAL',T87,'THRES.',T99,'TOTAL')                          PRT02970
  275 FORMAT(1H+,T5,'SUBCLASS',T70,'SUBCLASS',T111,'SUBCLASS')           PRT02980
  276 FORMAT(/)                                                          PRT02990
  280 FORMAT(T5,A4,T17,I6,T29,F6.2,T46,I6,T58,                           PRT03000
     *  F6.2,T70,F6.2,T87,I6,T99,F6.2,T111,F6.2)                        PRT03010
  285 FORMAT(// T15,'PTS. THRESHOLDED IN DISPLAY',1X,I10/                PRT03020
     *      T15,'PTS. THRESHOLDED IN CLASSIFY',I10/                     PRT03030
     *   T35,'TOTAL',3X,T10,T60,'PCT. = ',F6.2)                         PRT03040
  286 FORMAT(1H+,T5,'CLASS',T70,'CLASS',T111,'CLASS')                    PRT03050
  287 FORMAT(1H+,T5,'CATEGORY',T70,'CATEGORY',T111,'CATEGORY')           PRT03060
  288 FORMAT(T29,'CLSF.FLD.',T58,'CLSF.FLD.',T99,'CLSF.FLD.',           PRT03070
     *    T111,'THRES.')                                               PRT03080
      RETURN                                                             PRT03090
      END                                                                PRT03100
```

FILE: REDIF3 FORTRAN A

```
      SUBROUTINE REDIF3(TSTSAV,TSTFLD,TSTVER,VDIM,                          RED00010
     *  LISTSW,GTUNIT,GTFILE,AIUNIT,AIFILE,PPUNIT,PPFILE,                  RED00020
     *  NAMFCT,ALP)                                                        RED00030
      IMPLICIT INTEGER (A-H,O-Z)                                            RED00040
C                                                                           RED00050
C*** CODE ADDED TO INCLUDE LIST PROCESSING                                  RED00060
C                                                                           RED00070
      REAL ALP(2)                                                           RED00080
C|-------------------------------------------------------------------------|RED00090
C|-------------------------------------------------------------------------|RED00100
C|                                                                         |RED00110
C|     PURPOSE.. READS AND ANALYZES SUPERVISOR CONTROL CARDS               |RED00120
C|               FOR ' DISPLAY'                                           |RED00130
C|                                                                         |RED00140
C|                                                                         |RED00150
C|-------------------------------------------------------------------------|RED00160
C|-------------------------------------------------------------------------|RED00170
C                                                                           RED00180
C                                                                           RED00190
C                                                                           RED00200
C                                                                           RED00210
      EQUIVALENCE      (HED1(1),HEAD(4)), (DATE(1),HEAD(22)),               RED00220
     1                 (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))            RED00230
C                                                                           RED00240
C     ------------------------------------------------------------------    RED00250
      INCLUDE COMPK6                                                        RED00260
C     INCLUDE CMBK10,LIST                                                   RED00270
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             RED00280
     *              HISFIL,HISKEY,TRFORM,EPIPTP,ERRKEY,MAPUNT,NUFILE,     RED00290
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  RED00300
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                      RED00310
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                RED00320
     *  CRDUNT,PRTUNT,RANDIO                                              RED00330
CSEND                                                                       RED00340
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60),       RED00350
     *             SUBCAT(60),CLSSUB(60),NOMAP,TOTVT3,NOSUB3,             RED00360
     *             PCFDKY,TSTKEY,TRNKEY,THRSKY,STATKY,EMPTRS,THRSVA,      RED00370
     *             PLTKEY,HMFLG,BMCOMB,BMFEAT,CDATE(2),                   RED00380
     *             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,      RED00390
     *             THRES(60),SYMMTX(66),HIGH(60),CON(50)                  RED00400
     *             ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FFTVC2(30)                RED00410
     *             ,NOSUB2,NOTPED,TOTVT2,NOCLS2                           RED00420
     *             ,KATNO(60),NOCAT,FILTER,MAPFMT                         RED00430
     *             ,DESKEY,DESUNI,DESOTH,CROP ,ACROP,AOTHER,ATOTAL        RED00440
     *             ,SITE(6),ANALYS(5),CAM(15),CRPKEY,KEPPTS(60)           RED00450
     *             ,DOTKEY,DOTERR                                         RED00460
      REAL HIGH,CHIN                                                        RED00470
      REAL ACROP,AOTHER,ATOTAL,X                                            RED00480
C                                                                           RED00490
      REAL THRES                                                            RED00500
      DIMENSION TSTSAV(4,200),TSTFLD(5,200),TSTVER(VDIM),                   RED00510
     *          OPT(20),COMVEC(2),INF(7),ACARD(20)                        RED00520
     3,CODBCD(10),EQUCOM(3),CARD2(42),CARD1(90), SLASH(2)                   RED00530
     4 ,        COMENT(15), DATE(3), HED1(15), HED2(15)                    RED00540
C                                                                           RED00550
      DATA OPT/'SYMM','SITE','OPTI','THRE','ANAL',                          RED00560
     *         'COMM','HED1','HED2','DATE','PROC','ACRE',                 RED00570
     *        '*END','FORM','CROP','DO1F',                                RED00580
     *        'GTUN','AIUN','PPUN','NAME','ALPH'/                         RED00590
      DATA OPTNUM/12/,ENDHCD/'END'/,THSUCD/'THRS'/,BLANK/' '/ ,           RED00600
     1  COMMA/','/,CODBCD/' ','S','0','P','N','E','C','T','F'/            RED00610
     2,EQUCOM/2,'=',',','/,COMVEC/1,'.','/, SLASH/1,'/'/                  RED00620
C                                                                           RED00630
      DATA LO/'0'/,LI/'I'/,LE/'E'/,LU/'U'/                                  RED00640
      DATA LL/'L'/,LS/'S'/,LT/'T'/,LC/'C'/,LN/'N'/,OTHER/'OTHE'/,           RED00650
     *UNTDEN/'UNIT'/                                                        RED00660
      DIMENSION SYMMT(66),EQUVEC(2)                                         RED00670
      DATA SYMMT/'1','2','3','4','5','6','7','8','9','A','H','C','D',      RED00680
     1           'E','F','G','M','I','J','K','L','M','N','O','P','Q',      RED00690
     2           'R','S','T','U','V','W','X','Y','Z','-','+','|','/',      RED00700
     3           '=','(',')','$','*',' ','0','.',',','=','(',')','[',']',  RED00710
     4           '&','>','!','?','1','2','3','4','5',' '/                  RED00720
      DATA TRNSYM,TSTSYM,DUPSYM,THRSYM/'0','1','$',' '/                     RED00730
      DATA DESSYM/'2'/,EQUVEC/1,'='/                                        RED00740
      DATA C,L/'C','L'/                                                     RED00750
      LOGICAL*1 L(4),LCHOP(4)                                               RED00760
      EQUIVALENCE  (CARD2(1),CARD1(11)),(M,LP(1)),(LCHOP(1),KCHOP)          RED00770
      DO 5 I=1,66                                                           RED00780
    5 SYMMTX(I)=SYMMT(I)                                                    RED00790
```

FILE: REDIF3 FORTRAN A

```
C                                                                               REDOOH00
C     ------------------------------------------------------------------------REDOOH10
C     ------------------------------------------------------------------------REDOOH20
C                                                                               REDOOH30
C     INIZ                                                                    REDOOH40
C     ------------------------------------------------------------------------REDOOH50
C                                                                               REDOOH60
C                                                                               REDOOH70
C*** CODE ADDED NOV 13, 1978 TO INCLUDE LIST PROCESSING                         REDOOH80
C                                                                               REDOOH90
            LISTSW = 0                                                          RED00900
            GTUNIT = 0                                                          RED00910
            GTFILE = 0                                                          RED00920
            ATUNIT = 0                                                          RED00930
            ATFILE = 0                                                          RED00940
            PRUNIT = 0                                                          RED00950
            PRFILE = 0                                                          RED00960
            NAMECT = LS                                                         RED00970
           ALP(1) = 0.                                                          RED00980
           ALP(2) = 0.                                                          RED00990
      NOTERR=0                                                                  RED01000
      NOTKEY = 0                                                                RED01010
      TSTKEY = 0                                                                RED01020
      TRNKEY = 0                                                                RED01030
      STATKY = 0                                                                RED01040
      PCFDKY = 0                                                                RED01050
      PLTKEY = 0                                                                RED01060
      MAPFMT = 0                                                                RED01070
      NOMAP=1                                                                   RED01080
      FILTER=0                                                                  RED01090
      THRSVA = 0                                                                RED01100
      EMPTRS = 0                                                                RED01110
      SYMCNT = 0                                                                RED01120
      THRSKY=0                                                                  RED01130
      STOPFG = 0                                                                RED01140
      GRPTR = 0                                                                 RED01150
      NOGRP = 0                                                                 RED01160
      NOTREF = 0                                                                RED01170
      TYPE=0                                                                    RED01180
      CRPKEY=0                                                                  RED01190
      THSCNT=0                                                                  RED01200
      DESKEY=0                                                                  RED01210
      TSTCNT = 0                                                                RED01220
      LOGSWT = 0                                                                RED01230
      BCDLKY = 1                                                                RED01240
      SYMMAX = 50                                                               RED01250
      DO 4 I=1,SYMMAX                                                           RED01260
      THRES(I) = 0                                                              RED01270
    4 CONTINUE                                                                  RED01280
      NOFLD3=200                                                                RED01290
C                                                                               RED01300
C     ------------------------------------------------------------------------RED01310
C     READ IN SUPERVISOR CARDS                                                RED01320
C     ------------------------------------------------------------------------RED01330
C                                                                               RED01340
C     SET UP REREAD BUFFER                                                      RED01350
C                                                                               RED01360
      PRUNIT=30                                                                 RED01370
      CALL REREAD(PRUNIT,90)                                                    RED01380
C                                                                               RED01390
C      NOW READ THE CARD INTO THE BUFFER                                        RED01400
C                                                                               RED01410
   10 CONTINUE                                                                  RED01420
   15 READ(21,180)(ACARD(I),I=1,20)                                             RED01430
  180 FORMAT(20A4)                                                              RED01440
      WRITE(PRUNIT,180)(ACARD(I),I=1,20)                                        RED01450
      REWIND PRUNIT                                                             RED01460
      READ(30,104)CODE,CARD2                                                    RED01470
  104 FORMAT(A4,A4,62A1)                                                        RED01480
      REWIND PRUNIT                                                             RED01490
   17 WRITE(4,304)      CODE,CARD2                                              RED01500
  304 FORMAT(T7,A4,6X,62A1)                                                     RED01510
   19 COL = 0                                                                   RED01520
      DO 20 I=1,20                                                              RED01530
      IF(OPT(I).EQ.CODE)GO TO (100,200,300,400,600,710,720,730,                 RED01540
     *    740,760,770,800,750,780,785,210,211,212,230,240), I                  RED01550
   20 CONTINUE                                                                  RED01560
      GOTO 1500                                                                 RED01570
C                                                                               RED01580
```

FILE: REDIF3 FORTRAN A

```
C      GET SYMBOLS                                                              REDO1540
C      ---------------------------------------------------------------------    REDO1600
C                                                                               REDO1610
C                                                                               REDO1620
  100 IF ( SYMCNT .GE. SYMMAX ) GOTO 10                                         REDO1630
      SYMCNT = SYMCNT+1                                                         REDO1640
      SYMMTX(SYMCNT) = BLANK                                                    REDO1650
      M = NXTCHR(CARD2,COL)                                                     REDO1660
      IF ( M .EQ. BLANK ) GO TO 10                                              REDO1670
      IF ( M .EQ. COMMA) GO TO 100                                              REDO1680
      SYMMTX(SYMCNT) = CARD2(COL)                                               REDO1690
  110 M = NXTCHR(CARD2,COL)                                                     REDO1700
      IF ( M .EQ. BLANK ) GO TO 10                                              REDO1710
      IF ( M .NE. COMMA ) GO TO 110                                             REDO1720
      GO TO 100                                                                 REDO1730
C*                                                                              REDO1740
C*    SITE NAME                                                                 REDO1750
C*                                                                              REDO1760
  200 READ(30,201)SITE                                                          REDO1770
  201 FORMAT(10A,6A4)                                                           REDO1780
      REWIND RRUNIT                                                             REDO1790
      GO TO 10                                                                  REDO1800
C                                                                               REDO1810
C***  CODE ADDED NOV 13,1978, TO INCLUDE LIST PROCESSING                        REDO1820
C                                                                               REDO1830
C***   READ GT AI OR PP UNIT AND FILE NUMBERS                                   REDO1840
  210       IPAT = 16                                                           REDO1850
            GO TO 214                                                           REDO1860
  211       IPAT = 17                                                           REDO1870
            GO TO 214                                                           REDO1880
  212       IPAT = 18                                                           REDO1890
  214       M = NXTCHR(CARD2,COL)                                               REDO1900
            IF (M.EQ.BLANK) GO TO 216                                           REDO1910
            IF (M.NE.LU) GO TO 215                                              REDO1920
            M = FIND12(CARD2,COL,EQUVEC)                                        REDO1930
            IF(M.NE.2) GO TO 216                                                REDO1940
            ISTART = 0                                                          REDO1950
            M = NUMBER(CARD2,COL,IPATT,ISTART)                                  REDO1960
            M = FIND12(CARD2,COL,EQUVEC)                                        REDO1970
            IF(M.NE.2) GO TO 216                                                REDO1980
            ISTART = 0                                                          REDO1990
            M = NUMBER(CARD2,COL,IPATTT,ISTART)                                 REDO2000
            GO TO 214                                                           REDO2010
  215       IF(M.NE.LF) GO TO 216                                               REDO2020
            M = FIND12(CARD2,COL,EQUVEC)                                        REDO2030
            IF (M.NE.2) GO TO 216                                               REDO2040
            ISTART = 0                                                          REDO2050
            M = NUMBER(CARD2,COL,IPATTT,ISTART)                                 REDO2060
            M = FIND12(CARD2,COL,EQUVEC)                                        REDO2070
            IF(M.NE.2) GO TO 216                                                REDO2080
            ISTART = 0                                                          REDO2090
            M = NUMBER(CARD2,COL,IPATT,ISTART)                                  REDO2100
            GO TO 214                                                           REDO2110
  216       WRITE(6,217) OPT(IPAT)                                              REDO2120
  217       FORMAT(' ERROR ON ',A4,'  CONTROL CARD ')                           REDO2130
            GO TO 10                                                            REDO2140
  218       IF(IPAT.NE.16)  GO TO 219                                           REDO2150
            GTUNIT = IPATT                                                      REDO2160
            GTFILE = IPATTT                                                     REDO2170
            GO TO 221                                                           REDO2180
  219       IF(IPAT.NE.17) GO TO 220                                            REDO2190
            AIUNIT = IPATT                                                      REDO2200
            AIFILE = IPATTT                                                     REDO2210
            GO TO 221                                                           REDO2220
  220       PPUNIT = IPATT                                                      REDO2230
            PPFILE = IPATTT                                                     REDO2240
  221       LISTSW = 1                                                          REDO2250
            GO TO 10                                                            REDO2260
C***  SELECTED LIST CLASS NAME                                                  REDO2270
  230       NAMECT = NXTCHR(CARD2,COL)                                          REDO2280
            IF (NAMECT.NE.BLANK) GO TO 10                                       REDO2290
            WRITE(6,235)                                                        REDO2300
  235       FORMAT(' NO NAME APPEARS ON SELECTED CATEGORY CARD ')               REDO2310
            NAMECT = LS                                                         REDO2320
            GO TO 10                                                            REDO2330
C***      VALUES FOR ALPHA IN BIAS CORRECTION                                   REDO2340
C                                                                               REDO2350
  240       M = FLTNUM(CARD2,COL,ALP,2)                                         REDO2360
            IF (M.EQ.2) GO TO 10                                                REDO2370
            WRITE(6,245)
```

1244

246

FILE: REDIF3 FORTRAN A

```
 245          FORMAT(' ERROR ON ALPHA CARD. DEFAULTING TO ZERO ')
              ALP(1) = 0.
              ALP(2) = 0.
              GO TO 10
C
C****
C
C
C     SCAN  OPTION CARD
C     -------------------------
C
  300 M = NXTCHR(CARD2,COL)
      DO 310   I=1,9
      IF(M.EQ.CODRCD(I))GO TO (10,340,360,380,385,395,
     *     397,398,399),I
  310 CONTINUE
  315 WRITE(6,3152)  CODE, CARD2
 3152 FORMAT(//// 5X,'**** DSPLAY/REDIF3  ---  ERROR IN  ''OPTION''  CARD
     *RD  ...'//5X,2H''',A4,6X,62A1,2H'''//5X,'****  SCAN OF THIS CARD DIS
     2CONTINUED  ---  PROCEEDING TO NEXT CARD  ****' ///)
      GO TO 10
C     ---------
C
  340 STATKY = 1
      GO TO 390
C     ---------
C
  360 TRNKEY = 1
      GO TO 390
C     ---------
C
  380 M=NXTCHR(CARD2,COL)
      IF(M.EQ.C)PCFDKY=1
      IF(M.EQ.L)PLTKEY=1
      GO TO 390
C     -----
C
  385 NOMAP=0
C     ---------
C
  390 M = FIND12(CARD2,COL,EQUCOM)
      IF ( M .LE. 0 ) GO TO 10
      GO TO 300
C
C-    -- SET THRESHOLD KEYS FOR EMPERICAL THRESHOLDING,TURN OTHERS OFF
  395 THRSVA=0
      EMPTRS=2
      THRSKY=2
      GO TO 390
C     ---------------
C
C-    -- SET THRESHOLD KEYS FOR CHI SQUARE THRESHOLDING,TURN OTHERS OFF
  397 EMPTRS=0
      THRSVA=0
      THRSKY=1
      GO TO 390
C     --------------
C
C-    -- SET THRESHOLD KEYS FOR INPUT-VALUE THRESHOLDING,TURN OTHERS OFF
  398 EMPTRS=0
      THRSVA=3
      THRSKY=3
      GO TO 390
C-    CHECK FOR FISHER OR FILTER
  399 COL=COL+1
      M=NXTCHR(CARD2,COL)
      IF(M.EQ.LL)GO TO 3990
      IF(M.EQ.LS)GO TO 3991
      GO TO 315
 3990 FILTER=1
      GO TO 390
C-
 3991 IX=0
      DO 3995 I=1,NOSUB2
      IF(KEPPTS(I).GT.NOFFET2)GO TO 3995
      IF(IX.NE.0)GO TO 3993
```

FILE: REDIF3 FORTRAN A

```
      WRITE(6,3992)                                                     RED03170
 3992 FORMAT(1H0,////)                                                  RED03180
      IX=1                                                              RED03190
 3993 WRITE(6,3994)  I , KEPPTS(I) , NOFET?                             RED03200
 3994 FORMAT(' *****   FISHER THRESHOLD REQUESTED-NOT PERFORMED',/,AX,  RED03210
     1'*** NO. SAMPLES FOR SUBCLASS',I6,'(=',I6,') IS LESS THAN OR EQUAL RED03220
     2 TO NUMBER OF CHANNELS (=', I6,')',/)                             RED03230
 3995 CONTINUE                                                          RED03240
      IF(IX.EQ.0)GO TO 3996                                             RED03250
      WRITE(6,3992)                                                     RED03260
      GO TO 390                                                         RED03270
C-                                                                      RED03280
C-    -- SET THRESHOLD KEYS FOR  FISHER   THRESHOLDING,TURN OTHERS OFF  RED03290
 3996 EMPTRS=0                                                          RED03300
      THRSVA=0                                                          RED03310
      THRSKY=4                                                          RED03320
C-                                                                      RED03330
C-                                                                      RED03340
      GO TO 390                                                         RED03350
C                                                                       RED03360
C     READ IN THRESHOLDS                                                RED03370
C     ------------------                                                RED03380
C                                                                       RED03390
  400 CONTINUE                                                          RED03400
      I = SYMMAX-THSCNT                                                 RED03410
      THSCNT = THSCNT+FLTNUM(CARD2,COL,THRES(THSCNT+1),I)               RED03420
      GO TO 10                                                          RED03430
C                                                                       RED03440
C*                                                                      RED03450
C*    ANALYST NAME                                                      RED03460
C*                                                                      RED03470
  600 READ(30,601)ANALYS                                                RED03480
  601 FORMAT(10X,5A4)                                                   RED03490
      REWIND RRUNIT                                                     RED03500
      GO TO 10                                                          RED03510
C                                                                       RED03520
C     COMMENT                                                           RED03530
C     -------                                                           RED03540
C                                                                       RED03550
  710 READ (30,9998) COMENT                                             RED03560
 9998 FORMAT (10X,15A4)                                                 RED03570
      REWIND RRUNIT                                                     RED03580
      GOTO 10                                                           RED03590
C                                                                       RED03600
C     HED1                                                              RED03610
C     ----                                                              RED03620
C                                                                       RED03630
  720 READ (30,9998) HED1                                               RED03640
      REWIND RRUNIT                                                     RED03650
      GOTO 10                                                           RED03660
C                                                                       RED03670
C     HED2                                                              RED03680
C     ----                                                              RED03690
C                                                                       RED03700
  730 READ (30,9998) HED2                                               RED03710
      REWIND RRUNIT                                                     RED03720
      GOTO 10                                                           RED03730
C                                                                       RED03740
C     DATE                                                              RED03750
C     ----                                                              RED03760
C                                                                       RED03770
  740 READ (30,9998) DATE                                               RED03780
      REWIND RRUNIT                                                     RED03790
      GO TO 10                                                          RED03800
C                                                                       RED03810
C     MAP TAPE FORMAT                                                   RED03820
C     ---------------                                                   RED03830
  750 M = MATCHR(CARD2,COL)                                             RED03840
      IF(M.EQ.LU)MAPFMT=1                                               RED03850
      IF(M.EQ.LL)MAPFMT=2                                               RED03860
      IF(M.EQ.BLANK)MAPFMT=1                                            RED03870
      GO TO 10                                                          RED03880
C                                                                       RED03890
C                                                                       RED03900
C*                                                                      RED03910
C*    PROCEDURE CONFIGURATION TITLE                                     RED03920
C*                                                                      RED03930
C*                                                                      RED03940
  760 READ(30,9998)CAMS                                                 RED03950
      REWIND RRUNIT
```

FILE: REDIF3 FORTRAN A

```
      GO TO 10                                                          RED03960
C*                                                                      RED03970
C*    ACREAGE                                                           RED03980
C*                                                                      RED03990
  770 M=NXTCHR(CARD2,COL)                                               RED04000
      IF(M.EQ.BLANK)GO TO 10                                            RED04010
      IF(M.EQ.COMMA)GO TO 770                                           RED04020
      J=FIND12(CARD2,COL,EQUCOM)                                        RED04030
      IF(J.EQ.2)GO TO 773                                               RED04040
      WRITE(6,772)                                                      RED04050
  772 FORMAT(' *ERROR IN ACREAGE CARD - CARD IGNORED')                  RED04060
      GO TO 10                                                          RED04070
  773 J = FLTNUM(CARD2,COL,X,1)                                         RED04080
      IF(M.EQ.LT)ATOTAL=X                                               RED04090
      IF(M.EQ.LC)ACROP=X                                                RED04100
      IF(M.EQ.LO)AOTHER=X                                               RED04110
      GO TO 770                                                         RED04120
C*                                                                      RED04130
C*    INTENSIVE STUDY CROP NAME                                         RED04140
C*                                                                      RED04150
  780 M=NXTCHR(CARD2,COL)                                               RED04160
      IC=1                                                              RED04170
      KCROP=BLANK                                                       RED04180
      CRPKEY=1                                                          RED04190
  781 LCROP(IC)=LM(1)                                                   RED04200
      CROP=KCROP                                                        RED04210
      IC=IC+1                                                           RED04220
      IF(IC.GT.4)GO TO 10                                               RED04230
      M=NXTCHR(CARD2,COL)                                               RED04240
      IF(M.EQ.BLANK)GO TO 10                                            RED04250
      GO TO 781                                                         RED04260
C*                                                                      RED04270
C*    DOT DATA PROCESSING                                               RED04280
C*                                                                      RED04290
C*                                                                      RED04300
C*    DOTFILE'S USER ASSIGNED UNIT AND/OR FILE -- DEFAULT UNIT=8,DEFAULTRED04310
C*    FILE = 1  --  IF CLASSIFICATION NOT BY CATEGORY, TURN OFF DOT     RED04320
C*     PROCESSING .                                                     RED04330
C*                                                                      RED04340
C*                                                                      RED04350
C     DOTFILE CARD                                                      RED04360
  785 CONTINUE                                                          RED04370
      IF(NOCAT.EQ.0) GO TO 794                                          RED04380
      DOTKEY=1                                                          RED04390
      M=NXTCHR(CARD2,COL)                                               RED04400
  786 IF(M.EQ.BLANK) GO TO 794                                          RED04410
      IF(M.NE.LU)GO TO 788                                              RED04420
      M=FIND12(CARD2,COL,EQUVEC)                                        RED04430
      IF(M.NE.2)GO TO 794                                               RED04440
      ISTART=0                                                          RED04450
      M=NUMBER(CARD2,COL,DOTUNT,ISTART)                                 RED04460
      M=FIND12(CARD2,COL,EQUVEC)                                        RED04470
      IF(M.NE.2) GO TO 794                                              RED04480
      ISTART=0                                                          RED04490
      M=NUMBER(CARD2,COL,DOTFIL,ISTART)                                 RED04500
      DOTFIL=DOTFIL-1                                                   RED04510
      IF(DOTFIL.LT.0) DOTFIL=0                                          RED04520
      GO TO 10                                                          RED04530
  788 IF(M.NE.LI) GO TO 789                                             RED04540
      M=FIND12(CARD2,COL,SLASH)                                         RED04550
      IF(M.NE.2) GO TO 794                                              RED04560
      M=NXTCHR(CARD2,COL)                                               RED04570
      IF(M.EQ.LF) GO TO 790                                             RED04580
      IF(M.EQ.LU) GO TO 786                                             RED04590
      GO TO 794                                                         RED04600
  789 IF(M.NE.LF) GO TO 794                                             RED04610
  790 M=FIND12(CARD2,COL,EQUVEC)                                        RED04620
      IF(M.NE.2) GO TO 794                                              RED04630
      ISTART=0                                                          RED04640
      M=NUMBER(CARD2,COL,DOTFIL,ISTART)                                 RED04650
      DOTFIL=DOTFIL-1                                                   RED04660
      M=FIND12(CARD2,COL,EQUVEC)                                        RED04670
      IF(M.NE.2) GO TO 794                                              RED04680
      ISTART=0                                                          RED04690
      M=NUMBER(CARD2,COL,DOTUNT,ISTART)                                 RED04700
      GO TO 10                                                          RED04710
  794 DOTERR=1                                                          RED04720
      TRNKEY=0                                                          RED04730
      WRITE(6,4000)                                                     RED04740
 4000 FORMAT(//5X,' ERROR ON DOTFIL CARD')
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: REDIF3 FORTRAN A

```
      GO TO 10                                                          RED04750
C*                                                                      RED04760
C*                                                                      RED04770
  800 CONTINUE                                                          RED04780
C*                                                                      RED04790
C*    *END*    END OF CONTROL CARDS - NOW FIND CHI-SQUARE THRESHOLDS,   RED04800
C*             THEN READ IN TEST FIELDS                                 RED04810
      IF (THSCNT .EQ. 0)  GO TO 830                                     RED04820
C*    IF THRESHOLDS WERE INPUT WITHOUT OPTION - ASSUME CHI-SQUARE       RED04830
      IF(THRSKY.EQ.0)THRSKY=1                                           RED04840
      NDEGR = NOFET2                                                    RED04850
      IF (BMFLG .GT. 0) NDEGR = BMCOMB                                  RED04860
C-                                                                      RED04870
C-    BYPASS IF NOT EQUAL TO 1 OR 2.                                    RED04880
C-                                                                      RED04890
      IF(THRSKY.GT.2) GO TO 830                                         RED04900
C-                                                                      RED04910
      DO 825 MZ=1,NOSUR2                                                RED04920
      HIGH(MZ) = 1. - THRES(MZ)                                         RED04930
      THRES(MZ) = CHIN(HIGH(MZ),NDEGR,FLAG)                             RED04940
      IF(FLAG.EQ.1) GOTO 826                                            RED04950
      GO TO 825                                                         RED04960
  826 WRITE(6,827) MZ,THRES(MZ)                                         RED04970
  827 FORMAT(// ' *** A THRESHOLD VALUE IS OUTSIDE THE ALLOWABLE RANGE OF RED04980
     * - 1. THEREFORE'/'       NO THRESHOLDING HAS BEEN DONE IN THIS RUN RED04990
     3***' ,I5,F15.5//)                                                 RED05000
      THRSKY = 0                                                        RED05010
      GO TO 830                                                         RED05020
  825 CONTINUE                                                          RED05030
  830 CONTINUE                                                          RED05040
C*                                                                      RED05050
C*    NOW READ TEST OR DESIGNATED FIELDS                                RED05060
C*                                                                      RED05070
      TOTVT3=0                                                          RED05080
      NOFLD3=1                                                          RED05090
      IPT=1                                                             RED05100
  840 ICK=LAREAD(TSTSAV(1,NOFLD3),TSTVER(IPT),INF,NV)                   RED05110
      IF(ICK.EQ.-3)GO TO 865                                            RED05120
      IF(ICK.EQ.-2)GO TO 850                                            RED05130
      IF(ICK.EQ.-1)GO TO 860                                            RED05140
      IF(ICK.EQ. 0)GO TO 870                                            RED05150
      TSTSAV(2,NOFLD3)=CLSIND                                           RED05160
      TSTSAV(3,NOFLD3)=SUBIND                                           RED05170
      TSTSAV(4,NOFLD3)=NV                                               RED05180
      TSTFLD(1,NOFLD3)=INF(1)                                           RED05190
      TSTFLD(2,NOFLD3)=INF(2)                                           RED05200
      TSTFLD(3,NOFLD3)=INF(4)                                           RED05210
      TSTFLD(4,NOFLD3)=INF(5)                                           RED05220
      TSTFLD(5,NOFLD3)=IPT                                              RED05230
      IPT=IPT + 2*NV                                                    RED05240
      NOFLD3 = NOFLD3 + 1                                               RED05250
      TOTVT3=TOTVT3+NV                                                  RED05260
        GO TO 840                                                       RED05270
C*    SUBCLASS NAME                                                     RED05280
C*                                                                      RED05290
  850 READ(30,851)NAME                                                  RED05300
      REWIND MRUNIT                                                     RED05310
      IF(TYPE.EQ.2)GO TO 1400                                           RED05320
      TYPE=1                                                            RED05330
  851 FORMAT(10X,A4)                                                    RED05340
      DO 852 I=1,NOSUB2                                                 RED05350
      IF(NAME.EQ.SUBNAM(I))GO TO 854                                    RED05360
  852 CONTINUE                                                          RED05370
      WRITE(6,853)NAME                                                  RED05380
  853 FORMAT(' *ERROR ON SUBCLASS NAME CARD -',A4,' DOES NOT MATCH A SUBRED05390
     *CLASS FROM THE MAPTAP FILE *')                                    RED05400
      CALL CMERR                                                        RED05410
  854 SUBIND=I                                                          RED05420
      CLSIND=CLSSUB(I)                                                  RED05430
      GO TO 840                                                         RED05440
C*                                                                      RED05450
C*    CLASSNAME CARD                                                    RED05460
C*                                                                      RED05470
  860 READ(30,851)NAME                                                  RED05480
      REWIND MRUNIT                                                     RED05490
      IF(TYPE.EQ.2)GO TO 1400                                           RED05500
      TYPE=1                                                            RED05510
      DO 861 I=1,NOCLS2                                                 RED05520
      IF(NAME.EQ.CLSNAM(I))GO TO 863                                    RED05530
```

FILE: REDIF3 FORTRAN A

```
  841 CONTINUE                                                                  RED05540
      WRITE(6,842)NAME                                                          RED05550
  842 FORMAT(' *ERROR ON CLASSNAME CARD -',A4,' DOES NOT MATCH A CLASS NA       RED05560
     *AME FROM THE MAPTAP FILE *')                                              RED05570
      CALL CMERR                                                                RED05580
  843 CLSIND=I                                                                  RED05590
      SUBIND=0                                                                  RED05600
      GO TO 840                                                                 RED05610
C*                                                                              RED05620
C*    DESIGNATED FIELDS                                                         RED05630
C*                                                                              RED05640
  845 READ(30,851)TEST                                                          RED05650
      REWIND RRUNIT                                                             RED05660
      IF(TYPE.EQ.1)GO TO 1400                                                   RED05670
      TYPE=2                                                                    RED05680
      SUBIND=1                                                                  RED05690
        CLSIND = NOSUB3 + 4                                                     RED05700
      IF(TEST.EQ.OTHER)SUBIND=2                                                 RED05710
      IF(TEST.EQ.OTHER)CLSIND=NOSUB3+5                                          RED05720
C***                                                                            RED05730
C***                                                                            RED05740
C***  THIS CODE ADDED AUG 31,1978 TO ALLOW CLASSNAME TO APPEAR                  RED05750
C***  ON DESIGNATED CARD THIS FORCES RECLASSIFICATION OF                        RED05760
C***  DESIGNATED PIXELS INTO THE FIRST SUBCLASS ASSIGNED                        RED05770
C***  TO THE CLASS NAMED ON THE DESIG CARD STARTING AT COL 11                   RED05780
C***                                                                            RED05790
C***                                                                            RED05800
          IF (TEST.EQ.OTHER.OR.TEST.EQ.UNIDEN) GO TO 840                        RED05810
          DO 866 I = 1,60                                                       RED05820
          IF (TEST.NE.CLSNAM(I)) GO TO 866                                      RED05830
          II = I                                                                RED05840
          GO TO 867                                                             RED05850
  866     CONTINUE                                                              RED05860
          WRITE(PRTUNT,4005) TEST                                               RED05870
 4005     FORMAT(' DESIGNATED FIELD OF CLASSNAME',A4,'DOES NOT                  RED05880
     *  MATCH A CLASSNAME ON MAPTAP---DEFAULTING TO UNIDEN')                    RED05890
          GO TO 840                                                             RED05900
  867     DO 868 I =1,60                                                        RED05910
          IF (II.NE.CLSSUB(I)) GO TO 868                                        RED05920
          SUBIND = 0                                                            RED05930
          CLSIND = I                                                            RED05940
          GO TO 869                                                             RED05950
  868     CONTINUE                                                              RED05960
          WRITE(PRTUNT,4006) TEST                                               RED05970
 4006     FORMAT(' DESIGNATED FIELD',A4,'CANNOT BE MATCHED                      RED05980
     *   DEFAULTING TO UNIDENTIFIABLE')                                         RED05990
  869     GO TO 840                                                             RED06000
C*                                                                              RED06010
C*    $END*  - END OF TEST OR DESIGNATED FIELDS                                 RED06020
C*                                                                              RED06030
  870 NOFLD3 = NOFLD3-1                                                         RED06040
      IF(TYPE.EQ.1 .AND. NOFLD3.GT.0)TSTKEY=1                                   RED06050
C*                                                                              RED06060
C*  NEITHER TEST NOR TRAINING FIELDS ARE ALLOWED WITH DOT DATA                  RED06070
C*   PROCESSING  --  THE PERFORMANCE TABLE , PCTAB , IS UTILIZED FOR            RED06080
C*  DOT DATA CLASSIFICATION  PERFORMANCE SUMMARIES                              RED06090
C*                                                                              RED06100
      IF ( DOTKEY .GT. 0  .OR.  DOTERR .GT. 0 )  TSTKEY = 0                     RED06110
C                                                                               RED06120
C*** CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING                          RED06130
C                                                                               RED06140
              IF (LISTSW.EQ.0) GO TO 900                                        RED06150
              TSTKEY = 0                                                        RED06160
              DOTKEY = 0                                                        RED06170
              TRNKEY = 0                                                        RED06180
              DOTERR = 0                                                        RED06190
  900         CONTINUE                                                          RED06200
      IF(TYPE.EQ.2 .AND. NOFLD3.GT.0)DESKEY=1                                   RED06210
C*                                                                              RED06220
C*    SET THRESHOLD AND OUTLINE SYMMBOLS                                        RED06230
C*                                                                              RED06240
      SYMMTX(NOSUB3)=THRSYM                                                     RED06250
      SYMMTX(NOSUB3+1)=TRNSYM                                                   RED06260
      SYMMTX(NOSUB3+2)=TSTSYM                                                   RED06270
      SYMMTX(NOSUB3+3)=DUPSYM                                                   RED06280
      SYMMTX(NOSUB3+4)=DESSYM                                                   RED06290
      SYMMTX(NOSUB3+5)=DESSYM                                                   RED06300
C                                                                               RED06310
C     GO HOME                                                                   RED06320
```

FILE: REDIF3 FORTRAN A

```
C     ------------------------------------------------------------------REDO6330
C                                                                       RED06340
C     RETURN                                                            RED06350
C                                                                       RED06360
C     ERROR ROUTINES                                                    RED06370
C     ------------------------------------------------------------------RED06380
C                                                                       RED06390
 1400 WRITE(6,14000)                                                    RED06400
14000 FORMAT(' TEST FIELDS AND DESIGNATED FIELDS CANNOT BE INPUT TOGETHERED06410
     *R'/)                                                              RED06420
 1500 WRITE (6,15002) CODE, CARD2                                       RED06430
15002 FORMAT(/1X,A4,6X,62A1/' INVALID CONTROL CARD-CHECK SPELLING OF KEYRED06440
     * WORD')                                                          RED06450
      GO TO 10                                                          RED06460
       END                                                              RED06470
```

FILE: RNORM FORTRAN A

```
      FUNCTION RNORM(X)                                                 RN000010
      DIMENSION A(7)                                                    RN000020
      DATA A /.43063BE-4,.2765672E-3,.1520143E-3,                       RN000030
     *        .9270527E-2,.422820E-1,.70523E-1,1.0/                     RN000040
      Y=ABS(X)/1.414213                                                 RN000050
      RNORM=0.                                                          RN000060
      DO 1 I=1,7                                                        RN000070
      RNORM=RNORM*Y+A(I)                                                RN000080
      CALL OVERFL(INDCT)                                                RN000090
      IF(INDCT.NE.2) GO TO 3                                            RN000100
    1 CONTINUE                                                          RN000110
      RNORM=.5*(((((1./RNORM)**2)**2)**2)**2)                           RN000120
    2 IF(X.GT.0.0) RNORM = 1.0-RNORM                                    RN000130
      RETURN                                                            RN000140
    3 RNORM=0.0                                                         RN000150
      GO TO 2                                                           RN000160
      END                                                               RN000170
```

```
FILE: SETUP3   FORTRAN  A

      SUBROUTINE SETUP3(ARRAY,TOP,LISTSW,GTUNIT,GTFILE,                 SET00010
     *     AIUNIT,AIFILE,PPUNIT,PPFILE,NAMFCT,ALP)                      SET00020
      IMPLICIT INTEGER (A-Z)                                            SET00030
C                                                                       SET00040
C***  CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING                 SET00050
C                                                                       SET00060
      REAL ALP(2)                                                       SET00070
      LOGICAL OKAY                                                      SET00080
      INCLUDE COM3K6,LIST                                               SET00090
C     INCLUDE CMBK10,LIST                                               SET00100
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,         SET00110
     *              HISFIL,HISKEY,TRFORM,FRIPTP,ERPKEY,MAPUNT,NOFILE,   SET00120
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               SET00130
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                    SET00140
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,RMTRFL,HISTFL,PCHUNT,              SET00150
     *  CRDUNT,PRTUNT,RANDIO                                            SET00160
      COMMON/DISPL/CATFLG,CATNAM(61),CLSNAM(61),SUBNAM(61),SUBNO(60),   SET00170
     *             SUBCAT(60),CLSSUB(60),NUMAP,TOTVT3,NOSUB3,           SET00180
     *             PCFDKY,TSTKEY,TMNKEY,TMPSKY,STATKY,EMPTRS,THRSVA,    SET00190
     *             PLTKEY,RMFLG,RMCOMM,RMFEAT,CDATE(2),                 SET00200
     *             FLDSV2,FIELD2,VERTX2,FLDSV3,FIELD3,VERTX3,PCTID3,    SET00210
     *             TMKES(60),SYMMTX(66),HIGH(60),CON(50)                SET00220
     *             ,FLDKEY,NOFLD2,NOFLD3,NOFET2,FETVC2(30)              SET00230
     *             ,NOSUB2,NOTHED,TOTVT2,NOCLS2                         SET00240
     *             ,KATNO(60),NOCAT,FILTER,MAPFMT                       SET00250
     *             ,DESKEY,DESUNI,DESOTH,CROP ,ACROP,AOTHER,ATOTAL      SET00260
     *             ,SITE(6),ANALYS(5),CAM(15),CRPKEY,KEPPTS(60)         SET00270
     *             ,DOTKEY,DOTERR                                       SET00280
C$END                                                                   SET00290
C                                                                       SET00300
      DATA A/'A'/,B/'NO'/                                               SET00310
      DATA BLANK/'    '/                                                SET00320
CI----------------------------------------------------------------------ISET00330
CI----------------------------------------------------------------------ISET00340
CI                                                                      ISET00350
CI    PURPOSE.. LOCATES FILE ON 'MAPTAP' AND COORDINATES                ISET00360
CI              ROUTINES TO ANALYSE 'DISPLAY' CONTROL CARDS           I SET00370
CI                                                                      ISET00380
CI                                                                      ISET00390
CI----------------------------------------------------------------------ISET00400
CI                                                                      SET00410
CC                                                                      SET00420
C                                                                       SET00430
      DIMENSION NUMVEC(30),FILVEC(2)                                    SET00440
      DIMENSION CARD(62),ARRAY(1)                                       SET00450
C*                                                                      SET00460
      DIMENSION  DOTCAT(62)                                             SET00470
C*                                                                      SET00480
      EQUIVALENCE ( DOTCAT(1) , CARD(1)  )                              SET00490
C*                                                                      SET00500
      DATA  YBCD/ 'Y' /, NBCD/ 'N' /, FILVEC/ 1 , 'F' /                 SET00510
C*                                                                      SET00520
       DIMENSION EQUVEC(2)                                              SET00530
      DATA   DRUM/'DRUM' /                                              SET00540
       DATA IRCD/'I'/,FRCD/'F'/,IBCD/'I'/,EQUVEC/1,'='/                 SET00550
       DIMENSION SLASH(2)                                               SET00560
       DATA SLASH/1,'/'/                                                SET00570
C*                                                                      SET00580
CC    INIZ                                                              SET00590
CC    ----------------------------------------------------------------- SET00600
CCCCCCCCCC                                                              SET00610
CCCCCCCCCC                                                              SET00620
CCCCCCCCCC                                                              SET00630
CCCCCCCCCC  READ FIRST CONTROL CARD FOR MAPTAP UNIT AND FILE NUMBER     SET00640
CCCCCCCCCC                                                              SET00650
C                                                                       SET00660
      READ(CRDUNT,1) CARD                                               SET00670
    1 FORMAT(10X,62A1)                                                  SET00680
      NFILE = 1                                                         SET00690
      COL = 0                                                           SET00700
      J = NXTCHR(CARD,COL)                                              SET00710
    2 IF (J.EQ.BLANK) GO TO 6                                           SET00720
      IF (J.NE.IRCD) GO TO 3                                            SET00730
      J = FIND12(CARD,COL,EQUVEC)                                       SET00740
      IF (J.NE.2) GO TO 6                                               SET00750
      ISTART = 0                                                        SET00760
      J = NUMBER(CARD,COL,MAPTAP,ISTART)                                SET00770
      J = FIND12(CARD,COL,EQUVEC)                                       SET00780
      IF (J.NE.2) GO TO 6                                               SET00790
```

```
FILE: SETUP3   FORTRAN  A

      ISTART = 0                                                        SET00800
      J = NUMBER(CARD,COL,NFILE,ISTART)                                 SET00810
      GO TO 10                                                          SET00820
3     IF (J.NE.IRCD) GO TO 4                                            SET00830
      J = FIND12(CARD,COL,SLASH)                                        SET00840
      IF (J.NE.2) GO TO 6                                               SET00850
      J = NXTCHR(CARD,COL)                                              SET00860
      IF (J.EQ.FRCD) GO TO 5                                            SET00870
      IF (J.EQ.UFCD) GO TO 2                                            SET00880
      GO TO 6                                                           SET00890
4     IF (J.NE.FRCD) GO TO 6                                            SET00900
5     J = FIND12(CARD,COL,EQUVEC)                                       SET00910
      IF (J.NE.2) GO TO 6                                               SET00920
      ISTART = 0                                                        SET00930
      J = NUMBER(CARD,COL,NFILE,ISTART)                                 SET00940
      J = FIND12(CARD,COL,EQUVEC)                                       SET00950
      IF (J.NE.2) GO TO 6                                               SET00960
      ISTART = 0                                                        SET00970
      J = NUMBER(CARD,COL,MAPTAP,ISTART)                                SET00980
      GO TO 10                                                          SET00990
6     WRITE(6,7)                                                        SET01000
7      FORMAT(' ERROR ON MAPTAP CARD')                                  SET01010
10     CONTINUE                                                         SET01020
       IF (NFILE.LE.0)  NFILE = 1                                       SET01030
      SERIAL = NFILE                                                    SET01040
C                                                                       SET01050
C     GET TAPE READY                                                    SET01060
C                                                                       SET01070
 40  REWIND MAPTAP                                                      SET01080
     IF ( NFILE .EQ. 1 ) GO TO 50                                       SET01090
     NF = NFILE - 1                                                     SET01100
     CALL FSRSFL(MAPTAP,NF,ISTAT)                                       SET01110
     IF (ISTAT .EQ. 0) GO TO 50                                         SET01120
     WRITE(6, 45)  NF, ISTAT                                            SET01130
 45  FORMAT(//// 5X,'***** OSPLAY/SETUP3  ...  ERROR CONDITION ON ATTEMSET01140
    1PT TO POSITION MAPTAP OVER',I4,3X,'FILES'// 5X,                    SET01150
    2 '***** FSRSFL STATUS CODE =',I4,3X,'...  ABORTING RUN *****'/1H1)SET01160
     REWIND MAPTAP                                                      SET01170
     CALL CMERR                                                         SET01180
C                                                                       SET01190
C      READ  MAPTAP                                                     SET01200
C     ------------------------------------------------------------------SET01210
C                                                                       SET01220
C                                                                       SET01230
 50  CONTINUE                                                           SET01240
     CATFLG=1                                                           SET01250
     READ(MAPTAP) CDATE(1),CDATE(2),BMFLG,BMCOMB,BMFEAT,NOCLS2,         SET01260
    *              NOFLD2,NOSUB2,NOFET2,TOTVT2,NOCAT,VARSZ2,           SET01270
    *             (FETVC2(I),I=1,NOFET2)                               SET01280
     NCAT=NOCAT                                                         SET01290
     IF(NOCAT.GT.0)GO TO 55                                             SET01300
     CATFLG=0                                                           SET01310
     NCAT=NOCLS2                                                        SET01320
C*   SET BASE ADDRESSES FOR TRAINING FIELD INFORMATION                  SET01330
 55  CONTINUE                                                           SET01340
     NOSUB3=NOSUB2+1                                                    SET01350
     FLDSV2=1                                                           SET01360
     VERTX2=FLDSV2 + NOFLD2*4                                           SET01370
     FIELD2=VERTX2 + TOTVT2*2                                           SET01380
     TOP1= FIELD2 + NOFLD2*5                                            SET01390
     NV=TOTVT2*2                                                        SET01400
     NF=NOFLD2*4                                                        SET01410
     READ(MAPTAP)(CATNAM(I),I=1,NCAT),(CLSNAM(I),I=1,NOCLS2),           SET01420
    *             (SUBNO(I),I=1,NOCLS2),(SUBNAM(I),I=1,NOSUB2),       SET01430
    *             (ARRAY(FLDSV2-1+I),I=1,NF),                         SET01440
    *             (ARRAY(VERTX2-1+I),I=1,NV) ,                        SET01450
    *             (SUBCAT(I),I=1,NOSUB2),(CLSSUB(I),I=1,NOSUB2)       SET01460
    *            ,(KATNO(I),I=1,NOCLS2),(KEPPTS(I),I=1,NOSUB2)        SET01470
C*                                                                     SET01480
C*   GO READ CONTROL CARDS AND TEST FIELDS                             SET01490
C*                                                                     SET01500
     FLDSV3=TOP1                                                        SET01510
     FIELD3= FLDSV3 + 800                                               SET01520
C*   RESERVE ROOM FOR 200 TEXT FIELDS                                  SET01530
     VERTX3=FIELD3 + 1000                                               SET01540
     VDIM = TOP - VERTX3                                                SET01550
     CALL REDIF3(ARRAY(FLDSV3),ARRAY(FIELD3),ARRAY(VERTX3),VDIM,        SET01560
    *   LISTSW,GTUNIT,GTFILE,AIUNIT,AIFILE,PPUNIT,PPFILE,               SET01570
    *   NAMECT,ALP)                                                     SET01580
```

```
FILE: SETUP3   FORTRAN  A

      TOP2=VERTX3+TOTVT3+2                                                      SET01590
      IF(NOFLD3.LE.0)TOP2=TOP1                                                  SET01600
C*    WRITE OUT SAVED TRAINING FIELDS AND TEST FIELDS                           SET01610
      IF ( STOPFG .NE. 0 ) GOTO 200                                             SET01620
C                                                                               SET01630
CC    PRINT OUT SUPERVISOR INFORMATION                                          SET01640
CC    ------------------------------------------------------------------------- SET01650
C                                                                               SET01660
      IF(THRSKY .EQ. 1)  GO TO 80                                               SET01670
      IF(EMPTHS .EQ. 2)  GO TO 80                                               SET01680
      IF(THRSVA .EQ. 3)  GO TO 80                                               SET01690
C-                                                                              SET01700
C-    TEST FOR FISHER                                                           SET01710
      IF(THRSKY.EQ.4) GO TO 80                                                  SET01720
C-                                                                              SET01730
      NOTHRS = 4                                                                SET01740
   80 CONTINUE                                                                  SET01750
      WRITE(6,819)                                                              SET01760
C*                                                                              SET01770
      WRITE(6,800)                                                              SET01780
  800 FORMAT(T5,'YOU HAVE SELECTED THE FOLLOWING OPTIONS:' / )                  SET01790
C*                                                                              SET01800
C*    CALL TAPLAB(MUNIT , MTAPE)                                                SET01810
      MUNIT=MAPTAP                                                              SET01820
      MTAPE = MUNIT                                                             SET01830
C*                                                                              SET01840
      WRITE(6,801)  MTAPE , MUNIT , NFILE                                       SET01850
C*                                                                              SET01860
  801 FORMAT( T10, 'PROCESS THE CLASSIFICATION RESULTS FROM MAPTAP ( ' ,SET01870
     *  A6, ' ), UNIT' , I5, ' ,  FILE' , I5 )                                 SET01880
      IF (NOTHRS .EQ. 4) WRITE(6,803)                                           SET01890
      IF (THRSKY .EQ. 1) WRITE(6,804)                                           SET01900
      IF(EMPTHS .EQ. 2) WRITE(6,805)                                            SET01910
      IF (THRSVA .EQ. 3) WRITE(6,818)                                           SET01920
C-                                                                              SET01930
C-    TEST FOR FISHER                                                           SET01940
      IF(THRSKY.EQ.4) WRITE(6,817)                                              SET01950
C-                                                                              SET01960
C*                                                                              SET01970
C*                                                                              SET01980
C*    IF DOT DATA PROCESSING WAS REQUESTED, THE DOT DATA FILE,UNIT,             SET01990
C*    AND TAPE NUMBER WILL BE PRINTED OUT                                       SET02000
C*                                                                              SET02010
C*** CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING                         SET02020
C*                                                                              SET02030
           IF (LISTSW.NE.0) GO TO 81                                            SET02040
C*                                                                              SET02050
      IF ( DOTKEY .LE. 0 )  GO TO 81                                            SET02060
C*    CALL TAPLAB(DOTUNIT,DTAPE)                                                SET02070
      DTAPE = DOTUNT                                                            SET02080
C*                                                                              SET02090
       IPAT = DOTFIL + 1                                                        SET02100
       WRITE(6,825) DOTUNT,IPAT,DTAPE                                           SET02110
C*                                                                              SET02120
      IF ( TRNKEY .GT. 0 )  WRITE(6,826 )                                       SET02130
C*                                                                              SET02140
C*    DO NOT PRINT FORMAT STMT. 806 IF DOTKEY .GT.0                             SET02150
C*                                                                              SET02160
   81       CONTINUE                                                          SET02170
C*                                                                              SET02180
      IF (TRNKEY.EQ.1.AND.DOTKEY.LE.0) WRITE (6,806)                            SET02190
      IF ( TSTKEY  .EQ. 1 )  WRITE(6,808)                                       SET02200
      IF ( STATKY  .EQ. 1 )  WRITE(6,810)                                       SET02210
      IF ( PCFDKY  .EQ. 1 )  WRITE(6,812)                                       SET02220
      IF(NOMAP.EQ.0)WRITE(6,820)                                                SET02230
      IF(PLTKEY.LE.0)WRITE(6,821)                                               SET02240
      IF(FILTER.EQ.1)WRITE(6,822)                                               SET02250
      IF(OFSKEY.EQ.1)WRITE(6,823)                                               SET02260
      IF(CRPKEY.EQ.1)WRITE(6,824)CROP                                           SET02270
C                                                                               SET02280
C*** CODE ADDED NOV 13, 1978 TO INCLUDE LIST PROCESSING                        SET02290
C                                                                               SET02300
           IF (LISTSW.EQ.0) GO TO 82                                            SET02310
           WRITE(6,827)                                                         SET02320
           WRITE(6,828)                                                         SET02330
           WRITE(6,829) GTUNIT,GTFILE,AIUNIT,AIFILE,PPUNIT,PPFILE               SET02340
           WRITE(6,830)  NAMECT                                                 SET02350
           WRITE(6,841) ALP(1),ALP(2)                                           SET02360
  827      FORMAT('0     LIST PROCESSING OPTION SELECTED  ')                   SET02370
```

```
 A28       FORMAT(///,5X,' NUMBERS OF GROUND TRUTH ',  A1, ' AND                SET02380
     * DISCRIMINATOR UNITS AND FILES ARE AS FOLLOWS ')                          SET02390
 A29       FORMAT(///,5X,8(2X,I5))                                             SET02400
 A30       FORMAT(///,5X,'  SELECTED CATEGORY NAME FOR LIST IS  ',A4)          SET02410
 A31       FORMAT(///,5X,'  BIAS CORRECTION  ALPHAS ARE  ',2F10.6)             SET02420
  A03 FORMAT(T10,'APPLY NO THRESHOLDING')                                      SET02430
  A04 FORMAT(T10,'APPLY CHI SQUARE THRESHOLDS')                                SET02440
  A05 FORMAT(T10,'APPLY EMPIRICAL THRESHOLDS')                                 SET02450
  A06 FORMAT(T10,'OUTLINE THE TRAINING FIELDS')                                SET02460
  A08 FORMAT(T10,'OUTLINE THE TEST FIELDS')                                    SET02470
  A10 FORMAT(T10,'PRINT OUT THE STATISTICS')                                   SET02480
  A12 FORMAT(T10,'PRINT THE GROUND TRUTH PERFORMANCE SUMMARIES BY FIELD'       SET02490
     * )                                                                       SET02500
C-                                                                            SET02510
  A17 FORMAT(T10,'APPLY FISHER F-DISTRIBUTION THRESHOLDS')                     SET02520
C-                                                                            SET02530
  A18 FORMAT(T10, 'APPLY USER INPUT THRESHOLD VALUES')                         SET02540
  A19 FORMAT(////)                                                            SET02550
  A20 FORMAT(T10,'DO NOT DISPLAY A CLASSIFICATION MAP')                        SET02560
  A21 FORMAT(T10,'DISPLAY THE HISTOGRAMS OF THE QUADRATIC FORM FOR ALL S       SET02570
     *UBCLASSES')                                                             SET02580
 A22 FORMAT(T10,'PERFORM SPATIAL FILTERING')                                   SET02590
  A23 FORMAT(T10,'EXCLUDE PIXELS IN THE DESIGNATED AREAS FROM CLASSIFICA       SET02600
     *TION SUMMARIES')                                                        SET02610
  A24 FORMAT(T10,'PRINT THE INTENSIVE TEST SITE SUMMARY REPORT FOR ',A6)       SET02620
 A25  FORMAT (T10,'PRINT DOT DATA PERFORMANCE SUMMARIES FOR DOT DATA FRO       SET02630
     1M FORTRAN UNIT',I3,' ,FILE NO. ',I3, ' ,TAPE (OR FILE)  ', A4)          SET02640
C*                                                                            SET02650
 A26  FORMAT( T10, 'OUTLINE THE DOTS ON THE CLASSIFICATION MAP' )             SET02660
C                                                                             SET02670
  A2        CONTINUE                                                          SET02680
C                                                                             SET02690
CC                                                                            SET02700
C                                                                             SET02710
      CALL WRTFLD(ARRAY(FLDSV2),ARRAY(VERTX2),NOFLD2,1,CLSNAM,SUBNAM)         SET02720
      IF(NOFLD3.LE.0)GO TO A5                                                 SET02730
      IF(TSTKEY.EQ.1)IK=2                                                     SET02740
      IF(DESKEY.EQ.1)IK=3                                                     SET02750
      CALL WRTFLD(ARRAY(FLDSV3),ARRAY(VERTX3),NOFLD3,IK,CLSNAM,SUBNAM)        SET02760
   A5 CONTINUE                                                                SET02770
C*                                                                            SET02780
C*    A COMPARISON OF CATEGORY NAMES FROM MAPTAP AND RDDOTS WILL BE           SET02790
C*    PERFORMED IF DOTKEY = 1                                                 SET02800
C                                                                             SET02810
C*** CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING                        SET02820
C                                                                             SET02830
          IF (LISTSW.NE.0) GO TO A7                                           SET02840
C*                                                                            SET02850
      IF ( DOTKEY .LE. 0 .OR. DOTERR .GT. 0 )  GO TO A6                       SET02860
C*                                                                            SET02870
C*     MAKE SPACE AVAILABLE IN  ARRAY  FOR DOT DATA INFORMATION,              SET02880
C*     INCLUDING 1000 SCRATCH LOCATIONS FOR TEMPORARY STORAGE OF DOT          SET02890
C*     DATA RETURNED FROM  SUBR.  RDDOTS.                                     SET02900
C*                                                                            SET02910
C*     MOVE THE TEST/DESIG STORAGE ( TSTSAV,TSTFLD,TSTVER ) TO OVERLAY        SET02920
C*     THE INPUT ( MAPTAP )  TRAINING FIELD STORAGE .                         SET02930
C*                                                                            SET02940
C*                                                                            SET02950
      IF ( NOCAT .LE. 0 )  GO TO 108                                          SET02960
C*                                                                            SET02970
  A7        CONTINUE                                                          SET02980
      NTSAV = FIELD3 - FLDSV3                                                 SET02990
      NTFLD = VERTX3 - FIELD3                                                 SET03000
      NTVER = TOTVT3 * 2                                                      SET03010
      NMOVE = NTSAV + NTFLD + NTVER                                           SET03020
      FROMAD = FLDSV3                                                         SET03030
      DO  90    I=1,NMOVE                                                     SET03040
      II = I                                                                  SET03050
      ARRAY(I) = ARRAY(FROMAD)                                                SET03060
   90 FROMAD = FLDSV3 + II                                                    SET03070
C*                                                                            SET03080
C*     RESET THE TEST/DESIG FIELDS STORAGE BASE ADDRESSES IN ARRAY            SET03090
C*                                                                            SET03100
      FLDSV3 = 1                                                              SET03110
      FIELD3 = FLDSV3 + NTSAV                                                 SET03120
      VERTX3 = FIELD3 + NTFLD                                                 SET03130
      TOP1 = VERTX3 + NTVER                                                   SET03140
C                                                                             SET03150
C*** CODE ADDED NOV 13,1978 TO INCLUDE LIST PROCESSING                        SET03160
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SETUP3 FORTRAN A

```
            IF (LISTS+.EQ.0) GO TO 88                                          SET03170
            FLDSV2 = TOP1                                                      SET03180
            FIELD2 = TOP1 + 4                                                  SET03190
            VERTX2 = TOP1 + 5                                                  SET03200
            TOP2 = VERTX2 + 500                                                SET03210
            NOFLD2 = 209                                                       SET03220
            NOTREG = 209                                                       SET03230
            PCTID3 = TOP2                                                      SET03240
            RETURN                                                             SET03250
 88         CONTINUE                                                           SET03260
C*                                                                             SET03270
C*     SET THE BASE ADDRESS  IN ARRAY FOR THE DOT 'TRAINING FIELDS'            SET03280
C*     STORAGE ---  ALLOW FOR A MAX. OF 250  DOTS                              SET03290
C*                                                                             SET03300
C*     BEGIN TRNSAV                                                            SET03310
      FLDSV2 = TOP1                                                            SET03320
C*     BEGIN TRNFLD                                                            SET03330
      FIELD2=FLDSV2 + 1000                                                     SET03340
C*     BEGIN TRNVEP                                                            SET03350
      VERTX2= FIELD2 + 1250                                                    SET03360
C*     BEGIN SCRATCH AREA FOR DOT(4,NDOTS)                                     SET03370
      TOP2 = VERTX2 + 500                                                      SET03380
      DIFF = TOP - TOP2                                                        SET03390
      IF (DIFF.LT.1000) GO TO 508                                              SET03400
      TYPSWT = 2                                                               SET03410
      DO 91 I=1,62                                                             SET03420
      DOTCAT(I) = BLANK                                                        SET03430
 91   CONTINUE                                                                 SET03440
C*                                                                             SET03450
C*     TO  RDDOTS :  TYPSWT = 2                                                SET03460
C*                                                                             SET03470
C*     RECEIVED FROM RDDOTS/DOTFILE :                                          SET03480
C*  (1)  DOTS(4,NDOTS), PLACED IN ARRAY(TOP2) TO ARRAY(TOP2+ 4*NDOTS-1)       SET03490
C* WITH DOTS(1,I)=SAMPLE NO.  ,  DOTS(2,I)=LINE NO.  ,  DOTS(3,I)=TYPE ,      SET03500
C*   DOTS(4,I)=DOT CATEGORY NO.                                                SET03510
C*  (2) SIZES = 4  ,  (3) TOTVT2 = TOTAL NO. OF DOTS IN THE DOT FILE ,         SET03520
C*  (4) NDCAT = NO. OF DOT CATEGORIES IN THE DOT FILE ,                        SET03530
C*  AND  (5)  DOTCAT(1), ... , DOTCAT(NDCAT) = DOT CATEGORY NAMES              SET03540
C*                                                                             SET03550
C*  ( DOTCAT IS EQUIVALENCED TO CARD(1) , ... , CARD(62)                       SET03560
C*                                                                             SET03570
C*                                                                             SET03580
      SIZES = 0                                                                SET03590
      ZERO = 0                                                                 SET03600
      CALL RDDOTS(ARRAY(TOP2),DUMMY,ZERO,TYPSWT,SIZES,TOTVT2,NDCAT,            SET03610
     *  DOTCAT,DUMMY,DUMMY,DUMMY,DUMMY,DUMMY,DUMMY,DUMMY,DUMMY,DUMMY,         SET03620
     2  DUMMY , DUMMY )                                                        SET03630
C*                                                                             SET03640
C*                                                                             SET03650
C*     TEST FOR THE MAX. NO. OF DOTS                                           SET03660
C*     SET  NOFLD2 = TOTVT2 = TOTAL NO. OF DOTS                                SET03670
C*                                                                             SET03680
      IF ( TOTVT2 .GT. 250 )   GO TO 1091                                      SET03690
      NOFLD2 = TOTVT2                                                          SET03700
C*                                                                             SET03710
C*     STORAGE ALLOCATION FOR THE DOT DATA INFORMATION                         SET03720
C*                                                                             SET03730
C*     TRNSAV(1,1) = ARRAY(FLDSV2)                                             SET03740
C*     TRNVEP(1,1) = ARRAY(VERTX2)                                             SET03750
C*     TRNFLD(1,1) = ARRAY(FIELD2)                                             SET03760
C*                                                                             SET03770
C*     SUBROUTINE DTCHK WILL COMPARE THE CATEGORY NAMES. IF EACH DOT           SET03780
C*     CATEGORY NAME MATCHES WITH A CATEGORY NAME FROM MAPTAP, THE DOTS        SET03790
C*     WILL BE ORDERED BY ASCENDING LINE NUMBER, AND ASCENDING SAMPLES         SET03800
C*     FOR EACH LINE NUMBER                                                    SET03810
C*                                                                             SET03820
C*     THE DOTS IN DOTFILE NEED TO BE CLASSIFIED BY CATEGORY                   SET03830
C*                                                                             SET03840
C*                                                                             SET03850
C*                                                                             SET03860
C* TO  DTCHK :  DOTS(4,NDOTS)  AT ARRAY(TOP2)                                  SET03870
C*                                                                             SET03880
C*   FROM  DTCHK :                                                             SET03890
C*                TRNSAV(4,NDOTS) ,BEGINNING AT ARRAY(FLDSV2) ,                SET03900
C*  WITH  TRNSAV(1,I) = UNUSED (SCRATCH AREA)  ,                               SET03910
C*        TRNSAV(2,I) = DOT CATEGORY NO.                                       SET03920
C*        TRNSAV(3,I) = DOT TYPE  ( =1 OR =2 )                                 SET03930
C*        TRNSAV(4,I) = 1                                                      SET03940
C*                                                                             SET03950
```

FILE: SETUP3 FORTRAN A

```
C*                                                                              SET03960
C*                  TRNVER(2,NDOTS) , BEGINNING AT ARRAY(VERTX2) ,              SET03970
C*    WITH TRNVER(1,I) = DOT SAMPLE NO.  ,                                      SET03980
C*         TRNVER(2,I) = DOT LINE NO.                                           SET03990
      CONTINUE                                                                  SET04000
C*                                                                              SET04010
C*                                                                              SET04020
      IF( NDCAT .LE. 0 )   GO TO 110                                            SET04030
      GO TO 500                                                                 SET04040
   86 CONTINUE                                                                  SET04050
C*                                                                              SET04060
C*                                                                              SET04070
C*    WILL PCTAR FIT IN ARRAY                                                   SET04080
      PCTSZ = NOFLD2                                                            SET04090
      NOTRFD = NOFLD2                                                           SET04100
      GO TO 105                                                                 SET04110
C*                                                                              SET04120
C*                                                                              SET04130
  86  NOTRFD=NOFLD2                                                             SET04140
      IF(TSTKEY.EQ.1)NOTRFD=NOFLD3                                              SET04150
      PCTSZ=NOTRFD*NOSUB2                                                       SET04160
      IF ( PCTSZ .LE. ( TOP-TOP2) )  GO TO 106                                  SET04170
C*    MOVE TEST FIELD INFO SO NO GAPS ARE IN ARRAY (IF STORAGE IS NEEDED        SET04180
      IF(TSTKEY.EQ.0) GO TO 105                                                 SET04190
C*    MUST TRAINING FIELD INFO BE KEPT IN CORE                                  SET04200
      IAD=FLDSV2-1                                                              SET04210
      IF(TRNKEY.EQ.1)IAD=TOP1-1                                                 SET04220
      IBD=FLDSV3-1                                                              SET04230
      NF=4*NOFLD3                                                               SET04240
      TIME=0                                                                    SET04250
   99 DO 101 I=1,NF                                                             SET04260
  101 ARRAY(IAD+I)=ARRAY(IBD+I)                                                 SET04270
      IF(TIME.GT.0)GO TO 102                                                    SET04280
      IAD=IAD+NF                                                                SET04290
      IBD=FIELD3-1                                                              SET04300
      NF=5*NOFLD3                                                               SET04310
      TIME=TIME+1                                                               SET04320
      GO TO 99                                                                  SET04330
  102 IF(TIME.EQ.2)GO TO 103                                                    SET04340
      IAD=IAD+NF                                                                SET04350
      IBD=VERTX3-1                                                              SET04360
      NF=TOTVT3*2                                                               SET04370
      TIME=TIME+1                                                               SET04380
      GO TO 99                                                                  SET04390
  103 CONTINUE                                                                  SET04400
      FLDSV3=TOP1                                                               SET04410
      IF(TRNKEY.EQ.0)FLDSV3=1                                                   SET04420
      FIELD3=FLDSV3 + 4*NOFLD3                                                  SET04430
      VERTX3=FIELD3 + 5*NOFLD3                                                  SET04440
      TOP2 = VERTX3 + 2*TOTVT3                                                  SET04450
C*                                                                              SET04460
  105 IF ( PCTSZ .GT. ( TOP-TOP2) )   GO TO 508                                 SET04470
C*                                                                              SET04480
  106 PCTID3 = TOP2                                                             SET04490
C*    SET UP FIELD ARRAY FOR TRAINING FIELDS                                    SET04500
C*                                                                              SET04510
      IF( DOTERR .GT. 0 )   RETURN                                              SET04520
C*                                                                              SET04530
  107 IF(TSTKEY.EQ.0.OR.TRNKEY.EQ.1)GO TO 400                                   SET04540
  401 CONTINUE                                                                  SET04550
C*                                                                              SET04560
C*    SET FLDKEY                                                                SET04570
C*                                                                              SET04580
C*                                                                              SET04590
C*    IF DOTKEY=1,SETKEY WILL NOT BE CALLED AND THE RESETTING OF THE            SET04600
C*    KEYS WILL TAKE PLACE                                                      SET04610
C*                                                                              SET04620
      IF( DOTKEY .GT. 0 )   GO TO 201                                           SET04630
      IF(TSTKEY.NE.1)GO TO 340                                                  SET04640
      FLDKEY=1                                                                  SET04650
      DO 310 I=1,NOFLD3                                                         SET04660
      NS = FLDSV3-1+3*(I-1)*4                                                   SET04670
      IF(ARRAY(NS).EQ.0)GO TO 320                                               SET04680
  310 CONTINUE                                                                  SET04690
      GO TO 200                                                                 SET04700
  340 FLDKEY=1                                                                  SET04710
      DO 350 I=1,NOFLD2                                                         SET04720
      NS = FLDSV2-1+3*(I-1)*4                                                   SET04730
      IF(ARRAY(NS).EQ.0)GO TO 320                                               SET04740
```

FILE: SETUP3 FORTRAN A

```
  350 CONTINUE                                                          SET04750
      RETURN                                                            SET04760
  320 FLDKEY=0                                                          SET04770
  200 RETURN                                                            SET04780
  201 DOTKEY = NDCAT                                                    SET04790
      FLDKEY = 0                                                        SET04800
      RETURN                                                            SET04810
C                                                                       SET04820
CC    ERROR ROUTINES                                                    SET04830
CC    ------------------------------------------------------------------SET04840
CC                                                                      SET04850
C                                                                       SET04860
  500 WRITE (6,5044) DIFF                                               SET04870
 5044 FORMAT (/////5X,'****** DISPLAY/SETUP3 - CORE OVERFLOW (TOP-TOP2) SET04880
     1 BY',I6,'-- EXECUTION TERMINATED ******'/1H1)                     SET04890
      CALL CMERR                                                        SET04900
  108 WRITE (6,109)                                                     SET04910
  109 FORMAT(/////5X,'*** CLASSIFICATION BY CATEGORY (ON MAPTAP ) IS    SET04920
     1 REQUIRED IN ORDER TO PROCESS THE DOT DATA  ***' / 5X, '*** DOT PESET04930
     2RFORMANCE SUMMARIES WILL NOT BE OUTPUT  ***' //// )               SET04940
C*                                                                      SET04950
      GO TO 112                                                         SET04960
C*                                                                      SET04970
C*  ERROR RETURN -  MAX. NO. OF DOTS EXCEEDED                           SET04980
C*                                                                      SET04990
 1001 WRITE(6,1002)  TOTVT2                                             SET05000
C*                                                                      SET05010
 1002 FORMAT(////5X,'**** DSPLAY/DSPLY2 ****  NO. OF DOTS =',I6,'   --- SET05020
     * EXCEEDS THE MAX. ALLOWABLE (250)' / 5X,'****  DOT PERFORMANCE SUMSET05030
     *MARIES WILL NOT BE PRODUCED ****' /// )                           SET05040
C*                                                                      SET05050
      GO TO 112                                                         SET05060
C*                                                                      SET05070
  110 WRITE(6,111)   NDCAT                                              SET05080
C*                                                                      SET05090
  111 FORMAT(//////5X,'***** SETUP3**  FROM DOTFILE, THE NO. OF DOT CATEGSET05100
     *ORIES =',I6 /5X,'****  DOT PERFORMANCE SUMMARIES WILL NOT BE PROVISET05110
     *DED  ****' ////// )                                               SET05120
C*                                                                      SET05130
C*    ERROR IN DOTFILE - RESET DOTKEY AND TRNKEY -- TURN ON DOTERR      SET05140
C*                                                                      SET05150
  112 DOTKEY = 0                                                        SET05160
      TRNKEY=0                                                          SET05170
      DOTERR=1                                                          SET05180
      RETURN                                                            SET05190
C*                                                                      SET05200
C*                                                                      SET05210
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-SET05220
C*                                                                      SET05230
C*                                                                      SET05240
C*    INTERNAL ROUTINE TO FIND RECTANGULAR COORDINTES FOR TRAINING FIELDSET05250
C*                                                                      SET05260
  400 CONTINUE                                                          SET05270
      IR=1                                                              SET05280
      IPT=1                                                             SET05290
      DO 20 I=1,NOFLD2                                                  SET05300
      SAMSTR = 100000                                                   SET05310
      SAMEND = 0                                                        SET05320
      LINSTR = 100000                                                   SET05330
      LINEND = 0                                                        SET05340
      NS = FLDSV2-1+4*(I-1)*4                                           SET05350
      NV = ARRAY(NS)                                                    SET05360
      NS = FIELD2-1+5*(I-1)*5                                           SET05370
      ARRAY(NS) = IPT                                                   SET05380
      IPT=IPT+NV*2                                                      SET05390
      IE=IR+NV-1                                                        SET05400
      DO 410 J=IR,IE                                                    SET05410
      NS = VERTX2 -1+1*(J-1)*2                                          SET05420
      SAMSTR = MIN0(SAMSTR,ARRAY(NS))                                   SET05430
      SAMEND = MAX0(SAMEND,ARRAY(NS))                                   SET05440
      NS = VERTX2-1+2*(J-1)*2                                           SET05450
      LINSTR=MIN0(LINSTR,ARRAY(NS))                                     SET05460
      LINEND=MAX0(LINEND,ARRAY(NS))                                     SET05470
  410 CONTINUE                                                          SET05480
      NS=FIELD2-1+1*(I-1)*5                                             SET05490
      ARRAY(NS)=LINSTR                                                  SET05500
      NS=FIELD2-1+2*(I-1)*5                                             SET05510
      ARRAY(NS)=LINEND                                                  SET05520
      NS=FIELD2-1+3*(I-1)*5                                             SET05530
```

FILE: SETUP3 FORTRAN A

```
      ARRAY(NS)=SAMSTR                                                  SET05540
      NS=FIELD2-1+4*(I-1)*5                                             SET05550
      ARRAY(NS)=SAMEND                                                  SET05560
      IF=IF+1                                                           SET05570
   20 CONTINUE                                                          SET05580
      GO TO 401                                                         SET05590
C*                                                                      SET05600
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*SET05610
C*                                                                      SET05620
C*                                                                      SET05630
C*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*SET05640
C*                                                                      SET05650
C*    INTERNAL ROUTINE TO CHECK THE CATEGORY NAMES AND TO ORDER THE DOTSSET05660
C*    IF THE CATEGORY NAMES MATCH                                       SET05670
C*                                                                      SET05680
  500 CONTINUE                                                          SET05690
C*                                                                      SET05700
C*                                                                      SET05710
C*    COMPUTATION TO FIGURE OUT HOW MANY LINES ARE NEEDED TO WRITE OUT  SET05720
C*    THE DOT CATEGORY NAMES AND THE CATEGORY NAMES FROM MAPTAP         SET05730
C*                                                                      SET05740
C*  NCAT = NO. OF CLASSIFICATION (MAPTAP) CATEGORIES                    SET05750
C*  NDCAT = NO. OF DOT FILE CATEGORIES                                  SET05760
C*                                                                      SET05770
      NDOTS=NOFLD2                                                      SET05780
      NCAT=NOCAT                                                        SET05790
      NDLIN = ((NDCAT*7)/119) + 1                                       SET05800
      NMLIN = ((NCAT*7)/119) + 1                                        SET05810
C*                                                                      SET05820
C*     USE TRNSAV(1,I) AS SCRATCH AREA                                  SET05830
C*                                                                      SET05840
      DO 51 I=1,NDOTS                                                   SET05850
      NS=FLDSV2-1+1+(I-1)*4                                             SET05860
      ARRAY(NS)=0                                                       SET05870
   51 CONTINUE                                                          SET05880
C*                                                                      SET05890
C*                                                                      SET05900
C*    CATEGORY NAMES CHECK                                              SET05910
C*                                                                      SET05920
      OKAY = .TRUE.                                                     SET05930
C*                                                                      SET05940
C*    PRINT OUT DOT FILE INFORMATION                                    SET05950
C*                                                                      SET05960
      WRITE(6,9000)NDCAT,NDOTS,(DOTCAT(I),I=1,NDCAT)                    SET05970
C*                                                                      SET05980
 9000 FORMAT(////////////2X,'DOT FILE INFORMATION :' //2X,'NO. DOT CATEGORSET05990
     *IES =', I6/ 2X, 'TOTAL NO. OF DOTS =' , I6/ 2X, 'DOT CATEGORY NAMESET06000
     *S :' // ( 1A9 ) )                                                 SET06010
C*                                                                      SET06020
      WRITE(6,9001)                                                     SET06030
      DO 205 KK=1,NDOTS                                                 SET06040
      NS = TOP2-1+(KK-1)*4                                              SET06050
      WRITE(6,9002)KK,(ARRAY(NS+K),K=1,4)                               SET06060
  205 CONTINUE                                                          SET06070
C*                                                                      SET06080
 9001 FORMAT(//2X,'DOT RECORD :' // T10,'DOT NO.' , T22,'SAMPLE' , T32, SET06090
     *'LINE' , T40, 'TYPE' , T49, 'CATEGORY' / T9, '---------' , T21 ,  SET06100
     *  '--------' , T31, '------' , T40, '----' , T48, '----------' //  SET06110
     *)                                                                 SET06120
 9002 FORMAT(T12,I3,T23,I4,T32,I4,T41,I2,T51,I3)                        SET06130
      M = 0                                                             SET06140
      DO 510 I=1,NDCAT                                                  SET06150
      DO 520 J=1,NCAT                                                   SET06160
      IF(DOTCAT(I).EQ.CATNAM(J))GO TO 11                                SET06170
  520 CONTINUE                                                          SET06180
      IF ( M .NE. NDCAT )  OKAY = .FALSE.                               SET06190
      GO TO 2101                                                        SET06200
   11 M=M+1                                                             SET06210
C*                                                                      SET06220
C*    SETTING THE LABELED DOT CATEGORY NUMBER TO THE MAPTAP'S CATEGORY  SET06230
C*    NUMBER THAT CORRESPONDS TO THE DOT CATEGORY NAME                  SET06240
C*                                                                      SET06250
      DO 515 RSET=1,NDOTS                                               SET06260
      NS = TOP2-1+4*(RSET-1)*4                                          SET06270
      NSS = FLDSV2-1+1+(RSET-1)*4                                       SET06280
      IF(ARRAY(NS).EQ.I)ARRAY(NSS)=J                                    SET06290
  515 CONTINUE                                                          SET06300
  510 CONTINUE                                                          SET06310
C*                                                                      SET06320
```

FILE: SETUP3 FORTRAN A

```
      DO  100  I=1,NDOTS                                                    SET06330
      NS = FLDSV2-1+1+(I-1)*4                                               SET06340
      CHK=ARRAY(NS)                                                         SET06350
      IF ( CHK .EQ. 0 )   GO TO 100                                         SET06360
      NS = TOP2-1+4+(I-1)*4                                                 SET06370
      NSS = FLDSV2-1+1+(I-1)*4                                              SET06380
      ARRAY(NS)=ARRAY(NSS)                                                  SET06390
  100 CONTINUE                                                              SET06400
C*                                                                          SET06410
C*    FINDING THE MAX AND MIN LINE NUMBERS                                  SET06420
 2101 IF(.NOT.OKAY)GO TO 12                                                 SET06430
C*                                                                          SET06440
      MNLIN = ARRAY(TOP2-1+2)                                               SET06450
      MXLIN = ARRAY(TOP2-1+2)                                               SET06460
      DO 1000 RSET=1,NDOTS                                                  SET06470
      NS = TOP2+1+(RSET-1)*4                                                SET06480
      IF(ARRAY(NS).LT.MNLIN)MNLIN=ARRAY(NS)                                 SET06490
      IF(ARRAY(NS).GT.MXLIN)MXLIN=ARRAY(NS)                                 SET06500
 1000 CONTINUE                                                              SET06510
      IF( MXLIN .LE. 0  .OR. MNLIN .LE. 0                                   SET06520
     *    .OR. MNLIN .GT. MXLIN )   OKAY = .FALSE.                          SET06530
      IF(.NOT.OKAY)GO TO 540                                                SET06540
C*                                                                          SET06550
C*                                                                          SET06560
C*    CHECK TO SEE WHAT KIND OF CORRESPONDENCE EXISTS BETWEEN THE TWO       SET06570
C*    SETS OF CATEGORY NAMES                                                SET06580
C*                                                                          SET06590
      NT=0                                                                  SET06600
      IF (M.EQ.NCAT) GO TO 13                                               SET06610
      WRITE (6,14) B                                                        SET06620
   14 FORMAT(/////////////////2X,'THERE IS ',A2,' 1-1 CORRESPONDENCE BETWSET06630
     *EEN THE CATEGORY NAMES FROM DOTFILE --')                             SET06640
   29 K=1                                                                   SET06650
      M=17                                                                  SET06660
      DO 26 I=1,NDLIN                                                       SET06670
      IF (M.GT.NDCAT) M=NDCAT                                               SET06680
      WRITE(6,27)(DOTCAT(ND),ND=K,M)                                        SET06690
   27 FORMAT(/1X,17(3X,A4))                                                 SET06700
      K=M+1                                                                 SET06710
      M=M+17                                                                SET06720
   26 CONTINUE                                                              SET06730
      WRITE (6,30)                                                          SET06740
   30 FORMAT (/' AND THE CATEGORY NAMES FROM MAPTAP -- ')                   SET06750
      K=1                                                                   SET06760
      M=17                                                                  SET06770
      DO 28 J=1,NMLIN                                                       SET06780
      IF (M.GT.NCAT) M=NCAT                                                 SET06790
      WRITE (6,27) (CATNAM(NM),NM=K,M)                                      SET06800
      K=M+1                                                                 SET06810
      M=M+17                                                                SET06820
   28 CONTINUE                                                              SET06830
      IF (NT.EQ.1) GO TO 21                                                 SET06840
      IF (NT.EQ.2) GO TO 18                                                 SET06850
      GO TO 21                                                              SET06860
   13 WRITE (6,14) A                                                        SET06870
      NT=1                                                                  SET06880
      GO TO 29                                                              SET06890
C*                                                                          SET06900
C*    ORDERING THE DOT DATA                                                 SET06910
C*                                                                          SET06920
   21 CNTR = 0                                                              SET06930
      DO  37  I=MNLIN,MXLIN                                                 SET06940
      M = CNTR + 1                                                          SET06950
      DO 32 J=1,NDOTS                                                       SET06960
      NS=TOP2-1+2+(J-1)*4                                                   SET06970
      IF(ARRAY(NS).NE.I)GO TO 31                                            SET06980
      CNTR = CNTR + 1                                                       SET06990
      NS = VERTX2-1+2+(CNTR-1)*2                                            SET07000
      ARRAY(NS)=I                                                           SET07010
      NS = FLDSV2-1+4+(CNTR-1)*4                                            SET07020
      ARRAY(NS)=1                                                           SET07030
      NS = VERTX2-1+1+(CNTR-1)*2                                            SET07040
      NSS = TOP2+(J-1)*4                                                    SET07050
      ARRAY(NS) = ARRAY(NSS)                                                SET07060
      NS = FLDSV2+2+(CNTR-1)*4                                              SET07070
      NSS = TOP2+2+(J-1)*4                                                  SET07080
      ARRAY(NS )=ARRAY(NSS)                                                 SET07090
      NS = FLDSV2+1+(CNTR-1)*4                                              SET07100
      NSS = TOP2-1+4+(J-1)*4                                                SET07110
```

FILE: SETUP3 FORTRAN A

```
      ARRAY(NS)=ARRAY(NSS)                                                  SET07120
   31 CONTINUE                                                              SET07130
   32 CONTINUE                                                              SET07140
C*                                                                          SET07150
   37 CONTINUE                                                              SET07160
C*                                                                          SET07170
      GO TO 84                                                              SET07180
   12 WRITE (6,25)                                                          SET07190
   25 FORMAT (1H1//5X,'*** DOT DATA PERFORMANCE SUMMARIES WILL NOT BE       SET07200
     1 PRODUCED - THE CATEGORY NAMES FROM MAPTAP AND DOTFILE DO NOT MATCSET07210
     2H!//2X,'CATEGORY NAMES FROM DOTFILE ARE -!)                           SET07220
      NT=2                                                                  SET07230
      GO TO 29                                                              SET07240
C*                                                                          SET07250
  540 WRITE(6,41)MNLIN,MXLIN                                                SET07260
C*                                                                          SET07270
   41 FORMAT(///////5X,'*** SETUP3/DTCHK  ---  FROM DOTFILE, THE MIN. LISET07280
     *NE NO. =',I6,'  ,MAX. LINE NO.=',I6/ 5X,'*****  DOT PERFORMANCE SUSET07290
     *MMARIES WILL NOT BE PRODUCED  *****' ///////)                         SET07300
   18 DOTKEY = 0                                                            SET07310
C*                                                                          SET07320
      TRNKEY=0                                                              SET07330
      DOTERR=1                                                              SET07340
C*                                                                          SET07350
      GO TO 84                                                              SET07360
       END                                                                  SET07370
```

FILE: TINORM FORTRAN A

```
      FUNCTION TINORM(ALPHA,IFLAG)                                          TIN00010
      DIMENSION A(3),B(3)                                                   TIN00020
       DATA A/.010328,.802853,2.515517/,B/.001308,                          TIN00030
     1 .189269,1.432788/                                                    TIN00040
C-----------------------------------------------------------------------    TIN00050
C     APPROXIMATION TO INVERSE NORMAL DISTRIBUTION                          TIN00060
C-----------------------------------------------------------------------    TIN00070
      IF(.NOT.(ALPHA.GT.0..AND.ALPHA.LT.1.)) IFLAG=1                        TIN00080
      X=ALPHA                                                               TIN00090
      IF(X.GT..5) X=1.-X                                                    TIN00100
      Y=SQRT(-2.*ALOG(X))                                                   TIN00110
      TINORM=Y-(A(3)*Y*(A(2)+Y*A(1)))/(1.+Y*(B(3)+Y*(B(2)+Y* B(1))))        TIN00120
      CALL OVERFL(I)                                                        TIN00130
      IF(I.EQ.1) GO TO 1                                                    TIN00140
      IF(ALPHA.LT..5) TINORM=-TINORM                                        TIN00150
      RETURN                                                                TIN00160
    1 IFLAG=1                                                               TIN00170
      RETURN                                                                TIN00180
      END                                                                   TIN00190
```

ORIGINAL PAGE IS
OF POOR QUALITY

13. DATA-TR PROCESSOR

FILE: DATATR FORTRAN A

```
      SUBROUTINE DATATR(ARRAY,TOP)                                          DAT00010
C                                                                           DAT00020
      IMPLICIT INTEGER(A-Z)                                                 DAT00030
C                                                                           DAT00040
C                                                                           DAT00050
      REAL   BIAS(16), BMAT(480), MAX(16), MIN(16), CON(16), CONMIN(32)     DAT00060
C                                                                           DAT00070
      REAL   AMAX(16), AMIN(16), ACON(16)                                   DAT00080
C                                                                           DAT00090
C                                                                           DAT00100
      DIMENSION   ARRAY(TOP),  MAXPT(30),  FILHIS(1616)                     DAT00110
C                                                                           DAT00120
      DIMENSION   HDR1(15) , HDR2(15) , COMNT(15) , INDATE(3)               DAT00130
C     INCLUDE COMBK1,LIST                                                   DAT00140
C     INCLUDE COMBK9,LIST                                                   DAT00150
C     INCLUDE COMBK6,LIST                                                   DAT00160
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,              DAT00170
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBUS2,FLDSV2,VERTX2,        DAT00180
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),            DAT00190
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                 DAT00200
     *              GRPCHK(61),GROUPS(124)                                  DAT00210
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             DAT00220
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,       DAT00230
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    DAT00240
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                        DAT00250
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                  DAT00260
     *  CROUNT,PRTUNT,RANDIO                                                DAT00270
C DATA TRANSFORMATION COMMON BLOCK                                          DAT00280
      COMMON/TRBLCK/OUTFMT,NOFEAT,FLDINF(6),          FETVEC(30)            DAT00290
CSEND                                                                       DAT00300
      DIMENSION VERTCS(2,11)                                                DAT00310
      DATA BLANKS/'    '/                                                   DAT00320
C                                                                           DAT00330
CCCCC                                                                       DAT00340
CCCCC    RESCALING METHOD IS DETERMINED IN SETUP8 :                         DAT00350
CCCCC                                                                       DAT00360
CCCCC    SCAFLG = 1 ,  RESCALE BY HISTOGRAM METHOD                          DAT00370
CCCCC                                                                       DAT00380
CCCCC    SCAFLG = 2 , RESCALE BY STATISTICS METHOD                          DAT00390
CCCCC                                                                       DAT00400
CCCCC    SCAFLG = 3 , RESCALE WITH USER-INPUT SCALING PARAMETERS            DAT00410
CCCCC                                                                       DAT00420
CCCCC                                                                       DAT00430
CCCCC    IF THE FLAG   RESCAL   IS ZERO, NO RESCALING OCCURS                DAT00440
CCCCC                                                                       DAT00450
CCCCC                                                                       DAT00460
C                                                                           DAT00470
      DO 10 I=1,15                                                          DAT00480
      IPL3 = I * 3                                                          DAT00490
      HDR1(I) = HEAD(IPL3)                                                  DAT00500
      IPL29 = I + 29                                                        DAT00510
      HDR2(I) = HEAD(IPL29)                                                 DAT00520
   10 COMNT(I) = BLANKS                                                     DAT00530
C                                                                           DAT00540
      INDATE(1) = HEAD(22)                                                  DAT00550
      INDATE(2) = HEAD(23)                                                  DAT00560
      INDATE(3) = HEAD(24)                                                  DAT00570
C                                                                           DAT00580
CC                                                                          DAT00590
C                                                                           DAT00600
      CALL SETUP8(BMAT,LCOMB,BMTRIG,PEROUT,MAXPT,ARRAY,LAM,                DAT00610
     *   SCAFLG , TOP, TRANSF, RESCAL, BIAS, ADDNUM, CONMIN, NPUN,NF )      DAT00620
      IF (RESCAL.EQ.0) GO TO 50                                             DAT00630
C                                                                           DAT00640
C                                                                           DAT00650
      IF (SCAFLG.EQ.1) GO TO 30                                             DAT00660
C                                                                           DAT00670
      IF (SCAFLG.EQ.2) GO TO 20                                             DAT00680
C                                                                           DAT00690
      IF( SCAFLG .EQ. 3 )                                                   DAT00700
C                                                                           DAT00710
     *       CALL SETREM ( CONMIN, CON, MIN, ADDNUM, LCOMB )                DAT00720
      GO TO 50                                                              DAT00730
C                                                                           DAT00740
CCCCCC                                                                      DAT00750
CCCCCC    IF RESCALING BY THE STATISTICS METHOD, APPLY TRANSFORMATION TO    DAT00760
CCCCCC    STATS ( MEANS, COVARIANCES ) , OBTAIN TRANSFORMED MAX AND MIN     DAT00770
CCCCCC    USING TRANSFORMED STATS .                                         DAT00780
C                                                                           DAT00790
```

FILE: DATATR FORTRAN A

```
   20  CALL  KBTRAN ( BMAT, LCOMB, ARRAY, LAM, MAX, MIN, CON, TRANSF )  DAT00800
C                                                                       DAT00810
      GO TO 50                                                          DAT00820
C                                                                       DAT00830
CCCCCCCC                                                                DAT00840
         IF RESCALING BY THE HISTOGRAM METHOD,  OBTAIN THE PREDICTED    DAT00850
         MAX AND MIN OF EACH COMPONENT OF THE TRANSFORMED DATA          DAT00860
         ( VIA  MAXMAT )  AND PERFORM A HISTOGRAM OF THE TRANSFORMED    DAT00870
         DATA ( VIA  TRHIST ), IN ORDER  TO OBTAIN THE RESCALING        DAT00880
         PARAMETERS ,  CON  AND  MIN .                                  DAT00890
C                                                                       DAT00900
   30    CALL  MAXMAT ( AMAX, AMIN, ACON, BMAT, LCOMB, MAXPT )          DAT00910
     CALL TRHIST(ARRAY,AMAX,AMIN,ACON,BMAT,LCOMB,PEROUT,               DAT00920
     *FILHIS,TOP,LAR,FLDNAM,NC,VERTCS,MAX,MIN,CON,                     DAT00930
     *      BIAS)                                                      DAT00940
C                                                                       DAT00950
CCCCCCCC                                                                DAT00960
         APPLY TRANSFORMATION TO INPUT DATA, RESCALE ( IF OPTED ), APPLYDAT00970
     - . REJECTION ( PEROUT ) TO DISTRIBUTION OF TRANSFORMED DATA, AND  DAT00980
         OUTPUT THE TRANSFORMED DATA ON THE FILE  ,  TRFORM .           DAT00990
C                                                                       DAT01000
      IF (LAR.EQ.0) GO TO 60                                            DAT01010
C                                                                       DAT01020
C                                                                       DAT01030
   50 CALL LNTRAN(ARRAY,MAX,MIN,CON,BMAT,LCOMB,BMTRIG,SCAFLG,PEROUT,    DAT01040
     * FILHIS,TOP,LAR,FLDNAM,NC,VERTCS, RESCAL, BIAS,                   DAT01050
     *  NF , NPUN )                                                    DAT01060
C                                                                       DAT01070
C                                                                       DAT01080
      IF (SCAFLG.EQ.1) GO TO 30                                         DAT01090
C                                                                       DAT01100
C                                                                       DAT01110
   60 CONTINUE                                                          DAT01120
C                                                                       DAT01130
      DO 70 I=1,15                                                      DAT01140
      IPL3 = I + 3                                                      DAT01150
      HEAD(IPL3) = HDR1(I)                                              DAT01160
      IPL29 = I + 29                                                    DAT01170
      HEAD(IPL29) = HDR2(I)                                             DAT01180
      IPL47 = I + 47                                                    DAT01190
   70 HEAD(IPL47) = COMNT(I)                                            DAT01200
C                                                                       DAT01210
      HEAD(22) = INDATE(1)                                              DAT01220
      HEAD(23) = INDATE(2)                                              DAT01230
      HEAD(24) = INDATE(3)                                              DAT01240
C                                                                       DAT01250
C                                                                       DAT01260
      WRITE (6,80)                                                      DAT01270
   80 FORMAT(////////// 10X, '***  SnATA-TR  COMPLETED  ***' ///////)    DAT01280
C                                                                       DAT01290
      RETURN                                                            DAT01300
      END                                                               DAT01310
                                                                        DAT01320
                                                                        DAT01330
```

FILE: KBTRAN FORTRAN A

```
      SUBROUTINE KBTRAN                                                 KBT00010
     *     ( BMAT, LCOMB, ARRAY, LAM, MAX, MIN, EPS, TRANSF)           KBT00020
C                                                                       KBT00030
      IMPLICIT INTEGER(A-Z)                                             KBT00040
      REAL TMIN,TMAX                                                    KBT00050
      REAL   BMAT(480), MAX(16), MIN(16), EPS(16)                       KBT00060
      REAL   C(480), CC(480),  D(16),  DIAG(480),  BMEAN(900)           KBT00070
C                                                                       KBT00080
CCC   INCLUDE COMRK1.LIST                                               KBT00090
CCC   INCLUDE COMRK9.LIST                                               KBT00100
CCC   INCLUDE COMRK4.LIST                                               KBT00110
C     INCLUDE COMRK6.LIST                                               KBT00120
      COMMON/INFORM/NOCLS2,NOSUR2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          COM00010
     *              AVAR2,COVAR2,CLSID2,SURNO2,SUBOS2,FLDSV2,VERTX2,    COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        COM00030
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             COM00040
     *              GRPCHK(61),GROUPS(124)                              COM00050
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                    COM00010
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                 COM00020
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               COM00030
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         COM00010
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,   COM00020
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               COM00030
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                  COM00040
     *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,            COM00050
     *   CRDUNT,PRTUNT,RANDIO                                          COM00060
C DATA TRANSFORMATION COMMON BLOCK                                      KBT00140
      COMMON/TRBLCK/OUTFMT,NOFEAT,FLDINF(6),          FETVEC(30)        KBT00150
C$END                                                                   KBT00160
C                                                                       KBT00170
      DIMENSION  COVHD2(15)                                             KBT00180
      DIMENSION ARRAY(1)                                                KBT00190
      DIMENSION NSUB(75)                                                KBT00200
C                                                                       KBT00210
      DATA COVHD2/'... ','TRAN','SFOR','MED ','STAT','ISTI','CS .',     KBT00220
     * '..  '/, BLANKS/'    '/                                           KBT00230
C                                                                       KBT00240
C COMPUTE TRANSFORMED MEANS FOR EACH SUBCLASS                           KBT00250
C                                                                       KBT00260
      DO 10 I=9,15                                                      KBT00270
   10 COVHD2(I) = BLANKS                                                KBT00280
      DO 20 I=1,NOSUR2                                                  KBT00290
      IPP=SUBOS2+I-1                                                    KBT00300
   20 NSUB(I)=ARRAY(IPP)                                                KBT00310
      LZ=AVAR2                                                          KBT00320
      K=1                                                               KBT00330
      DO 30 I=1,NOCLS2                                                  KBT00340
      CALL MATVEC(BMAT,ARRAY(LZ),BMEAN(K),LCOMB,NOFET2)                 KBT00350
      LZ=LZ+NOFET2                                                      KBT00360
   30 K=K+LCOMB                                                         KBT00370
C                                                                       KBT00380
C COMPUTE TRANSFORMED COVARIANCE MATRIX FOR EACH SUBCLASS               KBT00390
C                                                                       KBT00400
      KK=0                                                              KBT00410
      K=COVAR2                                                          KBT00420
      KKK=1                                                             KBT00430
      DO 50 I=1,NOCLS2                                                  KBT00440
C MULTIPLY BMAT BY COVARIANCE MATRIX                                    KBT00450
      CALL MTMLSQ(BMAT,ARRAY(K),C,LCOMB,NOFET2)                         KBT00460
C MULTIPLY RESULTING MATRIX BY TRANSPOSE OF BMAT                        KBT00470
      CALL MTMDAT(C,BMAT,CC,LCOMB,NOFET2,LCOMB,D,ARRAY(KKK))            KBT00480
      DO 40 II=1,LCOMB                                                  KBT00490
   40 DIAG(KK+II)=D(II)                                                 KBT00500
      KKK=KKK+(LCOMB*(LCOMB+1))/2                                       KBT00510
      KK=KK+LCOMB                                                       KBT00520
      K=K+VARSZ2                                                        KBT00530
   50 CONTINUE                                                          KBT00540
C                                                                       KBT00550
C PRINT TRANSFORMED COVARIANCE MATRIX                                   KBT00560
C                                                                       KBT00570
      CV1=(LCOMB*(LCOMB+1))/2                                           KBT00580
C                                                                       KBT00590
      IF (TRANSF.EQ.0) GO TO 80                                         KBT00600
      DO 60 I=1,15                                                      KBT00610
      TEMP = COMENT(I)                                                  KBT00620
      COMENT(I) = COVHD2(I)                                             KBT00630
   60 COVHD2(I) = TEMP                                                  KBT00640
C
      CALL PRTCOV(ARRAY(1),BMEAN(1),CV1,LCOMB,NSUB(1))
```

FILE: KBTRAN FORTRAN A

```
      DO 70 I=1,15                                                      KBT00650
   70 COMENT(I) = COVHD2(I)                                             KBT00660
   80 CONTINUE                                                          KBT00670
C                                                                       KBT00680
C CALCULATE MINIMUM AND MAXIMUM FOR EACH SUBCLASS                       KBT00690
C                                                                       KBT00700
      DO 120 I=1,LCOMB                                                  KBT00710
      DO 110 J=1,NOCLS2                                                 KBT00720
        NEL = (J - 1) * LCOMB + I                                       KBT00730
        MAX(I) = AMEAN(NEL) + LAM * DIAG(NEL)                           KBT00740
      IF (J.NE.1) GO TO 90                                              KBT00750
       TMAX = MAX(I)                                                    KBT00760
   90  CONTINUE                                                         KBT00770
        IF( MAX(I) .GE. TMAX) TMAX = MAX(I)                             KBT00780
        MIN(I) = AMEAN(NEL) - LAM * DIAG(NEL)                           KBT00790
      IF (J.NE.1) GO TO 100                                             KBT00800
        TMIN = MIN(I)                                                   KBT00810
  100  CONTINUE                                                         KBT00820
        IF( MIN(I) .LE. TMIN) TMIN = MIN(I)                             KBT00830
  110 CONTINUE                                                          KBT00840
       MIN(I) = TMIN                                                    KBT00850
       MAX(I) = TMAX                                                    KBT00860
       FPS(I) = 255. /(MAX(I) - MIN(I))                                 KBT00870
  120 CONTINUE                                                          KBT00880
      RETURN                                                            KBT00890
      END                                                               KBT00900
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: LNTRAN FORTRAN A

```
      SUBROUTINE LNTRAN(IDATA,MAX,MIN,CON,BMAT,LCOMB,BMTRIG,SCAFLG,      LNT00010
     * PEROUT,FILHIS,TOP,LAB,FLDNAM,NC,VERTCS, RESCAL, BIAS,             LNT00020
     *   NF , NPUN )                                                     LNT00030
                                                                         LNT00040
C                                                                        LNT00050
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLNT00060
CCC      IF   SCAFLG = 1 ,   RESCALE BY HISTOGRAM METHOD                 LNT00070
CCC                                                                      LNT00080
CCC      IF  SCAFLG = 2 ,  RESCALE BY THE STATISTICS METHOD              LNT00090
CCC                                                                      LNT00100
CCC      IF  SCAFLG = 3 ,  RESCALE WITH USER-INPUT SCALING PARAMETERS    LNT00110
CCC                                                                      LNT00120
CCC      NOTE:  IF THE FLAG,  RESCAL  , IS ZERO, NO RESCALING IS         LNT00130
CCC             PERFORMED. HOWEVER, PEROUT IS APPLIED TO THE TRANSFORMEDLNT00140
CCC             DATA DISTRIBUTION PRIOR TO FINAL OUTPUT OF TRANSFORMED  LNT00150
CCC             DATA VALUES .                                           LNT00160
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCLNT00170
C                                                                        LNT00180
      IMPLICIT INTEGER(A-Z)                                              LNT00190
      REAL   TMIN(16), TMAX(16),  MATOT ,  MITOT                         LNT00200
      REAL  NEWMAX(16) , NEWMIN(16) ,  SUM ,  CUT                        LNT00210
      REAL NXCON,PMIN,PMAX,CMIN                                          LNT00220
      REAL   BIAS(16), XCON(16), XT(16),  YREAL(16),  NPER1,  NPER2      LNT00230
      REAL   MAX(16), MIN(16), CON(16),  BMAT(480) ,  XXCON(16)          LNT00240
      REAL   MINSAV(16) , MAXSAV(16) , CONSAV(16)                        LNT00250
C                                                                        LNT00260
      DIMENSION   TOTPTS(16) , PMIN(16) , PMAX(16)                       LNT00270
      DIMENSION HISBUF(101),VERTCS(2,11),FL(8)                           LNT00280
      DIMENSION   IDATA(TOP) ,  Y(8000) ,  FILHIS(LCOMB,101 )            LNT00290
      DIMENSION   BADMIN(16) , BADMAX(16) , MINCUT(16) , MAXCUT(16)      LNT00300
C                                                                        LNT00310
      DATA OP/'('/,CP/')'/,COMMA/','/                                    LNT00320
      DATA TTL/'TOTL'/                                                   LNT00330
C                                                                        LNT00340
CCCCC INCLUDE COMBK1,LIST                                                LNT00350
CCCCC INCLUDE COMBK9,LIST                                                LNT00360
CCCCC INCLUDE COMBK4,LIST                                                LNT00370
CCCCC INCLUDE COMBK6,LIST                                                LNT00380
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           LNT00390
     *              AVAR2,COVAR2,CLSTD2,SUBNO2,SUBOS2,FLDSV2,VERTX2,     LNT00400
     *              FETVC2(30),SUBVC2(75),SURPTR(75),CLSVC2(60),         LNT00410
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              LNT00420
     *              GRPCHK(61),GROUPS(124)                               LNT00430
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                     LNT00440
      EQUIVALENCE  (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                 LNT00450
     2             (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               LNT00460
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          LNT00470
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,    LNT00480
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 LNT00490
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                     LNT00500
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,               LNT00510
     *  CRDUNT,PRTUNT,RANDIO                                             LNT00520
C DATA TRANSFORMATION COMMON BLOCK                                       LNT00530
      COMMON/TRBLCK/OUTFMT,NOFEAT,FLDINF(6),          FETVEC(30)         LNT00540
C$END                                                                    LNT00550
C                                                                        LNT00560
C*** CODE ADDED JAN. 15,1979 TO ALLOW MULTI-FILE OUTPUT                  LNT00570
C                                                                        LNT00580
        REWIND TRFORM                                                    LNT00590
          SKIP = NF - 1                                                  LNT00600
        CALL FSFMFL(TRFORM,SKIP,ISTAT)                                   LNT00610
      IF (RESCAL.EQ.0) GO TO 50                                          LNT00620
C                                                                        LNT00630
CCC   CHECK FOR RESCALE FACTORS INPUT BY USER (  SCAFLG = 3 )            LNT00640
C                                                                        LNT00650
      IF (SCAFLG.NE.3) GO TO 20                                          LNT00660
C                                                                        LNT00670
CCC      COMPUTE THE TRANSFORMED DATA MAX , USING INPUT                  LNT00680
CCC      SCALING PARAMETERS , CON  AND  MIN .                            LNT00690
C                                                                        LNT00700
      DO 10 KF=1,LCOMB                                                   LNT00710
       MAX(KF) = 255./ CON(KF) + MIN(KF)                                 LNT00720
   10  CONTINUE                                                          LNT00730
C                                                                        LNT00740
   20  CONTINUE                                                          LNT00750
C                                                                        LNT00760
CCC      COMPUTE THE OUTPUT HISTOGRAM SCALE FACTOR,  XCON                LNT00770
C                                                                        LNT00780
      DO 30 KK=1,LCOMB                                                   LNT00790
      XCON(KK)=(MAX(KK)-MIN(KK))/80
```

FILE: LNTRAN FORTRAN A

```
   30 FETVC2(KK)=KK                                                          LNT00800
C                                                                            LNT00810
      IF (SCAFLG.EQ.1) GO TO 80                                              LNT00820
C                                                                            LNT00830
CCCCCC  FOR STATISTICAL OR INPUT SCALE PARAMETERS, SAVE THE INITIAL          LNT00840
CCCCCC  SCALING PARAMETERS ( MIN, MAX , CON )  FOR RE-INITIALIZATION         LNT00850
CCCCCC  OF THESE PARAMETERS ON THE SECOND AND SUCCEEDING FIELDS TO BE        LNT00860
CCCCCC  INPUT , TRANSFORMED , AND RESCALED ( IF RESCAL GT 0 )                LNT00870
C                                                                            LNT00880
                                                                             LNT00890
      DO 40 I=1,LCOMB                                                        LNT00900
      MAXSAV(I) = MAX(I)                                                     LNT00910
      MINSAV(I) = MIN(I)                                                     LNT00920
   40 CONSAV(I) = CON(I)                                                     LNT00930
C                                                                            LNT00940
CCCCCC                                                                       LNT00950
CCCCCC  POSITION THE INPUT DATA FILE, AND READ IN THE HEADER RECORD          LNT00960
C                                                                            LNT00970
   50 CONTINUE                                                               LNT00980
      CALL TAPHDR(DATAPE,DATFIL)                                             LNT00990
C                                                                            LNT01000
                                                                             LNT01010
 60       NF = NF + 1                                                        LNT01020
C                                                                            LNT01030
CCCCCC  READ THE COORDINATES ( VERTICES ) OF THE FIELD FOR THE DATA          LNT01040
CCCCCC  TO BE TRANSFORMED .                                                  LNT01050
C                                                                            LNT01060
                                                                             LNT01070
   62 LAM=LAREAD(FLDNAM,VERTCS,FLDINF,NC)                                    LNT01080
      IF (LAM.EQ.0) GO TO 920                                                LNT01090
      IF (LAM.LT.0) GO TO 900                                                LNT01100
C                                                                            LNT01110
CCCCCC                                                                       LNT01120
CCCCCC  FOR STATISTICAL OR INPUT SCALING PARAMETERS, INITIALIZE THE          LNT01130
CCCCCC  SCALING PARAMETERS  MAX , MIN , CON , XCON  FOR THIS FIELD           LNT01140
C                                                                            LNT01150
                                                                             LNT01160
      IF (RESCAL.EQ.0) GO TO 80                                              LNT01170
C                                                                            LNT01180
      DO 70 I=1,LCOMB                                                        LNT01190
      MAX(I) = MAXSAV(I)                                                     LNT01200
      MIN(I) = MINSAV(I)                                                     LNT01210
      CON(I) = CONSAV(I)                                                     LNT01220
   70 XCON(I) = ( MAX(I) - MIN(I) )/80.                                      LNT01230
   80          CONTINUE                                                      LNT01240
C                                                                            LNT01250
      DO 90 I=1,LCOMB                                                        LNT01260
      MAXCUT(I) = 0                                                          LNT01270
      MINCUT(I) = 0                                                          LNT01280
      NEWMAX(I) = 255.0                                                      LNT01290
      XXCON(I) = XCON(I)                                                     LNT01300
   90 NEWMIN(I) = 0.0                                                        LNT01310
C                                                                            LNT01320
      MTRAN = 0                                                              LNT01330
C                                                                            LNT01340
      NSAMP=(FLDINF(5)-FLDINF(4))/FLDINF(6)+1                                LNT01350
      IDIM=NOFEAT*NSAMP                                                      LNT01360
      IF (IDIM.GT.TOP) GO TO 130                                             LNT01370
      IN=NC-1                                                                LNT01380
      WRITE (6,100)                                                          LNT01390
      WRITE (6,110)FLDNAM,IN,FLDINF(6),FLDINF(3),((OP,VERTCS(1,K),COMMA,     LNT01400
     *VERTCS(2,K),CP),K=1,IN)                                                LNT01410
  100 FORMAT(1H1,T24,'NO. OF   SAMPLE  LINE'/,                               LNT01420
     *T12,'FIELDNAME  VERTICES    INC    INC  VERTICES(SAMPLE,LINE)')        LNT01430
  110 FORMAT(13X,A4,8X,I2,7X,I4,2X,I4,2X,5(A1,I4,A1,I4,A1,2X)/               LNT01440
     * 5(A1,I4,A1,I4,A1,2X))                                                 LNT01450
      XDIM=LCOMB*NSAMP                                                       LNT01460
C                                                                            LNT01470
C                                                                            LNT01480
      IF (XDIM.LE.8000) GO TO 150                                            LNT01490
C                                                                            LNT01500
C                                                                            LNT01510
      WRITE (6,120)                                                          LNT01520
  120 FORMAT(//' THE NUMBER OF COMPONENTS TIMES THE NUMBER OF SAMPLES EXLNT01530
     *CEEDS THE SIZE OF THE STORAGE AREA ***'//)                             LNT01540
      CALL CMERR                                                             LNT01550
  130 WRITE (6,140)                                                          LNT01560
  140 FORMAT(' *** NUMBER OF CHANNELS TIMES NUMBER OF SAMPLES EXCEEDS 106LNT01570
     *00 ***'//)                                                             LNT01580
```

```
FILE: LNTRAN   FORTRAN  A

      CALL CMERR                                                        LNT01590
                                                                        LNT01600
C                                                                       LNT01610
CCCC INITIALIZE TAPE READING FOR THIS FIELD                             LNT01620
CCCC                                                                    LNT01630
  150 CALL FLDINT(FLDINF,FETVEC,NOFEAT)                                 LNT01640
C     LINES=(FLDINF(2)-FLDINF(1))/FLDINF(3)+1                           LNT01650
                                                                        LNT01660
CCCCC                                                                   LNT01670
CCCCC    WRITE HEADER RECORD ON OUTPUT FILE , TRFORM .                  LNT01680
CCCCC                                                                   LNT01690
                                                                        LNT01700
      CALL WRTHED(LCOMB,FETVC2,NSAMP,OUTFMT,TRFORM)                     LNT01710
                                                                        LNT01720
CC                                                                      LNT01730
      DO 170 I=1,LCOMB                                                  LNT01740
      BADMAX(I) = 0                                                     LNT01750
      BADMIN(I) = 0                                                     LNT01760
      PMIN(I) = 0                                                       LNT01770
      PMAX(I) = 0                                                       LNT01780
      TOTPTS(I) = 0                                                     LNT01790
      TMIN(I) = 1.0E35                                                  LNT01800
      TMAX(I) = -1.0E35                                                 LNT01810
      XCON(I) = XXCON(I)                                                LNT01820
      DO 160 J=1,101                                                    LNT01830
  160 FILHIS(I,J) = 0                                                   LNT01840
  170 CONTINUE                                                          LNT01850
C                                                                       LNT01860
CC    LSTLIN=0                                                          LNT01870
      M=0                                                               LNT01880
  180 M=M+1                                                             LNT01890
      IF (M.GT.LINES) GO TO 350                                         LNT01900
                                                                        LNT01910
CCC      READ ONE SCAN  LINE OF DATA FROM THE INPUT TAPE                LNT01920
CCC                                                                     LNT01930
      CALL LINERD(IDATA,ENDTAP)                                         LNT01940
      IF (ENDTAP.NE.0) GO TO 350                                        LNT01950
      IF (M.NE.1) GO TO 190                                             LNT01960
      ILIN=FLDINF(1)                                                    LNT01970
      GO TO 200                                                         LNT01980
  190 ILIN=ILIN+FLDINF(3)                                               LNT01990
  200 CONTINUE                                                          LNT02000
                                                                        LNT02010
CCCCCC   DETERMINE THE SAMPLE INTERCEPTS ON THE CURRENT SCAN LINE.      LNT02020
CCCCCC   WHICH ARE CONTAINED IN THE DESIRED FIELD BOUNDARIES. PLACE     LNT02030
CCCCCC   THE SAMPLE INTERCEPTS IN  FL  , AND THE NUMBER OF INTERCEPTS   LNT02040
CCCCCC   IN  JJ .                                                       LNT02050
                                                                        LNT02060
      CALL FDLINT(VERTCS,NC,FL,ILIN,NS,JJ)                              LNT02070
      DO 210 K=1,NSAMP                                                  LNT02080
      DO 210 IH=1,LCOMB                                                 LNT02090
      ZSAMP=(IH-1)*NSAMP+K                                              LNT02100
  210 Y(ZSAMP)=0                                                        LNT02110
C     NXCON = 255./100.                                                 LNT02120
                                                                        LNT02130
CCCCCC                                                                  LNT02140
CCCCCC   TRANSFORM, RESCALE, AND HISTOGRAM EACH DATA SAMPLE             LNT02150
CCCCCC                                                                  LNT02160
      DO 330 K=1,NSAMP                                                  LNT02170
      KP=(K-1)*FLDINF(6)+FLDINF(4)                                      LNT02180
      DO 320 JK=1,JJ,2                                                  LNT02190
      JKP1 = JK + 1                                                     LNT02200
      IF (KP.LT.FL(JK)) GO TO 330                                       LNT02210
      IF (KP.GT.FL(JKP1)) GO TO 310                                     LNT02220
      DO 220 I=1,LCOMB                                                  LNT02230
      XT(I)=0.                                                          LNT02240
                                                                        LNT02250
CCC   CALL TRANSF TO DO A DATA TRANSFORMATION                           LNT02260
CCC                                                                     LNT02270
        CALL TRANSF                                                     LNT02280
C         (XT, BMAT, IDATA, TOP, I, K, LCOMB, NSAMP, BIAS)              LNT02290
  220 CONTINUE                                                          LNT02300
      DO 300 I=1,LCOMB                                                  LNT02310
      IF( XT(I) .LT. TMIN(I) )  TMIN(I) = XT(I)                         LNT02320
      IF( XT(I) .GT. TMAX(I) )  TMAX(I) = XT(I)                         LNT02330
                                                                        LNT02340
                                                                        LNT02350
                                                                        LNT02360
                                                                        LNT02370
```

FILE: LNTRAN FORTRAN A

```
C                                                                       LNT02380
C     IF RESCAL = 0 NO RESCALING IS APPLIED.  OTHERWISE RESCALE         LNT02390
C        USING SCALING PARAMETERS DERIVED FROM EITHER HISTOGRAM,        LNT02400
C        STATISTICS, OR USER-INPUT ( SCAFLG= 1, 2, OR 3 )               LNT02410
C                                                                       LNT02420
C                                                                       LNT02430
      IF (RESCAL.GT.0) GO TO 260                                        LNT02440
C                                                                       LNT02450
C                                                                       LNT02460
C        IF TRANSFORMED DATA IS NOT RESCALED ,                          LNT02470
C        TEST FOR OUT- OF - RANGE  TRANSFORMED  VALUES                  LNT02480
C                                                                       LNT02490
C        SET = 0  ANY VALUE LESS THAN 0, OR LESS THAN THE NEW MIN       LNT02500
C        AFTER APPLICATION OF PEROUT                                    LNT02510
C        SET = 255  ANY VALUE GREATER THAN 255, OR GREATER THAN         LNT02520
C        THE NEW MAX AFTER APPLICATION OF PEROUT                        LNT02530
C                                                                       LNT02540
      IF (XT(I).LT.NEWMIN(I)) GO TO 230                                 LNT02550
      IF (XT(I).GT.NEWMAX(I)) GO TO 240                                 LNT02560
      GO TO 250                                                         LNT02570
  230 IF( MTRAN .EQ. 0 )  BADMIN(I) = BADMIN(I) + 1                     LNT02580
      XT(I) = 0.0                                                       LNT02590
      GO TO 250                                                         LNT02600
  240 IF ( MTRAN .EQ. 0 )  BADMAX(I) = BADMAX(I) + 1                    LNT02610
      XT(I) = 255.                                                      LNT02620
  250 CONTINUE                                                          LNT02630
C                                                                       LNT02640
C                                                                       LNT02650
C        FOR THE CURRENT SCAN LINE,  HISTOGRAM THE TRANSFORMED DATA ,   LNT02660
C        AND STORE THE TRANSFORMED DATA INTO THE OUTPUT ARRAY,  Y  .    LNT02670
C                                                                       LNT02680
      DPT = XT(I)/NXCON + 1.1                                           LNT02690
        TOTPTS(I) = TOTPTS(I) + 1                                       LNT02700
        IF( DPT .GT. 101) DPT = 101                                     LNT02710
      IF ( DPT .LE. 0 ) DPT = 1                                         LNT02720
        FILHIS(I,DPT) = FILHIS(I,DPT) + 1                               LNT02730
        ZS = (I -1) * NSAMP + K                                         LNT02740
      Y(ZS) = XT(I) + 0.5                                               LNT02750
      GO TO 300                                                         LNT02760
  260  CONTINUE                                                         LNT02770
C                                                                       LNT02780
C        FOR THE CURRENT SCAN LINE, HISTOGRAM THE TRANSFORMED DATA, AND LNT02790
C        STORE THE TRANSFORMED DATA INTO THE OUTPUT ARRAY,  Y  .        LNT02800
C                                                                       LNT02810
      IF (XT(I).LT.MIN(I)) GO TO 270                                    LNT02820
      IF (XT(I).GT.MAX(I)) GO TO 280                                    LNT02830
      YREAL(I)=CON(I)*(XT(I)-MIN(I))                                    LNT02840
C                                                                       LNT02850
      DPT = ( XT(I) - MIN(I) )/ XCON(I) + 11                            LNT02860
C                                                                       LNT02870
      IF ( DPT .LE. 0 )  DPT = 1                                        LNT02880
      IF ( DPT .GT. 101 )  DPT = 101                                    LNT02890
C                                                                       LNT02900
      GO TO 290                                                         LNT02910
  270 DPT = ABS( MIN(I) - XT(I) )/XCON(I)                               LNT02920
C                                                                       LNT02930
      DPT = 10 - DPT                                                    LNT02940
C                                                                       LNT02950
      PMIN(I) = PMIN(I) + 1                                             LNT02960
      IF(DPT.LE.0) DPT=1                                                LNT02970
      YREAL(I)=0                                                        LNT02980
      GO TO 290                                                         LNT02990
  280 DPT = ABS( XT(I) - MAX(I) )/XCON(I)                               LNT03000
C                                                                       LNT03010
      DPT = DPT + 91                                                    LNT03020
C                                                                       LNT03030
      PMAX(I) = PMAX(I) + 1                                             LNT03040
      IF(DPT.GT.101) DPT=101                                            LNT03050
      YREAL(I)=255                                                      LNT03060
  290 TOTPTS(I)=TOTPTS(I)+1                                             LNT03070
      FILHIS(I,DPT)=FILHIS(I,DPT)+1                                     LNT03080
      ZSAMP=(I-1)*NSAMP+K                                               LNT03090
      Y(ZSAMP) = YREAL(I) + 0.5                                         LNT03100
  300  CONTINUE                                                         LNT03110
      GO TO 330                                                         LNT03120
  310 IF (JKP1.GE.JJ) GO TO 340                                         LNT03130
  320 CONTINUE                                                          LNT03140
  330 CONTINUE                                                          LNT03150
                                                                        LNT03160
```

FILE: LNTRAN FORTRAN A

```
  340 CONTINUE                                                                  LNT03170
      IF(M.EQ.LINES) LSTLIN=-1                                                  LNT03180
                                                                                LNT03190
C                                                                               LNT03200
C        OUTPUT ONE LINE OF TRANSFORMED DATA ON THE OUTPUT FILE . TRFORM        LNT03210
C                                                                               LNT03220
                                                                                LNT03230
      CALL WRTLN(Y,LSTLIN)                                                      LNT03240
      GO TO 180                                                                 LNT03250
                                                                                LNT03260
C        IF RESCALING THE TRANSFORMED DATA BY EITHER THE STATISTICAL            LNT03270
C        OR USER-INPUT SCALING PARAMETERS.                                      LNT03280
C        APPLY PEROUT (  OF POINTS TO BE REJECTED)  TO THE TRANSFORMED          LNT03290
C        DATA DISTRIBUTION - OBTAIN THE MAX. AND SCALING PARAMETERS             LNT03300
C        MIN AND CON , AFTER APPLICATION OF PEROUT ( ALSO, THE HISTOGRAM        LNT03310
C        SCALE FACTOR , XCON ) .                                                LNT03320
C                                                                               LNT03330
C        IF NOT RESCALING.  APPLY PEROUT TO THE TRANSFORMED DATA.               LNT03340
C     GET  NEW MAX AND MIN, RE-HISTOGRAM , AND OUTPUT THE REVISED DISTR.        LNT03350
C                                                                               LNT03360
C  350 CONTINUE                                                                 LNT03370
C                                                                               LNT03380
C     IF (PEROUT.LE.0) GO TO 600                                                LNT03390
C                                                                               LNT03400
C     IF (SCAFLG.EQ.1) GO TO 600                                                LNT03410
C                                                                               LNT03420
C     IF (MTRAN.EQ.1) GO TO 600                                                 LNT03430
C                                                                               LNT03440
C     IF (RESCAL.GT.0) GO TO 430                                                LNT03450
C                                                                               LNT03460
C     NPER1 = FLOAT(PEROUT)/200.0                                               LNT03470
C                                                                               LNT03480
      DO 420 I=1,LCOMB                                                          LNT03490
      CUT =  NPER1 *  FLOAT( TOTPTS(I) )                                        LNT03500
      SUM = 0.0                                                                 LNT03510
      DO 370 J=1,101,1                                                          LNT03520
      IF (SUM.GE.CUT) GO TO 360                                                 LNT03530
      GO TO 370                                                                 LNT03540
  360 MINCUT(I) = SUM                                                           LNT03550
      NEWMIN(I) = (J-1) * NXCON + 0.5                                           LNT03560
      GO TO 380                                                                 LNT03570
C                                                                               LNT03580
  370 SUM = SUM + FILHIS(I,J)                                                   LNT03590
  380 SUM = 0.0                                                                 LNT03600
      J=101                                                                     LNT03610
  385 J=J-1                                                                     LNT03620
      IF (SUM.GE.CUT) GO TO 390                                                 LNT03630
      GO TO 400                                                                 LNT03640
  390 MAXCUT(I) =  SUM                                                          LNT03650
      NEWMAX(I) = ( J - 1 ) * NXCON + 0.5                                       LNT03660
      GO TO 410                                                                 LNT03670
  400 SUM = SUM + FILHIS(I,J)                                                   LNT03680
      IF(J.GT.1) GO TO 385                                                      LNT03690
  410 CONTINUE                                                                  LNT03700
  420 CONTINUE                                                                  LNT03710
      GO TO 500                                                                 LNT03720
C                                                                               LNT03730
C                                                                               LNT03740
  430 NPER1 = PEROUT * .01 + .001                                               LNT03750
      NPER2 = PEROUT * .01 - .001                                               LNT03760
C                                                                               LNT03770
      RSET=0                                                                    LNT03780
      IG = 0                                                                    LNT03790
      IB = 0                                                                    LNT03800
      IE = 0                                                                    LNT03810
      IA = 0                                                                    LNT03820
C                                                                               LNT03830
      DO 560 I=1,LCOMB                                                          LNT03840
      MATOT=TOTPTS(I)*NPER1                                                     LNT03850
      MITOT=TOTPTS(I)*NPER2                                                     LNT03860
      IF (PMIN(I).GT.MATOT) GO TO 440                                           LNT03870
      GO TO 460                                                                 LNT03880
C RESET MIN SMALLER                                                             LNT03890
  440 CMIN=PMIN(I)                                                              LNT03900
      J=10                                                                      LNT03910
  445 J=J-1                                                                     LNT03920
      IG=IG+1                                                                   LNT03930
      CMIN=CMIN-FILHIS(I,J)                                                     LNT03940
      IF (CMIN.GT.MATOT) GO TO 450                                              LNT03950
```

FILE: LNTRAN FORTRAN A

```
      IF(CMIN.LT.MITOT) IG=IG-1                                          LNT03960
      RSET=1                                                             LNT03970
      MIN(I)=MIN(I)-IG*XCON(I)                                           LNT03980
      IG=0                                                               LNT03990
      GO TO 490                                                          LNT04000
  450 IF(J.GT.1) GO TO 445                                               LNT04010
  460 IF (PMIN(I).LT.MITOT) GO TO 470                                    LNT04020
      GO TO 490                                                          LNT04030
C RESET MIN LARGER                                                       LNT04040
  470 CMIN=PMIN(I)                                                       LNT04050
      DO 480 J=11,91                                                     LNT04060
      IB=IB+1                                                            LNT04070
      CMIN=CMIN+FILMIS(I,J)                                              LNT04080
      IF (CMIN.LT.MITOT) GO TO 480                                       LNT04090
      IF(CMIN.GT.MATOT) IB=IB-1                                          LNT04100
      MIN(I)=MIN(I)+IB*XCON(I)                                           LNT04110
      IB=0                                                               LNT04120
      RSET=1                                                             LNT04130
      GO TO 490                                                          LNT04140
  480 CONTINUE                                                           LNT04150
C CHECK MAX                                                              LNT04160
  490  CONTINUE                                                          LNT04170
      IF (PMAX(I).GT.MATOT) GO TO 500                                    LNT04180
      GO TO 520                                                          LNT04190
C RESET MAX LARGER                                                       LNT04200
  500 CMIN=PMAX(I)                                                       LNT04210
      DO 510 J=92,101                                                    LNT04220
      IE=IE+1                                                            LNT04230
      CMIN=CMIN-FILMIS(I,J)                                              LNT04240
      IF (CMIN.GT.MATOT) GO TO 510                                       LNT04250
      IF(CMIN.LT.MITOT) IE=IE-1                                          LNT04260
      RSET=1                                                             LNT04270
      MAX(I)=MAX(I)+IE*XCON(I)                                           LNT04280
      IE=0                                                               LNT04290
      GO TO 550                                                          LNT04300
  510 CONTINUE                                                           LNT04310
C RESET MAX SMALLER                                                      LNT04320
  520 IF (PMAX(I).LT.MITOT) GO TO 530                                    LNT04330
      GO TO 550                                                          LNT04340
  530 CMIN=PMAX(I)                                                       LNT04350
      J=91                                                               LNT04360
  535 J=J-1                                                              LNT04370
      IA=IA+1                                                            LNT04380
      CMIN=CMIN+FILMIS(I,J)                                              LNT04390
      IF (CMIN.LT.MITOT) GO TO 540                                       LNT04400
      IF(CMIN.GT.MATOT) IA=IA-1                                          LNT04410
      RSET=1                                                             LNT04420
      MAX(I)=MAX(I)-IA*XCON(I)                                           LNT04430
      IA=0                                                               LNT04440
      GO TO 550                                                          LNT04450
  540 IF(J.GT.1) GO TO 535                                               LNT04460
  550 CONTINUE                                                           LNT04470
  560 CONTINUE                                                           LNT04480
      IF (RSET.EQ.0) GO TO 600                                           LNT04490
      DO 570 I=1,LCOMB                                                   LNT04500
      XCON(I)=(MAX(I)-MIN(I))/80                                         LNT04510
      XXCON(I) = XCON(I)                                                 LNT04520
  570  CON(I) = 255. / (MAX(I) - MIN(I))                                 LNT04530
C                                                                        LNT04540
  580 IF (NF.EQ.1) GO TO 590                                             LNT04550
C                                                                        LNT04560
      SKIP=NF-1                                                          LNT04570
      CALL FSFMFL(TRFORM,SKIP,ISTAT)                                     LNT04580
      MTRAN=1                                                            LNT04590
      GO TO 150                                                          LNT04600
  590 REWIND TRFORM                                                      LNT04610
      MTRAN=1                                                            LNT04620
      GO TO 150                                                          LNT04630
  600 CONTINUE                                                           LNT04640
      MTRAN=0                                                            LNT04650
C                                                                        LNT04660
      IF (RESCAL.EQ.0) GO TO 710                                         LNT04670
C                                                                        LNT04680
      WRITE (6,610)NF                                                    LNT04690
  610 FORMAT(///// 22A,'* OUTPUT FILE ', I3, 1X,'** /////// 5X,         LNT04700
     *  '**** TRANSFORMED VALUES RESCALED TO A RANGE  0 - 255  **** )    LNT04710
C                                                                        LNT04720
      IF (SCAFLG.EQ.1) WRITE (6,620)                                     LNT04730
      IF (SCAFLG.EQ.2) WRITE (6,630)                                     LNT04740
```

FILE: LNTRAN FORTRAN A

```
      IF (SCAFLG.EQ.3) WRITE (6,640)                                    LNT04750
C                                                                       LNT04760
  620 FORMAT( 24X, '(HISTOGRAM METHOD)' / )                             LNT04770
  630 FORMAT( 23X, '(STATISTICS METHOD)' / )                            LNT04780
  640 FORMAT( 20X, '(INPUT SCALING PARAMETERS)' / )                     LNT04790
C                                                                       LNT04800
C                                                                       LNT04810
      WRITE (6,650)                                                     LNT04820
C                                                                       LNT04830
  650 FORMAT(//// 7X, '... ORIGINAL TRANSFORMED DATA RANGE  ...' //     LNT04840
     *  T11, 'MIN' , T32, 'MAX' , T51, '( BIAS )'  // )                LNT04850
C                                                                       LNT04860
      DO 660 M=1,LCOMB                                                  LNT04870
  660 WRITE (6,670)TMIN(M),TMAX(M),BIAS(M)                              LNT04880
C                                                                       LNT04890
  670 FORMAT( 5X, F11.4, 10X, F11.4, 9X, '(' , F11.4, 1X, ')' / )       LNT04900
C                                                                       LNT04910
      WRITE (6,680)                                                     LNT04920
  680 FORMAT(///// 7X, '... TRANSFORMED DATA RANGE, AFTER APPLICATION OFLNT04930
     * PEROUT  ...' // 7X, 'MIN' , 10X, 'MAX' , 10X,                    LNT04940
     *   'CON  = 255/(MAX-MIN) ' // )                                   LNT04950
C PRINT OUT NEW MAX,MIN,CON ARRAYS                                      LNT04960
      DO 690 M=1,LCOMB                                                  LNT04970
  690 WRITE (6,700)MIN(M),MAX(M),CON(M)                                 LNT04980
  700 FORMAT(5X,3(F8.4,5X))                                             LNT04990
C                                                                       LNT05000
C                                                                       LNT05010
      GO TO 850                                                         LNT05020
C                                                                       LNT05030
C                                                                       LNT05040
  710 IF (PEROUT.GT.0) GO TO 730                                        LNT05050
C                                                                       LNT05060
      DO 720 I=1,LCOMB                                                  LNT05070
      IF ( BADMIN(I) .EQ. 0 )  NEWMIN(I) = TMIN(I)                      LNT05080
      IF ( BADMAX(I) .EQ. 0 )  NEWMAX(I) = TMAX(I)                      LNT05090
  720 CONTINUE                                                          LNT05100
C                                                                       LNT05110
  730 WRITE (6,740)NF                                                   LNT05120
  740 FORMAT(///// 19X, '* OUTPUT FILE ', I3, 1X, '*' )                 LNT05130
C                                                                       LNT05140
      WRITE (6,750)                                                     LNT05150
  750 FORMAT(///// 5X,'***  TRANSFORMED VALUES NOT RESCALED  ***' // )  LNT05160
C                                                                       LNT05170
      WRITE (6,760)LCOMB,(TMIN(I),I=1,LCOMB)                            LNT05180
  760 FORMAT(/ 5X, 'TRANSFORMED MINIMUMS, COMPONENTS  1-' , I2, 2X,     LNT05190
     * '...' // 2(5X, 8F12.2/)  )                                       LNT05200
C                                                                       LNT05210
      WRITE (6,770)LCOMB,(TMAX(I),I=1,LCOMB)                            LNT05220
  770 FORMAT(// 5X, 'TRANSFORMED MAXIMUMS, COMPONENTS  1-' , I2, 2X,    LNT05230
     * '...' // 2( 5X, 8F12.2 / )  )                                    LNT05240
C                                                                       LNT05250
      WRITE (6,780)LCOMB,(BIAS(I),I=1,LCOMB)                            LNT05260
C                                                                       LNT05270
  780 FORMAT(/// 5X, 'TRANSFORMED VALUE BIAS, COMPONENTS  1 - ' , I4,2X,LNT05280
C                                                                       LNT05290
     *  ':' // 2( 5X, 8F12.2 / )  )                                    LNT05300
C                                                                       LNT05310
C                                                                       LNT05320
C                                                                       LNT05330
      WRITE (6,790)(I,BADMIN(I),I=1,LCOMB)                              LNT05340
C                                                                       LNT05350
  790 FORMAT(/// 5X,'NO. OF TRANSFORMED VALUES LESS THAN 0 ( SET = 0 ) :LNT05360
C                                                                       LNT05370
     * ' // (5X,'COMPONENT', 1X, I2, '...' , I6, 2X, 'VALUES' ) )       LNT05380
C                                                                       LNT05390
C                                                                       LNT05400
      WRITE (6,800)(I,BADMAX(I),I=1,LCOMB)                              LNT05410
C                                                                       LNT05420
  800 FORMAT(/// 5X, 'NO. OF TRANSFORMED VALUES GREATER THAN 255 ( SET =LNT05430
C                                                                       LNT05440
     * 255 ) :' // (5X,'COMPONENT', 1X, I2,'...',I6, 2X, 'VALUES' ) )   LNT05450
C                                                                       LNT05460
C                                                                       LNT05470
      NPER1 = FLOAT(PEROUT)/2.0                                         LNT05480
C                                                                       LNT05490
      WRITE (6,810)NPER1,LCOMB,(MINCUT(I),I=1,LCOMB)                    LNT05500
  810 FORMAT(/// 3X, 'NO. OF LOWER TAIL POINTS  REJECTED      ( SET = 0 FLNT05510
     *OR OUTPUT )  TO SATISFY', F6.1,2X,'  CUT-OFF, COMPONENTS  1  -' ,LNT05520
     * I4, 2X, '...' // 16(I8) )                                        LNT05530
```

FILE: LNTRAN FORTRAN A

```
C                                                                       LNT05540
      WRITE (6,820)NPER1,LCOMB,(MAXCUT(I),I=1,LCOMB)                    LNT05550
  820 FORMAT(/// 3X, 'NO. OF UPPER TAIL POINTS   REJECTED  ( SET = 255 FLNT05560
     *OR OUTPUT )  TO SATISFY', F6.1, 2X, '_ CUT-OFF,  COMPONENTS  1  -'LNT05570
     *  , I4, 2X, '...' // 16( I8 ) )                                  LNT05580
      NPCT = 100 - PEROUT                                               LNT05590
      WRITE (6,830)NPCT,LCOMB,(NEWMIN(I),I=1,LCOMB)                     LNT05600
  830 FORMAT(//// 3X,'****  FINAL OUTPUT TRANSFORMED VALUES, CENTRAL' ,  LNT05610
     *  I5,3X,'_' , 2X, 'OF DISTRIBUTION :' // 3X, 'MINIMUMS, COMPONENTSLNT05620
     *  1  - ' , I4, 2X, '...' //  2(5X, 8F12.2 / )  )                 LNT05630
C                                                                       LNT05640
      WRITE (6,840)LCOMB,(NEWMAX(I),I=1,LCOMB)                          LNT05650
  840 FORMAT(// 3X,'MAXIMUMS, COMPONENTS  1  - ', I4, 2X,'...' //       LNT05660
     *  2( 5X, 8F12.2 / )  )                                           LNT05670
C                                                                       LNT05680
C                                                                       LNT05690
C                                                                       LNT05700
C PRINT HISTOGRAMS                                                      LNT05710
C                                                                       LNT05720
  850 CONTINUE                                                          LNT05730
C                                                                       LNT05740
      XSIZ=101                                                          LNT05750
      XHGH=255                                                          LNT05760
      XLOW=0                                                            LNT05770
      YSIZ=15                                                           LNT05780
      CALL COMHST(FILHIS,HISBUF,TTL,LCOMB,FETVC2,XSIZ,XHGH,XLOW,YSIZ)   LNT05790
C                                                                       LNT05800
C                                                                       LNT05810
      IF (RESCAL.EQ.0) GO TO 60                                         LNT05820
C                                                                       LNT05830
C                                                                       LNT05840
C                                                                       LNT05850
      WRITE (6,860)NF                                                   LNT05860
C                                                                       LNT05870
  860 FORMAT(1H1 // 5X,'SCALING PARAMETERS USED ON TRANSFORMED VALUES, OLNT05880
C                                                                       LNT05890
     *UTPUT FILE', I5// 19X, 'MINIMUM', 7X, 'MAXIMUM', 7X,              LNT05900
C                                                                       LNT05910
     *  'SCALE FACTOR ( CON )'  )                                       LNT05920
C                                                                       LNT05930
      WRITE (6,870)(FETVC2(IL),MIN(IL),MAX(IL),CON(IL),IL=1,LCOMB)      LNT05940
C                                                                       LNT05950
  870 FORMAT(1X, 'COMPONENT',I3,1X,F12.3,2X,F12.3,7X,F12.3 )            LNT05960
C                                                                       LNT05970
C                                                                       LNT05980
      IF (NPUN.LE.0) GO TO 890                                          LNT05990
C                                                                       LNT06000
      PUNCH 880,(CON(MN),MIN(MN),MN=1,LCOMB)                            LNT06010
C                                                                       LNT06020
  880 FORMAT(  ('OPTION',4X, 'SCAFAC=' , 2( ' (', F9.3, ',' , F9.3,     LNT06030
     * ' ) ' , ' )  ) )                                                LNT06040
C                                                                       LNT06050
  890  CONTINUE                                                         LNT06060
C                                                                       LNT06070
C                                                                       LNT06080
      IF ( SCAFLG .EQ. 1 )   RETURN                                     LNT06090
C                                                                       LNT06100
C                                                                       LNT06110
      GO TO 60                                                          LNT06120
  900 IF (LAM.EQ.0) GO TO 920                                           LNT06130
      WRITE (6,910)FLDNAM                                               LNT06140
  910 FORMAT(//// 5X,'***** DATATR/LNTRAN *****  ERROR ON INPUT FIELD DELNT06150
     *FINITION CARD, FOR FIELD NAME  ',1H', A4, 1H', 3X, '*****' /      LNT06160
     *  10X, 'CONTINUING TO NEXT FIELD DEFINITION CARD(S)' //// )      LNT06170
      GO TO 60                                                          LNT06180
  920  CONTINUE                                                         LNT06190
      RETURN                                                            LNT06200
      END                                                               LNT06210
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: MAXMAT FORTRAN A

```
      SUBROUTINE MAXMAT ( MAX, MIN, CON, BMAT, LCOMB, MAXPT )
C
C   COMPUTE AN APPROXIMATE TRANSFORMED MAX AND MIN FOR EACH COMPONENT
C    OF THE TRANSFORMATION
C
      IMPLICIT INTEGER(A-Z)
      DIMENSION MAXPT(30)
C
C     REAL   BMAT(480), MAX(15), MIN(16), CON(16)
C
CCCCCC
CCCCCC       USING INPUT ( OR DEFAULT )  MAXIMUM DATA VALUE FOR EACH
CCCCCC       CHANNEL, COMPUTE THE TRANSFORMED VALUE RANGE ( MAX  AND  MIN )
CCCCCC       AND COMPUTE THE HISTOGRAM SCALING FACTOR, CON .
CCCCCC
CCCCCC
C     INCLUDE COMBK9,LIST
C DATA TRANSFORMATION COMMON BLOCK
      COMMON/TRBLCK/OUTFMT,NOFEAT,FLDINF(6),          FETVEC(30)
C$END
      DO 30 I=1,LCOMB
      MAX(I) = 0.0
      MIN(I) = 0.0
      DO 20 J=1,NOFEAT
      K=(J-1)*LCOMB+I
      IF (BMAT(K).LE.0.0) GO TO 10
C
      MAX(I) = MAX(I) + BMAT(K) * MAXPT(J)
C
      GO TO 20
   10 CONTINUE
C
      MIN(I) = MIN(I) + BMAT(K) * MAXPT(J)
C
   20 CONTINUE
      CON(I)=(MAX(I)-MIN(I))/100.
   30 CONTINUE
      RETURN
      END
```

MXM00010 MXM00020 MXM00030 MXM00040 MXM00050 MXM00060 MXM00070 MXM00080 MXM00090 MXM00100 MXM00110 MXM00120 MXM00130 MXM00140 MXM00150 MXM00160 MXM00170 MXM00180 MXM00200 MXM00210 MXM00220 MXM00230 MXM00240 MXM00250 MXM00260 MXM00270 MXM00280 MXM00290 MXM00300 MXM00310 MXM00320 MXM00330 MXM00340 MXM00350 MXM00360 MXM00370 MXM00380 MXM00390

FILE: SETREM FORTRAN A

```
      SUBROUTINE SETREM                                                 SRE00010
     *     (CONMIN, CON, MIN, ADDNUM , LCOMB )                         SRE00020
C                                                                       SRE00030
      IMPLICIT INTEGER (A-Z)                                            SRE00040
      REAL    CON(16)  ,  MIN(16)  ,  CONMIN(2, 16 )                    SRE00050
      NUMCM = ADDNUM / 2                                                SRE00060
      IF (NUMCM.NE.LCOMB) GO TO 20                                      SRE00070
      DO 10 NM=1,LCOMB                                                  SRE00080
      CON(NM) = CONMIN(1, NM)                                           SRE00090
      MIN(NM) = CONMIN(2,NM)                                            SRE00100
   10 CONTINUE                                                          SRE00110
      RETURN                                                            SRE00120
   20 WRITE (6,30)NUMCM,LCOMB                                           SRE00130
   30 FORMAT (5X, 'SETREM ERROR - THERE WERE ', I5, ' SCALE ',          SRE00140
     C     'FACTORS AND MINIMUM VALUES INPUT THROUGH THE SCAFAC ',      SRE00150
     C     'OPTION.',                                                   SRE00160
     C     /, 5X, I5, ' LINEAR COMBINATIONS WERE REQUESTED.',           SRE00170
     C     /, 5X, 'THERE MUST BE A SCALE FACTOR AND A MINIMUM ',        SRE00180
     C     'VALUE FOR EACH LINEAR COMBINATION.',                        SRE00190
     C     /, 5X, 'THE PROGRAM WILL TERMINATE THROUGH CMERR')           SRE00200
      CALL CMERR                                                        SRE00210
      END                                                               SRE00220
```

FILE: SETUP8 FORTRAN A

```
      SUBROUTINE SETUP8(BMAT,LCOMB,BMTRIG,PEROUT,MAXPT,ARRAY,LAM,SCAFLG,SET00010
     *   TOP, TRANSF, RESCAL, BIAS, ADDNUM, CONMIN,NPUN,NF)                SET00020
C                                                                         SET00030
      IMPLICIT INTEGER(A-Z)                                               SET00040
C                                                                         SET00050
      LOGICAL   NUDTAP , NUDFIL , NUSTAP , NUSFIL                         SET00060
      REAL   CONMIN(32) ,  BIAS(16) ,  BMAT(480)                          SET00070
C                                                                         SET00080
      DIMENSION MAXPT(30)                                                 SET00090
      DIMENSION ARRAY(1)                                                  SET00100
      DIMENSION EQUVEC(2)                                                 SET00110
       DIMENSION CINDEX(19),                                              SET00120
     *     SINVEC(3), FRVEC1(3) , FRVEC2(3), CARD2(62),                   SET00130
     *     BTEST(3)                                                       SET00140
      DIMENSION   COMHD1(15)   ,   OP(2)   ,   CP(2)                      SET00150
       DIMENSION MTX (5),ACARD(20)                                        SET00160
C                                                                         SET00170
C     INCLUDE COMBK1,LIST                                                 SET00180
C     INCLUDE COMBK4,LIST                                                 SET00190
C     INCLUDE COMBK6,LIST                                                 SET00200
C     INCLUDE COMBK9,LIST                                                 SET00210
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,            SET00220
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,      SET00230
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),          SET00240
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),               SET00250
     *              GRPCHK(61),GROUPS(124)                                SET00260
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                      SET00270
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                   SET00280
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               SET00290
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,           SET00300
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,     SET00310
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  SET00320
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                     SET00330
     *   ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,               SET00340
     *   CROUNT,PRTUNT,RANDIO                                             SET00350
C DATA TRANSFORMATION COMMON BLOCK                                        SET00360
      COMMON/TRBLCK/OUTFMT,NOFEAT,FLDINF(6),          FETVEC(30)          SET00370
C*END                                                                     SET00380
C                                                                         SET00390
      EQUIVALENCE (FLDINF(1),LINSTR),(FLDINF(2),LINEND),                  SET00400
     *            (FLDINF(3),LININC),(FLDINF(4),SAMSTR),                  SET00410
     *            (FLDINF(5),SAMEND),(FLDINF(6),SAMINC)                   SET00420
C                                                                         SET00430
      DATA CINDEX/'B-MA','CHAN','FORM','HED1','HED2',                     SET00440
     * 'DATE','COMM','MAXP','PERO','SUBC',                                SET00450
     *'MODU','LAM ','OPTI','*END','DATA','STAT',                         SET00460
     *     'RESC','BIAS','TROU'/                                         SET00470
       DATA MTX / 4, '0', 'T', 'S', 'P' /                                 SET00480
      DATA   OP/ 1 , '(' /  ,  CP/ 1 , ')' /  ,  ZERO/ 0 /               SET00490
      DATA EQUVEC/1,'='/                                                  SET00500
        DATA SINVEC/2,',','='/,                                           SET00510
     *    CINMAX /19/, MAXFET /30/,                                       SET00520
     * FRVEC1/2,'I','U'/ ,FRVEC2/2,'U','L'/,BLANK/' '/,                   SET00530
     * BTEST/2,'C','F'/                                                   SET00540
     4 ,CBCD/'C'/,FRCD/'F'/,UBCD/'U'/                                     SET00550
      DATA COMHD1/'... ','ORIG','INAL',' STA','TIST','ICS ',',... '/     SET00560
C                                                                         SET00570
C                                                                         SET00580
C        INITIALIZE FLAGS AND DEFAULT VALUES                              SET00590
C                                                                         SET00600
C                                                                         SET00610
      NOSUB2=0                                                            SET00620
      NOGRP=0                                                             SET00630
      BMTRIG=0                                                            SET00640
      NSF = 1                                                             SET00650
       RESCAL = 0                                                         SET00660
      SCAFLG = 0                                                          SET00670
      MPT = 0                                                             SET00680
      ORIG = 0                                                            SET00690
      TRANSF = 0                                                          SET00700
      NPUN = 0                                                            SET00710
      OUTFMT=2                                                            SET00720
      TRFORM=14                                                           SET00730
C                                                                         SET00740
C  INITIALIZE THE TRANSFORMATION BIAS VECTOR ( BIAS ) AND NO. OF          SET00750
C  BIAS VALUES  ( NBS )                                                   SET00760
C                                                                         SET00770
      DO 10 I=1,16                                                        SET00780
   10 BIAS(I) = 0.0                                                       SET00790
```

FILE: SETUP8 FORTRAN A

```
      NBS = 0                                                                   SET00800
      NUDTAP =  .FALSE.                                                         SET00810
      NUDFIL =  .FALSE.                                                         SET00820
      NUSTAP =  .FALSE.                                                         SET00830
      NUSFIL =  .FALSE.                                                         SET00840
      BMSWT=1                                                                   SET00850
C                                                                               SET00860
C  INITIALIZE THE MAXIMUM EXPECTED DATA VALUE, FOR EACH CHANNEL                 SET00870
C                                                                               SET00880
      DO 20 I=1,30                                                              SET00890
   20 MAXPT(I)=255                                                              SET00900
C                                                                               SET00910
C  INITIALIZE    DISTRIBUTION CUT-OFF,  PEROUT ,  AND                         SET00920
C  THE STANDARD DEVIATION MULTIPLE ,  LAM .                                     SET00930
C                                                                               SET00940
      LAM=2                                                                     SET00950
      PEROUT=5                                                                  SET00960
      DO 30 I=8,15                                                              SET00970
   30 COVHD1(I) = BLANK                                                         SET00980
C                                                                               SET00990
      DO 40 I=1,15                                                              SET01000
   40 COMENT(I) = BLANK                                                         SET01010
C                                                                               SET01020
C  NOW SET UP REREAD BUFFER.                                                    SET01030
C                                                                               SET01040
      CALL REREAD(30,80)                                                        SET01050
   50 COL=0                                                                     SET01060
C  NOW READ A CARD INTO THE BUFFER                                              SET01070
      READ(21,55)(ACARD(I),I=1,20)                                              SET01080
   55 FORMAT(20A4)                                                              SET01090
      WRITE(30,55)(ACARD(I),I=1,20)                                             SET01100
      REWIND 30                                                                 SET01110
C  STATFILE CARD READ                                                           SET01120
      IF (NUSTAP .OR. NUSFIL) SCAFLG = 2                                        SET01130
      READ (30,60)CODE,CARD2                                                    SET01140
   60 FORMAT(A4,6X,62A1)                                                        SET01150
      REWIND 30                                                                 SET01160
      WRITE (6,70)CODE,CARD2                                                    SET01170
   70 FORMAT(T5,A4,6X,62A1)                                                     SET01180
      DO 80 I=1,CINMAX                                                          SET01190
      IF (CINDEX(I).EQ.CODE)GO TO(110,150,160,180,190,210,200,230,250,          SET01200
     *270,360,280,290,500,380,430,480,490,600),I                                SET01210
   80 CONTINUE                                                                  SET01220
C                                                                               SET01230
   90 WRITE (6,100)CODE,CARD2                                                   SET01240
C                                                                               SET01250
  100 FORMAT(//// 5X,'*** BAD CONTROL CARD - DATATR/SETUP8 ***' // 5X,          SET01260
     *   A4, 6X, 62A1  /// )                                                    SET01270
C                                                                               SET01280
      GO TO 50                                                                  SET01290
C B-MATRIX CARD                                                                 SET01300
  110 J=NXTCHR(CARD2,COL)                                                       SET01310
      IF (J.EQ.BLANK) GO TO 540                                                 SET01320
      COL=COL-1                                                                 SET01330
      M=FIND12(CARD2,COL,BTEST)                                                 SET01340
      IF (M.EQ.-1) GO TO 540                                                    SET01350
      BMTRIG=1                                                                  SET01360
      IF (M.EQ.2) GO TO 120                                                     SET01370
C B-MATRIX DATA ON TAPE FILE                                                    SET01380
      KEY=2                                                                     SET01390
C READ B-MATRIX ARRAY FROM TAPE FILE                                            SET01400
      CALL BMFIL(BMAT,LCOMB,NOFEAT,FETVEC,KEY)                                  SET01410
      GO TO 130                                                                 SET01420
C B-MATRIX DATA READ FROM CARD FILE                                             SET01430
  120 KEY=1                                                                     SET01440
      CALL BMFIL(BMAT,LCOMB,NOFEAT,FETVEC,KEY)                                  SET01450
  130 NOFET2=NOFEAT                                                             SET01460
      NOFET4=LCOMB                                                              SET01470
      DO 140 B=1,NOFEAT                                                         SET01480
  140 FETVC2(B)=FETVEC(B)                                                       SET01490
      GO TO 50                                                                  SET01500
C FEATURE CARD                                                                  SET01510
  150 CONTINUE                                                                  SET01520
      GO TO 50                                                                  SET01530
C FORMAT CARD                                                                   SET01540
  160 CONTINUE                                                                  SET01550
  170 M=FIND12(CARD2,COL,FRVEC1)                                                SET01560
      IF (M.EQ.-1) GO TO 540                                                    SET01570
      KZ=FIND12(CARD2,COL,SINVEC)                                               SET01580
```

FILE: SETUP8 FORTRAN A

```
      KM=FIND12(CARD2,COL,FRVEC2)                                       SET01590
      IF((M.EQ.3).AND.(KM.EQ.2)) OUTFMT=1                               SET01600
      IF((M.EQ.3).AND.(KM.EQ.3)) OUTFMT=2                               SET01610
      Z=FIND12(CARD2,COL,SINVEC)                                        SET01620
      IF (Z.NE.2) GO TO 50                                              SET01630
      GO TO 170                                                         SET01640
C HED1 CARD                                                             SET01650
  180 READ (30,220)HED1                                                 SET01660
      REWIND 30                                                         SET01670
      GO TO 50                                                          SET01680
C HED2 CARD                                                             SET01690
  190 READ (30,220)HED2                                                 SET01700
      REWIND 30                                                         SET01710
      GO TO 50                                                          SET01720
C COMMENT CARD                                                          SET01730
  200 READ (30,220)COMENT                                               SET01740
      REWIND 30                                                         SET01750
      GO TO 50                                                          SET01760
C DATE CARD                                                             SET01770
  210 M=NXTCHR(CARD2,COL)                                               SET01780
      IF (M.EQ.BLANK) GO TO 50                                          SET01790
      READ (30,220)DATE                                                 SET01800
  220 FORMAT(10X,15A4)                                                  SET01810
      REWIND 30                                                         SET01820
      GO TO 50                                                          SET01830
C MAXPT CARD                                                            SET01840
  230 J=NXTCHR(CARD2,COL)                                               SET01850
      IF (J.EQ.BLANK) GO TO 540                                         SET01860
      COL=COL-1                                                         SET01870
      MPT = NUMBER( CARD2, COL, MAXPT, MPT )                            SET01880
C                                                                       SET01890
      IF (MPT.GT.30) GO TO 90                                           SET01900
C                                                                       SET01910
      GO TO 50                                                          SET01920
C PEROUT CARD                                                           SET01930
  250 J=NXTCHR(CARD2,COL)                                               SET01940
      IF (J.EQ.BLANK) GO TO 540                                         SET01950
      COL=COL-1                                                         SET01960
      M = NUMBER ( CARD2, COL, ARRAY, ZERO )                            SET01970
      PEROUT = ARRAY(1)                                                 SET01980
C                                                                       SET01990
      IF (M.NE.1) GO TO 90                                              SET02000
C                                                                       SET02010
      GO TO 50                                                          SET02020
C SUBCLASS CARD                                                         SET02030
  270 NOSUB2=NUMBER(CARD2,COL,SUBVC2,NOSUB2)                            SET02040
      CALL ORDER(SUBVC2,NOSUB2)                                         SET02050
      GO TO 50                                                          SET02060
C LAM CARD                                                              SET02070
  280 J=NXTCHR(CARD2,COL)                                               SET02080
      IF (J.EQ.BLANK) GO TO 540                                         SET02090
      COL=COL-1                                                         SET02100
      M = NUMBER ( CARD2, COL, ARRAY, ZERO )                            SET02110
      LAM = ARRAY(1)                                                    SET02120
C                                                                       SET02130
      IF (M.NE.1) GO TO 90                                              SET02140
C                                                                       SET02150
      GO TO 50                                                          SET02160
C                                                                       SET02170
CCCC                                                                    SET02180
CCCC     OPTION  CARD                                                   SET02190
CCCC                                                                    SET02200
  290 M=FIND12(CARD2,COL,MTX)                                           SET02210
C                                                                       SET02220
      M = IABS( M )                                                     SET02230
C                                                                       SET02240
      IF (M.EQ.0.OR.M.GT.5) GO TO 540                                   SET02250
C                                                                       SET02260
CCCCC                                                                   SET02270
CCCCC    IF  M = 1,  END-OF-CARD  HAS BEEN REACHED                      SET02280
CCCCC                                                                   SET02290
      GO TO (50,300,310,320,350),M                                      SET02300
C                                                                       SET02310
CCCCC                                                                   SET02320
CCCCC    IF  M = 2,  ''O''  ,  OR  ''ORIG''                             SET02330
CCCCC                                                                   SET02340
  300 ORIG = 1                                                          SET02350
C                                                                       SET02360
                                                                        SET02370
```

```
FILE: SETUP8   FORTRAN  A

      M = FIND12( CARD2, COL, SINVEC )                                  SET02380
C                                                                       SET02390
      IF (M.EQ.2) GO TO 290                                             SET02400
C                                                                       SET02410
      GO TO 50                                                          SET02420
C                                                                       SET02430
CCCCC                                                                   SET02440
CCCCC    IF  M = 3,  ''T''  OR  ''TRANSF''                              SET02450
CCCCC                                                                   SET02460
CCCCC                                                                   SET02470
  310 TRANSF = 1                                                        SET02480
C                                                                       SET02490
      M = FIND12( CARD2, COL, SINVEC )                                  SET02500
C                                                                       SET02510
      IF (M.EQ.2) GO TO 290                                             SET02520
C                                                                       SET02530
      GO TO 50                                                          SET02540
C                                                                       SET02550
CCCCC                                                                   SET02560
CCCCC     IF  M = 4,   ' S'  ---   CHECK FOR  ''SCAFAC=''               SET02570
CCCCC                                                                   SET02580
  320 J = NXTCHR ( CARD2, COL )                                         SET02590
C                                                                       SET02600
CCCCC                                                                   SET02610
CCCCC    IF NEXT CHARACTER IS  ''C''  ,  ASSUME   ''SCAFAC''            SET02620
CCCCC                                                                   SET02630
      IF (J .NE. CBCD) GO TO 540                                        SET02640
C                                                                       SET02650
      Z = FIND12( CARD2, COL, SINVEC )                                  SET02660
C                                                                       SET02670
      IF (Z.EQ.3) GO TO 330                                             SET02680
C                                                                       SET02690
      GO TO 540                                                         SET02700
C                                                                       SET02710
CCCCC                                                                   SET02720
CCCCC    SCALE FACTOR OPTION :  READ SCALING PAIRS, CON AND MIN , INTO  SET02730
CCCCC                           CONMIN                                  SET02740
CCCCC                                                                   SET02750
  330 SCAFLG = 3                                                        SET02760
C                                                                       SET02770
  340 Z = FIND12( CARD2, COL, UP )                                      SET02780
C                                                                       SET02790
      IF (Z.NE.2) GO TO 50                                              SET02800
C                                                                       SET02810
      NMN =  FLTNUM ( CARD2, COL, CONMIN(NSF) , 2 )                     SET02820
C                                                                       SET02830
      IF (NMN.NE.2) GO TO 540                                           SET02840
C                                                                       SET02850
      ADDNUM = NSF + 1                                                  SET02860
      IF ((NSF+NMN).GT.31) GO TO 50                                     SET02870
C                                                                       SET02880
      NSF = NSF + NMN                                                   SET02890
C                                                                       SET02900
      Z = FIND12( CARD2, COL, CP )                                      SET02910
C                                                                       SET02920
      IF (Z.EQ.2) GO TO 340                                             SET02930
C                                                                       SET02940
      GO TO 540                                                         SET02950
C                                                                       SET02960
CCCCC                                                                   SET02970
CCCCC    PUNCH  OPTION                                                  SET02980
CCCCC                                                                   SET02990
  350 NPUN =  1                                                         SET03000
C                                                                       SET03010
C                                                                       SET03020
      GO TO 290                                                         SET03030
C                                                                       SET03040
CCC MODULE STAT DECK                                                    SET03050
  360 MK=NXTCHR(CARD2,COL)                                              SET03060
      IF (MK.NE.BTEST(3)) GO TO 370                                     SET03070
C                                                                       SET03080
      SCAFLG = 2                                                        SET03090
C                                                                       SET03100
      GO TO 50                                                          SET03110
  370 CALL CRDSTA(ARRAY,TOP)                                            SET03120
C                                                                       SET03130
      SCAFLG = 2                                                        SET03140
                                                                        SET03150
                                                                        SET03160
```

FILE: SETUP8 FORTRAN A

```
C                                                                               SET03170
      GO TO 50                                                                  SET03180
C                                                                               SET03190
C DATAFILE POSITIONING CARD                                                     SET03200
C                                                                               SET03210
  380 IF (NUDTAP.AND.NUDFIL) GO TO 50                                           SET03220
C                                                                               SET03230
      M = NXTCHR ( CARD2 , COL )                                                SET03240
C                                                                               SET03250
      IF (M.EQ.BLANK) GO TO 50                                                  SET03260
C                                                                               SET03270
      IF (M .EQ. URCD) GO TO 410                                                SET03280
      IF (M .EQ. FRCD) GO TO 420                                                SET03290
  390 WRITE (6,400)                                                             SET03300
  400 FORMAT(///// 5X, '***** DATATP/SETUP8 ***** ERROR ON INPUT DATA          SET03310
C                                                                               SET03320
     *FILE CARD ---  CONTINUING TO PROCESS INPUT   *****' ///// )               SET03330
C                                                                               SET03340
      GO TO 50                                                                  SET03350
  410 J=FIND12(CARD2,COL,EQUVEC)                                                SET03360
      IF (J.EQ.-1) GO TO 390                                                    SET03370
      M=NUMBER(CARD2,COL,DATAPE,ZERO)                                           SET03380
      COL=COL-1                                                                 SET03390
C                                                                               SET03400
      IF (M.NE.1) GO TO 390                                                     SET03410
C                                                                               SET03420
      NUDTAP =  .TRUE.                                                          SET03430
C                                                                               SET03440
      GO TO 380                                                                 SET03450
  420 J=FIND12(CARD2,COL,EQUVEC)                                                SET03460
      IF (J.EQ.-1) GO TO 390                                                    SET03470
      FILNO = NUMBER ( CARD2, COL, DATFIL, ZERO )                               SET03480
C                                                                               SET03490
      IF (FILNO.NE.1) GO TO 390                                                 SET03500
C                                                                               SET03510
      NUDFIL =  .TRUE.                                                          SET03520
C                                                                               SET03530
      DATFIL=DATFIL-1                                                           SET03540
      COL=COL-1                                                                 SET03550
      GO TO 380                                                                 SET03560
C                                                                               SET03570
C STATFILE POSITIONING CARD                                                     SET03580
C                                                                               SET03590
  430 M=NXTCHR(CARD2,COL)                                                       SET03600
      IF (M.EQ.BLANK) GO TO 50                                                  SET03610
C                                                                               SET03620
      IF (M .EQ. URCD) GO TO 460                                                SET03630
      IF (M .EQ. FRCD) GO TO 470                                                SET03640
  440 WRITE (6,450)                                                             SET03650
  450 FORMAT( ///// 5X,'***** DATATR/SETUP8 *****  ERROR ON INPUT OR OUT       SET03660
C                                                                               SET03670
     *PUT CARD  ---  CONTINUING TO PROCESS INPUT  *****' /// )                  SET03680
C                                                                               SET03690
      NUSTAP = .FALSE.                                                          SET03700
      NUSFIL = .FALSE.                                                          SET03710
C                                                                               SET03720
      GO TO 50                                                                  SET03730
C                                                                               SET03740
  460 J=FIND12(CARD2,COL,EQUVEC)                                                SET03750
      IF (J.EQ.-1) GO TO 440                                                    SET03760
      M=NUMBER(CARD2,COL,SAVTAP,ZERO)                                           SET03770
      COL=COL-1                                                                 SET03780
C                                                                               SET03790
      IF (M.NE.1) GO TO 440                                                     SET03800
C                                                                               SET03810
C*****NUSTAP =  .TRUE.                                                          SET03820
C                                                                               SET03830
      GO TO 430                                                                 SET03840
  470 J=FIND12(CARD2,COL,EQUVEC)                                                SET03850
      IF (J.EQ.-1) GO TO 440                                                    SET03860
      FILST = NUMBER( CARD2, COL, STAFIL, ZERO )                                SET03870
C                                                                               SET03880
      IF (FILST.NE.1) GO TO 440                                                 SET03890
C                                                                               SET03900
C*****NUSFIL =  .TRUE.                                                          SET03910
C                                                                               SET03920
      STAFIL=STAFIL-1                                                           SET03930
      COL=COL-1                                                                 SET03940
      GO TO 430                                                                 SET03950
```

FILE: SETUP8 FORTRAN A

```
C                                                                       SET03960
C     RESCALE OPTION CARD                                               SET03970
C                                                                       SET03980
  480  CONTINUE                                                         SET03990
       RESCAL = 1                                                       SET04000
      GO TO 50                                                          SET04010
C                                                                       SET04020
C*** CODE ADDED JAN. 18,1979 TO OUTPUT MULTI-FILE OUTPUT                SET04030
C                                                                       SET04040
 600          M = NXTCHR(CARD2,COL)                                     SET04050
              IF (M.EQ.BLANK) GO TO 50                                  SET04060
              IF (M.EQ.UBCD) GO TO 660                                  SET04070
              IF (M.EQ.FBCD) GO TO 670                                  SET04080
 660          J = FIND12(CARD2,COL,EQUVEC)                              SET04090
              IF (J.EQ.-1) GO TO 440                                    SET04100
              M = NUMBER(CARD2,COL,TRFORM,ZERO)                         SET04110
              COL = COL -1                                              SET04120
              IF (M.NE.1) GO TO 440                                     SET04130
              GO TO 600                                                 SET04140
 670          J = FIND12(CARD2,COL,EQUVEC)                              SET04150
              IF (J.EQ.-1) GO TO 440                                    SET04160
              M = NUMBER(CARD2,COL,NF,ZERO)                             SET04170
              COL = COL - 1                                             SET04180
                GO TO 50                                                SET04190
C     BIAS CONTROL CARD                                                 SET04200
C                                                                       SET04210
  490  CONTINUE                                                         SET04220
         NB = NXTCHR(CARD2, COL)                                        SET04230
       IF (NB.EQ.BLANK) GO TO 540                                       SET04240
         COL = COL -1                                                   SET04250
       VECMAX = 15 - NBS                                                SET04260
        NK = NBS + 1                                                    SET04270
        NBS = FLTNUM (CARD2, COL, BIAS(NK), VECMAX)                     SET04280
       GO TO 50                                                         SET04290
C                                                                       SET04300
C *END* CARD                                                            SET04310
  500 CONTINUE                                                          SET04320
C                                                                       SET04330
C     IF ( RESCAL .GT. 0  .AND.  SCAFLG .EQ. 0 )   SCAFLG = 1           SET04340
C                                                                       SET04350
CCCCCC                                                                  SET04360
CCCCCC   IF RESCALING BY THE STATISTICAL METHOD, READ STATISTICS FROM   SET04370
CCCCCC   FILE ( SAVTAP ) , REDUCE THE STATISTICS TO THE SET OF CHANNELS SET04380
CCCCCC   SPECIFIED IN FETVC2 , AND STORE IN ARRAY .                     SET04390
CCCCCC                                                                  SET04400
C                                                                       SET04410
C     IF ( SCAFLG .EQ. 2 )  CALL REDSAV( ARRAY, TOP, BMSWT )            SET04420
C                                                                       SET04430
CC                                                                      SET04440
C     IF (ORIG.EQ.0) GO TO 530                                          SET04450
C                                                                       SET04460
CC                                                                      SET04470
C     DO 510 I=1,15                                                     SET04480
      TEMP = COMENT(I)                                                  SET04490
      COMENT(I) = COVHD1(I)                                             SET04500
  510 COVHD1(I) = TEMP                                                  SET04510
C                                                                       SET04520
      CALL PRTCOV(ARRAY(COVAR2),ARRAY(AVAR2),VARSZ2,NOFET2,ARRAY(SUBOS2)SET04530
     *)                                                                 SET04540
C                                                                       SET04550
CC                                                                      SET04560
C     DO 520 I=1,15                                                     SET04570
  520 COMENT(I) = COVHD1(I)                                             SET04580
C                                                                       SET04590
CCCCCC                                                                  SET04600
CCCCCC   PRINT OUT THE INPUT TRANSFORMATION MATRIX                      SET04610
CCCCCC                                                                  SET04620
C                                                                       SET04630
C                                                                       SET04640
  530 CALL WRTBM(RMAT,NOFET4,NOFET2,FETVC2)                             SET04650
C SET NOCLS2=NOSUB2 FOR REST OF PROGRAM                                 SET04660
      NOCLS2=NOSUB2                                                     SET04670
C                                                                       SET04680
C     IF ( RESCAL .EQ. 0 )  SCAFLG = 0                                  SET04690
C                                                                       SET04700
C     RETURN                                                            SET04710
C                                                                       SET04720
CC                                                                      SET04730
C                                                                       SET04740
```

FILE: SETUP8 FORTRAN A

```
  540 WRITE (6,550)CODE,CARD2                                                    SET04750
C                                                                                SET04760
  550 FORMAT(//// 5X,'*****  INVALID CONTROL CARD REJECTED BY DATATM/SETSET04770
C                                                                                SET04780
     *UP8  *****' // 5X, A4,6X, 62A1 //// )                                      SET04790
C                                                                                SET04800
C                                                                                SET04810
C                                                                                SET04820
      GO TO 50                                                                   SET04830
      END                                                                        SET04840
```

FILE: TRANSF FORTRAN A

```
      SUBROUTINE TRANSF                                                 TRA00010
     *     (XT, RMAT, IDATA, TOP, IL, K, LCOMB, NSAMP, BIAS)            TRA00020
      IMPLICIT INTEGER (A-Z)                                            TRA00030
      REAL   XT(16) ,  RMAT(480) ,  BIAS(16)                            TRA00040
C                                                                       TRA00050
C     INCLUDE COMMON, LIST                                              TRA00060
C DATA TRANSFORMATION COMMON BLOCK
      COMMON/TRBLCK/OUTFMT,NOFEAT,FLDINF(6),          FETVEC(30)
C$END                                                                   TRA00080
      DIMENSION IDATA (TOP)                                             TRA00090
C                                                                       TRA00100
CCCCCCCCCC    SUBROUTINE TRANSF DOES A DATA-TRANSFORMATION USING THE    TRA00110
CCCCCCCCCC    FORMULA                                                   TRA00120
CCCCCCCCCC          XT = IDATA * RMAT + BIAS                            TRA00130
CCCCCCCCCC          XT = COMPONENT(IL), TRANSFORMED DATA VECTOR         TRA00140
CCCCCCCCCC          IDATA = INPUT DATA VECTOR ( NOFEAT X 1 )            TRA00150
CCCCCCCCCC          RMAT = TRANSFORMATION MATRIX ( LCOMB X NOFEAT )     TRA00160
CCCCCCCCCC            BIAS = ADDITIVE BIAS                              TRA00170
C                                                                       TRA00180
      DO 10 IT=1,NOFEAT                                                 TRA00190
           JSAMP = (IT - 1) * NSAMP + K                                 TRA00200
           ZCOMB = LCOMB * (IT - 1) + IL                                TRA00210
           XT (IL) = XT (IL) +                                          TRA00220
C                IDATA(JSAMP) * RMAT(ZCOMB)                           TRA00230
   10  CONTINUE                                                         TRA00240
       XT(IL) = XT(IL) + BIAS(IL)                                       TRA00250
       RETURN                                                           TRA00260
       END                                                              TRA00270
```

FILE: TRHIST FORTRAN A

```
      SUBROUTINE TRHIST(IDATA,AMAX,AMIN,ACON,BMAT,LCOMB,
     *PEROUT,FILHIS,TOP,LAR,FLDNAM,NC,VERTCS,MAX,MIN,CON,
     *     BIAS)
C
C  HISTOGRAM THE TRANSFORMED DATA AND CALCULATE THE MIN MAX AND RANGE
C  FOR THIS DATA TO ALLOW RESCALING IN THE 0-255 RANGE
C
      IMPLICIT INTEGER(A-Z)
C
      REAL   BIAS(16), XT(16), PERCEN(16), MIN(16), MAX(16), CON(16)
C
      REAL   BMAT(480), AMIN(16), AMAX(16), ACON(16)
C
      REAL   XMIN , XPER ,  SUMFIL ,  DUMMY
C
C
      INCLUDE COMBA1.LIST
      INCLUDE COMBK3.LIST
      INCLUDE COMBK4.LIST
      INCLUDE COMBK6.LIST
      COMMON/INFORM/NUCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFLD2,
     *              AVAR2,COVAR2,CLSID2,SUBN02,SUMDS2,FLDSV2,VERTX2,
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),
     *              KEYPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),
     *              GRPCHR(61),GROUPS(124)
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))
      COMMON/GLOBAL/HEAD(53),MAPTAP,DATAPE,SAVTAP,HMFILE,BMKEY,
     *              HISFIL,HISKEY,TRFORM,EPIPTP,EPPKEY,MAPUNT,NOFILE,
     *     DRUMAD,DUMDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL
     *   ,NMSTUN,NMSTFI,SCTHUN,MAPFIL
     *   ,DDTUNT,DDTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,
     *   CRDUNT,PRTUNT,RANDIO
C DATA TRANSFORMATION COMMON BLOCK
      COMMON/TRFLCM/OUTFMT,NOFEAT,FLDINF(6),          FETVEC(30)
C$END
      DIMENSION  FILHIS(LCOMB,101), IDATA(TOP), TOTPTS(16) ,
     * VERTCS(2,11),FL(8),FLDINP(6)
C
C
C         READ THE  COORDINATES ( VERTICES ) OF THE FIELD, FOR THE
C        DATA TO BE TRANSFORMED
C
C
   10 LAR=LAREAD(FLDNAM,VERTCS,FLDINF,NC)
      IF (LAR.EQ.0) GO TO 210
      IF (LAR.LE.-1) GO TO 10
C
C
C         POSITION THE INPUT DATA TAPE AND READ IN THE HEADER RECORD
C
C
      CALL TAPHDR( DATAPE , DATFIL )
C
      DO 20 I=1,LCOMB
      MAX(I)=AMAX(I)
      MIN(I)=AMIN(I)
   20 CON(I)=ACON(I)
      NSAMP=(FLDINF(6)-FLDINF(4))/FLDINF(6)+1
      LINES=(FLDINF(2)-FLDINF(1))/FLDINF(3)+1
      DUMMY = (LINES*NSAMP)/2000
      ALP=SQRT(DUMMY)
      IF(ALP.LE.1) ALP=1
      FLDINP(1)=FLDINF(1)
      FLDINP(2)=FLDINF(2)
      FLDINP(4)=FLDINF(4)
      FLDINP(5)=FLDINF(5)
      FLDINP(3)=ALP
      FLDINP(6)=ALP
      LINES=(FLDINP(2)-FLDINP(1))/FLDINP(3)+1
      NSAMP=(FLDINP(5)-FLDINP(4))/FLDINP(6)+1
      CALL FLDIFT(FLDINP,FETVEC,NOFEAT)
      DO 30 I=1,LCOMB
      TOTPTS(I) = 0
      DO 30 J=1,101
   30 FILHIS(I,J)=0.
      DO 130 I=1,LINES
      CALL LINERD(IDATA,ENDTAP)
      IF (ENDTAP.NE.1) GO TO 140
```

FILE: TRHIST FORTRAN A

```
      IF (I.NE.1) GO TO 40                                                      TRH00800
      ILIN=FLOINP(1)                                                            TRH00810
      GO TO 50                                                                  TRH00820
   40 ILIN=ILIN+FLOINP(3)                                                       TRH00830
   50 CONTINUE                                                                  TRH00840
      CALL FOLINT(VERTCS,NC,FL,ILIN,NS,JJ)                                      TRH00850
      DO 110 K=1,NSAMP                                                          TRH00860
      KP=(K-1)*FLOINP(6)+FLOINP(4)                                              TRH00870
      DO 100 LK=1,JJ,2                                                          TRH00880
      LKP1 = LK + 1                                                             TRH00890
      IF (KP.LT.FL(LK)) GO TO 110                                               TRH00900
      IF (KP.GT.FL(LKP1)) GO TO 90                                              TRH00910
      DO 80 J=1,LCOMB                                                           TRH00920
      XT(J)=0.                                                                  TRH00930
C                                                                               TRH00940
CCC       CALL TRANSF TO DO A DATA TRANSFORMATION                               TRH00950
C                                                                               TRH00960
        CALL TRANSF                                                             TRH00970
     C      (XT, HMAT, IDATA, IOP, J, K, LCOMB, NSAMP, BIAS)                   TRH00980
C                                                                               TRH00990
CCC                                                                             TRH01000
CCC       HISTOGRAM THE TRANSFORMED DATA ( USING TRANSFORMED DATA MAX           TRH01010
CCC       AND MIN AND SCALE FACTOR, CON , COMPUTED IN SUBR. MAXMAT              TRH01020
CCC       TO OBTAIN THE HISTOGRAM  **BIN LEVEL**  FOR EACH TRANSFORMED          TRH01030
CCC       DATA POINT )                                                          TRH01040
CCC                                                                             TRH01050
C                                                                               TRH01060
      IF (XT(J).LE.MIN(J)) GO TO 60                                             TRH01070
      IF (XT(J).GE.MAX(J)) GO TO 70                                             TRH01080
      IOPT=(XT(J)-MIN(J))/CON(J)+1                                              TRH01090
C                                                                               TRH01100
      IF( IOPT .LE. 0)  IOPT = 1                                                TRH01110
      IF ( IOPT .GT. 101 )  IOPT = 101                                          TRH01120
C                                                                               TRH01130
      FILHIS(J,IOPT)=FILHIS(J,IOPT)+1                                           TRH01140
      GO TO 80                                                                  TRH01150
   60 FILHIS(J,1)=FILHIS(J,1)+1                                                 TRH01160
      GO TO 80                                                                  TRH01170
   70 FILHIS(J,101) = FILHIS(J,101) + 1                                         TRH01180
   80 TOTPTS(J)=TOTPTS(J)+1                                                     TRH01190
      GO TO 110                                                                 TRH01200
   90 IF (LKP1.GE.JJ) GO TO 120                                                 TRH01210
  100 CONTINUE                                                                  TRH01220
  110 CONTINUE                                                                  TRH01230
  120 CONTINUE                                                                  TRH01240
  130 CONTINUE                                                                  TRH01250
  140 CONTINUE                                                                  TRH01260
C                                                                               TRH01270
CCC                                                                             TRH01280
CCC       ELIMINATE PEROUT/2.    OF POINTS FROM UPPER AND LOWER TAILS OF        TRH01290
CCC       THE TRANSFORMED DATA DISTRIBUTION --- OBTAIN THE REVISED MAX          TRH01300
CCC       AND SCALING PARAMETERS CON AND MIN AFTER APPLICATION OF PEROUT        TRH01310
CCC                                                                             TRH01320
C                                                                               TRH01330
      XPER=PEROUT/200.                                                          TRH01340
C CALCULATE MIN,MAX,AND CON ARRAYS                                              TRH01350
      DO 150 I=1,LCOMB                                                          TRH01360
  150 PERCEN(I)=XPER*TOTPTS(I)                                                  TRH01370
      DO 200 I=1,LCOMB                                                          TRH01380
      XMIN=MIN(I)                                                               TRH01390
      SUMFIL=0.                                                                 TRH01400
      DO 160 J=1,101                                                            TRH01410
      SUMFIL=SUMFIL+FILHIS(I,J)                                                 TRH01420
      IF (SUMFIL.LT.PERCEN(I)) GO TO 160                                        TRH01430
      MIN(I)=(J-1)*CON(I)+XMIN                                                  TRH01440
      GO TO 170                                                                 TRH01450
  160 CONTINUE                                                                  TRH01460
  170 SUMFIL=0.                                                                 TRH01470
      J=101                                                                     TRH01480
  175 J=J-1                                                                     TRH01490
      SUMFIL=SUMFIL+FILHIS(I,J)                                                 TRH01500
      IF (SUMFIL.LT.PERCEN(I)) GO TO 180                                        TRH01510
      JM1 = J - 1                                                               TRH01520
      MAX(I) = JM1 * CON(I) + XMIN                                              TRH01530
      GO TO 190                                                                 TRH01540
  180 IF(J.GT.1) GO TO 175                                                      TRH01550
  190  CONTINUE                                                                 TRH01560
       CON(I) = 255. / (MAX(I) - MIN(I))                                        TRH01570
  200 CONTINUE                                                                  TRH01580
```

FILE: TRHIST FORTRAN A

```
C                                                                       TRH01590
C                                                                       TRH01600
C       RETURN THE SCALING PARAMETERS, CON AND MIN , REQUIRED TO        TRH01610
C       RESCALE THE TRANSFORMED DATA TO THE RANGE , 0 - 255  .          TRH01620
C                                                                       TRH01630
C                                                                       TRH01640
  210 RETURN                                                            TRH01650
      END                                                               TRH01660
```

14. TRSTAT PROCESSOR

FILE: TRSTAT FORTRAN A

```
      SUBROUTINE TRSTAT(ARRAY,TOP)                                      TRS00010
      IMPLICIT INTEGER(A-Z)                                             TRS00020
      DIMENSION ARRAY(TOP)                                              TRS00030
C     INCLUDE COMAK1.LIST                                               TRS00040
C     INCLUDE COMAK6.LIST                                               TRS00050
      COMMON/INFO##/NOCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFLD2,          COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        COM00030
     *              FERRTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             COM00040
     *              GRPCHK(61),GROUPS(124)                              COM00050
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         COM00010
     *              HISFIL,HISKEY,TRFORM,ERIPTR,ERRKEY,MAPUNT,NUFILE,   COM00020
     *    ORIMAD,DUMMDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                COM00030
     *  ,NHSTUN,NHSTFI,SCTHUN,MAPFIL                                    COM00040
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              COM00050
     *  CRDUNT,PRTUNT,RANDIO                                            COM00060
CSEND                                                                   TRS00070
      REAL AMAT(1900),B(30)                                             TRS00080
      CALL SETUP9(ARRAY,TOP,AMAT,ROW,IP,TRAN,B)                         TRS00090
      CALL TRAMTX(ARRAY,TOP,AMAT,ROW,IP,TRAN,B)                         TRS00100
      RETURN                                                            TRS00110
      END                                                               TRS00120
```

FILE: AMFIL FORTRAN A

```
      SUBROUTINE AMFIL(ROW,COLUMN,AMAT,VEC,B)                          AMF00010
      IMPLICIT INTEGER(A-Z)                                            AMF00020
C     INCLUDE COMMS                                                    AMF00030
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,AMFILE,BMKEY,        AMF00040
     *              HISFIL,HISKEY,TRFORM,EHIPTP,ERPKEY,MAPUNT,NOFILF,  AMF00050
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               AMF00060
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                   AMF00070
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,             AMF00080
     *  CRDUNT,PRTUNT,HANDIO                                           AMF00090
C$END                                                                  AMF00100
      REAL AMAT(1),B(30)                                               AMF00110
      DIMENSION VEC(1)                                                 AMF00120
    1 READ(21,2) ROW,COLUMN,(VEC(I),I=1,COLUMN)                        AMF00130
      IK=ROW*COLUMN                                                    AMF00140
      READ(21,3) (AMAT(I),I=1,IK)                                      AMF00150
      READ(21,3) (B(I),I=1,ROW)                                        AMF00160
      RETURN                                                           AMF00170
    2 FORMAT(5X,I2,5X,I2,2X,30I2)                                      AMF00180
    3 FORMAT(5X,5E15.8)                                                AMF00190
      END                                                              AMF00200
```

FILE: AMFILE FORTRAN A

```
      SUBROUTINE AMFILE(ROW,NOCHAN,CHNVEC,AMAT,BVEC)                    AMF00010
                                                                        AMF00020
C*                                                                      AMF00030
C*    AMFILE WILL READ INTO CORE THE A-MATRIX AND B VECTOR FROM UNIT     AMF00040
C*    7                                                                 AMF00050
C*                                                                      AMF00060
      IMPLICIT INTEGER (A-Z)                                            AMF00070
      REAL AMAT(1),BVEC(1)                                              AMF00080
C     INCLUDE COMRK6                                                    AMF00090
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,         AMF00100
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,   AMF00110
     *    DRUMAD,DRM4DS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                AMF00120
     *  ,NHSTUN,NHSTFI,SCTHUN,MAPFIL                                    AMF00130
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              AMF00140
     *  CROUNT,PRTUNT,RANDIO                                            AMF00150
CSEND                                                                   AMF00160
      DIMENSION CHNVEC(1)                                               AMF00170
      READ(7) ROW,NOCHAN,(CHNVEC(I),I=1,NOCHAN)                         AMF00180
      TOTAL = NOCHAN**2                                                 AMF00190
      READ(7) (  AMAT(I),I=1,TOTAL),(BVEC(I),I=1,NOCHAN)                AMF00200
C                                                                       AMF00210
      RETURN                                                            AMF00220
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FILE: SETUP9   FORTRAN  A

      SUBROUTINE SETUP9(ARRAY,TOP,AMAT,ROW,IP,THAN,B)                          SET00010
      IMPLICIT INTEGER(A-Z)                                                    SET00020
      REAL AMAT(1800),B(1)                                                     SET00030
C                                                                              SET00040
      DIMENSION ARRAY(TOP),ACARD(20)                                           SET00050
      DIMENSION CHNVEC(30)                                                     SET00060
      DIMENSION CARD(62),CINDEX(12),SINVEC(3),SLASH(2)                         SET00070
C     INCLUDE COMMK1,LIST                                                      SET00080
CC    INCLUDE COMMK4,LIST                                                      SET00090
C     INCLUDE COMMK6,LIST                                                      SET00100
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,                 SET00110
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,           SET00120
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),               SET00130
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                    SET00140
     *              GRPCHK(61),GROUPS(124)                                     SET00150
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                           SET00160
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                        SET00170
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                      SET00180
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,                SET00190
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,          SET00200
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                       SET00210
     *  ,NHSTUN,NHSTFI,SCTHUN,MAPFIL                                           SET00220
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                     SET00230
     *  CRDUNT,PRTUNT,RANDIO                                                   SET00240
CSEND                                                                          SET00250
      DATA BLANK/' '/,EQUAL/'='/,F/'F'/,                                       SET00260
     *  SINVEC/2,',',','='/    ,SLASH/1,'/'/,                                  SET00270
     * P/'P'/,O/'0'/,T/'T'/                                                    SET00280
      DATA CINDEX/'HED1','HED2','COMM',                                        SET00290
     *'DATE','SUBC','A-MA','OPTT',                                           SET00300
     *'MODU','STAT','CHAN','*END','SEND'/                                     SET00310
      DATA  FBCD/'F'/,IBCD/'I'/,OBCD/'O'/,UBCD/'U'/                            SET00320
      CINMAX=12                                                                SET00330
      NOGRP=0                                                                  SET00340
      NOSUB2=0                                                                 SET00350
      NOFET2=0                                                                 SET00360
      BMSWT=1                                                                  SET00370
      ASAV=9                                                                   SET00380
      ASAVFL=0                                                                 SET00390
C                                                                              SET00400
CC    SETUP REREAD BUFFER                                                      SET00410
C                                                                              SET00420
   13 COL=0                                                                    SET00430
C                                                                              SET00440
CC    READ A CARD INTO THE BUFFER                                              SET00450
C                                                                              SET00460
      READ(2,200)(ACARD(I),I=1,20)                                             SET00470
  200 FORMAT(20A4)                                                             SET00480
      WRITE(30,200)(ACARD(I),I=1,20)                                           SET00490
      REWIND 30                                                                SET00500
      READ(30,14) CODE,CARD                                                    SET00510
   14 FORMAT(A4,6X,62A1)                                                       SET00520
      REWIND 30                                                                SET00530
      WRITE(6,15) CODE,CARD                                                    SET00540
   15 FORMAT(T5,A4,6X,62A1)                                                    SET00550
      DO 12 I=1,CINMAX                                                         SET00560
      IF (CINDEX(I) .EQ. CODE) GO TO (2,3,4,5,6,8,9,11,20,40,60,80),I          SET00570
   12 CONTINUE                                                                 SET00580
      WRITE(6,31) CODE,CARD                                                    SET00590
   31 FORMAT(/1X,A4,6X,62A1/'*** BAD SUPERVISOR CONTROL CARD SETUP9***')       SET00600
      GO TO 13                                                                 SET00610
C HED1 CARD                                                                    SET00620
    2 READ(30,24) HED1                                                         SET00630
      REWIND 30                                                                SET00640
      GO TO 13                                                                 SET00650
   24 FORMAT(10X,15A4)                                                         SET00660
C HED2 CARD                                                                    SET00670
    3 READ(30,24) HED2                                                         SET00680
      REWIND 30                                                                SET00690
      GO TO 13                                                                 SET00700
C COMMENT CARD                                                                 SET00710
    4 READ(30,24) COMENT                                                       SET00720
      REWIND 30                                                                SET00730
      GO TO 13                                                                 SET00740
C DATE CARD                                                                    SET00750
    5 READ(30,24) DATE                                                         SET00760
      REWIND 30                                                                SET00770
      GO TO 13                                                                 SET00780
C SUBCLASS CARD                                                                SET00790
```

FILE: SETUP9 FORTRAN A

```
     A NOSUB2=NUMBER(CARD,COL,SUBVC2,NOSUB2)                          SET00800
       CALL ORDER(SUBVC2,NOSUB2),                                     SET00810
       GO TO 13                                                       SET00820
C READ IN A-MATRIX                                                    SET00530
     B CONTINUE                                                       SET00840
       AMTXSW = -1                                                    SET00850
       CALL AMFIL(POW,NOCMAN,AMAT,CHNVEC,B)                           SET00860
       GO TO 13                                                       SET00870
C OPTIONS                                                             SET00880
     9 J=NXTCHR(CARD,COL)                                             SET00890
       IF(J.EQ.BLANK) GO TO 16                                        SET00900
       IF(J.EQ.P) IP=1                                                SET00910
       IF(J.EQ.O) ORIG=1                                              SET00920
       IF(J.EQ.T) TRAN=1                                              SET00930
       M=FIND12(CARD,COL,SINVEC)                                      SET00940
       IF(M.NE.2) GO TO 13                                            SET00950
       GO TO 9                                                        SET00960
C MODULE CARD                                                         SET00970
    11 J=NXTCHR(CARD,COL)                                             SET00980
       IF(J.EQ.F) GO TO 13                                            SET00990
       CALL CRDSTA(ARRAY,TOP)                                         SET01000
       GO TO 13                                                       SET01010
C*                                                                    SET01020
C*     STAT FILE CARD                                                 SET01030
C*                                                                    SET01040
   20  M = NXTCHR(CARD,COL)                                           SET01050
   21  IF (M .EQ. BLANK)  GO TO 13                                    SET01060
       IF (M .EQ. IRCD) GO TO 25                                      SET01070
       IF (M .EQ. ORCD) GO TO 30                                      SET01080
   22  WRITE(4,222)                                                   SET01090
  222  FORMAT(' ERROR ON STAT FILE CARD')                             SET01100
       GO TO 13                                                       SET01110
   25  J = FIND12(CARD,COL,SLASH)                                     SET01120
       IF (J .EQ. -1)  GO TO 22                                       SET01130
   23  M = NXTCHR(CARD,COL)                                           SET01140
       IF (M .EQ. URCD) GO TO 26                                      SET01150
       IF (M .EQ. FRCD) GO TO 28                                      SET01160
       GO TO 21                                                       SET01170
   26  J = FIND12(CARD,COL,SINVEC)                                    SET01180
       IF (J .NE. 3)  GO TO 22                                        SET01190
       M = NUMBER(CARD,COL,SAVTAP,ZERO)                               SET01200
       COL = COL - 1                                                  SET01210
       GO TO 23                                                       SET01220
   28  J = FIND12(CARD,COL,SINVEC)                                    SET01230
       IF (J .NE. 3)  GO TO 22                                        SET01240
       M = NUMBER(CARD,COL,STAFIL,ZERO)                               SET01250
       STAFIL = STAFIL - 1                                            SET01260
       COL = COL - 1                                                  SET01270
       GO TO 23                                                       SET01280
   30  J = FIND12(CARD,COL,SLASH)                                     SET01290
       IF (J .EQ. -1)  GO TO 22                                       SET01300
   32  M = NXTCHR(CARD,COL)                                           SET01310
       IF (M .EQ. URCD) GO TO 34                                      SET01320
       IF (M .EQ. FRCD) GO TO 36                                      SET01330
       GO TO 21                                                       SET01340
   34  J = FIND12(CARD,COL,SINVEC)                                    SET01350
       IF (J .NE. 3)  GO TO 22                                        SET01360
       M = NUMBER(CARD,COL,ASAV,ZERO)                                 SET01370
       COL = COL - 1                                                  SET01380
       GO TO 32                                                       SET01390
   36  J = FIND12(CARD,COL,SINVEC)                                    SET01400
       IF (J .NE. 3)  GO TO 22                                        SET01410
       M = NUMBER(CARD,COL,ASAVFL,ZERO)                               SET01420
       ASAVFL = ASAVFL - 1                                            SET01430
       COL = COL - 1                                                  SET01440
       GO TO 32                                                       SET01450
C*                                                                    SET01460
C*     CHANNEL CARD                                                   SET01470
C*                                                                    SET01480
   40  M = NXTCHR(CARD,COL)                                           SET01490
       IF (M .EQ. BLANK)  GO TO 16                                    SET01500
       COL = COL - 1                                                  SET01510
       NOFET2 = NUMBER(CARD,COL,FETVC2,NOFET2)                        SET01520
       CALL ORDER(FETVC2,NOFET2)                                      SET01530
       GO TO 13                                                       SET01540
C *END* CARD                                                          SET01550
   60  CALL REDSAV(ARRAY,TOP,BMSWT)                                   SET01560
       VARS72=NOFET2*(NOFET2+1)/2                                     SET01570
       IF (AMTXSW .EQ. -1)  GO TO 65                                  SET01580
```

FILE: SETUP9 FORTRAN A

```
      CALL AMFILE(ROW,NOCHAN,CHNVEC,AMAT,B)                                 SET01590
      CALL AMFILE(ROW,NOCHAN,CHNVEC,AMAT,B)                                 SET01600
   65 CONTINUE                                                              SET01610
C WRITE A-MATRIX                                                            SET01620
      CALL WRTAMT(AMAT,ROW,NOFET2,FETVC2,B)                                 SET01630
      IF(ORIG.EQ.1) CALL PRTCOV(ARRAY(COVAR2),ARRAY(AVAR2),VARSZ2,          SET01640
     *                          NOFET2,ARRAY(SUBDS2))                       SET01650
      IF (NOFET2 .EQ. NOCHAN)  GO TO 13                                     SET01660
      WRITE(6,100)NOFET2,NOCHAN                                             SET01670
  100 FORMAT(' NO. OF CHANNELS FROM STAT FILE DOES NOT EQUAL THE'/          SET01680
     *' NO. OF CHANNELS ON A-MATRIX FILE.'/' CHANNELS ON STAT FILE = ',      SET01690
     * I2/' CHANNELS ON A-MATRIX = ', I2)                                   SET01700
      CALL CMERR                                                            SET01710
C SEND CARD                                                                 SET01720
   80 RETURN                                                                SET01730
   16 WRITE(6,19) CARD                                                      SET01740
   19 FORMAT(' INVALID CONTROL CARD REJECTED ***SETUP9***')                 SET01750
      GO TO 13                                                              SET01760
      END
```

FILE: TRAMTX FORTRAN A

```
      SUBROUTINE TRAMTX(ARRAY,TOP,AMAT,ROW,IP,TRAN,B)                    TRA00010
      IMPLICIT INTEGER(A-Z)                                              TRA00020
      REAL C(900),          B(30),AMAT(ROW,NOFET2),AMEAN(1800),          TRA00030
     *ND(15),CC (900)                                                    TRA00040
      INCLUDE COMMK1.LIST                                                TRA00050
C     INCLUDE COMMK4.LIST                                                TRA00060
C     INCLUDE COMMK6.LIST                                                TRA00070
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARS22,TOTVT2,NOFLD2,           TRA00080
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     TRA00090
     *              FETVC2(30),SUMVC2(75),SUBPTR(75),CLSVC2(60),         TRA00100
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              TRA00110
     *              GRPCHK(61),GROUPS(124)                               TRA00120
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                     TRA00130
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                  TRA00140
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                TRA00150
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          TRA00160
     *              HISFIL,HISKEY,TRFORM,ERIPTP,EPPKEY,MAPUNT,NOFILE,    TRA00170
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 TRA00180
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                     TRA00190
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,               TRA00200
     *  CROUNT,PRTUNT,RANDIO                                             TRA00210
CSEND                                                                    TRA00220
      DIMENSION ARRAY(TOP),NSUB(60)                                      TRA00230
      PUNCH=PCHUNT                                                       TRA00240
      DO 41 I=1,NOSUB2                                                   TRA00250
      IPP=SUBDS2+I-1                                                     TRA00260
   41 NSUB(I)=ARRAY(IPP)                                                 TRA00270
      NOFLD=NOFLD2                                                       TRA00280
      REWIND ASAV                                                        TRA00290
C                                                                        TRA00300
      IF (ASAVFL .EQ. 0) GO TO 100                                       TRA00310
C                                                                        TRA00320
C     POSITION STAT FILE                                                 TRA00330
C                                                                        TRA00340
      CALL FSRSFL(ASAV,ASAVFL,ISTAT)                                     TRA00350
      IF(ISTAT .EQ. 0) GO TO 100                                         TRA00360
      FILNO = ASAVFL + 1                                                 TRA00370
      WRITE(6,110)FILNO                                                  TRA00380
  110 FORMAT(' ERROR IN TRYING TO POSITION TRANSFORMED STAT FILE TO BEGI TRA00390
     *NNING OF FILE',I3)                                                 TRA00400
      CALL CMERR                                                         TRA00410
  100 CONTINUE                                                           TRA00420
C                                                                        TRA00430
      WRITE(ASAV) NOCLS2,NOSUB2,ROW,NOFLD,TOTVT2,                        TRA00440
     *            (FETVC2(I),I=1,NOFET2)                                 TRA00450
      IF(IP.NE.1) GO TO 38                                               TRA00460
      WRITE(PUNCH,33)                                                    TRA00470
   33 FORMAT('MODULE    TRAINING FIELD DECK')                          TRA00480
      WRITE(PUNCH,34) NOCLS2,NOSUB2,ROW,NOFLD,TOTVT2                     TRA00490
   34 FORMAT('NOCLS ',I4,' NOSUB  ',I2,' NOFEAT ',I2,' NOFLD ',I3,       TRA00500
     * ' TOTVRT ',I4)                                                    TRA00510
      WRITE(PUNCH,35) (FETVC2(I),I=1,NOFET2)                             TRA00520
   35 FORMAT('CHNVEC',4X,30I2)                                           TRA00530
   38 CONTINUE                                                           TRA00540
      J=FLDSV2                                                           TRA00550
      K=VERTX2                                                           TRA00560
    1 DO 2 I=1,NOFLD                                                     TRA00570
      JJ=J+3                                                             TRA00580
      KK=K+2*ARRAY(JJ)-1                                                 TRA00590
      WRITE(ASAV) (ARRAY(N),N=J,JJ)                                      TRA00600
      WRITE(ASAV) (ARRAY(N),N=K,KK)                                      TRA00610
      IF(IP.NE.1) GO TO 39                                               TRA00620
      WRITE(PUNCH,36) (ARRAY(N),N=J,JJ)                                  TRA00630
   36 FORMAT(A4,6X,I2,8X,I2,8X,I2)                                       TRA00640
      WRITE(PUNCH,37) (ARRAY(N),N=K,KK)                                  TRA00650
   37 FORMAT('VERTICES  ',14I5)                                          TRA00660
   39 CONTINUE                                                           TRA00670
      J=JJ+1                                                             TRA00680
      K=KK+1                                                             TRA00690
    2 CONTINUE                                                           TRA00700
      KK=SUBNO2+NOCLS2-1                                                 TRA00710
      LL=SUBDS2+NOSUB2-1                                                 TRA00720
      WRITE(ASAV)   (ARRAY(J),J=1,NOCLS2),(ARRAY(K),K=SUBNO2,KK),        TRA00730
     *(ARRAY(L),L=SUBDS2,LL)                                             TRA00740
      IF(IP.EQ.0) GO TO 23                                               TRA00750
      WRITE(PUNCH,31) (ARRAY(J),J=1,NOCLS2)                              TRA00760
   31 FORMAT(('CLSDFS  ',2X,A4,R(4X,A4)))                                TRA00770
      WRITE(PUNCH,32) (ARRAY(K),K=SUBNO2,KK)                             TRA00780
   32 FORMAT(('SUBNO  ',24(1X,I2)))                                      TRA00790
```

FILE: TRAMTX FORTRAN A

```
      WRITE(PUNCH,40) (ARRAY(L),L=SUBDS2,LL)                            TRA00800
   40 FORMAT('SUBDES  ',10(A4,3X))                                      TRA00810
   22 CONTINUE                                                          TRA00820
C FOR EACH SUBCLASS COMPUTE TRANS STATS AND OUTPUT THEM                 TRA00830
      J=AVAR2                                                           TRA00840
      K=COVAR2                                                          TRA00850
      L=1                                                               TRA00860
      KKK=1                                                             TRA00870
      DO 20 I=1,NOSUB2                                                  TRA00880
      RR=ROW*(ROW+1)/2                                                  TRA00890
      P=SUBVC2(I)                                                       TRA00900
      KCNT=KKK+RR-1                                                     TRA00910
C MULTIPLY A-MATRIX BY MEAN VECTOR                                      TRA00920
      CALL MATVEC(AMAT,ARRAY(J),AMEAN(L),ROW,NOFET2)                    TRA00930
C ADD B VECTOR TO GET TRANSFORMED MEANS                                 TRA00940
      KM=0                                                              TRA00950
      LL=L+ROW-1                                                        TRA00960
      DO 21 I21=L,LL                                                    TRA00970
      KM=KM+1                                                           TRA00980
   21 AMEAN(I21)=AMEAN(I21)+B(KM)                                       TRA00990
C COMPUTE TRANSFORMED COVAR MATRIX                                      TRA01000
C A-MATR * COVAR * A-MATR TRANSPOSE                                     TRA01010
C MULTIPLY A-MATR * COVAR AND STORE IN C                                TRA01020
      CALL MTMLS6(AMAT,ARRAY(K),C,ROW,NOFET2)                           TRA01030
C MULTIPLY C BY A TRANSPOSE AND STORE IN ARRAY                          TRA01040
      CALL MTMDAT(C,AMAT,CC,ROW,NOFET2,ROW,DD,ARRAY(KKK))               TRA01050
      WRITE(ASAV) KEPPTS(P),(ARRAY(II),II=KKK,KCNT),                    TRA01060
     *            (AMEAN(II),II=L,LL)                                   TRA01070
      IF(IP.EQ.0) GO TO 22                                              TRA01080
      WRITE(PUNCH,95) KEPPTS(P)                                         TRA01090
   95 FORMAT('NOPTS ',6X,I4)                                            TRA01100
      WRITE(PUNCH,96) (AMEAN(II),II=L,LL)                               TRA01110
   96 FORMAT('MEANS',5E15.8)                                            TRA01120
      WRITE(PUNCH,97) (ARRAY(II),II=KKK,KCNT)                           TRA01130
   97 FORMAT('COVAR',5E15.8)                                            TRA01140
   22 CONTINUE                                                          TRA01150
      J=J+ROW                                                           TRA01160
      L=LL+1                                                            TRA01170
      KKK=KKK+RR                                                        TRA01180
      K=K+VARSZ2                                                        TRA01190
   20 CONTINUE                                                          TRA01200
      IF(TRAN.NE.1) RETURN                                              TRA01210
      CV1=ROW*(ROW+1)/2                                                 TRA01220
      CALL PRTCOV(ARRAY(1),AMEAN(1),CV1,ROW,NSUB(1))                    TRA01230
      ENDFILE ASAV                                                      TRA01240
      REWIND ASAV                                                       TRA01250
      END                                                               TRA01260
```

15. NDHIST PROCESSOR

```
FILE: NDHIST   FORTRAN  A

      SUBROUTINE NDHIST(ARRAY,TOP)
C*
C*    NDHIST IS THE DRIVER ROUTINE FOR THE N-DIMENSIONAL PROCESSOR
C*    TWO LARGE ARRAYS ARE USED - BLANK COMMON ARRAY CALLED ARRAY(TOP)
C*                              AND AN ARRAY CALLED HIST(LIMIT)
C
      DATA LIMIT/12000/                                                    CH
C
      DIMENSION HIST(12000)
      DIMENSION ARRAY(1).
C
C     READ IN CONTROL CARDS
C
      CALL SET10(LIMIT)
C
C     NDHST1 IS THE ORGANIZER
      CALL NDHST1(HIST,ARRAY(1),ARRAY(2801),ARRAY(801),
     *     LIMIT,ARRAY(1),TOP)
C
      RETURN
C
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FILE: ADDMES   FORTRAN   A

      SUBROUTINE ADDRES(TOP,NSAMP,NOFEAT,BEGIN,BEGIN1)                          ADD00010
      IMPLICIT INTEGER (A-Z)                                                    ADD00020
C                                                                               ADD00030
CC    INCLUDE CMHR1].LIST                                                       ADD00040
C     INCLUDE COMT1].LIST                                                       ADD00050
      COMMON /ADIM/NCLRCH,CLRVEC(10),MAXVEC,MARKEY,    ,                        ADD00060
     *     CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(6),SIZE,TOTMNS              ADD00070
     *    ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,BUFLEN,ID3,COLOR3,NODUMP          ADD00080
     *    ,IDATA1,TOTVEC                                                       ADD00090
C*    COMMON BLOCK ADIM ,IS USED ONLY BY THE N-DIMENSIONAL HISTOGRAM          ADD00100
C*                          PROCESSOR                                         ADD00110
C*    DEFINITIONS:                                                            ADD00120
C*                                                                            ADD00130
C*    NCLRCH - NO. OF COLOR CHANNELS                                          ADD00140
C*    CLRVEC - ARRAY CONTAINING COLOR CHANNELS                                ADD00150
C*    MAXVEC - MAXIMUM NO. OF VECTORS ARRAY HIST CAN STORE                    ADD00160
C*    CLASS  - KEY INDICATING FIELDS WILL BE GROUPED ON CLASS BASES           ADD00170
C*    SUBCLS - KEY INDICATING FIELDS WILL BE GROUPED ON SUBCLASS BASES        ADD00180
C*    FIELD  - KEY INDICATING FIELDS WILL BE GROUPED ON FIELD  BASES          ADD00190
C*    MEANSW - KEY INDICATING MEANS FOR INPUT FIELDS WILL BE COMPUTED         ADD00200
C*    NOVEC  - NO. OF UNIQUE VECTOR HISTOGRAMMED                              ADD00210
C*    FLDINF - ARRAY CONTAINING RECTANGULAR FIELD COORDINATES AROUND THE      ADD00220
C*             INPUT FIELDS                                                   ADD00230
C*    SIZE   - NUMBER/4 - NO. OF COMPUTER WORDS TO STORE A PACKED             ADD00240
C*             HISTOGRAMMED VECTOR                                            ADD00250
C*    TOTMNS - TOTAL NO. OF ELEMENTS IN ARRAY CONTAINING MEANS                ADD00260
C*    CNTR2  - BEGINNING DRUM ADDRESS FOR STORING FREQUENCY                   ADD00270
C*    ID1    - ADDRESS FOR STORING ID CODES IN ARRAY                          ADD00280
C*    ID2    - BEGINNING DRUM ADDRESS FOR STORING ID CODES                    ADD00290
C*    COLOR1 - ADDRESS FOR STORING COLOR CODES IN ARRAY                       ADD00300
C*    COLOR2 - BEGINNING DRUM ADDRESS FOR STORING COLOR CODES                 ADD00310
C*    BUFLEN - AMOUNT OF STORAGE AVAILABLE FOR STORING ID/COLOR CODES         ADD00320
C*    ID3    - ACCUMULATIVE ID CODE DRUM ADDRESS                              ADD00330
C*    COLOR3 - ACCUMULATIVE COLOR CODE DRUM ADDRESS                           ADD00340
C*    NODUMP - NO OF TIMES ID/COLOR CODES WERE DUMPED ON DRUM                 ADD00350
C*    IDATA1 - ADDRESS FOR STORING IMAGERY DATA IN ARRAY                      ADD00360
C*    TOTVEC - TOTAL NO. OF VECTORS IN THE AREA HISTOGRAMMED                  ADD00370
      CONTINUE                                                                ADD00380
C ARRAYMAP AND HIST COMMON BLOCK                                              ADD00390
CSEND                                                                         ADD00400
C                                                                             ADD00410
C     BASE ADDRESSESS FOR ARRAY                                               ADD00420
C                                                                             ADD00430
      ID1 = 3501                                                              ADD00440
      IDATA1 = TOP - (NSAMP*NOFEAT + 1)                                       ADD00450
      IF (IDATA1 .GT. 8400) GO TO 110                                         ADD00460
      WRITE(6,100)                                                            ADD00470
  100 FORMAT(/ ' TOO MUCH DATA REQUESTED. REDUCE NO. OF SAMPLES PER SCAN      ADD00480
     * LINE'/' AND/OR NO. OF CHANNELS')                                       ADD00490
      CALL CMEH2                                                              ADD00500
  110 CONTINUE                                                                ADD00510
      REMD = 3000                                                             ADD00520
      IF (NCLRCH .EQ. 0) BUFLEN = REMD / 2                                    ADD00530
      IF (NCLRCH .EQ. 0) BUFLEN = REMD                                        ADD00540
      IF (MEANSW .EQ. 1) BUFLEN = REMD                                        ADD00550
      COLOR1 = ID1 + BUFLEN                                                   ADD00560
C                                                                             ADD00570
CC    DRUM ADDRESSES                                                          ADD00580
C                                                                             ADD00590
      CNTR2 = BEGIN                                                           ADD00600
      ID2 = CNTR2 + MAXVEC                                                    ADD00610
      COLOR2 = ID2 + MAXVEC                                                   ADD00620
C                                                                             ADD00630
CC    ORGINAL DRUM START ADDRESSES                                            ADD00640
      BEGIN1 = COLOR2 + MAXVEC                                                ADD00650
C                                                                             ADD00660
      ID3 = ID2                                                               ADD00670
      COLOR3 = COLOR2                                                         ADD00680
C                                                                             ADD00690
      RETURN                                                                  ADD00700
      END                                                                     ADD00710
```

```
FILE: FLDCLS   FORTRAN  A

      SUBROUTINE FLDCLS(FIELDS,STAMNT,*,*,*,*,IPT,VERTEX)                                FLD00010
C*                                                                                       FLD00020
C*    FLDCLS GROUPS THE FIELDS CARDS ON A CLASS BASES FOR PROCESSING                     FLD00030
C*                                                                                       FLD00040
      IMPLICIT INTEGER (A-Z)                                                             FLD00050
C     INCLUDE COMMON.LIST                                                                FLD00060
C     INCLUDE CMSP11.LIST                                                                FLD00070
      COMMON/INFO2M/NOCLS2,NOSUB2,NOFFT2,VARS2,TOTVT2,NOFLD2,                            FLD00080
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLDSV2,VERTX2,                     FLD00090
     *              FFTVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),                         FLD00100
     *              KEPPTS(40),NOGRP,GRPNAM(60),GRPDEX(61),                              FLD00110
     *              GRPCHR(41),GROUPS(124)                                               FLD00120
      COMMON /NDIM/NCLRCM,CLRVEC(30),MAXVEC,MAPKEY,                                      FLD00130
     *     CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(6),SIZE,TOTMNS                         FLD00140
     *   ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,HUFLEN,ID3,COLOR3,NODUMP                     FLD00150
     *   ,IDATA1,TOTVEC                                                                FLD00160
CSEND                                                                                    FLD00170
      DIMENSION FIELDS(4,1),VERTEX(1)                                                    FLD00180
      LOGICAL SWITCH                                                                     FLD00190
      DATA SWITCH/.TRUE./                                                                FLD00200
      IF (NOFLD2 .EQ. 0) IPT = 1                                                         FLD00210
      IF (NOFLD2 .EQ. 0) GO TO 75                                                        FLD00220
      IPT = IPT + FIELDS(4,NOFLD2)*2                                                     FLD00230
   75 CONTINUE                                                                           FLD00240
      GO TO (40,100),STAMNT                                                              FLD00250
   80 I = LAREAD(FIELDS(1,NOFLD2+1),VERTEX(IPT),FLDINF(1),FIELDS(4,NOFLD2FLD00260
     *+1))                                                                               FLD00270
                                                                                         FLD00280
C     WAS CLASS,SUBCLASS,FIELD, OR SEND* ENCOUNTERED                                     FLD00290
C                                                                                        FLD00300
      IF ( I .EQ. -1) GO TO 90                                                           FLD00310
      IF ( I .EQ. -2) GO TO 120                                                          FLD00320
      IF ( I .EQ.  1) GO TO 130                                                          FLD00330
      IF ( I .EQ.  0) GO TO 140                                                          FLD00340
                                                                                         FLD00350
C     CLASS CARD                                                                         FLD00360
C                                                                                        FLD00370
   90 IF (SWITCH) GO TO 100                                                              FLD00380
      STAMNT = 2                                                                         FLD00390
      RETURN 2                                                                           FLD00400
  100 READ(30,110)CLSVC2(1)                                                              FLD00410
      REWIND 30                                                                          FLD00420
      IPT = 1                                                                            FLD00430
      NOCLS2 = 1                                                                         FLD00440
      NOFLD2 = 0                                                                         FLD00450
      NOSUB2 = 0                                                                         FLD00460
      SWITCH = .FALSE.                                                                   FLD00470
      GO TO 80                                                                           FLD00480
C                                                                                        FLD00490
C     SUBCLASS CARD                                                                      FLD00500
C                                                                                        FLD00510
  110 FORMAT(10A6)                                                                       FLD00520
  120 NOSUB2 = NOSUB2 + 1                                                                FLD00530
      READ(30,110)SUBVC2(NOSUB2)                                                         FLD00540
      REWIND 30                                                                          FLD00550
      GO TO 80                                                                           FLD00560
C                                                                                        FLD00570
C     FIELD CARD                                                                         FLD00580
C                                                                                        FLD00590
  130 NOFLD2 = NOFLD2 + 1                                                                FLD00600
      STAMNT = 1                                                                         FLD00610
      RETURN 1                                                                           FLD00620
C                                                                                        FLD00630
C     *END*                                                                              FLD00640
C                                                                                        FLD00650
  140 RETURN 3                                                                           FLD00660
C                                                                                        FLD00670
      END                                                                                FLD00680
```

FILE: FLDFLD FORTRAN A

```
      SUBROUTINE FLDFLD(FIELDS,STAMNT,*,*,IPT,VERTEX)                   FLD00010
C                                                                       FLD00020
C*                                                                      FLD00030
C*    FLDFLD CONTROL THE PROCESSING OF FIELDS CARDS ON A PER FIELD BASESFLD00040
C*                                                                      FLD00050
C     IMPLICIT INTEGER (A-Z)                                            FLD00060
      INCLUDE COMMON.LIST                                               FLD00070
C     INCLUDE CARD1.LIST                                                FLD00080
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VAWS22,TOTVT2,NOFLD2,          FLD00090
     *              AVAW2,COVAW2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    FLD00100
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        FLD00110
     *              REMRTS(40),NOGRP,GRPNAM(60),GRPDEX(61),             FLD00120
     *              GRPCHR(61),GROUPS(124)                              FLD00130
      COMMON /NDIM/NCLNCH,CLSVEC(30),MARVEC,MAPKEY,                     FLD00140
     *     CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(6),SIZE,TOTWNS        FLD00150
     *   ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,BUFLEN,ID3,COLOR3,NODUMP    FLD00160
     *   ,IDATA1,TOTVEC                                                 FLD00170
CSEND                                                                   FLD00180
      DIMENSION FIELDS(4,1),VERTEX(1)                                   FLD00190
      NOFLD2 = 1                                                        FLD00200
      IPT = 1                                                           FLD00210
C                                                                       FLD00220
C                                                                       FLD00230
   80 I = LAREAD(FIELDS(1,NOFLD2),VERTEX(IPT),FLDINF(1),                FLD00240
     *       FIELDS(4,NOFLD2))                                          FLD00250
C                                                                       FLD00260
C     WAS CLASS,SUBCLASS,FIELD, OR SEND* ENCOUNTERED                    FLD00270
C                                                                       FLD00280
      IF ( I .EQ. -1) GO TO 90                                          FLD00290
      IF ( I .EQ. -2) GO TO 110                                         FLD00300
      IF ( I .EQ.  1) GO TO 120                                         FLD00310
      IF ( I .EQ.  0) GO TO 130                                         FLD00320
C                                                                       FLD00330
C     CLASS CARD                                                        FLD00340
C                                                                       FLD00350
   90 READ(30,100)CLSVC2(1)                                             FLD00360
      REWIND 30                                                         FLD00370
      NOCLS2 = 1                                                        FLD00380
  100 FORMAT(10X,A4)                                                    FLD00390
      GO TO 80                                                          FLD00400
C                                                                       FLD00410
C     SUBCLASS CARD                                                     FLD00420
C                                                                       FLD00430
  110 READ(30,100)SUBVC2(1)                                             FLD00440
      REWIND 30                                                         FLD00450
      NOSUB2=1                                                          FLD00460
      GO TO 80                                                          FLD00470
C                                                                       FLD00480
C     FIELD CARD                                                        FLD00490
C                                                                       FLD00500
  120 RETURN 1                                                          FLD00510
C                                                                       FLD00520
C     SEND*                                                             FLD00530
C                                                                       FLD00540
  130 RETURN 2                                                          FLD00550
C                                                                       FLD00560
      END                                                               FLD00570
```

```
FILE: FLDMEN   FORTRAN  A

      SUBROUTINE FLDMEN(IDATA,J,NSAMP,NOFEAT,MEANS,AGCHAN,N)                FLD00010
                                                                            FLD00020
C*    FLDMEN COMPUTES THE FIELD MEANS                                       FLD00030
C*                                                                          FLD00040
C*                                                                          FLD00050
      IMPLICIT INTEGER (A-Z)                                                FLD00060
      REAL MEANS,RND                                                        FLD00070
C     INCLUDE COMMON.LIST                                                   FLD00080
C     INCLUDE COMMON.LIST                                                   FLD00090
      COMMON/INFDEM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,              FLD00100
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,        FLD00110
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),            FLD00120
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                 FLD00130
     *              GRPCHK(61),GROUPS(124)                                  FLD00140
      COMMON /NDIM/NCLRCH,CLRVEC(30),MAXVEC,MAPKEY,                         FLD00150
     *     CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(6),SIZE,TOTMNS            FLD00160
     *   ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,RUFLEN,ID3,COLOR3,NODUMP        FLD00170
     *   ,IDATA1,TOTVEC                                                     FLD00180
C$END                                                                       FLD00190
      DIMENSION IDATA(NSAMP,NOFEAT),MEANS(NCLRCH,1)                         FLD00200
C                                                                           FLD00210
C     MEANS = ( (N-1)*OLD MEAN )/N +  DATA PT./N                            FLD00220
C                                                                           FLD00230
C                                                                           FLD00240
      RND = (FLOAT(N)-1.0)/FLOAT(N)                                         FLD00250
      ICHAN = 0                                                             FLD00260
C                                                                           FLD00270
      DO 100 K = AGCHAN,NOFEAT                                              FLD00280
      ICHAN = ICHAN + 1                                                     FLD00290
  100 MEANS(ICHAN,NOFLD2) = RND*MEANS(ICHAN,NOFLD2) +                       FLD00300
     *                      FLOAT(IDATA(J,K))/FLOAT(N)                     FLD00310
C                                                                           FLD00320
      RETURN                                                                FLD00330
      END
```

FILE: FLDSUB FORTRAN A

```
      SUBROUTINE FLDSUB(FIELDS,STAMNT,*,*,*,IPT,VERTEX)                 FLD00010
C                                                                       FLD00020
C*                                                                      FLD00030
C*    FLDSUB CONTROL THE PROCESSING OF THE FIELD CARDS ON THE SUBCLASS  FLD00040
C*    LEVEL                                                             FLD00050
C*                                                                      FLD00060
      IMPLICIT INTEGER (A-Z)                                            FLD00070
C     INCLUDE COMMK1.LIST                                               FLD00080
C     INCLUDE COMMK11.LIST                                              FLD00090
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFLD2,          COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        COM00030
     *              KEPPTS(40),NOGRP,GRPNAM(60),GRPDEX(61),             COM00040
     *              GRPCHK(61),GROUPS(124)                              COM00050
      COMMON /NDIM/NCLRCH,CLRVEC(30),MAXVEC,MAPKEY,
     *     CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(6),SIZE,TOTMNS
     *   ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,BUFLEN,ID3,COLOR3,NODUMP
     *   ,IDATA1,TOTVEC
C$END                                                                   FLD00110
      LOGICAL SWITCH                                                    FLD00120
      DATA SWITCH/.TRUE./                                               FLD00130
      DIMENSION FIELDS(4,1),VERTEX(1)                                   FLD00140
C                                                                       FLD00150
      IF (NOFLD2 .EQ. 0) IPT = 1                                        FLD00160
      IF (NOFLD2 .EQ. 0) GO TO 75                                       FLD00170
C                                                                       FLD00180
      IPT = IPT + FIELDS(4,NOFLD2)*2                                    FLD00190
   75 CONTINUE                                                          FLD00200
C                                                                       FLD00210
      GO TO (80,110,90),STAMNT                                          FLD00220
C                                                                       FLD00230
   80 I = LAREAD(FIELDS(1,NOFLD2+1),VERTEX(IPT),FLDINF(1),FIELDS(4,NOFLDFLD00240
     *2+1))                                                             FLD00250
C                                                                       FLD00260
CC    WAS CLASS,SUBCLASS,FIELD, OR $END* CARD ENCOUNTERED               FLD00270
C                                                                       FLD00280
      IF (I .EQ. -1) GO TO 85                                           FLD00290
      IF (I .EQ. -2) GO TO 100                                          FLD00300
      IF (I .EQ.  1) GO TO 130                                          FLD00310
      IF (I .EQ.  0) GO TO 140                                          FLD00320
C                                                                       FLD00330
CC    CLASS CARD                                                        FLD00340
C                                                                       FLD00350
   85 IF(SWITCH) GO TO 90                                               FLD00360
      STAMNT = 3                                                        FLD00370
      SWITCH = .TRUE.                                                   FLD00380
      RETURN 2                                                          FLD00390
   90 READ(30,120)CLSVC2(1)                                             FLD00400
      REWIND 30                                                         FLD00410
      NOCLS2 = 1                                                        FLD00420
      GO TO 80                                                          FLD00430
C                                                                       FLD00440
CC    SUBCLASS CARD                                                     FLD00450
C                                                                       FLD00460
  100 IF (SWITCH) GO TO 110                                             FLD00470
      STAMNT = 2                                                        FLD00480
      RETURN 2                                                          FLD00490
  110 READ(30,120)SUBVC2(1)                                             FLD00500
      REWIND 30                                                         FLD00510
  120 FORMAT(10X,A4)                                                    FLD00520
      NOFLD2 = 0                                                        FLD00530
      NOSUB2 = 1                                                        FLD00540
      IPT = 1                                                           FLD00550
      SWITCH = .FALSE.                                                  FLD00560
      GO TO 80                                                          FLD00570
C                                                                       FLD00580
CC    FIELD CARD                                                        FLD00590
C                                                                       FLD00600
  130 NOFLD2 = NOFLD2 + 1                                               FLD00610
      STAMNT = 1                                                        FLD00620
      RETURN 1                                                          FLD00630
C                                                                       FLD00640
CC    $END*                                                             FLD00650
C                                                                       FLD00660
  140 RETURN 3                                                          FLD00670
C                                                                       FLD00680
      END                                                               FLD00690
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: NDHST1 FORTRAN A

```
      SUBROUTINE NDHST1(HIST,FIELDS,      MEANS,VERTEX,LIMIT,ARRAY,TOP)  NDH00010
C*                                                                       NDH00020
C*    NDHST1 SETS UP THE LOGIC FOR HISTOGRAMMING THE DATA AND WRITING    NDH00030
C*    THE NDIM FILE                                                      NDH00040
C*                                                                       NDH00050
      IMPLICIT INTEGER (A-Z)                                             NDH00060
      REAL MEANS                                                         NDH00070
      DIMENSION HIST(1),ARRAY(1),MEANS(1)                                NDH00080
      DIMENSION FIELDS(4,1),FETVEC(30),FL(12),VERTEX(1)                  NDH00090
C     INCLUDE COMMK1.LIST                                                NDH00100
CC    INCLUDE COMMK4.LIST                                                NDH00110
C     INCLUDE COMFIL.LIST                                                NDH00120
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           NDH00130
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     NDH00140
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         NDH00150
     *              KEPPTS(40),NOGRP,GRPNAM(60),GRPDEX(61),              NDH00160
     *              GRPCHR(61),GROUPS(124)                               NDH00170
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          NDH00180
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,    NDH00190
     *    DPUMAD,DPMODS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 NDH00200
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                     NDH00210
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,               NDH00220
     *  CROUNT,PRTUNT,RANDIO                                             NDH00230
      COMMON /NDIM/NCLRCH,CLRVEC(30),MAXVEC,MAPKEY,                      NDH00240
     *    CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(6),SIZE,TOTMNS          NDH00250
     *   ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,BUFLEN,ID3,COLOR3,NODUMP     NDH00260
     *   ,IDATA1,TOTVEC                                                  NDH00270
C$END                                                                    NDH00280
C     INITIALIZE PARAMETERS                                              NDH00290
      IENTER = 0                                                         NDH00300
      STAMNT = 1                                                         NDH00310
   90 LENTH = NOVEC*SIZE                                                 NDH00320
      DO 80  JK=1,LENTH                                                  NDH00330
   80 HIST(JK) = 0                                                       NDH00340
      BEGIN = DPUMAD                                                     NDH00350
      OVFLO = 0                                                          NDH00360
      VECCNT = 0                                                         NDH00370
      NODUMP = 0                                                         NDH00380
      NOVEC = 0                                                          NDH00390
      TOTVEC = 0                                                         NDH00400
      NOFLD2 = 0                                                         NDH00410
      TOTVT2 = 0                                                         NDH00420
      IF (MEANS* .NE. 1) GO TO 85                                        NDH00430
      DO 82  I=1,TOTMNS                                                  NDH00440
   82 MEANS(I) = 0.0                                                     NDH00450
      TOTMNS = 0                                                         NDH00460
   85 IF (CLASS .EQ. 1 ) CALL FLDCLS(FIELDS,STAMNT,&100,&510,&520,IPT,   NDH00470
     *              VERTEX)                                             NDH00480
      IF (SUBCLS .EQ. 1) CALL FLDSUB(FIELDS,STAMNT,&100,&510,&520,IPT,   NDH00490
     *              VERTEX)                                             NDH00500
      IF (FIELD .EQ. 1) CALL FLDFLD(FIELDS,STAMNT,&100,&530,IPT,VERTEX)  NDH00510
C                                                                        NDH00520
  100 LINSTR = FLDINF(1)                                                 NDH00530
      LINEND = FLDINF(2)                                                 NDH00540
      LININC = FLDINF(3)                                                 NDH00550
      SAMSTR = FLDINF(4)                                                 NDH00560
      SAMEND = FLDINF(5)                                                 NDH00570
      SAMINC = FLDINF(6)                                                 NDH00580
      FIELDS(2,NOFLD2) = NOCLS2                                          NDH00590
      FIELDS(3,NOFLD2) = NOSUB2                                          NDH00600
      TOTVT2 = FIELDS(4,NOFLD2) + TOTVT2                                 NDH00610
      IF (MEANS* .EQ. 1) TOTMNS = NCLRCH * TOTMNS                        NDH00620
C                                                                        NDH00630
      ILINE = (LINEND-LINSTR)/LININC + 1                                 NDH00640
      NSAMP = (SAMEND-SAMSTR)/SAMINC + 1                                 NDH00650
C                                                                        NDH00660
CC    COMBINE PLOTTING AND COLOR CHANNELS                                NDH00670
C                                                                        NDH00680
      DO 110 I=1,NOFET2                                                  NDH00690
  110 FETVEC(I) = FETVC2(I)                                              NDH00700
      IF (NCLRCH .EQ. 0) GO TO 130                                       NDH00710
C                                                                        NDH00720
CC    ARE COLOR CHANNELS AND PLOTTING CHANNELS THE SAME CHANNELS         NDH00730
C                                                                        NDH00740
      IF (NOFET2 .NE. NCLRCH) GO TO 102                                  NDH00750
      DO 101 I=1,NOFET2                                                  NDH00760
      IF (FETVEC(I) .NE. CLRVEC(I))    TO 102                            NDH00770
  101 CONTINUE                                                           NDH00780
      NOFEAT = NOFET2                                                    NDH00790
```

```
FILE: NDHST1   FORTRAN  A

      BGCHAN = 1                                                        NDH00800
      GO TO 135                                                         NDH00810
  102 CONTINUE                                                          NDH00820
      DO 120  I=1,NCLRCH                                                NDH00830
  120 FETVEC(NOFET2+I) = CLRVEC(I)                                      NDH00840
  130 NOFEAT = NOFET2 + NCLRCH                                          NDH00850
C                                                                       NDH00860
      BGCHAN = NOFET2 + 1                                               NDH00870
  135 CONTINUE                                                          NDH00880
C     COMPUTE ADDRESSES .                                               NDH00890
C                                                                       NDH00900
      CALL ADDRES(TOP,NSAMP,NOFEAT,BEGIN,BEGIN1)                        NDH00910
C                                                                       NDH00920
CCC   IF REMAINING STORAGE IN HIST ARRAY IS LESS THAN 2000, EMPTY       NDH00930
CCC   HIST ARRAY (DATA VECTORS) ONTO DRUM, READ IN MAP TAPE,STORE ONTO  NDH00940
CCC   DRUM, THEN READ DATA VECTORS BACK INTO HIST                       NDH00950
C                                                                       NDH00960
      IF (MAPKEY .NE. 1) GO TO 105                                      NDH00970
      STORGE = LIMIT - NOVEC*SIZE                                       NDH00980
      IF(STORGE .LT. 2000) GO TO 103                                    NDH00990
      CALL STODAT(ILINE,NSAMP,HIST(NOVEC+1),STORGE,BEGIN1)              NDH01000
      GO TO 105                                                         NDH01010
  103 VECTR1 = COLOR2                                                   NDH01020
      WRDS = NOVEC*SIZE                                                 NDH01030
      CALL RWRITE(VECTR1,HIST,WRDS ,ISTAT)                              NDH01040
  104 IF (ISTAT .EQ. 1) GO TO 104                                       NDH01050
      BEGIN1 = VECTR1 + WRDS                                            NDH01060
      CALL STODAT(ILINE,NSAMP,HIST,LIMIT,BEGIN1)                        NDH01070
      CALL RREAD(VECTR1,HIST,WRDS,ISTAT2)                               NDH01080
C                                                                       NDH01090
CCC   INITIIZE IMAGE DATA TAPE                                          NDH01100
C                                                                       NDH01110
                                                                        NDH01120
  105 CALL TAPHDR(DATAMF,DATFIL)                                        NDH01130
  106 IF (ISTAT2 .EQ. 1) GO TO 106                                      NDH01140
C                                                                       NDH01150
CCC   POSITION IMAGE TAPE FOR THIS FIELD                                NDH01160
C                                                                       NDH01170
      CALL FLDINT(FLDINF(1),FETVEC,NOFEAT)                              NDH01180
      NLINE = 0                                                         NDH01190
      NPTS = 0                                                          NDH01200
C                                                                       NDH01210
CCC   READ A SCAN  LINE OF DATA AND PROCESS IT                          NDH01220
C                                                                       NDH01230
      DO 500 LINE=LINSTR,LINEND,LININC                                  NDH01240
      NLINE = NLINE + 1                                                 NDH01250
C                                                                       NDH01260
      CALL LINERD(ARRAY(IDATA1),ENDTAP)                                 NDH01270
      IF (ENDTAP .EQ. -1) GO TO 600                                     NDH01280
C                                                                       NDH01290
CCC   READ IN A SCAN LINE FROM CLASSIFICATION/CLUSTER MAP TAPE          NDH01300
C                                                                       NDH01310
      IF (MAPKEY .EQ. 1) CALL RESTO(NLINE,NSAMP,BEGIN1)                 NDH01320
C                                                                       NDH01330
CCC   FIND INTERSECTIONS FOR N-R FIELDS                                 NDH01340
C                                                                       NDH01350
      CALL FDLINT(VERTEX(IPT),FIELDS(4,NOFLD2),FL,LINE, SAMP,NI)        NDH01360
C                                                                       NDH01370
      DO 400 J=1,NI,2                                                   NDH01380
      IB = (FL(J)-SAMSTR)/SAMINC + 1                                    NDH01390
      IE = (FL(J+1)- SAMSTR)/SAMINC + 1                                 NDH01400
      IF (MOD(SAMSTR,SAMINC) .NE. MOD(FL(J),SAMINC)) IB = IB + 1        NDH01410
      IF (IB .GT. IE) GO TO 400                                         NDH01420
C                                                                       NDH01430
      DO 350 K=IB,IE                                                    NDH01440
C                                                                       NDH01450
      TOTVEC = TOTVEC + 1                                               NDH01460
      NPTS = NPTS + 1                                                   NDH01470
C     HISTOGRAM VECTOR                                                  NDH01480
CC                                                                      NDH01490
      CALL NDHST2(K,ARRAY(IDATA1),HIST,NOFET2,VECSWT,NSAMP,             NDH01500
     *    ARRAY(1) , VECCNT,OVRFLO,NOFEAT,BGCHAN)                       NDH01510
C                                                                       NDH01520
CCC   IF A NEW VECTOR WAS FOUND,VECSWT = 1                              NDH01530
C                                                                       NDH01540
  220 IF (MEANSW .EQ. 0) GO TO 330                                      NDH01550
C                                                                       NDH01560
CCC   COMPUTE MEANS FOR TRAINING/TEST FIELDS                            NDH01570
C                                                                       NDH01580
```

```
FILE: NDHST1   FORTRAN  A

      CALL FLOMEN(ARRAY(IDATA1),K,NSAMP,NOFEAT,MEANS,RGCHAN,NPTS)          NDH01590
                                                                           NDH01600
C                                                                          NDH01610
  330 IF (VECSWT .NE. 1) GO TO 350                                         NDH01620
C                                                                          NDH01630
      ARRAY(ID1 + VECCNT) = NOFLD2                                         NDH01640
      IF (MAPKEY .EQ. 0) GO TO 340                                         NDH01650
C                                                                          NDH01660
CC    RETRIEVE CLUSTERED/CLASSIFIED  DATA FROM DRUM                        NDH01670
C                                                                          NDH01680
      CALL RESTOR(K,ARRAY(ID1+VECCNT))                                     NDH01690
C                                                                          NDH01700
  340 VECCNT = VECCNT + 1                                                  NDH01710
      IF (VECCNT .LE. (BUFLEN-1)) GO TO 350                                NDH01720
C                                                                          NDH01730
CC    DUMP ONTO DRUM                                                       NDH01740
C                                                                          NDH01750
      NODUMP = NODUMP + 1                                                  NDH01760
      CALL RWRITE(ID2,ARRAY(ID1),BUFLEN,ISTAT1)                            NDH01770
      ID2 = ID2 + BUFLEN                                                   NDH01780
      VECCNT = 0                                                           NDH01790
      IF (MEANS4 .EQ. 1) GO TO 350                                         NDH01800
      IF (NCLRCH .EQ. 0) GO TO 350                                         NDH01810
      CALL RWRITE(COLOR2,ARRAY(COLOR1),BUFLEN,ISTAT2)                      NDH01820
      COLOR2 = COLOR2 + BUFLEN                                             NDH01830
C                                                                          NDH01840
  350 CONTINUE                                                             NDH01850
C                                                                          NDH01860
  400 CONTINUE                                                             NDH01870
  500 CONTINUE                                                             NDH01880
CC    IF OVRFLO IS GREATER THAN 0, HISTOGRAMMED DATA VECTOR TABLE IS       NDH01890
CC    FULL                                                                 NDH01900
C                                                                          NDH01910
      IF (OVRFLO .EQ. 0) GO TO 505                                         NDH01920
      WRITE(6,221)                                                         NDH01930
  221 FORMAT(////)                                                         NDH01940
      WRITE(6,225)OVRFLO                                                   NDH01950
  225 FORMAT(1X,I6,' VECTORS WERE NOT HISTOGRAMMED, BUT USED IN COMPUTINGNDH01960
     *G FIELD MEANS, IF APPLICABLE')                                       NDH01970
  505 CONTINUE                                                             NDH01980
      IF (FIELD .NE. 1) GO TO 85                                           NDH01990
      CALL WRTFIL(HIST,MEANS,ARRAY(ID1),ARRAY(COLOR1),FIELDS,VERTEX,       NDH02000
     *   IENTER)                                                           NDH02010
      GO TO 90                                                             NDH02020
C                                                                          NDH02030
CC    WRITE HISTOGRAM FILE                                                 NDH02040
C                                                                          NDH02050
C                                                                          NDH02060
  510 CALL WRTFIL(HIST,MEANS,ARRAY(ID1),ARRAY(COLOR1),FIELDS,VERTEX,       NDH02070
     *   IENTER)                                                           NDH02080
      GO TO 90                                                             NDH02090
C                                                                          NDH02100
CC    *END CARD FOUND                                                      NDH02110
C                                                                          NDH02120
  520 CALL WRTFIL(HIST,MEANS,ARRAY(ID1),ARRAY(COLOR1),FIELDS,VERTEX,       NDH02130
     *   IENTER)                                                           NDH02140
  530 ENDFILE NHSTUN                                                       NDH02150
      REWIND NHSTUN                                                        NDH02160
      RETURN                                                               NDH02170
  600 WRITE(6,610)                                                         NDH02180
  610 FORMAT(' ERROR IN FIELD CARD, ABORTING')                             NDH02190
      CALL CMERR                                                           NDH02200
      END                                                                  NDH02210
C
```

```
FILE: NDHST2   FORTRAN  A

      SUBROUTINE NDHST2(J,IDATA,HIST,NOFET2,VECSWT,NSAMP,ARRAY,VECCNT,   NDH00010
     *  OVRFLO,NOFEAT,AGCHAN)                                            NDH00020
C                                                                       NDH00030
C*    NDHST2 PERFORMS THE 1 TO 16 CHANNEL HISTOGRAM                     NDH00040
C*    THE HISTOGRAM IS COMPUTED FOR EITHER ONE OR TWO SETS OF CHANNELS   NDH00050
C*                                                                      NDH00060
      IMPLICIT INTEGER (A-Z)                                            NDH00070
      DIMENSION HIST(SIZE,MAXVEC),IDATA(1),COMWRD(4)                    NDH00080
      DIMENSION ARRAY(1)                                                NDH00090
C     INCLUDE COMMON.LIST                                               NDH00100
C                                                                       NDH00110
C     INCLUDE COMMON.LIST                                               NDH00120
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         NDH00130
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,   NDH00140
     *    DRUMAD,DRMWDS,FAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                NDH00150
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    NDH00160
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              NDH00170
     *  CRDUNT,PRTUNT,RANDIO                                            NDH00180
      COMMON /NDIM/NCLRCH,CLRVEC(30),MAXVEC,MAPKEY,                     NDH00190
     *    CLASS,SUBCLS,FIELD,MEANS,NOVEC,FLDINF(6),SIZE,TOTMNS          NDH00200
     *   ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,BUFLEN,ID3,COLOR3,NODUMP     NDH00210
     *   ,IDATA1,TOTVEC                                                 NDH00220
CSEND                                                                   NDH00230
C                                                                       NDH00240
      VECSWT = 0                                                        NDH00250
C                                                                       NDH00260
      LOGICAL*1 LDUM(4),LLDUM(4)                                        NDH00270
      EQUIVALENCE(IDUM,LDUM(1)),(IIDUM,LLDUM(1))                        NDH00280
      DO 50 I=1,4                                                       NDH00290
   50 COMWRD(I)=0                                                       NDH00300
      II=0                                                              NDH00310
      DO 60 I=1,NOFET2                                                  NDH00320
      II=II+1                                                           NDH00330
      III=1+(II-1)/4                                                    NDH00340
      IIII=J+(I-1)*NSAMP                                                NDH00350
      IBYTE=II-((III-1)/4)*4                                            NDH00360
      IF(IBYTE.EQ.1) IIDUM=0                                            NDH00370
      IDUM=IDATA(IIII)                                                  NDH00380
      LLDUM(IBYTE)=LDUM(4)                                              NDH00390
      COMWRD(III)=IIDUM                                                 NDH00400
   60 CONTINUE                                                          NDH00410
C  STORE LAST 8 BITS OF IDATA(IIII) INTO COMWRD(III) IN                 NDH00420
C  BYTE POSITION IBYTE, LEFT TO RIGHT                                   NDH00430
      IF (NCLRCH .EQ. 0) GO TO 100                                      NDH00440
      IF (MEANS .EQ. 1) GO TO 100                                       NDH00450
      CALL RICOLR(IDATA,J,NOFEAT,COLWRD,NSAMP,NOFET2,AGCHAN)            NDH00460
  100 CONTINUE                                                          NDH00470
      IF (NOVEC .EQ. 0) GO TO 135                                       NDH00480
C                                                                       NDH00490
C     COMPARING VECTORS ALREADY FOUND WITH IN COMING VECTOR             NDH00500
C                                                                       NDH00510
  110 DO 130  K=1,NOVEC                                                 NDH00520
C                                                                       NDH00530
      DO 120  L=1,SIZE                                                  NDH00540
      IF (COMWRD(L) .NE. HIST(L,K)) GO TO 130                           NDH00550
  120 CONTINUE                                                          NDH00560
C                                                                       NDH00570
C     FOUND A MATCHING VECTOR                                           NDH00580
C                                                                       NDH00590
C     DO ONE OF THE FOLLOWING :                                         NDH00600
C     1) HISTOGRAM THE VECTOR ONLY IF COLOR CODES FOR BOTH VECTORS ARE  NDH00610
C       THE SAME.                                                       NDH00620
C     2. DO NOT CHECK COLOR CODES IF MAP TAPE IS BEING INPUT OR MEANS   NDH00630
C       FOR FIELDS ARE BEING COMPUTED                                   NDH00640
C                                                                       NDH00650
      IF (NCLRCH .EQ. 0) GO TO 126                                      NDH00660
      IF (MEANS .EQ. 1) GO TO 126                                       NDH00670
C                                                                       NDH00680
C     IS COLOR CODE IN CORE OR ON DRUM                                  NDH00690
C                                                                       NDH00700
      IF (NODUMP .EQ. 0) GO TO 122                                      NDH00710
      IF (K .GT. (NODUMP*BUFLEN)) GO TO 122                             NDH00720
      ADDRES = COLOR3 + (NODUMP-1)*BUFLEN + K - 1                       NDH00730
      CALL RREAD(ADDRES,CODES,1,ISTAT2)                                 NDH00740
  121 IF (ISTAT2 .EQ. 1) GO TO 121                                      NDH00750
      GO TO 124                                                         NDH00760
  122 KK = K - (NODUMP*BUFLEN)                                          NDH00770
      CODES = ARRAY(COLOR1 + KK - 1)                                    NDH00780
  124 IF (CODES .NE. COLWRD) GO TO 150                                  NDH00790
```

```
FILE: NDHST2   FORTRAN   A

  124 CONTINUE                                                                  NDH00800
                                                                                NDH00810
C                                                                               NDH00820
      CNTR1 = CNTR2 + K - 1                                                     NDH00830
      CALL RREAD(CNTR1,COUNTR,1,ISTAT)                                          NDH00840
  125 IF(ISTAT .EQ. 1) GO TO 125                                                NDH00850
      COUNTR = COUNTR + 1                                                       NDH00860
      CALL RWRITE(CNTR1,COUNTR,1,ISTAT1)                                        NDH00870
  127 IF( ISTAT1 .EQ. 1) GO TO 127                                              NDH00880
      GO TO 150                                                                 NDH00890
  130 CONTINUE                                                                  NDH00900
                                                                                NDH00910
C                                                                               NDH00920
C     INSERT NEW VECTOR                                                         NDH00930
C                                                                               NDH00940
  135 IF (NOVEC .LT. MAXVEC) GO TO 137                                          NDH00950
      OVRFLO = OVRFLO + 1                                                       NDH00960
      IF (OVRFLO .GT. 1) GO TO 136                                              NDH00970
      WRITE(6,133)                                                              NDH00980
  133 FORMAT(//////)                                                            NDH00990
      WRITE(6,123)MAXVEC                                                        NDH01000
  123 FORMAT(/' CORE LIMITS EXCEEDED. MAXIMUM NO. OF VECTORS'/' ACCEPTED       NDH01010
     * IS ',I6)                                                                 NDH01020
C                                                                               NDH01030
C     VECTOR TABLE IS FULL--CONTINUE TO HISTOGRAM DATA VECTORS THAT             NDH01040
C       ALREADY EXIST                                                           NDH01050
C                                                                               NDH01060
  136 RETURN                                                                    NDH01070
  137 NOVEC = NOVEC + 1                                                         NDH01080
      DO 140  L=1,SIZE                                                          NDH01090
  140 HIST(L,NOVEC) = COMWRD(L)                                                 NDH01100
      CNTR1 = CNTR2 + NOVEC - 1                                                 NDH01110
      COUNTR = 1                                                                NDH01120
      CALL RWRITE(CNTR1,COUNTR, 1,ISTAT3)                                       NDH01130
  145 IF (ISTAT3 .EQ. 1) GO TO 145                                              NDH01140
      VECSWT = 1                                                                NDH01150
      IF (MFANSW .NE. 0 .OR. MAPKEY .NE. 0) GO TO 150                           NDH01160
      ARRAY(COLOR1 + VECCNT) = COLWRD                                           NDH01170
  150 CONTINUE                                                                  NDH01180
  200 CONTINUE                                                                  NDH01190
C                                                                               NDH01200
      RETURN
      END
```

FILE: PICOLR FORTRAN A

```
      SUBROUTINE PICOLR(IDATA,K,NOFEAT,COLWRD,NSAMP,NOFET2,RGCHAN)          PIC00010
      PICOLR EXTRACT THE COLOR CHANNELS FROM IDATA AND                      PIC00030
      PACK THE 8 BIT PIXELS INTO A COMPUTER WORD                            PIC00040
C                                                                           PIC00050
      IMPLICIT INTEGER (A-Z)                                                PIC00060
C     INCLUDE CAMELLI.LIST                                                  PIC00070
      COMMON /NDIM/NCLRCH,CLRVEC(30),MAXVEC,MARKEY,                         PIC00080
     *    CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(6),SIZE,TOTMNS             PIC00090
     *   ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,NUFLEN,ID3,COLOR3,NODIMP        PIC00100
     *   ,IDATA1,TOTVEC                                                     PIC00110
CSEND                                                                       PIC00120
      DIMENSION IDATA(1)                                                    PIC00130
      LOGICAL*1 LDUM(4), LLDUM(4)                                           PIC00140
      EQUIVALENCE (IDUM,LDUM(1)),(IIDUM,LLDUM(1))                          PIC00150
      IIDUM=0                                                               PIC00160
      DO 100 I=1,NCLRCH                                                     PIC00170
      II=K*(RGCHAN+I-2)*NSAMP                                               PIC00180
      IDUM=IDATA(II)                                                        PIC00190
  100 LLDUM(I)=LDUM(4)                                                      PIC00200
      COLWRD=IIDUM                                                          PIC00210
      RETURN                                                                PIC00220
C                                                                           PIC00230
      END
```

```
FILE: RESTO    FORTRAN  A

      SUBROUTINE RESTO (ILINE,NSAMP,REGIN1)                                  RES00010
C                                                                            RES00020
C*                                                                           RES00030
C*    RESTO  RETRIEVES THE CLUSTER MAP FROM DRUM A LINE AT A TIME            RES00040
C*                                                                           RES00050
C*    RESTOR PICKS OUT THE DESIRED ID FROM WITHIN THE LINE                   RES00060
C*                                                                           RES00070
C*                                                                           RES00080
      IMPLICIT INTEGER (A-Z)                                                 RES00090
C     INCLUDE COMMON.LIST                                                    COM00010
      COMMON/GLOBAL/HEAD(63),MAPTAB,DATAPE,SAVTAB,BMFILE,BMKEY,              COM00020
     *              HISFIL,HISKEY,THFORM,ERIPTP,EPPKEY,MAPUNT,NOFILE,        COM00030
     *     DRUMAD,DWORDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    COM00040
     *   ,NHSTUN,NHSTFI,SCTHUN,MAPFIL                                        COM00050
     *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                  COM00060
     *   CHOUNT,BWTUNT,HANDIO                                                RES00110
CSEND                                                                        RES00120
      COMMON /IDWORD/ IDWORD(1000)                                           RES00130
C                                                                            RES00140
      ADDRES = REGIN1 + (ILINE-1)*NSAMP                                      RES00150
C                                                                            RES00160
      CALL RREAD(ADDRES,IDWORD,NSAMP,ISTAT)                                  RES00170
  150 IF (ISTAT .EQ. 1) GO TO 150                                            RES00180
      IF (ISTAT .GE. 0) RETURN                                               RES00190
      WRITE(6,100)                                                           RES00200
  100 FORMAT(' ERROR READING DRUM')                                          RES00210
      CALL CMERR                                                             RES00220
C                                                                            RES00230
C                                                                            RES00240
      ENTRY RESTOR(K,NUMB)                                                   RES00250
C                                                                            RES00260
      NUMB = IDWORD(K)                                                       RES00270
C                                                                            RES00280
      RETURN                                                                 RES00290
C                                                                            RES00300
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FILE: SET10    FORTRAN  A

      SUBROUTINE SET10(LIMIT)                                          SET00010
C*    SET10 READS THE CONTROL CARDS FOR THE NDIM PROCESSOR             SET00020
C*                                                                     SET00030
      IMPLICIT INTEGER (A-Z)                                           SET00040
      DIMENSION CODE(10),CARD(62),EQUCOM(3),ACARD(20)                  SET00050
C                                                                      SET00060
      DATA NPUT/10/                                                    SET00070
      DATA  EQUCOM/2,'=',',','/                                        SET00080
      DATA BLANK/' '/,CMCD/'C'/,FMCD/'F'/,MMCD/'M'/,PMCD/'P'/,         SET00090
     1     SMCD/'S'/,UMCD/'U'/                                         SET00100
      DATA CODE/'CHAN','DATA','MAPF','HISF',                           SET00110
     * 'OPTI','DATE','COMM','HED1','HED2','*END'/                     SET00120
CCCCCC                                                                 SET00130
CCCCCC                                                                 SET00140
      INCLUDE COMBK1.LIST                                              SET00150
      INCLUDE COMBK4.  LIST                                            SET00160
      INCLUDE COMBKA.LIST                                              SET00170
C     INCLUDE CMBK11.LIST                                              SET00180
      COMMON/INFORM/NUCLS2,NOSUR2,NOFET2,VARSZ2,TOTVT2,NOFLD2,         SET00190
     *              AVAR2,COVAR2,CLSID2,SURNO2,SURDS2,FLDSV2,VERTX2,   SET00200
     *              FETVC2(30),SURVC2(75),SURPTR(75),CLSVC2(60),       SET00210
     *              KEYPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),            SET00220
     *              GRPCHK(61),GROUPS(124)                             SET00230
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                   SET00240
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                SET00250
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))              SET00260
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,        SET00270
     *              HISFIL,HISKEY,IRFORM,ERIPTR,ERPKEY,MAPUNT,NOFILE,  SET00280
     *     DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL              SET00290
     *   ,NHSTUN,NHSTFI,SCTNUN,MAPFIL                                  SET00300
     *   ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,            SET00310
     *   CRDUNT,PRTUNT,RANDIO                                          SET00320
      COMMON /NDIM/NCLSCH,CLRVEC(30),MAXVEC,MAPKEY,                    SET00330
     *      CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(4),SIZE,TOTMNS      SET00340
     *    ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,BUFLEN,ID3,COLOR3,NODUMP  SET00350
     *    ,IDATA1,TOTVEC                                               SET00360
CSEND                                                                  SET00370
CCCC                                                                   SET00380
CCCC  INITIALIZE PARAMETERS                                            SET00390
CCC                                                                    SET00400
      NOCLS2 = 0                                                       SET00410
      NOSUR2 = 0                                                       SET00420
      JFPO = 0                                                         SET00430
      NCLSCH = 0                                                       SET00440
      MEANSW = 0                                                       SET00450
      NOFET2 = 0                                                       SET00460
      FIELD  = 1                                                       SET00470
      MAPKEY = 0                                                       SET00480
      CLSVC2(1) = BLANK                                                SET00490
      SURPTR(75) = BLANK                                               SET00500
C                                                                      SET00510
      WRITE(6,100)                                                     SET00520
  100 FORMAT(/11X,'INPUT SUMMARY'//)                                   SET00530
C                                                                      SET00540
CCC  SETUP REREAD BUFFER                                               SET00550
CCC                                                                    SET00560
       CALL REREAD(30,80)                                              SET00570
C                                                                      SET00580
CCC  NOW READ THE NEXT CARD INTO THE BUFFER                            SET00590
CC                                                                     SET00600
  105 READ(21,115)(ACARD(I),I=1,20)                                    SET00610
  115 FORMAT(20A4)                                                     SET00620
      WRITE(30,115)(ACARD(I),I=1,20)                                   SET00630
      REWIND 30                                                        SET00640
C                                                                      SET00650
      READ(30,110)CODE1,CARD                                           SET00660
      REWIND 30                                                        SET00670
      COL = 0                                                          SET00680
      WRITE(6,120)CODE1,CARD                                           SET00690
  120 FORMAT(1X,A4,6X,62A1)                                            SET00700
  110 FORMAT(A4,6X,62A1)                                               SET00710
      DO 130  I=1,NPUT                                                 SET00720
      IF (CODE1 .EQ. CODE(I)) GO TO(150,140,210,250,        330,370,   SET00730
     *                              390,400,410,420),I                 SET00740
  130 CONTINUE                                                         SET00750
  135 WRITE(6,140)                                                     SET00760
  140 FORMAT(' INVALID CONTROL CARD - IGNORED ')                       SET00770
      GO TO 105                                                        SET00780
C                                                                      SET00790
```

```
FILE: SET10     FORTRAN  A

C       CHANNEL CARD                                                    SET00800
C                                                                       SET00810
  180 M=NXTCHR(CARD,COL)                                                SET00820
      IF (M .EQ. BLANK) GO TO 185                                       SET00830
      IF (M .EQ. PRCD) GO TO 160                                        SET00840
      IF (M .EQ. CHCD) GO TO 170                                        SET00850
  183 WRITE(A,155)                                                      SET00860
  185 FORMAT(' ERROR ON CHANNELS CARD')                                 SET00870
      GO TO 105                                                         SET00880
  160 J = FIND12(CARD,COL,EQUCOM)                                       SET00890
      IF ( J .NE. 2 ) GO TO 183                                         SET00900
      NOFET2 = NUMBER(CARD,COL,FETVC2,NOFET2)                           SET00910
      COL = COL - 1                                                     SET00920
      CALL ORDER(FETVC2,NOFET2)                                         SET00930
      GO TO 180                                                         SET00940
  170 J = FIND12(CARD,COL,EQUCOM)                                       SET00950
      IF (J .NE. 2 ) GO TO 183                                          SET00960
      NCLRCH = NUMBER(CARD,COL,CLRVEC,NCLRCH)                           SET00970
      COL = COL - 1                                                     SET00980
      CALL ORDER(CLRVEC,NCLRCH)                                         SET00990
      GO TO 180                                                         SET01000
C                                                                       SET01010
C       DATA FILE CARD                                                  SET01020
C                                                                       SET01030
  190 M = NXTCHR(CARD,COL)                                              SET01040
      IF (M .EQ. BLANK) GO TO 195                                       SET01050
      IF (M .EQ. UNCD) GO TO 190                                        SET01060
      IF (M .EQ. FNCD) GO TO 200                                        SET01070
  195 WRITE(A,197)                                                      SET01080
  197 FORMAT(' ERROR ON DATA FILE CARD')                                SET01090
      GO TO 105                                                         SET01100
  190 J = FIND12(CARD,COL,EQUCOM)                                       SET01110
      IF (J .NE. 2) GO TO 195                                           SET01120
      M = NUMBER(CARD,COL,DATAPE,ZERO)                                  SET01130
      COL = COL - 1                                                     SET01140
      GO TO 180                                                         SET01150
  200 J = FIND12(CARD,COL,EQUCOM)                                       SET01160
      IF ( J .NE. 2 ) GO TO 195                                         SET01170
      M = NUMBER(CARD,COL,DATFIL,ZERO)                                  SET01180
      DATFIL = DATFIL - 1                                               SET01190
      COL = COL - 1                                                     SET01200
      GO TO 180                                                         SET01210
C                                                                       SET01220
C       CLUSTER/CLASSIFICATION MAP TAPE                                 SET01230
C                                                                       SET01240
  210 M = NXTCHR(CARD,COL)                                              SET01250
      IF (M .EQ. BLANK) GO TO 105                                       SET01260
      IF (M .EQ. UNCD) GO TO 230                                        SET01270
      IF (M .EQ. FNCD) GO TO 240                                        SET01280
  215 WRITE(A,220)                                                      SET01290
  220 FORMAT(' ERROR ON DAS FILE CARD')                                 SET01300
      GO TO 105                                                         SET01310
  230 J = FIND12(CARD,COL,EQUCOM)                                       SET01320
      IF ( J .NE. 2)  GO TO 215                                         SET01330
      M = NUMBER(CARD,COL,MAPUNT,ZERO)                                  SET01340
      MAPKEY = 1                                                        SET01350
      COL = COL - 1                                                     SET01360
      GO TO 210                                                         SET01370
  240 J = FIND12(CARD,COL,EQUCOM)                                       SET01380
      IF (J .NE. 2) GO TO 215                                           SET01390
      M= NUMBER(CARD,COL,MAPFIL,ZERO)                                   SET01400
      MAPFIL = MAPFIL -1                                                SET01410
      COL = COL - 1                                                     SET01420
      GO TO 210                                                         SET01430
C                                                                       SET01440
C       N-DIM HISTOGRAM FILE                                            SET01450
C                                                                       SET01460
  250 M = NXTCHR(CARD,COL)                                              SET01470
      IF (M .EQ. BLANK) GO TO 105                                       SET01480
      IF (M .EQ. UNCD) GO TO 270                                        SET01490
      IF (M .EQ. FNCD) GO TO 280                                        SET01500
  260 WRITE(A,265)                                                      SET01510
  265 FORMAT(' ERROR ON N-DIM HISTOGRAM FILE CARD')                     SET01520
      GO TO 105                                                         SET01530
  270 J = FIND12(CARD,COL,EQUCOM)                                       SET01540
      IF (J .NE. 2) GO TO 260                                           SET01550
      M = NUMBER(CARD,COL,NHSTUN,ZERO)                                  SET01560
      COL = COL - 1                                                     SET01570
      GO TO 250                                                         SET01580
```

```
FILE: SET10    FORTRAN  A

  280 J = FIND12(CARD,COL,EQUCOM)                                        SET01580
      IF (J .NE. 2 ) GO TO 260                                           SET01600
      M = NUMBER(CARD,COL,NHSTFI,ZERO)                                   SET01610
      COL = COL - 1                                                      SET01620
      GO TO 250                                                          SET01630
C                                                                        SET01640
CCC   OPTION CARD                                                        SET01650
C                                                                        SET01660
  330 M = NXTCHR(CARD,COL)                                               SET01670
      IF (M .EQ. BLANK) GO TO 105                                        SET01680
      IF (M .EQ. CHCD) GO TO 340                                         SET01690
      IF (M .EQ. SBCD) GO TO 350                                         SET01700
      IF (M .EQ. FHCD) GO TO 360                                         SET01710
      IF (M .EQ. MHCD) GO TO 363                                         SET01720
  333 WRITE(6,335)                                                       SET01730
  335 FORMAT(' ERROR ON OPTION CARD')                                    SET01740
      GO TO 105                                                          SET01750
  340 CLASS = 1                                                          SET01760
      FIELD = 0                                                          SET01770
      GO TO 365                                                          SET01780
  350 SUMCLS = 1                                                         SET01790
      FIELD = 0                                                          SET01800
      GO TO 365                                                          SET01810
  360 FIELD = 1                                                          SET01820
      GO TO 365                                                          SET01830
  363 MEANSW = 1                                                         SET01840
  365 J = FIND12(CARD,COL,EQUCOM)                                        SET01850
      IF ( J .EQ. 3) GO TO 330                                           SET01860
      IF ( J .EQ. -1) GO TO 105                                          SET01870
C                                                                        SET01880
CCC   DATE CARD                                                          SET01890
C                                                                        SET01900
  370 M = NXTCHR(CARD,COL)                                               SET01910
      IF ( M .EQ. BLANK) GO TO 105                                       SET01920
      READ(30,380)DATE                                                   SET01930
  380 FORMAT(10X,15A6)                                                   SET01940
      REWIND 30                                                          SET01950
      GO TO 105                                                          SET01960
C                                                                        SET01970
CCC   COMMENT CARD                                                       SET01980
C                                                                        SET01990
  390 M = NXTCHR(CARD,COL)                                               SET02000
      IF (M .EQ. BLANK) GO TO 105                                        SET02010
      READ(30,3A0)COMENT                                                 SET02020
      REWIND 30                                                          SET02030
      GO TO 105                                                          SET02040
C                                                                        SET02050
CCC   HED1                                                               SET02060
C                                                                        SET02070
  400 M = NXTCHR(CARD,COL)                                               SET02080
      READ(30,3A0) HED1                                                  SET02090
      REWIND 30                                                          SET02100
      GO TO 105                                                          SET02110
C                                                                        SET02120
CCC   HED2                                                               SET02130
C                                                                        SET02140
  410 M = NXTCHR(CARD,COL)                                               SET02150
      READ(30,3B0) HED2                                                  SET02160
      REWIND 30                                                          SET02170
      GO TO 105                                                          SET02180
C                                                                        SET02190
CCC   *END*                                                              SET02200
C                                                                        SET02210
C 420 CONTINUE                                                           SET02220
C                                                                        SET02230
      SIZE = NOFET2/4                                                    SET02240
      IF (MOD(NOFET2,4) .NE. 0) SIZE = SIZE + 1                          SET02250
      MAXVEC = LIMIT/SIZE                                                SET02260
C                                                                        SET02270
CC                                                                       SET02280
      WRITE(6,1000)                                                      SET02290
      IF (NOFET2 .NE. 0)WRITE(6,1010)(FETVC2(I),I=1,NOFET2)              SET02300
      IF (NCLUCH .NE. 0)WRITE(6,1020)(CLRVEC(I),I=1,NCLRCH)              SET02310
      IF (MAPKEY .EQ. 1) WRITE(6,1030)                                   SET02320
      IF (CLASS .EQ. 1)WRITE(6,1040)                                     SET02330
      IF (SUMCLS .EQ. 1)WRITE(6,1050)                                    SET02340
      IF (FIELD .EQ. 1)WRITE(6,1060)                                     SET02350
      IF (MEANSW .EQ. 1)WRITE(6,1070)                                    SET02360
 1000 FORMAT(//' USER HAS REQUESTED THE FOLLOWING OPTIONS :'/)           SET02370
```

FILE: SET10 FORTRAN A

```
 1010 FORMAT(' HISTOGRAM DATA VECTORS FROM CHANNELS ',16(I2,1X))          SET02380
 1020 FORMAT(' COLOR CODES ARE FROM CHANNELS ',8(I2,1X))                  SET02390
 1030 FORMAT(' CLUSTER/CLASSIFICATION TAPE IS BEING INPUT')               SET02400
 1040 FORMAT(' HISTOGRAM FIELDS BY CLASS')                                SET02410
 1050 FORMAT(' HISTOGRAM FIELDS BY SUBCLASS')                             SET02420
 1060 FORMAT(' HISTOGRAM FIELDS ON PER FIELDS BASES')                     SET02430
 1070 FORMAT(' COMPUTE MEANS OF INPUT FIELDS')                            SET02440
      RETURN                                                              SET02450
C                                                                        SET02460
      END                                                                SET02470
```

ORIGINAL PAGE IS
OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

```
FILE: STODAT   FORTRAN  A

      SUBROUTINE STODAT(ILINE,NSAMP,HIST,LIMIT,BEGIN1)                  STO00010
C*                                                                      STO00020
C*    STODAT READS AND STORES THE CLASSIFICATION/CLUSTER MAP ON DRUM    STO00030
C*                                                                      STO00040
      IMPLICIT INTEGER (A-Z)                                            STO00050
C     INCLUDE COMMON.LIST                                               STO00060
      COMMON/GLOBAL/HEAD(43),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         COM00010
     *              HISFIL,HISKEY,TWFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,   COM00020
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               COM00030
     *  ,NWSTUN,NWSTFI,SCTWUN,MAPFIL                                    COM00040
     *  ,DOTUNT,DOTAIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              COM00050
     *  CRDUNT,PRTUNT,HANDIO                                            COM00060
C*END                                                                   STO00080
      DIMENSION HIST(LIMIT),FETVEC(1),FLD(6),NLINE(4)                   STO00090
      TOTWRD = ILINE*NSAMP                                              STO00100
      IF (TOTWRD .LE. (DRMWDS-(DRUMAD-BEGIN1))) GO TO 120               STO00110
      WRITE(6,110)                                                      STO00120
  110 FORMAT(' NOT ENOUGH DRUM SPACE TO STORE DAS TAPE DATA')           STO00130
      CALL CMERR                                                        STO00140
C                                                                       STO00150
  120 CALL TAPHDR(MAPUNT,MAPFIL)                                        STO00160
      FETVEC(1) = 1                                                     STO00170
      NOFEAT = 1                                                        STO00180
      FLD(1) = 1                                                        STO00190
      FLD(2) = ILINE                                                    STO00200
      FLD(3) = 1                                                        STO00210
      FLD(4) = 1                                                        STO00220
      FLD(5) = NSAMP                                                    STO00230
      FLD(6) = 1                                                        STO00240
      BEGIN = BEGIN1                                                    STO00250
      CALL FLDINT(FLD   ,FETVEC,NOFEAT)                                 STO00260
      DUMPS = TOTWRD / LIMIT                                            STO00270
      IF (MOD(TOTWRD,LIMIT) .NE. 0) DUMPS = DUMPS + 1                   STO00280
      TOTLNS = LIMIT / NSAMP                                            STO00290
      IF (TOTLNS .GE. ILINE)  GO TO 140                                 STO00300
      DMP = DUMPS - 1                                                   STO00310
      DO 130  I=1,DMP                                                   STO00320
  130 NLINE(I) = TOTLNS                                                 STO00330
      NLINE(DUMPS) = ILINE - TOTLNS*DMP                                 STO00340
      GO TO 150                                                         STO00350
  140 NLINE(1) = ILINE                                                  STO00360
C                                                                       STO00370
  150 DO 200 J=1,DUMPS                                                  STO00380
      NUMLIN = NLINE(J)                                                 STO00390
      DO 160 K=1,NUMLIN                                                 STO00400
      WORDS = NSAMP*(K-1)                                               STO00410
  160 CALL LINERD(HIST(WORDS+1),ENDTAP)                                 STO00420
C                                                                       STO00430
CC    STORE ON HIGH SPEED DRUM                                          STO00440
C                                                                       STO00450
      NWORDS = WORDS + NSAMP                                            STO00460
      CALL RWRITE(BEGIN,HIST(1),NWORDS,ISTAT)                           STO00470
  200 BEGIN = BEGIN + NLINE(J) * NSAMP                                  STO00480
C                                                                       STO00490
      MAPFIL = MAPFIL + 1                                               STO00500
C                                                                       STO00510
      RETURN                                                            STO00520
      END                                                               STO00530
```

FILE: WRTFIL FORTRAN A

```
      SUBROUTINE WRTFIL(HIST,MEANS,ID,COLOR,FIELDS,VERTEX,I)                WRT00010
C                                                                           WRT00020
C*                                                                          WRT00030
C*    WRTFIL WRITES THE NDIM FILE                                           WRT00040
C*                                                                          WRT00050
      IMPLICIT INTEGER (A-Z)                                                WRT00060
      INCLUDE COMBK1.LIST                                                   WRT00070
CCC   INCLUDE COMBK6.LIST                                                   WRT00080
CC    INCLUDE CMBK11.LIST                                                   WRT00090
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,              WRT00100
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLDSV2,VERTX2,        WRT00110
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),            WRT00120
     *              KEPPTS(60),NOGRP,GRPNAM(50),GRPDEX(61),                 WRT00130
     *              GRPCHR(61),GROUPS(124)                                  WRT00140
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,             WRT00150
     *              HISFIL,HISKEY,INFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,       WRT00160
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    WRT00170
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                        WRT00180
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTHFL,HISTFL,PCHUNT,                  WRT00190
     *  CRDUNT,PRTUNT,RANDIO                                                WRT00200
      COMMON /NDIM/NCLRCH,CLRVEC(30),MAXVEC,MAPKEY,                         WRT00210
     *    CLASS,SUBCLS,FIELD,MEANSW,NOVEC,FLDINF(6),SIZE,TOTMNS             WRT00220
     *   ,CNTR1,CNTR2,ID1,ID2,COLOR1,COLOR2,RUFLEN,ID3,COLOR3,NODUMP        WRT00230
     *   ,IDATA1,TOTVEC                                                     WRT00240
CSEND                                                                       WRT00250
      DIMENSION HIST(1),MEANS(1),ISTAT(4),ID(1),COLOR(1),VERTEX(2,1)        WRT00260
      DIMENSION FIELDS(4,1)                                                 WRT00270
      REAL MEANS                                                            WRT00280
      DATA BLANK/' '/                                                       WRT00290
      I = I + 1                                                             WRT00300
      IF (I .GT. 1) GO TO 100                                               WRT00310
C                                                                           WRT00320
CCC   WRITE HEADER RECORD                                                   WRT00330
CC                                                                          WRT00340
      WRITE(NHSTUN) TOTMNS,SIZE,NOFET2,(FETVC2(I),I=1,NOFET2),              WRT00350
     *      NCLRCH,(CLRVEC(I),I=1,NCLRCH)                                   WRT00360
      ENDFILE NHSTUN                                                        WRT00370
C                                                                           WRT00380
CCCC                                                                        WRT00390
CCCC  WRITE RECORD 1                                                        WRT00400
CCCC                                                                        WRT00410
  100 WRITE(NHSTUN) NOFLD2,NOSUB2,TOTVT2,NOVEC                              WRT00420
C                                                                           WRT00430
CCC   WRITE RECORD 2                                                        WRT00440
CCC                                                                         WRT00450
      WRITE(NHSTUN) CLSVC2(1),(SUBVC2(I),I=1,NOSUB2),((FIELDS(I,J),         WRT00460
     *        I=1,4),J=1,NOFLD2),((VERTEX(I,J),I=1,2),J=1,TOTVT2)           WRT00470
C                                                                           WRT00480
CCC   WRITE RECORD 3                                                        WRT00490
CCC                                                                         WRT00500
      IF (TOTMNS .GT. 0) WRITE(NHSTUN)(MEANS(I),I=1,TOTMNS)                 WRT00510
C                                                                           WRT00520
CCC   WRITE RECORD 4                                                        WRT00530
CCC                                                                         WRT00540
      II = SIZE*NOVEC                                                       WRT00550
      WRITE(NHSTUN)(HIST(I),I=1,II)                                         WRT00560
      IF (NODUMP .EQ. 0) GO TO 160                                          WRT00570
C                                                                           WRT00580
CCC   READ ID INFORMATION INTO CORE                                         WRT00590
CCCC  WRITE RECORD 5                                                        WRT00600
CC                                                                          WRT00610
      VECDRM = NODUMP*RUFLEN                                                WRT00620
      VECARY = NOVEC - VECDRM                                               WRT00630
      CALL RREAD(ID3,HIST(1),VECDRM,ISTAT(1))                               WRT00640
      IF (VECARY .EQ. 0) GO TO 115                                          WRT00650
      DO 110 I=1,VECARY                                                     WRT00660
  110 HIST(VECDRM+I) = ID(I)                                                WRT00670
  115 CONTINUE                                                              WRT00680
  120 IF (ISTAT(1) .EQ. 1) GO TO 120                                        WRT00690
      WRITE(NHSTUN)(HIST(I),I=1,NOVEC)                                      WRT00700
C                                                                           WRT00710
CCC   WRITE REC 6                                                           WRT00720
CC                                                                          WRT00730
      CALL RREAD(CNTR2,HIST(1),NOVEC,ISTAT(3))                              WRT00740
  125 IF (ISTAT(3) .EQ. 1) GO TO 125                                        WRT00750
      WRITE(NHSTUN)(HIST(I),I=1,NOVEC)                                      WRT00760
C                                                                           WRT00770
CCC   WRITE REC 7                                                           WRT00780
CC                                                                          WRT00790
```

FILE: WRTFIL FORTRAN A

```
      IF (TOTMNS .NE. 0) GO TO 180                                      WRT00800
      IF(NCLRCH .EQ. 0) GO TO 180                                       WRT00810
      CALL RREAD(COLOR3,HIST(1),VECDRM,ISTAT(2))                        WRT00820
      IF (VECARY .EQ. 0) GO TO 135                                      WRT00830
      DO 130 I=1,VECARY                                                 WRT00840
  130 HIST(VECDRM+I) = COLOR(I)                                         WRT00850
  135 IF (ISTAT(2) .EQ. 1) GO TO 135                                    WRT00860
  140 WRITE(NHSTUN)(HIST(I),I=1,NOVEC)                                  WRT00870
      GO TO 180                                                         WRT00880
C                                                                       WRT00890
CCC   INFORMATION DID NOT NEED TO BE STORED ON DRUM                     WRT00900
CCC                                                                     WRT00910
CCC                                                                     WRT00920
CCC   RECORD 5                                                          WRT00930
C     160 WRITE(NHSTUN)(ID(I),I=1,NOVEC)                                WRT00940
CCC                                                                     WRT00950
CCC   RECORD 6                                                          WRT00960
CCC                                                                     WRT00970
CCC                                                                     WRT00980
CCC   HISTOGRAM INFORMATION IS ALWAYS STORED ON HIGH SPEED DRUM         WRT00990
CCC                                                                     WRT01000
  165 CALL RREAD(CNTR2,HIST(1),NOVEC,ISTAT(3))                          WRT01010
  170 IF (ISTAT(3) .EQ. 1) GO TO 170                                    WRT01020
      WRITE(NHSTUN)(HIST(I),I=1,NOVEC)                                  WRT01030
C                                                                       WRT01040
CCC   WRITE REC 7                                                       WRT01050
CCC                                                                     WRT01060
      IF (TOTMNS .NE. 0) GO TO 180                                      WRT01070
      IF (NCLRCH .GT. 0) WRITE(NHSTUN)(COLOR(I),I=1,NOVEC)              WRT01080
C                                                                       WRT01090
  180 ENDFILE NHSTUN                                                    WRT01100
C                                                                       WRT01110
CCC   WRITE FILE INFO ON LINE PRINTER                                   WRT01120
CCC                                                                     WRT01130
      CALL WRTFLD(FIELDS,VERTEX,NOFLD2,2,CLSVC2,SURVC2)                 WRT01140
C                                                                       WRT01150
      WRITE(4,190)TOTVEC,NOVEC                                          WRT01160
  190 FORMAT(////55X,'TOTAL NO. OF VECTORS =',I6/48X,'TOTAL NO. OF UNIO WRT01170
     *UE VECTORS =',I6)                                                 WRT01180
C                                                                       WRT01190
      IF (TOTMNS .EQ. 0) RETURN                                         WRT01200
      WRITE(4,200)(BLANK,CLRVEC(I),I=1,NCLRCH)                          WRT01210
  200 FORMAT(////T60,'FIELD MEANS'//T44,4(A1,'CH(',I2,')',5X))          WRT01220
C                                                                       WRT01230
      WRITE(4,210)(MEANS(I),I=1,TOTMNS)                                 WRT01240
  210 FORMAT(T44,4(F7.2,5X))                                            WRT01250
C                                                                       WRT01260
      RETURN                                                            WRT01270
      END                                                               WRT01280
                                                                        WRT01290
                                                                        WRT01300
```

16. SCTRPL PROCESSOR

```
FILE: SCTRPL   FORTRAN  A

      SUBROUTINE SCTRPL(ARRAY,TOP)                                              SCT00010
      IMPLICIT INTEGER (A-Z)                                                    SCT00020
C*    SCTRPL IS THE DRIVER FOR THE SCATTER PLOT PROCESSOR                       SCT00030
C*                                                                              SCT00040
C*                                                                              SCT00050
      DATA LIMIT/12000/                                                         SCT00060
C     INCLUDE COMMK1.LIST                                                       SCT00070
C     INCLUDE CMMK12.LIST                                                       SCT00080
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,                  SCT00090
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLOSV2,VERTX2,            SCT00100
     *              FETVC2(30),SUMVC2(75),SUBPTR(75),CLSVC2(60),                SCT00110
     *             -KEEPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                     SCT00120
     *              GRPCHR(61),GROUPS(124)                                      SCT00130
      COMMON/SCTTER/RSCALE,XYSCLF,CLRVEC(30),NCLRCH,CLRKEY,LOG,                 SCT00140
     *  FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,                       SCT00150
     *  YHI,YSIZ,NBINS,SYMMTX(32),RMATRX(60),BVEC(30),NRVCHN,NOFEAT             SCT00160
     *   ,SCALKY,MENADR,FLDADR,PNTADR,IDADR,NC,RMFEAT,RMCOMB                    SCT00170
     *   ,NOVEC,TOTMNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,           SCT00180
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD              SCT00190
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE                                 SCT00200
     *  ,NOSUB                                                                  SCT00210
CSEND                                                                           SCT00220
      DIMENSION ARRAY(1),BUFF(12000)                                            SCT00230
C                                                                               SCT00240
      CALL SFT1(ARRAY(1),ARRAY(1),BUFF(1))                                      SCT00250
C                                                                               SCT00260
CC    COMPUTE ADDRESSES                                                         SCT00270
C                                                                               SCT00280
   10 CALL SETADR(&20,&30,TOP,BUFF,LIMIT)                                       SCT00290
C                                                                               SCT00300
C*    SCATTR IS THE MAIN DRIVER FOR CREATING THE SPECTRAL PLOTS                 SCT00310
C*                                                                              SCT00320
   20 CALL SCATTR(ARRAY(FLDADR),ARRAY(VERTX2),ARRAY(VECTR1),ARRAY(MENADR,       SCT00330
     *),      BUFF(1),BUFF(1),ARRAY(DATA1),TOP,LIMIT,BUFF(1))                   SCT00340
C                                                                               SCT00350
CC    PROCESS ANOTHER FILE                                                      SCT00360
C                                                                               SCT00370
      GO TO 10                                                                  SCT00380
C                                                                               SCT00390
CCC                                                                             SCT00400
CC    $END* CARD                                                                SCT00410
C                                                                               SCT00420
   30 READ(21,100)CARD                                                          SCT00430
  100 FORMAT(A4)                                                                SCT00440
C                                                                               SCT00450
      RETURN                                                                    SCT00460
      END                                                                       SCT00470
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: CLRCOD FORTRAN A

```
      SUBROUTINE CLRCOD(IB,MEANS,IDATA,IPOSTN,II)                                CLR00010
C                                                                                CLR00020
      IMPLICIT INTEGER (A-Z)                                                     CLR00030
      REAL MEANS(1)                                                              CLR00040
C                                                                                CLR00050
C     INCLUDE COMBK1,LIST                                                        CLR00060
C     INCLUDE CMBK12,LIST                                                        CLR00070
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,                   CLR00080
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,             CLR00090
     *              FETVC2(30),SUBVC2(75),SUPPTR(75),CLSVC2(60),                 CLR00100
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                      CLR00110
     *              GRPCHK(61),GROUPS(124)                                       CLR00120
      COMMON/SCTTER/RSCALF,XYSCLE,CLRVEC(30),NCLPCH,CLRKEY,LOG,                  CLR00130
     *  FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,                        CLR00140
     *  YHI,YSIZ,NBINS,SYMMTX(32),BMATRX(60),RVEC(30),NRVCHN,NOFEAT              CLR00150
     *   ,SCALKY,MENADR,FLDADR,PNTADR,IDADR,NC,BMFEAT,BMCOMB                     CLR00160
     *    ,NOVEC,TOTMNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,           CLR00170
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD               CLR00180
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE                                  CLR00190
     *  ,NOSUB                                                                   CLR00200
C$END                                                                            CLR00210
C                                                                                CLR00220
      DIMENSION IDATA(1)                                                         CLR00230
      LOGICAL*1 LDUM(4),LLDUM(4)                                                 CLR00240
      EQUIVALENCE (IDUM,LDUM(1)),(IIDUM,LLDUM(1))                                CLR00250
C                                                                                CLR00260
      IF (CLRKEY .NE. 3) GO TO 50                                                CLR00270
C                                                                                CLR00280
C     COLOR CODES (RADIANCE VALUES) ARE COMING FROM N-DIM HIST FILE              CLR00290
C                                                                                CLR00300
      COLADR = DRMCLR + NVEC*(II-1) + IB - 1                                     CLR00310
      CALL RREAD(COLADR,CODE,1,ISTAT1)                                           CLR00320
  105 IF (ISTAT1 .EQ. 1) GO TO 105                                               CLR00330
C                                                                                CLR00340
      IIDUM=0                                                                    CLR00350
      IDUM=CODE                                                                  CLR00360
      DO 10 I=1,NC                                                               CLR00370
      III=XSIZ*(I-1)+IPOSTN                                                      CLR00380
      LLDUM(4)=LDUM(I)                                                           CLR00390
   10 IDATA(III)=IIDUM                                                           CLR00400
C                                                                                CLR00410
      RETURN                                                                     CLR00420
C                                                                                CLR00430
C     COLOR CODES (STAT MEANS OR USER INPUT) ARE STORED IN CORE UNPACKED         CLR00440
C                                                                                CLR00450
   50 IDADR = DRMID + NVEC*(II-1) + IB - 1                                       CLR00460
      CALL RREAD(IDADR,IDNUM,1,ISTA1)                                            CLR00470
   55 IF (ISTA1 .EQ. 1) GO TO 55                                                 CLR00480
      DO 80 I=1,NC                                                               CLR00490
      III = IPOSTN + XSIZ*(I-1)                                                  CLR00500
      JJ = (IDNUM-1)*NC + I                                                      CLR00510
   80 IDATA(III) = MEANS(JJ) + 0.5                                               CLR00520
      RETURN                                                                     CLR00530
      END                                                                        CLR00540
```

FILE: CLRKYS FORTRAN A

```
      SUBROUTINE CLRKYS(XSIZ,IDATA,NOSUB2,CH,MEANS,NC)
C
C*
C*    CLRKYS ADDS  THE COLOR KEYS TO A UNIVERSAL FORMAT TAPE
C*    THE COLORS ARE OUTPUT AS SQUARES INAGES (10X10)
C
      IMPLICIT INTEGER (A-Z)
      REAL MEANS(NC,NOSUB2)
C
      DIMENSION IDATA(XSIZ,CH)
C
      LSTLIN = 0
      LINE = 0
      TOTKEY = 0
      NKEYS = XSIZ/11
      NOKEY = NOSUB2
C
   90 DO 100 J=1,CH
C*    WRITE A SCAN LINE OF ZEROS - USED FOR SEPARATING THE THE COLORS
C*
      DO 100 I=1,XSIZ
  100 IDATA(I,J) = 0.
C
      CALL WRTLN(IDATA,LSTLIN)
      LINE = LINE + 1
C
  110 IF (NKEYS .LE. NOKEY) NOKEY = NKEYS
      KK = 0
C
      DO 150 I=1,NOKEY
      TOTKEY = TOTKEY + 1
      DO 140 J=1,NC
      DO 130 K=1,10
      KK = (I-1)*11 + K
  130 IDATA(KK,J) = MEANS(J,TOTKEY) + 0.5
  140 CONTINUE
C*
C*    WRITE A SCAN LINE OF COLORS
C*
  150 CONTINUE
      NOKEY = NOSUB2 - TOTKEY
C
      DO 160 I=1,10
      IF (NOKEY .LE. 0 .AND. I .EQ. 10) LSTLIN = -1
  160 CALL WRTLN(IDATA,LSTLIN)
      LINE = LINE + 10
      IF (NOKEY .LE. 0) GO TO 170
      GO TO 90
C
  170 CONTINUE
C
      WRITE(6,200)LINE
  200 FORMAT(/T54,'COLOR KEYS = ',I4,' LINES')
C
      RETURN
      END
```

FILE: CNTER FORTRAN A

```
      SUBROUTINE CNTER(IB,IDATA,IPOSTN,II,COUNTR)                       CNT00010
C                                                                       CNT00020
C*    CNTER STORE THE FREQUENCY COUNT IN IDATA                          CNT00030
C*    THE FREQUENCY IS STORED ON DRUM                                   CNT00040
C*                                                                      CNT00050
      IMPLICIT INTEGER (A-Z)                                            CNT00060
C                                                                       CNT00070
C     INCLUDE CMRK12.LIST                                               CNT00080
      COMMON/SCTTER/RSCALE,XYSCLE,CLRVEC(30),NCLRCH,CLRKEY,LOG,
     *  FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,
     *  YHI,YSIZ,NRINS,SYMMTX(32),PMATRX(60),BVEC(30),NBVCHN,NOFEAT
     *   ,SCALKY,MENADR,FLDADR,PNTADR,IDADR,NC,BMFEAT,BMCOMB
     *    ,NOVEC,TOTMNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE
     *   ,NOSUB
CSEND
C                                                                       CNT00100
      DIMENSION IDATA(1)                                                CNT00110
C                                                                       CNT00120
C     COMPUTE DRUM ADDRESSES                                            CNT00130
C                                                                       CNT00140
      CTRADR = NVEC* (II-1) + IB + DRMCNT - 1                           CNT00150
      CALL RREAD(CTRADR,COUNTR,1,ISTAT1)                                CNT00160
  105 IF (ISTAT1 .EQ. 1) GO TO 105                                      CNT00170
C                                                                       CNT00180
C     THE VECTOR COUNTER IS THE LAST CHANNEL                            CNT00190
C                                                                       CNT00200
      I = XSIZ*NC + IPOSTN                                              CNT00210
      IF (COUNTR .GT. 255) COUNTR = 255                                 CNT00220
      IDATA (I) = COUNTR                                                CNT00230
C                                                                       CNT00240
      RETURN                                                            CNT00250
      END                                                               CNT00260
                                                                        CNT00270
```

```
FILE: LINPLT   FORTRAN  A
)
      SUBROUTINE LINPLT                                                          LIN00010
C                                                                                LIN00020
C*    LINPLT CREATES THE PIXEL FREQUENCY PLOT ON DRUM AND LATER PRINTS           LIN00030
C*    THE IMAGE ON THE LINE PRINTER                                              LIN00040
C*                                                                               LIN00050
C*    THERE ARE 3 ENTRIES :                                                      LIN00060
C*    1. SUBROUTINE LINPLT - COMPUTES THE SCALES                                 LIN00070
C*    2. ENTRY STOPTS      - COMPUTES THE POSITION OF THE PIXEL ON THE           LIN00080
C*                           PLOT (DRUM ADDRESS)                                 LIN00090
C*                           *** MUST BE CALLED FOR EVERY PIXEL ***              LIN00100
C*    3. ENTRY PRTPLT      - PRINTS THE PLOT ON THE LINE PRINTER                 LIN00110
C*                           *** PIXEL MUST BE POSITIVE ***                      LIN00120
C                                                                                LIN00130
      IMPLICIT INTEGER (A-Z)                                                     LIN00140
      REAL XSCALE,XSHFT,YSCALE,YSHFT,SCALEY,SCALEX,SHFTY,SHFTX                   LIN00150
      REAL SUM1,COUNT,LOG2                                                       LIN00160
C                                                                                LIN00170
      DIMENSION YAXIS(11),XAXIS(11)                                              LIN00180
C                                                                                LIN00190
C     INCLUDE COMBKA,LIST                                                        LIN00200
CC    INCLUDE COMBKA,LIST                                                        LIN00210
C     INCLUDE CMBK12,LIST                                                        LIN00220
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                             LIN00230
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                          LIN00240
     2           (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                         LIN00250
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,                  LIN00260
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,            LIN00270
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                         LIN00280
     *  ,NHSTUN,NHSTFI,SCTHUN,MAPFIL                                             LIN00290
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                       LIN00300
     *  CRDUNT,PRTUNT,RANDIO                                                     LIN00310
      COMMON/SCTTER/RSCALE,XYSCLE,CLRVEC(30),NCLRCH,CLRKEY,LOG,                  LIN00320
     *  FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,                        LIN00330
     *  YHI,YSIZ,NPAINS,SYMMTX(32),HMATRX(60),BVEC(30),NBVCHN,NOFEAT             LIN00340
     *   ,SCALKY,MENADR,FLDADR,PNTADR,IDADR,NC,HMFEAT,PMCOMB                     LIN00350
     *    ,NOVEC,TOTANS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,           LIN00360
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOHEAD,LREAD               LIN00370
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE                                  LIN00380
     *  ,NOSUB                                                                   LIN00390
C$END                                                                            LIN00400
C                                                                                LIN00410
CC                                                                               LIN00420
      DATA BLANK/' '/                                                            LIN00430
      LOG2 = ALOG10(2.0)                                                         LIN00440
      MAXSUM = 1                                                                 LIN00450
      IF (XYSCLE .EQ. 0) GO TO 70                                                LIN00460
C                                                                                LIN00470
CC    DATA IS RESCALED TO 101 BINS                                               LIN00480
C                                                                                LIN00490
      XSIZE = XSIZ                                                               LIN00500
      YSIZE = YSIZ                                                               LIN00510
      IF(XSIZE .GT. 101) XSIZE = 101                                             LIN00520
      IF(YSIZE .GT. 101) YSIZE = 101                                             LIN00530
C                                                                                LIN00540
CC    RANGES FOR THE X-AXIS                                                      LIN00550
C                                                                                LIN00560
      OSIZ =  XSIZE / 10 + 1                                                     LIN00570
      XSCALE = FLOAT(XLO-XHI)/(XSIZE - 1)                                        LIN00580
      XSHFT = FLOAT(XSIZE*XHI-XLO)/(XSIZE-1)                                     LIN00590
      DO 50  I=1,OSIZ                                                            LIN00600
       XAXIS(OSIZ-I+1) = (10*I-9)*XSCALE+XSHFT + .501                            LIN00610
   50 CONTINUE                                                                   LIN00620
      SCALEY = FLOAT(1-YSIZE)/YHI-YLO                                            LIN00630
      SHFTY = -YHI*SCALEY + 1.0                                                  LIN00640
      SCALEX = FLOAT(1-XSIZE)/XHI-XLO                                            LIN00650
      SHFTX = -XHI*SCALEX + 1.0                                                  LIN00660
C                                                                                LIN00670
CC    RANGES FOR THE Y-AXIS                                                      LIN00680
C                                                                                LIN00690
      ISIZ = YSIZE / 10 + 1                                                      LIN00700
      YSCALE = FLOAT(YLO-YHI)/(YSIZE-1)                                          LIN00710
      YSHFT = FLOAT(YSIZE*YHI-YLO)/(YSIZE-1)                                     LIN00720
      DO 60  I=1,ISIZ                                                            LIN00730
      YAXIS(ISIZ-I+1) = (10* I-9) *YSCALE + YSHFT + .501                         LIN00740
   60 CONTINUE                                                                   LIN00750
      RETURN                                                                     LIN00760
C                                                                                LIN00770
CC    DATA IS NOT RESCALED                                                       LIN00780
C                                                                                LIN00790
```

ORIGINAL PAGE IS OF POOR QUALITY

FILE: LINPLT FORTRAN A

```
   70  OSIZ = 11                                                          LIN00800
       ISIZ = 11                                                          LIN00810
       XSIZE = 101                                                        LIN00820
       YSIZE = 101                                                        LIN00830
       DO 75   I=1,OSIZ                                                  LIN00840
       YAXIS(I) = YLO + (I-1)*10                                          LIN00850
   75  XAXIS(I) = XLO + (I-1)*10                                          LIN00860
       RETURN                                                             LIN00870
C                                                                         LIN00880
CCC    * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * LIN00890
C                                                                         LIN00900
       ENTRY STOPTST(COUNTR,LINE,SAMPLE)                                  LIN00910
       REAL SAMPLE,LINE                                                   LIN00920
C                                                                         LIN00930
CCC    COMPUTE POSITION ON GRAPH FOR (SAMPLE,LINE)                        LIN00940
C                                                                         LIN00950
       IF (XYSCLE .EQ. 1) GO TO 110                                       LIN00960
       YPOINT = YSIZE - (LINE - YLO)                                      LIN00970
       XPOINT = SAMPLE-XLO                                                LIN00980
C                                                                         LIN00990
       IF (YPOINT .GT. 0) GO TO 80                                        LIN01000
       YPT = YPT + 1                                                      LIN01010
       RETURN                                                             LIN01020
   80  IF (YPOINT .LE. 101) GO TO 90                                      LIN01030
       YPT = YPT + 1                                                      LIN01040
       RETURN                                                             LIN01050
   90  IF (XPOINT .GE. 0) GO TO 100                                       LIN01060
       XPT = XPT + 1                                                      LIN01070
       RETURN                                                             LIN01080
  100  IF (XPOINT .LE. 100) GO TO 115                                     LIN01090
       XPT = XPT + 1                                                      LIN01100
       RETURN                                                             LIN01110
  110  CONTINUE                                                           LIN01120
       YPOINT = LINE*SCALEY + SHFTY + .501                                LIN01130
       XPOINT = XSIZE - (SAMPLE*SCALEX + SHFTX + .501) + 1                LIN01140
       IF (XPOINT .LT. 0)XPOINT = 0                                       LIN01150
       IF (XPOINT .GE. XSIZE) XPOINT = XSIZE - 1                          LIN01160
       IF (YPOINT .LT. 1) YPOINT = 1                                      LIN01170
       IF (YPOINT .GT. YSIZE) YPOINT = YSIZE                              LIN01180
  115  XYADR = (YPOINT - 1)*XSIZE + XPOINT + DRMPLT                       LIN01190
C                                                                         LIN01200
CCC    LOG X BASE 2 OF THE FREQUENCY                                      LIN01210
C                                                                         LIN01220
       IF ( LOG .NE. 1) GO TO 117                                         LIN01230
       CALL RREAD(XYADR,SUM1,1,ISTAT1)                                    LIN01240
  122  IF (ISTAT1 .EQ. 1) GO TO 122                                       LIN01250
       COUNT = COUNTR                                                     LIN01260
       SUM1 = SUM1 + ALOG10(COUNT) / LOG2                                 LIN01270
       CALL RWRITE(XYADR,SUM1,1,ISTAT2)                                   LIN01280
  121  IF (ISTAT2 .EQ. 1) GO TO 121                                       LIN01290
       SUM = SUM1 + 0.5                                                   LIN01300
       GO TO 125                                                          LIN01310
  117  CONTINUE                                                           LIN01320
       CALL RREAD(XYADR,SUM,1,ISTAT1)                                     LIN01330
  120  IF (ISTAT1 .EQ. 1) GO TO 120                                       LIN01340
C                                                                         LIN01350
CCCC   TOTAL NO. OF OCCURRENCES FOR ALL THE DATA VECTORS ASSIGNED         LIN01360
        TO THIS (SAMPLE,LINE) POSITION                                    LIN01370
C                                                                         LIN01380
       SUM = SUM + COUNTR                                                 LIN01390
       CALL RWRITE(XYADR,SUM,1,ISTAT2)                                    LIN01400
  123  IF (ISTAT2 .EQ. 1) GO TO 123                                       LIN01410
C                                                                         LIN01420
CCC    SAVE THE LARGEST NO. OF OCCURRENCES                                LIN01430
C                                                                         LIN01440
  125  CONTINUE                                                           LIN01450
       IF(SUM .GT. MAXSUM) MAXSUM = SUM                                   LIN01460
       RETURN                                                             LIN01470
C                                                                         LIN01480
CCC    * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * LIN01490
C                                                                         LIN01500
       ENTRY PRTPLT(PNTR,PNTRS)                                           LIN01510
C                                                                         LIN01520
       REAL PNTRS(1)                                                      LIN01530
       DIMENSION BINS(16),PNTR(1) ,PNTR1(101)                             LIN01540
C                                                                         LIN01550
       IF (LOG .NE. 1) GO TO 129                                          LIN01560
       CALL RREAD(DRMPLT,PNTRS,10201,ISTAT3)                              LIN01570
  126  IF (ISTAT3 .EQ. 1) GO TO 126                                       LIN01580
```

FILE: LINPLT FORTRAN A

```
C
CCC     IF FREQ. LESS THAN ZERO, SET THE FREQ. TO 1                          LIN01590
CC                                                                           LIN01600
        DO 128 I=1,10201                                                     LIN01610
        IF (PNTRS(I) .EQ. 0.0) GO TO 127                                     LIN01620
        IF (PNTRS(I) .GE. 1.0) GO TO 127                                     LIN01630
        PNTR(I) = 1                                                          LIN01640
        GO TO 128                                                            LIN01650
  127   PNTR(I) = PNTRS(I)          + 0.5                                    LIN01660
  128   CONTINUE                                                             LIN01670
        GO TO 131                                                            LIN01680
  129   CONTINUE                                                             LIN01690
        CALL BREAD(DRMPLT,PNTR,10201,ISTAT3)                                 LIN01700
  130   IF (ISTAT3 .EQ. 1) GO TO 130                                         LIN01710
  131   CONTINUE                                                             LIN01720
C                                                                            LIN01730
CCC     SET BIN LEVELS                                                       LIN01740
CC                                                                           LIN01750
        CALL SETMRG(66,0,66)                                                 LIN01760
        WRITE(6,HEAD)                                                        LIN01770
        WRITE(6,135)                                                         LIN01780
  135   FORMAT(//T53,' PIXEL FREQUENCY SCATTER PLOT')                        LIN01790
        IF (NBINS .GT. MAXSUM) NBINS = MAXSUM                                LIN01800
        RANGE = MAXSUM / NBINS                                               LIN01810
        IF (MOD(MAXSUM,NBINS) .NE. 0) RANGE = RANGE + 1                      LIN01820
        DO 140 I=1,NBINS                                                     LIN01830
  140   BINS(I) = RANGE*I                                                    LIN01840
        WRITE(6,150)(BINS(I),I=1,NBINS)                                      LIN01850
  150   FORMAT(//17X,1H1,2X,I3,9(3X,I3))                                     LIN01860
        NB = NBINS + 1                                                       LIN01870
        NBS = 2*NBINS                                                        LIN01880
        DO 155 JJ=1,4                                                        LIN01890
        WRITE(6,160)((SYMMTX(J),I=1,6),J=1,NBINS)                            LIN01900
        WRITE(6,163)((SYMMTX(J),I=1,6),J=NB,NBS)                             LIN01910
  163   FORMAT(1H+,15X,96A1)                                                 LIN01920
  155   CONTINUE                                                             LIN01930
  160   FORMAT(17X,96A1)                                                     LIN01940
        WRITE(6,165)                                                         LIN01950
  165   FORMAT(//)                                                           LIN01960
        DO 200 J=1,YSIZE                                                     LIN01970
        KK = (J-1)*XSIZE + 1                                                 LIN01980
        KKK = KK + YSIZE - 1                                                 LIN01990
        DO 180 I=1,XSIZE                                                     LIN02000
        II = (J-1)*XSIZE + I                                                 LIN02010
        IF (PNTR(II) .NE. 0) GO TO 170                                       LIN02020
        PNTR(II) = BLANK                                                     LIN02030
        PNTR1(I) = BLANK                                                     LIN02040
        GO TO 180                                                            LIN02050
  170   BINLEV = PNTR(II) / RANGE                                            LIN02060
        IF (MOD(PNTR(II),RANGE) .NE. 0) BINLEV = BINLEV + 1                  LIN02070
        PNTR(II) = SYMMTX(BINLEV)                                            LIN02080
        PNTR1(I) = SYMMTX(BINLEV+NBINS)                                      LIN02090
  180   CONTINUE                                                             LIN02100
C                                                                            LIN02110
CCC     PRINT A LINE                                                         LIN02120
CC                                                                           LIN02130
        L = 11 - J/10                                                        LIN02140
        IF (MOD(J,10) .EQ. 1) GO TO 190                                      LIN02150
        WRITE(6,193)                                                         LIN02160
  193   FORMAT(16X,1H-)                                                      LIN02170
        WRITE(6,195)(PNTR(K),K=KK,KKK)                                       LIN02180
        WRITE(6,195)(PNTR1(K),K=1,101)                                       LIN02190
  195   FORMAT( 1H+    ,T17,101A1)                                           LIN02200
        GO TO 200                                                            LIN02210
  190   WRITE(6,195)YAXIS(L)                                                 LIN02220
  195   FORMAT(10X,I5,1X,1H+)                                                LIN02230
        WRITE(6,197)(PNTR(K),K=KK,KKK)                                       LIN02240
        WRITE(6,197)(PNTR1(K) , K=1,101)                                     LIN02250
  197   FORMAT(1H+,T17,101A1)                                                LIN02260
  200   CONTINUE                                                             LIN02270
C                                                                            LIN02280
CCC     PRINT X-AXIS SCALES                                                  LIN02290
CC                                                                           LIN02300
        WRITE(6,220)          (XAXIS(I),I=1,DSIZ)                            LIN02310
  220   FORMAT(1H+,15X,  10(1H+,9(1H-)),1H+/14X,11(I3,7X))                   LIN02320
C                                                                            LIN02330
        TOTPTS = XPT + YPT                                                   LIN02340
        IF (TOTPTS .EQ. 0) RETURN                                            LIN02350
        WRITE(6,225)TOTPTS                                                   LIN02360
                                                                             LIN02370
```

FILE: LINPLT FORTRAN A

```
  225 FORMAT(///' A TOTAL OF ',I4,' POINTS WERE NOT DISPLAYED ON THE LINFLIN02380
     * POINTER GRAPH.'/' THE POINTS WERE OUT OF RANGE IN EITHER THE X DILIN02390
     *RECTION OR Y DIRECTION')                                          LIN02400
      CALL SETMRG(66,4,62)                                             LIN02410
      RETURN                                                           LIN02420
      END                                                              LIN02430
```

FILE: MATTNS FORTRAN A

```
      SUBROUTINE MATTNS(A,B,C,D,L,M)
C
C     MULTIPLY A BY B AND ADD D, STORE IN C
C
      INTEGER B
C     DIMENSION A(L,M),B(M),C(L),D(L)
C     DO 20 I=1,L
      SUM = 0.0
C     DO 10 K=1,M
C  10 SUM = SUM + A(I,K) * B(K)
   20 C(I) = SUM + D(I)
C     RETURN
      END
```

FILE: OFFSET FORTRAN A

```
      SUBROUTINE OFFSET(YSCALE,XSCALE)                                  OFF00010
C                                                                       OFF00020
      INTEGER BMKEY,RSCALE                                              OFF00030
      INTEGER YSIZ,XSIZ,XHI,YHI,XLO                                     OFF00040
      INTEGER CSCALE                                                    OFF00050
C                                                                       OFF00060
CC    INCLUDE COMRK6.LIST                                               OFF00070
CC    INCLUDE CMRK12.LIST                                               OFF00080
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         COM00010
     *              HISFIL,HISKEY,TFORM,ERPTP,ERPKEY,MAPUNT,NOFILE,     COM00020
     *    DRUMAD,DRMDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 COM00030
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                    COM00040
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              COM00050
     *  CRDUNT,PRTUNT,RANDIO                                            COM00060
      COMMON/SCTTER/RSCALE,YSCLE,CLRVEC(30),NCLRCH,CLRKEY,LOG,
     *  FREQ,XMAX,YMAX,XMIN,YMIN,PCKGND,XHI,XLO,YLO,XSIZ,
     *  YHI,YSIZ,NBINS,SYMMTX(32),BMATRX(60),HVEC(30),NHVCHN,NOFEAT
     *   ,SCALKY,MENADR,FLOADR,PNTADR,IDADR,NC,BMFEAT,HMCOMB
     *    ,NOVEC,TOTMNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCH1,DRMTNS,DRMTN1,
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE
     *  ,NOSUH
CSEND                                                                   OFF00100
C                                                                       OFF00110
      DIMENSION YSCALE(YSIZ),XSCALE(XSIZ)                               OFF00120
C                                                                       OFF00130
CC    SCALES ARE COMPUTED IN ONE OF 3 WAYS :                            OFF00140
CC    1) DATA HAS NOT BEEN TRANSFORMED                                  OFF00150
CC       BMKEY = 0                                                      OFF00160
CC    2) DATA HAS BEEN TRANSFORMED AND RESCALED                         OFF00170
CC       RESCALE = 1                                                    OFF00180
CC    3) DATA HAS BEEN TRANSFORMED, BUT NOT RESCALED                    OFF00190
CC       RESCALE = 0                                                    OFF00200
CC       CSCALE = 1 -- XSIZ AND/OR YSIZ HAS BEEN INPUT - MIN AND MAX    OFF00210
CC                     WILL BE USED FOR HI AND LO PARAMETERS           OFF00220
CC                                                                     OFF00230
      IF (BMKEY .EQ. 0) GO TO 80                                        OFF00240
      IF (RSCALE .EQ. 1) GO TO 80                                       OFF00250
      IF (CSCALE .EQ. 1) GO TO 80                                       OFF00260
C                                                                       OFF00270
CC                                                                      OFF00280
      XHIGH = XMAX                                                      OFF00290
      XLOW = XMIN                                                       OFF00300
      YHIGH = YMAX                                                      OFF00310
      YLOW = YMIN                                                       OFF00320
      GO TO 90                                                          OFF00330
C                                                                       OFF00340
CC                                                                      OFF00350
   80 XHIGH = XHI                                                       OFF00360
      XLOW = XLO                                                        OFF00370
      YHIGH = YHI                                                       OFF00380
      YLOW = YLO                                                        OFF00390
C                                                                       OFF00400
   90 CONTINUE                                                          OFF00410
      XINC = (XHIGH - XLOW) / FLOAT(XSIZ-1)                             OFF00420
      YINC = (YHIGH - YLOW) / FLOAT(YSIZ-1)                             OFF00430
      XSCALE(1) = XLOW                                                  OFF00440
      YSCALE(1) = YLOW                                                  OFF00450
      DO 100    I=2,XSIZ                                                OFF00460
  100 XSCALE(I) = XSCALE(I-1) + XINC                                    OFF00470
      DO 110    I=2,YSIZ                                                OFF00480
  110 YSCALE(I) = YSCALE(I-1) + YINC                                    OFF00490
      RETURN                                                            OFF00500
      END                                                               OFF00510
```

FILE: RESCLE FORTRAN A

```
      SUBROUTINE RESCLE(DATA,SWTCH,NVECT)                                RES00010
C                                                                        RES00020
CC*                                                                      RES00030
CC*   RESCLE RESCALES THE TRANSFORMED DATA                              RES00040
CC*        SWTCH = 0  RESCALE THE ENTIRE ARRAY                           RES00050
CC*        SWTCH = 1  RESCALE ONE PIXEL AND RETURN                       RES00060
CC*                                                                      RES00070
C                                                                        RES00080
      INTEGER HMCOMB,SWTCH                                               RES00090
      INTEGER XMI,XLO,YMI,YLO                                            RES00100
      REAL MAX(2),MIN(2),HEAL(2),RANGE(2),DATA(2,NVECT)                  RES00110
C                                                                        RES00120
CC    INCLUDE CMMK12,LIST
      COMMON/SCTTER/RSCALE,XYSCLE,CLRVEC(30),NCLRCM,CLRKEY,LOG,
     *  FMFO,XMAX,YMAX,XMIN,YMIN,RCKGND,XMI,XLO,YLO,XSIZ,
     *  YMI,YSIZ,NPINS,SYMMTX(32),HMATRX(40),RVEC(30),NAVCHN,NOFEAT
     *   ,SCALKY,MENADR,FLDADR,PNTADR,IDADR,NC,HMFEAT,HMCOMB
     *    ,NOVEC,TOTPNS,SIZE,DRMIU,DRMIO1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE
     *  ,NOSUB
C$END
C
      MAX(1) = XMAX
      MAX(2) = YMAX
C
      MIN(1) = XMIN
      MIN(2) = YMIN
      RANGE(1) = XMI - XLO
      RANGE(2) = YMI - YLO
      DO 200 J=1,NVECT
      DO 100  I=1,HMCOMB
C
      HEAL(I) = RANGE(I) / (MAX(I)-MIN(I))
  100 DATA(I,J) = HEAL(I)*ABS(MIN(I) - DATA(I,J))
      IF (SWTCH .EQ. 1) RETURN
C
  200 CONTINUE
      RETURN
      END
```

```
FILE: SCATTR   FORTRAN  A

      SUBROUTINE SCATTR(FIELDS,VERTEX,TNSDAT,MEANS,PLOT,PNTR,IDATA,TOP,     SCA00010
     *    LIMIT,BUFF)                                                       SCA00020
C                                                                           SCA00030
C*    SCATTR SETS UP THE LOGIC FOR CREATING THE SPECTRAL PLOTS              SCA00040
C*                                                                          SCA00050
      IMPLICIT INTEGER (A-Z)                                                SCA00060
      REAL XSCALE,YSCALE,XLOWER,XUPPER,YUPPER,TNSDAT(2,1)                   SCA00070
      REAL YLOWER,LINE,SAMPLE                                               SCA00080
      LOGICAL SWITCH(4)                                                     SCA00090
C                                                                           SCA00100
      DIMENSION BUFF(1)                                                     SCA00110
      DIMENSION XSCALE(200),YSCALE(200)                                     SCA00120
      DIMENSION IRG(4),IEN(4)                                               SCA00130
      DIMENSION FIELDS(4,1),VERTEX(2,1),PLOT(1),MEANS(1)                    SCA00140
     *,PNTR(1),IDATA(ASIZ,NC)                                               SCA00150
      DIMENSION LINADR(4),LINE(4)                                           SCA00160
      DIMENSION COVAR(465)                                                  SCA00170
C                                                                           SCA00180
CCC   INCLUDE COMMK1.LIST                                                   SCA00190
CCC   INCLUDE COMMK6.LIST                                                   SCA00200
C     INCLUDE CMAK12.LIST                                                   SCA00210
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFLD2,              SCA00220
     *              AVAR2,COVAR2,CLS1D2,SUBNO2,SUBDS2,FLDSV2,VERTX2,        SCA00230
     *              FETVC2(30),SUBVC2(75),SUMMTP(75),CLSVC2(60),            SCA00240
     *              KEPPTS(60),NOCHR,GRPNAM(40),GRPDEX(61),                 SCA00250
     *              GRPCHK(61),GROUPS(124)                                  SCA00260
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             SCA00270
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERRKEY,MAPUNT,NOFILE,       SCA00280
     *    DRUMAD,DRMNOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    SCA00290
     *  ,NHSTUN,NHSTFI,SCTHUN,MAPFIL                                        SCA00300
     *  ,DOTUNT,DOTFIL,NCHRAS,TRNSFL,BMTPFL,HISTFL,PCHUNT,                  SCA00310
     *  CRDUNT,PTUNT,RANDIO                                                 SCA00320
      COMMON/SCTTER/RSCALE,XYSCLE,CLRVEC(30),NCLRCH,CLRKEY,LOG,             SCA00330
     *  FREQ,XMAX,YMAX,XMIN,YMIN,HCHGND,XMI,XLO,YLO,XSIZ,                   SCA00340
     *  YMI,YSIZ,NBINS,SYMMTX(32),SMAT-X(60),NBVCHN(30),NBVCHN,NOFEAT       SCA00350
     *  ,SCALRY,MENADR,FLDADR,PNTADR,IDADR,NC,NMFEAT,BMCOMB                 SCA00360
     *  ,NOVEC,TOTBNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,        SCA00370
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD          SCA00380
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE                             SCA00390
     *  ,NOSUM                                                              SCA00400
CSEND                                                                       SCA00410
CCC                                                                         SCA00420
CCC   READ REC 2 FORM N-DIM HIST FILE                                       SCA00430
CC                                                                          SCA00440
      READ(NHSTUN)CLSVC2(1),(SUBVC2(I),I=1,NOSUB),((FIELDS(I,J),           SCA00450
     *            I=1,4),J=1,NOFLD2),((VERTEX(I,J),I=1,2),J=1,TOTVT2)     SCA00460
CCC                                                                         SCA00470
CCC   ARE COLOR CODES COMING FROM STAT FILE                                 SCA00480
CC                                                                          SCA00490
      IF (CLRKEY .NE. 1)  GO TO 90                                          SCA00500
      IF (NOFEAT .NE. 0)  GO TO 80                                          SCA00510
CCC                                                                         SCA00520
CCC   DEFAULT CHANNELS ARE  CHANNELS FROM N-DIM HIST FILE                   SCA00530
CC                                                                          SCA00540
      DO 60  I=1,NOFET2                                                     SCA00550
   60 FETVEC(I) = FETVC2(I)                                                 SCA00560
CCC                                                                         SCA00570
CCC   EXTRACT JUST MEANS FROM STAT FILE                                     SCA00580
CC                                                                          SCA00590
   80 CALL GETST(SAVTAP,STAFIL,MEANS,STDEV,NOSUB2,SUBVEC,NOFEAT,FETVEC,     SCA00600
     *  MEANS,COVAR,0)                                                      SCA00610
CCC                                                                         SCA00620
CCC   WRITE OUT SAVED TRAINING/TEST FIELDS                                  SCA00630
CC                                                                          SCA00640
   90 CALL WRTFLD(FIELDS,VERTEX,NOFLD2,2,CLSVC2,SUBVC2)                     SCA00650
CCC                                                                         SCA00660
CCC   READ N-DIM HIST FILE AND STORE INFO ON DRUM                           SCA00670
CCC                                                                         SCA00680
      CALL STOFIL(LIMIT,MEANS,BUFF)                                         SCA00690
      DRMVC1 = DRMVEC                                                       SCA00700
      NVECT = NVEC                                                          SCA00710
      DO 91 II=1,NOREAD                                                     SCA00720
      IF (II .EQ. NOREAD) NVECT = LREAD                                     SCA00730
      NWORDS = NVECT*SIZE                                                   SCA00740
      CALL WRREAD(DRMVC1,PLOT,NWORDS,STAT)                                  SCA00750
      DRMVC1 = DRMVC1 + NWORDS                                              SCA00760
   91 IF (STAT .EQ. 1) GO TO 91                                             SCA00770
C                                                                           SCA00780
C     APPLY TRANSFORMATION                                                  SCA00790
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SCATTR FORTRAN A

```
C
      IF (BMKEY .NE. 0) CALL TNSFER(PLOT,TNSDAT,NVECT,II)                    SCA00800
C                                                                            SCA00810
CC    NO TRANSFORMATION APPLIED                                              SCA00820
C                                                                            SCA00830
      IF (BMKEY .EQ. 0) CALL UNPCKV(PLOT,TNSDAT,NVECT)                       SCA00840
C                                                                            SCA00850
CC    SORT VECTORS IN DECENDING ORDER                                        SCA00860
C                                                                            SCA00870
      ICOL = 2                                                               SCA00880
      CALL SORTVC(TNSDAT,PNTR,ICOL,NVECT,IBG,IEN,II)                         SCA00890
      DRMWD  = 2*NVECT                                                       SCA00900
      CALL RWRITE(DRMTN1,TNSDAT,DRMWD,ISTAT1)                                SCA00910
   92 IF (ISTAT1 .EQ. 1) GO TO 92                                            SCA00920
      DRMTN1= DRMTN1 + DRMWD                                                 SCA00930
      CALL RWRITE(DRMPT1,PNTR,NVECT,ISTAT2)                                  SCA00940
   94 IF (ISTAT2 .EQ. 1) GO TO 94                                            SCA00950
      DRMPT1 = DRMPT1 + NVECT                                                SCA00960
   93 CONTINUE                                                               SCA00970
      IF (RSCALE .EQ. 0) GO TO 97                                            SCA00980
      IF (SCALKY .NE. 2) GO TO 97                                            SCA00990
      DRMTN1 = DRMTNS                                                        SCA01000
      NTNSVC = (TOP - VECTR1) / 2                                            SCA01010
      IF (NOVEC .LT. NTNSVC) GO TO 95                                        SCA01020
      NREAD = NOVEC/NTNSVC                                                   SCA01030
      IF (MOD(NOVEC,NTNSVC) .NE. 0) NREAD = NREAD + 1                        SCA01040
      LSREAD = (MOD(NOVEC,NTNSVC))                                           SCA01050
      IF (LSREAD .EQ. 0) LSREAD = NTNSVC                                     SCA01060
      GO TO 99                                                               SCA01070
   95 NREAD = 1                                                              SCA01080
      NTNSVC = NOVEC                                                         SCA01090
      LSREAD = NTNSVC                                                        SCA01100
   9A DO 99  II=1,NREAD                                                      SCA01110
      IF (II .EQ. NREAD) NTNSVC = LSREAD                                     SCA01120
      DRMWD = 2 * NTNSVC                                                     SCA01130
      CALL RREAD (DRMTN1,TNSDAT,DRMWD ,ISTAT2)                               SCA01140
  101 IF (ISTAT2 .EQ. 1) GO TO 101                                           SCA01150
      CALL RESCLE(TNSDAT,0,NTNSVC)                                           SCA01160
      CALL RWRITE(DRMTN1,TNSDAT,DRMWD, ,ISTAT3)                              SCA01170
  100 IF (ISTAT3 .EQ. 1) GO TO 100                                           SCA01180
      DRMTN1 = DRMTN1 + DRMWD                                                SCA01190
   99 CONTINUE                                                               SCA01200
C                                                                            SCA01210
CC    COMPUTE TAPE CO/ORDINATES                                              SCA01220
C                                                                            SCA01230
   97 CONTINUE                                                               SCA01240
      CALL OFFSET(YSCALE,XSCALE)                                             SCA01250
C                                                                            SCA01260
CCC                                                                          SCA01270
CC    WRITE TAPE PARAMETERS                                                  SCA01280
C                                                                            SCA01290
      WRITE(6,300)YSIZ,XSIZ                                                  SCA01300
  300 FORMAT(/T51,'SCATTER PLOT TAPE PARAMETERS',//T51,'NO. OF LINES PER     SCA01310
     * FILE  =',I4/T51,'NO. OF SAMPLES PER LINE =',I4)                       SCA01320
      IF (BMKEY .EQ. 0) WRITE(6,310)XLO,YLO,XHI,YHI                          SCA01330
  310 FORMAT(/T51,'XLO =',I4,T71,'YLO =',I4/T51,'XHI =',I4,T71,'YHI =',      SCA01340
     *  I4)                                                                  SCA01350
      IF (BMKEY .NE. 0 .AND. RSCALE .EQ. 1) WRITE(6,310)XLO,YLO,XHI,YHI      SCA01360
      IF (CSCALE .EQ. 1 .AND. BMKEY .GT. 0) WRITE(6,310)XLO,YLO,XHI,YHI      SCA01370
      IF (BMKEY .NE. 0) WRITE(6,320)XMIN,YMIN,XMAX,YMAX                      SCA01380
  320 FORMAT(/T50,'XMIN =',F10.5,T70,'YMIN =',F10.5/T50,'XMAX =',F10.5,      SCA01390
     * T70,'YMAX = ',F10.5)                                                  SCA01400
      FORMAT = 1                                                             SCA01410
      CH = NC + 1                                                            SCA01420
C                                                                            SCA01430
      DO 110 I=1,CH                                                          SCA01440
  110 CLRVEC(I) = I                                                          SCA01450
C                                                                            SCA01460
  120 CALL WRTHED(CH,CLRVEC,XSIZ,FORMAT,SCTRUN)                              SCA01470
C                                                                            SCA01480
      LSTLIN = 0                                                             SCA01490
C                                                                            SCA01500
      IF (LOG .EQ. 1 .OR. FREQ .EQ. 1) CALL LINPLT                           SCA01510
C                                                                            SCA01520
      DO 125 I=1,NOREAD                                                      SCA01530
  125 SWITCH(I) = .TRUE.                                                     SCA01540
C                                                                            SCA01550
      CALL RREAD(DRMPTH,PNTR,NOVEC,ISTAT3)                                   SCA01560
      I=YSIZ+1                                                               SCA01570
                                                                             SCA01580
```

FILE: SCATTR FORTRAN A

```
  130 I=I-1                                                                 SCA01590
      IF (I .EQ. 1 .AND. CLRKEY .EQ. 3) LSTLIN = -1                         SCA01600
      ILINE = I                                                             SCA01610
      YUPPER = YSCALE(I)                                                    SCA01620
      YLOWER = YSCALE(I-1)                                                  SCA01630
C                                                                           SCA01640
      DO 140   K=1,CH                                                       SCA01650
      DO 140   J=1,XSIZ                                                     SCA01660
  140 IDATA(J,K) = BCKGND                                                   SCA01670
C                                                                           SCA01680
  143 IF (ISTAT3 .EQ. 1) GO TO 143                                          SCA01690
C     COLLECT ALL POINTS THAT BELONG TO THIS LINE(I)                        SCA01700
CCC                                                                         SCA01710
CCC                                                                         SCA01720
CCC   THE DATA VECTORS WERE READ IN NVEC AT A TIME. EACH BLOCK OF DATA      SCA01730
CCC   VECTORS HAS ITS OWN POINTER ARRAY FOR SORTING THE DATA VECTORS IN     SCA01740
CCC   DESCENDING ORDER. EACH POINTER ARRAY PNTR(1...NOREAD) MUST BE         SCA01750
CCC   SEARCHED FOR POINTS BELONGING TO LINE(I)                              SCA01760
C                                                                           SCA01770
      DO 180 II=1,NOREAD                                                    SCA01780
      K = (II-1)*NVEC                                                       SCA01790
      IR = IRG(II)                                                          SCA01800
  145 IF (SWITCH(II)) LINADR(II) = (2*NVEC)*(II-1) + 2*IR + DRMTNS - 1      SCA01810
      IF (SWITCH(II)) CALL RREAD(LINADR(II),LINE(II),1,ISTA4)               SCA01820
  146 IF (ISTA4 .EQ. 1) GO TO 146                                           SCA01830
      IF (ILINE .NE. 1) GO TO 1465                                          SCA01840
      IF (LINE(II) .LE. YUPPER) GO TO 147                                   SCA01850
 1465 CONTINUE                                                              SCA01860
      IF (LINE(II) .LE. YUPPER .AND. LINE(II) .GT. YLOWER) GO TO 147        SCA01870
      SWITCH(II) = .FALSE.                                                  SCA01880
      IF (ILINE .NE. YSIZ) GO TO 180                                        SCA01890
      IF (LINE(II) .GT. YUPPER) GO TO 147                                   SCA01900
      GO TO 180                                                             SCA01910
CCC                                                                         SCA01920
CCC   POSITION POINT IN X CO-ORDINATES                                      SCA01930
C                                                                           SCA01940
  147 SAMADR = (2*NVEC)*(II-1) + 2*IR + DRMTNS - 2                          SCA01950
      CALL RREAD(SAMADR,SAMPLE,1,ISTAT5)                                    SCA01960
      SWITCH(II) = .TRUE.                                                   SCA01970
  148 IF (ISTAT5 .EQ. 1) GO TO 148                                          SCA01980
      DO 150 J=1,XSIZ                                                       SCA01990
      IPOSTN = J                                                            SCA02000
      XLOWER = XSCALE(J)                                                    SCA02010
      XUPPER = XSCALE(J+1)                                                  SCA02020
C                                                                           SCA02030
      IF (J .NE. 1) GO TO 149                                               SCA02040
      IF (SAMPLE .LE. XLOWER) GO TO 160                                     SCA02050
  149 CONTINUE                                                              SCA02060
      IF (SAMPLE .GE. XLOWER .AND. SAMPLE .LT. XUPPER) GO TO 160            SCA02070
  150 CONTINUE                                                              SCA02080
C                                                                           SCA02090
CCC   GET COLOR CODES                                                       SCA02100
C                                                                           SCA02110
  160 CALL CLRCOD(IR,MEANS,IDATA,IPOSTN,II)                                 SCA02120
      CALL CNTER(IR,IDATA,IPOSTN,II,COUNTR)                                 SCA02130
      IF (LOG .EQ. 1 .OR. FREQ .EQ. 1) CALL STOPTS(COUNTR,LINE(II),         SCA02140
     *                                             SAMPLE)                  SCA02150
CCC                                                                         SCA02160
CCC   CHECK NEXT VECTOR                                                     SCA02170
C                                                                           SCA02180
      IF(IR .EQ. TEN(II)) GO TO 180                                         SCA02190
      IF (SWITCH(II)) IR = PNTR(IR+K)                                       SCA02200
      IF (SWITCH(II)) IRG(II) = IR                                          SCA02210
      IF(SWITCH(II)) GO TO 145                                              SCA02220
C                                                                           SCA02230
  180 CONTINUE                                                              SCA02240
CCC                                                                         SCA02250
CCC   WRITE A LINE                                                          SCA02260
C                                                                           SCA02270
      CALL WRTLN(IDATA,LSTLIN)                                              SCA02280
  200 IF(I.GT.1) GO TO 130                                                  SCA02290
C                                                                           SCA02300
      IF (CLRKEY .EQ. 1) CALL CLRKYS(XSIZ,IDATA,NOSUB2,CH,MEANS,NC)         SCA02310
      IF (CLRKEY .EQ. 2) CALL CLRKYS(XSIZ,IDATA,NOSUB2,CH,MEANS,NC)         SCA02320
      IF (CLRKEY .EQ. 4) CALL CLRKYS(XSIZ,IDATA,NOFLD2,CH,MEANS,NC)         SCA02330
C                                                                           SCA02340
      IF (LOG .EQ. 1 .OR. FREQ .EQ. 1) CALL PRTPLT(BUFF,BUFF)               SCA02350
C                                                                           SCA02360
                                                                            SCA02370
```

FILE: SCATTR FORTRAN A

```
  400 RETURN                                                        SCA02380
      END                                                           SCA02390
```

```
FILE: SETADR   FORTRAN  A

      SUBROUTINE SETADR(*,*,TOP,BUFF,LIMIT)                                  SET00010
C                                                                            SET00020
C*                                                                           SET00030
C*    SETADR COMPUTES THE ADDRESS FOR STORING THE NDIM FILE ON DRUM          SET00040
C*    AND ADDRESS FOR THE TWO ARRAYS - BUFF(LIMIT) AND ARRAY(TOP)            SET00050
C*                                                                           SET00060
      IMPLICIT INTEGER (A-Z)                                                 SET00070
C                                                                            SET00080
C     INCLUDE COMBK1.LIST                                                    SET00090
C     INCLUDE COMBK6.LIST                                                    SET00100
C     INCLUDE CMBK12.LIST                                                    SET00110
C     INCLUDE COMT12.LIST                                                    SET00120
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARS22,TOTVT2,NOFLD2,               SET00130
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLDSV2,VERTX2,         SET00140
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),             SET00150
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                  SET00160
     *              GRPCHK(61),GROUPS(124)                                   SET00170
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,              SET00180
     *             HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,         SET00190
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                     SET00200
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                         SET00210
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                   SET00220
     *  CRDUNT,PRTUNT,RANDIO                                                 SET00230
      COMMON/SCTTER/PSCALE,XYSCLE,CLRVEC(30),NCLRCH,CLRKEY,LOG,              SET00240
     *  FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,                    SET00250
     *  YHI,YSIZ,NBINS,SYMMTX(32),RMATRX(60),RVEC(30),NRVCHN,NOFEAT          SET00260
     *   ,SCALKY,MENADR,FLDADR,PNTADR,IDADR,NC,BMFEAT,BMCOMB                 SET00270
     *    ,NOVEC,TOTMNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,       SET00280
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD           SET00290
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE                              SET00300
     *  ,NOSUB                                                               SET00310
C*    SCTTER IS A COMMON BLOCK LOADED ONLY WITH THE SCATTER PLOT              SET00320
C*                  PROCESSOR                                                SET00330
C*                                                                           SET00340
C*    PSCALE - KEY INDICATING THAT THE TRANSFORMED DATA IS TO BE RESCALE     SET00350
C*    XYSCLE - KEY INDICATING THAT THE PIXEL FREQUENCY PLOT DATA IS          SET00360
C*             TO BE RESCALED TO A RANGE OF 100                              SET00370
C*    CLRVEC - ARRAY CONTAINING THE COLOR CHANNELS                           SET00380
C*    NCLRCH - NO. OF COLOR CHANNELS                                         SET00390
C*    CLRKEY - KEY INDICATING THE MANNER THAT THE COLORS ARE BEING           SET00400
C*             DEFINED :                                                     SET00410
C*                  = 1 - INPUT BY STAT FILE                                 SET00420
C*                  = 2 - USER INPUT                                         SET00430
C*                  = 3 - RADIANCE VALUES FROM IMAGE TAPE- ON NDIM FILE      SET00440
C*                  = 4 - FIELD MEANS - ON NDIM FILE                         SET00450
C*    LOG    - KEY INDICATING LOG(2) OF FREQUENCY IS TO BE OUTPUT ON         SET00460
C*             LINE PRINTER PLOT                                             SET00470
C*    FREQ   - KEY INDICATING FREQUENCY IS TO BE OUTPUT ON LINE PRINT-       SET00480
C*             ER PLOT                                                       SET00490
C*    XMAX   - MAXIMUM VALUE OF 1ST COMPONENT OF TRANS. DATA                 SET00500
C*    YMAX   - MAXIMUM VALUE OF 2ND COMPONENT OF TRANS. DATA                 SET00510
C*    XMIN   - MINIMUM VALUE OF 1ST COMPONENT OF TRANS. DATA                 SET00520
C*    YMIN   - MINIMUM VALUE OF 2ND COMPONENT OF TRANS. DATA                 SET00530
C*    BCKGND - COLOR FOR TAPE BACKGROUND                                     SET00540
C*    XHI    - UPPER LIMIT OF SCAN LINE PARAMETER                            SET00550
C*    XLO    - LOWER LIMIT OF SCAN LINE PARAMETER                            SET00560
C*    YHI    - UPPER LIMIT OF LINE NO. PARAMETER                             SET00570
C*    YLO    - LOWER LIMIT OF LINE NO. PARAMETER                             SET00580
C*    XSIZ   - NO. OF SAMPLES PER SCAN LINE                                  SET00590
      CONTINUE                                                               SET00600
C*    YSIZ   - NO. OF LINES TO OUTPUT ON TAPE                                SET00610
C*    NBINS  - NO. OF BIN LEVELS OR SYMBOLS FOR PIXEL FREQ. PLOT             SET00620
C*    SYMMTX - ARRAY CONTAINING SYMBOLS FOR PIXEL FREQ. PLOT                 SET00630
C*    RMATRX - ARRAY CONTAINING R-MATRIX                                     SET00640
C*    RVEC   - ARRAY CONTAINING ADDITIVE VECTOR                              SET00650
C*    NRVCHN - NO. OF ADDITIVE VECTOR ELEMENTS                               SET00660
C*    NOFEAT - NOFET2 * NCLRCH                                               SET00670
C*    SCALKY - KEY INDICAING THE MANNER OF COLECTING THE MIN AND MAX         SET00680
C*             VALUES :                                                      SET00690
C*                  = 1 - USER INPUT                                         SET00700
C*                  = 2 - COMPUTE FROM NDIM FILE                             SET00710
C*    MENADR - ADDRESS FOR STORING MEANS                                     SET00720
C*    FLDADR - ADDRESS FOR STORING FIELD INFO                                SET00730
C*    IDADR  - ADDRESS FOR STORING IDS                                       SET00740
C*    NC     - NO. OF CHANNELS FOR COLORS TO BE OUTPUT ON TAPE               SET00750
C*    BMFEAT - NO. OF CHANNELS IN B-MATRIX                                   SET00760
C*    BMCOMB - NO. OF LINEAR COMB.                                           SET00770
C*    NOVEC  - NO. OF VECTORS ON NDIM FILE                                   SET00780
C*    TOTMNS - NO. OF MEAN ELEMENTS                                          SET00790
```

FILE: SETADR FORTRAN A

```
C*      SIZE    - NO. OF WORDS FOR PACKED HISTOGRAMMED VECTOR                  SET00800
C*      DRMID   - BEGINNING DRUM ADDRESS FOR STORING IDS                       SET00810
C*      DRMID1  - SUMMING   DRUM ADDRESS FOR  RETRIV. IDS                      SET00820
C*      DRMCLR  - BEGINNING DRUM ADDRESS FOR  STORING COLORS                   SET00830
C*      DRMCR1  - SUMMING   DRUM ADDRESS FOR  RETRIV. COLORS                   SET00840
C*      DRMTNS  - BEGINNIN  DRUM ADDRESS FOR  STORING TRANS. DATA              SET00850
        CONTINUE                                                               SET00860
C*      DRMTN1  - SUMMING   DRUM ADDRESS FOR  RETRIV. TRANS. DATA              SET00870
C*      DRMCNT  - BEGINNING DRUM ADDRESS FOR  STORING FREQ.                    SET00880
C*      DRMCT1  - SUMMING   DRUM ADDRESS FOR  RETRIV. FREQ.                    SET00890
C*      DRMVEC  - BEGINNING DRUM ADDRESS FOR  STORING VECTORS                  SET00900
C*      DRMVC1  - SUMMING   DRUM ADDRESS FOR  RETRIV. VECTORS                  SET00910
C*      VECTR1  - ADDRESS IN 'ARRAY' FOR STORING TRANSF. VECTOR                SET00920
C*      DATA1   - SAME ADDRESS AS VECTR1 - USED FOR CREATING A SCAN LINE       SET00930
C*                OF DATA OUTPUT TO THE SCATTER PLOT TAPE                      SET00940
C*      NVEC    - NO. OF VECTORS TO READ FROM DRUM AT ONE TIME                 SET00950
C*      NOREAD  - NO. OF READS TO DRUM                                         SET00960
C*      LREAD   - NO. OF VECTOR TO READ ON LAST DRUM READ                      SET00970
C*      DRMPTR  - BEGINNING DRUM ADDRESS FOR STORING POINTERS                  SET00980
C*      FETVEC  - ARRAY FOR CONTAINING FETVC2 AND CLRVEC CHANNELS              SET00990
C*      DRMPLT  - BEGINNING ADDRESS FOR STORING FREQ. PLOT IMAGE               SET01000
C*      CSCALE  - KEY INDICATING THAT XSIZ OR YSIZ PARAMETERS HAVE BEEN        SET01010
C*                INPUT                                                        SET01020
C*                                                                           SET01030
C$END                                                                        SET01040
C                                                                            SET01050
      DIMENSION BUFF(1)                                                      SET01060
C                                                                            SET01070
C                                                                            SET01080
      READ(NHSTUN,END=150) NOFLD2,NOSUB,TOTVT2,NOVEC                         SET01090
      IF (CLRKEY .EQ. 1) GO TO 100                                           SET01100
      IF (CLRKEY .EQ. 2) GO TO 100                                           SET01110
      IF (CLRKEY .EQ. 3) GO TO 120                                           SET01120
      IF (CLRKEY .EQ. 4) GO TO 130                                           SET01130
C                                                                            SET01140
  100 MENADR = 1                                                             SET01150
      FLDADR = MENADR + 60*NC                                                SET01160
      VERTX2 = FLDADR + 4*NOFLD2                                             SET01170
      DATA1 = FLDADR                                                         SET01180
      GO TO 133                                                              SET01190
C                                                                            SET01200
  120 FLDADR = 1                                                             SET01210
      VERTX2 = FLDADR + NOFLD2*4                                             SET01220
      DATA1 = FLDADR                                                         SET01230
      GO TO 133                                                              SET01240
C                                                                            SET01250
  130 MENADR = 1                                                             SET01260
      FLDADR = MENADR + TOTMNS                                               SET01270
      VERTX2 = FLDADR + 4*NOFLD2                                             SET01280
      DATA1 = FLDADR                                                         SET01290
C                                                                            SET01300
C                                                                            SET01310
C     COMPUTE MAXIMUM NO. OF VECTORS ARRAY MAY HOLD AT ONE TIME              SET01320
C                                                                            SET01330
  133 VECTR1 = DATA1                                                         SET01340
      NVEC = (TOP - DATA1) / 2                                               SET01350
      VECTR2 = LIMIT / SIZE                                                  SET01360
      IF (VECTR2 .LT. NVEC) NVEC = VECTR2                                    SET01370
      IF (NOVEC .LT. NVEC)  GO TO 135                                        SET01380
      NOREAD = NOVEC / NVEC                                                  SET01390
      IF (MOD(NOVEC,NVEC) .NE. 0) NOREAD = NOREAD + 1                        SET01400
      LREAD = MOD(NOVEC,NVEC)                                                SET01410
      IF (LREAD .EQ. 0) LREAD = NVEC                                         SET01420
      GO TO 140                                                              SET01430
  135 NOREAD = 1                                                             SET01440
      NVEC = NOVEC                                                           SET01450
      LREAD = NVEC                                                           SET01460
C                                                                            SET01470
C     ADDRESSES FOR HIGH SPEED DRUM                                          SET01480
C                                                                            SET01490
  140 DRMVEC = DRUMAD                                                        SET01500
      DRMID = DRMVEC + NOVEC*SIZE                                            SET01510
      DRMCNT = DRMID + NOVEC                                                 SET01520
      IF (CLRKEY .EQ. 3) DRMCLR = DRMCNT + NOVEC                             SET01530
      IF (CLRKEY .NE. 3) DRMCLR = DRMCNT                                     SET01540
      DRMTNS = DRMCLR + NOVEC                                                SET01550
      DRMPTR = DRMTNS + NOVEC*2                                              SET01560
      DRMPLT = DRMPTR + NOVEC                                                SET01570
      TOTDRM = DRMPLT                                                        SET01580
```

FILE: SETADR FORTRAN A

```
      IF (LOG .EQ. 1 .OR. FREQ .EQ. 1) TOTDRM = DRMPLT + 10201          SET01590
      IF ((TOTDRM-DRUMAD) .LE. DRMWDS) GO TO 143                        SET01600
      WRITE(6,142)TOTDRM,DRMWDS                                         SET01610
  142 FORMAT(/' NOT ENOUGH DRUM SPACE.'/' TOTAL WORDS OF DRUM SPACE =', SET01620
     * I12/' TOTAL WORDS OF DRUM SPACE AVAILABLE =',I12)                SET01630
      CALL CMERR                                                        SET01640
  143 CONTINUE                                                          SET01650
      DRMVC1 = DRMVEC                                                   SET01660
      DRMID1 = DRMID                                                    SET01670
      DRMCR1 = DRMCLR                                                   SET01680
      DRMCT1 = DRMCNT                                                   SET01690
      DRMTN1 = DRMTRS                                                   SET01700
      DRMPT1 = DRMPTR                                                   SET01710
C                                                                       SET01720
CC    ZERO OUT AREA OF DRUM                                             SET01730
C                                                                       SET01740
      IF (LOG .EQ. 0 .AND. FREQ .EQ. 0) GO TO 145                       SET01750
      DO 147  I=1,10201                                                 SET01760
  147 BUFF(I) = 0                                                       SET01770
      CALL RWRITE(DRMPLT,BUFF,10201,ISTAT)                              SET01780
  146 IF (ISTAT .EQ. 1) GO TO 146                                       SET01790
  145 RETURN 1                                                          SET01800
C                                                                       SET01810
  150 REWIND NMSTUN                                                     SET01620
      RETURN 2                                                          SET01830
      END                                                               SET01840
```

FILE: SET11 FORTRAN A

```
      SUBROUTINE SET11(MEANS,MENS,BUFF)                                 SET00010
                                                                        SET00020
      IMPLICIT INTEGER (A-Z)                                            SET00030
      REAL BMATRX,RVEC,MENS(1),YMAX,XMAX,XMIN,NUM,YMIN                  SET00040
C                                                                       SET00050
      DIMENSION BUFF(1),SYMTX(32)                                       SET00060
      DIMENSION EQUCOM(3),CODE(18),CARD(62),MEANS(1)                    SET00070
      DIMENSION RMVEC(16),ACARD(20)                                     SET00080
C                                                                       SET00090
      DATA NPUT/1H/                                                     SET00100
      DATA EQUCOM/2,'=',',','/                                          SET00110
      DATA CODE/'CHAN','STAT','HISF','PIXP','COLR','SIZE',             SET00120
     * 'SYMB','MODU','DATE','COMM','HED1','HED2','PLOT',               SET00130
     *  'B-MA','BCKG','RVEC','SCAL','*END'/                            SET00140
      DATA SYMTX/',',',','/','C','O','O','D','U','U','D','O',          SET00150
     *           ',',',','/','C','O','O',',',',','C',',','G',',','S',  SET00160
     *           ',',',',',',',',',',',',',',',',',',',',',',',',',',  SET00170
     *           ',',',',' '/                                           SET00180
      DATA BLANK/' '/, XBCD/'X'/, YBCD/'Y'/, LBCD/'L'/, SBCD/'S'/,     SET00190
     U HBCD/'H'/, KOMMA/','/, BBCD/'B'/, WBCD/'W'/,                     SET00200
     * FBCD/'F'/,CBCD/'C'/,RBCD/'R'/,UBCD/'U'/,IBCD/'I'/,ABCD/'A'/      SET00210
                                                                        SET00220
C     INCLUDE CMBK12.LIST                                               SET00230
C     INCLUDE COMBK1.LIST                                               SET00240
C     INCLUDE COMBK6.LIST                                               SET00250
C     INCLUDE CUMBK4.LIST                                               SET00260
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          SET00270
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    SET00280
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        SET00290
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             SET00300
     *              GRPCHK(6),GROUPS(124)                               SET00310
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15).                   SET00320
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                 SET00330
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               SET00340
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,BMKEY,         SET00350
     *              HISFIL,HISKEY,TRFORM,ERPTP,ERPKEY,MAPUNT,NOFILE,    SET00360
     *     DRUMAD,DRMLOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               SET00370
     *   ,NHSTUN,NHSTFI,SCTUN,MAPFIL                                    SET00380
     *   ,DOTUNT,DOTFIL,NCHPAS,TPNSFL,BMTRFL,HISTFL,PCHUNT,             SET00390
     *   CRUNT,PRTUNT,HANDIO                                            SET00400
      COMMON/SCTTER/RSCALE,XYSCLE,CLRVEC(30),NCLPCH,CLRKEY,LOG,         SET00410
     *   FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,              SET00420
     *   YHI,YSIZ,NRINS,SYMMTX(32),BMATRX(60),RVEC(30),NRVCHN,NOFEAT    SET00430
     *    ,SCALKY,MENADR,FLDADR,PNTADR,IDADR,NC,BMFEAT,RMCOMB           SET00440
     *     ,NOVEC,TOTNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,  SET00450
     *   DRMCNT,DRMC11,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD     SET00460
     *   ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE                        SET00470
     *   ,NOSUB                                                         SET00480
C$END                                                                   SET00490
C                                                                       SET00500
      CSCALE = 0                                                        SET00510
C                                                                       SET00520
C     DEFAULT VALUES                                                    SET00530
C                                                                       SET00540
C*                                                                      SET00550
C**   INITIALIZE PARAMETERS                                             SET00560
C*                                                                      SET00570
      DO 10  I=1,32                                                     SET00580
   10 SYMMTX(I)=SYMTX(I)                                                SET00590
      FREQ = 0                                                          SET00600
      LOG = 0                                                           SET00610
      NOFEAT = 0                                                        SET00620
      NCLSTR = 0                                                        SET00630
      NRINS = 10                                                        SET00640
      BMKEY = 0                                                         SET00650
      BCKGND = 255                                                      SET00660
      XSIZ = 101                                                        SET00670
      YSIZ = 101                                                        SET00680
      RSCALE = 0                                                        SET00690
      XYSCLE = 0                                                        SET00700
      YLO = 0                                                           SET00710
      SCALKY = 2                                                        SET00720
      YHI = 100                                                         SET00730
      XHI = 100                                                         SET00740
      XLO = 0                                                           SET00750
      WRITE(6,100)                                                      SET00760
  100 FORMAT(//' INPUT SUMMARY'//)                                      SET00770
C                                                                       SET00780
C   SETUP REREAD BUFFER                                                 SET00790
```

FILE: SET11 FORTRAN A

```
C                                                                       SET00800
      CALL REREAD(30,50)                                                SET00810
C     NOW READ CARD INTO BUFFER                                         SET00820
  105 READ(21,106)(ACARD(I),I=1,20)                                     SET00830
  106 FORMAT(20A4)                                                      SET00840
      WRITE(30,106)(ACARD(I),I=1,20)                                    SET00850
      REWIND 30                                                         SET00860
      READ(30,110)CODE1,CARD                                            SET00870
  110 FORMAT(A4,6X,62A1)                                                SET00880
      REWIND 30                                                         SET00890
      COL = 0                                                           SET00900
C                                                                       SET00910
      WRITE(6,120)CODE1,CARD                                            SET00920
  120 FORMAT(1X,A4,6X,62A1)                                             SET00930
C                                                                       SET00940
      DO  130  I=1,NPUT                                                 SET00950
      IF (CODE1 .EQ. CODE(I)) GO TO (150,180,210,250,290,300,340,       SET00960
     *  350,370,390,400,410,415,420,455,460,470,477),I                 SET00970
  130 CONTINUE                                                          SET00980
      WRITE(6,140)                                                      SET00990
  140 FORMAT(' INVALID CONTROL CARD -- IGNORED')                        SET01000
      GO TO 105                                                         SET01010
C                                                                       SET01020
CC    CHANNEL CARD -- NEEDED ONLY IF STATS FILE IS INPUT                SET01030
C                                                                       SET01040
  150 M=NXTCHR(CARD,COL)                                                SET01050
      IF (M .NE. BLANK) GO TO 160                                       SET01060
  153 WRITE(6,155)                                                      SET01070
  155 FORMAT(' ERROR ON CHANNELS CARD')                                 SET01080
      GO TO 105                                                         SET01090
  160 COL = COL - 1                                                     SET01100
      NOFEAT = NUMBER(CARD,COL,FETVEC,NOFEAT)                           SET01110
      CALL ORDER(FETVEC,NOFEAT)                                         SET01120
      NC = NOFEAT                                                       SET01130
      CLRKEY = 1                                                        SET01140
      GO TO 105                                                         SET01150
C                                                                       SET01160
CC    STAT FILE CARD                                                    SET01170
C                                                                       SET01180
  180 M = NXTCHR(CARD,COL)                                              SET01190
      IF (M .EQ. BLANK) GO TO 105                                       SET01200
      IF (M .EQ. URCD) GO TO 190                                        SET01210
      IF (M .EQ. FRCD) GO TO 200                                        SET01220
  185 WRITE(6,187)                                                      SET01230
  187 FORMAT(' ERROR ON STAT FILE CARD')                                SET01240
      GO TO 105                                                         SET01250
  190 J = FIND12(CARD,COL,EQUCOM)                                       SET01260
      IF (J .NE. 2) GO TO185                                            SET01270
      M = NUMBER(CARD,COL,SAVTAP,ZERO)                                  SET01280
      COL = COL - 1                                                     SET01290
      CLRKEY = 1                                                        SET01300
      GO TO 180                                                         SET01310
  200 J = FIND12(CARD,COL,EQUCOM)                                       SET01320
      IF ( J .NE. 2 ) GO TO 185                                         SET01330
      M = NUMBER(CARD,COL,STAFIL,ZERO)                                  SET01340
      COL = COL - 1                                                     SET01350
      GO TO 180                                                         SET01360
C                                                                       SET01370
CC    N-DIM HISTOGRAM FILE                                              SET01380
C                                                                       SET01390
  210 M = NXTCHR(CARD,COL)                                              SET01400
      IF (M .EQ. BLANK) GO TO 105                                       SET01410
      IF (M .EQ. URCD) GO TO 230                                        SET01420
      IF (M .EQ. FRCD) GO TO 240                                        SET01430
  220 WRITE(6,225)                                                      SET01440
  225 FORMAT(' ERROR ON N-DIM HISTOGRAM FILE CARD')                     SET01450
      GO TO 105                                                         SET01460
  230 J = FIND12(CARD,COL,EQUCOM)                                       SET01470
      IF (J .NE. 2) GO TO 220                                           SET01480
      M = NUMBER(CARD,COL,NHSTUN,ZERO)                                  SET01490
      COL = COL - 1                                                     SET01500
      GO TO 210                                                         SET01510
  240 J = FIND12(CARD,COL,EQUCOM)                                       SET01520
      IF (J .NE. 2) GO TO 220                                           SET01530
      M = NUMBER(CARD,COL,NHSTFI,ZERO)                                  SET01540
      COL = COL - 1                                                     SET01550
      GO TO 210                                                         SET01560
C                                                                       SET01570
C     PIXEL FREQ. PLOT CARD                                             SET01580
```

```
FILE: SET11    FORTRAN  A

C                                                                       SET01590
  250 M = NXTCHR(CARD,COL)                                              SET01600
      IF (M .EQ. BLANK) GO TO 105                                       SET01610
      IF (M .EQ. LRCD) GO TO 270                                        SET01620
      IF (M .EQ. FRCD) GO TO 280                                        SET01630
      IF (M .EQ. RRCD) GO TO 285                                        SET01640
  255 WRITE(6,260)                                                      SET01650
  260 FORMAT(' ERROR ON OPTION CARD')                                   SET01660
      GO TO 105                                                         SET01670
  270 LOG = 1                                                           SET01680
      GO TO 287                                                         SET01690
  280 FREQ = 1                                                          SET01700
      GO TO 287                                                         SET01710
  285 XYSCLE = 1                                                        SET01720
  287 J = FIND12(CARD,COL,EQUCOM)                                       SET01730
      IF (J .EQ. 3) GO TO 250                                           SET01740
      GO TO 105                                                         SET01750
C                                                                       SET01760
CC    COLOR CODES                                                       SET01770
C                                                                       SET01780
  290 NCLSTR = VECSCN(MEANS(ADR+1),NOCHAN,CARD,COL) + NCLSTR            SET01790
      ADR = NCLSTR*NOCHAN                                               SET01800
      NC = NOCHAN                                                       SET01810
      NOSUB2 = NCLSTR                                                   SET01820
      CLRKEY = 2                                                        SET01830
      GO TO 105                                                         SET01840
C                                                                       SET01850
CC    TAPE SIZE CARD                                                    SET01860
C                                                                       SET01870
  300 M = NXTCHR(CARD,COL)                                              SET01880
      IF (M .EQ. BLANK) GO TO 105                                       SET01890
      IF (M .EQ. XRCD) GO TO 320                                        SET01900
      IF (M .EQ. YRCD) GO TO 330                                        SET01910
  310 WRITE(315)                                                        SET01920
  315 FORMAT(' ERROR ON TAPE SIZE CARD')                                SET01930
      GO TO 105                                                         SET01940
  320 M = NXTCHR(CARD,COL)                                              SET01950
      J = FIND12(CARD,COL,EQUCOM)                                       SET01960
      IF (M .EQ. SRCD) CSCALE = 1                                       SET01970
      IF (J .NE. 2) GO TO 310                                           SET01980
      MM = NUMBER(CARD,COL,J,ZERO)                                      SET01990
      IF (M .EQ. RRCD) XHI = J                                          SET02000
      IF (M .EQ. LRCD) XLO = J                                          SET02010
      IF (M .EQ. SRCD) XSIZ = J                                         SET02020
      COL = COL - 1                                                     SET02030
      GO TO 300                                                         SET02040
  330 M = NXTCHR(CARD,COL)                                              SET02050
      IF (M .NE. SRCD) SCSALE = 1                                       SET02060
      J = FIND12(CARD,COL,EQUCOM)                                       SET02070
      IF (J .NE. 2) GO TO 310                                           SET02080
      MM = NUMBER(CARD,COL,J,ZERO)                                      SET02090
      IF (M .EQ. RRCD) YHI = J                                          SET02100
      IF (M .EQ. LRCD) YLO = J                                          SET02110
      IF (M .EQ. SRCD) YSIZ = J                                         SET02120
      COL = COL - 1                                                     SET02130
      GO TO 300                                                         SET02140
C                                                                       SET02150
CC    SYMBOLS CARD                                                      SET02160
C                                                                       SET02170
  340 M = NXTCHR(CARD,COL)                                              SET02180
      IF (M .EQ. BLANK) GO TO 105                                       SET02190
      IF (M .EQ. KOMMA)  GO TO 340                                      SET02200
      ICNT = ICNT + 1                                                   SET02210
      SYMMTX(ICNT) = M                                                  SET02220
      GO TO 340                                                         SET02230
C                                                                       SET02240
CC    MODULE STAT CARD DECK                                             SET02250
C                                                                       SET02260
  350 CALL CROSTA(BUFF,TOP)                                             SET02270
      GO TO 105                                                         SET02280
C                                                                       SET02290
CC    DATE CARD                                                         SET02300
C                                                                       SET02310
  370 M = NXTCHR(CARD,COL)                                              SET02320
      IF ( M .EQ. BLANK) GO TO 105                                      SET02330
      READ(30,380)DATE                                                  SET02340
  380 FORMAT(10X,15A4)                                                  SET02350
      REWIND 30                                                         SET02360
      GO TO 105                                                         SET02370
```

```
FILE: SET11    FORTRAN  A

C       COMMENT CARD                                                    SET02380
  390 M = NXTCHR(CARD,COL)                                              SET02390
      IF (M .EQ. BLANK) GO TO 105                                       SET02400
      READ(30,380)COMENT                                                SET02410
      REWIND 30                                                         SET02420
      GO TO 105                                                         SET02430
C                                                                       SET02440
CCC     HED1                                                            SET02450
C                                                                       SET02460
  400 M = NXTCHR(CARD,COL)                                              SET02470
      IF (M .EQ. BLANK) GO TO 105                                       SET02480
      READ(30,380) HED1                                                 SET02490
      REWIND 30                                                         SET02500
      GO TO 105                                                         SET02510
C                                                                       SET02520
CCC     HED2                                                            SET02530
C                                                                       SET02540
  410 M = NXTCHR(CARD,COL)                                              SET02550
      IF (M .EQ. BLANK) GO TO 105                                       SET02560
      READ(30,380) HED2                                                 SET02570
      REWIND 30                                                         SET02580
      GO TO 105                                                         SET02590
C                                                                       SET02600
CCC     SCATTER PLOT TAPE CARD                                          SET02610
C                                                                       SET02620
  415 M = NXTCHR(CARD,COL)                                              SET02630
      IF (M .EQ. BLANK) GO TO 105                                       SET02640
      IF (M .EQ. URCD) GO TO 418                                        SET02650
  416 WRITE(6,417)                                                      SET02660
  417 FORMAT(' ERROR ON SCATTER PLOT TAPE CARD')                        SET02670
      GO TO 105                                                         SET02680
  418 J = FIND12(CARD,COL,EQUCOM)                                       SET02690
      IF (J .NE. 2) GO TO 416                                           SET02700
      M = NUMBER(CARD,COL,SCTRUN,ZERO)                                  SET02710
      GO TO 105                                                         SET02720
C                                                                       SET02730
CCCC    B - MATRIX CARD                                                 SET02740
CCCC    BMKEY = 1 FOR CARDS                                             SET02750
CCCC    BMKEY = 2 FOR FILE                                              SET02760
CCC                                                                     SET02770
  420 M = NXTCHR(CARD,COL)                                              SET02780
      IF (M .EQ. BLANK) GO TO 433                                       SET02790
      IF (M .EQ. CRCD) GO TO 440                                        SET02800
      IF (M .EQ. FRCD) GO TO 450                                        SET02810
  433 WRITE(6,435)                                                      SET02820
  435 FORMAT(' ERROR ON B-MATRIX CARD')                                 SET02830
      GO TO 105                                                         SET02840
  440 BMKEY = 1                                                         SET02850
      CALL BMFIL(BMATRX,BMCOMB,BMFEAT,BMVEC,BMKEY)                      SET02860
      GO TO 105                                                         SET02870
  450 BMKEY = 2                                                         SET02880
      CALL BMFIL(BMATRX,BMCOMB,BMFEAT,BMVEC,BMKEY)                      SET02890
      GO TO 105                                                         SET02900
C                                                                       SET02910
CCC     BACK GROUND COLOR CARD                                          SET02920
C                                                                       SET02930
  455 M = NXTCHR(CARD,COL)                                              SET02940
      IF (M .EQ. BLANK) GO TO 105                                       SET02950
      IF (M .EQ. BRCD) BCKGND = 0                                       SET02960
      IF (M .EQ. WRCD) BCKGND = 255                                     SET02970
      GO TO 105                                                         SET02980
C                                                                       SET02990
CCC     B VECTOR                                                        SET03000
C                                                                       SET03010
  460 NBVCHN = FLTNUM(CARD,COL,BVEC(NBVCHN+1),30) + NBVCHN              SET03020
      GO TO 105                                                         SET03030
C                                                                       SET03040
CCC     SCALING CARD                                                    SET03050
C                                                                       SET03060
  470 M = NXTCHR(CARD,COL)                                              SET03070
      IF (M .EQ. BLANK) GO TO 105                                       SET03080
      IF (M .EQ. FSCD) GO TO 471                                        SET03090
      IF (M .EQ. YSCD) GO TO 472                                        SET03100
      IF (M .EQ. XSCD) GO TO 473                                        SET03110
      IF (M .EQ. RSCD) GO TO 476                                        SET03120
      GO TO 474                                                         SET03130
  471 SCALXY = 2                                                        SET03140
      GO TO 105                                                         SET03150
 4711 J = FIND12(CARD,COL,EQUCOM)                                       SET03160
```

ORIGINAL PAGE IS
OF POOR QUALITY


```
      IF (J .EQ. 3) GO TO 470                                          SET03170
      GO TO 105                                                        SET03180
  472 M = NXTCHR(CARD,COL)                                             SET03190
      M = NXTCHR(CARD,COL)                                             SET03200
      J = FIND12(CARD,COL,EQUCOM)                                      SET03210
      IF (J .NE. 2) GO TO 474                                          SET03220
      J = FLTNUM(CARD,COL,NUM,1)                                       SET03230
      IF (M .EQ. IRCD) YMIN = NUM                                      SET03240
      IF (M .EQ. ABCD) YMAX = NUM                                      SET03250
      SCALKY = 1                                                       SET03260
      GO TO 470                                                        SET03270
  473 M = NXTCHR(CARD,COL)                                             SET03280
      M = NXTCHR(CARD,COL)                                             SET03290
      J = FIND12(CARD,COL,EQUCOM)                                      SET03300
      IF (J .NE. 2) GO TO 474                                          SET03310
      J = FLTNUM(CARD,COL,NUM,1)                                       SET03320
      IF (M .EQ. IRCD) XMIN = NUM                                      SET03330
      IF (M .EQ. ABCD) XMAX = NUM                                      SET03340
      SCALKY = 1                                                       SET03350
      GO TO 470                                                        SET03360
  474 WRITE(6,475)                                                     SET03370
  475 FORMAT(' ERROR ON SCALING CARD')                                 SET03380
      GO TO 105                                                        SET03390
  476 RSCALE = 1                                                       SET03400
      GO TO 4711                                                       SET03410
C                                                                      SET03420
CCC   *END*                                                            SET03430
CCC                                                                    SET03440
  477 CONTINUE                                                         SET03450
C                                                                      SET03460
  480 IF (BMKEY .LE. 0 ) GO TO 490                                     SET03470
      IF (LOG .EQ. 1) GO TO 481                                        SET03480
      IF (FREQ .EQ. 0) GO TO 482                                       SET03490
  481 IF (RSCALE .EQ. 1) GO TO 482                                     SET03500
      WRITE(6,840)                                                     SET03510
  840 FORMAT(' DATA MUST BE RESCALED BEFORE PIXEL FREQUENCY PLOT OPTION SET03520
     *MAY BE SELECTED')                                                SET03530
      LOG = 0                                                          SET03540
      FREQ = 0                                                         SET03550
  482 CONTINUE                                                         SET03560
      CALL WRTBM(PMATRX,BMCOMB,BMFEAT,BMVEC)                           SET03570
      GO TO 490                                                        SET03580
  483 WRITE(6,485)NOFET2,BMFEAT                                        SET03590
  485 FORMAT(' NO. OF PLOTTING CHANNELS,  NO. OF B-MATRIX CHANNELS'    SET03600
     *                              /' MUST BE EQUAL. CHANNELS ARE ',I5, SET03610
     *' RESPECTIVELY')                                                 SET03620
      CALL CMERR                                                       SET03630
C                                                                      SET03640
  490 CONTINUE                                                         SET03650
CCC                                                                    SET03660
CCC   CHANGE MEANS TO FLOATING PT NO                                   SET03670
C                                                                      SET03680
      IF (NCLSTR .EQ. 0)  GO TO 497                                    SET03690
      DO 495 I=1,ADR                                                   SET03700
  495 MENS(I) = MEANS(I)                                               SET03710
  497 CONTINUE                                                         SET03720
CCC                                                                    SET03730
CCC   COMPUTE NO. OF BINS                                              SET03740
CCC                                                                    SET03750
      IF (ICNT .GT. 0) NBINS = ICNT/2                                  SET03760
CCC                                                                    SET03770
CCC   READ HEADER REC FROM N-DIM HIST FILE                             SET03780
CCC                                                                    SET03790
      READ(NHSTUN) TOTMNS,SIZE,NOFET2,(FETVC2(I),I=1,NOFET2),          SET03800
     *             NCLRCH,(CLRVEC(I),I=1,NCLRCH)                       SET03810
CCC                                                                    SET03820
CCC                                                                    SET03830
CCC   DEFAULT STAT CHANNELS                                            SET03840
CCC                                                                    SET03850
      IF (CLRKEY .NE. 1) GO TO 499                                     SET03860
      IF (NOFEAT .NE. 0) GO TO 499                                     SET03870
      DO 498 I=1,NOFET2                                                SET03880
      IF (I .GT. 4) GO TO 499                                          SET03890
      NOFEAT = I                                                       SET03900
      NC = I                                                           SET03910
  498 FETVEC(I) = FETVC2(I)                                            SET03920
  499 CONTINUE                                                         SET03930
      IF (BMKEY .NE. 0 .AND. NOFET2 .NE. BMFEAT)  GO TO 483            SET03940
C                                                                      SET03950
```

```
FILE: SET11     FORTRAN  A

C       ARE COLOR CODES COMING FROM TAPE                                        SET03960
C                                                                               SET03970
C                                                                               SET03980
        IF (NCLRCH .EQ. 0) GO TO 505                                            SET03990
        CLRKEY = 3                                                              SET04000
        NC = NCLRCH                                                             SET04010
  505 IF ( TOTMNS .EQ. 0) GO TO 500                                             SET04020
        CLRKEY = 4                                                              SET04030
        NC = NOFET2                                                             SET04040
  500 CONTINUE                                                                  SET04050
C                                                                               SET04060
C       POSITION TO DESIRED FILE                                                SET04070
C                                                                               SET04080
        CALL FSRSFL(NMSTUN,NMSTFI,ISTAT)                                        SET04090
        IF (ISTAT .EQ. 0) GO TO 520                                             SET04100
        WRITE(6,510)NMSTFI,ISTAT                                                SET04110
  510 FORMAT(' ERROR IN POSITIONING N-DIM HIST FILE TO FILE',                   SET04120
     *I5,' ISTAT = ',I5)                                                        SET04130
        REWIND NMSTUN                                                           SET04140
        CALL CMERR                                                              SET04150
  520 CONTINUE                                                                  SET04160
C                                                                               SET04170
        WRITE(6,700)                                                            SET04180
        IF (FREQ .EQ. 1) WRITE(6,710)                                           SET04190
        IF (LOG  .EQ. 1) WRITE(6,720)                                           SET04200
        IF (BMKEY .GT. 0)  WRITE(6,740)                                         SET04210
        IF (PCKGND .EQ. 0) WRITE(6,750)                                         SET04220
        IF (BCKGND .EQ. 255)WRITE(6,760)                                        SET04230
        IF (RSCALE .EQ. 0 .AND. BMKEY .GT. 0) WRITE(6,770)                      SET04240
        IF (SCALKY .EQ. 2 .AND. BMKEY .GT. 0) WRITE(6,780)                      SET04250
        IF (SCALKY .EQ. 1) WRITE(6,790)                                         SET04260
        IF (CLRKEY .EQ. 1) WRITE(6,800)(FETVEC(I),I=1,NOFEAT)                   SET04270
        IF (CLRKEY .EQ. 2) WRITE(6,810)                                         SET04280
        IF (CLRKEY .EQ. 3) WRITE(6,820)                                         SET04290
        IF (CLRKEY .EQ. 4) WRITE(6,830)                                         SET04300
  700 FORMAT(//' USER HAS SELECTED THE FOLLOWING OPTIONS :',//)                 SET04310
  710 FORMAT(' LINE PRINTER PIXEL FREQ. PLOT')                                  SET04320
  720 FORMAT(' LINE PRINTER LOG OF PIXEL FREQ. PLOT')                           SET04330
  740 FORMAT(' TRANSFORM DATA')                                                 SET04340
  750 FORMAT(' BACK GROUND COLOR FOR SPECTRAL PLOTS WILL BE BLACK')             SET04350
  760 FORMAT(' BACK GROUND COLOR FOR SPECTRAL PLOTS WILL BE WHITE')             SET04360
  770 FORMAT(' TRANSFORMED DATA WILL NOT BE RESCALED')                          SET04370
  780 FORMAT(' USER HAS INPUT RESCALING FACTORS')                               SET04380
  790 FORMAT(' RESCALING FACTORS WILL BE COMPUTED FROM N-DIM HIST FILE')        SET04390
  800 FORMAT(' COLOR CODES ARE COMING FROM STAT FILE USING CHANNELS ',          SET04400
     *   4(1X,I2))                                                            SET04410
  810 FORMAT(' COLOR CODES ARE COMING FROM USER INPUT')                         SET04420
  820 FORMAT(' COLOR CODES WILL BE COMPUTED FROM FIELD MEANS')                  SET04430
  830 FORMAT(' COLOR CODES ARE COMING FROM RADIANCE VALUES STORED ON N-         SET04440
     *DIM HIST FILE')                                                           SET04450
        RETURN                                                                  SET04460
C                                                                               SET04470
        END
```

FILE: SORTVC FORTRAN A

```
      SUBROUTINE SORTVC(HIST,PNTR,ICOL,NOVEC,IBG,IEN,II)
C
C     SORTVC SORTS THE ARRAY HIST INTO DESCENDING ORDER
C
C     HIST  - THE ARRAY TO BE ORDERED
C     PNTR  - ARRAY CONTAINING POINTERS
C     ICOL  - THE COLUMN WITHIN HIST THAT IS TO BE ORDERED
C     NOVEC - NO. OF VECTORS TO SORT
C     IBG   - ARRAY CONTAINING BEGINNING POINTER FOR EACH BLOCK SORTED
C     IEN   - ARRAY CONTAINING ENDING    POINTER FOR EACH BLOCK SORTED
C     II    - NO. OF BLOCK BEING SORTED
C
      IMPLICIT INTEGER (A-Z)
C
      REAL HIST(ICOL,NOVEC),NUMBR
C
      DIMENSION PNTR(1)
      DIMENSION IBG(1),IEN(1)
C
      IB = 1
      IE = 1
      NEXT = IB
      DO 200 J=1,NOVEC
C
      NUMBR = HIST(ICOL,J)
C     COMPARE AGAINST LARGEST NUMBER
C
      IF (NUMBR .GE. HIST(ICOL,IB)) GO TO 100
C
      COMPARE AGAINST SMALLEST NUMBER
C
      IF (NUMBR .LE. HIST(ICOL,IE)) GO TO 120
C
      COMPARE AGAINST NEXT LARGEST NUMBER
      INTRY = PNTR(NEXT)
      NOVEC1 = J - 1
      DO 80   I=1,NOVEC1
      PAST = NEXT
      NEXT = PNTR(NEXT)
      IF (NUMBR .GE. HIST(ICOL,NEXT)) GO TO 130
      IF (NUMBR .GT. HIST(ICOL,INTRY)) GO TO 130
      PAST = INTRY
      INTRY = PNTR(INTRY)
   80 CONTINUE
      WRITE(6,90)
   90 FORMAT(' ERROR IN SORTING VECTORS')
      STOP
C
C     LARGEST NUMBER FOUND THUS FAR
C
  100 PNTR(J) = IB
      IB = J
      NEXT = IB
      GO TO 200
C
C     SMALLEST NUMBER FOUND THUS FAR
C
  120 PNTR(IE) = J
      IE = J
      GO TO 200
C
C
  130 PNTR(J) = NEXT
      NEXT = IB
      PNTR(PAST) = J
  200 CONTINUE
C
      IBG(II) = IB
      IEN(II) = IE
      RETURN
C
      END
```

```
FILE: STOFIL   FORTRAN  A

      SUBROUTINE STOFIL(LIMIT,MEANS,BUFF)                                    ST000010
C*                                                                           ST000020
C*    STOFIL READS AND STORES NDIM FILE ON DRUM                              ST000030
C*                                                                           ST000040
      IMPLICIT INTEGER (A-Z)                                                 ST000050
C                                                                            ST000060
CCC   INCLUDE CMRK12.LIST                                                    ST000070
CCC   INCLUDE COMMRA.LIST                                                    ST000080
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,AMFILE,AMKEY,              COM00010
     *              HISFIL,HISKEY,TRFORM,LPTPTR,ERPKEY,MAPUNT,NOFILE,        COM00020
     *     DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    COM00030
     *  ,NMSTUN,NMSTF1,SCTNUN,MAPFIL                                         COM00040
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,AMTHFL,HISTFL,PCHUNT,                   COM00050
     *  CRDUNT,PRTUNT,HANDIO                                                 COM00060
      COMMON/SCTTER/RSCALE,XYSCLE,CLRVEC(30),NCLRCH,CLRKEY,LOG,
     *  FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,
     *  YHI,YSIZ,N3INS,SYMMTX(32),HMATRX(60),RVEC(30),NRVCHN,NOFEAT
     *   ,SCALKY,MEMADR,FLDADR,PNTADR,IDADR,NC,AMFEAT,BMCOMR
     *   ,NOVEC,TOTMNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE
     *   ,NOSUM
CSEND                                                                        ST000100
C                                                                            ST000110
      DIMENSION BUFF(1),ISTAT(4,2),ISTA(3,2),MEANS(1)                        ST000120
      DIMENSION REMDR(4)                                                     ST000130
C                                                                            ST000140
CCC                                                                          ST000150
CCC   READ MEANS INTO CORE                                                   ST000160
      IF (CLRKEY .EQ. 4) READ(NMSTUN)(MEANS(I),I=1,TOTMNS)                   ST000170
      AMT = 0                                                                ST000180
      TOTAL = NOVEC*SIZE                                                     ST000190
      DO 100 I=1,4                                                           ST000200
      REMD = TOTAL - LIMIT*I                                                 ST000210
      IF (REMD .GT. 0) GO TO 90                                              ST000220
      REMDR(I) = TOTAL - LIMIT*(I-1)                                         ST000230
      NTPRDS = I                                                             ST000240
      GO TO 105                                                              ST000250
   90 REMDR(I) = LIMIT                                                       ST000260
  100 CONTINUE                                                               ST000270
  105 CONTINUE                                                               ST000280
C                                                                            ST000290
CCC   READ IN REC 4 -- DATA VECTORS                                          ST000300
CCC                                                                          ST000310
      DO 130    I=1,NTPRDS                                                   ST000320
      NOWRDS = REMDR(I)                                                      ST000330
      READ (NMSTUN)(BUFF(J),J=1,NOWRDS)                                      ST000340
C                                                                            ST000350
CCC   DUMP ON HIGH SPEED DRUM                                                ST000360
CCC                                                                          ST000370
      CALL RWRITE(DRMVC1,BUFF(1),NOWRDS,ISTAT(I,1))                          ST000380
      DRMVC1 = DRMVC1 + NOWRDS                                               ST000390
  130 CONTINUE                                                               ST000400
      IF (CLRKEY .EQ. 3) NOREC = 3                                           ST000410
      IF (CLRKEY .NE. 3) NOREC = 2                                           ST000420
C                                                                            ST000430
CCC   READ IN REC 5 -- ID ARRAY                                              ST000440
CCC   READ IN REC 6 -- COUNTERS                                              ST000450
CCC   READ IN REC 7 -- COLOR CODES                                           ST000460
C                                                                            ST000470
      DO 200 K=1,NOREC                                                       ST000480
      NOWRDS = NOVEC                                                         ST000490
      READ(NMSTUN) (BUFF(J),J=1,NOWRDS)                                      ST000500
C                                                                            ST000510
CCC   DUMP ON HIGH SPEED DRUM                                                ST000520
CCC                                                                          ST000530
      IF (K .EQ. 1) GO TO 170                                                ST000540
      IF (K .EQ. 2) GO TO 180                                                ST000550
      IF (K .EQ. 3) GO TO 190                                                ST000560
  170 CALL RWRITE(DRMID1,BUFF(1),NOWRDS,ISTA(K,1))                          ST000570
      DRMID1 = DRMID1 + NOWRDS                                               ST000580
      GO TO 200                                                              ST000590
  180 CALL RWRITE(DRMCT1,BUFF(1),NOWRDS,ISTA(K,1))                          ST000600
      DRMCT1 = DRMCT1 + NOWRDS                                               ST000610
      GO TO 200                                                              ST000620
  190 CALL RWRITE(DRMCR1,BUFF(1),NOWRDS,ISTA(K,1))                          ST000630
      DRMCR1 = DRMCR1 + NOWRDS                                               ST000640
C                                                                            ST000650
  200 CONTINUE                                                               ST000660
```

FILE: STOFIL FORTRAN A

```
      CALL FSBSFL(NHSTUN,1,ISTAT1)                                  ST000670
      RETURN                                                        ST000680
      END                                                           ST000690
```

FILE: TNSFER FORTRAN A

```
      SUBROUTINE TNSFER(PLOT,TNSDAT,NVECT,IAAA)                          TNS00010
C                                                                        TNS00020
C*                                                                       TNS00030
C*    TNSFER TRANSFORMS 1 - 16 CHANNELS TO 2 COMPONENTS                  TNS00040
C*                                                                       TNS00050
      IMPLICIT INTEGER (A-Z)                                             TNS00060
      REAL XMAX,YMAX,XMIN,YMIN                                           TNS00070
      REAL TNSDAT                                                        TNS00080
C                                                                        TNS00090
CC    INCLUDE COMRK1,LIST                                                TNS00100
CC    INCLUDE COMRK6,LIST                                                TNS00110
C     INCLUDE CMRK12,LIST                                                TNS00120
      COMMON/INFORM/NUCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           TNS00130
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     TNS00140
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         TNS00150
     *              KEEPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              TNS00160
     *              GRPCHK(61),GROUPS(124)                               TNS00170
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          TNS00180
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,    TNS00190
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 TNS00200
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                     TNS00210
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,               TNS00220
     *  CRDUNT,PRTUNT,RANDIO                                             TNS00230
      COMMON/SCTTER/RSCALE,XYSCLE,CLRVEC(30),NCLRCH,CLRKEY,LOG,          TNS00240
     *  FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,                TNS00250
     *  YHI,YSIZ,NBINS,SYMMTX(32),BMATRX(60),BVEC(30),NBVCHN,NOFEAT      TNS00260
     *   ,SCALKY,MFNADR,FLDADR,PNTADR,IDADR,NC,BMFEAT,BMCOMB             TNS00270
     *    ,NOVEC,TOTBNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,   TNS00280
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD       TNS00290
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE                          TNS00300
     *  ,NOSUB                                                           TNS00310
CSEND                                                                    TNS00320
      DIMENSION DATA(16),PLOT(SIZE,NVECT),TNSDAT(2,NVECT)                TNS00330
C                                                                        TNS00340
C                                                                        TNS00350
      LOGICAL*1 LDUM(4),LLDUM(4)                                         TNS00360
      EQUIVALENCE (IDUM,LDUM(1)),(IIDUM,LLDUM(1))                        TNS00370
      DO 100 I=1,NVECT                                                   TNS00380
      DO 10 II=1,NOFET2                                                  TNS00390
      III=(II-1)/4+1                                                     TNS00400
      IBYTE=II-((III-1)/4)*4                                             TNS00410
      IDUM=PLOT(III,I)                                                   TNS00420
      IIDUM=0                                                            TNS00430
      LLDUM(4)=LDUM(IBYTE)                                               TNS00440
   10 DATA(II)=IIDUM                                                     TNS00450
C                                                                        TNS00460
CC    TRANSFORM DATA VECTOR                                              TNS00470
C                                                                        TNS00480
      CALL MATTNS(BMATRX,DATA,TNSDAT(1,I),BVEC,BMCOMB,BMFEAT)            TNS00490
C                                                                        TNS00500
CC    RSCALKY = 1   USER HAS INPUT SCALE FACTORS                         TNS00510
CC    R SCALKY = 2  COMPUTE SCALE FACTORS                                TNS00520
C                                                                        TNS00530
C                                                                        TNS00540
CC    USER HAS INPUT SCALE FACTORS                                       TNS00550
C                                                                        TNS00560
      IF (SCALKY .EQ. 2) GO TO 20                                        TNS00570
      SWTCH = 1                                                          TNS00580
      IF (RSCALE .EQ. 1) CALL RESCLE(TNSDAT(1,I),SWTCH,NVECT)            TNS00590
      GO TO 100                                                          TNS00600
C                                                                        TNS00610
CC    YMAX, XMAX, AND XMIN ARE TO BE TAKEN FROM HIST FILE                TNS00620
C                                                                        TNS00630
   20 IF ( I .NE. 1) GO TO 25                                            TNS00640
      IF ( II .NE. 1) GO TO 25                                           TNS00650
      XMIN = TNSDAT(1,I)                                                 TNS00660
      YMIN = TNSDAT(2,I)                                                 TNS00670
   25 IF (XMAX .LT. TNSDAT(1,I)) XMAX = TNSDAT(1,I)                      TNS00680
      IF (XMIN .GT. TNSDAT(1,I)) XMIN = TNSDAT(1,I)                      TNS00690
      IF (YMAX .LT. TNSDAT(2,I)) YMAX = TNSDAT(2,I)                      TNS00700
      IF (YMIN .GT. TNSDAT(2,I)) YMIN = TNSDAT(2,I)                      TNS00710
  100 CONTINUE                                                           TNS00720
C                                                                        TNS00730
      RETURN                                                             TNS00740
      END                                                                TNS00750
```

FILE: UNPCKV FORTRAN A

```
      SUBROUTINE UNPCKV(PLOT,TNSDAT,NVECT)                                  UNP00010
C                                                                           UNP00020
C*                                                                          UNP00030
C*    UNPCKV UNPACKS TWO 9 BITS BYTES AND STOES THE VALUES INTO TWO         UNP00040
C*    FLOATING POINT WORDS                                                  UNP00050
C*                                                                          UNP00060
      IMPLICIT INTEGER (A-Z)                                                UNP00070
      REAL TNSDAT(2,NVECT)                                                  UNP00080
C                                                                           UNP00090
C                                                                           UNP00100
C     INCLUDE CMHK12.LIST                                                   UNP00110
      COMMON/SCTTER/RSCALE,XYSCLF,CLRVEC(30),NCLRCH,CLRKEY,LOG,             UNP00120
     *  FREQ,XMAX,YMAX,XMIN,YMIN,BCKGND,XHI,XLO,YLO,XSIZ,                   UNP00130
     *  YHI,YSIZ,NBINS,SYMMTX(32),RMATRX(60),RVEC(30),NRVCHN,NOFEAT         UNP00140
     *   ,SCALKY,MENADR,FLDADR,PNTADR,IDADR,NC,RMFEAT,RMCOMB                UNP00150
     *    ,NOVEC,TOTVNS,SIZE,DRMID,DRMID1,DRMCLR,DRMCR1,DRMTNS,DRMTN1,      UNP00160
     *  DRMCNT,DRMCT1,DRMVEC,DRMVC1,VECTR1,DATA1,NVEC,NOREAD,LREAD          UNP00170
     *  ,DRMPTR,DRMPT1,FETVEC(16),DRMPLT,CSCALE                             UNP00180
     *   ,NOSUB                                                             UNP00190
CSEND                                                                       UNP00200
C                                                                           UNP00210
      DIMENSION PLOT(1)                                                     UNP00220
      LOGICAL*1 LDUM(4),LLDUM(4)                                            UNP00230
      EQUIVALENCE (LDUM(1),IDUM),(LLDUM(1),IIDUM)                           UNP00240
C                                                                           UNP00250
      DO 100 I=1,NVECT                                                      UNP00260
      IDUM=PLOT(I)                                                          UNP00270
      IIDUM=0                                                               UNP00280
      LLDUM(4)=LDUM(2)                                                      UNP00290
      VALUE2=IIDUM                                                          UNP00300
      LLDUM(4)=LDUM(1)                                                      UNP00310
      VALUE1=IIDUM                                                          UNP00320
      TNSDAT(2,I) = VALUE2                                                  UNP00330
  100 TNSDAT(1,I) = VALUE1                                                  UNP00340
      RETURN                                                                UNP00350
      END                                                                   UNP00360
```

FILE: VECSCN FORTRAN A

```
      FUNCTION VECSCN(VECTR,NVCELT,CARD,COL)                                VEC00010
C*                                                                          VEC00020
C*    VECSCN CONVERTS ALPHA CHARACTES TO INTEGERS                           VEC00030
C*    RETURNS THE NO. OF ELEMENTS WITHIN A SET OF PARENTHESIS               VEC00040
C*    RETURNS THE NO. OF PARENTHESIS                                        VEC00050
C*                                                                          VEC00060
      IMPLICIT INTEGER (A-Z)                                                VEC00070
C                                                                           VEC00080
      DIMENSION VECTR(1),COMMA(2),CARD(1)                                   VEC00090
C                                                                           VEC00100
      DATA STAR /'*'/,BLANK/' '/,LPRCD/'('/,RPRCD/')'/                      VEC00110
      DATA COMMA/1,','/,KOMMA/','/                                          VEC00120
C                                                                           VEC00130
      NTIMES = 0                                                            VEC00140
      TOTNUM = 0                                                            VEC00150
C                                                                           VEC00160
   80 TOTNUM = TOTNUM + 1                                                   VEC00170
      NVCELT = NVCELT + 1                                                   VEC00180
  100 M = NXTCHR(CARD,COL)                                                  VEC00190
C                                                                           VEC00200
      IF (M .EQ. BLANK) GO TO 140                                           VEC00210
      IF (M .EQ. LPRCD)  GO TO 100                                          VEC00220
      IF (M .EQ. RPRCD)  GO TO 130                                          VEC00230
      IF (M .EQ. KOMMA) GO TO 120                                           VEC00240
      IF (M .EQ. STAR)   GO TO 135                                          VEC00250
C                                                                           VEC00260
C     CHANGING NUMBER FROM ALPHA MODE TO INTEGER MODE                       VEC00270
C                                                                           VEC00280
  110 CALL I4A1HN(CARD(COL),1,NUM)                                          VEC00290
      NUMB = 10*NUMB + NUM                                                  VEC00300
      IF (NUM .LT. 0 .OR. NUM .GT. 9) GO TO 150                             VEC00310
      GO TO 100                                                             VEC00320
C                                                                           VEC00330
C     FOUND A COMMA                                                         VEC00340
C                                                                           VEC00350
  120 VECTR(TOTNUM) = NUMB                                                  VEC00360
      NUMB = 0                                                              VEC00370
      GO TO 80                                                              VEC00380
C                                                                           VEC00390
C     FOUND A ')'                                                           VEC00400
C                                                                           VEC00410
  130 VECTR(TOTNUM) = NUMB                                                  VEC00420
      NUMB = 0                                                              VEC00430
      IF (NTIMES .EQ. 0) GO TO 133                                          VEC00440
      DO 132 I=1,NTIMES                                                     VEC00450
      DO 132 J=1,NVCELT                                                     VEC00460
      JJ = TOTNUM + (I-1)*NVCELT + J                                        VEC00470
  132 VECTR(JJ) = VECTR(TOTNUM-NVCELT+J)                                    VEC00480
      TOTNUM = TOTNUM + NTIMES*NVCELT                                       VEC00490
      NTIMES = 0                                                            VEC00500
C                                                                           VEC00510
C*                                                                          VEC00520
  133 J = FIND12(CARD,COL,COMMA)                                            VEC00530
      IF (J .EQ. -1) GO TO 140                                              VEC00540
      NVCELT = 0                                                            VEC00550
      GO TO 80                                                              VEC00560
C                                                                           VEC00570
C     FOUND A MULTIPLICATIVE FACTOR                                         VEC00580
C                                                                           VEC00590
  135 NTIMES = NUMB - 1                                                     VEC00600
      NUMB = 0                                                              VEC00610
      GO TO 100                                                             VEC00620
C                                                                           VEC00630
C     FINISHED SCANNING CARD                                                VEC00640
C                                                                           VEC00650
  140 VECSCN = TOTNUM/NVCELT                                                VEC00660
      RETURN                                                                VEC00670
C                                                                           VEC00680
  150 VECSCN = -1                                                           VEC00690
      WRITE(6,200)                                                          VEC00700
  200 FORMAT(' ERROR OCCURRED SCANNING VECTOR CARD' )                       VEC00710
      RETURN                                                                VEC00720
      END                                                                   VEC00730
```

17. DOTDATA PROCESSOR

FILE: DOTDAT FORTRAN A

```
C
C     DOTDAT IS THE DRIVER ROUTINE FOR THE DOTDATA PROCESSOR
C
      SUBROUTINE DOTDAT(ARRAY,TOP)
C
      DIMENSION ARRAY(1)
      CALL SFT13
      CALL DOTS(ARRAY(1),ARRAY(5001),ARRAY(6001),TOP)
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FILE: DOTS     FORTRAN  A

C                                                                       DOT00010
C      DOTS IS THE CO-ORDINATOR FOR CREATING THE DOT DATA FILE          DOT00020
C                                                                       DOT00030
       SUBROUTINE DOTS(DATA,FIELDS,VERTEX,TOP)                          DOT00040
       IMPLICIT INTEGER (A-Z)                                           DOT00050
       DIMENSION DATA(SIZE,1),IDATA(10000)                              DOT00060
       DIMENSION FIELDS(4,1),          FL(12),VERTEX(1)                 DOT00070
C      INCLUDE COMBK1,LIST                                              DOT00080
C      INCLUDE COMBK6,LIST                                              DOT00090
C      INCLUDE CMBK14,LIST                                              DOT00100
C      INCLUDE CMBK16,LIST                                              DOT00110
C                                                                       DOT00120
       COMMON/INFORM/NOCLS2,NOSUR2,NOFET2,VARSZ2,TOTVT2,NOFLD2,         DOT00130
      *              AVAR2,COVAR2,CLSID2,SURNO2,SURDS2,FLDSV2,VERTX2,   DOT00140
      *              FETVC2(30),SURVC2(75),SURPTR(75),CLSVC2(60),       DOT00150
      *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),            DOT00160
      *              GRPCHK(61),GROUPS(124)                             DOT00170
       COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,        DOT00180
      *             HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,  DOT00190
      *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL              DOT00200
      *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                  DOT00210
      *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,            DOT00220
      *  CRDUNT,PRTUNT,RANDIO                                          DOT00230
      ,COMMON /DOTVEC/TYPE,CATNAM(60),NOCAT,TOTVEC,FLDINF(6),PRTKEY    DOT00240
      *              ,SIZE ,LACIE                                      DOT00250
       COMMON/ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP     DOT00260
C$END                                                                   DOT00270
C                                                                       DOT00280
       DATA BLANK/' '/                                                  DOT00290
       NOSUN = 8                                                        DOT00300
       ISUNT = 1                                                        DOT00310
       ISUNC = 0                                                        DOT00320
       STAMNT = 1                                                       DOT00330
       SWTCH = 0                                                        DOT00340
       IPT = 1                                                          DOT00350
       IF(LACIE.NE.0)IPT=0                                              DOT00360
C                                                                       DOT00370
C***    CODE ADDED NOV 21, 1978 TO SUPPORT LIST PROCESSING              DOT00380
C                                                                       DOT00390
            INIT = 0                                                    DOT00400
          ZERO = 0                                                      DOT00410
            SWCHG = 0                                                   DOT00420
       NOCAT = 0                                                        DOT00430
       NOFLD2=0                                                         DOT00440
       TYPE = 1                                                         DOT00450
C                                                                       DOT00460
C      INITIALIZE  IMAGE DATA TAPE                                      DOT00470
C                                                                       DOT00480
  90   CALL TAPHDR(DATAPE,DATFIL)                                       DOT00490
C                                                                       DOT00500
       DO 80 I=1,SIZE                                                   DOT00510
       DO 80 J=1,TOTVEC                                                 DOT00520
   80  DATA(I,J) = 0                                                    DOT00530
       TOTVT2 = 0                                                       DOT00540
       TOTVEC = 0                                                       DOT00550
  85    IF(LACIE.EQ.1)CALL FLDLAC(FIELDS,STAMNT,&100,&510,&520,IPT,     DOT00560
      *VERTEX)                                                          DOT00570
C                                                                       DOT00580
C*** ADDED NOV 21, 1978 IN SUPPORT OF LIST PROCESSING                   DOT00590
C                                                                       DOT00600
            IF (LACIE.GT.1) CALL LISTLC(FIELDS,STAMNT,&100,&510,        DOT00610
      *     &520,SWCHG,INIT,LACIE,ZERO,IPT,VERTEX)                     DOT00620
       CALL FLDTYP(FIELDS,STAMNT,&100,&510,&520,IPT,VERTEX)             DOT00630
C                                                                       DOT00640
 100   LINSTR = FLDINF(1)                                               DOT00650
       LINEND = FLDINF(2)                                               DOT00660
       LININC = FLDINF(3)                                               DOT00670
       SAMSTR = FLDINF(4)                                               DOT00680
       SAMEND = FLDINF(5)                                               DOT00690
       SAMINC = FLDINF(6)                                               DOT00700
       FIELDS(2,NOFLD2) = NOCAT                                         DOT00710
       TOTVT2 = FIELDS(4,NOFLD2) + TOTVT2                               DOT00720
C                                                                       DOT00730
       ILINE = (LINEND-LINSTR)/LININC + 1                               DOT00740
       NSAMP = (SAMEND-SAMSTR)/SAMINC + 1                               DOT00750
C                                                                       DOT00760
C      POSITION IMAGE TAPE FOR THIS FIELD                               DOT00770
C                                                                       DOT00780
       CALL FLDINT(FLDINF(1),FETVC2,NOFET2)                             DOT00790
```

FILE: DOTS FORTRAN A

```
C                                                                       DOT00800
C     READ A SCAN  LINE OF DATA, AND PROCESS IT                         DOT00810
C                                                                       DOT00820
      DO 600 LINE=LINSTR,LINEND,LININC                                  DOT00830
      NLINE = NLINE + 1                                                 DOT00840
C                                                                       DOT00850
      CALL LINEND(IDATA,ENDTAP)                                         DOT00860
      IF (ENDTAP .EQ. -1) GO TO 600                                     DOT00870
C                                                                       DOT00880
C     FIND INTERSECTIONS FOR N-R FIELDS                                 DOT00890
C                                                                       DOT00900
      CALL FDLINT(VERTEX(IPT),FIELDS(4,NOFLD2),FL,LINE, SAMP,NI)       DOT00910
C                                                                       DOT00920
      DO 400 J=1,NI,2                                                   DOT00930
      IB = (FL(J)-SAMSTR)/SAMINC + 1                                    DOT00940
      IE = (FL(J+1)- SAMSTR)/SAMINC + 1                                 DOT00950
      IF (MOD(SAMSTR,SAMINC) .NE. MOD(FL(J),SAMINC)) IB = IB + 1        DOT00960
      IF (IB .GT. IE) GO TO 400                                         DOT00970
C                                                                       DOT00980
CC                                                                      DOT00990
C     COLLECTING INFO FOR DATA REC FOR DOTFIL                           DOT01000
C                                                                       DOT01010
      DO 350 K=IB,IE                                                    DOT01020
C                                                                       DOT01030
      TOTVEC = TOTVEC + 1                                               DOT01040
      IF(TOTVEC.LE.250) GO TO 110                                       DOT01050
      TOTVEC=250                                                        DOT01060
C                                                                       DOT01070
      WRITE(6,108)                                                      DOT01080
  108 FORMAT( ////)                                                     DOT01090
      WRITE(6,109)                                                      DOT01100
  109 FORMAT(25X,'**** NOTE - TOTVEC WAS GREATER THAN 250, THEREFORE TOT DOT01110
     1VEC WAS SET TO 250 ****')                                         DOT01120
      GO TO 600                                                         DOT01130
  110 CONTINUE                                                          DOT01140
      DATA(1,TOTVEC) = SAMSTR + SAMINC * (K-1)                          DOT01150
      DATA(2,TOTVEC) = LINE                                             DOT01160
      DATA(3,TOTVEC) = TYPE                                             DOT01170
      DATA(4,TOTVEC) = NOCAT                                            DOT01180
      DO 150 I=1,NOFET2                                                 DOT01190
      KK = (I-1)*NSAMP + K                                              DOT01200
  150 DATA(4+I,TOTVEC) = IDATA(KK)                                      DOT01210
  350 CONTINUE                                                          DOT01220
C                                                                       DOT01230
  400 CONTINUE                                                          DOT01240
C                                                                       DOT01250
  500 CONTINUE                                                          DOT01260
  600 CONTINUE                                                          DOT01270
      GO TO 85                                                          DOT01280
C                                                                       DOT01290
C     WRITE DOT DATA FILE                                               DOT01300
C                                                                       DOT01310
  510 CALL WRTFLD(FIELDS,VERTEX,NOFLD2,2,CATNAM,DUMMY)                  DOT01320
      CALL WRTDOT(TOTVEC,NOSUN,FIELDS,VERTEX,SUNANG,DATA,NOCAT,         DOT01330
     *                  CATNAM,SIZE,NOFET2,FETVC2,TOTVT2,NOFLD2,      DOT01340
     *   DOTUNT,DOTFIL)                                                DOT01350
      DOTFIL = DOTFIL + 1                                               DOT01360
         NOCAT = 0                                                      DOT01370
      SWTCH = 1                                                         DOT01380
      IF(LACIE.NE.0)NOFLD2=0                                            DOT01390
      IF(PRTKEY.EQ.1)GO TO 530                                          DOT01400
         GO TO 90                                                       DOT01410
C                                                                       DOT01420
C     $END CARD FOUND                                                   DOT01430
C                                                                       DOT01440
  520 CALL WRTFLD(FIELDS,VERTEX,NOFLD2,2,CATNAM,DUMMY)                  DOT01450
      CALL WRTDOT(TOTVEC,NOSUN,FIELDS,VERTEX,SUNANG,DATA,NOCAT,         DOT01460
     *       CATNAM,SIZE,NOFET2,FETVC2,TOTVT2,NOFLD2 ,                  DOT01470
     *   DOTUNT,DOTFIL)                                                DOT01480
      SWTCH = 0                                                         DOT01490
      IF(PRTKEY.EQ.1)GO TO 530                                          DOT01500
C                                                                       DOT01510
CC                                                                      DOT01520
C     ROUTINE TO PRINT DOT DATA RECORD                                  DOT01530
CCC                                                                     DOT01540
C                                                                       DOT01550
  530 CONTINUE                                                          DOT01560
  700 FORMAT(//)                                                        DOT01570
  690 FORMAT(1X,' NO.',2X,'SAMPLE',2X,'LINE',2X,'TYPE',2X,'CATEGORY',   DOT01580
```

FILE: DOTS FORTRAN A

```
     1     30X,'DATA'/)                                                    DOT01590
           ISTART=1                                                        DOT01600
           IEND=10                                                         DOT01610
   799     CONTINUE                                                        DOT01620
           IKT=0                                                           DOT01630
           DO 800 II=1,TOTVEC                                              DOT01640
           IKT=IKT+1                                                       DOT01650
           IF(IEND.GT.NOFET2) IEND=NOFET2                                  DOT01660
           IF(II.NE.1.AND.IKT.EQ.1)WRITE(6,810)                            DOT01670
   810     FORMAT(1H1,5(/))                                                DOT01680
           IF(IKT.NE.1) GO TO 820                                          DOT01690
           WRITE(6,700)                                                    DOT01700
           WRITE(6,690)                                                    DOT01710
           WRITE(6,720)(BLANK,FETVC2(I),I=ISTART,IEND)                     DOT01720
   720     FORMAT(37X,10(A1,'CH(',I2,')'))                                 DOT01730
   820     CONTINUE                                                        DOT01740
           WRITE(6,710)II,(DATA(I,II),I=1,4),(DATA(4+JJ,II),JJ=ISTART,IEND) DOT01750
   710     FORMAT(1X,I3,1H.,3X,I4,3X,I4,2X,I2,6X,I2,8X,10(I3,4X))          DOT01760
           WRITE(6,712)                                                    DOT01770
   712     FORMAT( )                                                       DOT01780
   800     CONTINUE                                                        DOT01790
           IF(NOFET2.GT.10) GO TO 830                                      DOT01800
           GO TO 840                                                       DOT01810
   830     CONTINUE                                                        DOT01820
           IF(ITWO.EQ.1) GO TO 840                                         DOT01830
           ITWO=1                                                          DOT01840
           ISTART=IEND+ISTART                                              DOT01850
           IEND=NOFET2                                                     DOT01860
           GO TO 799                                                       DOT01870
   840     CONTINUE                                                        DOT01880
           IF (SWTCH .EQ. 1) GO TO 90                                      DOT01890
           RETURN                                                          DOT01900
           END                                                             DOT01910
```

FILE: FLDLAC FORTRAN A

```
C       FIELDS - CATEGORY NAME AND DOT TYPE FOR DOT I STORED IN              FLD00010
CC            FIELD(1,I) AND FIELD(4,I)                                      FLD00020
CC      STAMNT - INITIALLY SET TO 1. SWITCHED TO INDICATE DOTS BEING         FLD00030
CC             TAKEN FROM CURRENTLY READ CARD.                               FLD00040
CC      IPT - INITIALLY SET TO 1,INDEX NUMBER FOR FIELD VERTEX INFORMATION   FLD00050
CC      VERTEX - VERTEX INFORMATION FOR EACH DOT.                            FLD00060
C                                                                            FLD00070
        SUBROUTINE FLDLAC(FIELDS,STAMNT,*,*,*,IPT,VERTEX)                    FLD00080
        IMPLICIT INTEGER (A-Z)                                               FLD00090
        REAL DUM                                                             FLD00100
        DIMENSION FIELDS(4,1),VERTEX(1),CARD(75),NDOTS(30)                   FLD00110
        DIMENSION ACARD(80)                                                  FLD00120
        LOGICAL SWITCH                                                       FLD00130
        DATA SWITCH/.TRUE./,SWCHG/0/,ENDBCD/'SEN'/,                          FLD00140
       *CATNM1/' '/                                                          FLD00150
C       INCLUDE CMRK14                                                       FLD00160
CC      INCLUDE COMRK1                                                       FLD00170
        COMMON/INFORM/NOCLS2,NOSUR2,NOFET2,VARSZ2,TOTVT2,NOFLD2,             FLD00180
       *              AVAR2,COVAR2,CLSID2,SURNO2,SURDS2,FLDSV2,VERTX2,       FLD00190
       *              FETVC2(30),SUBVC2(75),SURPTR(75),CLSVC2(60),           FLD00200
       *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                FLD00210
       *              GRPCHK(61),GROUPS(124)                                 FLD00220
        COMMON /DOTVEC/TYPE,CATNAM(60),NOCAT,TOTVEC,FLDINF(6),PRTKEY         FLD00230
       *              ,SIZE ,LACIE                                           FLD00240
CSEND                                                                        FLD00250
        IF(STAMNT.EQ.2)GO TO 30                                              FLD00260
        IF(.NOT.SWITCH)GO TO 20                                              FLD00270
        CALL REREAD(30,80)                                                   FLD00280
 10     READ(21,103)(ACARD(I),I=1,80)                                        FLD00290
 103    FORMAT(80A1)                                                         FLD00300
        WRITE(30,103)(ACARD(I),I=1,80)                                       FLD00310
        REWIND 30                                                            FLD00320
        READ(30,1000)ID,TYPES,CARD                                           FLD00330
        REWIND 30                                                            FLD00340
 1000   FORMAT(A3,1X,I1,75A1)                                                FLD00350
        IF(TYPE.EQ.TYPES )GO TO 20                                           FLD00360
        IF(SWCHG.NE.0)GO TO 40                                               FLD00370
        TYPE = TYPES                                                         FLD00380
C                                                                            FLD00390
CC       READ CARD                                                           FLD00400
C                                                                            FLD00410
 20     COL = 0                                                              FLD00420
        CATNM = NXTCHR(CARD,COL)                                             FLD00430
C*      IF NEXT CHAR IS NOT A CAT. NAME, CORRECT COL COUNT TO READ NUM       FLD00440
        IF(CATNM.GT.0)GO TO 21                                               FLD00450
        IF(CATNM.EQ.CATNM1)GO TO 23                                          FLD00460
        NOCAT=NOCAT + 1                                                      FLD00470
        CATNAM(NOCAT)=CATNM                                                  FLD00480
        CATNM1 = CATNM                                                       FLD00490
        GO TO 23                                                             FLD00500
 21     COL=COL - 1                                                          FLD00510
 23     NDCARD=0                                                             FLD00520
        CALL NUMBR(NDOTS,NDCARD,CARD,COL)                                    FLD00530
        IF(NDCARD.EQ.0)GO TO 10                                              FLD00540
        ICNT = 0                                                             FLD00550
        STAMNT = 2                                                           FLD00560
        SWITCH = .TRUE.                                                      FLD00570
        GO TO 100                                                            FLD00580
C                                                                            FLD00590
CC      TEST FOR END OF DOTS TO BE PROCESSED ON CARD                         FLD00600
C                                                                            FLD00610
 30     IF(ICNT.LT.NDCARD)GO TO 100                                          FLD00620
C                                                                            FLD00630
CC      READ NEXT CARD                                                       FLD00640
C                                                                            FLD00650
        STAMNT = 1                                                           FLD00660
        ICNT = 0                                                             FLD00670
        READ(21,103)(ACARD(I),I=1,80)                                        FLD00680
        WRITE(30,103)(ACARD(I),I=1,80)                                       FLD00690
        REWIND 30                                                            FLD00700
        READ(30,1000 )ID,TYPES,CARD                                          FLD00710
        REWIND 30                                                            FLD00720
        IF(ID.EQ.ENDBCD)RETURN 3                                             FLD00730
        IF(TYPE.EQ.TYPES)GO TO 20                                            FLD00740
        SWITCH = .FALSE.                                                     FLD00750
        SWCHG = SWCHG + 1                                                    FLD00760
        IF(SWCHG.GT.1)GO TO 40                                               FLD00770
        TYPE = TYPES                                                         FLD00780
         IPT = 0                                                             FLD00790
```

FILE: FLDLAC FORTRAN A

```
C***** CHANGED JUNE 28 1978                                                    FLD00800
      RETURN 2                                                                 FLD00810
C                                                                              FLD00820
C                                                                              FLD00830
  100 ICNT = ICNT + 1                                                          FLD00840
      NOFLD2 = NOFLD2 + 1                                                      FLD00850
C                                                                              FLD00860
CC    COMPUTE LINE INCREMENT                                                   FLD00870
C                                                                              FLD00880
      NN = NDOTS(ICNT)                                                         FLD00890
      NI =IABS( NN) / 100000000                                                FLD00900
      LI = IABS(NN) - NI * 100000000                                           FLD00910
      IF(LI.GE.100000000)NI = NI +1                                            FLD00920
C                                                                              FLD00930
CC    COMPUTE SAMPLE INCREMENT                                                 FLD00940
C                                                                              FLD00950
      KK=1                                                                     FLD00960
      IF(NN.LT.0)KK=-1                                                         FLD00970
      LI = NI * KK                                                             FLD00980
      N2 = NN - LI * 100000000                                                 FLD00990
      N3 = IABS( N2)/10000                                                     FLD01000
      SI = IABS(N2)-N3 * 10000                                                 FLD01010
      IF(SI.GE.1000)N3 = N3 + 1                                                FLD01020
      KK=1                                                                     FLD01030
      IF(N2.LT.0)KK=-1                                                         FLD01040
      SI = N3 *KK                                                              FLD01050
      LACI = N2 - SI * 10000                                                   FLD01060
      LR = (LACI-1)/19                                                         FLD01070
      LR = (LR+1) * 10                                                         FLD01080
      LS = LR - 1                                                              FLD01090
      LS = LS /10                                                              FLD01100
      LS = 10 * (LACI - (LS*19))                                               FLD01110
      L = LR -LI                                                               FLD01120
      S = LS + SI                                                              FLD01130
C                                                                              FLD01140
CC    STORE DOT INFO                                                           FLD01150
C                                                                              FLD01160
      FIELDS(1,NOFLD2) = CATNM                                                 FLD01170
      FIELDS(4,NOFLD2) = 2                                                     FLD01180
      FLDINF(1) = L                                                            FLD01190
      FLDINF(2) = L                                                            FLD01200
      FLDINF(3) = 1                                                            FLD01210
      FLDINF(4) = S                                                            FLD01220
      FLDINF(5) = S                                                            FLD01230
      FLDINF(6) = 1                                                            FLD01240
      IF(IPT.NE.0)GO TO 35                                                     FLD01250
      IPT = -3                                                                 FLD01260
   35 IPT = IPT + 4                                                            FLD01270
      VERTEX(IPT) = S                                                          FLD01280
      VERTEX(IPT+1)=L                                                          FLD01290
      VERTEX(IPT+2)=S                                                          FLD01300
      VERTEX(IPT+3)=L                                                          FLD01310
      RETURN 1                                                                 FLD01320
   40 WRITE(6,2000)                                                            FLD01330
 2000 FORMAT(//5X,'ERROR HAS OCCURRED IN READING LACIE FORMATTED DOT CARFLD01340
     *DS - SUBROUTINE FLDLAC - EXIT TAKEN')                                    FLD01350
      RETURN 3                                                                 FLD01360
      END                                                                      FLD01370
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: FLDTYP FORTRAN A

```
      SUBROUTINE FLDTYP(FIELDS,STAMNT,*,*,*,IPT,VERTEX)                      FLD00010
      IMPLICIT INTEGER (A-Z)                                                 FLD00020
C     INCLUDE COMBK1.LIST                                                    FLD00030
C     INCLUDE CMBK14.LIST                                                    FLD00040
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,               FLD00050
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,         FLD00060
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),             FLD00070
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                  FLD00080
     *              GRPCHK(61),GROUPS(124)                                   FLD00090
      COMMON /DOTVEC/TYPE,CATNAM(60),NOCAT,TOTVEC,FLDINF(6),PRTKEY           FLD00100
     *     ,SIZE,LACIE                                                       FLD00110
CSEND                                                                        FLD00120
C                                                                            FLD00130
      DIMENSION FIELDS(4,1),VERTEX(1)                                        FLD00140
      DIMENSION CARD(62)                                                     FLD00150
      LOGICAL SWITCH                                                         FLD00160
      DATA SWITCH/.TRUE./                                                    FLD00170
      IPT = IPT + FIELDS(4,NOFLD2)*2                                         FLD00180
      GO TO (90,100),STAMNT                                                  FLD00190
   90 I = LAREAD(FIELDS(1,NOFLD2+1),VERTEX(IPT),FLDINF(1),FIELDS(4,NOFLD     FLD00200
     *2+1))                                                                  FLD00210
C                                                                            FLD00220
CCC   WAS CLASS,SUBCLASS,FIELD, OR $END* ENCOUNTERED                         FLD00230
C                                                                            FLD00240
      IF ( I .EQ. -4) GO TO 90                                               FLD00250
      IF ( I .EQ. -1) GO TO 120                                              FLD00260
      IF ( I .EQ.  1) GO TO 130                                              FLD00270
      IF ( I .EQ.  0) GO TO 140                                              FLD00280
C                                                                            FLD00290
CCC   TYPE CARD                                                              FLD00300
C                                                                            FLD00310
   100 IF (SWITCH) GO TO 100                                                 FLD00320
      STAMNT = 2                                                             FLD00330
      RETURN 2                                                               FLD00340
   100 READ(30,115)CARD                                                      FLD00350
   115 FORMAT(10X,62A1)                                                      FLD00360
      REWIND 30                                                              FLD00370
      COL = 0                                                                FLD00380
      ZERO = 0                                                               FLD00390
      M= NUMBER(CARD,COL,TYPE,ZERO)                                          FLD00400
      IPT = 1                                                                FLD00410
      NOFLD2 = 0                                                             FLD00420
      NOCAT = 0                                                              FLD00430
      SWITCH = .FALSE.                                                       FLD00440
      GO TO 80                                                               FLD00450
C                                                                            FLD00460
CCC   CLASSNAME CARD                                                         FLD00470
C                                                                            FLD00480
   110 FORMAT(10X,A4)                                                        FLD00490
   120 NOCAT = NOCAT + 1                                                     FLD00500
      READ(30,110)CATNAM(NOCAT)                                              FLD00510
      REWIND 30                                                              FLD00520
      GO TO 80                                                               FLD00530
C                                                                            FLD00540
CCC   FIELD CARD                                                             FLD00550
C                                                                            FLD00560
   130 NOFLD2 = NOFLD2 + 1                                                   FLD00570
      STAMNT = 1                                                             FLD00580
      RETURN 1                                                               FLD00590
C                                                                            FLD00600
CCC   $END*                                                                  FLD00610
C                                                                            FLD00620
   140 SWITCH=.TRUE.                                                         FLD00630
      RETURN 3                                                               FLD00640
C                                                                            FLD00650
      END                                                                    FLD00660
                                                                             FLD00670
```

FILE: NUMBR FORTRAN A

```
C*      SUBROUTINE NUMBR WILL PROCESS ONE CARD AT A TIME.                  NUM00010
C*      IT READS AND STORES ALL NUMBERS IN ARRAY NDOTS.WITH                NUM00020
C*      NDCARD AS AN INDEX.  BLANKS ARE THE ONLY RECONIZED                 NUM00030
C*       DELIMITERS.                                                       NUM00040
C*      SUBROUTINE NUMBR(NDOTS,NDCARD,CARD,COL)                            NUM00050
        IMPLICIT INTEGER (A-Z)                                             NUM00060
        DIMENSION NDOTS(1),CARD(1)                                         NUM00070
        DATA BLANK/' '/,CRDSIZ/75/                                         NUM00080
        NUM=0                                                              NUM00090
        NC = COL - 1                                                       NUM00100
    5   IF (NC.GT.CRDSIZ)GO TO 50                                          NUM00110
        DO 10 I=NC,CRDSIZ                                                  NUM00120
        IF(CARD(I).EQ.BLANK)GO TO 7                                        NUM00130
        CALL I4A1BN(CARD(I),1,NWORD)                                       NUM00140
        NUM = NUM*10 + NWORD                                               NUM00150
        GO TO 30                                                           NUM00160
    7   IF(NUM.LT.1)GO TO 30                                               NUM00170
        IF(NUM.GT.209)WRITE(6,500)NUM                                      NUM00180
        NDCARD=NDCARD + 1                                                  NUM00190
        NDOTS(NDCARD)=NUM                                                  NUM00200
        NUM = 0                                                            NUM00210
   30   CONTINUE                                                           NUM00220
   10   CONTINUE                                                           NUM00230
  500   FORMAT(//5X,'LACIE DOT READ THAT IS GREATER THAN SIZE LIMIT        NUM00240
       :OF 209 - EXECUTION CONTINUED WITH VALUE READ OF ',I4)              NUM00250
   50   CONTINUE                                                           NUM00260
        RETURN                                                             NUM00270
        END                                                                NUM00280
```

FILE: SET13 FORTRAN A

```
      SUBROUTINE SET13                                                  SET00010
      IMPLICIT INTEGER (A-Z)                                            SET00020
      DIMENSION CODE(9),CARD(62),EQUCOM(3),ACARD(20)                    SET00030
      DIMENSION SLASH(2)                                                SET00040
      DATA SLASH /1H/,1H/                                               SET00050
      DATA CODE/4HCHAN,4HDATA,4HDOTF,                                   SET00060
     * 4HOPTI,4HDATE,4HCOMM,4HHED1,4HHED2,4H*END/                       SET00070
      DATA  EQUCOM/2,1H=,1H,/                                           SET00080
      DATA D/1HD/,BLNK/1H /,U/1HU/,FF/1HF/,OO/1H0/,P/1HP/               SET00090
      DATA L/1HL/                                                       SET00100
C       INCLUDE COMMK1,LIST                                             SET00110
CCC     INCLUDE COMMK4,LIST                                             SET00120
CCC    INCLUDE COMMK6,LIST                                              SET00130
CC    INCLUDE CMMK14,LIST                                               SET00140
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFLD2,          SET00150
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    SET00160
     *              FETVC2(30),SURVC2(75),SURPTR(75),CLSVC2(60),        SET00170
     *              KEPPTS(40),NOGRP,GRPNAM(60),GRPDEX(61),             SET00180
     *              GRPCHR(A1),GROUPS(124)                              SET00190
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                    SET00200
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                 SET00210
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               SET00220
      COMMON/GLOBAL/HEAD(A3),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,         SET00230
     *              HISFIL,HISKEY,TPFORM,ERIPTR,ERRKEY,MAPUNT,NOFILE,   SET00240
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               SET00250
     *  ,NMSTUN,NMSTFI,SCTMUN,MAPFIL                                    SET00260
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              SET00270
     *  CRDUNT,PRTUNT,HARDIO                                            SET00280
      COMMON /DOTVEC/TYPE,CATNAM(60),NOCAT,TOTVEC,FLDINF(6),PRTKEY      SET00290
     *              ,SIZE , LACIE                                       SET00300
CSEND                                                                   SET00310
      ZERO = 0                                                          SET00320
      NOFET2 = 0                                                        SET00330
      FIELD  = 1                                                        SET00340
      PRTKEY = 0                                                        SET00350
      NPUT=0                                                            SET00360
      LACIE = 0                                                         SET00370
C                                                                       SET00380
      WRITE(6,100)                                                      SET00390
  100 FORMAT(/11X,'INPUT SUMMARY'//)                                    SET00400
CCC                                                                     SET00410
CCC   SET UP REREAD BUFFER                                              SET00420
C                                                                       SET00430
      RRUNIT = 30                                                       SET00440
      CALL REREAD(RRUNIT,80)                                            SET00450
CCC                                                                     SET00460
CCC   PUT CARD IN BUFFER                                                SET00470
C                                                                       SET00480
  105 READ(2,103)(ACARD(I),I=1,20)                                      SET00490
  103 FORMAT(20A4)                                                      SET00500
      WRITE(30,103)(ACARD(I),I=1,20)                                    SET00510
      REWIND RRUNIT                                                     SET00520
C                                                                       SET00530
      READ(30,110)CODE1,CARD                                            SET00540
      REWIND RRUNIT                                                     SET00550
      COL= 0                                                            SET00560
      WRITE(6,120)CODE1,CARD                                            SET00570
  120 FORMAT(1X,A4,6X,62A1)                                             SET00580
  110 FORMAT(A4,6X,62A1)                                                SET00590
      DO 130  I=1,NPUT                                                  SET00600
      IF (CODE1.EQ.CODE(I)) GO TO (150,180,210,330,370,                 SET00610
     *                             390,400,410,420),I                   SET00620
  130    CONTINUE                                                       SET00630
  135 WRITE(6,140)                                                      SET00640
  140 FORMAT(' INVALID CONTROL CARD - IGNORED ')                        SET00650
      GO TO 105                                                         SET00660
CCC                                                                     SET00670
CCC   CHANNEL CARD                                                      SET00680
C                                                                       SET00690
  150 M = NXTCHR(CARD,COL)                                              SET00700
      IF (M.EQ.0) GO TO 155                                             SET00710
      IF (M.EQ.BLNK) GO TO 105                                          SET00720
  152 WRITE(6,153)                                                      SET00730
  153 FORMAT(' ERROR ON DATA CARD')                                     SET00740
      GO TO 105                                                         SET00750
  155 J = FIND12(CARD,COL,EQUCOM)                                       SET00760
      IF (J .NE. 2) GO TO 152                                           SET00770
      NOFET2 = NUMBER(CARD,COL,FETVC2,NOFET2)                           SET00780
      NOFET2 = NUMBER(CARD,COL,FETVC2,NOFET2)                           SET00790
```

```
FILE: SET13    FORTRAN  A

      CALL ORDER(FETVC2,NOFET2)                                          SET00800
      GO TO 105                                                          SET00810
C                                                                        SET00820
C     DATA FILE CARD                                                     SET00830
C                                                                        SET00840
  180 M = NXTCHR(CARD,COL)                                               SET00850
      IF (M .EQ. BLNK ) GO TO 105                                        SET00860
      IF (M.EQ.U) GO TO 190                                              SET00870
      IF (M.EQ.FF) GO TO 200                                             SET00880
  185 WRITE(A,187)                                                       SET00890
  187 FORMAT(' ERROR ON DATA FILE CARD')                                 SET00900
      GO TO 105                                                          SET00910
  190 J = FIND12(CARD,COL,EQUCOM)                                        SET00920
      IF (J .NE. 2) GO TO 185                                            SET00930
      M = NUMBER(CARD,COL,DATAPE,ZERO)                                   SET00940
      COL = COL - 1                                                      SET00950
      GO TO 180                                                          SET00960
  200 J = FIND12(CARD,COL,EQUCOM)                                        SET00970
      IF ( J .NE. 2 ) GO TO 185                                          SET00980
      M = NUMBER(CARD,COL,DATFIL,ZERO)                                   SET00990
      DATFIL = DATFIL - 1                                                SET01000
      COL = COL - 1                                                      SET01010
      GO TO 180                                                          SET01020
C                                                                        SET01030
C     DOT FILE CARD                                                      SET01040
C                                                                        SET01050
  210 M = NXTCHR(CARD,COL)                                               SET01060
      IF (M.EQ.00) GO TO 213                                             SET01070
      IF (M.EQ.BLNK) GO TO 105                                           SET01080
      GO TO 215                                                          SET01090
  213 J = FIND12(CARD,COL,SLASH)                                         SET01100
      IF (J .EQ. -1) GO TO 215                                           SET01110
  214 M = NXTCHR(CARD,COL)                                               SET01120
      IF (M .EQ. BLNK ) GO TO 105                                        SET01130
      IF (M.EQ.U) GO TO 230                                              SET01140
      IF (M.EQ.FF) GO TO 240                                             SET01150
  215 WRITE(A,220)                                                       SET01160
  220 FORMAT(' ERROR ON DOT FILE CARD')                                  SET01170
      GO TO 105                                                          SET01180
  230 J = FIND12(CARD,COL,EQUCOM)                                        SET01190
      IF ( J .NE. 2)  GO TO 215                                          SET01200
      M = NUMBER(CARD,COL,DOTUNT,ZERO)                                   SET01210
      COL = COL - 1                                                      SET01220
      GO TO 214                                                          SET01230
  240 J = FIND12(CARD,COL,EQUCOM)                                        SET01240
      IF (J .NE.  2) GO TO 215                                           SET01250
      M = NUMBER(CARD,COL,DOTFIL,ZERO)                                   SET01260
      DOTFIL = DOTFIL - 1                                                SET01270
      COL = COL - 1                                                      SET01280
      GO TO 214                                                          SET01290
C                                                                        SET01300
C     OPTION CARD                                                        SET01310
C                                                                        SET01320
  330 M = NXTCHR(CARD,COL)                                               SET01330
      IF (M .EQ. BLNK ) GO TO 105                                        SET01340
      IF (M.EQ.P) GO TO 340                                              SET01350
      IF(M.EQ.L)GO TO 345                                                SET01360
C                                                                        SET01370
C*** CODE ADDED NOV 21, 1978 IN SUPPORT OF LIST PROCESSING               SET01380
C           IF (M.EQ.U) GO TO 350                                        SET01390
C                                                                        SET01400
C                                                                        SET01410
  335 WRITE(A,335)                                                       SET01420
  336 FORMAT(' ERROR ON OPTION CARD')                                    SET01430
      GO TO 105                                                          SET01440
  340 PRTKEY = 1                                                         SET01450
      GO TO 105                                                          SET01460
  345  LACIE = 1                                                         SET01470
      GO TO 105                                                          SET01480
  350       M = NUMBER(CARD,COL,LACIE,ZERO)                            SET01490
            GO TO 105                                                    SET01500
C                                                                        SET01510
C     DATE CARD                                                          SET01520
C                                                                        SET01530
  370 M = NXTCHR(CARD,COL)                                               SET01540
      IF ( M .EQ. BLNK ) GO TO 105                                       SET01550
      READ(30,380)DATE                                                   SET01560
  380  FORMAT(10X,1544)                                                  SET01570
      REWIND RRUNIT                                                      SET01580
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SET13 FORTRAN A

```
      GO TO 105                                                          SET01590
                                                                         SET01600
C     COMMENT CARD                                                       SET01610
C                                                                        SET01620
  390 M = NXTCHR(CARD,COL)                                               SET01630
      IF (M .EQ. BLNK ) GO TO 105                                        SET01640
      READ(30,380)COMENT                                                 SET01650
      REWIND RRUNIT                                                      SET01660
      GO TO 105                                                          SET01670
                                                                         SET01680
C                                                                        SET01690
C     HED1                                                               SET01700
  400 M = NXTCHR(CARD,COL)                                               SET01710
      READ(30,380) HED1                                                  SET01720
      REWIND RRUNIT                                                      SET01730
      GO TO 105                                                          SET01740
                                                                         SET01750
C                                                                        SET01760
C     HED2                                                               SET01770
C                                                                        SET01780
  410 M = NXTCHR(CARD,COL)                                               SET01790
      READ(30,380) HED2                                                  SET01800
      REWIND RRUNIT                                                      SET01810
      GO TO 105                                                          SET01820
                                                                         SET01830
C     *END*                                                              SET01840
C                                                                        SET01850
  420 CONTINUE                                                           SET01860
      IF (NOFET2 .NE. 0) GO TO 440                                       SET01870
      DO 430 I=1,30                                                      SET01880
      FETVC2(I) = I                                                      SET01890
  430    CONTINUE                                                        SET01900
      NOFET2 = I                                                         SET01910
C                                                                        SET01920
  440 SIZE = 4 + NOFET2                                                  SET01930
C                                                                        SET01940
C                                                                        SET01950
      WRITE(6,1000)                                                      SET01960
      IF (NOFET2 .NE. 0) WRITE(6,1010) (FETVC2(I),I=1,NOFET2)            SET01970
      IF (PRTKEY .EQ. 1) WRITE(6,1030)                                   SET01980
 1040 FORMAT(' LACIE FORMATTED DOT CARDS USED AS EOD-LARSYS FIELD CARDS'SET01990
     :)                                                                  SET02000
      IF(LACIE.EQ.1)WRITE(6,1040)                                        SET02010
 1000 FORMAT(//' USER HAS REQUESTED THE FOLLOWING OPTIONS :'/)           SET02020
 1010 FORMAT(' SELECTED CHANNELS ARE',30I3)                              SET02030
 1030 FORMAT(' PRINT DATA VECTORS')                                      SET02040
C                                                                        SET02050
      RETURN                                                             SET02060
C                                                                        SET02070
      END
```

18. LABEL PROCESSOR

FILE: LABEL FORTRAN A

```
      SUBROUTINE LABEL(ARRAY,TOP)                                        LAB00010
C                                                                        LAB00020
      IMPLICIT INTEGER (A-Z)                                             LAB00030
      DIMENSION FLDSAV(2000),ARRAY(1)                                    LAB00040
C                                                                        LAB00050
CCC   GO READ CONTROL CARDS                                              LAB00060
C                                                                        LAB00070
      CALL SET14(ARRAY,TOP)                                              LAB00080
C                                                                        LAB00090
CCC   READ IN REQUIRED FILES                                             LAB00100
C                                                                        LAB00110
      CALL FILERD(ARRAY,TOP,NOFLD,TOTVRT,FLDSAV(1),FLDSAV(1001))         LAB00120
C                                                                        LAB00130
CCC   READY TO PERFORM USER REQUESTS                                     LAB00140
C                                                                        LAB00150
      CALL LABLR(ARRAY,TOP,NOFLD,TOTVRT,FLDSAV(1),FLDSAV(1001),          LAB00160
     *     FLDSAV(1))                                                    LAB00170
C                                                                        LAB00180
CCC   FINISHED -- SEND* CARD                                             LAB00190
C                                                                        LAB00200
      READ(21,100)CARD                                                   LAB00210
  100 FORMAT(A4)                                                         LAB00220
      RETURN                                                             LAB00230
      END                                                                LAB00240
```

FILE: ALLKIN FORTRAN A

```
      SUBROUTINE ALLKIN(DOTS,SUBVEC,SUBNO,CATVEC,MEANS)                  ALL00010
C                                                                        ALL00020
CC    LABELS BY ALL-OF-A-KIND PROCEDURE                                  ALL00030
C                                                                        ALL00040
      IMPLICIT INTEGER (A-Z)                                             ALL00050
C                                                                        ALL00060
CCC   INCLUDE COMBK1.LIST                                                ALL00070
CCCC  INCLUDE COMBK4.LIST                                                ALL00080
CCCCC INCLUDE CMBK15.LIST                                                ALL00090
C     INCLUDE COMBK6.LIST                                                ALL00100
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           ALL00110
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     ALL00120
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         ALL00130
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              ALL00140
     *              GRPCHR(61),GROUPS(124)                               ALL00150
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                     ALL00160
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                  ALL00170
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                ALL00180
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),  ALL00190
     *            SUBRAY(120),PTR(60),CATPTR(250),CATDOT(500),           ALL00200
     *            DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,        ALL00210
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,            ALL00220
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),             ALL00230
     *           CLSSYM(62),STADRS,MEANAD,TABADR,MAPADR,SUNCOR(30),      ALL00240
     *       ODOTUN,ODOTFI,MANSTA,MANDOT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,    ALL00250
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                    ALL00260
     *    ,TOTDT3,FLDADR,VTXADR                                          ALL00270
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          ALL00280
     *             HISFIL,HISKEY,TRFORM,ERIPTP,EPPKEY,MAPUNT,NOFILE,     ALL00290
     *    DRUMAD,DRMBOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 ALL00300
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                     ALL00310
     *  ,DOTUNT,DOTFIL,NCHPAS,TPNSFL,BMTRFL,HISTFL,PCHUNT,               ALL00320
     *  CRDUNT,PRTUNT,RANDIO                                             ALL00330
C$END                                                                    ALL00340
      DIMENSION DOTS(SIZE,TOTDT2),SUBVEC(1),SUBNO(1),CATVEC(1),          ALL00350
     *          CATGRY(60 ),CLSTNO(250),              DOTNUM(250),       ALL00360
     *          DISTNC(250)                                            ALL00370
      DIMENSION MEANS(NOFET2,1)                                          ALL00380
      DIMENSION TIES(250)                                                ALL00390
C     RETRIEVE CLUSTER CLASSIFICATION FOR EACH DOT                       ALL00400
C                                                                        ALL00410
      NSAMP = (FLDINF(5)-FLDINF(4))/FLDINF(6) + 1                        ALL00420
      LINES =(FLDINF(2) - FLDINF(2))/FLDINF(3) + 1                       ALL00430
C                                                                        ALL00440
      DO 100 I=1,TOTDT2                                                  ALL00450
      ILINE = DOTS(2,I)                                                  ALL00460
      ISAMP = DOTS(1,I)                                                  ALL00470
      PIXADR = DRUMAD + (ILINE-1)*NSAMP + ISAMP-1                        ALL00480
      CALL RREAD(PIXADR,NUMBER,1,ISTAT)                                  ALL00490
C                                                                        ALL00500
CC    CLSTNO -- CONTAINS THE CLUSTER NO. OF EACH DOT                     ALL00510
C                                                                        ALL00520
   90 IF (ISTAT .EQ. 1) GO TO 90                                         ALL00530
  100 CLSTNO(I) = NUMBER                                                 ALL00540
C                                                                        ALL00550
      DO 300 I=1,NOSUB2                                                  ALL00560
      WRITE(6,HEAD)                                                      ALL00570
      WRITE(6,110)                                                       ALL00580
  110 FORMAT(T50,'LABELING BY ALL-OF-A-KIND PROCEDURE')                  ALL00590
C                                                                        ALL00600
CC    REINITIALIZE                                                       ALL00610
C                                                                        ALL00620
      DO 155 J=1,NOCAT                                                   ALL00630
  155 CATGRY(J) = 0                                                      ALL00640
      MAX = 0                                                            ALL00650
      NODOT = 0                                                          ALL00660
      TIE = 0                                                            ALL00670
C                                                                        ALL00680
CC    FIND ALL DOTS IN CLUSTER I                                         ALL00690
C                                                                        ALL00700
      DO 150 J=1,TOTDT2                                                  ALL00710
      IF (CLSTNO(J).NE. I) GO TO 150                                     ALL00720
      K = DOTS(4,J)                                                      ALL00730
      CATGRY(K) = CATGRY(K) + 1                                          ALL00740
      NODOT = NODOT + 1                                                  ALL00750
      DOTNUM(NODOT) = J                                                  ALL00760
  150 CONTINUE                                                           ALL00770
C                                                                        ALL00780
C     WERE DOTS - 1 OF A KIND                                            ALL00790
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: ALLKIN FORTRAN A

```
C       WERE DOTS- MIXED                                                        ALL00800
C       WERE DOTS-- NULL SET                                                    ALL00810
        DO 140 J=1,NOCAT                                                        ALL00820
        IF (CATGRY(J) .LT. MAX) GO TO 160                                       ALL00830
        MAX = CATGRY(J)                                                         ALL00840
        CATNUM = J                                                              ALL00850
  140 CONTINUE                                                                  ALL00860
C                                                                               ALL00870
        IF (MAX .NE. 0) GO TO 170                                               ALL00880
C                                                                               ALL00890
CCC     DEFAULT TO K-NEAREST NEIGHBOR                                           ALL00900
C                                                                               ALL00910
        ITER = 1                                                                ALL00920
        TAB1 = TABADR + (I-1)*TOTDT2                                            ALL00930
        SWTCH = 1                                                               ALL00940
C                                                                               ALL00950
C                                                                               ALL00960
        WRITE(6,165)                                                            ALL00970
  165 FORMAT(///T43,'** DEFAULTING TO K-NEAREST NEIGHBOR PROCEDURE **')         ALL00980
        CALL KNEAR(DOTS,DUM,DUM,CATVEC,ITER,TAB1,SWTCH,CATNUM,I,MEANS)          ALL00990
        GO TO 300                                                               ALL01000
C                                                                               ALL01010
CCC     LABEL BY MAJORITY                                                       ALL01020
C                                                                               ALL01030
  170 CONTINUE                                                                  ALL01040
C                                                                               ALL01050
CCCC    PRINT CLUSTER INFORMATION                                               ALL01060
C                                                                               ALL01070
C                                                                               ALL01080
        TAB1 = TABADR + (I-1)*TOTDT2                                            ALL01090
        CALL RREAD(TAB1,DISTNC,TOTDT2,ISTAT)                                    ALL01100
C                                                                               ALL01110
CCC   GO TO CODE FOR A TIE                                                      ALL01120
C                                                                               ALL01130
        GO TO 390                                                               ALL01140
  175 WRITE(6,180)CATNAM(CATNUM)                                                ALL01150
  180 FORMAT(/1X,'LABEL : ',A4)                                                 ALL01160
        FEATR = NOFET2                                                          ALL01170
        ISTART = 1                                                              ALL01180
        IEND = 15                                                               ALL01190
  185 IF (IEND .GT. FEATR) IEND = FEATR                                         ALL01200
        IENDS = ISTART + IEND - 1                                               ALL01210
        IF (ISTART .NE. 1) GO TO 187                                            ALL01220
        WRITE(6,190)(MEANS(J,I),J=ISTART,IENDS)                                 ALL01230
        GO TO 192                                                               ALL01240
  187 WRITE(6,188)(MEANS(J,I),J=ISTART,IENDS)                                   ALL01250
  188 FORMAT(/9X,15(F6.2,2X))                                                   ALL01260
  190 FORMAT(/1X,'MEANS : ',15(F6.2,2X))                                        ALL01270
C                                                                               ALL01280
  192 CONTINUE                                                                  ALL01290
        FEATR = FEATR - IEND                                                    ALL01300
        ISTART = IENDS + 1                                                      ALL01310
        IF (FEATR .GT. 0) GO TO 185                                             ALL01320
C                                                                               ALL01330
        WRITE(6,193)                                                            ALL01340
  193 FORMAT(/)                                                                 ALL01350
        WRITE(6,195)                                                            ALL01360
  195 FORMAT(40X,'DOTS',9X,'ANALYST LABEL',11X,'DISTANCE'/)                     ALL01370
        DO 210 J=1,NODOT                                                        ALL01380
        DST = DOTNUM(J)                                                         ALL01390
        K = DOTS(4,DST)                                                         ALL01400
        WRITE(6,205)DOTVEC(DST),CATNAM(K),DISTNC(DST)                           ALL01410
  205 FORMAT(40X,I3,1H,,13X,A4,14X,F7.2)                                        ALL01420
  210 CONTINUE                                                                  ALL01430
C                                                                               ALL01440
        IF (TIF .EQ. 0) GO TO 215                                               ALL01450
C                                                                               ALL01460
        WRITE(6,211)                                                            ALL01470
  211 FORMAT(//T43,'A TIE OCCURRED.  THE FOLLOWING DOT(S) WERE DISCARDED      ALL01480
     * :'/)                                                                     ALL01490
C                                                                               ALL01500
        DO 213  J=1,TIE                                                         ALL01510
        INDEX1 = TIES(J)                                                        ALL01520
        LABELS = DOTS(4,INDEX1)                                                 ALL01530
  213 WRITE(6,205)DOTVEC(INDEX1),CATNAM(LABELS),DISTNC(INDEX1)                  ALL01540
  215 CONTINUE                                                                  ALL01550
        WRITE(6,220)                                                            ALL01560
  220 FORMAT(///T60,'TOTALS',/T56,'LABEL',T65,'NO.'/)                           ALL01570
        DO 240 J=1,NOCAT                                                        ALL01580
```

FILE: ALLKIN FORTRAN A

```
  240 WRITE(6,230)CATNAM(J),CATGRY(J)                                   ALL01590
  230 FORMAT(T56,A4,T65,I3)                                             ALL01600
C                                                                       ALL01610
  300 CONTINUE                                                          ALL01620
C                                                                       ALL01630
C     GROUP LABELED CLUSTERS                                            ALL01640
C                                                                       ALL01650
      K = 0                                                             ALL01660
      DO 310 I=1,NOCAT                                                  ALL01670
      DO 310 J=1,NOSUR2                                                 ALL01680
      IF (CATVEC(J) .NE. I) GO TO 310                                   ALL01690
      SUBNO(I) = SUBNO(I) + 1                                           ALL01700
      K = K + 1                                                         ALL01710
      SUBVEC(K) = J                                                     ALL01720
  310 CONTINUE                                                          ALL01730
      RETURN                                                            ALL01740
C                                                                       ALL01750
C    CODE FOR A TIE                                                     ALL01760
C                                                                       ALL01770
  390 IF (MAX .EQ. 0) GO TO 175                                         ALL01780
      DO 400  II=1,NOCAT                                                ALL01790
      IF (II .EQ. CATNUM) GO TO 400                                     ALL01800
      IF (MAX .EQ. CATGRY(II)) GO TO 410                                ALL01810
  400 CONTINUE                                                          ALL01820
      CATVEC(I) = CATNUM                                                ALL01830
      GO TO 175                                                         ALL01840
C                                                                       ALL01850
  410 TIE = TIE + 1                                                     ALL01860
      MAXDST = 0                                                        ALL01870
      DO 420 J=1,NODOT                                                  ALL01880
      NO = DOTNUM(J)                                                    ALL01890
      DST = DISTNC(NO)                                                  ALL01900
      IF (MAXDST .GT. DST) GO TO 420                                    ALL01910
      DOTNO = NO                                                        ALL01920
      MAXDST = DST                                                      ALL01930
      INDEX = J                                                         ALL01940
      LABEL = DOTS(4,NO)                                                ALL01950
  420 CONTINUE                                                          ALL01960
C                                                                       ALL01970
      TIES(TIE) = DOTNO                                                 ALL01980
C                                                                       ALL01990
      CATGRY(LABEL) = CATGRY(LABEL) - 1                                 ALL02000
      NODOT = NODOT - 1                                                 ALL02010
      IF (INDEX .EQ. (NODOT + 1)) GO TO 445                             ALL02020
C                                                                       ALL02030
      DO 440  II=INDEX,NODOT                                            ALL02040
  440 DOTNUM(II) = DOTNUM(II+1)                                         ALL02050
C                                                                       ALL02060
  445 CONTINUE                                                          ALL02070
      MAX = 0                                                           ALL02080
      DO 470  II=1,NOCAT                                                ALL02090
      IF (CATGRY(II) .LE. MAX) GO TO 470                                ALL02100
      MAX = CATGRY(II)                                                  ALL02110
      CATNUM = II                                                       ALL02120
  470 CONTINUE                                                          ALL02130
      GO TO 390                                                         ALL02140
      END                                                               ALL02150
```

FILE: ASCEND FORTRAN A

```
      SUBROUTINE ASCEND(SCN,LNCAT,PTR1,PTR2)                            ASC00010
       IMPLICIT INTEGER(A-X)                                            ASC00020
       DIMENSION PTR1(LNCAT),PTR2(LNCAT)                                ASC00030
       REAL SCN(LNCAT),SAVE                                             ASC00040
       J=0                                                              ASC00050
 60    J=J+1                                                            ASC00060
       IF(J.GT.LNCAT)GO TO 90                                           ASC00070
       IF(J.EQ.LNCAT)GO TO 75                                           ASC00080
      IF (SCN(J) .GT. SCN(J+1)) GO TO 70                                ASC00090
       GO TO 60                                                         ASC00100
C                                                                       ASC00110
 70    SAVE=SCN(J)                                                      ASC00120
       SCN(J)=SCN(J+1)                                                  ASC00130
       SCN(J+1)=SAVE                                                    ASC00140
C                                                                       ASC00150
       SAVE1=PTR1(J)                                                    ASC00160
       PTR1(J)=PTR1(J+1)                                                ASC00170
       PTR1(J+1)=SAVE1                                                  ASC00180
C                                                                       ASC00190
       SAVE2=PTR2(J)                                                    ASC00200
       PTR2(J)=PTR2(J+1)                                                ASC00210
       PTR2(J+1)=SAVE2                                                  ASC00220
 75    K=J                                                              ASC00230
 80    IF(K.EQ.1)GO TO 60                                               ASC00240
      IF (SCN(K) .GT. SCN(K-1)) GO TO 60                                ASC00250
C                                                                       ASC00260
       SAVE=SCN(K-1)                                                    ASC00270
       SCN(K-1)=SCN(K)                                                  ASC00280
       SCN(K)=SAVE                                                      ASC00290
C                                                                       ASC00300
       SAVE1=PTR1(K-1)                                                  ASC00310
       PTR1(K-1)=PTR1(K)                                                ASC00320
       PTR1(K)=SAVE1                                                    ASC00330
C                                                                       ASC00340
       SAVE2=PTR2(K-1)                                                  ASC00350
       PTR2(K-1)=PTR2(K)                                                ASC00360
       PTR2(K)=SAVE2                                                    ASC00370
       K=K-1                                                            ASC00380
       GO TO 80                                                         ASC00390
 90    CONTINUE                                                         ASC00400
       RETURN                                                           ASC00410
       END                                                              ASC00420
```

FILE: CLRKEY FORTRAN A

```
      SUBROUTINE CLRKEY(XSIZ,IDATA,NOSUB2,CH,MEANS,NC)                  CLR00010
C                                                                       CLR00020
C*                                                                      CLR00030
C*    CLRKYS ADDS   THE COLOR KEYS TO A UNIVERSAL FORMAT TAPE           CLR00040
C*    THE COLORS ARE OUTPUT AS SQUARES INAGES (10X10)                   CLR00050
      IMPLICIT INTEGER (A-Z)                                            CLR00060
      DIMENSION IDATA(XSIZ,CH)                                          CLR00070
      REAL MEANS(NC,NOSUB2)                                             CLR00080
C                                                                       CLR00090
      LSTLIN = 0                                                        CLR00100
      LINE = 0                                                          CLR00110
      TOTKEY = 0                                                        CLR00120
      NKEYS = XSIZ/11                                                   CLR00130
      NOKEY = NOSUB2                                                    CLR00140
C                                                                       CLR00150
   90 DO 100 J=1,CH                                                     CLR00160
C*    WRITE A SCAN LINE OF ZEROS - USED FOR SEPARATING THE THE COLORS   CLR00170
C*                                                                      CLR00180
      DO 100 I=1,XSIZ                                                   CLR00190
  100 IDATA(I,J) = 0                                                    CLR00200
C                                                                       CLR00210
      CALL WRTLN(IDATA,LSTLIN)                                          CLR00220
      LINE = LINE + 1                                                   CLR00230
C                                                                       CLR00240
  110 IF (NKEYS .LE. NOKEY) NOKEY = NKEYS                               CLR00250
      KK = 0                                                            CLR00260
C                                                                       CLR00270
      DO 150 I=1,NOKEY                                                  CLR00280
      TOTKEY = TOTKEY + 1                                               CLR00290
      DO 140 J=1,NC                                                     CLR00300
      DO 130 K=1,10                                                     CLR00310
      KK = (I-1)*11 + K                                                 CLR00320
  130 IDATA(KK,J) = MEANS(J,TOTKEY) + 0.5                               CLR00330
  140 CONTINUE                                                          CLR00340
C*                                                                      CLR00350
C*    WRITE A SCAN LINE OF COLORS                                       CLR00360
C*                                                                      CLR00370
  150 CONTINUE                                                          CLR00380
      NOKEY = NOSUB2 - TOTKEY                                           CLR00390
C                                                                       CLR00400
      DO 160 I=1,10                                                     CLR00410
      IF (NOKEY .LE. 0 .AND. I .EQ. 10) LSTLIN = -1                     CLR00420
  160 CALL WRTLN(IDATA,LSTLIN)                                          CLR00430
      LINE = LINE + 10                                                  CLR00440
      IF (NOKEY .LE. 0) GO TO 170                                       CLR00450
      GO TO 90                                                          CLR00460
C                                                                       CLR00470
  170 CONTINUE                                                          CLR00480
C                                                                       CLR00490
C                                                                       CLR00500
      RETURN                                                            CLR00510
      END                                                               CLR00520
```

```
FILE: CLSMAP   FORTRAN  A

      SUBROUTINE CLSMAP(CATSUB,SWTCH,SUBNO,SUBVEC,SUBDES,CATVEC)          CLS00010
C                                                                         CLS00020
CCCC  OUTPUTS LINE PRINTER MAP(CONDITIONAL AND MIXED)                     CLS00030
C                                                                         CLS00040
CCCC  OUTPUTS MAPFIL TAPE                                                 CLS00050
C                                                                         CLS00060
CCCC  SWTCH = 1 -- COND. MAP                                              CLS00070
CCCC  SWTCH = 2 -- MIXED MAP                                              CLS00080
C                                                                         CLS00090
      IMPLICIT INTEGER (A-Z)                                              CLS00100
C     INCLUDE COMRK1.LIST                                                 CLS00110
CCC   INCLUDE CMRK15.LIST                                                 CLS00120
CCC   INCLUDE COMRK6.LIST                                                 CLS00130
C     INCLUDE COMRK4.LIST                                                 CLS00140
      COMMON/INFORM/NOCLS2,NOSUR2,NOFET2,VARSZ2,TOTVT2,NOFLD2,            CLS00150
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SURDS2,FLDSV2,VERTX2,      CLS00160
     *              FETVC2(30),SURVC2(75),SUBPTR(75),CLSVC2(60),          CLS00170
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),               CLS00180
     *              GRPCHK(61),GROUPS(124)                                CLS00190
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),   CLS00200
     *          SURRAY(120),PTR(60),CATPTR(250),CATDOT(500),              CLS00210
     *          DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,           CLS00220
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,             CLS00230
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),              CLS00240
     *          CLSSYM(62),SIADRS,MEANAD,TABADR,MAPADR,SUNCOR(30),        CLS00250
     *       ODOTUN,ODOTFI,MANSTA,MANDOT,OSPUNT,OSPFIL,OSPKEY,PRNSTS,     CLS00260
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                     CLS00270
     *    ,TOTDT3,FLDADR,VTXADR                                           CLS00280
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,           CLS00290
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,     CLS00300
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  CLS00310
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                      CLS00320
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                CLS00330
     *  CPDUNT,PRTUNT,HANDIO                                              CLS00340
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                      CLS00350
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                   CLS00360
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                CLS00370
CSEND                                                                     CLS00380
      DIMENSION SURVEC(1)                                                 CLS00390
      DIMENSION TITLE(3,2),            COL(3,110)                         CLS00400
      DIMENSION SYMBOL(62)                                                CLS00410
      DATA TITLE/'COND','ITIO','NAL ',' ',' MI','XED '/                   CLS00420
      DATA SYMBOL/'1','2','3','4','5','6','7','8','9','A',               CLS00430
     *           'B','C','D','E','F','G','H','I','J','K',               CLS00440
     *           'L','M','N','O','P','Q','R','S','T','U',               CLS00450
     *           'V','W','X','Y','Z','=','*','.','+','-',               CLS00460
     *           '&','0',',',' ','(',')','/','+','(','1H',              CLS00470
     *           '<','>','?','=','=','_','$','/','*','~',               CLS00480
     *           '#','$'/                                               CLS00490
C                                                                         CLS00500
      DIMENSION CATSUB(1),SUBNO(1),IR(1000),OUT(1000),COLORS(62)          CLS00510
      DIMENSION IDATA(1000),RELCLR(60)                                    CLS00520
      REAL RELCLR                                                         CLS00530
      DIMENSION CATVEC(1)                                                 CLS00540
      DIMENSION SUBDES(60)                                                CLS00550
      DIMENSION COLOR(64),COLATE(60),DELETE(60)                           CLS00560
      DATA COLOR/ 1,207,79,111,47,175,143,71,                             CLS00570
     * 167,105,107,171,199,135,39,71,                                     CLS00580
     *  205,137,75,41,45,69,73,75,                                       CLS00590
     * 77,133,137,139,141,33,35,37,                                       CLS00600
     *  39,41,43,45,47,103,105,107,                                      CLS00610
     *  109,197,201,203,205,169,171,173,                                 CLS00620
     *  131,135,137,141,67,71,75,77,                                     CLS00630
     *  161,163,165,167,169,171,225,239/                                 CLS00640
      DATA IPND/'#'/                                                      CLS00650
      COLORS(NOSUR2 + 1) = COLOR(1)                                       CLS00660
      COLORS(NOSUR2 + 2) = COLOR(1)                                       CLS00670
C     CHECK FOR DELETED CATEGORIES                                        CLS00680
      KEPT = 0                                                            CLS00690
      DO 2 II = 1,NOCAT                                                   CLS00700
      DELETE(II) = 0                                                      CLS00710
      DO 1 I = 1,NOSUR2                                                   CLS00720
      IF (CATSUB(I).EQ.II) DELETE(II) = 1                                 CLS00730
    1 CONTINUE                                                            CLS00740
      KEPT = KEPT + DELETE(II)                                            CLS00750
    2 CONTINUE                                                            CLS00760
      IF (KEPT.EQ.0) GO TO 7                                              CLS00770
      II = 1                                                              CLS00780
      SMALLS = 0                                                          CLS00790
```

```
FILE: CLSMAP   FORTRAN  A

    3 SMALL = 2**30 + (2**30-2)                                          CLS00800
      DO 4 I = 1,NOCAT                                                   CLS00810
      DUM = CATNAM(I)-2**31                                              CLS00820
      IF (DUM.LE.SMALLS) GO TO 4                                         CLS00830
      IF (DUM.GE.SMALL) GO TO 4                                          CLS00840
      III = I                                                            CLS00850
      SMALL = DUM                                                        CLS00860
 4    CONTINUE                                                           CLS00870
      COLATE(III) = II                                                   CLS00880
      SMALLS = SMALL                                                     CLS00890
      IF (DELETE(III).EQ.0) GO TO 6                                      CLS00900
      II = II + 1                                                        CLS00910
 6    IF (II.LE.KEPT) GO TO 3                                            CLS00920
 7    CONTINUE                                                           CLS00930
C                                                                       CLS00940
CC    ASSIGN CATEGORY SYMBOL TO EACH CLUSTER                            CLS00950
C                                                                       CLS00960
      DO 10 I=1,NOSUB2                                                   CLS00970
      CATNUM = CATSUB(I)                                                 CLS00980
      CLSSYM(I) = SYMBOL(CATNUM)                                         CLS00990
      IF (CATNUM.GT.NOCAT) GO TO 9                                       CLS01000
      DUM = COLATE(CATNUM)                                               CLS01010
      DELETE(I) = DUM                                                    CLS01020
      COLORS(I) = COLOR(DUM + 1)                                         CLS01030
      GO TO 10                                                           CLS01040
 9    COLORS(I) = COLOR(64 + KEPT - CATNUM)                              CLS01050
      DELETE(I) = 63 + KEPT - CATNUM                                     CLS01060
 10   CONTINUE                                                           CLS01070
       CLSSYM (NOSUB2 + 1) = IPND                                        CLS01080
       CLSSYM (NOSUB2 + 2) = IPND                                        CLS01090
      CALL SETMHG(68,0,68)                                               CLS01100
      WRITE(6,HEAD)                                                      CLS01110
C                                                                       CLS01120
      CALL MAPHND(NOCAT,CLSSYM,CATNAM,CATVEC,SUBDES,CATSUB)              CLS01130
C                                                                       CLS01140
      WRITE(6,5)(TITLE(I,SWTCH),I=1,3)                                   CLS01150
    5 FORMAT(//50X,34A,'CLUSTER MAP'/)                                   CLS01160
C                                                                       CLS01170
      ISTRT = FLDINF(4)                                                  CLS01180
      IEND = FLDINF(5)                                                   CLS01190
      SAMINC = FLDINF(6)                                                 CLS01200
      LININC = FLDINF(3)                                                 CLS01210
      LINSTR = FLDINF(1)                                                 CLS01220
      LINEND = FLDINF(2)                                                 CLS01230
      ILINE = (LINEND - LINSTR)/LININC + 1                               CLS01240
      NSAMP = (IEND-ISTRT)/SAMINC + 1                                    CLS01250
      PTS = 0                                                            CLS01260
C                                                                       CLS01270
CC    POSITION TAPE                                                     CLS01280
C                                                                       CLS01290
      REWIND OMAPUN                                                      CLS01300
      IF(OMAPFI.NE.0) CALL FSFMFL(OMAPUN,OMAPFI,ISTAT)                   CLS01310
      CALL WRTHED(1,1,NSAMP,1,OMAPUN)                                    CLS01320
C                                                                       CLS01330
C                                                                       CLS01340
CCC   PRINT LINE PRINTER MAP                                            CLS01350
C                                                                       CLS01360
       IPFLAG = 1                                                       CLS01370
       GO TO 500                                                        CLS01380
   14  CONTINUE                                                         CLS01390
      II = 0                                                             CLS01400
C                                                                       CLS01410
      IPTS = NSAMP                                                       CLS01420
      IF (IPTS .GT. 110) IPTS = 110                                      CLS01430
      IF (NSAMP .LE. 110) GO TO 15                                       CLS01440
      IPD = NSAMP - 110                                                  CLS01450
   15 CONTINUE                                                          CLS01460
      DO 300 I=LINSTR,LINEND,LININC                                      CLS01470
      II = II + 1                                                        CLS01480
      PIXADR = DRUMAD + (II-1)*NSAMP                                     CLS01490
      CALL RREAD(PIXADR,IR,NSAMP,ISTAT)                                  CLS01500
   20 IF (ISTAT .EQ. 1) GO TO 20                                         CLS01510
C                                                                       CLS01520
CC    DRUM ADDRESS MAPADR                                               CLS01530
C                                                                       CLS01540
      DO 30  J=1,NSAMP                                                   CLS01550
      L   = IR(J)                                                        CLS01560
      IF (L .NE. 0) GO TO 25                                             CLS01570
      IDATA(J) = 0                                                       CLS01580
```

```
FILE: CLSMAP   FORTRAN  A

      GO TO 30                                                          CLS01590
   25 OUT(J) = CLSSYM(L)                                                CLS01600
   30 IDATA(J) = COLORS(L)                                              CLS01610
C                                                                       CLS01620
CCC   WRITE IDATA OUT TO TAPE                                           CLS01630
C                                                                       CLS01640
      CALL WRTLN(IDATA,LSTLIN)                                          CLS01650
C                                                                       CLS01660
CCC   WRITE REMAINDER OF PIXELS ON DRUM FOR SUBSEQUENCE WRITING         CLS01670
C                                                                       CLS01680
      MAPDRM = MAPADR + (II-1)*IPD                                      CLS01690
      CALL RWRITE(MAPDRM,OUT(111),IPD,ISTAT)                            CLS01700
   50 WRITE(6,60)I,(OUT(IK),IK=1,IPTS)                                  CLS01710
   60 FORMAT(2X,I5,2X,110A1)                                            CLS01720
  300 CONTINUE                                                          CLS01730
      XSIZ = NSAMP                                                      CLS01740
      CH = 1                                                            CLS01750
      DO 80 I = 1,NOSUB2                                                CLS01760
      RELCLR(I) = COLORS(I)                                             CLS01770
   80 CONTINUE                                                          CLS01780
      CALL CLRKEY(XSIZ,IDATA,NOSUB2,CH,RELCLR,CH)                       CLS01790
      ISTART = 1                                                        CLS01800
      IPD1 = IPD                                                        CLS01810
  305 II = 0                                                            CLS01820
      IPTS = IPTS + IPTS                                                CLS01830
      IF (IPTS .GE. NSAMP) GO TO 360                                    CLS01840
      IF (IPD1 .GT. 110) IPTS = 110                                     CLS01850
      IF ( IPD1 .LE. 110) IPTS = IPD1                                   CLS01860
      IENDS = ISTART + IPTS - 1                                         CLS01870
      IPD1= IPD1- IPTS                                                  CLS01880
C                                                                       CLS01890
CCC   PRINT REST OF MAP                                                 CLS01900
C                                                                       CLS01910
      IPFLAG = 2                                                        CLS01920
      GO TO 500                                                         CLS01930
  308  CONTINUE                                                         CLS01940
C                                                                       CLS01950
      DO 350 I=LINSTR,LINEND,LININC                                     CLS01960
      II =II + 1                                                        CLS01970
      MAPDRM =  MAPADR + (II-1)*IPD                                     CLS01980
      CALL RREAD(MAPDRM,OUT,IPD,ISTAT1)                                 CLS01990
  310 IF (ISTAT1 .EQ. 1) GO TO 310                                      CLS02000
  350 WRITE(6,240)I,(OUT(IK),IK=ISTART,IENDS)                           CLS02010
  240 FORMAT(2X,I6,1X,110A1)                                            CLS02020
      ISTART = IENDS + 1                                                CLS02030
      GO TO 305                                                         CLS02040
C                                                                       CLS02050
CCC   FINISHED                                                          CLS02060
C                                                                       CLS02070
  360 CONTINUE                                                          CLS02080
      WRITE(6,370)                                                      CLS02090
  370 FORMAT(1H1,'                    NEW ORDERING AND COLOR KEY CODES') CLS02100
      WRITE(6,375)                                                      CLS02110
  375 FORMAT(/// 23X,' OLD       NEW CAT     COLOR ORDER     COLORS'//)  CLS02120
      DO 11 I = 1,NOSUB2                                                CLS02130
      WRITE(6,380) I,CATSUB(I),DELETE(I),COLORS(I)                      CLS02140
  380 FORMAT(20X,4(I6,6X))                                              CLS02150
   11 CONTINUE                                                          CLS02160
      CALL SETMRG(68,4,62)                                              CLS02170
      RETURN                                                            CLS02180
C                                                                       CLS02190
  500  CONTINUE                                                         CLS02200
      J = 0                                                             CLS02210
       DO 81 IJ = ISTRT, IEND, SAMINC                                   CLS02220
      J = J +1                                                          CLS02230
      COL(1,J) = IJ/100                                                 CLS02240
      COL(2,J) = MOD(IJ,100)/10                                         CLS02250
      COL(3,J) = MOD(IJ,10)                                             CLS02260
       IF (J .EQ. 110) GO TO 82                                         CLS02270
   81  CONTINUE                                                         CLS02280
   82  SAMEN = IJ                                                       CLS02290
      ISTRT = SAMEN + SAMINC                                            CLS02300
      JPTS = J                                                          CLS02310
      WRITE(6,95)                                                       CLS02320
       DO 93 IJ = 1, 3                                                  CLS02330
   93  WRITE (6, 90) (COL (IJ, J), J = 1, JPTS)                         CLS02340
   90 FORMAT(9X,110I1)                                                  CLS02350
      WRITE(6,95)                                                       CLS02360
   95 FORMAT(/)                                                         CLS02370
```

FILE: CLSMAP FORTRAN A

```
      IF (IPFLAG .EQ. 1) GO TO 14                                    CLS02380
      IF (IPFLAG .EQ. 2) GO TO 308                                   CLS02390
      END                                                            CLS02400
```

FILE: CNDMAP FORTRAN A

```
      SUBROUTINE CNDMAP(DOTS,CNDSUB,CATVEC)                                  CND00010
C                                                                            CND00020
CC    FLAGS   THE CONDITIONAL CLUSTERS                                       CND00030
C                                                                            CND00040
      IMPLICIT INTEGER (A-Z)                                                 CND00050
      DIMENSION CNDSUB(60),DISTNC(250),DOTS(SIZE,TOTDT2),CATVEC(1)           CND00060
      REAL DSTN,DISTNC                                                       CND00070
      REAL T                                                                 CND00080
C     INCLUDE COMCR1,LIST                                                    CND00090
CC    INCLUDE CMHF15,LIST                                                    CND00100
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,               COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,         COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),             COM00030
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                  COM00040
     *              GRPCHR(41),GROUPS(124)                                   COM00050
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),
     *             SUBRAY(120),PTR(60),CATPTR(250),CATDOT(500),
     *             DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),
     *             CLSSYM(62),STADMS,MEANAD,TABADR,MAPADR,SUNCOR(30),
     *        ODOTUN,ODOTFI,MANSTA,MANOUT,OSPUNT,OSPFIL,OSPKEY,PRNSTS,
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)
     *    ,TOTDT3,FLDADR,VTXADR
CSEND                                                                        CND00120
C                                                                            CND00130
CC    FIND ALL CONDITIONAL CLUSTERS                                          CND00140
C                                                                            CND00150
      NEXT = 63                                                              CND00160
      DO 100 I=1,NOSUB2                                                      CND00170
  100 CNDSUB(I) = CATVEC(I)                                                  CND00180
      DO 200 I=1,NOSUB2                                                      CND00190
      TAB1 = TABADR + (I-1)*TOTDT2                                           CND00200
C                                                                            CND00210
      CALL RREAD(TAB1,DISTNC,TOTDT2,ISTAT)                                   CND00220
  110 IF (ISTAT .EQ. 1) GO TO 110                                            CND00230
C                                                                            CND00240
      DSTN = 255.                                                            CND00250
      LABEL = CATVEC(I)                                                      CND00260
      DO 120 J=1,TOTDT2                                                      CND00270
      IF (DSTN .LT. DISTNC(J)) GO TO 120                                     CND00280
      IF (DOTS(4,J) .NE. LABEL) GO TO 120                                    CND00290
      DSTN = DISTNC(J)                                                       CND00300
  120 CONTINUE                                                               CND00310
C                                                                            CND00320
CC    COMPARE THRESHOLD VALUE T                                              CND00330
C                                                                            CND00340
      IF (DSTN .LE. T) GO TO 200                                             CND00350
C                                                                            CND00360
CC    FLAG AS CONDITIONAL                                                    CND00370
C                                                                            CND00380
      NEXT = NEXT - 1                                                        CND00390
      CNDSUB(I) = NEXT                                                       CND00400
C                                                                            CND00410
  200 CONTINUE                                                               CND00420
C                                                                            CND00430
      RETURN                                                                 CND00440
      END                                                                    CND00450
```

FILE: CRDSCN FORTRAN A

```
      FUNCTION CRDSCN(CARD,GRPDEX,GRPNAM,GROUPS,NOGRP,GRPTR)                    CRD00010
C                                                                               CRD00020
      IMPLICIT INTEGER (A-H,O-Z)                                                CRD00030
      DIMENSION BUF(8), CARD(62), COMVEC(2), NUMVEC(30)                         CRD00040
      DIMENSION GRPDEX(1),GRPNAM(1),GROUPS(1)                                   CRD00050
      LOGICAL*1 L1(4),L2(32)                                                    CRD00060
      EQUIVALENCE (WRD1,L1(1)),(L2(1),BUF(1))                                   CRD00070
C                                                                               CRD00080
      DATA  BLANK/' '/, COMMA/','/, COMVEC/1,','/                               CRD00090
C                                                                               CRD00100
CCCC  ------------------------------------------------------------------------  CRD00110
      ------------------------------------------------------------------------  CRD00120
C                                                                               CRD00130
      COL = 0                                                                   CRD00140
      J = NXTCHR(CARD,COL)                                                      CRD00150
      IF (J.EQ.BLANK) GO TO 110                                                 CRD00160
      DO 10 I=1,8                                                               CRD00170
      J2 = CARD(COL-1+I)                                                        CRD00180
      IF (J2.EQ.COMMA) GO TO 20                                                 CRD00190
   10 BUF(I) = J2                                                               CRD00200
      GO TO 40                                                                  CRD00210
C                                                                               CRD00220
   20 DO 30 J=I,8                                                               CRD00230
   30 BUF(J) = BLANK                                                            CRD00240
C                                                                               CRD00250
   40 N = 1                                                                     CRD00260
      DO 50 I=1,4                                                               CRD00270
      L1(I) = L2(N)                                                             CRD00280
      N = N + 4                                                                 CRD00290
   50 CONTINUE                                                                  CRD00300
      GRPNAM(NOGRP+1) = WRD1                                                    CRD00310
      J = FIND12(CARD,COL,COMVEC)                                               CRD00320
      IF (J.LE.0) GO TO 110                                                     CRD00330
      J = NUMBER(CARD,COL,NUMVEC,0)                                             CRD00340
      II = 0                                                                    CRD00350
      DO 90 I=1,J                                                               CRD00360
      JJ = NUMVEC(I)                                                            CRD00370
   80 II = II+1                                                                 CRD00380
      NUMVEC(II) = JJ                                                           CRD00390
   90 CONTINUE                                                                  CRD00400
      IF (II.LE.0) GO TO 110                                                    CRD00410
C                                                                               CRD00420
      NOGRP = NOGRP+1                                                           CRD00430
      GRPTR = GRPTR +1                                                          CRD00440
      GRPDEX(NOGRP) = GRPTR                                                     CRD00450
      GROUPS(GRPTR) = II                                                        CRD00460
      DO 100 I=1,II                                                             CRD00470
  100 GROUPS(GRPTR+I) = NUMVEC(I)                                               CRD00480
      GRPTR = GRPTR+II                                                          CRD00490
      CRDSCN = 0                                                                CRD00500
      RETURN                                                                    CRD00510
C                                                                               CRD00520
C                                                                               CRD00530
  110 RETURN                                                                    CRD00540
      END                                                                       CRD00550
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: DOTDST FORTRAN A

```
      SUBROUTINE DOTDST(MEANS,DOTS,TABLE,TOP)                                   DOT00010
C     COMPUTES  L1 OR L2 DISTANCES AND STORE ON DRUM                            DOT00020
C                                                                               DOT00030
      IMPLICIT INTEGER (A-Z)                                                    DOT00040
C     INCLUDE COMMK1.LIST                                                       DOT00050
C     INCLUDE COMMK4.LIST                                                       DOT00060
C     INCLUDE COMMK6.LIST                                                       DOT00070
C     INCLUDE COMMK15.LIST                                                      DOT00080
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFLD2,                  DOT00090
     *              AVAR2,COVAR2,CLSID2,SURNO2,SURDS2,FLDSV2,VERTX2,            DOT00100
     *              FETVC2(30),SURVC2(75),SUMPTR(75),CLSVC2(60),                DOT00110
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                     DOT00120
     *              GRPCMK(61),GROUPS(124)                                      DOT00130
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                            DOT00140
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                         DOT00150
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                       DOT00160
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,                 DOT00170
     *              HISFIL,HISKEY,THFORM,EMPTP,EMPKEY,MAPUNT,NOFILE,            DOT00180
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                        DOT00190
     *   ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                           DOT00200
     *   ,DOTUNT,DOTFIL,MCHPAS,TRNSFL,BMTHFL,HISTFL,PCHUNT,                     DOT00210
     *   CRDUNT,PRTUNT,HANDIO                                                   DOT00220
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),         DOT00230
     *             SURRAY(120),PTR(60),CATPTR(250),CATDOT(500),                 DOT00240
     *             DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,              DOT00250
     *   SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,                  DOT00260
     *    OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),                   DOT00270
     *             CLSSYM(62),STANKS,MEANAD,TABADR,MAPADR,SUNCOR(30),           DOT00280
     *        DDOTUN,DDOTFI,MANSTA,MANDOT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,          DOT00290
     *   PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                          DOT00300
     *     ,TOTDT3,FLDADR,VTXADR                                                DOT00310
CSEND                                                                           DOT00320
      DIMENSION DOTS(SIZE,1)                                                    DOT00330
      REAL TABLE(TOTDT2,1)                                                      DOT00340
      DATA CLST2/'CLST'/,BLANK/' '/                                             DOT00350
      REAL MEANS(NOFET2,1),DSTN,DISTNC(250),SUN,SUNCOR                          DOT00360
C                                                                               DOT00370
C     TABADR - DRUM ADDRESS FOR STORING DISTANCE TABLE                          DOT00380
C                                                                               DOT00390
      TAB1 = TABADR                                                             DOT00400
C                                                                               DOT00410
C     RETRIEVE SUN ANGLE  CORRECTIONS                                           DOT00420
C                                                                               DOT00430
      IF (SUNANG .EQ. 0) GO TO 19                                               DOT00440
C                                                                               DOT00450
C     SWTCH = 1 - SUNANGLES ARE USER INPUT                                      DOT00460
C           = 0 -SUN ANGLES ARE ON DOT FILE                                     DOT00470
C                                                                               DOT00480
      IF (SUNANG .NE. 1) GO TO 18                                               DOT00490
      SWTCH = 0                                                                 DOT00500
      CALL SUNFAC(SUNCOR,ANGLE,FETVEC,NOFEAT,SWTCH,DUMMY)                       DOT00510
      GO TO 19                                                                  DOT00520
   18 SWTCH = 1                                                                 DOT00530
      CALL SUNFAC(SUNCOR,ANGLES,FETVEC,NOFEAT,SWTCH,DUMMY)                      DOT00540
   19 CONTINUE                                                                  DOT00550
      DO 200 I=1,NOSUB2                                                         DOT00560
C                                                                               DOT00570
C     ZERO OUT ARRAY                                                            DOT00580
C                                                                               DOT00590
      DO 20   IJ=1,TOTDT2                                                       DOT00600
   20 DISTNC(IJ) = 0                                                            DOT00610
C                                                                               DOT00620
C     COMPUTE DISTANCE BETWEEN ALL DOTS FOR EACH CLUSTER                        DOT00630
C                                                                               DOT00640
C                                                                               DOT00650
C     DIST = 1  -- L1 DISTANCE                                                  DOT00660
C          = 2  -- L2 DISTANCE                                                  DOT00670
C                                                                               DOT00680
      DO 100 J=1,TOTDT2                                                         DOT00690
      DSTN = 0                                                                  DOT00700
      DO 50  K=1,NOFET2                                                         DOT00710
      SUN = SUNCOR(K)                                                           DOT00720
      DSTN = SUN*DOTS(4+K,J) - SUN*MEANS(K,I)                                   DOT00730
      GO TO (30,40),DIST                                                        DOT00740
C                                                                               DOT00750
   30 DISTNC(J) = DISTNC(J) + ABS(DSTN)                                         DOT00760
      GO TO 50                                                                  DOT00770
C                                                                               DOT00780
```

FILE: DOTDST FORTRAN A

```
   40 DISTNC(J) = DISTNC(J) + DSTN**2                                    DOT00800
   50 CONTINUE                                                           DOT00810
C                                                                        DOT00820
  100 IF (DIST .EQ. 2) DISTNC(J) = SQRT(DISTNC(J))                       DOT00830
      CALL PWRITE(TAB1,DISTNC,TOTDT2,ISTAT)                              DOT00840
      TAB1 = TAB1 + TOTDT2                                               DOT00850
  110 IF (ISTAT .EQ. 1) GO TO 110                                        DOT00860
  200 CONTINUE                                                           DOT00870
C                                                                        DOT00880
      WRITE(6,HEAD)                                                      DOT00890
      WRITE(6,250)                                                       DOT00900
  250 FORMAT(//T50,'CLUSTER-DOT INTER-DISTANCE TABLE')                   DOT00910
C                                                                        DOT00920
      SUB2 = NOSUB2                                                      DOT00930
      TAB1 = TABADR                                                      DOT00940
      NSUB2 = 1                                                          DOT00950
      NOGRPS = TOP/TOTDT2                                                DOT00960
      IF (NOGRPS .GT. 15) NOGRPS = 15                                    DOT00970
  245 IF (NOGRPS.GT. SUB2) NOGRPS = SUB2                                 DOT00980
      TOTWDS = TOTDT2 * NOGRPS                                           DOT00990
C                                                                        DOT01000
      CALL PREAD(TAB1,TABLE,TOTWDS,ISTAT)                                DOT01010
      TSUB2 = NSUB2 + NOGRPS - 1                                         DOT01020
      WRITE(6,260)(CLSTR ,I=1,NOGRPS)                                    DOT01030
      WRITE(6,261)(BLANK,SUBVC2(K),K=NSUB2,TSUB2)                        DOT01040
  260 FORMAT(//8X,15(4X,A4))                                             DOT01050
  261 FORMAT(4X,'DOTS',3X,15(A1,'(',I2,')',3X))                          DOT01060
      WRITE(6,262)                                                       DOT01070
  262 FORMAT( )                                                          DOT01080
  265 IF (ISTAT .EQ. 1) GO TO 265                                        DOT01090
      DO 290 I=1,TOTDT2                                                  DOT01100
      WRITE(6,275)DOTVEC(I),(TABLE(I,J),J=1,NOGRPS)                      DOT01110
  275 FORMAT(4X,I3,1H.,15(1X,F7.2))                                      DOT01120
  290 CONTINUE                                                           DOT01130
      TAB1 = TAB1 + TOTWDS                                               DOT01140
      NSUB2 = TSUB2 + 1                                                  DOT01150
      SUB2 = SUB2 - NOGRPS                                               DOT01160
      IF (SUB2 .GT. 0) GO TO 245                                         DOT01170
C                                                                        DOT01180
      RETURN                                                             DOT01190
      END                                                                DOT01200
```

FILE: DSPTAP FORTRAN A

```
      SUBROUTINE DSPTAP(SURNO,SURDES,FLDSAV,VERTX,CATVEC,SUBVEC,MEANS,    DSP00010
     *   COVAR,TOP,DATA,NOFLD,TOTVRT)                                     DSP00020
      IMPLICIT INTEGER (A-Z)                                              DSP00030
C     INCLUDE COMRK1.LIST                                                 DSP00040
C     INCLUDE COMRK4.LIST                                                 DSP00050
C     INCLUDE COMRK6.LIST                                                 DSP00060
C     INCLUDE COMRK15.LIST                                                DSP00070
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,            DSP00080
     *              AVAR2,COVAR2,CLSID2,SURNO2,SURDS2,FLDSV2,VERTX2,      DSP00090
     *              FETVC2(30),SURVC2(75),SUBPTR(75),CLSVC2(60),          DSP00100
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),               DSP00110
     *              GRPCHK(61),GROUPS(124)                                DSP00120
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                      DSP00130
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                   DSP00140
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))               DSP00150
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,           DSP00160
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,     DSP00170
     *    DRUMAD,DRUMDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  DSP00180
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                      DSP00190
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                DSP00200
     *  ERDUNT,PRTUNT,RANDIO                                              DSP00210
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),   DSP00220
     *             SURRAY(120),PTR(60),CATPTR(250),CATDOT(500),           DSP00230
     *             DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,        DSP00240
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,              DSP00250
     *  OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),               DSP00260
     *             CLSSYM(62),STADRS,MEANAD,TABADR,MAPADR,SUNCOR(30) ,    DSP00270
     *       ODOTUN,ODOTFI,MANSTA,MANDOT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,     DSP00280
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                     DSP00290
     *    ,TOTDT3,FLDADR,VTXADR                                           DSP00300
C*END                                                                     DSP00310
      DIMENSION SURNO(1),SURDES(1),FLDSAV(4,1),VERTX(2,1),CATVEC(1)       DSP00320
      DIMENSION SUBVEC(1),DATA(1),IR(1000),NEWSUB(62)                     DSP00330
      DIMENSION CLSVEC(60),DUMMY(100)                                     DSP00340
      REAL MEANS(NOFET2,1),COVAR(VARSZ2,1),VR(1000)                       DSP00350
C                                                                         DSP00360
C      POSITION TAPE                                                      DSP00370
C                                                                         DSP00380
      CALL FSBSFL(DSPUNT,DSPFIL,IST) .                                    DSP00390
C                                                                         DSP00400
C     RUN HEADER RECORD NO. 1                                             DSP00410
C                                                                         DSP00420
      DO 10 I=1,100                                                       DSP00430
   10 DUMMY(I)=I                                                          DSP00440
      NOFLD2 = NOFLD                                                      DSP00450
      TOTVT2 = TOTVRT                                                     DSP00460
      NCAT = 0                                                            DSP00470
C****  JULY 12 1978  NOCAT USED INSTEAD OF NCAT IN MAPTAP FIRST RECORD    DSP00480
C                                                                         DSP00490
      WRITE(DSPUNT)(DATE(I),I=1,2),(DUMMY(I),I=1,3),NOCAT,NOFLD2,NOSUB2,  DSP00500
     *   NOFET2,TOTVT2,NOCAT,VARSZ2,(FETVC2(I),I=1,NOFET2)                DSP00510
C                                                                         DSP00520
      K = 0                                                               DSP00530
      DO 40  I=1,NOCAT                                                    DSP00540
      III = SUBNO(I)                                                      DSP00550
      DO 40   J=1,III                                                     DSP00560
      K = K + 1                                                           DSP00570
   40 CLSVEC(K) = I                                                       DSP00580
C     RUN HEADER RECORD NO. 2                                             DSP00590
      WRITE(DSPUNT)(CATNAM(I),I=1,NOCAT),(CATNAM(I),I=1,NOCAT),           DSP00600
     *    (SUBNO(I),I=1,NOCAT),(SUBDES                                   DSP00610
     *    (I),I=1,NOSUB2),((FLDSAV(I,J),I=1,4),J=1,NOFLD2),((VERTX(I,J)  DSP00620
     *,I=1,2),J=1,TOTVT2),(CLSVEC(I),I=1,NOSUB2),(CLSVEC(I),I=1,NOSUB2)  DSP00630
     *    ,(DUMMY(I),I=1,NOCAT),(KEPPTS(I),I=1,NOSUB2)                   DSP00640
C                                                                         DSP00650
C     RUN HEADER RECORD NO. 3                                             DSP00660
C                                                                         DSP00670
      MEAN1 = STADRS + VARSZ2 * NOSUB2                                    DSP00680
      DO 100 J=1,NOSUB2                                                   DSP00690
      KK = SUBVEC(J)                                                      DSP00700
      MEANS2 = MEAN1 + NOFET2*(KK-1)                                      DSP00710
      COVAR1 = STADRS + VARSZ2*(KK-1)                                     DSP00720
      CALL DREAD(COVAR1,COVAR(1,J),VARSZ2,ISTAT)                          DSP00730
   50 IF (ISTAT .EQ. 1)  GO TO 50                                         DSP00740
      CALL DREAD(MEANS2,MEANS(1,J),NOFET2,ISTAT1)                         DSP00750
   60 IF (ISTAT1 .EQ. 1) GO TO 60                                         DSP00760
C                                                                         DSP00770
  100 CONTINUE                                                            DSP00780
C                                                                         DSP00790
```

FILE: DSPTAP FORTRAN A

```
      WRITE(DSPUNT)((COVAR(I,J),I=1,VARSZ2),J=1,NOSUB2),((MEANS(I,J),   DSP00800
     *     I=1,NOFET2),J=1,NOSUB2)                                       DSP00810
C                                                                        DSP00820
CC    RUN HEADER RECORD NO. 4                                            DSP00830
C                                                                        DSP00840
      WRITE(DSPUNT)((COVAR(I,J),I=1,VARSZ2),J=1,NOSUB2),                 DSP00850
     *  (DUMMY(I),I=1,NOSUB2),(DUMMY(I),I=1,NOSUB2)                     DSP00860
C                                                                        DSP00870
CC    FIELD RECORD                                                       DSP00880
C                                                                        DSP00890
      LINSTR = FLDINF(1)                                                 DSP00900
      LINEND = FLDINF(2)                                                 DSP00910
      LININC = FLDINF(3)                                                 DSP00920
      SAMSTR = FLDINF(4)                                                 DSP00930
      SAMEND = FLDINF(5)                                                 DSP00940
      SAMINC = FLDINF(6)                                                 DSP00950
      PTS = (SAMEND-SAMSTR)/SAMINC + 1                                   DSP00960
      LINES = (LINEND-LINSTR)/LININC + 1                                 DSP00970
C                                                                        DSP00980
      WRITE(DSPUNT)(FLDINF(I),I=1,6),PTS,LINES,FLDNAM,NOVRT,             DSP00990
     *   (VERTEX(I),I=1,NOVRT),(VERTEX(I+NOVRT),I=1,NOVRT)               DSP01000
C                                                                        DSP01010
CC                                                                       DSP01020
CC    NEWSUB -- NEW SUBCLASS NUMBERS                                     DSP01030
C                                                                        DSP01040
      DO 120 I=1,NOSUB2                                                  DSP01050
      K = SUBVEC(I)                                                      DSP01060
  120 NEWSUB(K) = I                                                      DSP01070
      DO 130 I=1,PTS                                                     DSP01080
  130 VR(I) = 0.0                                                        DSP01090
      MAP = DMUMAP                                                       DSP01100
      ILINE = LINES                                                      DSP01110
      NOLINE = TOP/PTS                                                   DSP01120
  135 IF (NOLINE .GT. ILINE)  NOLINE = ILINE                             DSP01130
      TOTPIX = NOLINE * PTS                                              DSP01140
      CALL MREAD(MAP,DATA,TOTPIX,ISTAT2)                                 DSP01150
      MAP = MAP + TOTPIX                                                 DSP01160
  137 IF (ISTAT2 .EQ. 1) GO TO 137                                       DSP01170
      DO 150  I=1,NOLINE                                                 DSP01180
      II = II + 1                                                        DSP01190
      N = LINSTR + LININC*(II-1)                                         DSP01200
C                                                                        DSP01210
      DO 140 J=1,PTS                                                     DSP01220
      IDUM=(I-1)*PTS+J                                                   DSP01230
      JJ = DATA(IDUM)                                                    DSP01240
  140 IR(J) = NEWSUB(JJ)                                                 DSP01250
C                                                                        DSP01260
      WRITE(DSPUNT)N,(IR(K),K=1,PTS),(VR(K),K=1,PTS)                     DSP01270
  150 CONTINUE                                                           DSP01280
      ILINE = ILINE - NOLINE                                             DSP01290
      IF (ILINE .LE. 0) GO TO 155                                        DSP01300
      GO TO 135                                                          DSP01310
  155 N = 0                                                              DSP01320
      WRITE(DSPUNT)N,(IR(I),I=1,PTS),(VR(I),I=1,PTS)                     DSP01330
      PTS = 0                                                            DSP01340
      WRITE(DSPUNT)(FLDINF(I),I=1,6),PTS,LINES,FLDNAM,NOVRT,             DSP01350
     *   (VERTEX(I),I=1,NOVRT),(VERTEX(I+NOVRT),I=1,NOVRT)               DSP01360
      ENDFILE DSPUNT                                                     DSP01370
C                                                                        DSP01380
      RETURN                                                             DSP01390
      END                                                                DSP01400
```

FILE: FILERD FORTRAN A

```
      SUBROUTINE FILERD(ARRAY,TOP,NOFLD,TOTVRT,FLDSAV,VERTX)                    FIL00010
C                                                                               FIL00020
CCC   READS IN ALL NEEDED FILES                                                 FIL00030
CCC                                                                             FIL00040
CCC                                                                             FIL00050
CCC   DRUM ADDRESSES                                                            FIL00060
CCC                                                                             FIL00070
CCC   DRUMAD - BEGINNING ADDRESS FOR MAPFIL                                     FIL00080
CCC   STADRS - BEGINNING ADDRESS FOR COVAR AND MEANS                            FIL00090
CCC   TABADR - BEGINNING ADDRESS FOR DISTANCE TABLE                             FIL00100
CCC   MAPADR - BEGINNING ADDRESS FOR NSAMP-110 PTS OF COND. OR                  FIL00110
CCC            MIXED CLUSTER MAP                                                FIL00120
CCC   FLDADR - BEGINNING ADDRESS FOR FIELD INFO                                 FIL00130
CCC   VTXADR - BEGINNING ADDRESS FOR VERTICES                                   FIL00140
CCC   COVAR2 - BEGINNING ADDRES FOR DOT DATA                                    FIL00150
C                                                                               FIL00160
      IMPLICIT INTEGER (A-Z)                                                    FIL00170
       LIMIT = 5000                                                             FIL00180
C     INCLUDE COMBK1.LIST                                                       FIL00190
C     INCLUDE COMBKA.LIST                                                       FIL00200
C     INCLUDE CMBK15.LIST                                                       FIL00210
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,                  FIL00220
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,            FIL00230
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),                FIL00240
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                     FIL00250
     *              GRPCHK(61),GROUPS(124)                                      FIL00260
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,                 FIL00270
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,           FIL00280
     *    DRUMAD,DRMADS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                        FIL00290
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                            FIL00300
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                      FIL00310
     *  CRDUNT,PRTUNT,HANDIO                                                    FIL00320
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),         FIL00330
     *             SUBRAY(120),PTR(60),CATPTR(250),CATDOT(500),                 FIL00340
     *             DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,              FIL00350
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,                   FIL00360
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),                    FIL00370
     *             CLSSYM(62),STADRS,MEANAD,TABADR,MAPADR,SUNCOR(30),           FIL00380
     *       DDOTUN,DDOTFI,MANSTA,MANDOT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,           FIL00390
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                           FIL00400
     *    ,TOTDT3,FLDADR,VTXADR                                                 FIL00410
C$END                                                                           FIL00420
CCC                                                                             FIL00430
C                                                                               FIL00440
      DIMENSION ARRAY(1)                                                        FIL00450
      DIMENSION FLDSAV(4,1),VERTX (2,1)                                         FIL00460
      DIMENSION FETVC3(30)                                                      FIL00470
C                                                                               FIL00480
      COVAR2 = 1                                                                FIL00490
C                                                                               FIL00500
CCC   READ IN MAPFIL AND STORE ON THE DRUM                                      FIL00510
CCC                                                                             FIL00520
      IF (MAPKEY .EQ. 0) GO TO 100                                              FIL00530
C                                                                               FIL00540
      I = LAREAD(FLDNAM,VERTEX,FLDINF,NOVRT)                                    FIL00550
C                                                                               FIL00560
      NOLINE =(FLDINF(2) - FLDINF(1))/FLDINF(3) + 1                             FIL00570
      NSAMP  =(FLDINF(5) - FLDINF(4))/FLDINF(6) + 1                             FIL00580
      TOTPIX = NOLINE*NSAMP                                                     FIL00590
C                                                                               FIL00600
CC                                                                              FIL00610
      CALL STOMAP(NOLINE,NSAMP,ARRAY,TOP,DRUMAD)                                FIL00620
C                                                                               FIL00630
CCC   READ IN STAT FILE                                                         FIL00640
CCC                                                                             FIL00650
  100 IF (STATKY .EQ. 0 ) GO TO 200                                             FIL00660
C                                                                               FIL00670
      CALL REDSAV(ARRAY,TOP,BMFLG)                                              FIL00680
      STASIZ = (VARSZ2 + NOFET2) * NOSUB2                                       FIL00690
      STADRS = DRUMAD + TOTPIX                                                  FIL00700
      CALL RWRITE(STADRS,ARRAY(COVAR2),STASIZ,ISTAT)                            FIL00710
  110 IF (ISTAT .EQ. 1) GO TO 110                                               FIL00720
CCC                                                                             FIL00730
CCC   READ IN DOTFIL                                                            FIL00740
CCC                                                                             FIL00750
  200 IF (DOTKEY .EQ. 0) RETURN                                                 FIL00760
C                                                                               FIL00770
      TOTSTO = TOP - COVAR2                                                     FIL00780
      IF (TOTSTO .GE. LIMIT) GO TO 220                                          FIL00790
```

FILE: FILERD FORTRAN A

```
      WRITE(6,210)                                                          FIL00800
  210 FORMAT(' NOT ENOUGH CORE TO STORE DOTFIL')                            FIL00810
      CALL CMERR                                                            FIL00820
C                                                                           FIL00830
  220 TYPSWT = 3                                                            FIL00840
      CALL ROOOTS(ARRAY(COVAR2),DOTVEC,TOTDT3,TYPSWT,SIZE,TOTDT2,NOCAT,     FIL00850
     1   CATNAM,NOFFAT,FETVEC,NOFET3,FETVC3,NOSUN,ANGLE,NOFLD,              FIL00860
     2  TOTVRT,FLDSAV,VERTX,DUMMY)                                          FIL00870
      CALL WRTFLD(FLDSAV,VERTX,NOFLD,2,CATNAM,DUMMY)                        FIL00880
      TARADR = STADRS + (VARSZ2+NOFEAT)*NOSUB2                              FIL00890
      MAPADR = TARADR + NOSUB2*TOTDT2                                       FIL00900
      TOTAL = MAPADR                                                        FIL00910
      FLDADR = TOTAL                                                        FIL00920
      IF (MIX .EQ. 0 .AND. COND .EQ. 0) GO TO 222                           FIL00930
      FLDADR = TOTAL + (NSAMP-110)*NOLINE                                   FIL00940
      IF ( FLDADR .LT. TOTAL) FLDADR = TOTAL                                FIL00950
      TOTAL = FLDADR                                                        FIL00960
  222 IF (DSPKEY .EQ. 0) GO TO 225                                          FIL00970
      VTXADR = TOTAL + 4*NOFLD                                              FIL00980
      TOTAL = VTXADR + TOTVRT*2                                             FIL00990
C                                                                           FIL01000
  225 CONTINUE                                                              FIL01010
      IF (TOTAL .LE. (DRUMAD+DRMWDS)) GO TO 230                             FIL01020
      WRITE(6,400)                                                          FIL01030
  400 FORMAT(/' NOT ENOUGH CORE DRUM SPACE OF CLUSTER MAP INFO')            FIL01040
  230 CONTINUE                                                              FIL01050
C                                                                           FIL01060
C                                                                           FIL01070
      RETURN                                                                FIL01080
C                                                                           FIL01090
      END                                                                   FIL01100
```

FILE: KNEAR FORTRAN A

```
      SUBROUTINE KNEAR(DOTS,SUBVEC,SUBNO,CATVEC,ITER,TAB1,SWTCH,          KNE00010
     *  CATNUM,CLUNUM,MEANS)                                              KNE00020
C                                                                         KNE00030
C     LABELS BY THE K-NEAREST NEIGHBOR PROCEDURE                          KNE00040
C                                                                         KNE00050
      IMPLICIT INTEGER (A-Z)                                              KNE00060
C                                                                         KNE00070
      REAL DISTNC(250)                                                    KNE00080
      DIMENSION DOTS(SIZE,1),SUBNO(1),SUBVEC(60)                          KNE00090
      DIMENSION CATVEC(1),DOTNAM(60)                                      KNE00100
      DIMENSION CATGRY(60)                                                KNE00110
C     INCLUDE COMBK1,LIST                                                 KNE00120
C     INCLUDE COMBK4,LIST                                                 KNE00130
C     INCLUDE CMBK15,LIST                                                 KNE00140
C     INCLUDE COMBK6,LIST                                                 KNE00150
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,            COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,      COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),          COM00030
     *              KEYPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),               COM00040
     *              GRPCHK(61),GROUPS(124)                                COM00050
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                      COM00010
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                   COM00020
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                 COM00030
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),
     *            SUBRAY(120),PTR(60),CATPTR(250),CATDOT(500),
     *             DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,
     *  SUNANG,T,NEAREST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),
     *             CLSSYM(62),STADRS,MEANAD,TABADR,MAPADR,SUNCOR(30) ,
     *       ODOTUN,ODOTFI,MANSTA,MANDUT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)
     *    ,TOTDT3,FLDADR,VTXADR
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,           COM00010
     *              HISFIL,HISKEY,TRFORM,ERRTP,ERRKEY,MAPUNT,NOFILE,      COM00020
     *    DRUMAD,DRMADS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  COM00030
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                      COM00040
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTHFL,HISTFL,PCHUNT,                COM00050
     *  CRDUNT,PRTUNT,RANDIO                                              COM00060
CSEND                                                                     KNE00170
C                                                                         KNE00180
C     NO OF DOTS TO COMPARE                                               KNE00190
C                                                                         KNE00200
      DIMENSION MEANS(NOFET2,1),TIES(250)                                 KNE00210
      DIMENSION DOTVC2(250)                                               KNE00220
C                                                                         KNE00230
C     SAVE DOTVEC                                                         KNE00240
C                                                                         KNE00250
      DO 5  I=1,TOTDT2                                                    KNE00260
    5 DOTVC2(I) = DOTVEC(I)                                               KNE00270
C                                                                         KNE00280
      DO 500  I=1,ITER                                                    KNE00290
      IF (SWTCH .EQ. 0) WRITE(6,HEAD)                                     KNE00300
      WRITE(6,10)NEAREST                                                  KNE00310
   10 FORMAT(/T43,'LABELING BY',I3,'-NEAREST NEIGHBOR PROCEDURE'/)         KNE00320
      KNGHBR = NEAREST                                                    KNE00330
C                                                                         KNE00340
C     READ IN DISTANCES FROM DRUM ONE CLUSTER AT A TIME                   KNE00350
C                                                                         KNE00360
      TIE = 0                                                             KNE00370
      II = I                                                              KNE00380
      CALL RREAD(TAB1,DISTNC,TOTDT2,ISTAT)                                KNE00390
   50 IF (ISTAT .EQ. 1) GO TO 50                                          KNE00400
      TAB1 = TAB1 + TOTDT2                                                KNE00410
C                                                                         KNE00420
C     SORT DISTANCES IN ASCENDING ORDER                                   KNE00430
C                                                                         KNE00440
      DO 55  J=1,TOTDT2                                                   KNE00450
      DOTVEC(J) = DOTVC2(J)                                               KNE00460
   55 DOTNAM(J) = DOTS(4,J)                                               KNE00470
      CALL ASCEND(DISTNC,TOTDT2,DOTNAM,DOTVEC)                            KNE00480
C                                                                         KNE00490
C     REINITIALIZE ARRAYS                                                 KNE00500
C                                                                         KNE00510
   57 DO 60  J=1,NOCAT                                                    KNE00520
   60 CATGRY(J) = 0                                                       KNE00530
      MAX = 0                                                             KNE00540
      DO 70 JJ=1,KNGHBR                                                   KNE00550
      L = DOTNAM(JJ)                                                      KNE00560
C                                                                         KNE00570
```

FILE: KNEAR FORTRAN A

```
C       RETRIEVE CATEGORY NO.                                                   KNE00580
C                                                                               KNE00590
        CATGRY(L) = CATGRY(L) + 1                                               KNE00600
        IF(CATGRY(L) .LE. MAX) GO TO  70                                        KNE00610
        MAX = CATGRY(L)                                                         KNE00620
        CATNUM = L                                                              KNE00630
   70 CONTINUE                                                                  KNE00640
C                                                                               KNE00650
CC      CHECK FOR A TIE                                                         KNE00660
C                                                                               KNE00670
        IF (KNGHBR .EQ. 1) GO TO 100                                            KNE00680
C                                                                               KNE00690
        DO 80 III=1,NOCAT                                                       KNE00700
        IF (III .EQ. CATNUM) GO TO 80                                           KNE00710
        IF (MAX .EQ. CATGRY(III)) GO TO 90                                      KNE00720
   80 CONTINUE                                                                  KNE00730
C                                                                               KNE00740
CC      NO TIES OCCURRED                                                        KNE00750
C                                                                               KNE00760
        GO TO 100                                                               KNE00770
C                                                                               KNE00780
CC      A TIE  OCCURRED - DECREASE K-DOTS BY 1 AND REPEAT PROCESS               KNE00790
C                                                                               KNE00800
   90 KNGHBR = KNGHBR - 1                                                       KNE00810
        TIE = TIE + 1                                                           KNE00820
        TIES(TIE) = KNGHBR + 1                                                  KNE00830
        GO TO 57                                                                KNE00840
C                                                                               KNE00850
C       ASSIGN CLUSTER TO CATEGORY                                              KNE00860
  100 IF (SWTCH .EQ. 1) II = CLUNUM                                             KNE00870
        CATVEC(II) = CATNUM                                                     KNE00880
C                                                                               KNE00890
CC      PRINT CLUSTER INFORMATION                                               KNE00900
C                                                                               KNE00910
        WRITE(6,110)CATNAM(CATNUM)                                              KNE00920
  110 FORMAT(/1X,'LABEL : ',A4)                                                 KNE00930
        FEATR = NOFET2                                                          KNE00940
        ISTART = 1                                                              KNE00950
        IEND = 15                                                               KNE00960
  115 IF (IEND .GT. FEATR) IEND = FEATR                                         KNE00970
        IENDS = ISTART + IEND - 1                                               KNE00980
        IF (ISTART .NE. 1) GO TO 117                                            KNE00990
        WRITE(6,120)(MEANS(J,II),J=ISTART,IENDS)                                KNE01000
        GO TO 122                                                               KNE01010
  117 WRITE(6,118)(MEANS(J,II),J=ISTART,IENDS)                                  KNE01020
  118 FORMAT(/9X,15(F6.2,2X))                                                   KNE01030
  120 FORMAT(/1X,'MEANS : ',15(F6.2,2X))                                        KNE01040
C                                                                               KNE01050
  122 CONTINUE                                                                  KNE01060
        FEATR = FEATR - IEND                                                    KNE01070
        ISTART = IENDS + 1                                                      KNE01080
        IF (FEATR .GT. 0) GO TO 115                                             KNE01090
C                                                                               KNE01100
        WRITE(6,125)                                                            KNE01110
  125 FORMAT(/)                                                                 KNE01120
        WRITE(6,130)                                                            KNE01130
  130 FORMAT(40X,'DOTS',9X,'ANALYST LABEL',11X,'DISTANCE'/)                     KNE01140
        DO 150 J=1,KNGHBR                                                       KNE01150
        K = DOTNAM(J)                                                           KNE01160
        WRITE(6,140) DOTVEC(J),CATNAM(K),DISTNC(J)                              KNE01170
  140 FORMAT(40X,I3,1H,,13X,A4,14X,F7.2)                                        KNE01180
  150 CONTINUE                                                                  KNE01190
C                                                                               KNE01200
        WRITE(6,160)                                                            KNE01210
  160 FORMAT(///T60,'TOTALS',/T56,'LABEL',T65,'NO.'/)                           KNE01220
        DO 170 J=1,NOCAT                                                        KNE01230
  170 WRITE(6,180)CATNAM(J),CATGRY(J)                                           KNE01240
  180 FORMAT(T56,A6,T65,I3)                                                     KNE01250
        IF (TIE .EQ. 0) GO TO 490                                               KNE01260
        WRITE(6,185)                                                            KNE01270
  185 FORMAT(/)                                                                 KNE01280
        WRITE(6,190)                                                            KNE01290
  190 FORMAT(T51,'A TIE OCCURRED.'/T51,'THE FOLLOWING DOT(S) WERE DISCAR        KNE01300
     *DED'/)                                                                    KNE01310
        DO 200  JJ=1,TIE                                                       KNE01320
        J = TIES(JJ)                                                            KNE01330
        K = DOTNAM(J)                                                           KNE01340
        WRITE(6,140)DOTVEC(J),CATNAM(K),DISTNC(J)                               KNE01350
  200 CONTINUE                                                                  KNE01360
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: KNEAR FORTRAN A

```
C
  490 CONTINUE                                                          KNE01370
      IF (SWTCH .NE. 1) GO TO 500                                       KNE013A0
C                                                                       KNE01390
C     RESTORE DOTVEC                                                    KNE01400
C                                                                       KNE01410
      DO 210 J=1,TOTOT2                                                 KNE01420
  210 DOTVEC(J) = DOTVC2(J)                                             KNE01430
  500 CONTINUE                                                          KNE01440
      IF (SWTCH .EQ. 1) RETURN                                          KNE01450
C                                                                       KNE01460
C     GROUP LABELED CLUSTER ACCORDING TO CATEGORY                       KNE01470
C                                                                       KNE014A0
      K = 0                                                             KNE01490
      DO 510 I=1,NOCAT                                                  KNE01500
      DO 510 J=1,NOSUB2                                                 KNE01510
C                                                                       KNE01520
      IF (CATVEC(J) .NE. I) GO TO 510                                   KNE01530
      SUBNO(I) = SUBNO(I) + 1                                           KNE01540
      K = K + 1                                                         KNE01550
      SUBVEC(K) = J                                                     KNE01560
C                                                                       KNE01570
  510 CONTINUE                                                          KNE015A0
C                                                                       KNE01590
      RETURN                                                            KNE01600
      END                                                               KNE01610
                                                                        KNE01620
```

FILE: LABDOT FORTRAN A

```
      SUBROUTINE LABDOT(DOTS)                                                   LAB00010
C                                                                               LAB00020
CC    LABDOT UPDATES DOTFIL                                                     LAB00030
C                                                                               LAB00040
      IMPLICIT INTEGER (A-Z)                                                    LAB00050
C                                                                               LAB00060
CC    INCLUDE CMRK1S.LIST                                                       LAB00070
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),         LAB00080
     *            SUBRAY(120),PTR(60),CATPTR(250),CATDOT(500),                 LAB00090
     *             DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,             LAB00100
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,                   LAB00110
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),                    LAB00120
     *            CLSSYM(62),STADRS,MEANAD,TAHADR,MAPADR,SUNCOR(30) ,           LAB00130
     *       OOOTUN,OOOTFI,MANSTA,MANDOT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,           LAB00140
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                           LAB00150
     *    ,TOTDT3,FLDADR,VTXADR                                                 LAB00160
C$END                                                                           LAB00170
      DIMENSION DOTS(SIZE,1) ,CATNO(60)                                         LAB00180
C                                                                               LAB00190
CC    CHECK CATEGORY NAMES FOR NEW ENTRIES                                      LAB00200
C                                                                               LAB00210
      DO 100 I=1,NOCAT2                                                         LAB00220
      DO 90  J=1,NOCAT                                                          LAB00230
      IF (CATNM2(I) .EQ. CATNAM(J))GO TO 95                                     LAB00240
   90 CONTINUE                                                                  LAB00250
      NOCAT = NOCAT + 1                                                         LAB00260
C                                                                               LAB00270
CC    INSERT NEW CATEGORY                                                       LAB00280
C                                                                               LAB00290
      CATNAM(NOCAT) = CATNM2(I)                                                 LAB00300
        CATNO(I) = NOCAT                                                        LAB00310
        GO TO 100                                                               LAB00320
C                                                                               LAB00330
   95 CATNO(I) = J                                                              LAB00340
C                                                                               LAB00350
  100 CONTINUE                                                                  LAB00360
      DO 150 I=1,NOCAT2                                                         LAB00370
C                                                                               LAB00380
CC    RETRIEVE BEGIN. AND END. POINTER                                          LAB00390
CC                                                                              LAB00400
C                                                                               LAB00410
      IB = CATPTR(I) + 1                                                        LAB00420
      IE = IB + CATDOT(IB-1) - 1                                                LAB00430
C                                                                               LAB00440
      DO 120 J=IB,IE                                                            LAB00450
C                                                                               LAB00460
CC    RETRIEVE DOT NO AND CHANGE CATEGORY NO FOR DOT                            LAB00470
C                                                                               LAB00480
      K = CATDOT(J)                                                             LAB00490
  120 DOTS(4,K) = CATNO(I)                                                      LAB00500
C                                                                               LAB00510
  150 CONTINUE                                                                  LAB00520
C                                                                               LAB00530
      RETURN                                                                    LAB00540
      END                                                                       LAB00550
```

```
FILE: LABLR    FORTRAN  A

      SUBROUTINE LABLR(ARRAY,TOP,NOFLD,TOTVRT,FLDSAV,VERTX,MEANS)          LAB00010
C                                                                          LAB00020
      IMPLICIT INTEGER (A-Z)                                               LAB00030
      LIMIT = 3135                                                         LAB00040
C                                                                          LAB00050
CCC   INCLUDE COMBK1.LIST                                                  LAB00060
CCC   INCLUDE COMBK6.LIST                                                  LAB00070
C     INCLUDE CMBK15.LIST                                                  LAB00080
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,             LAB00090
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,       LAB00100
     *             FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),            LAB00110
     *             KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                 LAB00120
     *             GRPCHR(61),GROUPS(124)                                  LAB00130
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,            LAB00140
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NUFILE,      LAB00150
     *    DRUMAD,DRMNOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                   LAB00160
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                       LAB00170
     *  ,DOTUNT,DOTFIL,NCHPAS,TPNSFL,BMTRFL,HISTFL,PCHUNT,                 LAB00180
     *  CRDUNT,PRTUNT,MANDIO                                               LAB00190
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),    LAB00200
     *             SUBRAY(120),PTR(60),CATPTR(250),CATDOT(500),            LAB00210
     *             DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,         LAB00220
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,              LAB00230
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTOT2,FLDINF(6),               LAB00240
     *             CLSSYM(62),STADRS,MEANAD,TABADR,MAPADR,SUNCOR(30),      LAB00250
     *        ODOTUN,ODOTFI,MANSTA,MANDOT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,     LAB00260
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                      LAB00270
     *    ,TOTOT3,FLDADR,VTXADR                                            LAB00280
C$END                                                                      LAB00290
C                                                                          LAB00300
      DIMENSION FLDSAV(4,1),VERTX(2,1),ARRAY(1),SUBVEC(60),SUBNOS(60)      LAB00310
      DIMENSION N(60),CATVEC(60),SUBNAM(60),CNOSUB(60),MIXSUB(60)          LAB00320
      REAL MEANS(NOFET2,1)                                                 LAB00330
      DIMENSION DOTVC2(250)                                                LAB00340
      DIMENSION TABLE(3135)                                                LAB00350
      DATA BLANK/' '/                                                      LAB00360
      SWTCH = 2                                                            LAB00370
CCC                                                                        LAB00380
CCC   MANNUALLY RELABEL STATS                                              LAB00390
C                                                                          LAB00400
      IF (MANSTA .EQ. 0) GO TO 20                                          LAB00410
C                                                                          LAB00420
      CALL MANORD(ARRAY(CLSID2),CLSVC2,SUBVEC,NOCLS2,SUBNOS,NOSUB2)        LAB00430
CCC                                                                        LAB00440
CCC   UPDATE INFO IN ARRAY                                                 LAB00450
CCC                                                                        LAB00460
      CALL REORDER(ARRAY,SUBVEC,N)                                         LAB00470
CCC                                                                        LAB00480
CCC   OUTPUT REORDRED STATS                                                LAB00490
CCC                                                                        LAB00500
      CALL LABMAN(OSAVTP,OSTAFI,NOCLS2,NOSUB2,NOFET2,NOFLD2,TOTVT2,        LAB00510
     *            FETVC2,ARRAY(FLDSV2),ARRAY(VERTX2),ARRAY(CLSID2),        LAB00520
     *  SUBNOS,ARRAY(SUBDS2),N,STADRS,VARSZ2,PUNCH,SUBVEC,PRNSTS,SWTCH)    LAB00530
C                                                                          LAB00540
   20 IF (MANDOT .EQ. 0) GO TO 40                                          LAB00550
CCC                                                                        LAB00560
CCC   UPDATE DOTFIL                                                        LAB00570
CCC                                                                        LAB00580
      CALL LABDOT(ARRAY(COVAR2))                                           LAB00590
CCC                                                                        LAB00600
CCC   OUTPUT UPDATE DOTFIL                                                 LAB00610
CCC                                                                        LAB00620
      CALL WRTDOT(TOTOT2,NOSUN,FLDSAV,VERTX,ANGLE,ARRAY(COVAR2),           LAB00630
     *        NOCAT,CATNAM,SIZE,NOFEAT,FETVEC,TOTVRT,NOFLD,              LAB00640
     *    ODOTUN,ODOTFI)                                                   LAB00650
      IF (PRNDOT.EQ.1) GOTO 798                                            LAB00660
   30 CONTINUE                                                             LAB00670
C                                                                          LAB00680
CCC   EXECUTING A PRODEDURE                                                LAB00690
CCCCC                         PROC = 1 -- K-NEAREST NEIGHBOR               LAB00700
CCCCC                         PROC = 2 -- ALL-OF-A-KIND                    LAB00710
CCCCC                         PROC = 3 -- MANUALLY RELABEL A FILE          LAB00720
CCC                                                                        LAB00730
   40 IF (PROC .EQ. 3) RETURN                                              LAB00740
C                                                                          LAB00750
CCC    HAVE LABELS BEEN DEFINED                                            LAB00760
C                                                                          LAB00770
      IF (NOCAT.EQ.0) WRITE(6,42)                                          LAB00780
   42 FORMAT( //// 25X,'****CATEGORIES HAVE NOT BEEN DEFINED****'//)      LAB00790
```

FILE: LABLR FORTRAN A

```
C                                                                       LAB00800
CC      SET UP DOTVEC ARRAY                                             LAB00810
C                                                                       LAB00820
        IF (TOTDT3 .EQ. 0) GO TO 49                                     LAB00830
        K = 0                                                           LAB00840
        TOTDOT = TOTDT2 + TOTDT3                                        LAB00850
        DO 47  I=1,TOTDOT                                               LAB00860
        DO 45  J=1,TOTDT3                                               LAB00870
        IF (DOTVEC(J) .EQ. I) GO TO 47                                  LAB00880
   45 CONTINUE                                                          LAB00890
        K = K + 1                                                       LAB00900
        DOTVC2 (K) = I                                                  LAB00910
   47   CONTINUE                                                        LAB00920
C                                                                       LAB00930
        DO 48  I=1,TOTDT2                                               LAB00940
   48   DOTVEC(I) = DOTVC2(I)                                           LAB00950
        GO TO 52                                                        LAB00960
C                                                                       LAB00970
   49 DO 51  I=1,TOTDT2                                                 LAB00980
   51 DOTVEC(I) = I                                                     LAB00990
   52 CONTINUE                                                          LAB01000
C                                                                       LAB01010
CC      STORE FIELD INFO AND VERTICES ON DRUM                           LAB01020
C                                                                       LAB01030
        IF (DSPKEY .EQ. 0) GO TO 56                                     LAB01040
        TOTWDS = 4*NOFLD                                                LAB01050
        CALL RWRITE(FLDADR,FLDSAV,TOTWDS,ISTAT)                         LAB01060
   53   IF (ISTAT .EQ. 1) GO TO 53                                      LAB01070
        TOTVTS = TOTVRT*2                                               LAB01080
        CALL RWRITE(VTXADR,VERTX,TOTVTS,ISTAT)                          LAB01090
   54   IF (ISTAT .EQ. 1) GO TO 54                                      LAB01100
   56   CONTINUE                                                        LAB01110
C                                                                       LAB01120
CC      READ MEANS INTO CORE -- USE SPACE FOR FIELD INFO                LAB01130
C                                                                       LAB01140
        MEAN1 = STADRS + VARS72*NOSUB2                                  LAB01150
        TOTWRD = NOFEAT*NOSUB2                                          LAB01160
C                                                                       LAB01170
        CALL RREAD(MEAN1,*MEANS,TOTWRD,ISTAT)                           LAB01180
   50   IF (ISTAT .EQ. 1) GO TO 50                                      LAB01190
C                                                                       LAB01200
CZERO OUT SUBNOS (WILL CONTAIN NO. OF CLUSTERS IN CATEGORY I)           LAB01210
C                                                                       LAB01220
        DO 55 I=1,NOCAT                                                 LAB01230
   55   SUBNOS(I) = 0                                                   LAB01240
C       COMPUTE DISTANCE TABLE                                          LAB01250
C                                                                       LAB01260
        CALL DOTDST(MEANS,ARRAY(COVAR2),TABLE,LIMIT)                    LAB01270
C                                                                       LAB01280
CC      K-NEAREST NEIGHBOR PROCEDURE                                    LAB01290
C                                                                       LAB01300
        IF ( PROC .NE. 1) GO TO 60                                      LAB01310
        ITER = NOSUB2                                                   LAB01320
        TAB1 = TABADR                                                   LAB01330
        SWTCH = 0                                                       LAB01340
        CALL KNEAR(ARRAY(COVAR2),SUBVEC,SUBNOS,CATVEC,ITER,TAB1,SWTCH,  LAB01350
     *    DUMMY,DUMMY,MEANS)                                            LAB01360
        SWTCH = 2                                                       LAB01370
        GO TO 70                                                        LAB01380
C                                                                       LAB01390
CC      ALL-OF-A-KIND                                                   LAB01400
C                                                                       LAB01410
   60   CALL ALLKIN(ARRAY(COVAR2),SUBVEC,SUBNOS,CATVEC,MEANS)          LAB01420
C                                                                       LAB01430
CC      ASSIGN APPROPRIATE NAMES TO CLUSTERS                            LAB01440
C                                                                       LAB01450
   70 CALL NAMSTA(SUBNAM,CATVEC,SUBNOS,NOSUB2,CATNAM,NOCAT)            LAB01460
C                                                                       LAB01470
CC      OUTPUT LABELED STATS                                            LAB01480
C                                                                       LAB01490
        CALL LABMAN(OSAVTP,OSTAFI,NOCAT,NOSUB2,NOFET2,NOFLD2,TOTVT2,    LAB01500
     *          FETVC2,ARRAY(FLDSV2) ,ARRAY(VERTX2),CATNAM,SUBNOS,SUBNAM,LAB01510
     *  KERRTS,STADRS,VARS72,PUNCH,SUBVEC,PRNSTS,SWITCH)                LAB01520
        IF (COND .EQ. 0) GO TO 90                                       LAB01530
C                                                                       LAB01540
CC      FLAG CONDITIONAL CLUSTERS                                       LAB01550
C                                                                       LAB01560
        CALL CNDMAP(ARRAY(COVAR2),CNDSUB,CATVEC)                        LAB01570
                                                                        LAB01580
```

```
FILE: LABLR    FORTRAN  A

C                                                                       LAB01590
C     OUTPUT CONDITIONAL MAP    .                                       LAB01600
CCCC                                                                    LAB01610
C                                                                       LAB01620
C                                                                       LAB01630
      CALL CLSMAP(CNDSUB,1,SUBNOS,SUBVEC,SUBNAM,CATVEC)                 LAB01640
   90 IF (MIX .EQ. 0) GO TO 100                                         LAB01650
      IF (COND.NE.0) OMAPFI = OMAPFI + 1                                LAB01660
C                                                                       LAB01670
CC    FLAG MIXED CLUSTERS                                               LAB01680
C                                                                       LAB01690
      CALL MIXMAP(ARRAY(COVAR2),MIXSUB,NOSUB2,CATVEC)                   LAB01700
C                                                                       LAB01710
CC    OUTPUT MIXED MAP                                                  LAB01720
C                                                                       LAB01730
      CALL CLSMAP(MIXSUB,2,SUBNOS,SUBVEC,SUBNAM,CATVEC)                 LAB01740
C                                                                       LAB01750
CC    OUTPUT DISPLAY INTERFACE TAPE--MAPTAP                             LAB01760
C                                                                       LAB01770
  100 IF (DSPKEY .EQ. 0) GO TO 110                                      LAB01780
      CALL PREAD(FLDADR,FLDSAV,TOTWDS,ISTAT)                            LAB01790
  103 IF (ISTAT .EQ. 1) GO TO 103                                       LAB01800
      CALL  PREAD(VTXADR,VERTX,TOTVTS,ISTAT)                            LAB01810
  105 IF (ISTAT .EQ. 1) GO TO 105                                       LAB01820
C                                                                       LAB01830
      CALL DSPTAP(SUBNOS,SUBNAM,FLDSAV,VERTX,CATVEC,SUBVEC,MEANS,       LAB01840
     *     ARRAY(COVAR2),TOP,ARRAY,NOFLD,TOTVRT)                        LAB01850
  110 CONTINUE                                                          LAB01860
      RETURN                                                            LAB01870
C                                                                       LAB01880
CCCC                                                                    LAB01890
C                                                                       LAB01900
CCCC  CODE TO PRINT DOT DATA RECORD                                     LAB01910
CC                                                                      LAB01920
C                                                                       LAB01930
  798 CONTINUE                                                          LAB01940
      ISTART=1                                                          LAB01950
      IEND=10                                                           LAB01960
  799 CONTINUE                                                          LAB01970
      IKT=0                                                             LAB01980
      DO 800 II=1,TOTDT2                                                LAB01990
      IKT=IKT+1                                                         LAB02000
      IF(IEND.GT.NOFEAT) IEND=NOFEAT                                    LAB02010
      IF(II.NE.1.AND.IKT.EQ.1)WRITE(6,810)                              LAB02020
  810 FORMAT(1H1,5(/))                                                  LAB02030
      IF(IKT.NE.1) GO TO 820                                            LAB02040
      WRITE(6,700)                                                      LAB02050
  700 FORMAT(//)                                                        LAB02060
      WRITE(6,690)                                                      LAB02070
  690 FORMAT(1X,'NO.',2X,'SAMPLE',2X,'LINE',2X,'TYPE',2X,'CATEGORY',    LAB02080
     1       30X,'DATA'/)                                               LAB02090
      WRITE(6,720)(BLANK,FETVC2(I),I=ISTART,IEND)                       LAB02100
  720 FORMAT(37X,10(A1,'CH(',I2,')'))                                   LAB02110
  820 CONTINUE                                                          LAB02120
      IDUM=COVAR2+(II-1)*SIZE-1                                         LAB02130
      IDMM=COVAR2+(II-1)*SIZE-3                                         LAB02140
      WRITE(6,710)II,(ARRAY(IDUM+I),I=1,4),(ARRAY(IDMM+JJ),JJ=ISTART,IEN LAB02150
     *D)                                                                LAB02160
  710 FORMAT(1X,I3,1H.,3X,I4,3X,I4,2X,I2,6X,I2,8X,10(I3,4X))            LAB02170
      WRITE(6,712)                                                      LAB02180
  712 FORMAT( )                                                         LAB02190
  800 CONTINUE                                                          LAB02200
      IF(NOFEAT.GT.10) GO TO 830                                        LAB02210
      GO TO 840                                                         LAB02220
  830 CONTINUE                                                          LAB02230
      IF(ITWO.EQ.1) GO TO 840                                           LAB02240
      ITWO=1                                                            LAB02250
      ISTART=IEND+ISTART                                                LAB02260
      IEND=NOFEAT                                                       LAB02270
      GO TO 799                                                         LAB02280
  840 CONTINUE                                                          LAB02290
      GOTO 30                                                           LAB02300
      END                                                               LAB02310
```

FILE: MANORD FORTRAN A

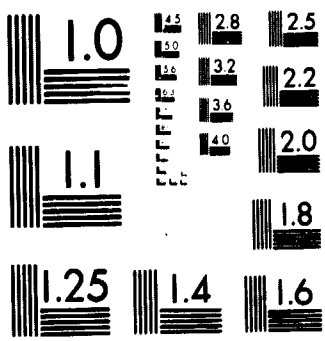
```
      SUBROUTINE MANORD(CLSNAM,CLSVEC,SUBVEC,NOCLS2,SUBNO,NOSUB2)          MAN00010
C                                                                          MAN00020
CCC   MANORD REGROUPS THE SUBCLASS IN THE ARRAY SUBVEC                     MAN00030
C                                                                          MAN00040
      IMPLICIT INTEGER (A-Z)                                               MAN00050
C                                                                          MAN00060
C     INCLUDE CMARK19.LIST                                                 MAN00070
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),    MAN00080
     *            SUBRAY(120),PTR(60),CATPTR(250),CATDOT(500),             MAN00090
     *             OUTVEC(250),COND,MIX,PROC,MARKEY,OUTKEY,STATKY,         MAN00100
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,              MAN00110
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTOT2,FLOINF(6),               MAN00120
     *             CLSSYM(42),STATMS,MEANAD,TABAUR,MAPAUR,SUNCOM(30),      MAN00130
     *       OOOTUN,OOOTFI,MANSTA,MANOOT,OSPUNT,OSPFIL,OSPKEY,PRNSTS,      MAN00140
     *  PRNOOT,FLONAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                      MAN00150
     *    ,TOTOT3,FLOAUR,VTXADR                                            MAN00160
C$END                                                                      MAN00170
      DIMENSION SUBNO(60) ,CLSVEC(1),CLSNAM(1),SUBVEC(1)                   MAN00180
      DIMENSION CLSNO(60)                                                  MAN00190
C                                                                          MAN00200
CCCCC                                                                      MAN00210
CCCCC                                                                      MAN00220
CCCCC                                                                      MAN00230
CCC   CHECK VALIDITY OF CLASS NAMES                                        MAN00240
C                                                                          MAN00250
      DO 150 I=1,NOCL2                                                     MAN00260
      DO 120 J=1,NOCLS2                                                    MAN00270
      IF (CLSNM2(I) .EQ. CLSNAM(J)) GO TO 145                              MAN00280
  120 CONTINUE                                                             MAN00290
      WRITE(6,130)(CLSNAM(K),K=1,NOCLS2)                                   MAN00300
  130 FORMAT(' ERROR IN INPUT OF CLASS NAMES. NAMES ON STAT FILE ARE :'/   MAN00310
     *    ,(7(A4,3X) / T55 ))                                             MAN00320
      WRITE(6,140)(CLSNM2(K),K=1,NOCL2)                                    MAN00330
  140 FORMAT(/ ' NAMES INPUT ARE : ',  (10(A4,3X)/))                       MAN00340
  145 CLSNO(I) = J                                                         MAN00350
  150 CONTINUE                                                             MAN00360
C                                                                          MAN00370
CCCC  REARRANGE SUBVC2 ARRAY SO ALL SUBCLASSES FOR A GIVEN CLASS ARE       MAN00380
      GROUPED TOGETHER                                                     MAN00390
C                                                                          MAN00400
      DO 160 I=1,NOCL2                                                     MAN00410
C                                                                          MAN00420
CCC   BEG. AND END  PTRS FOR SUBRAY                                        MAN00430
C                                                                          MAN00440
      IB = PTR(I) + 1                                                      MAN00450
      IE = IB + SUBRAY(IB-1) - 1                                           MAN00460
C     DO 160 J=IB,IE                                                       MAN00470
C                                                                          MAN00480
CCC   STORE NEW CLASS NO FOR RELABELED SUBCLASS                            MAN00490
C                                                                          MAN00500
      M = SUBRAY(J)                                                        MAN00510
  160 CLSVEC(M) = CLSNO(I)                                                 MAN00520
C                                                                          MAN00530
CCC   ZERO SUBNO ARRAY                                                     MAN00540
C                                                                          MAN00550
      DO 165 I=1,NOSUB2                                                    MAN00560
  165 SUBNO(I) = 0                                                         MAN00570
C                                                                          MAN00580
CCC   COMPUTE NO. OF SUBCLASSES IN EACH NEW CLASS                          MAN00590
C                                                                          MAN00600
      DO 170 I=1,NOSUB2                                                    MAN00610
      K = CLSVEC(I)                                                        MAN00620
  170 SUBNO(K) = SUBNO(K) + 1                                              MAN00630
C                                                                          MAN00640
CCCCC                 ORDER THE SUBCLASS NOS. ACCORDING TO THE             MAN00650
CCCCC  NEWLY ASSIGNED CLASS NO -- STORE IN SUBVEC.                         MAN00660
C                                                                          MAN00670
CCCCC                                                                      MAN00680
C                                                                          MAN00690
      KK = 0                                                               MAN00700
      DO 180 I=1,NOCLS2                                                    MAN00710
C                                                                          MAN00720
      DO 180 J=1,NOSUB2                                                    MAN00730
      IF (CLSVEC(J) .NE. I) GO TO 180                                      MAN00740
      KK = KK + 1                                                          MAN00750
      SUBVEC(KK) = J                                                       MAN00760
  180 CONTINUE                                                             MAN00770
C                                                                          MAN00780
      RETURN                                                               MAN00790
```

FILE: MANORD FORTRAN A

```
C
      END                                                          MAN00800
                                                                   MAN00810
```

ORIGINAL PAGE IS
OF POOR QUALITY

50
N80-26726 UNCLAS



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

FILE: MAPHNO FORTRAN A

```
C     THIS ROUTINE PRINTS THE HEADER INFORMATION FOR THE CLASSIFICATION MAP00010
C     MAP IN CLASSIFY AND DISPLAY                                       MAP00020
C                                                                       MAP00030
C                                                                       MAP00040
      SUBROUTINE MAPHNO(NOCAT,CLSSYM,CATNAM,KATNO,SUBDES,CATSUB)        MAP00050
C               NOCAT  -- NO. OF CATEGORIES                             MAP00060
C               CLSSYM -- SYMBOLS FOR CATEGORIES OR SUBCLASSES          MAP00070
C               CATNAM -- CATEGORY NAMES                                MAP00080
C               KATNO  -- CATEGORY EACH CLASS WAS ASSIGNED TO           MAP00090
C               CLSMTX -- CLASS NAMES                                   MAP00100
C               SUBNO  -- NO. OF SUBCLASSES IN EACH CLASS               MAP00110
C               SUBDES -- SUBCLASS NAMES                                MAP00120
C               CLSVC2 -- CLASS EACH SUBCLASS WAS ASSIGNED TO (IN COMMON MAP00130
C                         BLOCK INFORM)                                 MAP00140
C                                                                       MAP00150
      IMPLICIT INTEGER (A-Z)                                            MAP00160
C                                                                       MAP00170
C     INCLUDE COMBK1,LIST                                               MAP00180
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLDSV2,VERTX2,    COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        COM00030
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             COM00040
     *              GRPCHK(61),GROUPS(124)                              COM00050
CSEND                                                                   MAP00200
C                                                                       MAP00210
      LOGICAL ISWTH                                                     MAP00220
      DIMENSION CLSSYM(1),CATNAM(1),KATNO(1),SUBDES(1)                  MAP00230
      DIMENSION CATSUB(1)                                               MAP00240
C                                                                       MAP00250
C     PRINTS CATEGORY CLASSIFIER INFORMATION                            MAP00260
C                                                                       MAP00270
      WRITE(6,200)                                                      MAP00280
  200 FORMAT(//T54,'LABELED CLUSTER MAP'//T49,'( ** - DENOTES MIXED/COND MAP00290
     * CLUSTER )'/T50,'  # - DENOTES DO/DU AREA',9X,')'///T33,'LABEL',   MAP00300
     *  T81,'SUBCLASS'/T31,'NO.',T35,'NAME',T64,'UNLABELED NO.',T80,     MAP00310
     *'LABELED NO.',T94,'NAME',T101,'SYMBOL')                           MAP00320
C                                                                       MAP00330
      JJ = 0                                                            MAP00340
      KK = 0                                                            MAP00350
      DO 68   I=1,NOCAT                                                 MAP00360
C                                                                       MAP00370
      WRITE(6,210)I,CATNAM(I)                                           MAP00380
  210 FORMAT(/T31,I2,T37,A4)                                            MAP00390
      ISWTH = .TRUE.                                                    MAP00400
      DO 63  J=1,NOSUB2                                                 MAP00410
      IF (KATNO(J) .EQ. I) GO TO 64                                     MAP00420
      GO TO 63                                                          MAP00430
   64 KK = KK + 1                                                       MAP00440
      IF (ISWTH) GO TO 72                                               MAP00450
      WRITE(6,250)J,KK,SUBDES(KK),CLSSYM(J)                             MAP00460
  250 FORMAT(     T69,I2,T84,I2,T94,A4,T105,A1)                         MAP00470
      GO TO 75                                                          MAP00480
   72 WRITE(6,240)J,KK,SUBDES(KK),CLSSYM(J)                             MAP00490
  240 FORMAT(1H+,T69,I2,T84,I2,T94,A4,T105,A1)                          MAP00500
      ISWTH = .FALSE.                                                   MAP00510
   75 IF (CATSUB(J) .GT. (NOCAT  + 2)) WRITE(6,80)                      MAP00520
   80 FORMAT(1H+,T92,2H**)                                              MAP00530
   63 CONTINUE                                                          MAP00540
   68 CONTINUE                                                          MAP00550
      RETURN                                                            MAP00560
      END                                                               MAP00570
```

FILE: MIXMAP FORTRAN A

```
      SUBROUTINE MIXMAP(DOTS,MIXSUB,NOSUB2,CATVEC)                          MIX00010
C                                                                           MIX00020
CC    FLAGS THE MIXED CLUSTERS                                              MIX00030
C                                                                           MIX00040
      IMPLICIT INTEGER (A-Z)                                                MIX00050
C                                                                           MIX00060
CC    INCLUDE COMMK6.LIST                                                   MIX00070
CC    INCLUDE CMMKIS.LIST                                                   MIX00080
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             MIX00090
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,       MIX00100
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    MIX00110
     *  ,NHSTUN,NHSTFI,SCTWUN,MAPFIL                                        MIX00120
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                  MIX00130
     *  CRDUNT,PRTUNT,PANDIO                                                MIX00140
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),     MIX00150
     *             SUMRAY(120),PIP(60),CATPTR(250),CATDOT(500),             MIX00160
     *              DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,         MIX00170
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,               MIX00180
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),                MIX00190
     *              CLSSYM(62),STADRS,MEANAD,TABADR,MAPADR,SUNCOR(30) ,     MIX00200
     *       ODOTUN,ODOTFI,MANSTA,MANOUT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,       MIX00210
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                       MIX00220
     *    ,TOTDT3,FLDADR,VTXADR                                             MIX00230
C$END                                                                       MIX00240
C                                                                           MIX00250
      DIMENSION DOTS(SIZE,TOTDT2),MIXSUB(1),CLSTNO(250)                     MIX00260
      DIMENSION CATVEC(1)                                                   MIX00270
C                                                                           MIX00280
CC    INITIALIZE                                                            MIX00290
C                                                                           MIX00300
      DO 90 I=1,NOSUB2                                                      MIX00310
   90 MIXSUB(I) = CATVEC(I)                                                 MIX00320
C                                                                           MIX00330
      NSAMP = (FLDINF(5)-FLDINF(4))/FLDINF(6) + 1                           MIX00340
      NEXT = 63                                                             MIX00350
      DO 100 I=1,TOTDT2                                                     MIX00360
      ILINE = DOTS(2,I)                                                     MIX00370
      ISAMP = DOTS(1,I)                                                     MIX00380
      PIXADR = DRUMAD + (ILINE-1)*NSAMP+ISAMP-1                             MIX00390
      CALL RREAD(PIXADR,NUMBER,1,ISTAT)                                     MIX00400
  110 IF (ISTAT .EQ. 1) GO TO 110                                           MIX00410
  100 CLSTNO(I) = NUMBER                                                    MIX00420
C                                                                           MIX00430
CC    CLSTNO CONTAINS THE CLUSTER CLASSIFICATION NUMBER                     MIX00440
CC    MIXSUB FLAGS    THE MIXED CLUSTER                                     MIX00450
CC    CATVEC CONTAINS THE LABELED CATEGORY NUMBER PER CLUSTER               MIX00460
CC                                                                          MIX00470
CC    ARE ALL DOTS WITHIN A CLUSTER OF THE SAME LABEL                       MIX00480
C                                                                           MIX00490
      DO 150 I=1,NOSUB2                                                     MIX00500
      K = 0                                                                 MIX00510
      DO 140 J=1,TOTDT2                                                     MIX00520
      IF (CLSTNO(J) .NE. I) GO TO 140                                       MIX00530
      K = K + 1                                                             MIX00540
      IF (K .NE.1) GO TO 130                                                MIX00550
      CATNUM = DOTS(4,J)                                                    MIX00560
      GO TO 140                                                             MIX00570
C                                                                           MIX00580
  130 CATNBR = DOTS(4,J)                                                    MIX00590
      IF (CATNBR .EQ. CATNUM) GO TO 140                                     MIX00600
C                                                                           MIX00610
CC    FLAG CLUSTER AS MIXED                                                 MIX00620
C                                                                           MIX00630
      NEXT = NEXT - 1                                                       MIX00640
C                                                                           MIX00650
      MIXSUB(I) = NEXT                                                      MIX00660
      GO TO 150                                                             MIX00670
  140 CONTINUE                                                              MIX00680
  150 CONTINUE                                                              MIX00690
C                                                                           MIX00700
      RETURN                                                                MIX00710
      END                                                                   MIX00720
```

FILE: REORDER FORTRAN A

```
      SUBROUTINE REORDER(ARRAY,SURVEC,N)                                RE000010
C                                                                       RE000020
C     REORDER ID INFORMATION IN ARRAY                                   RE000030
C                                                                       RE000040
      IMPLICIT INTEGER (A-Z)                                            RE000050
C     INCLUDE COMMON.LIST                                               RE000060
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        COM00030
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             COM00040
     *              GRPCHK(61),GROUPS(124)                              COM00050
C$END                                                                   RE000080
      DIMENSION ARRAY(1),SUBNAM(60),N(1)                                RE000090
      DIMENSION SURVEC(1)                                               RE000100
C                                                                       RE000110
C     UPDATE SUBCLASS NAMES                                             RE000120
C                                                                       RE000130
      DO 100 I=1,NOSUB2                                                 RE000140
      K = SURVEC(I)                                                     RE000150
  100 SUBNAM(I) = ARRAY(SUBDS2 + K-1)                                   RE000160
      DO 110 I=1,NOSUB2                                                 RE000170
  110 ARRAY(SUBDS2 + I-1) = SUBNAM(I)                                   RE000180
C                                                                       RE000190
C                                                                       RE000200
C     UPDATE POPULATION ARRAY                                           RE000210
C                                                                       RE000220
      DO 130 I=1,NOSUB2                                                 RE000230
      K = SURVEC(I)                                                     RE000240
  130 N(I) = KEPPTS(K)                                                  RE000250
      RETURN                                                            RE000260
C                                                                       RE000270
      END                                                               RE000280
                                                                        RE000290
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SET14 FORTRAN A

```
      SUBROUTINE SET14(ARRAY,TOP)                                          SET00010
C*                                                                         SET00020
C*    SET14 READS IN THE CONTROL CARDS FOR THE LABEL PROCESSOR             SET00030
C*                                                                         SET00040
      IMPLICIT INTEGER (A-Z)                                               SET00050
       DIMENSION CODE (21), CARD (62), EQUCOM (3)                          SET00060
       DATA CODE / 'CHAN', 'DATA', 'MAPF', 'DOTF', 'OPTI', 'EXCL',         SET00070
     * 'STAT', 'DOTL', 'STAL', 'DIST', 'THRE', 'NEAR', 'PROC', 'MODU',     SET00080
     * 'SUNA', 'MAPT', 'DATE', 'COMM', 'HED1', 'HED2', '*END' /            SET00090
C      IN THE CONVERSION FROM 1108 TO IBM, STATLA BECOMES STAL             SET00100
      DIMENSION ARRAY(1),ACARD(20)                                         SET00110
      DIMENSION SLASH(2),EQUVEC(2)                                         SET00120
      DATA SLASH/1,'/'/                                                    SET00130
      DATA EQUVEC/1,'='/                                                   SET00140
      DATA  EQUCOM/2,'=',',',/                                             SET00150
      DATA CODE1/'    '/                                                   SET00160
C     INCLUDE COMBK1,LIST                                                  SET00170
C     INCLUDE COMBK4,LIST                                                  SET00180
C     INCLUDE COMBK6,LIST                                                  SET00190
C     INCLUDE CMBK15,LIST                                                  SET00200
      COMMON/INFORM/NOCLS2,NOSUR2,NOFET2,VARS72,TOTVT2,NOFLD2,             SET00210
     *              AVAP2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,       SET00220
     *              FETVC2(30),SURVC2(75),SURPTR(75),CLSVC2(60),           SET00230
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                SET00240
     *              GRPCHK(61),GROUPS(124)                                 SET00250
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                       SET00260
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                    SET00270
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                SET00280
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,            SET00290
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,      SET00300
     *    DRUMAD,DRMLOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                   SET00310
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                       SET00320
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                 SET00330
     *  CRDUNT,PRTUNT,HANDIO                                               SET00340
      COMMON /LABS/NOCAT,CATNAM(60),NOCL2,CLSNM2(60),NOCAT2,CATNM2(60),    SET00350
     *             SUBRAY(120),PTR(60),CATPTR(250),CATDOT(500),            SET00360
     *             DOTVEC(250),COND,MIX,PROC,MAPKEY,DOTKEY,STATKY,         SET00370
     *  SUNANG,T,NEARST,DIST,NOFEAT,FETVEC(30),OMAPUN,OMAPFI,              SET00380
     *   OSAVTP,OSTAFI,NOSUN,ANGLE(8),SIZE,TOTDT2,FLDINF(6),               SET00390
     *             CLSSYM(62),STADRS,MEANAD,TABADR,MAPADR,SUNCOR(30),      SET00400
     *        ODOTUN,ODOTFI,MANSTA,MANDOT,DSPUNT,DSPFIL,DSPKEY,PRNSTS,     SET00410
     *  PRNDOT,FLDNAM,VERTEX(22),NOVRT,NSUN,ANGLES(8)                      SET00420
     *    ,TOTDT3,FLDADR,VTXADR                                            SET00430
C$END                                                                      SET00440
      REAL  T,SUNCOR                                                       SET00450
       DATA BLANK / ' '/, SRCD / 'S'/, DBCD /'D'/, UBCD /'U'/,             SET00460
     * FRCD /'F'/, IPCD /'I'/, ORCD /'O'/, COMMA /','/, CBCD /'C'/,        SET00470
     * MRCD /'M'/, RCD1 /'1'/, RCD2 /'2'/, KBCD /'K'/, ABCD /'A'/          SET00480
C     INITIALIZE PARAMETERS                                                SET00490
C                                                                          SET00500
       NPUT = 21                                                           SET00510
      TOTDT3 = 0                                                           SET00520
      COND   = 0                                                           SET00530
      MIX    = 0                                                           SET00540
      PROC   = 1                                                           SET00550
      PRNDOT = 0                                                           SET00560
      PRNSTS = 0                                                           SET00570
      MANSTA = 0                                                           SET00580
      MANDOT = 0                                                           SET00590
      MAPKEY = 0                                                           SET00600
      DOTKEY = 0                                                           SET00610
      DSPKEY = 0                                                           SET00620
      STATKY = 0                                                           SET00630
      NOSUN  = 0                                                           SET00640
      SUNANG = 0                                                           SET00650
      T      = 25.0                                                        SET00660
      NEARST = 1                                                           SET00670
      DIST   = 1                                                           SET00680
      NOFET2 = 0                                                           SET00690
      NOFEAT = 0                                                           SET00700
       PTR1 = 0                                                            SET00710
       PTR2 =0                                                             SET00720
       NOCAT2 = 0                                                          SET00730
       NOCL2 =0                                                            SET00740
C                                                                          SET00750
C                                                                          SET00760
      CLSVC2(1) = BLANK                                                    SET00770
      SURPTR(75) = BLANK                                                   SET00780
C                                                                          SET00790
```

```
FILE: SET14    FORTRAN  A

      DO 20 I=1,30                                                          SET00800
   20 SUNCOR(I) = 1.0                                                       SET00810
      WRITE(6,100)                                                          SET00820
  100 FORMAT(/11X,'INPUT SUMMARY'//)                                        SET00830
C                                                                           SET00840
CC    PUT THE NEXT CARD IN THE REREAD BUFFER                                SET00850
C                                                                           SET00860
      RRUNIT=30                                                             SET00870
  105 READ(21,107)(ACARD(I),I=1,20)                                         SET00880
  107 FORMAT(20A4)                                                          SET00890
      WRITE(30,107)(ACARD(I),I=1,20)                                        SET00900
      REWIND RRUNIT                                                         SET00910
      READ(30,110)CODE1,CARD                                                SET00920
      REWIND RRUNIT                                                         SET00930
      COL= 0                                                                SET00940
      WRITE(6,120)CODE1,CARD                                                SET00950
  110  FORMAT (A4, 6X, 62A1)                                                SET00960
  120  FORMAT (1X, A4, 6X, 62A1)                                            SET00970
      DO 130  I=1,NPUT                                                      SET00980
       IF(CODE1.EQ.CODE(I)) GO TO (150,180,210,250,330,365,370,             SET00990
     *  390,400,410,420,430,440,460,470,490,510,530,540,550,560),I         SET01000
  130  CONTINUE                                                             SET01010
  135 WRITE(6,140)                                                          SET01020
  140 FORMAT( ' INVALID CONTROL CARD - IGNORED ')                           SET01030
      GO TO 105                                                             SET01040
C                                                                           SET01050
CC    CHANNEL CARD                                                          SET01060
C                                                                           SET01070
  150 M=NXTCHR(CARD,COL)                                                    SET01080
       IF (M .EQ. BLANK)GO TO 105                                           SET01090
       IF (M .EQ. SRCD) GO TO 160                                           SET01100
       IF (M .EQ. DRCD) GO TO 170                                           SET01110
  153 WRITE(6,155)                                                          SET01120
  155 FORMAT( ' ERROR ON CHANNELS CARD')                                    SET01130
      GO TO 105                                                             SET01140
  160 J = FIND12(CARD,COL,EQUCOM)                                           SET01150
      IF ( J .NE. 2  ) GO TO 153                                            SET01160
      NOFET2 = NUMBER(CARD,COL,FETVC2,NOFET2)                               SET01170
      COL = COL - 1                                                         SET01180
      CALL ORDER(FETVC2,NOFET2)                                             SET01190
      GO TO 150                                                             SET01200
  170 J = FIND12(CARD,COL,EQUCOM)                                           SET01210
      IF (J .NE. 2 ) GO TO 153                                              SET01220
      NOFEAT = NUMBER(CARD,COL,FETVEC,NOFEAT)                               SET01230
      COL = COL - 1                                                         SET01240
      CALL ORDER(FETVEC,NOFEAT)                                             SET01250
      GO TO 150                                                             SET01260
C                                                                           SET01270
CC    DATA FILE CARD                                                        SET01280
C                                                                           SET01290
  180 M = NXTCHR(CARD,COL)                                                  SET01300
       IF (M .EQ. BLANK)GO TO 105                                           SET01310
       IF (M .EQ. URCD) GO TO 190                                           SET01320
       IF (M .EQ. FRCD) GO TO 200                                           SET01330
  185 WRITE(6,187)                                                          SET01340
  187 FORMAT(' ERROR ON DATA FILE CARD')                                    SET01350
      GO TO 105                                                             SET01360
  190 J = FIND12(CARD,COL,EQUCOM)                                           SET01370
      IF (J .NE. 2) GO TO 185                                               SET01380
      M = NUMBER(CARD,COL,DATAPE,ZERO)                                      SET01390
      COL = COL - 1                                                         SET01400
      GO TO 180                                                             SET01410
  200 J = FIND12(CARD,COL,EQUCOM)                                           SET01420
      IF ( J .NE. 2 ) GO TO 185                                             SET01430
      M = NUMBER(CARD,COL,DATFIL,ZERO)                                      SET01440
      DATFIL = DATFIL - 1                                                   SET01450
      COL = COL - 1                                                         SET01460
      GO TO 180                                                             SET01470
C                                                                           SET01480
CC    CLUSTER/CLASSIFICATION MAP TAPE                                       SET01490
C                                                                           SET01500
  210 M = NXTCHR(CARD,COL)                                                  SET01510
  211   IF (M .EQ. IRCD) GO TO 213                                          SET01520
        IF (M .EQ. ORCD) GO TO 220                                          SET01530
        IF (M .EQ. BLANK)GO TO 105                                          SET01540
  212 WRITE(6,216)                                                          SET01550
  216 FORMAT(' ERROR ON MAPFIL CARD')                                       SET01560
      GO TO 105                                                             SET01570
C                                                                           SET01580
```

```
FILE: SET14    FORTRAN  A

  213 J = FIND12(CARD,COL,SLASH)                                        SET01590
      IF (J .EQ. -1) GO TO 212                                          SET01600
      MAPKEY = 1                                                        SET01610
  214 M = NXTCHR(CARD,COL)                                              SET01620
       IF (M .EQ. COMMA)GO TO 214                                       SET01630
       IF (M .EQ. FRCD) GO TO 215                                       SET01640
       IF (M .NE. URCD) GO TO 211                                       SET01650
      J = FIND12(CARD,COL,EQUVEC)                                       SET01660
      IF (J .EQ. -1) GO TO 212                                          SET01670
      M = NUMBER(CARD,COL,MAPUNT,ZERO)                                  SET01680
      COL = COL - 1                                                     SET01690
      GO TO 214                                                         SET01700
  215 J = FIND12(CARD,COL,EQUVEC)                                       SET01710
      IF (J .EQ. -1) GO TO 212                                          SET01720
      M = NUMBER(CARD,COL,MAPFIL,ZERO)                                  SET01730
      COL = COL - 1                                                     SET01740
      MAPFIL = MAPFIL - 1                                               SET01750
      GO TO 214                                                         SET01760
  220 J = FIND12(CARD,COL,SLASH)                                        SET01770
      IF (J .EQ. -1) GO TO 212                                          SET01780
  221 M = NXTCHR(CARD,COL)                                              SET01790
       IF (M .EQ. COMMA)GO TO 221                                       SET01800
       IF (M .EQ. FRCD) GO TO 222                                       SET01810
       IF (M .NE. URCD) GO TO 211                                       SET01820
      J = FIND12(CARD,COL,EQUVEC)                                       SET01830
      IF (J .EQ. -1) GO TO 212                                          SET01840
      M = NUMBER(CARD,COL,OMAPUN,ZERO)                                  SET01850
      COL = COL - 1                                                     SET01860
      GO TO 221                                                         SET01870
  222 J = FIND12(CARD,COL,EQUVEC)                                       SET01880
      IF (J .EQ. -1) GO TO 212                                          SET01890
      M = NUMBER(CARD,COL,OMAPFI,ZERO)                                  SET01900
      COL = COL - 1                                                     SET01910
      OMAPFI = OMAPFI - 1                                               SET01920
      GO TO 221                                                         SET01930
C                                                                       SET01940
C     DOTFIL CARD                                                       SET01950
C                                                                       SET01960
  250 M = NXTCHR(CARD,COL)                                              SET01970
  251  IF (M .EQ. IRCD) GO TO 254                                       SET01980
       IF (M .EQ. ORCD) GO TO 260                                       SET01990
       IF (M .EQ. BLANK)GO TO 105                                       SET02000
  252 WRITE(4,253)                                                      SET02010
  253 FORMAT(' ERROR ON MAPFILE CARD')                                  SET02020
      GO TO 105                                                         SET02030
C                                                                       SET02040
  254 J = FIND12(CARD,COL,SLASH)                                        SET02050
      IF ( J .EQ. -1) GO TO 252                                         SET02060
      DOTKEY = 1                                                        SET02070
  256 M = NXTCHR(CARD,COL)                                              SET02080
       IF (M .EQ. COMMA)GO TO 256                                       SET02090
       IF (M .EQ. FRCD) GO TO 258                                       SET02100
       IF (M .NE. URCD) GO TO 251                                       SET02110
      J = FIND12(CARD,COL,EQUVEC)                                       SET02120
      IF (J .EQ. -1) GO TO 252                                          SET02130
      M = NUMBER(CARD,COL,DOTUNT,ZERO)                                  SET02140
      COL = COL - 1                                                     SET02150
      GO TO 256                                                         SET02160
  258 J = FIND12(CARD,COL,EQUVEC)                                       SET02170
      IF (J .EQ. -1) GO TO 252                                          SET02180
      M = NUMBER(CARD,COL,DOTFIL,ZERO)                                  SET02190
      COL = COL -1                                                      SET02200
      DOTFIL = DOTFIL - 1                                               SET02210
      GO TO 256                                                         SET02220
C                                                                       SET02230
  260 J = FIND12(CARD,COL,SLASH)                                        SET02240
      IF (J .EQ. -1) GO TO 252                                          SET02250
  262 M = NXTCHR(CARD,COL)                                              SET02260
       IF (M .EQ. COMMA)GO TO 262                                       SET02270
       IF (M .EQ. FRCD) GO TO 264                                       SET02280
       IF (M .NE. URCD) GO TO 251                                       SET02290
      J = FIND12(CARD,COL,EQUVEC)                                       SET02300
      IF ( J .EQ. -1) GO TO 252                                         SET02310
      M = NUMBER(CARD,COL,ODOTUN,ZERO)                                  SET02320
      COL = COL - 1                                                     SET02330
      GO TO 262                                                         SET02340
  264 J = FIND12(CARD,COL,EQUVEC)                                       SET02350
      IF (J .EQ. -1) GO TO 252                                          SET02360
      M = NUMBER(CARD,COL,ODOTFI,ZERO)                                  SET02370
```

```
FILE: SET14    FORTRAN  A

      COL = COL - 1                                                    SET02380
      ODOTFI = ODOTFI - 1                                              SET02390
      GO TO 262                                                        SET02400
C                                                                      SET02410
CC    OPTION CARD                                                      SET02420
C                                                                      SET02430
  330 M = NXTCHR(CARD,COL)                                             SET02440
       IF (M .EQ. BLANK)GO TO 105                                      SET02450
       IF (M .EQ. CHCD) GO TO 340                                      SET02460
       IF (M .EQ. SRCD) GO TO 345                                      SET02470
       IF (M .EQ. DRCD) GO TO 350                                      SET02480
       IF (M .EQ. MBCD) GO TO 363                                      SET02490
  333 WRITTE(6,335)                                                    SET02500
  335 FORMAT(' ERROR ON OPTION CARD')                                  SET02510
      GO TO 105                                                        SET02520
  340 COND = 1                                                         SET02530
      GO TO 360                                                        SET02540
  345 PRNSTS = 1                                                       SET02550
      GO TO 360                                                        SET02560
  350 PRNDOT = 1                                                       SET02570
      GO TO 360                                                        SET02580
  363 MIX = 1                                                          SET02590
  360 J = FIND12(CARD,COL,EQUCOM)                                      SET02600
      IF ( J .EQ. 3) GO TO 330                                         SET02610
      IF ( J .EQ. -1) GO TO 105                                        SET02620
C                                                                      SET02630
CC                                                                     SET02640
CC    EXCLUDE CARD                                                     SET02650
C                                                                      SET02660
  365 TOTDT3 = NUMBER(CARD,COL,DOTVEC,TOTDT3)                          SET02670
      CALL ORDER(DOTVEC,TOTDT3)                                        SET02680
      GO TO 105                                                        SET02690
C                                                                      SET02700
CC    STATFILE CARD                                                    SET02710
C                                                                      SET02720
  370 M = NXTCHR(CARD,COL)                                             SET02730
  371  IF (M .EQ. IRCD) GO TO 374                                      SET02740
       IF (M .EQ. ORCD) GO TO 380                                      SET02750
       IF (M .EQ. BLANK)GO TO 105                                      SET02760
  372 WRITE(6,373)                                                     SET02770
  373 FORMAT(' ERROR ON STATFI CARD')                                  SET02780
      GO TO 105                                                        SET02790
C                                                                      SET02800
  374 J = FIND12(CARD,COL,SLASH)                                       SET02810
      IF (J .EQ. -1) GO TO 372                                         SET02820
      STATKY = 1                                                       SET02830
  375 M = NXTCHR(CARD,COL)                                             SET02840
       IF (M .EQ. COMMA)GO TO 375                                      SET02850
       IF (M .EQ. FRCD) GO TO 376                                      SET02860
       IF (M .NE. URCD) GO TO 370                                      SET02870
      J = FIND12(CARD,COL,EQUVEC)                                      SET02880
      IF (J .EQ. -1) GO TO 372                                         SET02890
      M = NUMBER(CARD,COL,SAVTAP,ZERO)                                 SET02900
      COL = COL - 1                                                    SET02910
      GO TO 375                                                        SET02920
  376 J = FIND12(CARD,COL,EQUVEC)                                      SET02930
      IF (J .EQ. -1) GO TO 372                                         SET02940
      M = NUMBER(CARD,COL,STAFIL,ZERO)                                 SET02950
      COL = COL - 1                                                    SET02960
      STAFIL = STAFIL - 1                                              SET02970
      GO TO 375                                                        SET02980
  380 J = FIND12(CARD,COL,SLASH)                                       SET02990
      IF (J .EQ. -1) GO TO 372                                         SET03000
  381 M = NXTCHR(CARD,COL)                                             SET03010
       IF (M .EQ. COMMA)GO TO 381                                      SET03020
       IF (M .EQ. FRCD) GO TO 382                                      SET03030
       IF (M .NE. URCD) GO TO 370                                      SET03040
      J = FIND12(CARD,COL,EQUVEC)                                      SET03050
      IF (J .EQ. -1) GO TO 372                                         SET03060
      M = NUMBER(CARD,COL,OSAVTP,ZERO)                                 SET03070
      COL = COL - 1                                                    SET03080
      GO TO 381                                                        SET03090
  382 J = FIND12(CARD,COL,EQUVEC)                                      SET03100
      IF (J .EQ. -1) GO TO 372                                         SET03110
      M = NUMBER(CARD,COL,OSTAFI,ZERO)                                 SET03120
      COL = COL - 1                                                    SET03130
      OSTAFI = OSTAFI - 1                                              SET03140
      GO TO 381                                                        SET03150
C                                                                      SET03160
```

```
FILE: SET14    FORTRAN  A

C       DOTLABEL CARD                                                          SET03170
C                                                                              SET03180
  390 M = CRDSCN(CARD,CATPTR,CATNM2,CATDOT,NOCAT2,PTR1)                        SET03190
      MANDOT = 1                                                               SET03200
      GO TO 105                                                                SET03210
C                                                                              SET03220
C       STATLABEL CARD                                                         SET03230
C                                                                              SET03240
  400 M = CRDSCN(CARD,PTR,CLSNM2,SUBRAY,NOCL2,PTR2)                            SET03250
      MANSTA = 1                                                               SET03260
      GO TO 105                                                                SET03270
C                                                                              SET03280
C       DISTANCE                                                               SET03290
C                                                                              SET03300
  410 M = NXTCHR(CARD,COL)                                                     SET03310
      M = NXTCHR(CARD,COL)                                                     SET03320
        IF (M .EQ. 4CD1) DIST = 1                                              SET03330
        IF (M .EQ. 4CD2) DIST = 2                                              SET03340
      GO TO 105                                                                SET03350
C                                                                              SET03360
C       THRESHOLD CARD                                                         SET03370
C                                                                              SET03380
  420 M = FLTNUM(CARD,COL,T,1)                                                 SET03390
      GO TO 105                                                                SET03400
C                                                                              SET03410
C       K NEAREST DOTS                                                         SET03420
C                                                                              SET03430
  430 J = NUMBER(CARD,COL,NEARST,ZERO)                                         SET03440
      GO TO 105                                                                SET03450
C                                                                              SET03460
C       PROCEDURE CARD                                                         SET03470
C                                                                              SET03480
C       K= NEAREST PROCEDURE = 1                                               SET03490
C       ALL-OF-A-KIND        = 2                                               SET03500
C       MANUAL LABELING OF FILES = 3                                           SET03510
C                                                                              SET03520
  440 M = NXTCHR(CARD,COL)                                                     SET03530
        IF (M .EQ. KRCD) GO TO 446                                             SET03540
        IF (M .EQ. ARCD) GO TO 448                                             SET03550
        IF (M .EQ. MRCD) GO TO 450                                             SET03560
      WRITE(6,445)                                                             SET03570
  445 FORMAT(' ERROR ON PROCEDURE CARD')                                       SET03580
      GO TO 105                                                                SET03590
  446 PROC = 1                                                                 SET03600
      GO TO 105                                                                SET03610
  448 PROC = 2                                                                 SET03620
      GO TO 105                                                                SET03630
  450 PROC = 3                                                                 SET03640
      GO TO 105                                                                SET03650
C                                                                              SET03660
C       MODULE CARD DECK                                                       SET03670
C                                                                              SET03680
  460 CALL CRDSTA(ARRAY,TOP)                                                   SET03690
      STATKY = 1                                                               SET03700
      GO TO 105                                                                SET03710
C                                                                              SET03720
C       SUN ANGLE CARD                                                         SET03730
C                                                                              SET03740
C       SUNANG = 1 -- ANGLES ARE ON DOTFIL                                     SET03750
C              = 2 -- ANGLES ARE ON CARDS                                      SET03760
  470 M = NXTCHR(CARD,COL)                                                     SET03770
      IF (M .NE. FRCD) GO TO 475                                               SET03780
      SUNANG = 1                                                               SET03790
      GO TO 105                                                                SET03800
  475 COL = COL - 1                                                            SET03810
      IF(M.NE.FRCD) GO TO 475                                                  SET03820
      SUNANG = 2                                                               SET03830
      GO TO 105                                                                SET03840
C                                                                              SET03850
C       MAPTAP CARD --DISPLAY INTERFACE TAPE                                   SET03860
C                                                                              SET03870
  490 M = NXTCHR(CARD,COL)                                                     SET03880
        IF (M .EQ. ORCD) GO TO 496                                             SET03890
  491   IF (M .EQ. BLANK)GO TO 105                                             SET03900
  492 WRITE(6,494)                                                             SET03910
  494 FORMAT(' ERROR ON MAPTAP CARD')                                          SET03920
      GO TO 105                                                                SET03930
                                                                               SET03940
                                                                               SET03950
```

```
FILE: SET14    FORTRAN  A

  496 J = FIND12(CARD,COL,SLASH)                                        SET03960
      IF (J .EQ. -1) GO TO 492                                          SET03970
      DSPKEY = 1                                                        SET03980
  498 M = NXTCHR(CARD,COL)                                              SET03990
       IF (M .EQ. COMMA)GO TO 498                                       SET04000
       IF (M .EQ. FRCD) GO TO 500                                       SET04010
       IF (M .NE. URCD) GO TO 491                                       SET04020
      J = FIND12(CARD,COL,EQUVEC)                                       SET04030
      IF (J .EQ. -1) GO TO 492                                          SET04040
      M = NUMBER(CARD,COL,DSPUNT,ZERO)                                  SET04050
      COL = COL - 1                                                     SET04060
      GO TO 498                                                         SET04070
  500 J = FIND12(CARD,COL,EQUVEC)                                       SET04080
      IF (J .EQ. -1) GO TO 492                                          SET04090
      M = NUMBER(CARD,COL,DSPFIL,ZERO)                                  SET04100
      DSPFIL = DSPFIL - 1                                               SET04110
      COL = COL - 1                                                     SET04120
      GO TO 498                                                         SET04130
C     DATE CARD                                                         SET04140
C                                                                       SET04150
  510 M = NXTCHR(CARD,COL)                                              SET04160
       IF (M .EQ. BLANK)GO TO 105                                       SET04170
      READ(30,520)DATE                                                  SET04180
      REWIND RRUNIT                                                     SET04190
  520 FORMAT(10X,15A4)                                                  SET04200
      GO TO 105                                                         SET04210
C                                                                       SET04220
CC    COMMENT CARD                                                      SET04230
C                                                                       SET04240
  530 M = NXTCHR(CARD,COL)                                              SET04250
       IF (M .EQ. BLANK)GO TO 105                                       SET04260
      READ(30,520)COMENT                                                SET04270
      REWIND RRUNIT                                                     SET04280
      GO TO 105                                                         SET04290
C                                                                       SET04300
C                                                                       SET04310
CC    HED1                                                              SET04320
  540 M = NXTCHR(CARD,COL)                                              SET04330
       IF (M .EQ. BLANK)GO TO 105                                       SET04340
      READ(30,520)HED1                                                  SET04350
      REWIND RRUNIT                                                     SET04360
      GO TO 105                                                         SET04370
C                                                                       SET04380
CC    HED2                                                              SET04390
C                                                                       SET04400
  550 M = NXTCHR(CARD,COL)                                              SET04410
       IF (M .EQ. BLANK)GO TO 105                                       SET04420
      READ(30,520)HED2                                                  SET04430
      REWIND RRUNIT                                                     SET04440
      GO TO 105                                                         SET04450
C                                                                       SET04460
CC    *END*                                                             SET04470
C                                                                       SET04480
  560 CONTINUE                                                          SET04490
C                                                                       SET04500
C                                                                       SET04510
C                                                                       SET04520
CC    PROCEDURE 1 AND 2 CANNOT BE SELECTED ALONG WITH PROCEDURE 3       SET04530
C                                                                       SET04540
      IF (PROC .EQ. 3) GO TO 570                                        SET04550
      IF (NOFET2 .EQ. 0 .OR. NOFEAT .EQ. 0) GO TO 570                   SET04560
      IF (NOFET2 .EQ. NOFEAT) GO TO 570                                 SET04570
      WRITE(6,2050)                                                     SET04580
C                                                                       SET04590
  570 IF (MANOOT .EQ. 0) GO TO 580                                      SET04600
      IF (PROC .EQ. 3) GO TO 580                                        SET04610
      WRITE(6,2060)                                                     SET04620
      PROC = 3                                                          SET04630
C                                                                       SET04640
  580 IF (MANSTA .EQ. 0) GO TO 590                                      SET04650
      IF (PROC .EQ. 3) GO TO 590                                        SET04660
      WRITE(6,2060)                                                     SET04670
      PROC = 3                                                          SET04680
C                                                                       SET04690
  590 CONTINUE                                                          SET04700
      WRITE(6,1000)                                                     SET04710
      IF (NOFET2 .NE. 0)WRITE(6,1010)(FETVC2(I),I=1,NOFET2)             SET04720
      IF (NOFEAT .NE. 0) WRITE(6,1020)(FETVEC(I),I=1,NOFEAT)            SET04730
      IF (MARKEY .EQ. 1) WRITE(6,1030)                                  SET04740
```

ORIGINAL PAGE IS
OF POOR QUALITY


```
      IF (MANDOT .EQ. 1) WRITE(6,1040)                                  SET04750
      IF (MANSTA .EQ. 1) WRITE(6,1050)                                  SET04760
      IF (DSPKEY .EQ. 1) WRITE(6,1055)                                  SET04770
      IF (PROC .EQ. 1)WRITE(6,1060)                                     SET04780
      IF (PROC .EQ. 2)WRITE(6,1070)                                     SET04790
      IF (DIST .EQ. 1)WRITE(6,1080)                                     SET04800
      IF (DIST .EQ. 2)WRITE(6,1090)                                     SET04810
      WRITE(6,2000)T                                                    SET04820
      WRITE(6,2010)NEARST                                               SET04830
      IF (SUNANG .EQ.1)WRITE(6,2020)                                    SET04840
      IF (SUNANG .EQ.2)WRITE(6,2030)                                    SET04850
      IF (SUNANG .EQ.0)WRITE(6,2040)                                    SET04860
      IF(DOTKEY .EQ. 1) WRITE(6,2070)                                   SET04870
      IF (STATKY .EQ. 1) WRITE(6,2080)                                  SET04880
      IF (COND .EQ. 1) WRITE(6,2090)                                    SET04890
      IF (MIX .EQ. 1) WRITE(6,3000)                                     SET04900
      IF (PWNDOT.EQ.1) WRITE(6,3020)                                    SET04910
      IF (PRNSTS.EQ.1) WRITE(6,3030)                                    SET04920
C                                                                       SET04930
      IERR = 0                                                          SET04940
      IF (PROC .EQ. 3) GO TO 600                                        SET04950
      IF (STATKY .NE. 1) IERR = 1                                       SET04960
      IF (DOTKEY .NE. 1) IERR = 1                                       SET04970
      IF (COND .EQ. 1 .OR. MIX .EQ. 1) GO TO 594                        SET04980
  593 IF (PROC .EQ. 1) GO TO 595                                        SET04990
  594 IF (MAPKEY .NE. 1) IERR = 1                                       SET05000
  595 IF (IERR .EQ. 1) WRITE(6,3010)                                    SET05010
  600 CONTINUE                                                          SET05020
C                                                                       SET05030
 1000 FORMAT(//' USER HAS REQUESTED THE FOLLOWING OPTIONS :'/)          SET05040
 1010 FORMAT(' STAT CHANNELS ARE',30(I2,1X))                            SET05050
 1020 FORMAT(' DOTDATA CHANNELS ARE',30(I2,1X))                         SET05060
 1030 FORMAT(' CLUSTER/CLASSIFICATION TAPE IS BEING INPUT')             SET05070
 1040 FORMAT(' DOTFIL WILL BE RELABELED')                               SET05080
 1050 FORMAT(' STAT FILE WILL BE RELABELED')                            SET05090
 1055 FORMAT(' MAPTAP FILE WILL BE OUTPUT')                             SET05100
 1060 FORMAT(' K-NEAREST PROCEDURE WILL BE USED')                       SET05110
 1070 FORMAT(' ALL-OF-A-KIND PROCEDURE WILL BE USED')                   SET05120
 1080 FORMAT(' L1 DISTANCE WILL BE USED')                               SET05130
 1090 FORMAT(' L2 DISTANCE WILL BE USED')                               SET05140
 2000 FORMAT(' THRESHOLD DISTANCE = ',F10.3)                            SET05150
 2010 FORMAT( 1X,I3,'-NEAREST DOTS WILL BE USED')                       SET05160
 2020 FORMAT(' SUN ANGLES WILL BE EXTRACTED FROM DOTFIL')               SET05170
 2030 FORMAT(' SUN ANGLES WILL BE READ IN FROM CARDS')                  SET05180
 2040 FORMAT(' NO SUN ANGLE CORRECTION WILL BE APPLIED')                SET05190
 2050 FORMAT(/' NO. OF STAT CHANNEL AND DOT DATA CHANNELS MUST BE EQUAL'SET05200
     *)                                                                 SET05210
 2060 FORMAT(/' A LABELING PROCEDURE MAY NOT BE CHOSEN WHEN UPDATING THESET05220
     * DOTFIL OR SAVTAP FILES')                                        SET05230
 2070 FORMAT(' DOTFIL FILE IS BEING INPUT')                             SET05240
 2080 FORMAT(' SAVTAP FILE IS BEING INPUT')                             SET05250
 2090 FORMAT(' CONDITIONAL CLUSTER MAP WILL BE OUTPUT')                 SET05260
 3000 FORMAT(' MIXED CLUSTER MAP WILL BE OUTPUT')                       SET05270
 3010 FORMAT(/' USER HAS NOT INPUT ONE OF THE REQUIRED FILES:'/T20,     SET05280
     *'SAVTAP      MAPFIL      OR      DOTFIL')                        SET05290
 3020 FORMAT(' PRINT UPDATED DOTFILE')                                  SET05300
 3030 FORMAT(' PRINT MEANS AND COVARIANCES')                            SET05310
C                                                                       SET05320
      RETURN                                                            SET05330
C                                                                       SET05340
      END                                                               SET05350
```

FILE: STOMAP FORTRAN A

```
      SUBROUTINE STOMAP(ILINE,NSAMP,HIST,LIMIT,BEGIN1)                  ST0000010
C*                                                                      ST0000020
C*    STODAT READS AND STORES THE CLASSIFICATION/CLUSTER MAP ON DRUM    ST0000030
C*                                                                      ST0000040
C*                                                                      ST0000050
      IMPLICIT INTEGER (A-Z)                                            ST0000060
C     INCLUDE COMMON.LIST                                               ST0000070
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         ST0000080
     *             HISFIL,HISKEY,TRFORM,EMIPTP,EMPKEY,MAPUNT,NOFILE,    ST0000090
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                ST0000100
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                    ST0000110
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              ST0000120
     *  CROUNT,PRTUNT,RANDIO                                            ST0000130
CSEND                                                                   ST0000140
      DIMENSION HIST(LIMIT),FETVEC(1),FLD(6),NLINE(4)                   ST0000150
      TOTWRD = ILINE*NSAMP                                              ST0000160
      IF (TOTWRD .LE. (DRMWDS-(DRUMAD-BEGIN1))) GO TO 120               ST0000170
      WRITE(6,110)                                                      ST0000180
  110 FORMAT(' NOT ENOUGH DRUM SPACE TO STORE DAS TAPE DATA')           ST0000190
      CALL CMERR                                                        ST0000200
C                                                                       ST0000210
  120 CALL TAPHDR(MAPUNT,MAPFIL)                                        ST0000220
      FETVEC(1) = 1                                                     ST0000230
      NOFEAT = 1                                                        ST0000240
      FLD(1) = 1                                                        ST0000250
      FLD(2) = ILINE                                                    ST0000260
      FLD(3) = 1                                                        ST0000270
      FLD(4) = 1                                                        ST0000280
      FLD(5) = NSAMP                                                    ST0000290
      FLD(6) = 1                                                        ST0000300
      BEGIN = BEGIN1                                                    ST0000310
      CALL FLDINI(FLD   ,FETVEC,NOFEAT)                                 ST0000320
      DUMPS = TOTWRD / LIMIT                                            ST0000330
      IF (MOD(TOTWRD,LIMIT) .NE. 0) DUMPS = DUMPS + 1                   ST0000340
      TOTLNS = LIMIT / NSAMP                                            ST0000350
      IF (TOTLNS .GE. ILINE)  GO TO 140                                 ST0000360
      DMP = DUMPS - 1                                                   ST0000370
      DO 130  I=1,DMP                                                   ST0000380
  130 NLINE(I) = TOTLNS                                                 ST0000390
      NLINE(DUMPS) = ILINE - TOTLNS*DMP                                 ST0000400
      GO TO 150                                                         ST0000410
  140 NLINE(1) = ILINE                                                  ST0000420
C                                                                       ST0000430
  150 DO 200 J=1,DUMPS                                                  ST0000440
      NUMLIN = NLINE(J)                                                 ST0000450
      DO 160 K=1,NUMLIN                                                 ST0000460
      WORDS = NSAMP*(K-1)                                               ST0000470
  160 CALL LINERD(HIST(WORDS+1),ENDTAP)                                 ST0000480
C                                                                       ST0000490
C     STORE ON HIGH SPEED DRUM                                          ST0000500
C                                                                       ST0000510
      NWORDS = WORDS + NSAMP                                            ST0000520
      CALL RWRITE(BEGIN,HIST(1),NWORDS,ISTAT)                           ST0000530
  200 BEGIN = BEGIN + NLINE(J) * NSAMP                                  ST0000540
C                                                                       ST0000550
      MAPFIL = MAPFIL + 1                                               ST0000560
C                                                                       ST0000570
      RETURN                                                            ST0000580
      END
```

19. UTILITY SUBPROGRAMS

FILE: BMFIL FORTRAN A

```
      SUBROUTINE BMFIL (BMAT,LCOMB,NOFET,VEC,KEY)                                BMF00010
      IMPLICIT INTEGER(A-Z)                                                      BMF00020
      REAL BMAT                                                                  BMF00030
C*                                                                               BMF00040
C*    DEPENDING ON THE VALUE OF 'KEY', THIS SUBROUTINE PERFORMS ONE OF           BMF00050
C*    FIVE I/O FUNCTIONS. WITH REGARD TO THE TRANSFORMATION MATRIX 'B'           BMF00060
C*                                                                               BMF00070
C*        KEY=1  B-MATRIX IS READ FROM CARDS AND STORED ON FILE.                 BMF00080
C*               STORAGE MUST BE PROVIDED IN BMAT ARGUMENT                       BMF00090
C*        KEY=2  B-MATRIX IS READ FROM FILE.                                     BMF00100
C*        KEY=3  THE VALUES OF LCOMB,NOFET AND VEC ARE READ FROM FILE.           BMF00110
C*               THIS FUNCTION CAN BE USED FOR ESTABLISHING DIMENSIONS           BMF00120
C*               FOR THE B-MATRIX.                                               BMF00130
C*        KEY=4  THE B-MATRIX IS PUNCHED ON CARDS.                               BMF00140
C*        KEY=5  THE B-MATRIX IS WRITTEN ON FILE.                                BMF00150
C*                                                                               BMF00160
C*************************************************************************      BMF00170
C*    ARGUMENTS:      DEFINITION                  INPUT FOR     OUTPUT FOR       BMF00180
C*        BMAT    LINEAR TRANSFORMATION MATRIX     KEY=4,5       KEY=1,2       BMF00190
C*                DIMENSIONS LCOMB*NOFET                                         BMF00200
C*        LCOMB   NO. OF LINEAR COMBINATIONS       KEY=4,5       KEY=1,2,3     BMF00210
C*        NOFET   NO. OF FEATURES                  KEY=4,5       KEY=1,2,3     BMF00220
C*        VEC     VECTOR CONTAINING FEATURES       KEY=4,5       KEY=1,2,3     BMF00230
C*                USED IN OBTAINING B-MATRIX.                                    BMF00240
C*                DIMENSION=NOFET                                                BMF00250
C*************************************************************************      BMF00260
      DIMENSION BMAT(1),VEC(1)                                                   BMF00270
C     INCLUDE COMMON.LIST                                                        BMF00280
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,                  BMF00290
     *              HISFIL,HISKEY,TRFORM,ERRPTP,ERRKEY,MAPUNT,NOFILE,            BMF00300
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                         BMF00310
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                             BMF00320
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                       BMF00330
     *  CRDUNT,PRTUNT,RANDIO                                                     BMF00340
CSEND                                                                          BMF00350
      GO TO(10,20,20,30,40),KEY                                                BMF00360
   10 READ(21,100)LCOMB,NOFET,(VEC(I),I=1,NOFET)                               BMF00370
      IK=LCOMB*NOFET                                                           BMF00380
      READ(21,200)(BMAT(I),I=1,IK)                                             BMF00390
      GO TO 40                                                                 BMF00400
   20 REWIND BMFILE                                                            BMF00410
      READ(BMFILE)LCOMB,NOFET,(VEC(I),I=1,NOFET)                               BMF00420
      IF(KEY.EQ.3)RETURN                                                       BMF00430
      IK=LCOMB*NOFET                                                           BMF00440
      READ(BMFILE)(BMAT(I),I=1,IK)                                             BMF00450
      RETURN                                                                   BMF00460
   30 WRITE(PCHUNT,300)                                                        BMF00470
      WRITE(PCHUNT,400)LCOMB,NOFET,(VEC(I),I=1,NOFET)                          BMF00480
      IK=LCOMB*NOFET                                                           BMF00490
      WRITE(PCHUNT,500)(BMAT(I),I=1,IK)                                        BMF00500
      RETURN                                                                   BMF00510
   40 REWIND BMFILE                                                            BMF00520
      WRITE(BMFILE)LCOMB,NOFET,(VEC(I),I=1,NOFET)                              BMF00530
      IK=LCOMB*NOFET                                                           BMF00540
      WRITE(BMFILE)(BMAT(I),I=1,IK)                                            BMF00550
      RETURN                                                                   BMF00560
  100 FORMAT(6X,I2,5X,I2,3X,30I2)                                              BMF00570
  200 FORMAT(5A,5E15.6)                                                        BMF00580
  300 FORMAT('B-MATRIX    CARDS')                                              BMF00590
  400 FORMAT('COMB ',I2,'FEAT ',I2,'VEC',30I2)                                 BMF00600
  500 FORMAT('BMTRX',5E15.8)                                                   BMF00610
      END                                                                      BMF00620
```

FILE: BN14A1 FORTRAN A

```
      SUBROUTINE BN14A1(IFLD,INCHR,IBN)                                 BN100010
C     DAVID LEE SMITH  17 OCTOBER 1977.                                 HN100020
C     A SUBROUTINE TO CONVERT INTERNAL BINARY NUMBERS FROM              HN100030
C     THE INTEGER*4 FORM TO A STRING OF EBCDIC CHARACTERS.              BN100040
C                                                                       HN100050
C     THE PARAMETERS ARE:                                               BN100060
C          IFLD IS THE FIRST WORD OF A FIELD IN AN ARRAY IN WHICH       BN100070
C               TO STORE THE OUTPUT EBCDIC CHARACTERS, ONE PER          BN100080
C               WORD, IN A1 FORM, I.E., ONE CHARACTER PER WORD, WITH    BN100090
C               BLANK FILL TO THE RIGHT.                                HN100100
C          INCHR IS THE NUMBER OF EBCDIC CHARACTERS = THE NUMBER OF     HN100110
C               WORDS = THE WIDTH OF THE FIELD TO BE FILLED.            HN100120
C               INCHR SHOULD ALSO BE OF FORM INTEGER*4.                 BN100130
C           IBN  IS THE INPUT TO THIS ROUTINE, AN INTEGER IN INTERNAL   BN100140
C               BINARY FORM, POSITIVE OR NEGATIVE (TWO'S COMPLEMENT)    HN100150
C               IN THE LEGAL RANGE -2**31 TO (2**31)-1).                BN100160
      INTEGER*4 IFLD(20), INCHR, IBN, IDIG(10), IPL, IHMI               HN100170
      DATA IDIG / 1H0, 1H1, 1H2, 1H3, 1H4, 1H5, 1H6, 1H7, 1H8, 1H9 /    HN100180
      DATA IHMI / 1H- /                                                 BN100190
      DATA IHA / 1H8 /                                                  BN100200
      LAST = INCHR                                                      BN100210
      LBN = IBN                                                         HN100220
      NEG = 0                                                           HN100230
      IF ( LBN ) 10,30,30                                               HN100240
10    NEG = 1                                                           HN100250
      LBN = -LBN                                                        BN100260
      IF ( LBN ) 20,30,30                                               HN100270
20    IFLD(LAST) = IHA                                                  BN100280
      LBN = 214 748 364                                                 BN100290
      LAST = LAST - 1                                                   HN100300
30    IF (LAST .LE. 0 ) GO TO 60                                        BN100310
      IPT = LAST                                                        BN100320
      DO 50 I = 1, LAST                                                 BN100330
      INDEX = MOD( LBN, 10 )                                            BN100340
      IFLD(IPT) = IDIG( INDEX + 1 )                                     BN100350
      LBN = LBN / 10                                                    HN100360
      IPT = IPT - 1                                                     BN100370
50    CONTINUE                                                          BN100380
      IF ( NEG .NE. 0 ) IFLD(1) = IHMI                                  BN100390
60    RETURN                                                            BN100400
      END                                                               BN100410
```

FILE: BUFILL FORTRAN A

```
      SUBROUTINE BUFILL(IREC,IUNIT,MAXREC,/IBUF/,NRPDS,ENDTAP,IERR)
      IMPLICIT INTEGER (A-Z)
C
CC    BUFILL READS THE MSS TAPE ONE RECORD AT A TIME.  A MAXIMUM OF 10
CC    RECORDS PER DATA SET MAY BE PROCESSED AT A TIME
C
      DIMENSION IBUF(765)
C
      IERR = 0
      ENDTAP = 0
      K = 0
      DO 20  I=1,10
C
      MAX = MAXREC / 4
      READ(IUNIT,100,ERR=50,END=60)(IBUF(K+J),J=1,MAX)
  100 FORMAT(15(20A4))
      IREC = IREC + 1
      IF (IREC .GE. NRPDS) RETURN
      K = K + MAXREC/4
C
   20 CONTINUE
      RETURN
C
   50 WRITE(6,110)MAXREC,J
  110 FORMAT(1X,I4,' BYTES EXPECTED'/ 1X,I4, ' BYTES ON RECORD')
      IERR = -1
      RETURN
C
CC    ENCOUNTERED AN E-O-F
C
   60 ENDTAP = -1
      RETURN
C
      END
```

BUF00010 BUF00020 BUF00030 BUF00040 BUF00050 BUF00060 BUF00070 BUF00080 BUF00090 BUF00100 BUF00110 BUF00120 BUF00130 BUF00140 BUF00150 BUF00160 BUF00170 BUF00180 BUF00190 BUF00200 BUF00210 BUF00220 BUF00230 BUF00240 BUF00250 BUF00260 BUF00270 BUF00280 BUF00290 BUF00300 BUF00310 BUF00320 BUF00330 BUF00340 BUF00350

FILE: CLSCHK FORTRAN A

```
      SUBROUTINE CLSCHK(CLSDES,SUBDES,FLDSAV,VERTEX,SUBNO,                  CLS00010
     *                  NOFEAT,FETVEC,NOCLS,NOFLD,BMFLG,NOSUB)              CLS00020
C                                                                           CLS00030
      IMPLICIT INTEGER (A-H,O-Z)                                            CLS00040
CI                                                                          ICLS00050
CI----------------------------------------------------------------------   ICLS00060
CI----------------------------------------------------------------------   ICLS00070
CI                                                                          ICLS00080
C*                                                                          CLS00090
C*    THIS SUBROUTINE IS CALLED FROM REDSAV TO CHECK THE VALIDITY OF        CLS00100
C*    USER REQUESTS REGARDING SUBCLASSES, GROUPING AND CHANNELS             CLS00110
C*                                                                          CLS00120
CI----------------------------------------------------------------------   ICLS00130
CI----------------------------------------------------------------------   ICLS00140
C                                                                           CLS00150
C                                                                           CLS00160
C                                                                           CLS00170
C     INCLUDE COMBK1.LIST                                                   CLS00180
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,              COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,        COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),            COM00030
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                 COM00040
     *              GRPCHK(61),GROUPS(124)                                  COM00050
C$END                                                                       CLS00200
C                                                                           CLS00210
C     ----------------------------------------------------------------      CLS00220
C                                                                           CLS00230
      DIMENSION INVERT(30)                                                  CLS00240
      DIMENSION SETI(60)                                                    CLS00250
      DATA MAXFET/30/                                                       CLS00260
      DIMENSION FETVEC(30)                                                  CLS00270
C                                                                           CLS00280
      DIMENSION CLSDES(1),SUBDES(1),FLDSAV(4,NOFLD),VERTEX(1)               CLS00290
      DIMENSION SUBNO(1)                                                    CLS00300
C     ----------------------------------------------------------------      CLS00310
      IF(NOSUB2.LE.0)GO TO 40                                               CLS00320
      II = 0                                                                CLS00330
      IBIG = 0                                                              CLS00340
      DO 30 I=1,NOSUB2                                                      CLS00350
      J = SUBVC2(I)                                                         CLS00360
      IF(J.LE.IBIG .OR. J.GT.NOSUB)GO TO 10                                 CLS00370
      II = II+1                                                             CLS00380
      SUBVC2(II)=J                                                          CLS00390
      IBIG = J                                                              CLS00400
      GO TO 30                                                              CLS00410
   10 WRITE(6,20) J                                                         CLS00420
   20 FORMAT(//5X,'**CLSCHK** - REQUESTED SUBCLASS NO.',I3,' IS NOT AVA     CLS00430
     *ILABLE IN INPUT STATISTICS -- REQUEST IGNORED'/)                       CLS00440
   30 CONTINUE                                                              CLS00450
      NOSUB2=II                                                             CLS00460
      IF(NOSUB2.GT.0)GO TO 60                                               CLS00470
   40 NOSUB2=NOSUB                                                          CLS00480
      IF(NOSUB.GT.60)NOSUB2=60                                              CLS00490
      DO 50 I=1,NOSUB2                                                      CLS00500
   50 SUBVC2(I) = I                                                         CLS00510
C                                                                           CLS00520
C     CHECK THE GROUPS FOR VALIDITY                                         CLS00530
C     ----------------------------------------------------------------      CLS00540
C                                                                           CLS00550
   60 GRPTR = 0                                                             CLS00560
      IF (NOGRP.LE.0) GO TO 110                                             CLS00570
      II = 0                                                                CLS00580
      DO 100 I=1,NOGRP                                                      CLS00590
      JB = GRPDEX(I)+1                                                      CLS00600
      JE = JB+GROUPS(JB-1)-1                                                CLS00610
      GRPTR = GRPTR+1                                                       CLS00620
      II = GRPTR                                                            CLS00630
      DO 90 J=JB,JE                                                         CLS00640
      JJ = GROUPS(J)                                                        CLS00650
      IF(JJ.GT.NOSUB)GO TO 70                                               CLS00660
      GRPTR = GRPTR+1                                                       CLS00670
      GROUPS(GRPTR) = JJ                                                    CLS00680
      GO TO 90                                                              CLS00690
   70 WRITE (6,80)JJ,I                                                      CLS00700
   80 FORMAT(//5X,'**CLSCHK** - REQUESTED SUBCLASS NO.',I3,' FOR GROUP      CLS00710
     *NO.',I3,' IS NOT AVAILABLE IN INPUT STATISTICS FILE'/)                 CLS00720
   90 CONTINUE                                                              CLS00730
      GRPTR = GRPTR-1                                                       CLS00740
      IF (GRPTR.LT.II) GO TO 100                                            CLS00750
```

ORIGINAL PAGE IS
OF POOR QUALITY


```
      GRPTR = GRPTR+1                                                   CLS00760
      GROUPS(II) = GRPTR-II                                             CLS00770
      II = II+1                                                         CLS00780
      GRPDEX(II) = II                                                   CLS00790
      GRPNAM(II) = GRPNAM(I)                                            CLS00800
  100 CONTINUE                                                          CLS00810
      NOGRP = II                                                        CLS00820
C                                                                       CLS00830
C*    DELETE ALL GROUP SUBCLASSES                                       CLS00840
CC    ---------------------------------------------------------------CLS00850
C                                                                       CLS00860
  110 IC=NOSUB2                                                         CLS00870
      IF (NOGRP.LE.0) GO TO 170                                         CLS00880
      DO 140 I=1,NOGRP                                                  CLS00890
      JB = GRPDEX(I)+1                                                  CLS00900
      JE = JB+GROUPS(JB-1)-1                                            CLS00910
      DO 130 J=JB,JE                                                    CLS00920
      KK = 0                                                            CLS00930
      DO 120 K=1,NOSUB2                                                 CLS00940
      IF(SUBVC2(K).EQ.GROUPS(J))GO TO 120                               CLS00950
      KK = KK+1                                                         CLS00960
      SUBVC2(KK) = SUBVC2(K)                                            CLS00970
  120 CONTINUE                                                          CLS00980
  130 NOSUB2 = KK                                                       CLS00990
  140 CONTINUE                                                          CLS01000
C                                                                       CLS01010
CC    ADD BACK FIRST CLASS FROM EACH GROUP                              CLS01020
CC    ---------------------------------------------------------------CLS01030
C                                                                       CLS01040
  150 IC = NOSUB2                                                       CLS01050
      DO 160 I=1,NOGRP                                                  CLS01060
      NOSUB2 = NOSUB2+1                                                 CLS01070
      WAT=GRPDEX(I)                                                     CLS01080
      II=GROUPS(WAT+1)                                                  CLS01090
      SUBDES(II)=GRPNAM(I)                                              CLS01100
  160 SUBVC2(NOSUB2)=II                                                 CLS01110
C                                                                       CLS01120
CC    CONSTRUCT GROUP FOR EACH SUBCLASS NOT EXPLICITLY GROUPED          CLS01130
CC    ---------------------------------------------------------------CLS01140
C                                                                       CLS01150
  170 IF (IC.LE.0) GO TO 190                                            CLS01160
      DO 180 I=1,IC                                                     CLS01170
      NOGRP = NOGRP+1                                                   CLS01180
      GRPTR = GRPTR+1                                                   CLS01190
      GRPDEX(NOGRP) = GRPTR                                             CLS01200
      GROUPS(GRPTR) = 1                                                 CLS01210
      II = SUBVC2(I)                                                    CLS01220
      GRPNAM(NOGRP) = SUBDES(II)                                        CLS01230
      GRPTR = GRPTR+1                                                   CLS01240
      GROUPS(GRPTR) = II                                                CLS01250
  180 CONTINUE                                                          CLS01260
C                                                                       CLS01270
CC    ARRANGE 'SUBVC2' IN ORDER                                         CLS01280
CC    ---------------------------------------------------------------CLS01290
C                                                                       CLS01300
  190 CALL ORDER(SUBVC2,NOSUB2)                                         CLS01310
C*    USE SUBPTR STORAGE TEMPORARILY TO SEE IF ANY ENTIRE CLASSES       CLS01320
C*    HAVE BEEN ELIMINATED.                                             CLS01330
      IK=0                                                              CLS01340
      DO 191 I=1,NOCLS                                                  CLS01350
      K=SUBNO(I)                                                        CLS01360
      SUBNO(I)=0                                                        CLS01370
      DO 191 L=1,K                                                      CLS01380
      IK=IK+1                                                           CLS01390
  191 SUBPTR(IK)=I                                                      CLS01400
      KNT=0                                                             CLS01410
      LC=1                                                              CLS01420
      DO 194 I=1,NOSUB2                                                 CLS01430
      IK=SUBVC2(I)                                                      CLS01440
      IC=SUBPTR(IK)                                                     CLS01450
      IF(IC.EQ.LC)GO TO 193                                             CLS01460
      SUBNO(LC)=KNT                                                     CLS01470
      LC=IC                                                             CLS01480
      KNT=1                                                             CLS01490
      GO TO 194                                                         CLS01500
  193 KNT=KNT+1                                                         CLS01510
  194 CONTINUE                                                          CLS01520
      SUBNO(LC)=KNT                                                     CLS01530
      DO 192 I=1,NOCLS                                                  CLS01540
```

FILE: CLSCHK FORTRAN A

```
  192 SETI(I)=-1                                                          CLS01550
      IK=0                                                                CLS01560
      DO 195 I=1,NOCLS                                                    CLS01570
      IF(SUBNO(I).EQ.0)GO TO 195                                          CLS01580
      IK=IK+1                                                             CLS01590
      SETI(I)=IK                                                          CLS01600
      SUBNO(IK)=SUBNO(I)                                                  CLS01610
      CLSDES(IK)=CLSDES(I)                                                CLS01620
  195 CONTINUE                                                            CLS01630
       NOCLS2=IK                                                          CLS01640
C*                                                                        CLS01650
C*    INITIALIZE SUBPTR ARRAY FOR REDSAV - THIS ARRAY IS AN INDEX ARRAY   CLS01660
C*    THAT CONTAINS THE NEW INDEX FOR EACH SUBCLASS TO BE USED IN THE     CLS01670
C*    PROCESSING.                                                         CLS01680
C*                                                                        CLS01690
      DO 200 I=1,NOSUB                                                    CLS01700
  200 SUBPTR(I)=0                                                         CLS01710
      DO 205 I=1,NOSUB2                                                   CLS01720
      IK = SUBVC2(I)                                                      CLS01730
      SUBDES(I)=SUBDES(IK)                                                CLS01740
  205 SUBPTR(IK)=I                                                        CLS01750
C*    RESET GROUPED SUBCLASSES IN SUBPTR                                  CLS01760
C*                                                                        CLS01770
      IF(NOGRP.EQ.0)GO TO 235                                             CLS01780
      DO 220 I=1,NOGRP                                                    CLS01790
       JB = GRPNEX(I)+1                                                   CLS01800
      JE = JB + GROUPS(JB-1)-1                                            CLS01810
      DO 220 J=1,NOSUB2                                                   CLS01820
      IF(SUBVC2(J).NE.GROUPS(JB))GO TO 220                                CLS01830
      DO 210 K=JB,JE                                                      CLS01840
      IK=GROUPS(K)                                                        CLS01850
      SUBPTR(IK)=J                                                        CLS01860
  210 CONTINUE                                                            CLS01870
  220 CONTINUE                                                            CLS01880
  235 CONTINUE                                                            CLS01890
C                                                                         CLS01900
CC    ------------------------------------------------------------------CLS01910
C     ------------------------------------------------------------------CLS01920
C                                                                         CLS01930
C*                                                                        CLS01940
C*    NOW CHECK ON CHANNELS REQUESTED                                     CLS01950
C*                                                                        CLS01960
      IF(NOFET2.GT.0)GO TO 230                                            CLS01970
      DO 225 I=1,NOFEAT                                                   CLS01980
  225 FETVC2(I)=FETVEC(I)                                                 CLS01990
      NOFET2=NOFEAT                                                       CLS02000
  230 CALL ORDER(FETVC2,NOFET2)                                           CLS02010
C                                                                         CLS02020
CC    SET UP INVERT TABLE                                                 CLS02030
C     ------------------------------------------------------------------CLS02040
C                                                                         CLS02050
      DO 240 I=1,MAXFET                                                   CLS02060
  240 INVERT(I) = 0                                                       CLS02070
      DO 250 I=1,NOFEAT                                                   CLS02080
      WAT=FETVEC(I)                                                       CLS02090
  250 INVERT(WAT) = I                                                     CLS02100
C                                                                         CLS02110
C      CHECK VALIDITY OF REQUESTED CHANNELS                               CLS02120
C     ------------------------------------------------------------------CLS02130
C                                                                         CLS02140
      II = 0                                                              CLS02150
      IBIG = 0                                                            CLS02160
      DO 280 I=1,NOFET2                                                   CLS02170
      J =FETVC2(I)                                                        CLS02180
      IF (J.GT.MAXFET.OR.J.LE.0) GO TO 260                                CLS02190
      K = INVERT(J)                                                       CLS02200
      IF (K.LE.IBIG) GO TO 260                                            CLS02210
      II = II+1                                                           CLS02220
      FETVC2(II) = J                                                      CLS02230
      IBIG = K                                                            CLS02240
      GO TO 280                                                           CLS02250
  260 WRITE(6,270)  J , (FETVEC(K), K=1,NOFEAT)                           CLS02260
  270 FORMAT(//// 5X,'***** CLSFY/FETCHK ---  CHANNEL ',I3,'  NOT IN TRACLS02270
     1INING DATA ---  TRAINING DATA CHANNELS ARE ...' //                  CLS02280
     2  (5X,30I4/) )                                                     CLS02290
      IF( RMFLG .GT. 0)  GO TO 272                                        CLS02300
      WRITE(6,271)  J                                                     CLS02310
  271 FORMAT(// 5X,'*****  CHANNEL ',I3,'  IGNORED (NOT USED) IN CLASSIFCLS02320
     1ICATION'/)                                                          CLS02330
```

FILE: CLSCHK FORTRAN A

```
      GO TO 280                                                          CLS02340
  272 WRITE(6,273) (FETVC2(K),K=1,NOFET2)                                CLS02350
  273 FORMAT(// 5X,'*****  B-MATRIX CHANNELS MUST BE EQUAL TO OR A SUB-SCLS02360
     1ET OF AVAILABLE TRAINING DATA CHANNELS --- THE INPUT B-MATRIX CHANCLS02370
     2NEL SET IS ...'// (5X,3014//) )                                    CLS02380
      WRITE(6,274)                                                       CLS02390
  274 FORMAT(///// 5X,'*****  TERMINATING PROGRAM EXECUTION  FROM FETCHKCLS02400
     1*****'/1H1)                                                        CLS02410
      CALL EXIT                                                          CLS02420
  280 CONTINUE                                                           CLS02430
      NOFET2 = II                                                        CLS02440
C                                                                        CLS02450
C     SET UP REVISED INVERT TABLE                                        CLS02460
C     ---------------------------------------------------------------------CLS02470
C                                                                        CLS02480
      DO 290 I=1,MAXFET                                                  CLS02490
  290 INVERT(I) = 0                                                      CLS02500
      DO 300 I=1,NOFET2                                                  CLS02510
      WAT=FETVC2(I)                                                      CLS02520
  300 INVERT(WAT) = I                                                    CLS02530
C*                                                                       CLS02540
C*    SET UP CLSVC2 ARRAY SO THAT IT CONTAINS THE CLASS NO. TO WHICH THECLS02550
C*    CORRESPONDING SUBCLASS BELONGS.                                    CLS02560
      J=0                                                                CLS02570
      DO 305 I=1,NOCLS2                                                  CLS02580
      IK=SUBNO(I)                                                        CLS02590
      DO 305 K=1,IK                                                      CLS02600
      J=J+1                                                              CLS02610
  305 CLSVC2(J)=I                                                        CLS02620
C*                                                                       CLS02630
C*    SAVE FIELD DESCRIPTIONS FOR CLASSES AND SUBCLASSES TO BE USED.     CLS02640
C*                                                                       CLS02650
      NOFLD2=0                                                           CLS02660
      IV=0                                                               CLS02670
      JV=0                                                               CLS02680
      DO 330 I=1,NOFLD                                                   CLS02690
      JB=FLDSAV(3,I)                                                     CLS02700
C*    ARE FIELDS ASSOCIATED WITH SUBCLASSES                              CLS02710
      IF(JB.NE.0)GO TO 306                                               CLS02720
      IC=FLDSAV(2,I)                                                     CLS02730
C*    HAS THIS CLASS BEEN ELIMINATED                                     CLS02740
      NC=SETI(IC)                                                        CLS02750
      IF(NC.EQ.-1)GO TO 320                                              CLS02760
      NSUB=0                                                             CLS02770
      GO TO 307                                                          CLS02780
  306 CONTINUE                                                           CLS02790
C*    HAS THIS SUBCLASS BEEN ELIMINATED                                  CLS02800
      NSUB = SUBPTR(JB)                                                  CLS02810
      IF(NSUB.EQ.0)GO TO 320                                             CLS02820
      NC=CLSVC2(NSUB)                                                    CLS02830
  307 CONTINUE                                                           CLS02840
      NOFLD2=NOFLD2+1                                                    CLS02850
      FLDSAV(1,NOFLD2)=FLDSAV(1,I)                                       CLS02860
      FLDSAV(2,NOFLD2)=NC                                                CLS02870
      FLDSAV(3,NOFLD2)=NSUB                                              CLS02880
      FLDSAV(4,NOFLD2)=FLDSAV(4,I)                                       CLS02890
C*                                                                       CLS02900
C*    NOW SAVE VERTICES                                                  CLS02910
      NV=FLDSAV(4,I)*2                                                   CLS02920
      DO 310 J=1,NV                                                      CLS02930
      IV=IV+1                                                            CLS02940
  310 VERTEX(IV) = VERTEX(JV+J)                                          CLS02950
      JV=JV+NV                                                           CLS02960
      GO TO 330                                                          CLS02970
  320 JV = JV + FLDSAV(4,I)*2                                            CLS02980
  330 CONTINUE                                                           CLS02990
      TOTVT2=IV/2                                                        CLS03000
      RETURN                                                             CLS03010
       END                                                               CLS03020
```

FILE: CLSHIS FORTRAN A

```
      SUBROUTINE CLSHIS(TALLY,HISBUF,TTL,XSIZ,XHGH,XLOW,YSIZ,          CLS00010
     *NOHIST,FLDPTS,HISVEC)                                            CLS00020
C                                                                       CLS00030
      IMPLICIT INTEGER (A-H,O-Z)                                        CLS00040
      REAL XSCALE,XSHFT                                                 CLS00050
C                                                                       CLS00060
C     INCLUDE COMRK6,LIST                                               CLS00070
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         CLS00080
     *             HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,    CLS00090
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                CLS00100
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                    CLS00110
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              CLS00120
     *  CRDUNT,PRTUNT,RANDIO                                            CLS00130
CSEND                                                                   CLS00140
C                                                                       CLS00150
      DIMENSION HISVEC(30)                                              CLS00160
      DIMENSION TALLY(NOHIST,XSIZ),HISBUF(XSIZ),XAXIS(11)               CLS00170
      LOGICAL*1 DUMM(4),SYMM(4)                                         CLS00180
      EQUIVALENCE (SYM,SYMM(1)),(DUM,DUMM(1))                           CLS00190
C                                                                       CLS00200
      DATA  STAR /'*'/, BLANK/' '/                                      CLS00210
      DATA  DOLBCD/'$'/, NUMIC/ZF0/, ALPHA1/ZC0/                        CLS00220
C                                                                       CLS00230
CCCC  ------------------------------------------------------------------CLS00240
CCCC  ------------------------------------------------------------------CLS00250
C                                                                       CLS00260
C                                                                       CLS00270
      GO = 1                                                            CLS00280
C                                                                       CLS00290
CCCC  INIZ                                                              CLS00300
CCCC  ------------------------------------------------------------------CLS00310
C                                                                       CLS00320
   10 JPTCNT = ( PAGSIZ-8)/(YSIZ + 10 )                                 CLS00330
      DSIZ = (XSIZ+8)/10 + 1                                            CLS00340
      XSCALE = FLOAT(XLOW-XHGH)/(XSIZ-1)                                CLS00350
      XSHFT = FLOAT(XSIZ*XHGH-XLOW)/(XSIZ-1)                            CLS00360
      DO 20 I=1,DSIZ                                                    CLS00370
   20 XAXIS(I) = (10*I-9)*XSCALE + XSHFT + 0.501                        CLS00380
C                                                                       CLS00390
      JCNT = JPTCNT                                                     CLS00400
      DO 160 JFEAT=1,NOHIST                                             CLS00410
      IF (JCNT.LT.JPTCNT) GO TO 40                                      CLS00420
      WRITE (6,HEAD)                                                    CLS00430
      IF(GO.EQ.4) WRITE(6,11)                                           CLS00440
   11 FORMAT(' DATA TR')                                                CLS00450
      IF(GO.EQ.1) WRITE(6,200) TTL                                      CLS00460
      IF(GO.EQ.0) WRITE(6,201) TTL                                      CLS00470
  200 FORMAT(T63,'HISTOGRAM'/T59,'-----------------'//T55,'TRAINING SUBC CLS00480
     *LASS      ',A4/T53,'-------------------------------'/)            CLS00490
  201 FORMAT(T63,'HISTOGRAM'/T59,'-----------------'//T55,'TRAINING FIEL CLS00500
     *D         ',A4/T53,'                           '/)                CLS00510
      IF (GO .EQ. 2) WRITE(6,145)          TTL , FLDPTS                 CLS00520
      IF (GO .EQ. 3)  WRITE(6,147)    TTL                               CLS00530
      IF(GO.EQ.0) WRITE(6,30) FLDPTS, TITLE                             CLS00540
   30 FORMAT(T48,'(NO. SAMPLES= ',I7,' , SUBCLASS= ',A4,')'/)           CLS00550
      JCNT = 0                                                          CLS00560
C                                                                       CLS00570
CCCC  SCALE AND PRINT THE HISTOGRAM                                     CLS00580
CCCC  ------------------------------------------------------------------CLS00590
C                                                                       CLS00600
   40 MAX = 0                                                           CLS00610
      YSCALE = 1                                                        CLS00620
      JCNT = JCNT +1                                                    CLS00630
      DO 50 J=1,XSIZ                                                    CLS00640
      IF (TALLY(JFEAT,J) .GT. MAX) MAX = TALLY(JFEAT,J)                 CLS00650
   50 CONTINUE                                                          CLS00660
      IF(MAX .GT. YSIZ) YSCALE = (MAX+(YSIZ-1))/YSIZ                    CLS00670
      K = HISVEC(JFEAT)                                                 CLS00680
      IF(GO.EQ.4) GO TO 62                                              CLS00690
      WRITE(6,60) K                                                     CLS00700
   60 FORMAT(' CHANNEL ',I2/1X,'----------')                            CLS00710
   62 CONTINUE                                                          CLS00720
      WRITE (6,70)YSCALE                                                CLS00730
   70 FORMAT('0',T5,'EACH * REPRESENTS',I8,'  POINT(S).'/)               CLS00740
C                                                                       CLS00750
      DO 120 JY=1,YSIZ                                                  CLS00760
      JH = (YSIZ-(JY-1))*YSCALE                                         CLS00770
      IK = JH - YSCALE                                                  CLS00780
      JEMP=XSIZ                                                         CLS00790
```

FILE: CLSHIS FORTRAN A

```
      DO 100 IAZ=1,JEMP                                                    CLS00800
      I=IAZ                                                                CLS00810
      HISBUF(I) = BLANK                                                    CLS00820
      JK = TALLY(JFEAT,I)                                                  CLS00830
      SYM = STAR                                                           CLS00840
      IF (JK.GE.JM) GO TO 90                                               CLS00850
      IF (JK.LE.IK) GO TO 100                                              CLS00860
      JK = JK-IK                                                           CLS00870
      ZONE = NUMIC                                                         CLS00880
      IF (JK.LT.10) GO TO 80                                               CLS00890
      ZONE = ALPHA1                                                        CLS00900
      JK = JK-9                                                            CLS00910
      IF (JK.LT.10) GO TO 80                                               CLS00920
      SYM = DOLBCD                                                         CLS00930
      GO TO 90                                                             CLS00940
   80 DUM = ZONE + JK                                                      CLS00950
      SYMM(1)=DUMM(4)                                                      CLS00960
   90 HISBUF(I) = SYM                                                      CLS00970
  100 CONTINUE                                                             CLS00980
      WRITE (6,110)JM,(HISBUF(I),I=1,XSIZ)                                CLS00990
  110 FORMAT(1X,I6,' I',1X,112A1)                                          CLS01000
  120 CONTINUE                                                             CLS01010
      WRITE (6,130)(XAXIS(I),I=1,OSIZ)                                     CLS01020
  130 FORMAT(10X,10('+----------'),'+',/9X,11(I3,7X))                      CLS01030
      WRITE (6,140)                                                        CLS01040
  140 FORMAT('0')                                                          CLS01050
  145 FORMAT(T63,'HISTOGRAM'/T59,18(1H-)//T61,'FIELD  ',2X,A4              CLS01060
     *    / T57, 22(1H-),// T56, '( NO. SAMPLES : ', I7, ')' )             CLS01070
  147 FORMAT(T63,'HISTOGRAM',/T59,18(1H-)//T65,A4,                         CLS01080
     *    / T57, 22(1H-))                                                  CLS01090
      DO 150 I=1,XSIZ                                                      CLS01100
  150 TALLY(JFEAT,I) = 0                                                   CLS01110
  160 CONTINUE                                                             CLS01120
      CALL SETMRG( 66,1,65 )                                               CLS01130
      RETURN                                                               CLS01140
C     ------------------------------------------------------------------   CLS01150
C                                                                          CLS01160
      ENTRY FLDHIS(TALLY,HISBUF,TTL,XSIZ,XHGH,XLOW,YSIZ,                   CLS01170
     *NOHIST,FLDPTS,TITLE,HISVEC)                                          CLS01180
C                                                                          CLS01190
      GO = 0                                                               CLS01200
      GO TO 10                                                             CLS01210
C     ------------------------------------------------------------------   CLS01220
C                                                                          CLS01230
      ENTRY HSTGRM(TALLY,HISBUF,TTL,PRINT,XSIZ,XHGH,XLOW,YSIZ,             CLS01240
     *NOHIST,FLDPTS,HISVEC)                                                CLS01250
C                                                                          CLS01260
      GO = PRINT                                                           CLS01270
C                                                                          CLS01280
      GO TO 10                                                             CLS01290
C                                                                          CLS01300
      ENTRY COMHST(TALLY,HISBUF,TTL,NOHIST,HISVEC,XSIZ,XHGH,XLOW,YSIZ)     CLS01310
      GO =4                                                                CLS01320
      DO 12 I=1,NOHIST                                                     CLS01330
      DO 61 J=1,50                                                         CLS01340
      ITEM=TALLY(I,J)                                                      CLS01350
      JT=102-J                                                             CLS01360
      TALLY(I,J)=TALLY(I,JT)                                               CLS01370
   61 TALLY(I,JT)=ITEM                                                     CLS01380
   12 CONTINUE                                                             CLS01390
      GO TO 10                                                             CLS01400
       END                                                                 CLS01410
                                                                           CLS01420
                                                                           CLS01430
```

FILE: CMERR FORTRAN A

```
      SUBROUTINE CMERR                                                  CME00010
      WRITE(6,100)                                                      CME00020
  100 FORMAT(' ERROR HAS OCCURRED!')                                    CME00030
      CALL EXIT                                                         CME00040
      RETURN                                                            CME00050
      END                                                               CME00060
```

FILE: CRDSTA FORTRAN A

```
      SUBROUTINE CRDSTA(ARRAY,TOP)                                          CRD00010
      IMPLICIT INTEGER (A-Z)                                                CRD00020
C     INCLUDE COMMR1.LIST                                                   CRD00030
C     INCLUDE COMMR6.LIST                                                   CRD00040
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,              CRD00050
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,        CRD00060
     *             FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),             CRD00070
     *             KEEPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                  CRD00080
     *             GRPCHK(61),GROUPS(124)                                   CRD00090
      COMMON/GLOBAL/HEAD(43),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             CRD00100
     *             HISFIL,HISKEY,TRFORM,FRIPTR,ERRKEY,MAPUNT,NOFILE,        CRD00110
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                    CRD00120
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                        CRD00130
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                  CRD00140
     *  CRDUNT,PRTUNT,HANDIO                                                CRD00150
C$END                                                                       CRD00160
      COMMON /PASSB/ NOCLS,NOSUB,NOFEAT,NOFLD,TOTVRT,                       CRD00170
     *               FETVEC(30),FLDSV1,CLSID1,VARSIZ                        CRD00180
      DIMENSION ARRAY(1)                                                    CRD00190
C                                                                           CRD00200
C                                                                           CRD00210
C                                                                           CRD00220
C     READ FROM CARDS KEY WORDS TO BE USED IN BASE ADDRESSES                CRD00230
C                                                                           CRD00240
      READ(CRDUNT,200)NOCLS,NOSUB,NOFEAT,NOFLD,TOTVRT                       CRD00250
  200 FORMAT(5X,I4,8X,I2,8X,I2,7X,I3,8X,I4)                                 CRD00260
      READ(CRDUNT,210)(FETVEC(I),I=1,NOFEAT)                                CRD00270
  210 FORMAT(10X,3012)                                                      CRD00280
C                                                                           CRD00290
C     COMPUTE BASE ADDRESSES                                                CRD00300
C                                                                           CRD00310
      VARSIZ = NOFEAT * (NOFEAT + 1) / 2                                    CRD00320
      MAXFLD = NOFLD                                                        CRD00330
      COVAR1 = 1                                                            CRD00340
      AVAR1 = COVAR1 + VARSIZ                                               CRD00350
      MAXCLS = NOSUB                                                        CRD00360
      CLSID1 = AVAR1 + NOFEAT                                               CRD00370
      SUBNO1 = CLSID1 + NOCLS                                               CRD00380
      SUBDS1 = SUBNO1 + NOCLS                                               CRD00390
      FLDSV1 = SUBDS1 + NOSUB                                               CRD00400
      VERTX1 = FLDSV1 + NOFLD*4                                             CRD00410
      TIPTOP = VERTX1 +   TOTVRT                                            CRD00420
      BADCOR = TOP - TIPTOP                                                 CRD00430
      IF (BADCOR .LT. 0) GO TO 100                                          CRD00440
C                                                                           CRD00450
      CALL RDMODK(ARRAY(AVAR1),ARRAY(COVAR1),ARRAY(CLSID1),ARRAY(SUBNO1)    CRD00460
     *           ,ARRAY(SUBDS1),ARRAY(FLDSV1),ARRAY(VERTX1),ARRAY(1))       CRD00470
C                                                                           CRD00480
      GO TO 90                                                              CRD00490
  100 WRITE(PRTUNT,190)                                                     CRD00500
  190 FORMAT(' EXCEEDED CORE LIMITS, REDUCE NO. OF TRAINING CLASSES OR F    CRD00510
     *EATURES'/ ' EXITING FROM CRDSTA')                                     CRD00520
      CALL CMERR                                                            CRD00530
   90 CONTINUE                                                              CRD00540
      RETURN                                                                CRD00550
      END                                                                   CRD00560
```

FILE: FDLINT FOPTRAN A

```
      SUBROUTINE FDLINT(FIELD,NPTS,FL,YLINE,NSAMP,JJ)                    FDL00010
C                                                                        FDL00020
C     THIS SUBROUTINE WILL RETURN THE PIXEL NUMBERS OF THOSE             FDL00030
C     PIXELS ON A A GIVEN LINE THAT ARE CONTAINED WITHIN THE             FDL00040
C     BOUNDARIES OF A NON -RECTANGULAR FIELD                             FDL00050
C                                                                        FDL00060
C                                                                        FDL00070
C     INPUT    FIELD - NONE-RECTANGULAR FIELD TABLE                      FDL00080
C                      ALL THE VERTICES MUST BE IN CLOCKWISE             FDL00090
C                      ORDER AND THE LAST VERTEX HAS TO BE EQUAL         FDL00100
C                      TO THE FIRST VERTEX FOR FIELD CLOSURE             FDL00110
C                      THE FIRST VERTEX MUST HAVE MINIMUM                FDL00120
C                      PIXEL VALUE                                       FDL00130
C              NPTS  - NO OF POINTS OF THE N-R FIELD                     FDL00140
C              YLINE - SCAN LINE NUMBER                                  FDL00150
C                                                                        FDL00160
C     OUTPUT   FL    - ARRAY CONTAINING THE ORDERED PIXEL INTERCEPTS     FDL00170
C              NSAMP - NO OF SAMPLES CONTAINED IN THE FIELD OF           FDL00180
C                      A GIVEN SCAN LINE                                 FDL00190
C              JJ    - THE LENGTH OF THE ARRAY FL                        FDL00200
C                                                                        FDL00210
      DIMENSION FIELD(2,NPTS),FL(8)                                      FDL00220
      INTEGER X1,X2,Y1,Y2,XX,FL,FIELD,YLINE                              FDL00230
      INTEGER XNM1,YNM1,XNP2,YNP2                                        FDL00240
      IF(NPTS.EQ.2)GO TO 35                                              FDL00250
C*    ONE VERTEX FIELD                                                   FDL00260
      L= YLINE                                                           FDL00270
      DO 7 N = 1,8                                                       FDL00280
    7 FL(N) = 0                                                          FDL00290
      NPTSE = NPTS-1                                                     FDL00300
      I = 1                                                              FDL00310
      JJ = 0                                                             FDL00320
  100 X1=FIELD(1,I)                                                      FDL00330
      Y1=FIELD(2,I)                                                      FDL00340
      J = I+1                                                            FDL00350
      X2=FIELD(1,J)                                                      FDL00360
      Y2=FIELD(2,J)                                                      FDL00370
      IF ( I .EQ. 1 ) GO TO 200                                          FDL00380
      IM1 = I-1                                                          FDL00390
      XNM1=FIELD(1,IM1)                                                  FDL00400
      YNM1=FIELD(2,IM1)                                                  FDL00410
      GO TO 300                                                          FDL00420
  200 XNM1=FIELD(1,NPTSE)                                                FDL00430
      YNM1=FIELD(2,NPTSE)                                                FDL00440
  300 IP1 = I+1                                                          FDL00450
      XNP1=FIELD(1,IP1)                                                  FDL00460
      YNP1=FIELD(2,IP1)                                                  FDL00470
      IF ( I .EQ. NPTSE) GO TO 400                                       FDL00480
      IP2 = I+2                                                          FDL00490
      XNP2=FIELD(1,IP2)                                                  FDL00500
      YNP2=FIELD(2,IP2)                                                  FDL00510
      GO TO 500                                                          FDL00520
  400 XNP2=FIELD(1,2)                                                    FDL00530
      YNP2=FIELD(2,2)                                                    FDL00540
  500 IF ( Y1 .EQ. Y2 ) GO TO 1000                                       FDL00550
      IF((L.EQ.Y2).AND.(Y2.EQ.YNP2)) GO TO 2000                          FDL00560
      IF((L.EQ.Y1).AND.(Y1.EQ.YNM1)) GO TO 2000                          FDL00570
      RL = L                                                             FDL00580
        RX1 = X1                                                         FDL00590
        RX2 = X2                                                         FDL00600
        RY1 = Y1                                                         FDL00610
        RY2 = Y2                                                         FDL00620
        RXX =(((RL-RY1)*(RX2-RX1))/(RY2-RY1))+RX1                        FDL00630
        XX = RXX+.5                                                      FDL00640
      IF(Y1.LT.Y2)  GO TO 510                                            FDL00650
      XX=RXX                                                             FDL00660
      IF((RXX-XX).GT..5) XX=XX+1                                         FDL00670
  510 CONTINUE                                                           FDL00680
      IF ((XX.GE. X1) .AND. (XX .LE. X2) ) GO TO 600                     FDL00690
      IF ((XX.LE. X1) .AND. (XX .GE. X2) ) GO TO 600                     FDL00700
 2000 I = I+1                                                            FDL00710
      IF ( I .GT. NPTSE ) GO TO 5                                        FDL00720
      GO TO 100                                                          FDL00730
  600 IF(L.LE.Y1.AND.L.GE.Y2) GO TO 700                                  FDL00740
      IF(L.LE.Y2.AND.L.GE.Y1) GO TO 700                                  FDL00750
      GO TO 2000                                                         FDL00760
  700 JJ = JJ+1                                                          FDL00770
      FL(JJ) = XX                                                        FDL00780
      IF ( JJ .EQ. 1 ) GO TO 2000                                        FDL00790
```

FILE: FDLINT FORTRAN A

```
      IF ( I .NE. NPTSE ) GO TO 3000                                    FDL00800
      IF(L.NE.Y2) GO TO 3000                                            FDL00810
      XNM1=X1                                                           FDL00820
      YNM1=Y1                                                           FDL00830
      X1=X2                                                             FDL00840
      Y1=Y2                                                             FDL00850
      X2=FIELD(1,2)                                                     FDL00860
      Y2=FIELD(2,2)                                                     FDL00870
      GO TO 3001                                                        FDL00880
 3000 IF ( L .NE. Y1 ) GO TO 2000                                       FDL00890
 3001   IF ((Y1.LT. YNM1) .AND. (Y1 .GT. Y2 )) GO TO 4000               FDL00900
      IF ((Y1 .GT. YNM1) .AND. (Y1 .LT. Y2)) GO TO 4000                 FDL00910
      GO TO 2000                                                        FDL00920
 4000 FL(JJ) = 0                                                        FDL00930
      JJ = JJ-1                                                         FDL00940
      GO TO 2000                                                        FDL00950
 1000 IF(L.NE.Y1) GO TO 2000                                            FDL00960
      IF(X1.GT.X2) GO TO 5000                                           FDL00970
      IF(YNM1.LT.Y1) GO TO 6000                                         FDL00980
      IF ( YNP2 .GT. Y2 ) GO TO 7000                                    FDL00990
      JJ = JJ+1                                                         FDL01000
      FL(JJ) = X1                                                       FDL01010
      GO TO 2000                                                        FDL01020
 7000 JJ = JJ+1                                                         FDL01030
      FL(JJ) = X1                                                       FDL01040
      MM = JJ+1                                                         FDL01050
      FL(MM) = X2                                                       FDL01060
      JJ = MM                                                           FDL01070
      GO TO 2000                                                        FDL01080
 6000 IF ( YNP2 .LT. Y2 ) GO TO 2000                                    FDL01090
      JJ = JJ+1                                                         FDL01100
      FL(JJ) = X2                                                       FDL01110
      GO TO 2000                                                        FDL01120
 5000 IF ( YNM1 .LT. Y1 ) GO TO 9000                                    FDL01130
      IF ( YNP2 .GT. Y2 ) GO TO 2000                                    FDL01140
      JJ = JJ+1                                                         FDL01150
      FL(JJ) = X2                                                       FDL01160
      IF(NPTSE.EQ.2)FL(JJ)=X1                                           FDL01170
      GO TO 2000                                                        FDL01180
 9000 IF ( YNP2 .GT. Y2 ) GO TO 8000                                    FDL01190
      JJ = JJ+1                                                         FDL01200
      FL(JJ) = X1                                                       FDL01210
      MM = JJ+1                                                         FDL01220
      FL(MM) = X2                                                       FDL01230
      JJ = MM                                                           FDL01240
      GO TO 2000                                                        FDL01250
 8000 JJ = JJ+1                                                         FDL01260
      FL(JJ) = X1                                                       FDL01270
      GO TO 2000                                                        FDL01280
    5 NPTS1 = JJ-1                                                      FDL01290
      DO 29 NI = 1,NPTS1                                                FDL01300
      NP1 = NI+1                                                        FDL01310
      DO 29 NJ = NP1,JJ                                                 FDL01320
      IF ( FL(NI) - FL(NJ) ) 29,29,28                                   FDL01330
   28 NTEMP = FL(NI)                                                    FDL01340
      FL(NI) = FL(NJ)                                                   FDL01350
      FL(NJ) = NTEMP                                                    FDL01360
   29 CONTINUE                                                          FDL01370
      NSAMP = 0                                                         FDL01380
      DO 30 N = 1,JJ,2                                                  FDL01390
      NN = N+1                                                          FDL01400
      NSAMP = NSAMP+(FL(NN) -FL(N)+1)                                   FDL01410
   30 CONTINUE                                                          FDL01420
      RETURN                                                            FDL01430
   35 IF(YLINE.NE.FIELD(2,1))RETURN                                     FDL01440
      FL(1)=FIELD(1,1)                                                  FDL01450
      FL(2)=FIELD(1,1)                                                  FDL01460
      NSAMP=1                                                           FDL01470
      JJ=2                                                              FDL01480
      RETURN                                                            FDL01490
      END                                                               FDL01500
```

FILE: FIND12 FORTRAN A

```
      FUNCTION FIND12(CARD,COL,VECTOR)                                  FIN00010
C     IMPLICIT INTEGER (A-H,O-Z)                                        FIN00020
C     |-----------------------------------------------------------------|FIN00030
C     |                                                                 |FIN00040
C     |-----------------------------------------------------------------|FIN00050
C     |                                                                 |FIN00060
C     |   CALL..    J = FIND12(CARD,COL,VECTOR)                         |FIN00070
C     |                                                                 |FIN00080
C     |   ARGS..    CARD   = BCD BUFFER                                 |FIN00090
C     |             COL    = PTR TO POSTION IN 'CARD'                   |FIN00100
C     |             VECTOR = VEC CONTAINING N SYMBOLS                   |FIN00110
C     |                      TO BE LOCATED IN CARD                      |FIN00120
C     |                      (N IS GIVEN IN VECTOR(1)                   |FIN00130
C     |                      EG.   /2,'$','='/      )                   |FIN00140
C     |                                                                 |FIN00150
C     |   REQUIRES. NOR                                                 |FIN00160
C     |                                                                 |FIN00170
C     |   PURPOSE.. USED TO LOCATE SPECIAL SYMBOLS IN 'CARD'            |FIN00180
C     |                                                                 |FIN00190
C     |   RETURNS.. COL    = PTS AT SYMBOL ( IF LOCATED)                |FIN00200
C     |             J      = PTS AT SYMBOL LOCATED IN 'VECTOR'          |FIN00210
C     |                                                                 |FIN00220
C     |-----------------------------------------------------------------|FIN00230
C     |                                                                 |FIN00240
C     |                                                                 |FIN00250
C     |-----------------------------------------------------------------|FIN00260
C     |                                                                 |FIN00270
C     |                                                                 |FIN00280
C                                                                       FIN00290
C                                                                       FIN00300
      DIMENSION CARD(1),VECTOR(1)                                       FIN00310
C                                                                       FIN00320
      DATA CRDSIZ/62/                                                   FIN00330
      K = VECTOR(1)+1                                                   FIN00340
      L = COL+1                                                         FIN00350
      DO 10 COL=L,CRDSIZ                                                FIN00360
      DO 10 I=2,K                                                       FIN00370
      II = I                                                            FIN00380
      IF (CARD(COL).EQ.VECTOR(I)) GO TO 20                              FIN00390
   10 CONTINUE                                                          FIN00400
      I =1                                                              FIN00410
      COL = L-1                                                         FIN00420
   20 FIND12 =I                                                         FIN00430
C     WRITE( 6,102) (CARD(K),K=1,62),COL,I,VECTOR(I)                    FIN00440
C 102 FORMAT(' FIND12 ENTERED'/' ',62A1,I10/' ',I5,A4)                  FIN00450
      RETURN                                                            FIN00460
      END                                                               FIN00470
                                                                        FIN00480
```

FILE: FLDINT FORTRAN A

```
      SUBROUTINE FLDINT(/BLOCK/,/FETVEC/,NOFEAT)                                FLD000010
      IMPLICIT INTEGER (A-Z)                                                    FLD000020
C*                                                                              FLD000030
C*    ENTRY FOR POSITIONING TAPE TO CORRECT SCAN LINE FOR A SPECIFIC FIE        FLD000040
C*                                                                              FLD000050
      COMMON /TAPERD/ IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,              FLD000060
     *  LINC,ID(200),DSL,LBUF(30),JREC(30),IPYTE(30),NHUFS,FILENO,LINEN         FLD000070
     *D   ,LINEND,LININC,NSAMP,NOCHAN,FORMT                                     FLD000080
      DIMENSION BLOCK(6)                                                        FLD000090
      DIMENSION FETVEC(NOFEAT)                                                  FLD000100
      EQUIVALENCE (ID(1),NRPDS ),(ID(2),NCPR  ),                                FLD000110
     *            (ID(3),NPRC  ),(ID(4),ANCLNG),                                FLD000120
     *            (ID(5),NC    ),(ID(6),NS    ),                                FLD000130
     *            (ID(7),NBITS ),(ID(8),DOI),                                   FLD000140
     *            (ID(9),NOSPR),(ID(10),NCAR ),                                 FLD000150
     *      (ID(11),SVD),(ID(16),PRS2)                                          FLD000160
      READY = 1                                                                 FLD000170
      NOCHAN = NOFEAT                                                           FLD000180
C*    CHECK FETVEC                                                              FLD000190
      DO 125 I=1,NOFEAT                                                         FLD000200
      IF (FETVEC(I) .GT. NC) GO TO 126                                          FLD000210
  125 CONTINUE                                                                  FLD000220
      GO TO 127                                                                 FLD000230
  126 WRITE(6,470)NC                                                            FLD000240
      NOFEAT = I - 1                                                            FLD000250
      IF(NOFEAT.LE.0)CALL CMERR                                                 FLD000260
  127 CONTINUE                                                                  FLD000270
      LINSTR=BLOCK(1)                                                           FLD000280
      IF (LINSTR.GE.IFRST) GO TO 130                                            FLD000290
      WRITE(6,430)IFRST                                                         FLD000300
      CALL CMERR                                                                FLD000310
  130 CONTINUE                                                                  FLD000320
      IF(BLOCK(2).GE.IFRST)GO TO 132                                            FLD000330
      WRITE(6,430)IFRST                                                         FLD000340
      CALL CMERR                                                                FLD000350
  132 CONTINUE                                                                  FLD000360
C*    FLINE=FIRST SCAN ON RECORD CONTAINING LINSTR                              FLD000370
      FLINE=LINSTR-MOD((LINSTR-IFRST),NOSPR)                                    FLD000380
      LSKIP=((FLINE-FSCAN)/NOSPR-1)*NRPDS                                       FLD000390
      IF (LSKIP) 135,138,138                                                    FLD000400
  135 FSKIP = ((BLOCK(1) - IFRST) / NOSPR ) * NRPDS + 1                         FLD000410
C*                                                                              FLD000420
C*    ON MULTI-FILE TAPES FOR FILES OTHER THAN FILE 1. DO THE FOLLOWING-        FLD000430
C*        1. BACK SPACE 1 FILE                                                  FLD000440
C*        2. READ FORWARD 1 E-O-F                                               FLD000450
C*        3. READ FORWARD NO. OF DESIRED RECORDS                                FLD000460
C*                                                                              FLD000470
      IF (FILENO .EQ. 0 .AND. FSKIP .LE. IABS(LSKIP)) GO TO 136                 FLD000480
      FSKIP = IABS(LSKIP)                                                       FLD000490
      DO 134  II=1,FSKIP                                                        FLD000500
  134 BACKSPACE IUNIT                                                           FLD000510
      GO TO 139                                                                 FLD000520
C*                                                                              FLD000530
C*    FOR FILE 1 DO A REWIND AND SKIP FORWARD THE DESIRED NO. OF RECORDS        FLD000540
C*                                                                              FLD000550
  136 REWIND IUNIT                                                              FLD000560
      DO 137  II=1,FSKIP                                                        FLD000570
  137 READ(IUNIT,480)DUMMY                                                      FLD000580
      GO TO 139                                                                 FLD000590
C*                                                                              FLD000600
C*    SKIP DOWN THE TAPE TO BEGINNING LINE OF THIS FIELD.                       FLD000610
C*                                                                              FLD000620
  138 IF (FSCAN.EQ.FLINE) GO TO 140                                             FLD000630
      IF ( LSKIP .EQ. 0) GO TO 139                                              FLD000640
      DO 141 II=1,LSKIP                                                         FLD000650
  141 READ(IUNIT,480) DUMMY                                                     FLD000660
  139 CONTINUE                                                                  FLD000670
      FSCAN=FLINE                                                               FLD000680
  140 CONTINUE                                                                  FLD000690
      NSCAN=LINSTR                                                              FLD000700
      IF(BLOCK(5).LE.NS)GO TO 145                                               FLD000710
      WRITE(6,440)NS                                                            FLD000720
      CALL CMERR                                                                FLD000730
  145 IF(BLOCK(6).LE.NS)GO TO 146                                               FLD000740
      WRITE(6,440)NS                                                            FLD000750
      CALL CMERR                                                                FLD000760
  146 CONTINUE                                                                  FLD000770
      LINEND=BLOCK(2)                                                           FLD000780
      LININC=BLOCK(3)                                                           FLD000790
```

```
FILE: FLDINT   FORTRAN  A

      SAMSTR=BLOCK(6)                                                          FLD00800
      SAMEND=BLOCK(7)                                                          FLD00810
      SAMINC=BLOCK(8)                                                          FLD00820
C*    LINC=NO.OF RECORDS TO SKIP AFTER EACH SCCAN LINE                         FLD00830
      LINC=(LININC/NOSPR - 1)*NRPDS                                            FLD00840
      IF(LINC.LT.0)LINC=0                                                      FLD00850
C*                                                                             FLD00860
C*    ESTABLISH AREAS ON EACH SCAN LINE TO UNPACK                              FLD00870
C*                                                                             FLD00880
      ANC=ANCLNG * SAMSTR * SVD - 1                                            FLD00890
      IF ( FORMT .EQ. 1) ANC = ANC * 2                                         FLD00900
      NBUFS=NWPDS/LD                                                           FLD00910
      IF(MOD(NWPDS,10).NE.0)NBUFS=NBUFS+1                                      FLD00920
      FC=1                                                                     FLD00930
      LC=NCAR                                                                  FLD00940
      K=1                                                                      FLD00950
      DO 190 I=1,NOFEAT                                                        FLD00960
      TRY = 0                                                                  FLD00970
  185  CONTINUE                                                                FLD00980
      DO 170 IREC=K,NRPDS                                                      FLD00990
      IF(IREC.GT.1)ANC=2 * SAMSTR * SVD - 1                                    FLD01000
      IF (FETVEC(I).GE.FC.AND.FETVEC(I).LE.LC) GO TO 150                       FLD01010
      IF (FETVEC(I).GT.LC.AND.IREC.LT.NWPDS) GO TO 160                         FLD01020
      FC = 1                                                                   FLD01030
      LC = NCAR                                                                FLD01040
      K = 1                                                                    FLD01050
      ANC = (ANCLNG * 2) * SAMSTR * SVD - 1                                    FLD01060
      TRY = TRY + 1                                                            FLD01070
      IF (TRY .LE. 2) GO TO 185                                                FLD01080
      WRITE (6,380)FETVEC(I)                                                   FLD01090
      CALL CHEAN                                                               FLD01100
  150 TRYTF(I)=(FETVEC(I)-FC)*NS + ANC                                         FLD01110
      JREC(I)=MOD(IREC,10)                                                     FLD01120
      IF(JREC(I).EQ.0) JREC(I)=10                                              FLD01130
      LBUF(I)=IREC/10 + 1                                                      FLD01140
      IF(MOD(IREC,10).EQ.0)LBUF(I)=LBUF(I)-1                                   FLD01150
      GO TO 180                                                                FLD01160
  160 FC=LC+1                                                                  FLD01170
      LC=LC+NCAR                                                               FLD01180
  170 CONTINUE                                                                 FLD01190
  180 K=IREC                                                                   FLD01200
  190 CONTINUE                                                                 FLD01210
C*                                                                             FLD01220
C*    NSAMP - NO. OF SAMPLES TO UNPACK FOR EACH FEATURE IN FETVEC              FLD01230
C*                                                                             FLD01240
      NSAMP = (SAMEND - SAMSTR) / SAMINC + 1                                   FLD01250
      RETURN                                                                   FLD01260
  380 FORMAT(' FEATURE NUMBERS',I5,' AND ABOVE ARE NOT ON DATA TAPE'/          FLD01270
     *)                                                                        FLD01280
  430 FORMAT(' FIRST SCAN ON THIS TAPE IS NUMBERED',IA,' FIELD DEFINITIO       FLD01290
     *N IN ERROR')                                                           FLD01300
  440 FORMAT(' NUMBER OF SAMPLES OF PER SCAN ON THIS TAPE IS',I6,' FIELD       FLD01310
     * DEFINITION IN ERROR')                                                  FLD01320
  470 FORMAT(' THIS TAPE CONTAINS ONLY',I6,' CHANNELS')                        FLD01330
  480 FORMAT(1A4)                                                              FLD01340
      END                                                                      FLD01350
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: FLTNUM FORTRAN A

```
      FUNCTION FLTNUM(CARD,COL,NUMVEC,VECMAX)                                FLT00010
C                                                                            FLT00020
      IMPLICIT INTEGER (A-H,O-Z)                                             FLT00030
CCI---------------------------------------------------------------------IFLT00040
CCI---------------------------------------------------------------------IFLT00050
CCCCI                                                                    IFLT00060
CCCCI     CALL..    J = FLTNUM(CARD,COL,NUMVEC,VECMAX)                   IFLT00070
CCCCI                                                                    IFLT00080
CCCCI     ARGS..    CARD   - 62 COL CARD BUFFER                          IFLT00090
CCCCI               COL    - PTR TO FIRST COL IN CARD TO SCAN            IFLT00100
CCCCI               NUMVEC - BUFFER IN WHICH TO RETURN THE NUMBERS       IFLT00110
CCCCI               VECSIZ - LENGTH OF NUMVEC                            IFLT00120
CCCCI                                                                    IFLT00130
CCCCI     REQUIRES. NONE                                                 IFLT00140
CCCCI                                                                    IFLT00150
CCCCI     PURPOSE.. INTERPRETS REAL NUMBERS SEPARTED BY COMMAS ON CARD   IFLT00160
CCCCI               AND RETURNS THEM IN NUMVEC.                          IFLT00170
CCCCI                        STOPS AT FIRST 'NONUMERIC'                  IFLT00180
CCCCI                        (NOTE. NUMBERS MAY APPEAR IN                IFLT00190
CCCCI                              'DATA STATEMENT FORMAT' )             IFLT00200
CCCCI                                                                    IFLT00210
CCCCI     RETURNS.. COL    -  COLUMN WHERE SCAN TERMINATED               IFLT00220
CCCCI               NUMVEC - VECTOR OF REAL NUMBERS FOUND                IFLT00230
CCCCI               FLTNUM - NO OF REAL NUMBERS RETURNED                 IFLT00240
CCCCI                                                                    IFLT00250
CCI---------------------------------------------------------------------IFLT00260
CCI---------------------------------------------------------------------IFLT00270
CCC                                                                      FLT00280
C                                                                        FLT00290
      REAL NUMVEC(20),PNUM                                               FLT00300
      DIMENSION CARD(62)                                                 FLT00310
      DATA BLANK/' '/,COMMA/','/,PLUS/'+'/,MINUS/'-'/,STAR/'*'/,         FLT00320
     1   PERIOD/'.'/, ZERO/'0'/, CRDSIZ/62/                              FLT00330
       LOGICAL*1 LLNUM(1)                                                FLT00340
       LOGICAL*1 LNUM(4)                                                 FLT00350
       DIMENSION INUM(1)                                                 FLT00360
       EQUIVALENCE (INUM(1),LNUM(1))                                     FLT00370
        DATA XX/Z000000F0/                                               FLT00380
       DATA LLNUM/Z00/                                                   FLT00390
C                                                                        FLT00400
C     ---------------------------------------------------------------    FLT00410
C                                                                        FLT00420
      L = COL+1                                                          FLT00430
      VECPOS = 1                                                         FLT00440
   10 WNUM = 0                                                           FLT00450
      PCNT = 0                                                           FLT00460
      PNUM = 0.0                                                         FLT00470
      ITER = 1                                                           FLT00480
      SIDE = -1                                                          FLT00490
      SIGN = +1                                                          FLT00500
C                                                                        FLT00510
      DO 60 COL=L,CRDSIZ                                                 FLT00520
      IF (CARD(COL).EQ.BLANK) GO TO 60                                   FLT00530
      IF (CARD(COL).EQ.PLUS) GO TO 60                                    FLT00540
      IF (CARD(COL).EQ.COMMA) GO TO 70                                   FLT00550
      IF (CARD(COL).NE.MINUS) GO TO 20                                   FLT00560
      SIGN = -SIGN                                                       FLT00570
      GO TO 60                                                           FLT00580
   20 IF (CARD(COL).NE.STAR) GO TO 30                                    FLT00590
      ITER = WNUM                                                        FLT00600
      WNUM = 0                                                           FLT00610
      PCNT = 0                                                           FLT00620
      PNUM = 0.0                                                         FLT00630
      SIDE = -1                                                          FLT00640
      SIGN = +1                                                          FLT00650
      GO TO 60                                                           FLT00660
   30 IF (CARD(COL).NE.PERIOD) GO TO 40                                  FLT00670
      SIDE = 1                                                           FLT00680
      GO TO 60                                                           FLT00690
   40 IF (CARD(COL).LT.ZERO) GO TO 90                                    FLT00700
       INUM(1)=CARD(COL)                                                 FLT00710
       LNUM(4)=LNUM(1)                                                   FLT00720
       LNUM(1)=LLNUM(1)                                                  FLT00730
       LNUM(2)=LLNUM(1)                                                  FLT00740
       LNUM(3)=LLNUM(1)                                                  FLT00750
       MORNUM=INUM(1)-XX                                                 FLT00760
       IF (SIDE.LT.0) GO TO 50                                           FLT00770
      PCNT = PCNT+1                                                      FLT00780
      PNUM = PNUM+MORNUM*(0.1**PCNT)                                     FLT00790
```

FILE: FLTNUM FORTRAN A

```
      GO TO 60                                                                  FLT00800
   50 WNUM = 10*WNUM+MORNUM                                                     FLT00810
   60 CONTINUE                                                                  FLT00820
C                                                                               FLT00830
      COL = CRDSIZ+1                                                            FLT00840
      GO TO 90                                                                  FLT00850
   70 VECFIN = VECPOS+ITER-1                                                    FLT00860
      IF ( VECFIN .GT. VECMAX ) VECFIN = VECMAX-1                               FLT00870
      DO 80 I=VECPOS,VECFIN                                                     FLT00880
   80 NUMVEC(I) = SIGN*(WNUM+PNUM)                                              FLT00890
      L = COL+1                                                                 FLT00900
      VECPOS = VECFIN+1                                                         FLT00910
      IF (VECPOS.LE.VECMAX) GO TO 10                                            FLT00920
      GO TO 110                                                                 FLT00930
   90 COL = COL-1                                                               FLT00940
      VECFIN = VECPOS+ITER-1                                                    FLT00950
      IF ( VECFIN .GT. VECMAX ) VECFIN = VECMAX                                 FLT00960
      DO 100 I=VECPOS,VECFIN                                                    FLT00970
  100 NUMVEC(I) = SIGN*(WNUM+PNUM)                                              FLT00980
  110 FLTNUM = VECFIN                                                           FLT00990
C     WRITE(6,706)  (CARD(K),K=1,62), COL,FLTNUM,(NUMVEC(K),K=1,FLTNUM)        FLT01000
C 706 FORMAT(' FLTNUM ENTERED'/' ', 62A1,I10/' ',I5,20F8.2,/' ',10F8.2)         FLT01010
      RETURN                                                                    FLT01020
       END                                                                      FLT01030
```

FILE: FSASFL FORTRAN A

```
      SUBROUTINE FSASFL(UNIT,FILE,ISTAT)                                FSA00010
      IMPLICIT INTEGER (A-Z)                                            FSA00020
      N=0                                                               FSA00030
C                                                                       FSA00040
      ISTAT = 0                                                         FSA00050
      IF (FILE .EQ. 0) RETURN                                           FSA00060
      IF ( FILE .LT. 0) GO TO 100                                       FSA00070
C                                                                       FSA00080
CC    MOVE UNIT FORWARD N E-O-F'S                                       FSA00090
C                                                                       FSA00100
   40 READ(UNIT,END=50)                                                 FSA00110
      GO TO 40                                                          FSA00120
   50 N = N + 1                                                         FSA00130
      IF ( N .EQ. FILE) RETURN                                          FSA00140
      GO TO 40                                                          FSA00150
C                                                                       FSA00160
  100 WRITE(6,110)                                                      FSA00170
  110 FORMAT(' FSASFL ONLY SKIPS FORWARD'/)                             FSA00180
      ISTAT = 2                                                         FSA00190
      RETURN                                                            FSA00200
C                                                                       FSA00210
      END                                                               FSA00220
```

~~19-19~~

45

FILE: FSFMFL FORTRAN A

```
      SUBROUTINE FSFMFL(UNIT,FILE,ISTAT)                                FSB00010
      IMPLICIT INTEGER (A-Z)                                            FSA00020
      N=0                                                               FSB00030
C                                                                       FSB00040
      ISTAT = 0                                                         FSB00050
      IF (FILE .EQ. 0) RETURN                                           FSB00060
      IF ( FILE .LT. 0) GO TO 100                                       FSB00070
C                                                                       FSB00080
C     MOVE UNIT FORWARD N E-O-F'S                                       FSB00090
C                                                                       FSB00100
   40 READ(UNIT,30,END=50)DUMMY                                         FSB00110
   30 FORMAT(1A4)                                                       FSB00120
      GO TO 40                                                          FSB00130
   50 N = N + 1                                                         FSB00140
      IF ( N .EQ. FILE) RETURN                                          FSB00150
      GO TO 40                                                          FSB00160
C                                                                       FSB00170
  100 WRITE(6,110)                                                      FSB00180
  110 FORMAT(' FSFMFL ONLY SKIPS FORWARD'/)                             FSB00190
      ISTAT = 2                                                         FSB00200
      RETURN                                                            FSB00210
C                                                                       FSB00220
      END                                                               FSB00230
```

FILE: GETST FORTRAN A

```
      SUBROUTINE GETST(UNIT,FILE,MENS,STDEV,NOSUB2,SUBVEC,NOCHAN,        GET00010
     * CHNVEC,MEANS,COVAR,ITRIG)                                         GET00020
C*                                                                       GET00030
C*************************************************************************GET00040
C*                                                                       GET00050
C*    SUBROUTINE GETSTA RETRIEVES THE MEANS AND STANDARD DEVIATIONS      GET00060
C*    FROM A STATISTICS FILE IN THE LARSYS 'SAVTAP' FORMAT.              GET00070
C*                                                                       GET00080
C*    INPUT ARGUMENTS:                                                   GET00090
C*                                                                       GET00100
C*      UNIT   - FORTRAN UNIT NUMBER FROM WHICH THE STATS ARE TO BE      GET00110
C*               RETRIEVED.                                              GET00120
C*      FILE   - FILE NO. ON 'UNIT' FROM WHICH THE STATS ARE TO BE       GET00130
C*               RETRIEVED.                                              GET00140
C*      NOCHAN   - NO. OF CHANNELS REQUESTED FROM TRAINING SEGMENT.      GET00150
C*     *           NOCHAN LESS THAN OR = NO. OF CHANNELS ON STAT FILE    GET00160
C*      CHNVEC   - ARRAY CONTAINING ACTUAL CHANNELS REQUESTED FROM       GET00170
C*     *           TRAINING SEGMENT. MUST BE A SUBSET OF CHANNELS        GET00180
C*                 ON STAT FILE                                          GET00190
C*                                                                       GET00200
C*     ITRIG   - IF ITRIG=1, ST. DEV. WILL BE RETURNED ALONG WITH MEANS. GET00210
C*                                                                       GET00220
C*    OUTPUT ARGUMENTS:                                                  GET00230
C*                                                                       GET00240
C*       MENS   - ARRAY CONTAINING THE MEAN VECTORS FOR EACH SUBCLASS    GET00250
C*     *           ( A SUBSET OF THE CHANNELS MAY BE SELECTED, BUT NOT   GET00260
C*     *           A SUBSET OF THE SUBCLASSES).                          GET00270
C*      STDEV   - ARRAY CONTAINING THE SUBSET OF ST. DEV. FOR REQUES-    GET00280
C*                 TED CHANNELS IN EACH SUBCLASS                         GET00290
C*                IN EACH SUBCLASS.                                      GET00300
C*      NOSUB  - NUMBER OF SUBCLASSES ON THE STAT FILE                   GET00310
C*    CONTINUE                                                           GET00320
C*    CHNVEC - ARRAY CONTAINING ACTUAL CHANNELS REQUESTED FROM           GET00330
C*               TRAINING SEGMENT                                        GET00340
C*                                                                       GET00350
C*************************************************************************GET00360
C*************************************************************************GET00370
C*                                                                       GET00380
C*  **NOTE**  THE STORAGE ARRAYS PASSED TO THIS SUBROUTINE FOR THE       GET00390
C*            MEANS AND STANDARD DEVIATIONS SHOULD BE SINGLY DIMENSIONEDGET00400
C*            IN THE CALLING ROUTINE.  ON OUTPUT THE ITEMS ARE STORED   GET00410
C*            AS FOLLOWS: (SAME FOR STDEV)                              GET00420
C*                                                                      GET00430
C*                 MEANS(1)        =  CHANNEL 1, SUBCLASS 1            GET00440
C*                       (2)       =  CHANNEL 2, SUBCLASS 1            GET00450
C*                       3         =  CHANNEL 3, SUBCLASS 1            GET00460
C*                       .         =          .                        GET00470
C*                       .         =          .                        GET00480
C*                     (NOCHAN)    =          NOCHAN,SUBCLASS 1        GET00490
C*                     (NOCHAN+1)  =  CHANNEL 1, SUBCLASS 2            GET00500
C*                     (NOCHAN+2)  =  CHANNEL 2, SUBCLASS 2            GET00510
C*                     (NOCHAN+3)  =  CHANNEL 3, SUBCLASS 2            GET00520
C*                           .     =                                    GET00530
C*                           .     =                                    GET00540
C*                     (2*NOCHAN)  =  CHANNEL NOCHAN OF SUBCLASS 2      GET00550
C*                        .                                            GET00560
C*                        .                                            GET00570
C*                        .                                            GET00580
C*                      ETC.                                           GET00590
C*                      THRU                                           GET00600
      CONTINUE                                                         GET00610
C*                    (NOCHAN*NOSUB)                                   GET00620
C*                                                                     GET00630
C*************************************************************************GET00640
C*                                                                       GET00650
      IMPLICIT INTEGER (A-Z)                                           GET00660
      DIMENSION CHNVC1(30),DUMVEC(30),CHNVEC(1)                        GET00670
      DATA BLANK/' '/                                                  GET00680
       DIMENSION SUBVEC(1)                                             GET00690
      DIMENSION MEANS(1),STDEV(1)                                      GET00700
      REAL MEANS,STDEV,COVAR(465)                                      GET00710
      REAL MENS(1)                                                     GET00720
      REWIND UNIT                                                      GET00730
      NF=FILE-1                                                        GET00740
      CALL FSBSFL(UNIT,NF,ISTAT1)                                      GET00750
      IF(ISTAT1.EQ.0)GO TO 5                                           GET00760
      WRITE(6,100)UNIT,FILE                                            GET00770
  100 FORMAT(1X,'ERROR IN POSITIONING UNIT',I3,' TO FILE', I3)         GET00780
      CALL CMERR                                                       GET00790
```

FILE: GETST FORTRAN A

```
     5 CONTINUE                                                                 GET00800
       READ(UNIT)NOCLS,NOSUB,NCHAN,NOFLD,TOTVRT,(CHNVC1(I),I=1,NCHAN)          GET00810
C                                                                               GET00820
C*     DEFAULT ALL SUBCLASSES FROM STATISTICS FILE                              GET00830
C*                                                                             GET00840
       IF(NOSUB2.NE.0)GO TO 7                                                   GET00850
       DO 6 I=1,NOSUB                                                           GET00860
  6    SURVEC(I)=I                                                              GET00870
       NOSUB2=NOSUB                                                             GET00880
  7    CONTINUE                                                                 GET00890
       DO 77 I=1,NOSUB2                                                         GET00900
       IF (SURVEC(I) .LE. NOSUB) GO TO 77                                       GET00910
       WRITE(6,70)NOSUB                                                         GET00920
   70 FORMAT(/' REQUESTED SUBCLASS IS NOT ON STAT FILE. STAT FILE CONTAIGET00930
      *NS ',I3,' SUBCLASSES')                                                   GET00940
       CALL CMERR                                                               GET00950
   77 CONTINUE                                                                  GET00960
C*                                                                              GET00970
C*     DEFAULT -- ALL CHANNELS FROM STAT FILE                                   GET00980
C*                                                                             GET00990
       IF (NOCHAN .NE. 0)  GO TO 9                                              GET01000
       DO 8  I=1,NCHAN                                                          GET01010
    8  CHNVEC(I) = CHNVC1(I)                                                    GET01020
       NOCHAN = NCHAN                                                           GET01030
    9  CONTINUE                                                                 GET01040
C*                                                                              GET01050
C*     READ PAST THE TRAINING FIELD INFORMATION                                 GET01060
C*                                                                             GET01070
       DO 10 I=1,NOFLD                                                          GET01080
       READ(UNIT)DUM                                                            GET01090
       READ(UNIT)DUM                                                            GET01100
   10 CONTINUE                                                                  GET01110
       READ(UNIT)DUM                                                            GET01120
       VARSIZ = NCHAN*(NCHAN+1)/2                                               GET01130
       MB=1                                                                     GET01140
       ME = NCHAN                                                               GET01150
        IC=1                                                                    GET01160
       DO 30 I=1,NOSUB                                                          GET01170
       READ(UNIT) N,(COVAR(J),J=1,VARSIZ),(MEANS(J),J=MB,ME)                    GET01180
        IF(SURVEC(IC).NE.I)GO TO 30                                             GET01190
       IF(ITRIG.NE.1)GO TO 25                                                   GET01200
C*                                                                              GET01210
C*     GET STANDARD DEVIATIONS                                                  GET01220
C*                                                                             GET01230
       JK=0                                                                     GET01240
       DO 20 JA=1,NOCHAN                                                        GET01250
       JK= JK+ JA                                                               GET01260
       STDEV(MB+JA-1) = SQRT(COVAR(JK))                                         GET01270
   20 CONTINUE                                                                  GET01280
   25 CONTINUE                                                                  GET01290
       IC = IC + 1                                                              GET01300
       MB = MB + NCHAN                                                          GET01310
       ME = ME + NCHAN                                                          GET01320
   30 CONTINUE                                                                  GET01330
C                                                                               GET01340
C      GET SUBSET OF MEANS                                                      GET01350
C      AND GET SUBSET OF ST. DEV.                                               GET01360
C                                                                              GET01370
       DO 50   J=1,NOCHAN                                                       GET01380
       DO 40   K=1,NCHAN                                                        GET01390
       IF (CHNVEC(J) .EQ. CHNVC1(K))  GO TO 50                                  GET01400
   40 CONTINUE                                                                  GET01410
       WRITE(6,110)CHNVEC(J),(CHNVC1(L),L=1,NCHAN)                              GET01420
  110 FORMAT(' CHANNEL NO. ',I2,' IS NOT ON TRAINING STAT FILE. CHANNELSGET01430
     1 ARE '/ 10X,30(I2,1X))                                                    GET01440
       CALL EXIT                                                                GET01450
   50 DUMVEC(J) = K                                                             GET01460
C                                                                              GET01470
       JJJ = 0                                                                  GET01480
       DO 60 K=1,NOSUB2                                                         GET01490
       DO 60  J=1,NOCHAN                                                        GET01500
       JJ = DUMVEC(J)  + (K-1)*NCHAN                                            GET01510
       JJJ = JJJ + 1                                                            GET01520
       MENS(JJJ) = MEANS(JJ)                                                    GET01530
   60 IF ( ITRIG .NE. 0) STDEV(JJJ) = STDEV(JJ)                                 GET01540
       WRITE(6,200)                                                             GET01550
  200 FORMAT(//157,'MEANS')                                                     GET01560
       ISTART = 1                                                               GET01570
       IEND = 12                                                                GET01580
```

FILE: GETST FORTRAN A

```
      LOOPCT = NOCHAN/12                                                GET01590
      LOOPC1 = MOD(NOCHAN,12)                                           GET01600
      IF (LOOPC1 .GT. 0) LOOPCT = LOOPCT + 1                            GET01610
      IF (LOOPCT .EQ. 1) IEND = NOCHAN                                  GET01620
      DO 240 II=1,LOOPCT                                                GET01630
      START = ISTART                                                    GET01640
      END = IEND                                                        GET01650
      WRITE(6,210)(BLANK,CHNVEC(I),I=ISTART,IEND)                       GET01660
  210 FORMAT(/2X,'CLUSTER',5X,12(A1,'CH(',I2,')',2X))                   GET01670
      DO 230 J=1,NOSUB2                                                 GET01680
      WRITE(6,220)SURVEC(J),(MENS(K),K=START,END)                       GET01690
  220 FORMAT(5X,I2,7X,12(F7.2,2X))                                      GET01700
      START = ISTART + NOCHAN*J                                         GET01710
      END = IEND + NOCHAN*J                                             GET01720
  230 CONTINUE                                                          GET01730
      WRITE(6,235)                                                      GET01740
  235 FORMAT(/)                                                         GET01750
      ISTART = IEND + 1                                                 GET01760
      IEND = IEND + ISTART - 1                                          GET01770
      IF (IEND .GT. NOCHAN) IEND = NOCHAN                               GET01780
  240 CONTINUE                                                          GET01790
      RETURN                                                            GET01800
      END                                                               GET01810
```

FILE: GRPSCN FORTRAN A

```
      FUNCTION GRPSCN(CARD,NNCLAS,GRPTR)                                GRP00010
C                                                                       GRP00020
      IMPLICIT INTEGER (A-H,O-Z)                                        GRP00030
CCCCC|-----------------------------------------------------------------|GRP00040
CCCCC|                                                                 |GRP00050
CCCCC|-----------------------------------------------------------------|GRP00060
CCCCC|-----------------------------------------------------------------|GRP00070
CCCCC|                                                                 |GRP00080
CCCCC|     CALL..    CALL GRPSCN(CARD,NNCLAS,GRPTR)                   |GRP00090
CCCCC|                                                                 |GRP00100
CCCCC|     ARGS..    CARD   - 62 COL CARD BUFFER                      |GRP00110
CCCCC|                NNCLAS - MAX NO OF CLASSES TO ALLOW              |GRP00120
CCCCC|                GRPTR  - PTR TO 'GROUPS'                         |GRP00130
CCCCC|                                                                 |GRP00140
CCCCC|     REQUIRES.  COMMONS  /INFORM/  /INFORS/  /DISCOM/            |GRP00150
CCCCC|                ROUTINES NXTCHR    FIXUP     NUMBER             |GRP00160
CCCCC|                                                                 |GRP00170
CCCCC|     PURPOSE..  SCANS ALL 'GROUP' (TRAIN/TEST) CARDS             |GRP00180
CCCCC|                AND SET UP 'GRPDEX','GRPNAM','GROUPS'            |GRP00190
CCCCC|                                                                 |GRP00200
CCCCC|     RETURNS..  GRPTR  - SEE ARGS                                |GRP00210
CCCCC|                                                                 |GRP00220
CCCCC|-----------------------------------------------------------------|GRP00230
CCCCC|-----------------------------------------------------------------|GRP00240
CCCCC                                                                   GRP00250
CCCCC                                                                   GRP00260
CCCCC                                                                   GRP00270
CCCCC   INCLUDE COMBK1,LIST                                             GRP00280
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          GRP00290
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    GRP00300
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),        GRP00310
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),             GRP00320
     *              GRPCHK(61),GROUPS(124)                              GRP00330
C$END                                                                   GRP00340
                                                                        GRP00350
CCCCC|-----------------------------------------------------------------|GRP00360
CCCCC                                                                   GRP00370
      DIMENSION BUF(4), CARD(62), COMVEC(2), NUMVEC(30)                 GRP00380
       LOGICAL*1 LCHAR(4)                                               GRP00390
       DIMENSION ICHAR(1)                                               GRP00400
       EQUIVALENCE (LCHAR(1),ICHAR(1))                                  GRP00410
       LOGICAL*1 LLCHAR(4)                                              GRP00420
       DIMENSION IICHAR(1)                                              GRP00430
       EQUIVALENCE (LLCHAR(1),IICHAR(1))                                GRP00440
C                                                                       GRP00450
      DATA  BLANK/' '/, COMMA/','/, COMVEC/1,','/                       GRP00460
CCCCC                                                                   GRP00470
CCCCC|-----------------------------------------------------------------|GRP00480
CCCCC|-----------------------------------------------------------------|GRP00490
C                                                                       GRP00500
      COL = 0                                                           GRP00510
      GRPSCN = 1                                                        GRP00520
      J = NXTCHR(CARD,COL)                                              GRP00530
      IF (J.EQ.BLANK) GO TO 110                                         GRP00540
      DO 10 I=1,4                                                       GRP00550
      J2 = CARD(COL-1+I)                                                GRP00560
      IF (J2.EQ.COMMA) GO TO 20                                         GRP00570
   10 BUF(I) = J2                                                       GRP00580
      GO TO 40                                                          GRP00590
C                                                                       GRP00600
   20 DO 30 J=I,4                                                       GRP00610
   30 BUF(J) = BLANK                                                    GRP00620
C                                                                       GRP00630
   40 N = ABS(0.0)                                                      GRP00640
      DO 50 I=1,4                                                       GRP00650
       IICHAR(1)=BUF(I)                                                 GRP00660
   50  LCHAR(I)=LLCHAR(1)                                               GRP00670
       WRD1=ICHAR(1)                                                    GRP00680
      GRPNAM(NOGRP+1) = WRD1                                            GRP00690
      J = FIND12(CARD,COL,COMVEC)                                       GRP00700
      IF (J.LE.0) GO TO 110                                             GRP00710
      J = NUMBER(CARD,COL,NUMVEC,0)                                     GRP00720
      II = 0                                                            GRP00730
      LAST = 0                                                          GRP00740
      DO 90 I=1,J                                                       GRP00750
      JJ = NUMVEC(I)                                                    GRP00760
      IF (JJ.GT.LAST.AND.JJ.LE.NNCLAS.AND.GRPCHK(JJ).EQ.0) GO TO 80     GRP00770
      WRITE(6, 70)  JJ, JJ, CARD                                        GRP00780
   70 FORMAT(// 5X, '////// FROM SUBR. GRPSCN  ---  CLASS ',I5,         GRP00790
```

FILE: GRPSCN FORTRAN A

```
     1 ' INCORRECT ---   CLASS ',I5,' IGNORED' // 11X,'CARD BEING SCANNGRP00800
     2ED IS ...'/9X,2H'', 62A1, 2H'' /)                                GRP00810
      GO TO 90                                                         GRP00820
   80 II = II+1                                                        GRP00830
      NUMVEC(II) = JJ                                                  GRP00840
      GRPCHK(JJ) = 1                                                   GRP00850
      LAST = JJ                                                        GRP00860
   90 CONTINUE                                                         GRP00870
      IF (II.LE.0) GO TO 110                                           GRP00880
C                                                                      GRP00890
      NOGRP = NOGRP+1                                                  GRP00900
      GRPTR = GRPTR+1                                                  GRP00910
      GRPDEX(NOGRP) = GRPTR                                            GRP00920
      GROUPS(GRPTR) = II                                               GRP00930
      DO 100 I=1,II                                                    GRP00940
  100 GROUPS(GRPTR+I) = NUMVEC(I)                                      GRP00950
      GRPTR = GRPTR+II                                                 GRP00960
      GRPSCN = 0                                                       GRP00970
      RETURN                                                           GRP00980
C                                                                      GRP00990
C                                                                      GRP01000
  110 RETURN                                                           GRP01010
      END                                                              GRP01020
```

FILE: HISTGM FORTRAN A

```
      SUBROUTINE HISTGM(FILHIS,FLDTAL,TOTTAL)                                   HIS00010
CI*****************************************************************************CI HIS00020
CI                                                                              CI HIS00030
CI    PURPOSE  -- CALCULATES HISTOGRAMS AND WRITES TOTAL HISTOGRAMMED           CI HIS00040
CI                STATISTICS ON UNIT 13 TO BE READ BY GRAYMP PROCESSOR        CI HIS00050
CI*****************************************************************************CI HIS00060
C                                                                               HIS00070
      IMPLICIT INTEGER(A-T)                                                     HIS00080
      INCLUDE COMBK3.LIST                                                       HIS00090
      INCLUDE COMBK4.LIST                                                       HIS00100
      INCLUDE COMBK6.LIST                                                       HIS00110
      COMMON /GRCBLK/MAXFET,NOFEAT,NOFET2,FETVEC(30),                           HIS00120
     *               FETVC2(30),FLDINF(6),INFMT,FILESV,NOHIST,                  HIS00130
     *               HISVEC(30),NOFLD,          FLDPTS                         HIS00140
     *,XSIZ,XLOW,XHGH,YSIZ                                                      HIS00150
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                            HIS00160
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                         HIS00170
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                       HIS00180
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,                 HIS00190
     *              HISFIL,HISKEY,TRFORM,ERRPTP,ERRKEY,MAPUNT,NOFILE,           HIS00200
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                        HIS00210
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                            HIS00220
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                      HIS00230
     *  CRDUNT,PRTUNT,RANDIO                                                    HIS00240
C$END                                                                           HIS00250
C                                                                               HIS00260
      COMMON /HISTOR/HF                                                         HIS00270
C                                                                               HIS00280
      INTEGER XSIZ,YSIZ,XHGH,XLOW                                               HIS00290
      INTEGER VERTCS                                                            HIS00300
      DIMENSION IDATA(12000),FILHIS(NOFEAT,256),                                HIS00310
     * FLDTAL(NOHIST,XSIZ),TOTTAL(NOHIST,XSIZ),IFLD(50,24),                     HIS00320
     * VERTCS(2,11),FL(8)                                                       HIS00330
      DATA BLANK/' '/,DIM/12000/                                                HIS00340
      DATA TOTAL/'TOTA'/                                                        HIS00350
      EQUIVALENCE (FLDINF(1),LINSTR),(FLDINF(2),LINEND),                        HIS00360
     *            (FLDINF(3),LININC),(FLDINF(4),SAMSTR),                        HIS00370
     *            (FLDINF(5),SAMEND),(FLDINF(6),SAMINC)                         HIS00380
      CALL TAPHDR(DATAPE,DATFIL)                                                HIS00390
      HISKEY=1                                                                  HIS00400
      FILESV = 0                                                                HIS00410
      ISWTH = 0                                                                 HIS00420
      DO 7    J=1,XSIZ                                                          HIS00430
      DO 7    I=1,NOHIST                                                        HIS00440
      FLDTAL(I,J) = 0                                                           HIS00450
    7 TOTTAL(I,J) = 0                                                           HIS00460
      NOFLD=0                                                                   HIS00470
      IF(HF.NE.1) GO TO 10                                                      HIS00480
      NC=5                                                                      HIS00490
      VERTCS(1,1)=1                                                             HIS00500
      VERTCS(2,1)=1                                                             HIS00510
      VERTCS(1,2)=200                                                           HIS00520
      VERTCS(2,2)=1                                                             HIS00530
      VERTCS(1,3)=200                                                           HIS00540
      VERTCS(2,3)=500                                                           HIS00550
      VERTCS(1,4)=1                                                             HIS00560
      VERTCS(2,4)=500                                                           HIS00570
      VERTCS(1,5)=1                                                             HIS00580
      VERTCS(2,5)=1                                                             HIS00590
      FLDNAM=BLANK                                                              HIS00600
      FLDINF(1)=1                                                               HIS00610
      FLDINF(2)=500                                                             HIS00620
      FLDINF(3)=10                                                              HIS00630
      FLDINF(4)=1                                                               HIS00640
      FLDINF(5)=200                                                             HIS00650
      FLDINF(6)=10                                                              HIS00660
      GO TO 15                                                                  HIS00670
C                                                                               HIS00680
C     READ IN FIELD CARDS                                                       HIS00690
C                                                                               HIS00700
   10 ICK=LAREAD(FLDNAM,VERTCS,FLDINF,NC)                                       HIS00710
      IF(ICK.EQ.1) GO TO 15                                                     HIS00720
      IF(ICK.EQ.0) GO TO 60                                                     HIS00730
      IF(ICK.LE.-1) GO TO 10                                                    HIS00740
   15 NSAMP = ( FLDINF(5) - FLDINF(4) ) / FLDINF(6) + 1                         HIS00750
C                                                                               HIS00760
C     CHECK DATA DIMENSIONS. IF TOO MUCH DATA REQUESTED, RESET SAMPLE END       HIS00770
C                                                                               HIS00780
      TOTPTS = NSAMP*NOFEAT                                                     HIS00790
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: HISTGM FORTRAN A

```
      IF (TOTPTS .LE. DIM)   GO TO 14                                          HIS00800
      NSAMP = DIM / NOFEAT                                                     HIS00810
      FLDINF(5) = (NSAMP-1)*FLDINF(6) + FLDINF(4)                              HIS00820
      WRITE(6,300) NSAMP                                                       HIS00830
  300 FORMAT(' TOO MUCH DATA REQUESTED -- SAMPLE END WAS RESET TO',I5/)        HIS00840
C                                                                              HIS00850
CCC   STORE FIELD INFORMATION                                                  HIS00860
C                                                                              HIS00870
   14 NOFLD = NOFLD + 1                                                        HIS00880
      IF(NOFLD.GT.50) FILESW=-2                                                HIS00890
      IF(NOFLD.GT.50) GO TO 19                                                 HIS00900
      K=0                                                                      HIS00910
      DO 100 J=1,10                                                            HIS00920
      DO 100 I=1,2                                                             HIS00930
      K=K+1                                                                    HIS00940
  100 IFLD(NOFLD,K)=VERTCS(I,J)                                                HIS00950
      IFLD(NOFLD,21)=FLDNAM                                                    HIS00960
      IFLD(NOFLD,22)=SAMINC                                                    HIS00970
      IFLD(NOFLD,23)=LININC                                                    HIS00980
      IFLD(NOFLD,24)=NC                                                        HIS00990
   19 CONTINUE                                                                 HIS01000
      ISWTH = ISWTH + 1                                                        HIS01010
C                                                                              HIS01020
CCC   ZERO OUT PART OF FIELD HISTOGRAM ARRAY                                   HIS01030
C                                                                              HIS01040
      DO 20    I=1,NOFEAT                                                      HIS01050
      DO 20    J=1,256                                                         HIS01060
   20 FILHIS(I,J) = 0                                                          HIS01070
C                                                                              HIS01080
CCC   SCALING FACTORS USED FOR PLOTTING ROUTINE                                HIS01090
C                                                                              HIS01100
      XSCALE = FLOAT(1-XSIZ)/(XHGH-XLOW)                                       HIS01110
      XSHFT  = -XHGH*XSCALE + 1.0                                              HIS01120
C                                                                              HIS01130
CCC   PRINT FIELD STATS                                                        HIS01140
C                                                                              HIS01150
      CALL FLDINT(FLDINF,FETVEC,NOFEAT)                                        HIS01160
      LINES =(FLDINF(2)-FLDINF(1)) / FLDINF(3) + 1                             HIS01170
      FLOPTS=0                                                                 HIS01180
      DO 30     I=1,LINES                                                      HIS01190
      N = 1                                                                    HIS01200
      CALL LINERD(IDATA,ENDTAP)                                                HIS01210
      IF (ENDTAP .NE. 0)   GO TO 33                                            HIS01220
      IF(I.NE.1) GO TO 101                                                     HIS01230
      II=LINSTH                                                                HIS01240
      GO TO 102                                                                HIS01250
  101 II=II+LININC                                                             HIS01260
  102 CALL FDLINT(VERTCS,NC    ,FL,II,NS,JJ)                                   HIS01270
      KC=0                                                                     HIS01280
      DO 30    J=1,NOFEAT                                                      HIS01290
      L=1                                                                      HIS01300
      DO 34 KK=1,NSAMP                                                         HIS01310
      KPT=(KK-1)*SAMINC+SAMSTR                                                 HIS01320
      DO 103 JK=L,JJ,2                                                         HIS01330
      IF(KPT.LT.FL(JK)) GO TO 34                                               HIS01340
      IF(KPT.GT.FL(JK+1)) GO TO 105                                            HIS01350
      IF(J.EQ.1) FLOPTS=FLOPTS+1                                               HIS01360
      IPOS = (J-1)*NSAMP + KK                                                  HIS01370
      K    = IDATA(IPOS)                                                       HIS01380
      IF (K .LT.1)         K = 1                                               HIS01390
      FILHIS(J,K) = FILHIS(J,K) + 1                                            HIS01400
      IF(HISVEC(N).NE.FETVEC(J)) GO TO 34                                      HIS01410
      KC=1                                                                     HIS01420
      IPUT = IDATA(IPOS) * XSCALE + XSHFT + 0.501                              HIS01430
      IF (IPUT .LT. 1)    IPUT = 1                                             HIS01440
      IF (IPUT .GT. XSIZ)         IPUT = XSIZ                                  HIS01450
      FLOTAL(N,IPUT) = FLOTAL(N,IPUT) + 1                                      HIS01460
      TOTTAL(N,IPUT) = 1 + TOTTAL(N,IPUT)                                      HIS01470
      IF (KK .EQ. NSAMP)    N = N + 1                                          HIS01480
      GO TO 34                                                                 HIS01490
  105 L=L+2                                                                    HIS01500
      IF(L.GT.JJ) GO TO 106                                                    HIS01510
  103 CONTINUE                                                                 HIS01520
   34 CONTINUE                                                                 HIS01530
      IF(KK.EQ.NSAMP) GO TO 30                                                 HIS01540
  104 CONTINUE                                                                 HIS01550
      IF(KC.EQ.1) N=N+1                                                        HIS01560
      KC=0                                                                     HIS01570
   30 CONTINUE                                                                 HIS01580
```

FILE: HISTGM FORTRAN A

```
 33   CONTINUE                                                          MISS01590
      IP = 0                                                            MISS01600
      CALL HISTIC(FILHIS,IP,IFLD,      VERTCS,NC)                       MISS01610
      WRITE(HISFIL) ((FILHIS(I,J),J=1,256),I=1,NOFEAT)                  MISS01620
      IF(HF.EQ.1) GO TO 60                                              MISS01630
      IF (NOHIST .EQ. 0)  GO TO 10                                      MISS01640
C                                                                       MISS01650
CCC   CALL PLOTTING ROUTINE TO PLOT HISTOGRAM FOR THE FIELDS            MISS01660
C                                                                       MISS01670
      CALL HSTGRM(FLDTAL,IDATA,FLDNAM,2,XSIZ,XHGH,XLOW,YSIZ,            MISS01680
     *NOHIST,FLDPTS,HISVEC)                                             MISS01690
      GO TO 10                                                          MISS01700
C                                                                       MISS01710
CCC   WRITE TOTAL HIST ON TAPE --UNIT 13                                MISS01720
C                                                                       MISS01730
   60 CONTINUE                                                          MISS01740
      REWIND HISFIL                                                     MISS01750
      DO 104 I=1,NOFEAT                                                 MISS01760
      DO 104 J=1,256                                                    MISS01770
  104 FILHIS(I,J)=0                                                     MISS01780
      DO 107 K=1,NOFLD                                                  MISS01790
      JK=NOFEAT*256                                                     MISS01800
      READ(HISFIL) (IDATA(I),I=1,JK)                                    MISS01810
      M=0                                                               MISS01820
      DO 108 I=1,NOFEAT                                                 MISS01830
      DO 108 J=1,256                                                    MISS01840
      M=M+1                                                             MISS01850
  108 FILHIS(I,J)=IDATA(M)+FILHIS(I,J)                                  MISS01860
  107 CONTINUE                                                          MISS01870
      REWIND HISFIL                                                     MISS01880
      WRITE(HISFIL)  NOFEAT, (FETVEC(I),I=1,NOFEAT)                     MISS01890
      WRITE(HISFIL) ((FILHIS(I,J),J=1,256),I=1,NOFEAT)                  MISS01900
      REWIND HISFIL                                                     MISS01910
C                                                                       MISS01920
CCC   PRINT TOTAL STATS                                                 MISS01930
C                                                                       MISS01940
      IP = -1                                                           MISS01950
      CALL HISTIC(FILHIS,IP,IFLD,      VERTCS,NC)                       MISS01960
      IF (NOHIST .EQ. 0)      RETURN                                    MISS01970
      IF (NOFLD .EQ. 1) RETURN                                          MISS01980
C                                                                       MISS01990
CCC   CALL PLOTTING ROUTINE TO PLOT TOTAL HISTOGRAM                     MISS02000
C                                                                       MISS02010
      CALL HSTGRM(TOTTAL,IDATA,TOTAL,3,XSIZ,XHGH,XLOW,YSIZ,             MISS02020
     *NOHIST,FLDPTS,HISVEC)                                             MISS02030
      RETURN                                                            MISS02040
C 200 FORMAT(13A6,A2)                                                   MISS02050
      END                                                               MISS02060
```

FILE: HISTIC FORTRAN A

```
      SUBROUTINE HISTIC(IMG,NI,IFLD,      VERTCS,NC)                          M|S00010
C|****************************************************************C|S00020
C|    PURPOSE  -- COMPUTES AND DISPLAYS STATS FOR HISTOGRAM ROUTINE  C|S00030
C|****************************************************************C|S00040
C     INCLUDE COMMK3.LIST                                                    M|S00050
C     INCLUDE COMMK4.LIST                                                    M|S00060
C     INCLUDE COMMK5.LIST                                                    M|S00070
      COMMON /GRCALK/MARFET,NOFEAT,NOFET2,FETVEC(30),                        M|S00080
     *               FETVC2(30),FLDINF(6),INFMT,FILESV,NOHIST,               M|S00090
     *               HISVEC(30),NOFLD,          FLOPTS                      M|S00100
     *,XSIZ,XLOW,XHGH,YSIZ                                                   M|S00110
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                         M|S00120
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                      M|S00130
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                   M|S00140
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,              M|S00150
     *              HISFIL,HISKEY,INFORM,ERRPTP,ERRKEY,MAPUNT,NOFILE,        M|S00160
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                     M|S00170
     *  ,NHSTUN,NHSTFI,SCTHUN,MAPFIL                                         M|S00180
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                   M|S00190
     *  CRDUNT,PRTUNT,RANDIO                                                 M|S00200
CSEND                                                                        M|S00210
      DIMENSION RANGE(30,2),ZMEAN(30), STDDEV(30),                           M|S00220
     * NRANGE(30,2),IMG(NOFEAT,256),IFLD(50,24)                              M|S00230
      INTEGER VERTCS(1),OP,CP,COMMA                                          M|S00240
      DATA OP/'('/,CP/')'/,COMMA/','/                                        M|S00250
      INTEGER FLDINF,FETVEC,HEAD,FILESV                                      M|S00260
      REAL NRANGE                                                            M|S00270
C                                                                            M|S00280
C     COMPUTES THE DATA RANGE                                                M|S00290
C                                                                            M|S00300
      DO 10       I = 1,NOFEAT                                               M|S00310
      DO 20       J = 1,256                                                  M|S00320
      ISTR        = J                                                        M|S00330
      IF (IMG(I,J) .NE. 0)   GO TO 30                                        M|S00340
   20 CONTINUE                                                               M|S00350
   30 RANGE(I,1) = J                                                         M|S00360
      K=257                                                                  M|S00370
   40 K=K-1                                                                  M|S00380
      IF (IMG(I,K) .NE.0)    GO TO 50.                                       M|S00390
      IF(K.GT.ISTR) GO TO 40                                                 M|S00400
   50 RANGE(I,2) = K                                                         M|S00410
C                                                                            M|S00420
C     COMPUTES THE MEAN -- (1/N) * ( 1*IMG(1)+ 2*IMG(2)+ ... N*IMG(N) )      M|S00430
C                                                                            M|S00440
      STDDV1 = 0                                                             M|S00450
      AMEAN  = 0                                                             M|S00460
      N      = 0                                                             M|S00470
      DO 60       L = 1,256                                                  M|S00480
      N           = IMG(I,L) + N                                             M|S00490
      AMEAN       = L*IMG(I,L)                                               M|S00500
      STDDV       = AMEAN * L                                                M|S00510
      BMEAN       = AMEAN + AMEAN                                            M|S00520
   60 STDDV1      = STDDV + STDDV1                                           M|S00530
      ZMEAN(I)    = BMEAN / N                                                M|S00540
      STDDV1 = STDDV1 /N                                                     M|S00550
C                                                                            M|S00560
C     COMPUTES THE STD. DEV. -- SQRT.( (1/N)*(1**2 *IMG(1) + 2**2 * IMG(1)  M|S00570
C                                          2) + ... + N**2 * IMG(N) - MEAN** M|S00580
C                                          2 )                               M|S00590
C                                                                            M|S00600
      A           = STDDV1 - ZMEAN(I)**2                                     M|S00610
      STDDEV(I) = SQRT(A)                                                    M|S00620
C                                                                            M|S00630
C     COMPUTES NORMALIZED RANGE                                              M|S00640
C                                                                            M|S00650
      NRANGE(I,1) = ZMEAN(I) - 3*STDDEV(I)                                   M|S00660
   10 NRANGE(I,2) = ZMEAN(I) + 3*STDDEV(I)                                   M|S00670
      IN=NC-1                                                                M|S00680
      NNC=2*(IN)                                                             M|S00690
C                                                                            M|S00700
C                                                                            M|S00710
      WRITE(6,HEAD)                                                          M|S00720
      IF (NI.EQ. -1)  GO TO 85                                               M|S00730
      WRITE(6,510)                                                           M|S00740
      DO 70       II = 1,NOFEAT                                              M|S00750
   70 WRITE(6,520) FETVEC(II),IFLD(NOFLD,21),IN,FLDINF(6),FLDINF(3),         M|S00760
     * ((OP,VERTCS(J),COMMA,VERTCS(J+1),CP),J=1,NNC,2)                       M|S00770
      WRITE(6,530)                                                           M|S00780
   75 CONTINUE                                                               M|S00790
```

10-29

425

FILE: HISTIC FORTRAN A

```
      DO 80      I3 = 1,NOFEAT                                                  HIS00800
   80 WRITE(6,540) FETVEC(I3),RANGE(I3,1),RANGE(I3,2),ZMEAN(I3),              HIS00810
     *STDDEV(I3),NRANGE(I3,1),NRANGE(I3,2)                                    HIS00820
      RETURN                                                                  HIS00830
   85 WRITE(6,550)                                                            HIS00840
      IF(NOFLD.GT.50) NOFLD=50                                                HIS00850
      DO 1 I=1,NOFLD                                                          HIS00860
      IN=IFLD(I,24)-1                                                         HIS00870
      NNC=IN*2                                                                HIS00880
      KJ=10                                                                   HIS00890
      IF(NNC.LE.10) KJ=NNC                                                    HIS00900
      DO 2 J=1,NNC                                                            HIS00910
    2 VERTCS(J)=IFLD(I,J)                                                     HIS00920
    1 WRITE(6,560) IFLD(I,21),IN,IFLD(I,22),IFLD(I,23),                       HIS00930
     *((OP,VERTCS(K),COMMA,VERTCS(K+1),CP),K=1,KJ,2)                          HIS00940
      IF(NNC.LE.10) GO TO 2017                                                HIS00950
      WRITE(6,561) ((OP,VERTCS(K),COMMA,VERTCS(K+1),CP),K=11,NNC,2)           HIS00960
 2017 CONTINUE                                                                HIS00970
      IF (FILES. .EQ. -2)  WRITE(6,565)                                       HIS00980
      WRITE(6,570)                                                            HIS00990
      GO TO 75                                                                HIS01000
  530 FORMAT( ///////  T53,                                                   HIS01010
     *                'HISTOGRAM STATISTICS'// 14X,'CHANNEL',9X,'DATA',     HIS01020
     *       ' RANGE',4X,'MEAN', 7X,'STANDARD DEVIATION', 9X,'NORMALIZED'    HIS01030
     *      ,' RANGE'/ 86X,'(MEAN + AND - 3 STD DEV)'/)                      HIS01040
  540 FORMAT(18X,I2,11X,F5.1,2X,F5.1,7X,F5.1,12X,F5.1,17X,F6.1,2X,F6.1)       HIS01050
  510 FORMAT(////T53.'DATA BLOCK(S) HISTOGRAMMED'//,                          HIS01060
     * T24,'NO. OF   SAMPLE  LINE'/                                          HIS01070
     *,T3,'CHANNEL  FIELDNAME  VERTICES    INC   INC  VERTICES(SAMPLE,LI     HIS01080
     *NE)')                                                                   HIS01090
  550 FORMAT(////T53.'DATA BLOCK(S) HISTOGRAMMED'//,                          HIS01100
     * T24,'NO. OF   SAMPLE  LINE'/                                          HIS01110
     *,T12  ,      'FIELDNAME  VERTICES    INC   INC  VERTICES(SAMPLE,LI     HIS01120
     *NE)')                                                                   HIS01130
  520 FORMAT(4X,I2,7X,A4,8X,I2,7X,I4,2X,I4,1X,                                HIS01140
     * 5(A1,I4,A1,I4,A1,2X)/T46,5(A1,I4,A1,I4,A1,2X))                        HIS01150
  540 FORMAT(13X,A4,3X,I2,7X,I4,2X,I4,1X,                                     HIS01160
     *5(A1,I4,A1,I4,A1,2X))                                                  HIS01170
  541 FORMAT(T46,5(A1,I4,A1,I4,A1,2X))                                        HIS01180
  565 FORMAT(T2,'ONLY THE FIRST 50 FIELD DESCRIPTIONS WERE PRINTED,           HIS01190
     * BUT ALL THE FIELDS WERE INCLUDED IN THE TOTAL HISTOGRAMMED STATS'      HIS01200
     *)                                                                       HIS01210
  570 FORMAT(////// T60, 'TOTAL'//    T53,                                    HIS01220
     *                'HISTOGRAM STATISTICS'// 14X,'CHANNEL',9X,'DATA',     HIS01230
     *       ' RANGE',4X,'MEAN', 7X,'STANDARD DEVIATION', 9X,'NORMALIZED'    HIS01240
     *      ,' RANGE'/ 86X,'(MEAN + AND - 3 STD DEV)'/)                      HIS01250
      END                                                                     HIS01260
```

FILE: I4A1BN FORTRAN A

```
      SUBROUTINE I4A1BN(IFLD,NCHFLD,NCVTED)                                  I4A00010
C  DAVID LEE SMITH  9 SEPTEMBER 1977.                                        I4A00020
C  THIS SUBROUTINE ACCEPTS AN ARRAY OF EBCDIC CHARACTERS AND CONVERTS        I4A00030
C  EBCDIC DIGITS TO A BINARY INTEGER.                                        I4A00040
C                                                                            I4A00050
C  CALLING SEQUENCE:                                                         I4A00060
C     CALL I4A1BN( FIELD, LENGTH, OUTPUT )                                   I4A00070
C  WHERE  FIELD IS THE FIRST WORD OF AN ARRAY OF EBCDIC CHARACTERS           I4A00080
C                TO BE CONVERTED TO BINARY.  CHARACTERS STORED ONE PER       I4A00090
C                WORD, LEFT JUSTIFIED, AS BY AN A1 FORMAT.                   I4A00100
C          LENGTH IS THE NUMBER OF CHARACTERS IN THE FIELD.                  I4A00110
C    AND   OUTPUT IS THE ONE WORD RESULT.                                    I4A00120
      INTEGER * 4  IDUM(2), IFLD(20)                                         I4A00130
      LOGICAL * 1  L(8)                                                      I4A00140
      EQUIVALENCE (L(1),IDUM(1)),(ILCH,IDUM(1)),(ICHAR,IDUM(2))              I4A00150
      DATA ICHAR / 0 /                                                       I4A00160
      DATA IR0 / 240 /                                                       I4A00170
      DATA IR9 / 249 /                                                       I4A00180
      DATA IRBL / 64 /                                                       I4A00190
      DATA IRPL / 78 /                                                       I4A00200
      DATA IRMI / 96 /                                                       I4A00210
      NCVTED = 0                                                             I4A00220
      IERFLG = 0                                                             I4A00230
      MINUS = 1                                                              I4A00240
      DO 30 I = 1, NCHFLD                                                    I4A00250
      ILCH = IFLD( I )                                                       I4A00260
      L(8) = L(1)                                                            I4A00270
      IF ( ICHAR .LT. IR0 ) GO TO 10                                         I4A00280
      IF ( ICHAR .GT. IR9 ) GO TO 10                                         I4A00290
      JDIG = I                                                               I4A00300
      GO TO 200                                                              I4A00310
10    NEXT = I + 1                                                           I4A00320
      IF ( ICHAR .EQ. IRBL ) GO TO 30                                        I4A00330
      IF ( ICHAR .EQ. IRPL ) GO TO 100                                       I4A00340
      IF ( ICHAR .NE. IRMI ) GO TO 20                                        I4A00350
      MINUS = - MINUS                                                        I4A00360
      GO TO 100                                                              I4A00370
20    IERFLG = I                                                             I4A00380
30    CONTINUE                                                               I4A00390
      IERFLG = NCHFLD + 1                                                    I4A00400
      GO TO 240                                                              I4A00410
100   IF ( NEXT .GT. NCHFLD ) GO TO 130                                      I4A00420
      DO 120 I = NEXT, NCHFLD                                                I4A00430
      ILCH = IFLD( I )                                                       I4A00440
      L(8) = L(1)                                                            I4A00450
      IF (ICHAR .LT. IR0 ) GO TO 110                                         I4A00460
      IF (ICHAR .GT. IR9 ) GO TO 110                                         I4A00470
      JDIG = I                                                               I4A00480
      GO TO 200                                                              I4A00490
110   IF ( ICHAR .EQ. IRBL ) GO TO 120                                       I4A00500
      IERFLG = I                                                             I4A00510
120   CONTINUE                                                               I4A00520
130   IERFLG = NCHFLD + 1                                                    I4A00530
      GO TO 240                                                              I4A00540
200   DO 230 I = JDIG, NCHFLD                                                I4A00550
      ILCH = IFLD( I )                                                       I4A00560
      L(8) = L(1)                                                            I4A00570
      IF ( ICHAR .LT. IR0 ) GO TO 210                                        I4A00580
      IF ( ICHAR .LE. IR9) GO TO 220                                         I4A00590
210   IF (ICHAR .NE. IRBL ) IERFLG = I                                       I4A00600
      ICHAR = IR0                                                            I4A00610
220   IVAL = ICHAR - IR0                                                     I4A00620
      NCVTED = NCVTED * 10  -  IVAL                                          I4A00630
230   CONTINUE                                                               I4A00640
      IF ( MINUS .EQ. 1 ) NCVTED = - NCVTED                                  I4A00650
240   IF ( IERFLG .EQ. 0 ) GO TO 250                                         I4A00660
      NCH = NCHFLD                                                           I4A00670
      IF ( NCH .GT. 80 ) NCH = 80                                            I4A00680
      WRITE ( 6, 1000 ) IERFLG, NCHFLD, (IFLD(K), K = 1, NCH )               I4A00690
1000  FORMAT(' EBCDIC TO BINARY INTEGER CONVERSION ERROR'/' AT CHARACTER'I4A00700
     1 ,I5,' OF ',I5,' CHARACTER FIELD:'/1X,80A1)                           I4A00710
250   RETURN                                                                 I4A00720
      END                                                                    I4A00730
```

```
FILE: LARMAN   FORTRAN  A

      SUBROUTINE LARMAN(UNIT,FILE,NOCLS,TOTSUB,NOFEAT,TOTFLD,TOTVRT,     LAB00010
     *                  FETVEC,FLDSAV,VERTEX,CLSNMS,NOSUBS,SUBNM,N,      LAB00020
     *  STADPS,VARSIZ,PUNCH,SUBVEC,PRNSTS,SWTCH)                         LAB00030
C*****************************************************************     LAB00040
C*                                                                      LAB00050
C*                                                                      LAB00060
C*    SUBROUTINE STATFL WILL WRITE A STATISTICS FILE IN THE FORMAT      LAB00070
C*    EXPECTED BY THE LARSYS PROGRAM.  IT WILL ALSO PUNCH THE MODULE    LAB00080
C*    DECK IF REQUESTED.                                                LAB00090
C*                                                                      LAB00100
C*    ARGUMENTS ARE ALL INPUT.                                          LAB00110
C*                                                                      LAB00120
C*      UNIT   - FORTRAN UNIT NUMBER WHERE THE FILE IS TO BE WRITTEN.   LAB00130
C*      FILE   - FILE NUMBER ON 'UNIT' WHERE THE FILE IS TO BE WRITTEN. LAB00140
C*      NOCLS  - NUMBER OF CLASSES.                                     LAB00150
C*      TOTSUB - TOTAL NUMBER OF SUBCLASSES FOR ALL CLASSES.            LAB00160
C*      NOFEAT - NUMBER OF CHANNELS.                                    LAB00170
C*      TOTFLD - TOTAL NUMBER OF TRAINING FIELDS.                       LAB00180
C*      TOTVRT - TOTAL NUMBER OF VERTICES FOR ALL TRAINING FIELDS.      LAB00190
C*      FETVEC - VECTOR CONTAINING THE CHANNEL NUMBERS FOR WHICH        LAB00200
C*               THE STATISTICS WERE COMPUTED. (DIMENSIONED - NOFEAT)   LAB00210
C*     FLDSAV - ARRAY DIMENSIONED - 4 BY TOTFLD AND CONTAINING THE      LAB00220
C*              FOLLOWING INFORMATION FOR EACH TRAINING FIELD.          LAB00230
C*                FLDSAV(1,I) - NAME OF FIELD I.                        LAB00240
C*                FLDSAV(2,I) - CLASS NO. THAT FIELD I BELONGS TO.      LAB00250
C*                FLDSAV(3,I) - SUBCLASS NO. THAT FIELD I BELONGS TO.   LAB00260
C*                              ZERO IF THE FIELD IS NOT ASSOCIATED     LAB00270
C*                              WITH A SUBCLASS.                        LAB00280
C*                FLDSAV(4,I) - NO. OF VERTICES FOR THIS FIELD.         LAB00290
C*                              INCLUDING THE CLOSURE POINT.            LAB00300
C*      VERTEX - ARRAY CONTAINING VERTICES FROM ALL TRAINING FIELDS.    LAB00310
C*               DIMENSIONED - 2*TOTVRT. THE VERTICES SHOULD BE STORED  LAB00320
C*               SAMPLE NO. FIRST THEN LINE NUMBER FOR EACH VERTEX.     LAB00330
C*               CLOSURE POINTS MUST BE INCLUDED FOR EACH VERTEX.       LAB00340
C*               (I.E. THE FIRST VERTEX IS REPEATED AS THE LAST VERTEX) LAB00350
C*      CLSNMS - ARRAY CONTAINING ALPHANUMERIC CLASS NAMES.             LAB00360
C*      NOSUBS - ARRAY CONTAINING THE NO. OF SUBCLASSES IN EACH CLASS.  LAB00370
C*      SUBNM  - ARRAY CONTAINING ALPHANUMERIC SUBCLASS NAMES.          LAB00380
C*      N      - ARRAY CONTAINING THE NO. OF PIXELS IN EACH SUBCLASS.   LAB00390
C*      PUNCH  - IF PUNCH=1 THE MODULE DECK WILL BE PUNCHED.            LAB00400
C*    SWTCH = 1 CALLED BY ISOCLS                                        LAB00410
C*    SWTCH = 2 CALLED BY LABEL                                         LAB00420
C*    SWTCH = 3 CALLED BY STAT                                          LAB00430
C*    SWTCH = 4 CALLED BY TRSTAT                                        LAB00440
C*                                                                      LAB00450
C*                                                                      LAB00460
C*****************************************************************     LAB00470
C*                                                                      LAB00480
      IMPLICIT INTEGER(A-Z)                                             LAB00490
      DIMENSION FETVEC(NOFEAT),FLDSAV(4,TOTFLD),VERTEX(TOTVRT)          LAB00500
      DIMENSION CLSNMS(NOCLS),NOSUBS(NOCLS),SUBNM(TOTSUB),N(TOTSUB)     LAB00510
      DIMENSION SUBVEC(1)                                               LAB00520
      DATA PCHUNT /7/                                                   LAB00530
      REAL COVAR(465),MEANS(30)                                         LAB00540
C*                                                                      LAB00550
C*    POSITION 'UNIT' TO CORRECT FILE NO.                               LAB00560
C*                                                                      LAB00570
      SAVTAP=UNIT                                                       LAB00580
      REWIND SAVTAP                                                     LAB00590
      NF = FILE                                                         LAB00600
      CALL FSBSFL(SAVTAP,NF,ISTAT1)                                     LAB00610
      IF(ISTAT1.EQ.0)GO TO 1                                            LAB00620
      WRITE(4,220)FILE                                                  LAB00630
  220 FORMAT(/T5,'ERROR IN POSITIONING SIG. EXTENSION TAPE TO FILE',I3/  LAB00640
     *  T5,'OUTPUT FILE NOT WRITTEN')                                    LAB00650
    1 CONTINUE                                                          LAB00660
      IA=1                                                              LAB00670
      DATA PCHUNT/7/                                                    LAB00680
      IF(PUNCH.NE.1)GO TO 6                                             LAB00690
      WRITE(PCHUNT,110)                                                 LAB00700
      WRITE(PCHUNT,120)NOCLS,TOTSUB,NOFEAT,TOTFLD,TOTVRT                LAB00710
      WRITE(PCHUNT,130)(FETVEC(I),I=1,NOFEAT)                           LAB00720
    6 CONTINUE                                                          LAB00730
      IF (ISTAT1 .NE. 0)  GO TO 11                                      LAB00740
      WRITE(SAVTAP)NOCLS,TOTSUB,NOFEAT,TOTFLD,TOTVRT,                   LAB00750
     *             (FETVEC(J),J=1,NOFEAT)                               LAB00760
   11 CONTINUE                                                          LAB00770
      DO 2 I=1,TOTFLD                                                   LAB00780
      NV = FLDSAV(4,I)                                                  LAB00790
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: LABMAN FORTRAN A

```
      IE=IB +NV*2 -1                                                    LAB00800
      IF(PUNCH.NE.1)GO TO 7                                             LAB00810
      WRITE(PCHUNT,135) (FLDSAV(J,I),J=1,4)                             LAB00820
      WRITE(PCHUNT,140) (VERTEX(J),J=IB,IE)                             LAB00830
    7 CONTINUE                                                          LAB00840
      IF(ISTAT1.NE.0)GO TO 3                                            LAB00850
      WRITE(SAVTAP) (FLDSAV(J,I),J=1,4)                                 LAB00860
      WRITE(SAVTAP) (VERTEX(J),J=IB,IE)                                 LAB00870
    3 IB=IE+1                                                           LAB00880
    2 CONTINUE                                                          LAB00890
      IF(PUNCH.NE.1)GO TO 8                                             LAB00900
      WRITE(PCHUNT,145)(CLSNMS(I),I=1,NOCLS)                            LAB00910
      WRITE(PCHUNT,150)(NOSUBS(I),I=1,NOCLS)                            LAB00920
      WRITE(PCHUNT,155)(SUBNM(I),I=1,TOTSUB)                            LAB00930
    8 CONTINUE                                                          LAB00940
      IF (ISTAT1 .NE. 0)  GO TO 18                                      LAB00950
      WRITE(SAVTAP)(CLSNMS(I),I=1,NOCLS),(NOSUBS(I),I=1,NOCLS),         LAB00960
     *             (SUBNM(I),I=1,TOTSUB)                                LAB00970
   18 CONTINUE                                                          LAB00980
      III = 0                                                           LAB00990
      WRITE(6,225)                                                      LAB01000
  225 FORMAT(1H1)                                                       LAB01010
      NUMSUB = 0                                                        LAB01020
      TOTSTA = 0                                                        LAB01030
      DO 20 ICLAS=1,NOCLS                                               LAB01040
      TOTSTA = NUMSUB + TOTSTA                                          LAB01050
      NUMSUB=NOSUBS(ICLAS)                                              LAB01060
      DO 20 J=1,NUMSUB                                                  LAB01070
      III=III+1                                                         LAB01080
C     STATS ARE COMING FROM ISOCLS                                      LAB01090
      IF ( SWTCH .NE. 1) GO TO 50                                       LAB01100
      MEAN1 = STADRS + (VARSIZ*NOFEAT)*TOTSTA                           LAB01110
      MEAN2 = MEAN1 + (J-1)*NOFEAT                                      LAB01120
      COVAR1 = MEAN1 + NUMSUB*NOFEAT + VARSIZ*(J-1)                     LAB01130
      KK=III                                                            LAB01140
      GO TO 80                                                          LAB01150
C     STAT ARE COMING LABEL                                             LAB01160
   50 IF ( SWTCH .NE. 2 ) GO TO 60                                      LAB01170
      KK = SUBVEC(III)                                                  LAB01180
      MEAN1 = STADRS + VARSIZ* TOTSUB                                   LAB01190
      MEAN2 = MEAN1 + NOFEAT*(KK-1)                                     LAB01200
      COVAR1 = STADRS + VARSIZ*(KK-1)                                   LAB01210
      GO TO 80                                                          LAB01220
C     STATS ARE COMING FROM STAT                                        LAB01230
   60 CONTINUE                                                          LAB01240
      GO TO 80                                                          LAB01250
C     STATS ARE COMING TRSTAT                                           LAB01260
   80 CONTINUE                                                          LAB01270
C                                                                       LAB01280
C     READ MEANS AND COVARIANCES INTO CORE FROM DRUM                    LAB01290
C                                                                       LAB01300
      CALL BREAD(COVAR1,COVAR,VARSIZ,ISTAT)                             LAB01310
   30 IF (ISTAT .EQ. 1) GO TO 30                                        LAB01320
      CALL BREAD(MEAN2,MEANS,NOFEAT,ISTAT1)                             LAB01330
   40 IF (ISTAT1 .EQ. 1) GO TO 40                                       LAB01340
C                                                                       LAB01350
      IF(PUNCH.NE.1)GO TO 9                                             LAB01360
      WRITE(PCHUNT,170)N(KK)                                            LAB01370
      WRITE(PCHUNT,90) (MEANS(K),K=1,NOFEAT)                            LAB01380
      WRITE(PCHUNT,100)(COVAR(K),K=1,VARSIZ)                            LAB01390
    9 CONTINUE                                                          LAB01400
      IF (ISTAT1 .NE. 0)  GO TO 20                                      LAB01410
      WRITE(SAVTAP)N(KK),(COVAR(K),K=1,VARSIZ),(MEANS(K),K=1,NOFEAT)    LAB01420
      IF(PRNSTS.NE.1) GOTO 20                                           LAB01430
C                                                                       LAB01440
C     PRINTS THE STATS ON THE LINE PRINTER                              LAB01450
C                                                                       LAB01460
      DATA RCDTWO/'2'/                                                  LAB01470
      WRITE(6,65)                                                       LAB01480
   65 FORMAT(/)                                                         LAB01490
      WRITE(6,310)CLSNMS(ICLAS),SUBNM(III)                              LAB01500
  310 FORMAT(//'  CLASS    : ',A6/'  SUBCLASS: ',A6)                    LAB01510
      DO 340 LOC=1,NOFEAT,12                                            LAB01520
      STOP=LOC+11                                                       LAB01530
      IF(STOP.GT.NOFEAT)STOP=NOFEAT                                     LAB01540
  340 WRITE(6,350)(MEANS(I),I=LOC,STOP)                                 LAB01550
  350 FORMAT('0 MEAN:',3X,12F9.2)                                       LAB01560
      WRITE(6,360)                                                      LAB01570
  360 FORMAT(//'  COVARIANCE MATRIX:')                                  LAB01580
```

FILE: LARMAN FORTRAN A

```
      CALL WRTMTX(COVAR(1),NOFEAT,RCDTWO)                                LAB01590
   20 CONTINUE                                                           LAB01600
      IF (ISTAT1 .NE. 0)  GO TO 19                                       LAB01610
      ENDFILE SAVTAP                                                     LAB01620
   19 CONTINUE                                                           LAB01630
      WRITE(6,180)NOCLS,TOTSUB                                           LAB01640
      WRITE(6,190)                                                       LAB01650
      K=0                                                                LAB01660
      I=1                                                                LAB01670
   21 NSUBS=NOSUBS(I)                                                    LAB01680
      WRITE(6,205)                                                       LAB01690
      WRITE(6,200)I,CLSNMS(I),NSUBS                                      LAB01700
      WRITE(6,205)                                                       LAB01710
      DO 25 J=1,NSUBS                                                    LAB01720
      K=K+1                                                              LAB01730
      WRITE(6,210)K,SUBNM(K)                                             LAB01740
   25 CONTINUE                                                           LAB01750
      I=I+1                                                              LAB01760
      IF(I.LE.NOCLS)GO TO 21                                             LAB01770
      RETURN                                                             LAB01780
   90 FORMAT('MEAN ',5E15.8)                                             LAB01790
  100 FORMAT('COVAR',5E15.8)                                             LAB01800
  110 FORMAT('MODULE DECK FROM SLABEL')                                  LAB01810
  120 FORMAT('NOCLS ',I4,' NOSUB',I4,' NOFEAT',I3,' NOFLD',I4,' TOTVRT',LAB01820
     *  I5)                                                             LAB01830
  130 FORMAT('VECTR',5X,30I2)                                            LAB01840
  135 FORMAT(A6,4X,3(I2,8X))                                             LAB01850
  140 FORMAT('VERTICES ',14I5)                                           LAB01860
  145 FORMAT(('CLSDESC ',9(2X,A6)))                                      LAB01870
  150 FORMAT(('NOSUBS ',24(1X,I2)))                                      LAB01880
  155 FORMAT(('SUBDESC ',10(A6,1X)))                                     LAB01890
  160 FORMAT('FREQ ',12F6.2)                                             LAB01900
  170 FORMAT('NOPTS',7X,I8)                                              LAB01910
  180 FORMAT(///' THE STATISTICS FILE FOR',I4,' CLASSES AND',I4,         LAB01920
     *  ' SUBCLASSES HAS BEEN WRITTEN'/)                                LAB01930
  185 FORMAT(1H+,' THE STATS WERE WRITTEN ON FILE ',I3)                  LAB01940
  190 FORMAT(' THE STATS FOR A PARTICULAR CLASS OR SUBCLASS SHOULD BE RELAB01950
     *FERRED TO IN LATER RUNS BY '/' THE FOLLOWING NAMES AND NUMBERS (W LAB01960
     *HICHEVER APPLICABLE)'/)                                         LAB01970
  200 FORMAT(5X,'CLASS',I3,2X,A6,5X,'SUBCLASSES (TOTAL=',I3,')')        LAB01980
  205 FORMAT(/)                                                          LAB01990
  210 FORMAT(25X,I3,2X,A6)                                               LAB02000
      END                                                                LAB02010
```

FILE: LAREAD FORTRAN A

```
      FUNCTION LAREAD(FLDNAM,VERTCS,FLDINF,NC)                                  LAR00010
      IMPLICIT INTEGER(A-Z)                                                     LAR00020
      DIMENSION CARD(62),      FLDINF(6),VERTCS(2,11),VER(2,10)                 LAR00030
      DIMENSION ACARD(20)                                                       LAR00040
      DATA BLANK/' '/,COMMA/','/,OP/'('/,CP/')'/,                               LAR00050
     * AST/'*'/,ENDBCD/'$END'/                                                  LAR00060
      DATA C/'CLAS'/,S/'SUBC'/                                                  LAR00070
      DATA D/'DESI'/                                                            LAR00080
      DATA T/'TYPE'/                                                            LAR00090
      DO 50 I=1,2                                                               LAR00100
      DO 50 J=1,10                                                              LAR00110
   50 VER(I,J)=0                                                                LAR00120
C  SET UP REREAD BUFFER                                                       LAR00130
      RRUNIT=30                                                                 LAR00140
      CALL REREAD(RRUNIT,80)                                                    LAR00150
C  NOW PUT THE CARD INTO THE BUFFER                                           LAR00160
    1 READ(21,100)(ACARD(I),I=1,20)                                             LAR00170
  100 FORMAT(20A4)                                                              LAR00180
      WRITE(RRUNIT,100)(ACARD(I),I=1,20)                                        LAR00190
      REWIND RRUNIT                                                             LAR00200
C READ IN FIRST CARD                                                            LAR00210
      READ(RRUNIT,20) FLDNAM                                                    LAR00220
   20 FORMAT(A4)                                                                LAR00230
      REWIND RRUNIT                                                             LAR00240
      IF (FLDNAM .NE. T) GO TO 55                                               LAR00250
      LAREAD = -4                                                               LAR00260
      RETURN                                                                    LAR00270
   55 CONTINUE                                                                  LAR00280
      IF(FLDNAM.NE.D)GO TO 19                                                   LAR00290
      LAREAD=-3                                                                 LAR00300
      RETURN                                                                    LAR00310
   19 CONTINUE                                                                  LAR00320
      IF(FLDNAM    .NE.C) GO TO 24                                              LAR00330
      LAREAD=-1                                                                 LAR00340
      RETURN                                                                    LAR00350
   24 IF(FLDNAM    .NE.S) GO TO 21                                              LAR00360
      LAREAD=-2                                                                 LAR00370
      RETURN                                                                    LAR00380
   21 IF(FLDNAM    .NE.ENDBCD) GO TO 22                                         LAR00390
      LAREAD=0                                                                  LAR00400
      RETURN                                                                    LAR00410
C REREAD FIRST CARD                                                             LAR00420
   22 READ(RRUNIT,23) CARD                                                      LAR00430
   23 FORMAT(10X,62A1)                                                          LAR00440
      REWIND RRUNIT                                                             LAR00450
      COL=0                                                                     LAR00460
      II=0                                                                      LAR00470
      NC=0                                                                      LAR00480
C FIND (                                                                        LAR00490
   11 J=NXTCHR(CARD,COL)                                                        LAR00500
      IF(J.EQ.BLANK) GO TO 1                                                    LAR00510
      IF(J.EQ.AST) GO TO 10                                                     LAR00520
      IF(J.NE.OP) GO TO 3                                                       LAR00530
    9 I=0                                                                       LAR00540
      K=0                                                                       LAR00550
      KK=0                                                                      LAR00560
      NUM=0                                                                     LAR00570
    4 COL=COL+1                                                                 LAR00580
      IF(COL.GT.62) GO TO 35                                                    LAR00590
      IF(CARD(COL).EQ.BLANK) GO TO 4                                            LAR00600
      IF((I.EQ.0).AND.(CARD(COL).EQ.COMMA)) GO TO 3                             LAR00610
      IF((I.EQ.1).AND.(CARD(COL).EQ.COMMA)) GO TO 7                             LAR00620
      IF((KK.EQ.0).AND.(CARD(COL).EQ.CP)) GO TO 3                               LAR00630
      IF((KK.EQ.1).AND.(CARD(COL).EQ.CP)) GO TO 8                               LAR00640
      CALL I4A1B(CARD(COL),1,NW)                                                LAR00650
      NUM=10*NUM+NW                                                             LAR00660
      IF((NW.LT.0).OR.(NW.GT.9)) GO TO 3                                        LAR00670
      I=1                                                                       LAR00680
      IF(K.EQ.1) KK=1                                                           LAR00690
      GO TO 4                                                                   LAR00700
    7 IF(II.EQ.0) GO TO 30                                                      LAR00710
C VERTEX SAMPLE NUMBER                                                          LAR00720
      NC=NC+1                                                                   LAR00730
      VER(1,NC)=NUM                                                             LAR00740
      K=1                                                                       LAR00750
      NUM=0                                                                     LAR00760
      GO TO 4                                                                   LAR00770
    8 IF II.EQ.0) GO TO 31                                                      LAR00780
C VERTEX LINE NUMBER                                                            LAR00790
```

FILE: LAREAD FORTRAN A

```
      VER(2,NC)=NUM                                                     LAR00800
C CHECK FOR COMMA OR ASTERISK                                           LAR00810
    6 J=NXTCHR(CARD,COL)                                                LAR00820
      IF(J.EQ.BLANK) GO TO 2                                            LAR00830
      IF(J.EQ.AST) GO TO 10                                             LAR00840
      IF(J.NE.COMMA) GO TO 3                                            LAR00850
      GO TO 11                                                          LAR00860
    3 WRITE(6,13) CARD                                                  LAR00870
   13 FORMAT(' ERROR IN FIELD CARD TERMINATING RUN'/10X,62A1)            LAR00880
      CALL CMERR                                                        LAR00890
    5 WRITE(6,15) CARD                                                  LAR00900
   15 FORMAT(10X,62A1/' INCORRECT FIELD CARD,TERMINATING RUN')          LAR00910
      CALL CMERR                                                        LAR00920
C DETERMINE RECTANGULAR FIELD COORDINATES                               LAR00930
    2 IF((NC.LT.1).OR.(NC.GT.10)) GO TO 3                               LAR00940
      NT1=0                                                             LAR00950
      NT3=0                                                             LAR00960
      NT2=1000000                                                       LAR00970
      NT4=1000000                                                       LAR00980
      DO 14 N=1,NC                                                      LAR00990
      IF((VER(1,N).EQ.0).OR.(VER(2,N).EQ.0)) GO TO 5                    LAR01000
      IF(VER(1,N).GT.NT1) NT1=VER(1,N)                                  LAR01010
      IF(VER(2,N).LT.NT4) NT4=VER(2,N)                                  LAR01020
      IF(VER(2,N).GT.NT3) NT3=VER(2,N)                                  LAR01030
      IF(VER(1,N).GE.NT2) GO TO 14                                      LAR01040
      NT2=VER(1,N)                                                      LAR01050
      CNT=N                                                             LAR01060
   14 CONTINUE                                                          LAR01070
      FLDINF(1)=NT4                                                     LAR01080
      FLDINF(2)=NT3                                                     LAR01090
      FLDINF(4)=NT2                                                     LAR01100
      FLDINF(5)=NT1                                                     LAR01110
C SET UP VERTICES IN CLOCKWISE ORDER WITH SMALLEST SAMPLE FIRST         LAR01120
      DO 32 I=1,NC                                                      LAR01130
      IF(CNT.GT.NC) CNT=1                                               LAR01140
      VERTCS(1,I)=VER(1,CNT)                                            LAR01150
      VERTCS(2,I)=VER(2,CNT)                                            LAR01160
   32 CNT=CNT+1                                                         LAR01170
      VERTCS(1,NC+1)=VERTCS(1,1)                                        LAR01180
      VERTCS(2,NC+1)=VERTCS(2,1)                                        LAR01190
   35 LAREAD=1                                                          LAR01200
      NC=NC+1                                                           LAR01210
      RETURN                                                            LAR01220
   30 FLDINF(6)=NUM                                                     LAR01230
      K=1                                                               LAR01240
      NUM=0                                                             LAR01250
      GO TO 4                                                           LAR01260
   31 FLDINF(3)=NUM                                                     LAR01270
      II=1                                                              LAR01280
      GO TO 6                                                           LAR01290
C READ CONTINUATION CARD                                                LAR01300
   10 READ(21,23) CARD                                                  LAR01310
      COL=0                                                             LAR01320
      GO TO 11                                                          LAR01330
      END                                                               LAR01340
```

FILE: LINERD FORTRAN A

```
      SUBROUTINE LINERD(/IDATA/,ENDTAP)                                  LIN00010
      IMPLICIT INTEGER (A-Z)                                             LIN00020
C*                                                                       LIN00030
C*                                                                       LIN00040
C*    ENTRY FOR READING AND UNPACKING ONE SCAN LINE OF DATA              LIN00050
C*                                                                       LIN00060
      LOGICAL*1 ISCAN(4),BYTE(4),IDATA(1),IBUF(13500),IZERO(4)           LIN00070
C     INCLUDE CMBK17                                                     LIN00080
C$START                                                                  LIN00090
      COMMON /TAPERD/ IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,       LIN00100
     *  LINC,ID(200),DSL,LBUF(30),JREC(30),IBYTE(30),NBUFS,FILENO,LINEND LIN00110
     *   ,LININC,NSAMP,NOCHAN,FORMT                                      LIN00120
C$END                                                                    LIN00130
      EQUIVALENCE (SCAN,ISCAN(1)),(IZERO(1),ZERO)                        LIN00140
      EQUIVALENCE (ID(1),NRPDS ),(ID(2),NCPR   ),                        LIN00150
     *            (ID(3),NPRC  ),(ID(4),ANCLNG),                        LIN00160
     *            (ID(5),NC    ),(ID(6),NS     ),                       LIN00170
     *            (ID(7),NBITS ),(ID(8),DOI),                          LIN00180
     *            (ID(9),NDSPR),(ID(10),NCAR ),                        LIN00190
     *      (ID(11),SVD),(ID(16),PRSZ)                                 LIN00200
C  ZERO OUT IDATA                                                        LIN00210
C                                                                        LIN00220
      TOTPIX=NSAMP*NOCHAN*4                                              LIN00230
      ZERO=0                                                             LIN00240
      DO 180 J=1,TOTPIX                                                  LIN00250
  180 IDATA(J)=IZERO(4)                                                  LIN00260
      IF (READY)190,190,200                                              LIN00270
  190 WRITE (6,410)                                                      LIN00280
C*                                                                       LIN00290
  200 IADR = 4                                                           LIN00300
      MAXREC = PRSZ                                                      LIN00310
      BUF = 1                                                            LIN00320
      REC = 0                                                            LIN00330
      CALL BUFILL(REC,IUNIT,MAXREC,IBUF,NRPDS,ENDTAP,IERR)               LIN00340
      IF ( FORMT .EQ. 1) ISCAN(3) = IBUF(71)                             LIN00350
      IF ( FORMT .EQ. 1) ISCAN(4) = IBUF(72)                             LIN00360
      IF ( FORMT .EQ. 2) ISCAN(3) = IBUF(1)                              LIN00370
      IF ( FORMT .EQ. 2) ISCAN(4) = IBUF(2)                              LIN00380
C                                                                        LIN00390
      IF ( SCAN .EQ. FSCAN) GO TO 196                                    LIN00400
C                                                                        LIN00410
      CALL SEARCH(&250,&235,ENDTAP,IBUF,NRPDS,NDSPR)                     LIN00420
      MAXREC=PRSZ                                                        LIN00430
      BUF = 1                                                            LIN00440
      REC = 0                                                            LIN00450
      CALL BUFILL(REC,IUNIT,MAXREC,IBUF,NRPDS,ENDTAP,IERR)               LIN00460
  196 CONTINUE                                                           LIN00470
      ADD = (NSCAN-FSCAN)*DSL                                            LIN00480
      DO 230 IFT=1,NOCHAN                                                LIN00490
  201 IF(LBUF(IFT).EQ.BUF)GO TO 205                                      LIN00500
      CALL BUFILL(REC,IUNIT,MAXREC,IBUF,NRPDS,ENDTAP,IERR)               LIN00510
      BUF=BUF+1                                                          LIN00520
      GO TO 201                                                          LIN00530
  205 CONTINUE                                                           LIN00540
      J=JREC(IFT)                                                        LIN00550
      JJ=(J-1)*MAXREC                                                    LIN00560
C*                                                                       LIN00570
C*    CHECK STATUS OF THIS RECORD BEFORE UNPACKING                       LIN00580
C*                                                                       LIN00590
      IF (ENDTAP .EQ. -1) GO TO 250                                      LIN00600
C*                                                                       LIN00610
C*    UNPACK DATA FOR THIS FEATURE                                       LIN00620
C*                                                                       LIN00630
  220 IP = ADD + IBYTE(IFT)+JJ                                           LIN00640
      DO 225  II=1,NSAMP                                                 LIN00650
      IDATA(IADR+4*(II-1)) = IBUF(IP+SAMINC*(II-1))                      LIN00660
  225 CONTINUE                                                           LIN00670
      IADR = IADR + NSAMP*4                                              LIN00680
  230 CONTINUE                                                           LIN00690
C*    FINISHED UNPACKING ONE SCAN LINE OF DATA                           LIN00700
      IF ((NSCAN+LININC).GT.LINEND) GO TO 260                            LIN00710
C*                                                                       LIN00720
C*    MAKE SURE ALL BUFFERS FOR THIS DATA SET HAVE BEEN READ             LIN00730
C*                                                                       LIN00740
  231 IF(BUF.EQ.NBUFS)GO TO 235                                          LIN00750
      CALL BUFILL(REC,IUNIT,MAXREC,IBUF,NRPDS,ENDTAP,IERR)               LIN00760
      IF (ENDTAP .EQ. -1) GO TO 250                                      LIN00770
      BUF=BUF+1                                                          LIN00780
      GO TO 231                                                          LIN00790
```

FILE: LINERD FORTRAN A

```
  235 CONTINUE                                                                  LIN00800
      NSCAN=NSCAN+LININC                                                        LIN00810
      IF(NSCAN.LT.(FSCAN+NDSPR))RETURN                                          LIN00820
      IF (NSCAN .GT.(FSCAN+NDSPR)) GO TO 236                                    LIN00830
      FSCAN = FSCAN + NDSPR*(1 + LINC/NRPDS)                                    LIN00840
      RETURN                                                                    LIN00850
  236 FSCAN=FSCAN + NDSPR*(1 + LINC/NRPDS)                                      LIN00860
      DO 237  II=1,LINC                                                         LIN00870
  237 READ(IUNIT,420)DUMMY                                                      LIN00880
      IF (NSCAN.LT.(FSCAN+NDSPR)) GO TO 240                                     LIN00890
      READ(IUNIT,420)DUMMY                                                      LIN00900
      FSCAN=FSCAN+NDSPR                                                         LIN00910
  240 CONTINUE                                                                  LIN00920
      RETURN                                                                    LIN00930
  250 IF(NSCAN.GT.LINSTR)NSCAN=NSCAN-LININC                                     LIN00940
      WRITE (6,320)NSCAN                                                        LIN00950
      IF (FILENO .EQ. 0)  GO TO 255                                             LIN00960
C*                                                                              LIN00970
C*    BACK SPACE 1 FILE AND POSITION AT FIRST SCAN LINE                         LIN00980
C*                                                                              LIN00990
      BSKIP = (NSCAN-IFRST) * NDSPR + 1                                         LIN01000
      WRITE(6,560)BSKIP                                                         LIN01010
  560 FORMAT(' BACKSPACE',I3)                                                   LIN01020
      DO 253  II=1,BSKIP                                                        LIN01030
  253 BACKSPACE IUNIT                                                           LIN01040
      GO TO 257                                                                 LIN01050
C*                                                                              LIN01060
C*    REWIND TAPE AND POSITION AT FIRST SCAN LINE                               LIN01070
C*                                                                              LIN01080
  255 REWIND IUNIT                                                              LIN01090
      READ(IUNIT,420)DUMMY                                                      LIN01100
  257 FSCAN = IFRST                                                             LIN01110
  260 READY = -1                                                                LIN01120
      RETURN                                                                    LIN01130
  320 FORMAT(' FIELD BOUNDARY FOR THIS FIELD DEFINED BEYOND SCOPE  OF DA        LIN01140
     *TA'/' THIS FLIGHT LINE CONTAINS',I6,' SCAN LINES')                        LIN01150
  410 FORMAT(' FLDINT MUST BE CALLED TO INITIALIZE PARAMETERS FOR A NEW         LIN01160
     *FIELD')                                                                   LIN01170
  420 FORMAT(1A4)                                                               LIN01180
      END                                                                       LIN01190
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: LISTLC FORTRAN A

```
C       FIELDS - CATEGORY NAME AND DOT TYPE FOR DOT I STORED IN                 LIS00010
C                FIELD(1,I) AND FIELD(4,I)                                      LIS00020
C       STAMNT - INITALLY SET TO 1. SWITCHED TO INDICATE DOTS BEING             LIS00030
C                TAKEN FROM CURRENTLY READ CARD.                                LIS00040
C       IPT - INITIALLY SET TO 1.INDEX NUMBER FOR FIELD VERTEX INFORMATION      LIS00050
C       VERTEX - VERTEX INFORMATION FOR EACH DOT.                               LIS00060
C                                                                               LIS00070
        SUBROUTINE LISTLC(FIELDS,STAMNT,*,*,*,SWCHG,INIT,IUNIT,IFILE,IPT,       LIS00080
       *VERTEX)                                                                 LIS00090
        IMPLICIT INTEGER (A-Z)                                                  LIS00100
        REAL DUM                                                                LIS00110
        DIMENSION FIELDS(4,1),VERTEX(1),CARD(75),NDOTS(30)                      LIS00120
        DIMENSION ACARD(80)                                                     LIS00130
        LOGICAL SWITCH                                                          LIS00140
          DATA SWITCH/.TRUE./,ENDBCD/'SEN'/,                                    LIS00150
       *CATNM1/' '/                                                             LIS00160
C       INCLUDE CMMK14                                                          LIS00170
C       INCLUDE COMMK1                                                          LIS00180
        COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,                LIS00190
       *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,          LIS00200
       *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),              LIS00210
       *              KEYPTS(40),NOGRP,GRPNAM(60),GRPDEX(61),                   LIS00220
       *              GRPCHK(61),GROUPS(124)                                    LIS00230
        COMMON /DOTVEC/TYPE,CATNAM(60),NOCAT,TOTVEC,FLDINF(6),PRTKEY            LIS00240
       *              ,SIZE,LACIE                                               LIS00250
              DIMENSION IBUF(80)                                                LIS00260
CSEND                                                                           LIS00270
          IF (INIT.NE.0) GO TO 5                                                LIS00280
          REWIND IUNIT                                                          LIS00290
          CALL FSFMFL(IUNIT,IFILE,ISTAT)                                        LIS00300
            READ (IUNIT,1010) (IBUF(I), I=1,80)                                 LIS00310
 1010         FORMAT(80A1)                                                      LIS00320
          WRITE(6,1020) (IBUF(I), I=1,80)                                       LIS00330
 1020     FORMAT(1H0,80A1)                                                      LIS00340
 5        INIT = 1                                                              LIS00350
        IF(STAMNT.EQ.2)GO TO 30                                                 LIS00360
        IF(.NOT.SWITCH)GO TO 20                                                 LIS00370
            CALL REREAD(30,80)                                                  LIS00380
 10     READ(IUNIT,103)(ACARD(I),I=1,80)                                        LIS00390
 103    FORMAT(80A1)                                                            LIS00400
        WRITE(30,103)(ACARD(I),I=1,80)                                          LIS00410
        REWIND 30                                                               LIS00420
        READ(30,1000)ID,TYPES,CARD                                              LIS00430
        REWIND 30                                                               LIS00440
 1000   FORMAT(A3,1X,I1,75A1)                                                   LIS00450
        IF(TYPE.EQ.TYPES )GO TO 20                                              LIS00460
        IF(SWCHG.NE.0)GO TO 40                                                  LIS00470
        TYPE = TYPES                                                            LIS00480
C                                                                               LIS00490
CC       READ CARD                                                              LIS00500
C                                                                               LIS00510
 20     COL = 0                                                                 LIS00520
        CATNM = NXTCHR(CARD,COL)                                                LIS00530
C*      IF NEXT CHAR IS NOT A CAT. NAME, CORRECT COL COUNT TO READ NUM          LIS00540
        IF(CATNM.GT.0)GO TO 21                                                  LIS00550
        IF(CATNM.EQ.CATNM1)GO TO 23                                             LIS00560
        NOCAT=NOCAT + 1                                                         LIS00570
        CATNAM(NOCAT)=CATNM                                                     LIS00580
        CATNM1 = CATNM                                                          LIS00590
        GO TO 23                                                                LIS00600
 21     COL=COL - 1                                                             LIS00610
 23     NDCARD=0                                                                LIS00620
        CALL NUMBR(NDOTS,NDCARD,CARD,COL)                                       LIS00630
        IF(NDCARD.EQ.0)GO TO 10                                                 LIS00640
        ICNT = 0                                                                LIS00650
        STAMNT = 2                                                              LIS00660
        SWITCH = .TRUE.                                                         LIS00670
        GO TO 100                                                               LIS00680
C                                                                               LIS00690
C       TEST FOR END OF DOTS TO BE PROCESSED ON CARD                            LIS00700
C                                                                               LIS00710
 30     IF(ICNT.LT.NDCARD)GO TO 100                                             LIS00720
C                                                                               LIS00730
CC      READ NEXT CARD                                                          LIS00740
C                                                                               LIS00750
        STAMNT = 1                                                              LIS00760
        ICNT = 0                                                                LIS00770
        READ(IUNIT,103)(ACARD(I),I=1,80)                                        LIS00780
        WRITE(30,103)(ACARD(I),I=1,80)                                          LIS00790
```

```
FILE: LISTLC   FORTRAN  A

      REWIND 30                                                                 LIS00800
      READ(30,1000 )ID,TYPES,CARD                                               LIS00810
      REWIND 30                                                                 LIS00820
      IF(ID.EQ.ENDFCD)RETURN 3                                                  LIS00830
      IF(TYPE.EQ.TYPES)GO TO 20                                                 LIS00840
      SWITCH = .FALSE.                                                          LIS00850
      SWCHG = SWCHG + 1                                                         LIS00860
      IF(SWCHG.GT.1)GO TO 40                                                    LIS00870
      TYPE = TYPES                                                              LIS00880
       IPT = 0                                                                  LIS00890
C***** CHANGED JUNE 28 1978                                                     LIS00900
      RETURN 2                                                                  LIS00910
C                                                                               LIS00920
C                                                                               LIS00930
  100 ICNT = ICNT + 1                                                           LIS00940
      NOFLD2 = NOFLD2 + 1                                                       LIS00950
C                                                                               LIS00960
CCC   COMPUTE LINE INCREMENT                                                    LIS00970
C                                                                               LIS00980
      NN = NDOTS(ICNT)                                                          LIS00990
      NI =IABS( NN) / 100000000                                                 LIS01000
      LI = IABS(NN) - NI * 100000000                                            LIS01010
      IF(LI.GE.100000000)NI = NI +1                                             LIS01020
C                                                                               LIS01030
CCC   COMPUTE SAMPLE INCREMENT                                                  LIS01040
C                                                                               LIS01050
      KK=1                                                                      LIS01060
      IF(NN.LT.0)KK=-1                                                          LIS01070
      LI = NI * KK                                                              LIS01080
      N2 = NN - LI * 100000000                                                  LIS01090
      N3 = IABS( N2)/10000                                                      LIS01100
      SI = IABS(N2)-N3 * 10000                                                  LIS01110
      IF(SI.GE.1000)N3 = N3 + 1                                                 LIS01120
      KK=1                                                                      LIS01130
      IF(N2.LT.0)KK=-1                                                          LIS01140
      SI = N3 *KK                                                               LIS01150
      LACI = N2 - SI * 10000                                                    LIS01160
      LR = (LACI-1)/19                                                          LIS01170
      LR = (LR+1) * 10                                                          LIS01180
      LS = LR - 1                                                               LIS01190
      LS = LS /10                                                               LIS01200
      LS = 10 * (LACI - (LS*19))                                                LIS01210
      L = LR -LI                                                                LIS01220
      S = LS + SI                                                               LIS01230
C                                                                               LIS01240
CCC   STORE DOT INFO                                                            LIS01250
C                                                                               LIS01260
      FIELDS(1,NOFLD2) = CATNM                                                  LIS01270
      FIELDS(4,NOFLD2) = 2                                                      LIS01280
      FLDINF(1) = L                                                             LIS01290
      FLDINF(2) = L                                                             LIS01300
      FLDINF(3) = 1                                                             LIS01310
      FLDINF(4) = S                                                             LIS01320
      FLDINF(5) = S                                                             LIS01330
      FLDINF(6) = 1                                                             LIS01340
      IF(IPT.NE.0)GO TO 35                                                      LIS01350
      IPT = -3                                                                  LIS01360
   35 IPT = IPT + 4                                                             LIS01370
      VERTEX(IPT) = S                                                           LIS01380
      VERTEX(IPT+1)=L                                                           LIS01390
      VERTEX(IPT+2)=S                                                           LIS01400
      VERTEX(IPT+3)=L                                                           LIS01410
      RETURN 1                                                                  LIS01420
   40 WRITE(6,2000)                                                             LIS01430
 2000 FORMAT(//5X,'ERROR HAS OCCURRED IN READING LACIE FORMATTED DOT CAR LIS01440
     *DS - SUBROUTINE FLDLAC - EXIT TAKEN')                                     LIS01450
      RETURN 3                                                                  LIS01460
      END                                                                       LIS01470
```

FILE: MATVEC FORTRAN A

```
      SUBROUTINE MATVEC(A,M,C,L,M)
C MULTIPLY MATRIX A BY VECTOR B AND STORE IN VECTOR C
      DIMENSION A(L,M),B(M),C(L)
      DO 10 I=1,L
      SUM=0.
      DO 5 K=1,M
    5 SUM=SUM+A(I,K)*B(K)
   10 C(I)=SUM
      RETURN
      END
```

FILE: MTMDAT FORTRAN A

```
      SUBROUTINE MTMDAT(A,B,C,L,M,N,D,DD)
C
C     MULTIPLY MATRIX A BY THE TRANSPOSE OF B AND STORE IN DD
C     A LOWER TRIANGULAR MATRIX
C
      DIMENSION A(L,M),B(N,M),C(L,N),D(L),DD(1)
      DO 60 I=1,L
      DO 60 J=1,N
      SUM=0.0
      DO 55 K=1,M
   55 SUM=SUM+A(I,K)*B(J,K)
      C(I,J)=SUM
      IF(I.EQ.J) D(I)=SQRT(SUM)
   60 CONTINUE
      MM=0
      KK=0
      DO 1 II=1,L
      KK=KK+1
      DO 1 LL=1,KK
      MM=MM+1
      DD(MM)=C(II,LL)
    1 CONTINUE
      RETURN
      END
```

FILE: MTMLS6 FORTRAN A

```
      SUBROUTINE MTMLS6(A,B,C,M,N)                                      MTM00010
C                                                                       MTM00020
CC MULTIPLY MATRIX A BY B AND STORE IN C.  B IS STORED IN SYMMETRIC NOTATMTM00030
C                                                                       MTM00040
      DIMENSION A(M,N),B(1),C(M,N)                                      MTM00050
      DO 50 J=1,M                                                       MTM00060
      LB=0                                                              MTM00070
      DO 40 I=1,N                                                       MTM00080
      LB=LB+1                                                           MTM00090
      LE=LE+I                                                           MTM00100
      SUM=0.                                                            MTM00110
      K=0                                                               MTM00120
      DO 35 L=LB,LE                                                     MTM00130
      K=K+1                                                             MTM00140
   35 SUM=SUM+A(J,K)*B(L)                                               MTM00150
      IF(I.EQ.N) GO TO 40                                               MTM00160
      KS=K+1                                                            MTM00170
      L=LE                                                              MTM00180
      DO 36 K=KS,N                                                      MTM00190
      L=L+I+K-KS                                                        MTM00200
   36 SUM=SUM+A(J,K)*B(L)                                               MTM00210
   40 C(J,I)=SUM                                                        MTM00220
   50 CONTINUE                                                          MTM00230
      RETURN                                                            MTM00240
      END                                                               MTM00250
```

~~1963~~

FILE: NAMSTA FORTRAN A

```
C       NAMSTA ASSIGNS NAMES TO CLUSTERS AND UPDATES STAT INFO                  NAM00010
C                                                                               NAM00020
        SUBROUTINE NAMSTA(SUBNAM,CATVEC,SUBNO,NOSUB2,CATNAM,NOCAT)              NAM00030
C                                                                               NAM00040
        IMPLICIT INTEGER (A-Z)                                                  NAM00050
        DIMENSION SUBNAM(60),CATNAM(60)                                         NAM00060
        DIMENSION CATVEC(60),SUBNO(1)                                           NAM00070
        INTEGER*4 I4(3)                                                         NAM00080
        LOGICAL*1 L1(12)                                                        NAM00090
        EQUIVALENCE(L1(1),I4(1))                                                NAM00100
        K = 0                                                                   NAM00110
C                                                                               NAM00120
CCC     ASSIGN NAMES TO CLUSTERS                                                NAM00130
C                                                                               NAM00140
        DO 20 I=1,NOCAT                                                         NAM00150
        L = 0                                                                   NAM00160
        DO 20 J=1,NOSUB2                                                        NAM00170
        IF (CATVEC(J) .NE. I) GO TO 20                                          NAM00180
        K = K + 1                                                               NAM00190
        L = L + 1                                                               NAM00200
C                                                                               NAM00210
CCC     USE FIRST 2 CHAR OF CATEGORY NAME  + 2 DIGITS                           NAM00220
C                                                                               NAM00230
        I4(1) = CATNAM(I)                                                       NAM00240
        CALL RN[4A](I4(2),2,L)                                                  NAM00250
        L1(3)=L1(5)                                                             NAM00260
        L1(4)=L1(9)                                                             NAM00270
C                                                                               NAM00280
        SUBNAM(K) = I4(1)                                                       NAM00290
C                                                                               NAM00300
   20 CONTINUE                                                                  NAM00310
C                                                                               NAM00320
CCCCC                                                                           NAM00330
CCCCC                                                                           NAM00340
CCCCC   CHECK FOR NULL CATEGORY                                                 NAM00350
C                                                                               NAM00360
        I=NOCAT                                                                 NAM00370
   30   IF (SUBNO(I) .NE. 0) GO TO 60                                           NAM00380
        IF (I .EQ. NOCAT) GO TO 55                                              NAM00390
        DO 50 J=I,NOCAT                                                         NAM00400
        CATNAM(J) = CATNAM(J+1)                                                 NAM00410
   50   SUBNO(J) = SUBNO(J+1)                                                   NAM00420
   55   NOCAT = NOCAT - 1                                                       NAM00430
   60   I=I-1                                                                   NAM00440
        IF(I.GT.0) GOTO 30                                                      NAM00450
        RETURN                                                                  NAM00460
C                                                                               NAM00470
        END                                                                     NAM00480
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: NUMBER FORTRAN A

```
      FUNCTION NUMBER(CARD,COL,NUMVEC,NOW)
      IMPLICIT INTEGER (A-Z)
      DIMENSION CARD(1),NUMVEC(1)
      DATA CRDSIZ/62/,VECSIZ/100/,BLANK/' '/,COMMA/','/
      DATA ZERO/'0'/,NINE/'9'/
      NEXT = NOW + 1
      IF( NEXT .LE. 0 .OR. NEXT .GT. VECSIZ)  NEXT = 1
      J = 0
      L = COL+1
      IF( L .GT. CRDSIZ)  GO TO 92
      VK=VECSIZ
      DO  80  J=NEXT,VK,1
      JJ = J
      NUM = 0
      ITRIG=0
      DO  60  COL=L,CRDSIZ,1
      IF (CARD(COL).EQ.BLANK) GO TO 60
      IF (CARD(COL).EQ.COMMA) GO TO 70
      IF(CARD(COL).LT.ZERO.OR.CARD(COL).GT.NINE) GO TO 90
      CALL I4A1HN(CARD(COL),1,NWORD)
      NUM = 10 * NUM + NWORD
      ITRIG=1
   60 CONTINUE
      COL = CRDSIZ
      GO TO 90
   70 NUMVEC(J) = NUM
      L = COL+1
      IF( L .GT. CRDSIZ)  GO TO 92
   80 CONTINUE
      J = VECSIZ
   90 IF(ITRIG.EQ.1)GO TO 91
      J = J - 1
      GO TO 92
   91 NUMVEC(J) = NUM
   92 NUMBER = J
C     WRITE( 6,106) (CARD(K),K=1,62),COL,NUMBER,(NUMVEC(K),K=1,J)
C106  FORMAT(' NUMBER ENTERED'/' ',62A1,I10/' ' ,I5,18I3)
      RETURN
C*****
C*****   FUNCTION ENTRIES MUST RETURN VALUE IN ORIGINAL FUNCTION NAME
C*****
      END
```

FILE: NXTCHR FORTRAN A

```
      FUNCTION NXTCHR(CARD,COL)                                          NXT00010
C                                                                        NXT00020
C                                                                        NXT00030
C                                                                        NXT00040
C     CALL      J=NXTCHR(CARD,COL)                                       NXT00050
C                                                                        NXT00060
C     ARGS      CARD  - BCD BUFFER                                       NXT00070
C               COL   - PTR TO POSITION IN 'CARD'                        NXT00080
C                                                                        NXT00090
C     PURPOSE   LOCATES THE NEXT NON BLANK SYMBOL IN 'CARD'              NXT00100
C                                                                        NXT00110
C     RETURNS   J   - LOCATED CHARACTER (BLANK IF EOC)                   NXT00120
C               COL   - PTS AT CHARACTER                                 NXT00130
C                                                                        NXT00140
      IMPLICIT INTEGER (A-Z)                                             NXT00150
      DIMENSION CARD(1)                                                  NXT00160
      DATA CRDSIZ/62/,BLANK/' '/                                         NXT00170
      L = COL+1                                                          NXT00180
      IF (L.GT.CRDSIZ) GO TO 40                                          NXT00190
      DO 30 COL=L,CRDSIZ                                                 NXT00200
      NXTCHR = CARD(COL)                                                 NXT00210
      IF(NXTCHR.NE.BLANK) GO TO 50                                       NXT00220
   30 CONTINUE                                                           NXT00230
      COL=CRDSIZ-1                                                       NXT00240
   40 NXTCHR = BLANK                                                     NXT00250
   50 CONTINUE                                                           NXT00260
C     WRITE (6,104) (CARD(K),K=1,62),COL,NXTCHR                          NXT00270
  104 FORMAT(' NXTCHR ENTERED'/' ',62A1,I10/' ',A4)                      NXT00280
      RETURN                                                             NXT00290
      END                                                                NXT00300
```

FILE: ORDER FORTRAN A

```
      SUBROUTINE ORDER(VEC,N)
      IMPLICIT INTEGER(A-Z)
      DIMENSION VEC(1)
      LOGICAL SWITCH
      IF(N.LE.1)RETURN
      M=N-1
    5 SWITCH=.FALSE.
      DO 10 I=1,M
      IF(VEC(I).LE.VEC(I+1))GO TO 10
      TEMP=VEC(I)
      VEC(I)=VEC(I+1)
      VEC(I+1)=TEMP
      SWITCH=.TRUE.
   10 CONTINUE
      IF(SWITCH)GO TO 5
      RETURN
      END
```

FILE: PRTCOV FORTRAN A

```
      SUBROUTINE PRTCOV(COVMTX,AVEMTX,CV1,AV1,CLSMTX)                    PRT00010
C                                                                        PRT00020
CC WRITE  HEADING FOR TRANSFORMED COVARIANCE MATRIX                      PRT00030
C                                                                        PRT00040
      IMPLICIT INTEGER(A-Z)                                              PRT00050
      REAL COVMTX(CV1,NOSUB2),AVEMTX(AV1,NOSUB2)                         PRT00060
C                                                                        PRT00070
CC    INCLUDE COMRK1.LIST                                                PRT00080
CC    INCLUDE COMRK4.LIST                                                PRT00090
CC    INCLUDE COMRK6.LIST                                                PRT00100
C     INCLUDE COMRK9.LIST                                                PRT00110
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         COM00030
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              COM00040
     *              GRPCHK(61),GROUPS(124)                               COM00050
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                     COM00010
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                  COM00020
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                COM00030
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,AMFILE,AMKEY,          COM00010
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,    COM00020
     *    ORUMAD,ORMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 COM00030
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    COM00040
     *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              COM00050
     *   CRDUNT,PRTUNT,RANDIO                                            COM00060
C DATA TRANSFORMATION COMMON BLOCK
      COMMON/TRBLCK/OUTFMT,NOFEAT,FLDINF(6),          FETVEC(30)
C$END                                                                    PRT00130
C                                                                        PRT00140
      DIMENSION CLSMTX(NOSUB2)                                           PRT00150
C                                                                        PRT00160
      DATA BCDTWO/'2'/                                                   PRT00170
C                                                                        PRT00180
      CNT=7+(5+3+2*AV1)*((AV1+11)/12)                                    PRT00190
      CNT=PAGSIZ/CNT                                                     PRT00200
      INC=CNT                                                            PRT00210
      DO 1  ICLAS=1,NOSUB2                                               PRT00220
      IF(INC.LT.CNT) GO TO 2                                             PRT00230
      WRITE(6,HEAD)                                                      PRT00240
      INC=0                                                              PRT00250
    2 WRITE(6,3) CLSMTX(ICLAS)                                           PRT00260
    3 FORMAT(/' SUBCLASS ',A4)                                           PRT00270
      DO 4 LOC=1,AV1,12                                                  PRT00280
      STOP=LOC+11                                                        PRT00290
      IF(STOP.GT.AV1) STOP=AV1                                           PRT00300
    4 WRITE(6,5) (AVEMTX(I,ICLAS),I=LOC,STOP)                            PRT00310
    5 FORMAT(' MEAN',3X,12F9.2)                                          PRT00320
      WRITE(6,6)                                                         PRT00330
    6 FORMAT(' COVARIANCE MATRIX')                                       PRT00340
      CALL WRTMTX(COVMTX(1,ICLAS),AV1,BCDTWO)                            PRT00350
      INC=INC+1                                                          PRT00360
    1 CONTINUE                                                           PRT00370
      RETURN                                                             PRT00380
      END                                                                PRT00390
```

FILE: RDDOTS FORTRAN A

```
C                                                                       RDD00010
C     UTILITY ROUTINE THAT READS THE DOTFIL                             RDD00020
C                                                                       RDD00030
      SUBROUTINE RDDOTS(DOTS,DOTVEC,TOTDT3,TYPSWT,SIZES,TOTDT2,         RDD00040
     *    NOCAT,CATNAM,NOFET2,FETVC2,NOFEAT,FETVEC,NOSUN,ANGLE,         RDD00050
     * NOFLD,TOTVRT,FLDSAV,VERTEX,/KVAR/)                               RDD00060
      IMPLICIT INTEGER(A-Z)                                             RDD00070
      DIMENSION KVAR(SIZES,1)                                           RDD00080
      REAL KVAR                                                         RDD00090
C                                                                       RDD00100
C     TYPSWT = 1 -RETURNS SPECTRAL INFO                                 RDD00110
C     TYPSWT = 2 -RETURNS SPATIAL INFO                                  RDD00120
C     TYPSWT = 3 -RETURNS BOTH SPATIAL AND SPECTRAL INFO                RDD00130
C                                                                       RDD00140
C                                                                       RDD00150
      DATA BLANK/' '/                                                   RDD00160
C     INCLUDE COMRK6,LIST                                               RDD00170
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         RDD00180
     *              HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,   RDD00190
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                RDD00200
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    RDD00210
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              RDD00220
     *  CRDUNT,PRTUNT,RANDIO                                            RDD00230
C$END                                                                   RDD00240
C                                                                       RDD00250
      DIMENSION CATNAM(1),DOTVEC(1),DOTS(SIZES,1)                       RDD00260
      DIMENSION FETVEC(30),FETVC3(30),FETVC2(1)                         RDD00270
      DIMENSION FLDSAV(4,1),VERTEX(2,1),ANGLE(1)                        RDD00280
      DIMENSION TEMDOT(5000)                                            RDD00290
C                                                                       RDD00300
CC    READ REC NO. 1 FOR INDICES                                        RDD00310
C                                                                       RDD00320
      REWIND DOTUNT                                                     RDD00330
      CALL FSBSFL(DOTUNT,DOTFIL,ISTAT)                                  RDD00340
C                                                                       RDD00350
      READ(DOTUNT)NOCAT,NOFEAT,NOFLD,TOTVRT,TOTDOT,NOSUN,(CATNAM(I),    RDD00360
     *            I=1,NOCAT),SIZE                                       RDD00370
C                                                                       RDD00380
CC    COMPUTE ADDRESSES FOR ARRAY                                       RDD00390
C                                                                       RDD00400
      DOTS1  = 1                                                        RDD00410
C                                                                       RDD00420
C                                                                       RDD00430
CC    READ REC. NO. 2                                                   RDD00440
C                                                                       RDD00450
      IF (TYPSWT .EQ. 1) READ(DOTUNT)(FETVEC(I),I=1,NOFEAT)             RDD00460
      IF (TYPSWT .EQ. 2) READ(DOTUNT) DUMMY                             RDD00470
      IF (TYPSWT .EQ. 3) READ(DOTUNT)(FETVEC(I),I=1,NOFEAT)             RDD00480
     *    ,(((FLDSAV(I,J),I=1,4),J=1,NOFLD),((VERTEX(I,J),I=1,2),       RDD00490
     *    J=1,TOTVRT),(ANGLE(I),I=1,NOSUN)                              RDD00500
C                                                                       RDD00510
      IF (TYPSWT .EQ. 2) GO TO 87                                       RDD00520
      IF (NOFET2 .NE. 0) GO TO 60                                       RDD00530
C                                                                       RDD00540
CC    SET DEFAULT CHANNELS                                              RDD00550
C                                                                       RDD00560
      DO 50 I=1,NOFEAT                                                  RDD00570
      FETVC2(I) = I                                                     RDD00580
   50 FETVC3(I) = I                                                     RDD00590
      NOFET2 = NOFEAT                                                   RDD00600
      GO TO 87                                                          RDD00610
   60 DO 80 J=1,NOFET2                                                  RDD00620
      DO 70 K=1,NOFEAT                                                  RDD00630
      IF( FETVEC(K) .EQ. FETVC2(J)) GO TO 75                            RDD00640
   70 CONTINUE                                                          RDD00650
      WRITE(6,85)FETVC2(J),(FETVEC(I),I=1,NOFEAT)                       RDD00660
   85 FORMAT(//' CHANNEL ',I2,' IS NOT ON DOTFIL'/ ' CHANNELS ARE',30I3)RDD00670
      CALL CMERR                                                        RDD00680
   75 FETVC3(J) = K                                                     RDD00690
C                                                                       RDD00700
   80 CONTINUE                                                          RDD00710
   87 IF (TOTDT3 .EQ. 0) GO TO 96                                       RDD00720
C                                                                       RDD00730
C***                                                                    RDD00740
C***   CODE ADDED NOV 21, 1978 FOR LIST PROCESSING                      RDD00750
C                                                                       RDD00760
        IF (TOTDT3.GT.TOTDOT) TOTDT3 = TOTDOT                           RDD00770
C                                                                       RDD00780
      DO 95  J=1,TOTDT3                                                 RDD00790
```

FILE: RDDOTS FORTRAN A

```
      IF (DOTVEC(J).LE.TOTDOT) GO TO 95                                 RDD00800
      WRITE(6,90)DOTVEC(J),TOTDOT                                       RDD00810
   90 FORMAT(/ ' DOT NO. ',I3,' IS NOT ON DOTFIL'/ ' FILE CONTAINS ',I3,RDD00820
     *       ' DOTS' )                                                  RDD00830
      CALL CMERR                                                        RDD00840
   95 CONTINUE                                                          RDD00850
   96 CONTINUE                                                          RDD00860
C                                                                       RDD00870
CC                                                                      RDD00880
CC    READ REC  NO.3 -- DOTS                                            RDD00890
CC                                                                      RDD00900
      TOTAL = SIZE * TOTDOT                                             RDD00910
      READ(DOTUNT)(TEMDOT(DOTS1-1+I),I=1,TOTAL)                         RDD00920
C                                                                       RDD00930
      IF (TYPSWT .EQ. 1 ) GO TO 130                                     RDD00940
      IF (TYPSWT .EQ. 2 ) GO TO 150                                     RDD00950
      IF (TYPSWT .EQ. 3 ) GO TO 180                                     RDD00960
C                                                                       RDD00970
CC    RETRIEVE SPECTRAL INFO                                            RDD00980
C                                                                       RDD00990
  130 CONTINUE                                                          RDD01000
      SIZES = NOFET2                                                    RDD01010
      TOTDT2 = TOTDT3                                                   RDD01020
      CALL RDDOT1(TEMDOT,DOTS,KVAR,SIZES,TOTDT2,DOTVEC,FETVC3,          RDD01030
     * SIZE,TOTDOT,TOTDT3,NOFET2,TYPSWT)                                RDD01040
      RETURN                                                            RDD01050
C                                                                       RDD01060
CC    RETRIEVE SPATIAL INFO                                             RDD01070
C                                                                       RDD01080
  150 CONTINUE                                                          RDD01090
      SIZES = 4                                                         RDD01100
      TOTDT2 = TOTDOT                                                   RDD01110
C                                                                       RDD01120
      CALL RDDOT1(TEMDOT,DOTS,KVAR,SIZES,TOTDT2,DOTVEC,FETVC3,          RDD01130
     * SIZE,TOTDOT,TOTDT3,NOFET2,TYPSWT)                                RDD01140
      RETURN                                                            RDD01150
C                                                                       RDD01160
  180 CONTINUE                                                          RDD01170
C                                                                       RDD01180
CC    RETRIEVE SPECTRAL AND SPATIAL INFO                                RDD01190
CC                                                                      RDD01200
      SIZES = SIZE                                                      RDD01210
      TOTDT2 = TOTDOT - TOTDT3                                          RDD01220
      CALL RDDOT1(TEMDOT,DOTS,KVAR,SIZES,TOTDT2,DOTVEC,FETVC3,          RDD01230
     * SIZE,TOTDOT,TOTDT3,NOFET2,TYPSWT)                                RDD01240
      RETURN                                                            RDD01250
C                                                                       RDD01260
      END                                                               RDD01270
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: RDDOT1 FORTRAN A

```
      SUBROUTINE RDDOT1(TEMDOT,DOTS,KVAR,SIZES,TOTDT2,DOTVEC,FETVC3,    RDD00010
     *                  SIZE,TOTDOT,TOTDT3,NOFET2,TYPSWT)               RDD00020
C                                                                       RDD00030
      IMPLICIT INTEGER (A-Z)                                            RDD00040
      DIMENSION TEMDOT(1),DOTS(SIZES,1),DOTVEC(1),FETVC3(1)             RDD00050
      REAL KVAR(SIZES,1)                                                RDD00060
C                                                                       RDD00070
CCCCC TYPSWT = 1 -- RETRIEVE SPECTRAL INFO                              RDD00080
CCCCC        = 2 -- RETRIEVE SPATIAL INFO                               RDD00090
CCCCC        = 3 -- RETRIEVE SPECTRAL AND SPATIAL INFO                  RDD00100
C                                                                       RDD00110
      GO TO (130,150,180),TYPSWT                                        RDD00120
C                                                                       RDD00130
CCC   RETRIEVE SPECTRAL INFO                                            RDD00140
C                                                                       RDD00150
  130 CONTINUE                                                          RDD00160
C                                                                       RDD00170
CCC   PICK SUBSET OF DOTS AND CHANNELS                                  RDD00180
C                                                                       RDD00190
      DO 140  K=1,TOTDT2                                                RDD00200
      KK = DOTVEC(K)                                                    RDD00210
      DO 140  J=1,SIZES                                                 RDD00220
      JJ = FETVC3(J)                                                    RDD00230
      JJJ =(KK-1)*SIZE + JJ+4                                           RDD00240
      KVAR(J,K) = FLOAT(TEMDOT(JJJ))                                    RDD00250
  140 CONTINUE                                                          RDD00260
      RETURN                                                            RDD00270
C                                                                       RDD00280
CCC   RETRIEVE SPATIAL INFO                                             RDD00290
C                                                                       RDD00300
  150 CONTINUE                                                          RDD00310
      DO 170  I=1,TOTDT2                                                RDD00320
      KK =(I-1)*SIZE                                                    RDD00330
      DO 170  K=1,SIZES                                                 RDD00340
  170 DOTS(K,I) = TEMDOT(KK+K)                                          RDD00350
C                                                                       RDD00360
      RETURN                                                            RDD00370
C                                                                       RDD00380
  180 CONTINUE                                                          RDD00390
C                                                                       RDD00400
CCC   RETRIEVE SPECTRAL AND SPATIAL INFO                                RDD00410
C                                                                       RDD00420
      JJ = 0                                                            RDD00430
      KK = 1                                                            RDD00440
      DO 300 K=1,TOTDOT                                                 RDD00450
      IF (KK.GT. TOTDT3) GO TO 190                                      RDD00460
      IF (K  .NE. DOTVEC(KK)) GO TO 190                                 RDD00470
      KK = KK+1                                                         RDD00480
      GO TO 300                                                         RDD00490
  190 JJ = JJ + 1                                                       RDD00500
      JJJ = (K-1)*SIZE                                                  RDD00510
      DO 200 J=1,4                                                      RDD00520
  200 DOTS(J,JJ) = TEMDOT(JJJ+J)                                        RDD00530
C                                                                       RDD00540
      DO 210 J=1,NOFET2                                                 RDD00550
      I = FETVC3(J)                                                     RDD00560
  210 DOTS(4+J,JJ) = TEMDOT(JJJ+4+I)                                    RDD00570
  300 CONTINUE                                                          RDD00580
      RETURN                                                            RDD00590
C                                                                       RDD00600
      END                                                               RDD00610
```

FILE: RDMODK FORTRAN A

```
CCOOPPYY                                                                RDM00010
      SUBROUTINE RDMODK(AVAR,COVAR,CLSDES,SUBNO,SUBDES,FLDSAV,VERTEX,   RDM00020
     *      ARRAY)                                                      RDM00030
      IMPLICIT INTEGER (A-Z)                                            RDM00040
C                                                                       RDM00050
CCCCCC                                                                  RDM00060
CCCCCC* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * RDM00070
CCCCCC                                                                  RDM00080
CCCCCC  READ IN REST OF MODULE DECK AND STORE IN THE                    RDM00090
CCCCCC  FOLLOWING MANNER :  COVARIANCES                                 RDM00100
CCCCCC                      MEANS                                       RDM00110
CCCCCC                      CLASS DESCRIPTIONS                          RDM00120
CCCCCC                      NO OF SUBCLASS IN EACH CLASS                RDM00130
CCCCCC                      SUBCLASS DESCRIPTIONS                       RDM00140
CCCCCC                      FIELD INFORMATION                           RDM00150
CCCCCC                      VERTICIES                                   RDM00160
CCCCCC  AND WRITE SAVTAP FILE                                           RDM00170
CCCCCC                                                                  RDM00180
CCCCCC* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * RDM00190
CCC   INCLUDE COMRK1.LIST                                               RDM00200
C     INCLUDE COMRK6.LIST                                               RDM00210
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,          RDM00220
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,    RDM00230
     *             FETVC2(30),SURVC2(75),SUBPTR(75),CLSVC2(60),         RDM00240
     *             KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              RDM00250
     *             GRPCHK(61),GROUPS(124)                               RDM00260
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         RDM00270
     *             HISFIL,HISKEY,TRFORM,ERIPTP,ERRKEY,MAPUNT,NOFILE,    RDM00280
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                RDM00290
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    RDM00300
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              RDM00310
     *  CRDUNT,PRTUNT,RANDIO                                            RDM00320
C$END                                                                   RDM00330
      COMMON/PASSR/NOCLS,NOSUB,NOFEAT,NOFLD,TOTVRT,FETVEC(30),          RDM00340
     *             FLDSV1,CLSID1,VARSIZ                                 RDM00350
      DIMENSION COVAR(VARSIZ),AVAR(NOFEAT),CLSDES(NOCLS),SUBNO(NOCLS),  RDM00360
     *          SUBDES(NOSUB),FLDSAV(4,NOFLD),VERTEX(2,TOTVRT),         RDM00370
     *       ARRAY(1)                                                   RDM00380
      REAL COVAR,AVAR                                                   RDM00390
C                                                                       RDM00400
      REWIND SAVTAP                                                     RDM00410
C                                                                       RDM00420
      IF (STAFIL .EQ. 0)  GO TO 1                                       RDM00430
C                                                                       RDM00440
CCC   POSITION STAT TAPE TO DESIRED FILE                                RDM00450
C                                                                       RDM00460
      CALL FSBSFL(SAVTAP,STAFIL,ISTAT)                                  RDM00470
      IF (ISTAT .EQ. 0)  GO TO 1                                        RDM00480
      FILNO = STAFIL + 1                                                RDM00490
      WRITE(6,290)FILNO                                                 RDM00500
  290 FORMAT(// 15H*ERROR IN TRYING TO POSITION STAT FILE TO FILE ',I3, RDM00510
     *  ' IN CRDSTA')                                                  RDM00520
C                                                                       RDM00530
    1 IF (ISTAT .GT. 0)  GO TO 3                                        RDM00540
      WRITE(SAVTAP)NOCLS,NOSUB,NOFEAT,NOFLD,TOTVRT,(FETVEC(I),I=1,      RDM00550
     *             NOFEAT)                                              RDM00560
C                                                                       RDM00570
CCC   IF ISTAT IS NOT ZERO, THE FILE CANNOT BE WRITTEN, BUT THE MODULE  RDM00580
C*    STAT DECK WILL BE READ TO CHECK FOR INPUT ERRORS.                 RDM00590
C                                                                       RDM00600
    3 CONTINUE                                                          RDM00610
      K2 = 0                                                            RDM00620
      DO 5     J=1,NOFLD                                                RDM00630
       READ(CRDUNT,215)(FLDSAV(I,J),I=1,4)                              RDM00640
      K1 = 1 + K2                                                       RDM00650
      K2 = K1 + FLDSAV(4,J) - 1                                         RDM00660
      READ(CRDUNT,220) ((VERTEX(I,K),I=1,2),K=K1,K2)                    RDM00670
C                                                                       RDM00680
      IF (ISTAT .GT. 0)  GO TO 5                                        RDM00690
C                                                                       RDM00700
      WRITE(SAVTAP) (FLDSAV(I,J),I=1,4)                                 RDM00710
      WRITE(SAVTAP) ((VERTEX(I,K),I=1,2),K=K1,K2)                       RDM00720
    5 CONTINUE                                                          RDM00730
      READ(CRDUNT,230) (CLSDES(I),I=1,NOCLS)                            RDM00740
      READ(CRDUNT,240) (SUBNO(I),I=1,NOCLS)                             RDM00750
      READ(CRDUNT,250) (SUBDES(I),I=1,NOSUB)                            RDM00760
C                                                                       RDM00770
      IF (ISTAT .GT. 0)   GO TO 7                                       RDM00780
C                                                                       RDM00790
```

FILE: RDMODK FORTRAN A

```
      SUB02 = FLDSV1 - 1                                                RDM00800
      WRITE(SAVTAP)(ARRAY(I),I=CLSID1,SUB02)                            RDM00810
    7 CONTINUE                                                          RDM00820
C                                                                       RDM00830
       MEAN2 = CLSID1 - 1                                               RDM00840
      DO 10 I=1,NOSUB                                                   RDM00850
       READ(CRDUNT,240) KEPPTS(I)                                       RDM00860
       READ(CRDUNT,280) (AVAR(J),J=1,NOFEAT)                            RDM00870
       READ(CRDUNT,290)(COVAR(J),J=1,VARSIZ)                            RDM00880
C                                                                       RDM00890
      IF (ISTAT .GT. 0)  GO TO 10                                       RDM00900
CI                                                                      RDM00910
      WRITE(SAVTAP)KEPPTS(I),(ARRAY(J),J=1,MEAN2)                       RDM00920
   10 CONTINUE                                                          RDM00930
CI                                                                      RDM00940
      IF (ISTAT .GT. 0)  STAFIL = -1                                    RDM00950
      IF (ISTAT .GT. 0)  RETURN                                         RDM00960
CI                                                                     CIRDM00970
      END FILE SAVTAP                                                   RDM00980
      RETURN                                                            RDM00990
C                                                                       RDM01000
  215 FORMAT(A4,6X,I2,8X,I2,8X,I2)                                      RDM01010
  220 FORMAT (10X,14I5)                                                 RDM01020
  230 FORMAT((8X,9(2X,A4,2X)))                                          RDM01030
  240 FORMAT((7X,24(1X,I2)))                                            RDM01040
  250 FORMAT((8X,10(A4,3X)))                                            RDM01050
  260 FORMAT(12X,I8)                                                    RDM01060
  270 FORMAT (6X,12F6.2)                                                RDM01070
  280 FORMAT (5X,5E15.8)                                                RDM01080
      END                                                               RDM01090
```

FILE: REDDAT FORTRAN A

```
      SUBROUTINE REDDAT(COVAR,AVAR,CLSDES,SURNO,SUBDES,FLDSAV,VERTEX,    REDD00010
     *                  COV,AVEN,CLSDS,SURNOS,SUBDS,FLDSV,VERTX,         REDD00020
     *                  NOFEAT,VARSIZ,NOCLS,NOFLD,NOSUB,FETVEC)          REDD00030
      IMPLICIT INTEGER (A-Z)                                             REDD00040
      DIMENSION FETVEC(30)                                               REDD00050
C                                                                        REDD00060
CCCCCC* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * REDD00070
CCCCCC                                                                   REDD00080
CCCCCC READS COVARIANCES AND MEANS FROM FILE AND REDUCES STATS           REDD00090
CCCCCC* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * REDD00100
CCCCCC                                                                   REDD00110
CCCCCC                                                                   REDD00120
CCCCCC                                                                   REDD00130
CCCCCC                                                                   REDD00140
C     INCLUDE COMBK1.LIST                                                REDD00150
C     INCLUDE COMBK5.LIST                                                REDD00160
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,           REDD00170
     *              AVAR2,COVAR2,CLSID2,SURNO2,SUBDS2,FLDSV2,VERTX2,     REDD00180
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         REDD00190
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              REDD00200
     *              GRPCHK(61),GROUPS(124)                               REDD00210
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          REDD00220
     *              HISFIL,HISKEY,TPFORM,EPIPTP,ERPKEY,MAPUNT,NOFILE,    REDD00230
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                 REDD00240
     *  ,NHSTUN,NHSIFI,SCTRUN,MAPFIL                                     REDD00250
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,               REDD00260
     *  CRDUNT,PRTUNT,RANDIO                                             REDD00270
C$END                                                                    REDD00280
      COMMON/RESTKN/KEPPTS(60),IPRIOR,KREST,NCPASS                       REDD00290
      REAL COVAR(VARSZ2,NOSUB2),AVAR(NOFEAT,NOSUB),COV(VARSIZ),          REDD00300
     *          AVEN(NOFET2,NOSUB2),R                                    REDD00310
      DIMENSION CLSDES(NOCLS),SURNO(NOCLS),SUBDES(NOSUB),CLSDS(NOCLS2),  REDD00320
     *          SURNOS(NOCLS2),SUBDS(NOSUB2),FLDSV(4,NOFLD2),            REDD00330
     *          VERTX(2,TOTVT2),FLDSAV(4,NOFLD),VERTEX(2,TOTVT2)         REDD00340
     *       ,DUMVEC(30)                                                 REDD00350
C                                                                        REDD00360
CC    REDUCE CLASS DESCRIPTION AND ARRAY CONTAINING NO OF SUBCLASSES     REDD00370
C                                                                        REDD00380
      DO 150  I=1,NOCLS2                                                 REDD00390
      CLSDS(I) = CLSDES(I)                                               REDD00400
  150 SURNOS(I) = SURNO(I)                                               REDD00410
C                                                                        REDD00420
CC    REDUCE SUBCLASS DESCRIPTIONS                                       REDD00430
C                                                                        REDD00440
      DO 160  I=1,NOSUB2                                                 REDD00450
  160 SUBDS(I) = SUBDES(I)                                               REDD00460
C                                                                        REDD00470
CCC   REDUCE FIELD INFORMATION                                           REDD00480
C                                                                        REDD00490
      DO 170  I=1,NOFLD2                                                 REDD00500
       DO 170 J=1,4                                                      REDD00510
  170  FLDSV(J,I) = FLDSAV(J,I)                                           REDD00520
C                                                                        REDD00530
C     REDUCE VERTICES                                                    REDD00540
      DO 180 I=1,TOTVT2                                                  REDD00550
       DO 180 J=1,2                                                      REDD00560
  180  VERTX(J,I) = VERTEX(J,I)                                           REDD00570
C                                                                        REDD00580
CC    ZERO OUT JUST PORTION OF COVAR THAT WILL CONTAIN SUBCLASSES        REDD00590
CC    THAT HAVE BEEN GROUPED                                             REDD00600
C                                                                        REDD00610
      DO 200  J=1,NOGRP                                                  REDD00620
      KA = GRPDEX(J) + 1                                                 REDD00630
      KE = KA + GROUPS(KA-1) - 1                                         REDD00640
      IF (KA .GE. KE)  GO TO 200                                         REDD00650
      KK = SUBPTR(KA)                                                    REDD00660
      DO 195  LL = 1,VARSZ2                                              REDD00670
  195  COVAR(LL,KK) = 0.0                                                REDD00680
  200  CONTINUE                                                          REDD00690
C                                                                        REDD00700
CC    CHECK CLASSIFICATION CHANNELS AGAINST TRAINING CHANNELS            REDD00710
C                                                                        REDD00720
      DO 220    J=1,NOFET2                                               REDD00730
      DO 210    L=1,NOFEAT                                               REDD00740
      IF ( FETVC2(J) .EQ. FETVEC(L)) GO TO 220                           REDD00750
  210 CONTINUE                                                           REDD00760
      WRITE(4,230) FETVC2(J),(FETVEC(K),K=1,NOFEAT)                      REDD00770
  230 FORMAT(1  ** CHANNEL NO. ',I2,' IS NOT A TRAINING CHANNEL **'/     REDD00780
     *    40X,'TRAINING CHANNELS ARE - '/ 10X,30(I2,1X))                 REDD00790
```

FILE: REDDAT FORTRAN A

```
      CALL EXIT                                                          REED00800
  220 DUMVEC(J) = L                                                      REED00810
      DO 100  JJ=1,NOSUB                                                 REED00820
C     IS THIS SUBCLASS A MEMBER OF SELECTED SUBCLASSES                   REED00830
      IF (SUBPTR(JJ) .LE. 0) READ(SAVTAP) DUMMY                          REED00840
      IF (SUBPTR(JJ) .LE. 0) GO TO 100                                   REED00850
      READ(SAVTAP)KEPPTS(JJ),COV,(AVAR(I,JJ),I=1,NOFEAT)                 REED00860
C                                                                        REED00870
CC    REDUCE BY CHANNELS                                                 REED00880
C                                                                        REED00890
      NEWSUB = SUBPTR(JJ)                                                REED00900
      KK= 0                                                              REED00910
      DO 20  J=1,NOFET2                                                  REED00920
      K = DUMVEC(J)                                                      REED00930
      LOC = K*(K-1)/2                                                    REED00940
       DO 40 L=1,J                                                       REED00950
       KK = KK + 1                                                       REED00960
      WAT = LOC + DUMVEC(L)                                              REED00970
   40  COV(KK) = COV(WAT)                                                REED00980
   20 AVAR(J,JJ)    =   AVAR(K,JJ)                                       REED00990
C                                                                        REED01000
CC    GROUP SUBCLASSES                                                   REED01010
C                                                                        REED01020
      IF (GRPCHK(JJ) .LE. 0) GO TO 60                                    REED01030
       KK = 0                                                            REED01040
      DO 50  J1=1,NOFET2                                                 REED01050
       DO 50 J2=1,J1                                                     REED01060
       KK = KK + 1                                                       REED01070
       R  = COVAR(KK,NEWSUB)                                             REED01080
       COVAR(KK,NEWSUB) = R + (COV(KK)*(KEPPTS(JJ)-1)) +                 REED01090
     *                        (AVAR(J1,JJ)*AVAR(J2,JJ)*KEPPTS(JJ))      REED01100
   50 CONTINUE                                                           REED01110
      GO TO 100                                                          REED01120
   60 DO 70 I=1,VARSZ2                                                   REED01130
   70 COVAR(I,NEWSUB) = COV(I)                                           REED01140
  100 CONTINUE                                                           REED01150
C                                                                        REED01160
CC    GROUP MEANS                                                        REED01170
C                                                                        REED01180
      DO 125   I=1,NOGRP                                                 REED01190
      KB = GRPDEA(I) + 1                                                 REED01200
      KE = KB + GROUPS(KB-1) - 1                                         REED01210
      KB1 = GROUPS(KB)                                                   REED01220
      IF (KB .GE. KE) GO TO 130                                          REED01230
       DO 120  J=1,NOFET2                                                REED01240
       R = 0.0                                                           REED01250
       KPTS = 0                                                          REED01260
        DO 110  K=KB,KE                                                  REED01270
        KB2 = GROUPS(K)                                                  REED01280
        KPTS = KPTS + KEPPTS(KB2)                                        REED01290
  110   R = R + AVAR(J,KB2) * KEPPTS(KB2)                                REED01300
  120 AVAR(J,KB1) = R / KPTS                                             REED01310
      KEPPTS(KB1)=KPTS                                                   REED01320
      JJ = 0                                                             REED01330
      NEWSUB = SUBPTR(KB1)                                               REED01340
      DO 123   J1=1,NOFET2                                               REED01350
      DO 123   J2=1,J1                                                   REED01360
      JJ = JJ + 1                                                        REED01370
  123 COVAR(JJ,NEWSUB) = (COVAR(JJ,NEWSUB) - (AVAR(J1,KB1)*              REED01380
     *                   AVAR(J2,KB1))*KPTS) / (KPTS-1)                  REED01390
  125 CONTINUE                                                           REED01400
C     REDUCE MEANS                                                       REED01410
  130  DO 140   K=1,NOSUB2                                               REED01420
       II = SUBVC2(K)                                                    REED01430
       KEPPTS(K)=KEPPTS(II)                                              REED01440
        DO 140 J=1,NOFET2                                                REED01450
  140    AVEN(J,K) = AVAR(J,II)                                          REED01460
C                                                                        REED01470
C                                                                        REED01480
       RETURN                                                            REED01490
C                                                                        REED01500
      END                                                                REED01510
```

FILE: REDSAV FORTRAN A

```
      SUBROUTINE REDSAV(ARRAY,TOP,BMFLG)                                        RED00010
C                                                                               RED00020
CC                                                                              RED00030
      IMPLICIT INTEGER (A-Z)                                                    RED00040
      DIMENSION ARRAY(1)                                                        RED00050
C                                                                               RED00060
CCC   INCLUDE COMRK1.LIST                                                       RED00070
CC    INCLUDE COMRKA.LIST                                                       RED00080
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARS72,TOTVT2,NOFLD2,                  RED00090
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,            RED00100
     *              FETVC2(30),SUMVC2(75),SUMPTR(75),CLSVC2(60),                RED00110
     *              KEPPTS(40),NOGRP,GRPNAM(60),GRPDEX(61),                     RED00120
     *              GRPCHK(61),GROUPS(124)                                      RED00130
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,                 RED00140
     *              HISFIL,HISKEY,TRFORM,ERPTP,ERPKEY,MAPUNT,NOFILE,            RED00150
     *    DRUMAD,DRMSIZ,MAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                        RED00160
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                            RED00170
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTHFL,HISTFL,PCHUNT,                      RED00180
     *  CRDUNT,PRTUNT,RANDIO                                                    RED00190
CSEND                                                                           RED00200
      DIMENSION FETVEC(30)                                                      RED00210
      REWIND SAVTAP                                                             RED00220
      IF (STAFIL .EQ. -1)  WRITE(6,100)                                         RED00230
  100 FORMAT(// TR,'STAT FILE WAS NOT CREATED. EXITING FROM **REDSAV***')       RED00240
      IF (STAFIL .EQ. -1)  CALL CMERR                                           RED00250
      IF (STAFIL .EQ.  0)  GO TO 10                                             RED00260
      CALL FSRSFL(SAVTAP,STAFIL,ISTAT)                                          RED00270
      IF (ISTAT .EQ. 0)  GO TO 10                                               RED00280
      FILNO = STAFIL + 1                                                        RED00290
      WRITE(6,110) FILNO                                                        RED00300
  110 FORMAT(// TR,'ERROR IN POSITIONING STAT FILE TO FILE ',I3,                RED00310
     *   ' EXITING FROM REDSAV')                                                RED00320
   10 CONTINUE                                                                  RED00330
      READ(SAVTAP)NOCLS,NOSUB,NOFEAT,NOFLD,TOTVRT,(FETVEC(I),I=1,NOFEAT)       RED00340
C                                                                               RED00350
CC    COMPUTE BASES                                                             RED00360
C                                                                               RED00370
      VARSIZ = NOFEAT*(NOFEAT+1)/2                                              RED00380
      CLSID1 = 1                                                                RED00390
      SUBNO1 = CLSID1 + NOCLS                                                   RED00400
      SUBDS1 = SUBNO1 + NOCLS                                                   RED00410
      FLDSV1 = SUBDS1 + NOSUB                                                   RED00420
      VERTX1 = FLDSV1 + NOFLD*4                                                 RED00430
C                                                                               RED00440
      CALL SAVFIL(ARRAY(FLDSV1),ARRAY(VERTX1),ARRAY(CLSID1),                    RED00450
     *            ARRAY(SUBNO1),ARRAY(SUBDS1),NOFLD,NOCLS,NOSUB)                RED00460
C                                                                               RED00470
      CALL CLSCHK(ARRAY(CLSID1),ARRAY(SUBDS1),ARRAY(FLDSV1),                    RED00480
     *            ARRAY(VERTX1),ARRAY(SUBNO1),NOFEAT,FETVEC,                    RED00490
     *            NOCLS,NOFLD,BMFLG,NOSUB)                                      RED00500
C                                                                               RED00510
CCC                                                                             RED00520
CC    COMPUTE REDUCED BASES                                                     RED00530
C                                                                               RED00540
      VARS72 = NOFET2 * (NOFET2 + 1) / 2                                        RED00550
      CLSID2 = 1                                                                RED00560
      SUBNO2 = CLSID2 + NOCLS2                                                  RED00570
      SUBDS2 = SUBNO2 + NOCLS2                                                  RED00580
      FLDSV2 = SUBDS2 + NOSUB2                                                  RED00590
      VERTX2 = FLDSV2 + NOFLD2*4                                                RED00600
      COVAR2 = VERTX2 + TOTVT2 * 2                                              RED00610
      AVAR2 = COVAR2 + VARS72 * NOSUB2                                          RED00620
      COV1 = AVAR2 + (NOFEAT*NOSUB )                                            RED00630
      TIPTOP = COV1 + VARSIZ                                                    RED00640
      RANCOM = TOP - TIPTOP                                                     RED00650
      IF (RANCOM .LT. 0)  GO TO 50                                              RED00660
C                                                                               RED00670
      CALL REDDAT(ARRAY(COVAR2),ARRAY(AVAR2),ARRAY(CLSID1),ARRAY(SUBNO1),       RED00680
     *            ARRAY(SUBDS1),ARRAY(FLDSV1),ARRAY(VERTX1),                    RED00690
     *            ARRAY(COV1),ARRAY(AVAR2),ARRAY(CLSID2),                       RED00700
     *            ARRAY(SUBNO2),ARRAY(SUBDS2),ARRAY(FLDSV2),                    RED00710
     *            ARRAY(VERTX2),NOFEAT,VARSIZ,NOCLS,NOFLD,                      RED00720
     *            NOSUB,FETVEC)                                                 RED00730
C                                                                               RED00740
      GO TO 70                                                                  RED00750
   50 WRITE(6,60)NOFET2,NOSUB2,NOCLS2                                           RED00760
   60 FORMAT(' USER HAS REQUESTED ',I2,' CHANNELS, ',I2,' SUMCLASSES, ',        RED00770
     *  'AND ',I2,' CLASSES.'// THIS COMBINATION OF STATS WILL NOT FIT IN       RED00780
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: REDSAV FORTRAN A

```
     *CORE. PLEASE REDUCE REQUEST.')                                    RED000A00
      CALL CMERR                                                        RED000610
C                                                                       RED000620
   70 CONTINUE                                                          RED000830
      RETURN                                                            RED000840
      END                                                               RED000850
```

```
FILE: RREAD    FORTRAN  A

      SUBROUTINE RREAD(BEGADD,WHERE,TOTWDS,STATUS)                          RRE00010
C                                                                           RRE00020
C     THIS SUBROUTINE SIMULATES THE RANDOM ACCESS READ OF A                 RRE00030
C     WORKFILE USED TO STORE PROGRAM DATA TEMPORARILY DURING A              RRE00040
C     LARSYS PROCESSOR RUN.  THE CALLING ARGUMENTS ARE:                     RRE00050
C                                                                           RRE00060
C     BEGADD - THE NUMBER OF WORDS FROM THE BEGINNING OF THE FILE           RRE00070
C              WHERE THE READ IS TO BEGIN.                                  RRE00080
C     WHERE -  WHERE THE DATA READ IS TO BE PUT (OUTPUT AREA).              RRE00090
C     TOTWDS - THE TOTAL NUMBER OF WORDS TO BE READ.                        RRE00100
C     STATUS - SET TO ZERO WHEN I/O IS COMPLETE (NO LONGER USED,            RRE00110
C              BUT MUST BE RETURNED AS 0).                                  RRE00120
C                                                                           RRE00130
C                                                                           RRE00140
      IMPLICIT INTEGER (A-Z)                                                RRE00150
C                                                                           RRE00160
C  HANSEN  /  VERSION 0800/A/31/77                                          RRE00170
C                                                                           RRE00180
      DIMENSION BUFFER(200),WHERE(1)                                        RRE00190
      BUFSIZ=200                                                            RRE00200
C                                                                           RRE00210
C     BUFFER AND BUFSIZ ARE SET TO THE MOST EFFICIENT SIZE TO               RRE00220
C     MATCH THE PHYSICAL RECORD SIZE OF THE I/O BUFFER.  IN                 RRE00230
C     CMS IT IS 800 BYTES - 200 WORDS.                                      RRE00240
C                                                                           RRE00250
      STATUS=0                                                              RRE00260
      LUN=22                                                                RRE00270
C                                                                           RRE00280
C     LUN IS THE LOGICAL UNIT NUMBER WHERE THE FORTRAN DIRECT               RRE00290
C     ACCESS FILE IS STORED.                                                RRE00300
C                                                                           RRE00310
      I1=1                                                                  RRE00320
      J1=MOD(BEGADD,BUFSIZ)                                                 RRE00330
      IF(J1.EQ.0) J1=BUFSIZ                                                 RRE00340
C                                                                           RRE00350
C     J1 IS THE RELATIVE ADDRESS OF THE BEGINNING WORD IN THE               RRE00360
C     FIRST RECORD TO BE READ.  IF IT IS 0, IT IS THE LAST WORD             RRE00370
C     IN THE RECORD.                                                        RRE00380
C                                                                           RRE00390
      J2=BEGADD                                                             RRE00400
      J3=BEGADD + TOTWDS - 1                                                RRE00410
C                                                                           RRE00420
C     J2 AND J3 ARE THE BEGINNING AND ENDING WORDS OF THE DATA              RRE00430
C     TO BE READ.                                                           RRE00440
C                                                                           RRE00450
      J4=MOD(J3,BUFSIZ)                                                     RRE00460
      IF(J4.EQ.0) J4=BUFSIZ                                                 RRE00470
C                                                                           RRE00480
C     J4 IS THE RELATIVE ADDRESS OF THE ENDING WORD IN THE FINAL            RRE00490
C     RECORD TO BE READ.  IF IT IS 0, IT IS THE LAST WORD IN                RRE00500
C     THE RECORD.                                                           RRE00510
C                                                                           RRE00520
      BEGREC=((J2-1)/BUFSIZ) + 1                                            RRE00530
      ENDREC=((J3-1)/BUFSIZ) + 1                                            RRE00540
      IF(BEGREC.EQ.ENDREC) GOTO 300                                         RRE00550
C                                                                           RRE00560
C     BEGREC AND ENDREC ARE THE RELATIVE ADDRESSES (RECORD                  RRE00570
C     NUMBERS) OF THE FIRST AND LAST RECORDS TO BE READ.  IF                RRE00580
C     THEY ARE EQUAL THEN WE ARE TO BEGIN AND END IN THE                    RRE00590
C     SAME RECORD.                                                          RRE00600
C                                                                           RRE00610
      K1=BEGREC                                                             RRE00620
C                                                                           RRE00630
C     READ THE FIRST RECORD AND MOVE THE REQUIRED PORTION TO                RRE00640
C     THE OUTPUT AREA.                                                      RRE00650
C                                                                           RRE00660
      READ(LUN'K1) BUFFER                                                   RRE00670
      DO 200 K2=J1,BUFSIZ                                                   RRE00680
      WHERE(I1)=BUFFER(K2)                                                  RRE00690
      I1=I1+1                                                               RRE00700
  200 CONTINUE                                                              RRE00710
  210 K1=K1+1                                                               RRE00720
C                                                                           RRE00730
C     READ IN THE NEXT RECORD.  IF IT IS THE FINAL RECORD TO                RRE00740
C     BE READ, GO TO THE FINAL RECORD MOVE CODE ELSE MOVE THE               RRE00750
C     ENTIRE RECORD TO THE OUTPUT AREA.                                     RRE00760
C                                                                           RRE00770
      IF(K1.EQ.ENDREC) GOTO 230                                             RRE00780
                                                                            RRE00790
```

FILE: RREAD FORTRAN A

```
      READ(LUD'K1) BUFFER                                              RRE00800
      DO 220 K2=1,BUFSIZ                                               RRE00810
      WHERE(I1)=BUFFER(K2)                                             RRE00820
      I1=I1+1                                                          RRE00830
  220 CONTINUE                                                         RRE00840
      GOTO 210                                                         RRE00850
C                                                                      RRE00860
C     READ THE FINAL RECORD, MOVE THE REQUIRED PORTION TO THE          RRE00870
C     OUTPUT AREA AND RETURN.                                          RRE00880
C                                                                      RRE00890
  230 READ(LUD'K1) BUFFER                                              RRE00900
      DO 240 K2=1,J4                                                   RRE00910
      WHERE(I1)=BUFFER(K2)                                             RRE00920
      I1=I1+1                                                          RRE00930
  240 CONTINUE                                                         RRE00940
      RETURN                                                           RRE00950
C                                                                      RRE00960
C     HERE WE BEGIN AND END IN THE SAME RECORD. THEREFORE WE           RRE00970
C     ONLY MOVE THE REQUIRED PORTION OF THE DATA TO THE OUTPUT         RRE00980
C     AREA AND RETURN.                                                 RRE00990
C                                                                      RRE01000
  300 READ(LUD'BEGREC) BUFFER                                          RRE01010
      DO 310 K2=J1,J4                                                  RRE01020
      WHERE(I1)=BUFFER(K2)                                             RRE01030
      I1=I1+1                                                          RRE01040
  310 CONTINUE                                                         RRE01050
      RETURN                                                           RRE01060
      END                                                              RRE01070
```

FILE: RWRITE FORTRAN A

```
      SUBROUTINE RWRITE(BEGADD,/WHERE/,TOTWDS,STATUS)                   RRE00010
C                                                                       RRE00020
C                                                                       RRE00030
C     THIS SUBROUTINE SIMULATES THE RANDOM ACCESS WRITE OF A            RRE00040
C     WORKFILE USED TO STORE PROGRAM DATA TEMPORARILY DURING            RRE00050
C     A LARSYS PROCESSOR RUN.  THE CALLING ARGUMENTS ARE:               RRE00060
C                                                                       RRE00070
C     BEGADD - THE NUMBER OF WORDS FROM THE BEGINNING OF THE            RRE00080
C              FILE WHERE THE WRITE IS TO BEGIN.                        RRE00090
C     WHERE -  WHERE THE DATA TO BE WRITTEN IS STORED (INPUT            RRE00100
C              AREA).                                                   RRE00110
C     TOTWDS - THE TOTAL NUMBER OF WORDS TO BE WRITTEN.                 RRE00120
C     STATUS - SET TO ZERO WHEN I/O IS COMPLETE (NO LONGER              RRE00130
C              USED, BUT MUST BE RETURNED AS 0)                         RRE00140
C                                                                       RRE00150
C                                                                       RRE00160
      IMPLICIT INTEGER (A-Z)                                            RRE00170
C                                                                       RRE00180
C   HANSEN  /  VERSION 0800/8/31/77                                     RRE00190
C                                                                       RRE00200
      DIMENSION BUFFER(200),WHERE(1)                                    RRE00210
      BUFSIZ=200                                                        RRE00220
C                                                                       RRE00230
C     BUFFER AND BUFSIZ ARE SET TO THE MOST EFFICIENT SIZE TO           RRE00240
C     MATCH THE PHYSICAL RECORD SIZE OF THE I/O BUFFER.  IN             RRE00250
C     CMS IT IS 800 BYTES = 200 WORDS.                                  RRE00260
C                                                                       RRE00270
      STATUS=0                                                          RRE00280
      LUD=22                                                            RRE00290
C                                                                       RRE00300
C     LUD IS THE LOGICAL UNIT NUMBER WHERE THE FORTRAN DIRECT           RRE00310
C     ACCESS FILE IS STORED.                                            RRE00320
C                                                                       RRE00330
      I1=1                                                              RRE00340
      J1=MOD(BEGADD,BUFSIZ)                                             RRE00350
      IF(J1.EQ.0) J1=BUFSIZ                                             RRE00360
C                                                                       RRE00370
C     J1 IS THE RELATIVE ADDRESS OF THE BEGINNING WORD IN THE           RRE00380
C     FIRST RECORD TO BE WRITTEN.  IF IT IS 0, IT IS THE LAST           RRE00390
C     WORD IN THE RECORD.                                               RRE00400
C                                                                       RRE00410
      J2=BEGADD                                                         RRE00420
      J3=BEGADD + TOTWDS - 1                                            RRE00430
C                                                                       RRE00440
C     J2 AND J3 ARE THE BEGINNING AND ENDING WORDS OF THE DATA          RRE00450
C     TO BE WRITTEN.                                                    RRE00460
C                                                                       RRE00470
      J4=MOD(J3,BUFSIZ)                                                 RRE00480
      IF(J4.EQ.0) J4=BUFSIZ                                             RRE00490
C                                                                       RRE00500
C     J4 IS THE RELATIVE ADDRESS OF THE ENDING WORD IN THE              RRE00510
C     FINAL RECORD TO BE WRITTEN.  IF IT IS 0, IT IS THE                RRE00520
C     LAST WORD IN THE RECORD.                                          RRE00530
C                                                                       RRE00540
      BEGREC=((J2-1)/BUFSIZ) + 1                                        RRE00550
      ENDREC=((J3-1)/BUFSIZ) + 1                                        RRE00560
      IF(BEGREC.EQ.ENDREC) GOTO 300                                     RRE00570
C                                                                       RRE00580
C     BEGREC AND ENDREC ARE THE RELATIVE ADDRESSES (RECORD              RRE00590
C     NUMBERS) OF THE FIRST AND LAST RECORDS TO BE WRITTEN.             RRE00600
C     IF THEY ARE EQUAL, THEN WE ARE TO BEGIN AND END IN                RRE00610
C     THE SAME RECORD.                                                  RRE00620
C                                                                       RRE00630
      K1=BEGREC                                                         RRE00640
C                                                                       RRE00650
C     READ THE FIRST RECORD (THERE MIGHT BE DATA IN THE PORTION         RRE00660
C     OF THE RECORD WE ARE NOT WRITING).  MOVE THE REQUIRED             RRE00670
C     PORTION OF THE DATA FROM THE INPUT AREA TO THE BUFFER             RRE00680
C     AND WRITE IT OUT.                                                 RRE00690
C                                                                       RRE00700
      READ(LUD'K1) BUFFER                                               RRE00710
      DO 200 K2=J1,BUFSIZ                                               RRE00720
      BUFFER(K2)=WHERE(I1)                                              RRE00730
      I1=I1+1                                                           RRE00740
  200 CONTINUE                                                          RRE00750
      WRITE(LUD'K1) BUFFER                                              RRE00760
C                                                                       RRE00770
C     BUMP THE RECORD COUNTER AND CHECK TO SEE WHETHER WE               RRE00780
C     ARE AT THE FINAL RECORD TO BE WRITTEN.  IF WE ARE, GO             RRE00790
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: RWRITE FORTRAN A

```
C       TO THE FINAL RECORD WRITE CODE ELSE MOVE AND WRITE                    RRE00800
C       THE ENTIRE RECORD.                                                    RRE00810
C                                                                             RRE00820
 210    K1=K1+1                                                               RRE00830
        IF(K1.EQ.ENDREC) GOTO 230                                             RRE00840
        DO 220 K2=1,BUFSIZ                                                    RRE00850
        BUFFER(K2)=WHERE(I1)                                                  RRE00860
        I1=I1+1                                                               RRE00870
 220    CONTINUE                                                              RRE00880
        WRITE(LUD'K1) BUFFER                                                  RRE00890
        GOTO 210                                                              RRE00900
C                                                                             RRE00910
C       READ THE FINAL RECORD, MOVE THE REQUIRED PORTION OF THE               RRE00920
C       DATA FROM THE INPUT AREA TO THE BUFFER, WRITE IT OUT                  RRE00930
C       AND RETURN.                                                           RRE00940
C                                                                             RRE00950
 230    READ(LUD'K1) BUFFER                                                   RRE00960
        DO 240 K2=1,J4                                                        RRE00970
        BUFFER(K2)=WHERE(I1)                                                  RRE00980
        I1=I1+1                                                               RRE00990
 240    CONTINUE                                                              RRE01000
        WRITE(LUD'K1) BUFFER                                                  RRE01010
        RETURN                                                                RRE01020
C                                                                             RRE01030
C       HERE WE BEGIN AND END IN THE SAME RECORD. THEREFORE WE                RRE01040
C       ONLY MOVE THE REQUIRED PORTION OF THE DATA FROM THE                   RRE01050
C       INPUT AREA TO THE BUFFER, WRITE IT OUT AND RETURN.                    RRE01060
C                                                                             RRE01070
 300    READ(LUD'BEGREC) BUFFER                                               RRE01080
        DO 310 K2=J1,J4                                                       RRE01090
        BUFFER(K2)=WHERE(I1)                                                  RRE01100
        I1=I1+1                                                               RRE01110
 310    CONTINUE                                                              RRE01120
        WRITE(LUD'BEGREC) BUFFER                                              RRE01130
        RETURN                                                                RRE01140
        END                                                                   RRE01150
```

FILE: SAVFIL FORTRAN A

```
      SUBROUTINE SAVFIL(FLDSAV,VERTEX,CLSID,SUBNO,SUBDES,                SAV00010
     *                  NOFLD,NOCLS,NOSUB)                               SAV00020
C                                                                        SAV00030
      IMPLICIT INTEGER (A-Z)                                             SAV00040
C     INCLUDE COMBK1.LIST                                                SAV00050
C     INCLUDE COMBK6.LIST                                                SAV00060
      COMMON/INFORM/NOCLS2,NOSUB2,NOFET2,VARS22,TOTVT2,NOFLD2,           COM00010
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBDS2,FLDSV2,VERTX2,     COM00020
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),         COM00030
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),              COM00040
     *              GRPCHK(61),GROUPS(124)                               COM00050
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          COM00010
     *              HISFIL,HISKEY,TRFORM,ERIPTR,ERRKEY,MAPUNT,NOFILE,    COM00020
     *     DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                COM00030
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    COM00040
     *   ,DDTUNT,DDTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              COM00050
     *   CRDUNT,PRTUNT,RANDIO                                            COM00060
C$END                                                                    SAV00080
      DIMENSION FLDSAV(4,NOFLD),VERTEX(1),CLSID(1),SUBNO(1),             SAV00090
     *     SUBDES(1)                                                     SAV00100
      IB=0                                                               SAV00110
      DO 40  J=1,NOFLD                                                   SAV00120
      READ(SAVTAP)  (FLDSAV(I,J),I=1,4)                                  SAV00130
      FLD = 2 * FLDSAV(4,J)                                              SAV00140
      READ(SAVTAP)  (VERTEX(I+IB),I=1,FLD)                               SAV00150
      IB = IB + FLD                                                      SAV00160
   40 CONTINUE                                                           SAV00170
      READ(SAVTAP)  (CLSID(I),I=1,NOCLS),(SUBNO(I),I=1,NOCLS),           SAV00180
     *              (SUBDES(I),I=1,NOSUB)                                SAV00190
      RETURN                                                             SAV00200
C                                                                        SAV00210
      END                                                                SAV00220
```

19-62
4/58

FILE: SEARCH FORTRAN A

```
      SUBROUTINE SEARCH(*,*,ENDTAP,IBUF,NRPDS,NDSPR)                    SEA00010
      IMPLICIT INTEGER (A-Z)                                            SEA00020
C*                                                                      SEA00030
C*    INTERNAL ROUTINE TO SEARCH FOR CORRECT SCAN LINE                  SEA00040
C*                                                                      SEA00050
      LOGICAL*1 LSCAN(4),ISCAN(4),KSCAN(4)                              SEA00060
      DIMENSION IBUF(765),KDUM(1),LDUM(1)                               SEA00070
      COMMON /TAPERD/ IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,      SEA00080
     *  LINC,ID(200),DSL,LBUF(30),JREC(30),IBYTE(30),NBUF,FILENO,LINEND,SEA00090
     *  LININC,NSAMP,NOCHAN,FORMT                                       SEA00100
      EQUIVALENCE(ISCAN,SCAN),(KDUM(1),KSCAN(1)),(LDUM(1),LSCAN(1))     SEA00110
      SCAN=0                                                            SEA00120
      WRITE(6,600)FSCAN                                                 SEA00130
  600 FORMAT(' SEARCHING FOR LINE',I5)                                  SEA00140
      WRITE(6,610)NRPDS,NDSPR                                           SEA00150
  610 FORMAT(' RECORDS PER SCAN',I5,'  SCANS PER RECORD',I5)            SEA00160
      IBACK=-5                                                          SEA00170
      IF(NRPDS.GT.5)IBACK=-NRPDS                                        SEA00180
      IF(FSCAN.LE.5)IBACK=-NRPDS                                        SEA00190
      BSKIP =-IBACK                                                     SEA00200
      DO 620  I=1,BSKIP                                                 SEA00210
  620 BACKSPACE IUNIT                                                   SEA00220
      ITRY=1                                                            SEA00230
    5 READ(IUNIT,910,END=630)    (IBUF(I),I=1,18)                     SEA00240
  910 FORMAT(18A4)                                                      SEA00250
      GO TO 640                                                         SEA00260
  630 ENDTAP = -1                                                       SEA00270
      RETURN 1                                                          SEA00280
  640 CONTINUE                                                          SEA00290
      KDUM(1)=IBUF(1)                                                   SEA00300
      LDUM(1)=IBUF(18)                                                  SEA00310
      IF (FORMT .EQ. 1) ISCAN(3) = LSCAN(3)                             SEA00320
      IF (FORMT .EQ. 1) ISCAN(4) = LSCAN(4)                             SEA00330
      IF (FORMT .EQ. 2) ISCAN(3) = KSCAN(1)                             SEA00340
      IF (FORMT .EQ. 2) ISCAN(4) = KSCAN(2)                             SEA00350
      WRITE(6,650)SCAN                                                  SEA00360
  650 FORMAT(' SCAN NO',I5,3X)                                          SEA00370
      IF(SCAN.EQ.FSCAN)GO TO 10                                         SEA00380
      IF(SCAN.EQ.FSCAN+NDSPR)GO TO 30                                   SEA00390
      IF(ITRY.EQ.10)GO TO 20                                            SEA00400
    6 CONTINUE                                                          SEA00410
      ITRY=ITRY+1                                                       SEA00420
      GO TO 5                                                           SEA00430
   10 WRITE(6,700)ITRY                                                  SEA00440
  700 FORMAT(' FOUND IT AFTER',I3,' TRIES'/)                            SEA00450
      DO 15  II=1,NRPDS                                                 SEA00460
   15 BACKSPACE IUNIT                                                   SEA00470
      RETURN                                                            SEA00480
   20 IF(ITRY.LT.2*NRPDS)GO TO 6                                        SEA00490
      WRITE(6,800)ITRY                                                  SEA00500
  800 FORMAT(' FAILED AFTER',I5,' TRIES--ABORTING')                     SEA00510
      CALL CMERR                                                        SEA00520
   30 CONTINUE                                                          SEA00530
      WRITE(6,900)FSCAN                                                 SEA00540
  900 FORMAT(' SCAN',I5,' IS MISSING--USING PREVIOUS SCAN INSTEAD')     SEA00550
      BACKSPACE IUNIT                                                   SEA00560
      RETURN 2                                                          SEA00570
      END                                                               SEA00580
```

FILE: SETMRG FORTRAN A

```
      SUBROUTINE SETMRG(A,B,C)                                          SET00010
      IMPLICIT INTEGER (A-C)                                            SET00020
      RETURN                                                            SET00030
      END                                                               SET00040
```

```
FILE: SUNFAC   FORTRAN  A

      SUBROUTINE SUNFAC(SUNCOR,SUNANG,FETVEC,NOFEAT,ISUNC,ISUNT)          SUN00010
      INTEGER SUNANG,FETVEC,SUNA                                          SUN00020
      EQUIVALENCE (EXTRA(1),DUM1(1)),(EXTRA(109),DUM2(1)),                SUN00030
     *(EXTRA(217),DUM3(1))                                                SUN00040
      DIMENSION EXTRA(324),SUNANG(1),FETVEC(1),SUNCOR(1),                 SUN00050
     *DUM1(108),DUM2(108),DUM3(108)                                       SUN00060
C     INCLUDE COMMK6                                                      SUN00070
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,           SUN00080
     *     HISFIL,HISKEY,TRFORM,ERPTP,ERPKEY,MAPUNT,NOFILE,               SUN00090
     *    DRUMAD,DRM,OS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  SUN00100
     *   ,NHSTUN,NHSTFL,SCTRUN,MAPFIL                                     SUN00110
     *   ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,               SUN00120
     *   CRDUNT,PRTUNT,RANDIO                                             SUN00130
CSEND                                                                     SUN00140
C                                                                         SUN00150
      DATA  DUM1 /16.413,14.887,14.089,13.401,                            SUN00160
     *13.015,11.901,11.323,10.916,10.755,9.893,9.449,9.052,               SUN00170
     *9.094,8.413,8.063,7.745,7.832,7.285,7.002,6.744,                    SUN00180
     *6.880,6.425,6.190,5.973,6.098,5.722,5.527,5.346,                    SUN00190
     *5.474,5.156,4.991,4.838,4.967,4.694,4.551,4.419,                    SUN00200
     *4.530,4.297,4.174,4.061,                                            SUN00210
     *4.143,3.961,3.855,3.757,3.852,3.675,3.583,3.497,                    SUN00220
     *3.580,3.426,3.344,3.269,3.342,3.207,3.135,3.069,                    SUN00230
     *3.135,3.015,2.952,2.893,2.953,2.846,2.789,2.737,                    SUN00240
     *2.788,2.693,2.643,2.597,2.642,2.557,2.513,2.471,                    SUN00250
     *2.511,2.435,2.395,2.358,                                            SUN00260
     *2.393,2.325,2.289,2.255,2.285,2.223,2.191,2.161,                    SUN00270
     *2.187,2.131,2.102,2.076,2.097,2.047,2.021,1.997,                    SUN00280
     *2.016,1.971,1.947,1.925,1.940,1.899,1.878,1.858,                    SUN00290
     *1.871,1.834,1.815,1.797,1.807,1.773,1.756,1.740/                    SUN00300
      DATA DUM2 /1.748,1.717,1.701,1.687,                                 SUN00310
     *1.693,1.665,1.651,1.637,1.642,1.617,1.603,1.591,                    SUN00320
     *1.594,1.571,1.559,1.548,1.550,1.529,1.518,1.508,                    SUN00330
     *1.508,1.489,1.479,1.470,1.469,1.452,1.443,1.435,                    SUN00340
     *1.432,1.417,1.409,1.402,1.398,1.384,1.377,1.370,                    SUN00350
     *1.366,1.353,1.347,1.341,                                            SUN00360
     *1.336,1.324,1.318,1.313,1.307,1.297,1.291,1.286,                    SUN00370
     *1.280,1.271,1.266,1.261,1.255,1.246,1.242,1.238,                    SUN00380
     *1.230,1.223,1.219,1.216,1.208,1.201,1.198,1.195,                    SUN00390
     *1.186,1.180,1.177,1.174,1.166,1.161,1.158,1.155,                    SUN00400
     *1.146,1.142,1.140,1.137,                                            SUN00410
     *1.128,1.124,1.122,1.120,1.111,1.107,1.106,1.104,                    SUN00420
     *1.094,1.091,1.090,1.089,1.078,1.076,1.075,1.074,                    SUN00430
     *1.064,1.062,1.061,1.060,1.050,1.048,1.047,1.047,                    SUN00440
     *1.036,1.035,1.035,1.034,1.023,1.023,1.022,1.022/                    SUN00450
      DATA DUM3 /1.011,1.011,1.011,1.011,                                 SUN00460
     *1.000,1.000,1.000,1.000,.989,.989,.990,.990,                        SUN00470
     *.979,.979,.980,.980,.969,.970,.970,.971,                            SUN00480
     *.960,.961,.961,.962,.951,.952,.953,.954,                            SUN00490
     *.943,.944,.945,.946,.935,.937,.937,.938,                            SUN00500
     *.928,.929,.930,.931,                                                SUN00510
     *.921,.923,.924,.925,.914,.916,.917,.918,                            SUN00520
     *.908,.910,.911,.912,.902,.904,.906,.907,                            SUN00530
     *.897,.899,.901,.902,.892,.894,.896,.897,                            SUN00540
     *.887,.890,.891,.892,.882,.885,.887,.888,                            SUN00550
     *.878,.881,.883,.884,                                                SUN00560
     *.875,.878,.879,.881,.871,.874,.876,.878,                            SUN00570
     *.868,.871,.873,.875,.865,.869,.870,.872,                            SUN00580
     *.863,.866,.868,.870,.861,.864,.866,.868,                            SUN00590
     *.859,.862,.864,.866,.857,.861,.863,.864/                            SUN00600
      DATA BLANK/' '/                                                     SUN00610
      KS = 0                                                              SUN00620
      DO 200 I=1,NOFEAT                                                   SUN00630
      K=(FETVEC(I)-1)/NCHPAS                                              SUN00640
      IF (ISUNC.NE.0.AND.I.EQ.1) KS=K                                     SUN00650
      KR=FETVEC(I)-K*NCHPAS                                               SUN00660
      K=K+1                                                               SUN00670
      SUNA = SUNANG(K-KS)                                                 SUN00680
  100 IND=(SUNA-5)*NCHPAS+KR                                              SUN00690
      SUNCOR(I)=EXTRA(IND)                                                SUN00700
  200 CONTINUE                                                            SUN00710
      WRITE(6,90)                                                         SUN00720
   90 FORMAT(//T61,'SUN ANGLES'/)                                         SUN00730
      WRITE (6,210) (SUNANG(I), I = 1,8)                                  SUN00740
  210 FORMAT(T45,8I5)                                                     SUN00750
      WRITE(6,215)                                                        SUN00760
  215 FORMAT(//T52,'CORRECTIONS FOR SUN ANGLES')                          SUN00770
C                                                                         SUN00780
      NOFETR = NOFEAT                                                     SUN00790
```

FILE: SUNFAC FORTRAN A

```
      ISTART = 1                                                        SUN00800
      IEND = 16                                                         SUN00810
  217 IF (IEND .GE. NOFETR) IEND = NOFETR                               SUN00820
      IENDS = ISTART + IEND - 1                                         SUN00830
      WRITE(6,220)(BLANK,FETVEC(I),I=ISTART,IENDS)                      SUN00840
  220 FORMAT(//1X,16(A1,'CH(',I2,')',1X))                               SUN00850
      WRITE(6,230)(SUNCOR(I),I=ISTART,IENDS)                            SUN00860
  230 FORMAT( 16(2X,F6.4))                                              SUN00870
C                                                                       SUN00880
      NOFETR = NOFETR - IEND                                            SUN00890
      ISTART = IENDS + 1                                                SUN00900
      IF (NOFETR .LE. 0) RETURN                                         SUN00910
      GO TO 217                                                         SUN00920
      END                                                               SUN00930
```

19 66
462

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: TAPHDR FORTRAN A

```
      SUBROUTINE TAPHDR(DATAPE,IFILE)                                   TAP00010
      IMPLICIT INTEGER(A-Z)                                             TAP00020
C*****************************************************************     TAP00030
C*                                                                      TAP00040
C*    TAPERD READS THE MULTISPECTRAL SCANNER DATA TAPE, UNPACKS THE     TAP00050
C*    REQUESTED DATA AND RETURNS IT UNPACKED TO THE CALLING ROUTINE.    TAP00060
C*    THERE ARE THREE SEPARATE  SUBROUTINES, TAPHDR, FLDINT             TAP00070
CC*    AND LINERD, NEEDED TO READ A TAPE                                TAP00080
C*                                                                      TAP00090
C*    TAPHDR MUST BE CALLED ONCE TO READ THE HEADER RECORD AND UNPACK   TAP00100
C*     NECESSARY DATA FROM THE RECORD                                   TAP00110
C*                                                                      TAP00120
C*    CALL TAPHDR(DATAPE,IFILE)                                         TAP00130
C*        DATAPE-INPUT UNIT NUMBER FOR DATA TAPE                        TAP00140
C*        IFILE - NO. OF E-O-F'S ON DATA TAPE USER WISHES TO READ OVER   TAP00150
C*                IN ORDER TO POSITION TAPE TO DESIRED FILE             TAP00160
C*                                                                      TAP00170
C*    FLDINT MUST BE CALLED ONCE FOR EACH FIELD. THE TAPE IS POSITIONED TAP00180
C*    TO THE CORRECT RECORD AND PARAMETERS ARE INITIALIZED FOR THE FIELDTAP00190
C*                                                                      TAP00200
C*    CALL FLDINT(BLOCK,FETVEC,NOFEAT)                                  TAP00210
C*        BLOCK(1)=LINE START                                           TAP00220
C*        BLOCK(2)=LINE END                                             TAP00230
C*        BLOCK(3)=LINE INCREMENT                                       TAP00240
C*        BLOCK(4)=SAMPLE START                                         TAP00250
C*        BLOCK(5)=SAMPLE END                                           TAP00260
      CONTINUE                                                          TAP00270
C*        BLOCK(6)=SAMPLE INCREMENT                                     TAP00280
C*        FETVEC- (INPUT) VECTOR CONTAINING FEATURES REQUESTED          TAP00290
C*        NOFEAT (INPUT) NO. OF FEATURES IN FETVEC                      TAP00300
C*                                                                      TAP00310
C*    CALL LINERD(IDATA,ENDTAP)                                         TAP00320
C*        IDATA-(OUTPUT) ARRAY CONTAINING UNPACKED DATA                 TAP00330
C*        ENDTAP - TIGGER INDICATING WHETHER OF NOT AN E-O-F HAS BEEN   TAP00340
C*                 REACHED  WHILE TRYING TO READ A GIVEN  SCAN LINE NO. TAP00350
C*                 IF AN E-O-F IS FOUND ENDTAP = -1, OTHERWISE, ENDTAP  TAP00360
C*                 = 0.                                                 TAP00370
C*                                                                      TAP00380
C*                                                                      TAP00390
C*****************************************************************     TAP00400
C*    READY IS A  INDICATOR TO TEST WHETHER THE TAPE HAS BEEN           TAP00410
C*    POSITIONED AND PARAMETERS SET FOR A FIELD                         TAP00420
C*                                                                      TAP00430
C*    THE ARRAYS NB AND BWRD ARE PRECALCULATED WORD AND BIT             TAP00440
C*    POSITIONS OF INFORMATION IN THE HEADER RECORD OF THE UNIVERSAL    TAP00450
C*    FORMAT WHICH MUST BE EXTRACTED.                                   TAP00460
      CONTINUE                                                          TAP00470
C*                                                                      TAP00480
C*      NRPDS - NO. OF RECORDS PER DATA SET                             TAP00490
C*      NCPR - NO. OF CHANNELS PER RECORD ON RECORDS PAST ANCILLARY RECOTAP00500
C*      NPRC - NO. OF PHYSICAL RECORDS PER CHANNEL                      TAP00510
C*      ANCLNG - ANCILLARY LENGTH IN BYTES                              TAP00520
C*      NC - NO. OF CHANNELS                                            TAP00530
C*      NS - NO. OF SAMPLES PER CHANNEL PER SCAN                        TAP00540
C*      NBITS - NO. OF BITS PER PIXEL                                   TAP00550
C*      DOI - DATA ORDER INDICATOR                                      TAP00560
C*      NDSPR - NO. OF DATA SETS PER RECORD                             TAP00570
C*      NCAR - NO. OF CHANNELS OF VIDEO DATA ON SAME RECORD             TAP00580
C*             WITH ANCILLARY DATA)                                     TAP00590
C*      SVD - START OF VIDEO DATA. (BYTE POSITION WITHIN DATA FOR       TAP00600
C*            A GIVEN CHANNEL)                                          TAP00610
C*                                                                      TAP00620
      LOGICAL*1 IBUF(30600)                                             TAP00630
      LOGICAL*1 IWORD(4), ILINE(4)                                      TAP00640
      DIMENSION FRM(3,2)                                                TAP00650
       DIMENSION NB(28),BWRD(28)                                        TAP00660
      COMMON /TAPERD/ IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,      TAP00670
     *  LINC,ID(200),DSL,LBUF(30),JREC(30),IBYTE(30),NBUFS,FILENO,LINENDTAP00680
     *   ,LININC,NSAMP,NOCHAN,FORMT                                     TAP00690
      COMMON /ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP     TAP00700
      EQUIVALENCE (IWORD,JORD)                                          TAP00710
      EQUIVALENCE (IFRST,ILINE)                                         TAP00720
      EQUIVALENCE (ID(1),NRPDS ),(ID(2),NCPR  ),                        TAP00730
     *            (ID(3),NPRC  ),(ID(4),ANCLNG),                        TAP00740
     *            (ID(5),NC    ),(ID(6),NS    ),                        TAP00750
     *            (ID(7),NBITS ),(ID(8),DOI),                           TAP00760
     *            (ID(9),NDSPR),(ID(10),NCAR ),                         TAP00770
     *          (ID(11),SVD),(ID(16),PRS7)                              TAP00780
      DATA FRM/'UNIV','ERSA','L   ','LABS','YS 2',                      TAP00790
```

FILE: TAPHDR FORTRAN A

```
     *                 '  */                                                  TAP00800
      DATA HWRD/106,102,103,105,90,1787,91,107,1774,1785,92,108,110,         TAP00810
     *     1789,1791,100,2201,2203,2205,2207,2209,2211,2213,2215,            TAP00820
     * 61,62,63,67/                                                          TAP00830
      DATA NA/8,8,8,16,8,16,8,8,8,16,16,16,16,16,16,16,                      TAP00840
     *  16,16,16,16,16,16,16,16,8,8,8,16/                                    TAP00850
C*                                                                          TAP00860
C*                                                                          TAP00870
C*    ENTRY FOR READING HEADER INFORMATION                                  TAP00880
C*                                                                          TAP00890
C*                                                                          TAP00900
C*    INFORMATION IN EBCDIC OR IBM FLOATING POINT IS NOT UNPACKED           TAP00910
C*    FROM THE HEADER RECORD AT THIS TIME.                                  TAP00920
C*                                                                          TAP00930
      READY = -1                                                            TAP00940
      IFRST = 0                                                             TAP00950
      FILENO = IFILE                                                        TAP00960
      REC = 0                                                               TAP00970
      IUNIT=DATAPE                                                          TAP00980
      KBUF = 3060                                                           TAP00990
      IF (ISUNC.NE.0) GO TO 6                                               TAP01000
      DO 5 I = 1,8                                                          TAP01010
    5 SUNANG(I) = 60                                                        TAP01020
    6 CONTINUE                                                              TAP01030
C*                                                                          TAP01040
      REWIND IUNIT                                                          TAP01050
      NRPDS=1                                                               TAP01060
      IF ( FILENO .EQ. 0) GO TO 600                                         TAP01070
      DO 8  I=1,FILENO                                                      TAP01080
      WRITE(6,610) I                                                        TAP01090
  610    FORMAT(' SKIPPING FILE',I8)                                        TAP01100
    7 READ(IUNIT,510,END=8)DUMMY                                            TAP01110
      GO TO 7                                                               TAP01120
    8 CONTINUE                                                              TAP01130
  600 IF(FORMT.EQ.2) GO TO 11                                               TAP01140
      CALL BUFILL(REC,IUNIT,KBUF,IBUF,NRPDS,ENDTAP,IERR)                    TAP01150
      IF (IERR .EQ. -1) GO TO 11                                            TAP01160
      FORMT = 1                                                             TAP01170
      GO TO 20                                                              TAP01180
   11 KBUF = 800                                                            TAP01190
      REC = 0                                                               TAP01200
      CALL BUFILL(REC,IUNIT,KBUF,ID,NRPDS,ENDTAP,IERR)                      TAP01210
      IF ( IERR .EQ. -1) GO TO 10                                           TAP01220
      FORMT = 2                                                             TAP01230
      GO TO 20                                                              TAP01240
   10 WRITE (6,280)                                                         TAP01250
      WRITE(6,340)                                                          TAP01260
      CALL CMERR                                                            TAP01270
   20 IF (FORMT .EQ. 1) GO TO 40                                            TAP01280
      IF (FORMT .NE. 2) GO TO 265                                           TAP01290
      NCS=NC*NS                                                             TAP01300
      PRSZ = NCS + 4                                                        TAP01310
      IF (PRSZ .LE. 30800) GO TO 35                                         TAP01320
      WRITE(6,330)NC,NS                                                     TAP01330
      CALL CMERR                                                            TAP01340
   35 CONTINUE                                                              TAP01350
      NRPDS=1                                                               TAP01360
       MAXREC = PRSZ                                                        TAP01370
      NCAP=NC                                                               TAP01380
      ANCLNG=4                                                              TAP01390
      SVD=1                                                                 TAP01400
      NBITS=8                                                               TAP01410
      DOI=0                                                                 TAP01420
      NCPR=0                                                                TAP01430
      NDSPR=1                                                               TAP01440
      NPRC = 0                                                              TAP01450
      WRITE(6,481)(FRM(I,2),I=1,3),NC,NS                                    TAP01460
      SMSTR=1                                                               TAP01470
      GO TO 100                                                             TAP01480
C*                                                                          TAP01490
C*    UNPACK NECESSARY INFORMATION FROM HEADER RECORD-UNIVERSAL FORMAT       TAP01500
C*                                                                          TAP01510
   40   ILIM = 156                                                          TAP01520
        DO 60 I = 1,ILIM                                                    TAP01530
        IWD = 112 + (I - 29)*4                                              TAP01540
        IF (I.LT.29) IWD = HWRD(I)                                          TAP01550
       WORD = 0                                                             TAP01560
        NBYTES = 4                                                          TAP01570
        IF (I.LT.29) NBYTES = NA(I)/8                                       TAP01580
```

FILE: TAPHDR FORTRAN A

```
      DO 55  J=1,NBYTES                                                 TAP01590
      LOC = 4 * J - NBYTES                                              TAP01600
      IPOS=IWD+J-1                                                      TAP01610
   55 IWORD(LOC) = IBUF(IWD+J-1)                                        TAP01620
      ID(I) = WORD                                                      TAP01630
   60 CONTINUE                                                          TAP01640
      SMSTR = ID(12)                                                    TAP01650
      SMSTP = ID(13)                                                    TAP01660
      SMINC = ID(14)                                                    TAP01670
      LINSKP = ID(15)                                                   TAP01680
      IF (ISUNT.EQ.0) GO TO 65                                          TAP01690
      DO 62 I=1,8                                                       TAP01700
      IF(ID(16+I).LT.5.OR.ID(16+I).GT.85) ID(16+I) = 60                 TAP01710
   62 SUNANG(I) = ID(16+I)                                              TAP01720
   65 CONTINUE                                                          TAP01730
           WORD = 0                                                     TAP01740
           DO 66 I=1,4                                                  TAP01750
           IPAT = 2254 + (I - 1)*8                                      TAP01760
           IWORD(4) = IBUF(IPAT)                                        TAP01770
   66      ID(159 + I) = WORD                                           TAP01780
C***                                                                    TAP01790
C***    CODE JUST ABOVE ADDED OCT.20.1978 TO UNPACK SOIL LINES          TAP01800
C***                                                                    TAP01810
      IF (ISUNT.GT.0) WRITE(6,482) (SUNANG(I), I=1,8)                   TAP01820
  482 FORMAT(1H//' SUN ANGLES : ',8I6)                                  TAP01830
      WRITE(6,481)(FRM(I,1),I=1,3),NC,NS                                TAP01840
      MAXREC = PRSZ                                                     TAP01850
   70 IF (NREC.LE.1) GO TO 80                                           TAP01860
      WRITE (6,360)                                                     TAP01870
      CALL CMERR                                                        TAP01880
   80 CONTINUE                                                          TAP01890
      IF(SVD.LE.0)SVD=1                                                 TAP01900
      IF(NDSPR.LE.0)NDSPR=1                                             TAP01910
      IF (NBITS.EQ.8) GO TO 90                                          TAP01920
      WRITE (6,390)NBITS                                                TAP01930
      NBITS=8                                                           TAP01940
   90 IF (DOI.EQ.0) GO TO 100                                           TAP01950
      WRITE (6,400)DOI                                                  TAP01960
      CALL CMERR                                                        TAP01970
  100 CONTINUE                                                          TAP01980
      KPTS=0                                                            TAP01990
      IRO=0                                                             TAP02000
C*                                                                      TAP02010
C*    DATA SET LENGTH IN BYTES                                          TAP02020
      DSL=ANCLNG+NS*NC                                                  TAP02030
C*                                                                      TAP02040
C*    READ FIRST DATA SET TO DETERMINE FIRST SCAN LINE NUMBER           TAP02050
C*                                                                      TAP02060
      BUF=1                                                             TAP02070
      REC=0                                                             TAP02080
      CALL BUFILL(REC,IUNIT,MAXREC,IBUF,NRPDS,ENDTAP,IERR)              TAP02090
C                                                                       TAP02100
C     IFRST = ILINE(1-4)                                                TAP02110
C                                                                       TAP02120
      IF (FORMT .EQ. 1) ILINE(3) = IBUF(71)                             TAP02130
      IF (FORMT .EQ. 1) ILINE(4) = IBUF(72)                             TAP02140
      IF (FORMT .EQ. 2) ILINE(3) = IBUF(1)                              TAP02150
      IF (FORMT .EQ. 2) ILINE(4) = IBUF(2)                              TAP02160
      IF (IFRST.GT.0) GO TO 120                                         TAP02170
      WRITE (6,300)                                                     TAP02180
      WRITE(6,340)                                                      TAP02190
      CALL CMERR                                                        TAP02200
  120 FSCAN=IFRST                                                       TAP02210
      WRITE(6,500)IFRST,SMSTR                                           TAP02220
      RETURN                                                            TAP02230
  265 WRITE(6,340)                                                      TAP02240
      CALL CMERR                                                        TAP02250
      RETURN                                                            TAP02260
  280 FORMAT(' UNRECOVERABLE ERROR READING HEADER RECORD')              TAP02270
  300 FORMAT(' A LINE NO. IS LESS THAN OR EQUAL ZERO')                  TAP02280
  310 FORMAT(' LAST SCAN LINE READ=',I5,' ISTAT=',I5)                   TAP02290
  330 FORMAT(/' INTERNAL DIMENSIONS TOO SMALL FOR DATA'/' NO. OF CHANNEL TAP02300
     *S ON DATA TAPE=',I7,' NO. OF POINTS/CHANNEL=',I7/)                TAP02310
  360 FORMAT(' CHECK THE FOLLOWING POSSIBLE ERRORS'/' 1. DATA TAPE IS NOTAP02320
     *T IN UNIVERSAL OR LARSYS FORMAT')                                 TAP02330
  340 FORMAT(' ONLY ONE OR LESS RECORDS PER CHANNEL ACCEPTABLE AT THIS TTAP02340
     *IME')                                                             TAP02350
  370 FORMAT(' NO. OF RECORDS PER DATA SET=',I5,' MUST BE LESS THAN OR ETAP02360
     *QUAL 15')                                                         TAP02370
```

FILE: TAPHOR FORTRAN A

```
  390 FORMAT(' NO. OF BITS/PIXEL=',I5,' ONLY 8 BITS ACCEPTABLE AT THIS TTAP02380
     *IME')                                                          TAP02390
  400 FORMAT(' DATA ORDER INDICATOR=',I5/' DATA MUST BE ORDERED BY PIXELTAP02400
     *')                                                             TAP02410
  481 FORMAT(1H  ///' INPUT IMAGE DATA TAPE INFORMATION'//           TAP02420
     *   5X,'FORMAT',T30,3A4/                                        TAP02430
     *   5X,'NO. OF CHANNELS',T30,I4 /                               TAP02440
     *   5X,'NO. OF PIXELS/LINE',T30,I4)                             TAP02450
  500 FORMAT(5X,'FIRST SCAN LINE NO.',T30,I4/                       TAP02460
     *   5X,'FIRST PIXEL REFERENCE PT.',T30,I4)                      TAP02470
  510 FORMAT(1A4)                                                   TAP02480
      END                                                           TAP02490
```

FILE: WRTBMT FORTRAN A

```
      SUBROUTINE WRTBMT(BMAT,NOFET4,NOFET2,FETVC2)                      WRT00010
      DIMENSION CH(2)                                                   WRT00020
      DATA CH/'CH( ',' )  '/                                            WRT00030
      INTEGER FETVC2(30)                                                WRT00040
C     INCLUDE COMMON.LIST                                               WRT00050
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,         WRT00060
     *              HISFIL,HISKEY,TRFORM,ENIPTP,ERMKEY,MAPUNT,NOFILE,   WRT00070
     *     DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL               WRT00080
     *  ,NRSTUN,NRSTFI,SCTHDR,MAPFIL                                    WRT00090
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              WRT00100
     *  CRDUNT,PRTUNT,RANDIO                                            WRT00110
C     END                                                               WRT00120
      DIMENSION BMAT(NOFET4,NOFET2)                                     WRT00130
      DOUBLE PRECISION BMAT                                             WRT00140
      GO TO 4                                                           WRT00150
      ENTRY WRTRM(RBMAT,NOFET4,NOFET2,FETVC2)                           WRT00160
      DIMENSION RBMAT(NOFET4,NOFET2)                                    WRT00170
      K=1                                                               WRT00180
    4 CONTINUE                                                          WRT00190
      WRITE(6,HEAD)                                                     WRT00200
      WRITE(6,100)NOFET4,NOFET2                                         WRT00210
      IB=1                                                              WRT00220
      IK=12                                                             WRT00230
    5 IF(IK.GT.NOFET2)IK=NOFET2                                         WRT00240
      WRITE(6,200)(CH(1),CH(2), I=IB,IK)                                WRT00250
      WRITE(6,300)(FETVC2(I),I=IB,IK)                                   WRT00260
      WRITE(6,350)                                                      WRT00270
      IF(K.EQ.0) GO TO 11                                               WRT00280
      DO 12 J=1,NOFET4                                                  WRT00290
   12 WRITE(6,400 ) J,(RBMAT(J,I),I=IB,IK)                              WRT00300
      GO TO 13                                                          WRT00310
   11 CONTINUE                                                          WRT00320
      DO 10 J=1,NOFET4                                                  WRT00330
   10 WRITE(6,400)J,(BMAT(J,I),I=IB,IK)                                 WRT00340
   13 CONTINUE                                                          WRT00350
      IF(IK.EQ.NOFET2)GO TO 20                                          WRT00360
      IB=IK+1                                                           WRT00370
      IK=IK+12                                                          WRT00380
      GO TO 5                                                           WRT00390
   20 RETURN                                                            WRT00400
  100 FORMAT(///45X,'LINEAR TRANSFORMATION (B) MATRIX'//                WRT00410
     *   50X,'NO. LINEAR COMB.  =',I3/                                  WRT00420
     *   50X,'NO. CHANNELS      =',I3/)                                 WRT00430
  200 FORMAT(/12X,12(A4,A4,2X))                                         WRT00440
  300 FORMAT(1H0,14X,11(I2,8X),I2)                                      WRT00450
  350 FORMAT(1X,'LIN. CMB.')                                            WRT00460
  400 FORMAT(1X,I5,4X,12(1X,E9.3))                                      WRT00470
      END                                                               WRT00480
```

FILE: WRTDOT FORTRAN A

```
C
CCC     DOTFIL OUTPUTS THE DOT DATA FILE                                        WRT00010
CCC     A FILE IS CREATED FOR TYPE OF DOTS                                      WRT00020
CC                                                                              WRT00030
        SUBROUTINE WRTDOT(TOTDOT,NOSUN,FLDSAV,VERTEX,ANGLE,DOTS,NOCAT,         WRT00040
     *                    CATNAM,SIZE,NOFET2,FETVC2,TOTVT2,NOFLD2,             WRT00050
     *                    UNIT,FILE)                                           WRT00060
C                                                                               WRT00070
C       IMPLICIT INTEGER (A-Z)                                                  WRT00080
C                                                                               WRT00090
        DIMENSION CATNAM(NOCAT),FLDSAV(4,1),VERTEX(2,1),ANGLE(1)                WRT00100
        DIMENSION FETVC2(30),DOTS(SIZE,TOTDOT)                                  WRT00110
C                                                                               WRT00120
        DOTFIL = FILE                                                           WRT00130
        DOTUNT = UNIT                                                           WRT00140
C                                                                               WRT00150
CCC     POSITION TO DESIRED FILE                                                WRT00160
C                                                                               WRT00170
        REWIND DOTUNT                                                           WRT00180
        CALL FSBSFL(DOTUNT,DOTFIL,ISTAT)                                        WRT00190
C                                                                               WRT00200
CCC     REC NO. 1  -- INDICES FOR REC NO. 2                                     WRT00210
C                                                                               WRT00220
        WRITE(DOTUNT)NOCAT,NOFET2,NOFLD2,TOTVT2,TOTDOT,NOSUN,(CATNAM(I),        WRT00230
     *               I=1,NOCAT),SIZE                                            WRT00240
C                                                                               WRT00250
CCC     REC NO. 2                                                               WRT00260
C                                                                               WRT00270
        WRITE(DOTUNT)(FETVC2(I),I=1,NOFET2),((FLDSAV(I,J),I=1,4),J=1,           WRT00280
     *      NOFLD2),((VERTEX(I,J),I=1,2),J=1,TOTVT2),(ANGLE(I),I=1,NOSUN)       WRT00290
C                                                                               WRT00300
CCC     REC NO. 3 -- DOT DATA INFO                                              WRT00310
C                                                                               WRT00320
        WRITE(DOTUNT)((DOTS(I,J),I=1,SIZE),J=1,TOTDOT)                          WRT00330
C                                                                               WRT00340
        END FILE DOTUNT                                                         WRT00350
          RETURN                                                                WRT00360
C                                                                               WRT00370
        END                                                                     WRT00380
                                                                                WRT00390
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: WRTFLD FORTRAN A

```
      SUBROUTINE WRTFLD(FLDSAV,VERTEX,NOFLD,KEY,CLSNAM,SUBNAM)          WRT00010
      IMPLICIT INTEGER(A-Z)                                             WRT00020
      DIMENSION CLSNAM(1),SUBNAM(1)                                     WRT00030
C*                                                                      WRT00040
C*    THIS SUBROUTINE PRINTS SAVED TRAINING OR TEST FIELDS              WRT00050
C*                                                                      WRT00060
      DIMENSION FLDSAV(4,NOFLD),VERTEX(2,1)                             WRT00070
      DATA LPRN/'('/                                                    WRT00080
C     INCLUDE COMRK6.LIST                                               WRT00090
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,         WRT00100
     *              HISFIL,HISKEY,TRFORM,FHIPTP,ERRKEY,MAPUNT,NOFILE,   WRT00110
     *    DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                WRT00120
     *  ,NHSTUN,NHSTFI,SCTMUN,MAPFIL                                    WRT00130
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              WRT00140
     *  CRDUNT,PRTUNT,RAND;0                                            WRT00150
C*END                                                                   WRT00160
      IB=1                                                              WRT00170
      WRITE(6,HEAD)                                                     WRT00180
      IF(KEY.EQ.1)WRITE(6,100)                                          WRT00190
      IF(KEY.EQ.2)WRITE(6,200)                                          WRT00200
      IF(KEY.NE.3)WRITE(6,300)                                          WRT00210
      IF(KEY.EQ.3)WRITE(6,250)                                          WRT00220
      DO 10 I=1,NOFLD                                                   WRT00230
      NV=FLDSAV(4,I)                                                    WRT00240
      NQ=NV-1                                                           WRT00250
      NR=NQ-5                                                           WRT00260
      IF(NQ.GT.5)NQ=5                                                   WRT00270
      IE=IB+NQ-1                                                        WRT00280
      IC=FLDSAV(2,I)                                                    WRT00290
      IS=FLDSAV(3,I)                                                    WRT00300
      FLDNAM=FLDSAV(1,I)                                                WRT00310
      IF(KEY.NE.3)GO TO 5                                               WRT00320
      WRITE(6,700)I,FLDNAM,(LPRN,VERTEX(1,J),VERTEX(2,J),J=IB,IE)       WRT00330
      IF(IS.EQ.1)WRITE(6,705)                                           WRT00340
      IF(IS.EQ.2)WRITE(6,710)                                           WRT00350
      GO TO 6                                                           WRT00360
    5 CONTINUE                                                          WRT00370
      IF(IS.EQ.0)WRITE(6,400)I,FLDNAM,IC,CLSNAM(IC),                    WRT00380
     *                   (LPRN,VERTEX(1,J),VERTEX(2,J),J=IB,IE)         WRT00390
      IF(IS.NE.0)WRITE(6,500)I,FLDNAM,IC,CLSNAM(IC),IS,SUBNAM(IS),      WRT00400
     *                   (LPRN,VERTEX(1,J),VERTEX(2,J),J=IB,IE)         WRT00410
    6 CONTINUE                                                          WRT00420
      IF(NR.LE.0)GO TO 7                                                WRT00430
      IB=IE+1                                                           WRT00440
      IE=IB+NR-1                                                        WRT00450
      WRITE(6,650)(LPRN,VERTEX(1,J),VERTEX(2,J),J=IB,IE)                WRT00460
    7 CONTINUE                                                          WRT00470
      IB=IE+2                                                           WRT00480
   10 CONTINUE                                                          WRT00490
      RETURN                                                            WRT00500
  100   FORMAT(/// 20X,'AREA USED TO COMPUTE TRAINING STATISTICS'/)     WRT00510
  200 FORMAT(/// 50X,'INPUT FIELDS'/)                                   WRT00520
  250 FORMAT(///45X,'DESIGNATED FIELDS'///T18,'FIELD',T40,'DESIGNATED', WRT00530
     *    T80,'VERTICES (SAMPLE,LINE)'/)                                WRT00540
  300 FORMAT(1X,T14,'FIELD',T34,'CLASS',T47,'SUBCLASS',T80,'VERTICES  (SWRT00550
     *AMPLE,LINE)'/)                                                    WRT00560
  400 FORMAT(T15,I3,T20,A4,T30,I3,T35,A4,T65,5(A1,I4,',',I4,')',1X))    WRT00570
  500 FORMAT(T15,I3,T20,A4,T30,I3,T35,A4,T45,I3,T50,A4,T65,             WRT00580
     *  5(A1,I4,',',I4,')',1X))                                        WRT00590
  650 FORMAT(1X,T65,5(A1,I4,',',I4,')',1X))                             WRT00600
  700 FORMAT(T15,I3,T20,A4,T65,5(A1,I4,',',I4,')',1X))                  WRT00610
  705 FORMAT(1H+,T40,'UNIDENTIFIABLE')                                  WRT00620
  710 FORMAT(1H+,T40,'OTHER')                                           WRT00630
      END                                                               WRT00640
```

FILE: WRTHED FORTRAN A

```
                                                                                WRT00010
      SUBROUTINE WRTHED(NCHAN,FEAT,NSAMP,FRMAT,IUNIT)                           WRT00020
      IMPLICIT INTEGER (A-Z)                                                    WRT00030
CI*****************************************************************CIWRT00040
CI                                                                    CIWRT00050
CI    THE PURPOSE OF TAPWRT IS TO WRITE A DATA TAPE IN EITHER UNIVER- CIWRT00060
CI    SAL FORMAT OR LARSYS II FORMAT. THERE ARE TWO ENTRY POINTS TO   CIWRT00070
CI    THE SUBROUTINE -- WRTHED AND WRTLN .                            CIWRT00080
CI    WRTHED WRITES THE HEADER RECORD IN 32 BIT BYTES FOR LARSYS II   CIWRT00090
CI    AND 8 BIT BYTES FOR UNIVERSAL. ONE CALL TO WRTHED MUST BE MADE  CIWRT00100
CI    FOR EACH REEL OF TAPE. THIS INFORMATION IS PACKED.              CIWRT00110
CI                                                                    CIWRT00120
C     CALL WRTHED(NC,FEAT,NSAMP,FORMAT,TRFORM)                        WRT00130
CI         NC     -- NO.OF CHANNELS TO BE WRITTEN FOR EACH DATA SET   CIWRT00140
CI         FEAT   -- ARRAY CONTAINING CHANNELS TO BE WRITTEN          CIWRT00150
CI         NSAMP  -- NO. OF SAMPLES PER CHANNEL                       CIWRT00160
CI         FORMAT -- =1 FOR UNIVERSAL                                 CIWRT00170
C          TRFORM -- NO. OF TAPE OUTPUT UNIT                          WRT00180
CI                   =2 FOR LARSYS II                                 CIWRT00190
CI    WRTLN  WRITES THE DATA IN 8 BIT BYTES AND IS ALSO PACKED. A     CIWRT00200
CI    CALL TO THIS ROUTINE MUST BE MADE FOR EACH DATA SET TO BE WRIT- CIWRT00210
CI    TEN                                                             CIWRT00220
CI                                                                    CIWRT00230
CI    CALL WRTLN (IDATA,LSTLIN)                                       CIWRT00240
CI         IDATA  -- ARRAY CONTAINING DATA TO BE WRITTEN              CIWRT00250
CI         LSTLIN -- = 0 FOR N-1 DATA SETS                            CIWRT00260
CI                   =-1 FOR LAST DATA SET                            CIWRT00270
CI                                                                    CIWRT00280
CI         ICHAN  -- ACTIVE CHANNELS HAVE CORRESPONDING BIT POSITION  CIWRT00290
CI                   TURNED ON                                        CIWRT00300
CI         PROFLG --                                                  CIWRT00310
      CONTINUE                                                        WRT00320
CI         NCS    -- NO. OF CHANNELS                                  CIWRT00330
CI         NBITS  -- NO. OF BITS PER BYTE                             CIWRT00340
CI         SVD    -- START OF VIDEO DATA                              CIWRT00350
CI         NVE    -- SAME  AS NSAMP                                   CIWRT00360
CI         PRSZ   -- PHYSICAL RECORD SIZE IN BYTES                    CIWRT00370
CI         NCPR   -- NO. CHANNELS PER RECORD                          CIWRT00380
CI         NPRC   -- NO. PHYICAL RECORDS PER CHANNEL                  CIWRT00390
CI         NRPDS  -- NO. OF RECORDS PER DATA SET                      CIWRT00400
CI         ANCLNG -- LENGTH OF ANCILLARY BLOCK IN BYTES               CIWRT00410
CI         DOT    -- DATA ORDER INDICATOR                             CIWRT00420
CI         SAMSTR -- SAMPLE START                                     CIWRT00430
CI         COMWRD -- SIZE OF COMPUTER WORD IN BITS                    CIWRT00440
CI         NOSPR  -- NO. OF DATA SETS PER RECORD                      CIWRT00450
CI         NCAR   -- NO. OF CHANNELS ON ANCILLARY RECORD              CIWRT00460
CI    PACRAY -- DATA IS PACKED INTO THIS ARRAY AND THEN WRITTEN ON    CIWRT00470
CI              TAPE BY CALLING NTRAN                                 CIWRT00480
CI    ICOUNT -- RUNNING TOTAL OF NO. OF DATA SETS WRITTEN             CIWRT00490
CI                                                                    CIWRT00500
CI*****************************************************************CIWRT00510
      LOGICAL*1 PACRAY(3060)                                          WRT00520
      REAL RAY(200)                                                   WRT00530
      DIMENSION NB(18),PACK(765)                                      WRT00540
      DIMENSION IBYTES(18),FEAT(30),IRAY(200)                         WRT00550
       LOGICAL*1 VARIAB(2400)                                         WRT00560
      COMMON /WRTAP/ICOUNT,FORMT,UNIT,VARBL(600),IREMD                WRT00570
      DATA IBYTES/81,89,90,91,92,96,100,102,103,104,105,107,108,110,  WRT00580
     *      753,1775,1785,1787/                                       WRT00590
      DATA NB/4,1,1,2,2,2,1,1,1,2,1,2,2,1,1,2,2/                      WRT00600
      EQUIVALENCE (VARBL,VARIAB),(PACK,PACRAY),(IRAY,RAY)             WRT00610
      EQUIVALENCE                                  (VARBL(14),NOSAM), WRT00620
     1            (VARBL(1),ICHAN), (VARBL(7),PRSZ),                 WRT00630
     2            (VARBL(2),PROFLG),(VARBL(8),NCPR),  (VARBL(15),COMWRD) WRT00640
     3,           (VARBL(3),NC),    (VARBL(9),NPRC),                 WRT00650
     4            (VARBL(4),NBITS), (VARBL(10),NRPDS), (VARBL(16),NOSPR), WRT00660
     5            (VARBL(5),SVD),   (VARBL(11),ANCLNG),              WRT00670
     6                              (VARBL(12),DOT),   (VARBL(17),NCAR), WRT00680
     7            (VARBL(6),NVE),   (VARBL(13),SAMSTR),(VARBL(18),NSAM) WRT00690
      NC = NCHAN                                                      WRT00700
      FORMT = FRMAT                                                   WRT00710
      UNIT = IUNIT                                                    WRT00720
      ICOUNT=0                                                        WRT00730
      SAMSTR=1                                                        WRT00740
       ICHAN = 0                                                      WRT00750
      ISTAT =0                                                        WRT00760
      NVE = NSAMP                                                     WRT00770
      NSAM = NSAMP                                                    WRT00780
      NOSAM = NSAMP                                                   WRT00790
```

FILE: WRTHED FORTRAN A

```
      IF ( FORMT .EQ. 1) GO TO 40                                       WRT00800
C                                                                       WRT00810
CC    ZERO OUT HEADER RECORD STORAGE                                    WRT00820
C                                                                       WRT00830
      DO 5 I=1,200                                                      WRT00840
    5 IRAY(I) = 0                                                       WRT00850
C                                                                       WRT00860
CC    PACKING HEADER RECORD IN LARSYS II FORMAT                         WRT00870
C                                                                       WRT00880
      IRAY(5) =NC                                                       WRT00890
      IRAY(6) =NSAMP                                                    WRT00900
C                                                                       WRT00910
CC    IRAY(51) THRU IRAY(NC*5) WILL BE FILLED WITH ONES                 WRT00920
C                                                                       WRT00930
      IWD = 5*NC                                                        WRT00940
      DO 10    I=1,IWD                                                  WRT00950
   10 RAY(50+I) = 1.0                                                   WRT00960
        NOBYTE = 800                                                    WRT00970
      CALL WRTREC(UNIT,NOBYTE,IRAY)                                     WRT00980
      RETURN                                                            WRT00990
C                                                                       WRT01000
CC    PACKING HEADER RECORD IN UNIVERSAL FORMAT                         WRT01010
C                                                                       WRT01020
   40 DO 50    I= 1,NC                                                  WRT01030
      II = FEAT(I)                                                      WRT01040
   50 ICHAN = ICHAN + 2**(32-II)                                        WRT01050
      NPRC = 0                                                          WRT01060
      PROFLG = 1                                                        WRT01070
      NBITS = 8                                                         WRT01080
      SVD = 1                                                           WRT01090
      PRSZ = 3060                                                       WRT01100
        PRSZHD = 3060                                                   WRT01110
      ANCLNG = 70                                                       WRT01120
      ANC = ANCLNG + 2                                                  WRT01130
      DOT = 0                                                           WRT01140
      COMWRD = 32                                                       WRT01150
      NOSPR = 1                                                         WRT01160
      NCPR = 0                                                          WRT01170
      IF (NSAMP .GT. 2998) WRITE(6,510)                                 WRT01180
  510 FORMAT(' NO. OF SAMPLES WAS RESET TO 2998')                       WRT01190
      IF (NSAMP .GT. 2998) NSAMP = 2998                                 WRT01200
      ILEN = (NC*NSAMP) + ANC                                           WRT01210
      ILENTH   = (NC*NSAMP + ANC    )/ 3000                             WRT01220
      IREMD    = MOD((NC*NSAMP + ANC    ),3000)                        WRT01230
      IF (ILENTH .EQ. 0) NCAR = NC                                      WRT01240
      IF (ILENTH .EQ. 0) NRPDS = 1                                      WRT01250
      IF (ILENTH .EQ. 0) GO TO 80                                       WRT01260
      DO 60    J=1,NC                                                   WRT01270
      IF((NSAMP*J+ANC    ) .GT. 3000)    NCAR = J-1                     WRT01280
      IF((NSAMP*J+ANC    ) .GT. 3000)    GO TO 70                       WRT01290
   60 CONTINUE                                                          WRT01300
   70 CONTINUE                                                          WRT01310
      NOCHAN = NC - NCAR                                                WRT01320
      DO 75    J=1,NOCHAN                                               WRT01330
      IF(NSAMP*J .GT. 2998)  NCPR = J-1                                 WRT01340
      IF(NSAMP*J .GT. 2998)  GO TO 76                                   WRT01350
   75 CONTINUE                                                          WRT01360
        NCPR = NOCHAN                                                   WRT01370
   76 CONTINUE                                                          WRT01380
      NRPDS = NOCHAN / NCPR + 1                                         WRT01390
      IF (MOD(NOCHAN,NCPR) .NE. 0)    NRPDS = NRPDS + 1                 WRT01400
   80 CONTINUE                                                          WRT01410
         IF (ILENTH.NE.0) GO TO 82                                      WRT01420
         PRSZ = (ILEN/180)*180                                          WRT01430
         IF (PRSZ.NE.ILEN) PRSZ =PRSZ + 180                             WRT01440
   82    CONTINUE                                                       WRT01450
C                                                                       WRT01460
CC    ZERO OUT PACRAY                                                   WRT01470
C                                                                       WRT01480
      DO 85  K=1,765                                                    WRT01490
   85 PACK(K) = 0                                                       WRT01500
      DO 100  K=1,18                                                    WRT01510
      NBYTES = NB(K)                                                    WRT01520
      DO 90  L=1,NBYTES                                                 WRT01530
      LOC = 4 + L - NBYTES + (K-1)*4                                    WRT01540
      BYTE = IBYTES(K)                                                  WRT01550
   90 PACRAY(BYTE+L-1) = VARIAB(LOC)                                    WRT01560
  100 CONTINUE                                                          WRT01570
       PACRAY(61) = VARIAB(73)                                          WRT01580
```

FILE: WRTHED FORTRAN A

```
      PACRAY(62) = VARIAB(74)                                          WRT01590
      PACRAY(63) = VARIAB(75)                                          WRT01600
      PACRAY(67) = VARIAB(79)                                          WRT01610
      PACRAY(68) = VARIAB(80)                                          WRT01620
       DO 110 L = 1,512                                                WRT01630
  110   PACRAY(111 + L) = VARIAB(111 + L)                              WRT01640
       DO 120 L = 1,16                                                 WRT01650
  120   PACRAY(2200 + L) = VARIAB(2200 + L)                            WRT01660
            DO 130 I = 1,4                                            WRT01670
            IPAT = 2254 + (I - 1)*8                                   WRT01680
  130       PACRAY(IPAT) = VARIAB(IPAT)                               WRT01690
C                                                                     WRT01700
C***  THE ABOVE THREE LINES IS AN AD HOC ADDITION FOR SOIL LINES      WRT01710
C***  ADDED OCT. 23,1978                                              WRT01720
C                                                                     WRT01730
      CALL WRTREC(UNIT,PRSZHD,PACRAY)                                 WRT01740
      RETURN                                                          WRT01750
      END                                                             WRT01760
```

19-76

423

FILE: WRTLN FORTRAN A

```
      SUBROUTINE WRTLN(/IDATA/,LSTLIN)                                  WRT00010
C                                                                       WRT00020
      IMPLICIT INTEGER (A-Z)                                            WRT00030
      COMMON /WRTAP/ICOUNT,FORMT,UNIT,VARBL(600),IREMD                  WRT00040
      LOGICAL*1 PACRAY(13500),ISCAN(4),IDATA(1),IRECNO(4)               WRT00050
      LOGICAL*1  ZERO(4)                                                WRT00060
      EQUIVALENCE (ICOUNT,ISCAN)                                        WRT00070
      EQUIVALENCE (IRECNO,IRECNO)                                       WRT00080
      EQUIVALENCE (ZERO,IZERO)                                          WRT00090
      EQUIVALENCE (VARBL(3),NC),(VARBL(7),PRSZ),                        WRT00100
     *            (VARBL(8),NCPR),(VARBL(9),NPPC),                     WRT00110
     *            (VARBL(10),NRPDS),(VARBL(11),ANCLNG),                WRT00120
     *            (VARBL(18),NSAMP),(VARBL(17),NCAR)                   WRT00130
      ICOUNT = ICOUNT + 1                                               WRT00140
      IZERO = 0                                                         WRT00150
      ANC = ANCLNG + 2                                                  WRT00160
      IF (FORMT .EQ. 1)     GO TO 140                                   WRT00170
C                                                                       WRT00180
C     WRITES PACKED DATA ON TAPE IN LARSYS II FORMAT                    WRT00190
C                                                                       WRT00200
  650 FORMAT(3I5)                                                       WRT00210
C     PACKING ONE SET OF DATA INTO ONE RECORD                           WRT00220
C                                                                       WRT00230
      NBITS = 8                                                         WRT00240
      ANCLNG = 4                                                        WRT00250
      NBYTES = NSAMP*NC                                                 WRT00260
      PACRAY(1) = ISCAN(3)                                              WRT00270
      PACRAY(2) = ISCAN(4)                                              WRT00280
      PACRAY(3) = ZERO(4)                                               WRT00290
      PACRAY(4) = ZERO(4)                                               WRT00300
      DO 120  I=1,NBYTES                                                WRT00310
  120 PACRAY(4+I) = IDATA(4*I)                                          WRT00320
      NBYTES=NBYTES+ANCLNG                                              WRT00330
      IDUM=(NBYTES/4)*4                                                 WRT00340
      IIDUM=NBYTES-IDUM                                                 WRT00350
      IF(IIDUM.NE.0) NBYTES=NBYTES+4-IIDUM                              WRT00360
      CALL WRTREC(UNIT,NBYTES,PACRAY)                                   WRT00370
      IF (LSTLIN .EQ. -1) ENDFILE UNIT                                  WRT00380
      RETURN                                                            WRT00390
C                                                                       WRT00400
C     WRITE PACKED DATA ON TAPE IN UNIVERSAL FORMAT                     WRT00410
C                                                                       WRT00420
  140 DO 150    I=1,72                                                  WRT00430
  150 PACRAY(I) = ZERO(4)                                               WRT00440
      POW     = 1                                                       WRT00450
      NR      = NRPDS - 1                                               WRT00460
      IF(NCPR.EQ.0) GO TO 155                                           WRT00470
      NCLR    = MOD((NC-NCAR),NCPR)                                     WRT00480
  155 WORD = 73                                                         WRT00490
C                                                                       WRT00500
C     PACKING ANCILLARY INFORMATION INTO PACRAY                         WRT00510
C                                                                       WRT00520
      RECNO = 1                                                         WRT00530
      PACRAY(2) = IRECNO(4)                                             WRT00540
      PACRAY(71) = ISCAN(3)                                             WRT00550
      PACRAY(72) = ISCAN(4)                                             WRT00560
C                                                                       WRT00570
C     DATA IS NOT PACKED WITH ANCILLARY RECORD                          WRT00580
C                                                                       WRT00590
      IF (NCAR .NE. 0)                GO TO 160                         WRT00600
      NBYTES = NSAMP * NC                                               WRT00610
      KA = 1                                                            WRT00620
      GO TO 210                                                         WRT00630
C                                                                       WRT00640
C     ALL DATA IS PACKED       ON ANCILLARY RECORD                      WRT00650
  160 IF (NCAR .NE. NC)               GO TO 170                         WRT00660
      NBYTES = IREMD - ANC                                              WRT00670
      KA = 2                                                            WRT00680
      GO TO 210                                                         WRT00690
C                                                                       WRT00700
C     PART OF DATA IS PACKED ON ANCILLARY RECORD                        WRT00710
C                                                                       WRT00720
  170 NBYTES = NCAR*NSAMP                                               WRT00730
      KA = 3                                                            WRT00740
      GO TO 210                                                         WRT00750
C                                                                       WRT00760
C     DATA IS PACKED ON MORE THAN ONE RECORD                            WRT00770
C                                                                       WRT00780
  180 ANC     = 2                                                       WRT00790
```

FILE: WRTLN   FORTRAN A

```
      WORD    = 3                                                                    WRT00800
      KA      = 4                                                                    WRT00810
      J = 0                                                                          WRT00820
  185 J = J + 1                                                                      WRT00830
      IF ( J.GT.NR) GO TO 200                                                        WRT00840
      WRITE(6,660)NR                                                                 WRT00850
  660 FORMAT(' NR',I5)                                                               WRT00860
      RECNO = RECNO + 1                                                              WRT00870
      PACRAY(2) = IRECNO(4)                                                          WRT00880
      NBYTES = NCPR * NSAMP                                                          WRT00890
      IF (NCLR .NE.  0   .AND. J .EQ. NR)  NBYTES = NCLR*NSAMP                       WRT00900
      GO TO 210                                                                      WRT00910
  200 CONTINUE                                                                       WRT00920
      IF (LSTLIN .EQ. -1) ENDFILE UNIT                                               WRT00930
      RETURN                                                                         WRT00940
  210 IF (NCAR .EQ. 0) GO TO 220                                                     WRT00950
      II = (ROW-1)*NSAMP*4                                                           WRT00960
      DO 215  I=1,NBYTES                                                             WRT00970
  215 PACRAY(WORD+I-1) = IDATA(4*I+II)                                               WRT00980
  220 CALL WRTREC(UNIT,PRSZ,PACRAY)                                                  WRT00990
      IF (KA .NE. 4)                ROW = NCAR + ROW                                 WRT01000
      IF (KA .EQ. 4)                ROW = ROW  + NCPR                                WRT01010
      GO TO (180,200,180,185),KA                                                     WRT01020
      END                                                                            WRT01030
```

10-78

```
FILE: WRTMTX   FORTRAN  A

      SUBROUTINE WRTMTX(MATICE,SIZE,BCD)                                WRT00010
C                                                                       WRT00020
      IMPLICIT INTEGER (A-H,O-Z)                                        WRT00030
C                                                                       WRT00040
      DIMENSION FORMAT(6)                                               WRT00050
      REAL MATICE(1)                                                    WRT00060
CI----------------------------------------------------------------------IWRT00070
CI----------------------------------------------------------------------IWRT00080
CI                                                                      IWRT00090
CI    CALL..    CALL WRTMTX(MATICE,SIZE,FREG,BCD,MAXFET)                IWRT00100
CI                                                                      IWRT00110
CI    ARGS..    MATICE - COVARIANCE MATRICE                             IWRT00120
CI              SIZE   - RANK OF 'MATRICE' ('DMATIC')                   IWRT00130
CI              FREQ   - FREQUENCY MATRIX                               IWRT00140
CI              BCD    - CONTAINS BCD PRECISION FOR PRINTOUT            IWRT00150
CI              MAXFET - NUMBER OF FEATURES PER LINE                    IWRT00160
CI                                                                      IWRT00170
CI    REQUIRES. NONE                                                    IWRT00180
CI                                                                      IWRT00190
CI    PURPOSE.. PRINTS THE SINGLE-PRECISION COVARIANCE MATRICES         IWRT00200
CI                                                                      IWRT00210
CI    RETURNS.. NO CHANGE                                               IWRT00220
CI                                                                      IWRT00230
CI----------------------------------------------------------------------IWRT00240
      CONTINUE                                                          WRT00250
CI                                                                      IWRT00260
CI    CALL..    CALL DWRTMX(MATICE,SIZE,FREG,BCD,MAXFET)                IWRT00270
CI                                                                      IWRT00280
CI    ARGS..    SEE ABOVE                                               IWRT00290
CI                                                                      IWRT00300
CI    PURPOSE.. PRINTS THE DOUBLE-PRECISION COVARIANCE MATRICES         IWRT00310
CI                                                                      IWRT00320
CI    RETURNS..  SEE ABOVE                                              IWRT00330
CI                                                                      IWRT00340
CI----------------------------------------------------------------------IWRT00350
CI----------------------------------------------------------------------IWRT00360
                                                                        WRT00370
CCCCC                                                                   WRT00380
      ----------------------------------------------------------------WRT00390
      ----------------------------------------------------------------WRT00400
CCCCC                                                                   WRT00410
      DATA FORMAT/'(1H0',',6X,',12F9',',   ',    ',')    '/           WRT00420
      DOUPRE = 0                                                        WRT00430
      GO TO10                                                           WRT00440
      ENTRY DWRTMX(DMATIC,SIZE,BCD)                                     WRT00450
      DOUBLE PRECISION DMATIC(1)                                        WRT00460
      DOUPRE=1                                                          WRT00470
 10   FORMAT(5)=BCD                                                     WRT00480
      DO 100 LOC=1,SIZE,12                                              WRT00490
      STOP = LOC+11                                                     WRT00500
      IF ( STOP .GT. SIZE) STOP = SIZE                                  WRT00510
      II =1                                                             WRT00520
      KINC = 1                                                          WRT00530
      DO 90 I=LOC,SIZE                                                  WRT00540
      K = I*(I+1)/2-II+1                                                WRT00550
      JK = K+KINC-1                                                     WRT00560
      IF(DOUPRE.EQ.0) WRITE(6,FORMAT) (MATICE(J),J=K,JK)                WRT00570
      IF (DOUPRE.EQ.1) WRITE(6,FORMAT)(DMATIC(J),J=K,JK)                WRT00580
      II = II+1                                                         WRT00590
   90 IF(KINC.LT.12.AND.KINC.LT.STOP)KINC=KINC+1                        WRT00600
      WRITE(6,1004)                                                     WRT00610
  100 CONTINUE                                                          WRT00620
 1004 FORMAT('0')                                                       WRT00630
      RETURN                                                            WRT00640
C                                                                       WRT00650
CCCCC                                                                   WRT00660
      ----------------------------------------------------------------WRT00670
C                                                                       WRT00680
       END                                                              WRT00690
```

FILE: WRTREC FORTRAN A

```
      SUBROUTINE WRTREC(UNIT,LENGTH,IBUF)                              WRT00010
      IMPLICIT INTEGER (A-Z)                                           WRT00020
C                                                                      WRT00030
CC    OUTPUTS A SCAN LINE OF DATA                                      WRT00040
C                                                                      WRT00050
      DIMENSION IBUF(3000)                                             WRT00060
      LENTH = LENGTH/4                                                 WRT00070
C     WRITE(6,200)UNIT,LENTH,IBUF(18)                                  WRT00080
      WRITE(UNIT,100)(IBUF(I),I=1,LENTH)                               WRT00090
C     WRITE(6,200)UNIT,LENGTH,IBUF(18)                                 WRT00100
  200 FORMAT(/3I6)                                                     WRT00110
  100 FORMAT(15(20A4))                                                 WRT00120
      RETURN                                                           WRT00130
      END                                                              WRT00140
```

~~1980~~

20. DAMRG PROCESSOR

FILE: DAMRG FORTRAN A

```
      SUBROUTINE DAMRG(ARRAY,TOP)                                          DAM00010
      IMPLICIT INTEGER (A-Z)                                               DAM00020
      REAL SUNCOR,DUM                                                      DAM00030
      LOGICAL*1 IDL(800),VARIAB(2400),LOGSUN(32)                           DAM00040
      EQUIVALENCE (ID,IDL),(VARBL,VARIAB),(SUNANG,LOGSUN)                  DAM00050
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,            DAM00060
     *HISFIL,HISKEY,TRFORM,ERRPTP,ERRKEY,MAPUNT,NOFILE,                    DAM00070
     *DRUMAD,DRMAOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL,                      DAM00080
     *NHSTUN,NHSTFI,SCTRUN,MAPFIL,                                         DAM00090
     *DOTUNT,DOTFIL,ACHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                    DAM00100
     *CRDUNT,PRTUNT,RANDIO                                                 DAM00110
      COMMON/TAPERD/IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,           DAM00120
     *LINC,ID(200),DSL,LBUF(30),JREC(30),IBYTE(30),NBUFS,FILENO,LINEND,    DAM00130
     *LININC,NSAMP,NOCHAN,FORMT                                            DAM00140
      COMMON/WRTAP/ICOUNT,DUMMY,UNIT,VARBL(600),IREMD                      DAM00150
      COMMON/ISOLNR/SUNANG(8),ISUNT,ISUNC,SMSTR,SMINC,LINSKP               DAM00160
      COMMON/MRGDAT/IMOPT,ISOPT,NUMFIL,IDATTP(6),IDATFL(6),                DAM00170
     *NOFEAT,NFEAT(6),FETVEC(30,6),ISUN(8,6),SUNCOR(30),                   DAM00180
     *FLDINF(6,6),NOSAMP,NOLINE,NSS(6),NACROS,NLINES(6),LINPTR(7),         DAM00190
     *LINES(600),FORMM                                                     DAM00200
      DIMENSION ARRAY(1)                                                   DAM00210
      CALL SETIA(ARRAY,TOP)                                                DAM00220
C                                                                          DAM00230
C***  MAJOR LOOP ON FILES                                                  DAM00240
C                                                                          DAM00250
      N1 = 0                                                               DAM00260
      IPPPP = 0                                                            DAM00270
      ICCT = 0                                                             DAM00280
      DO 700 I = 1,NUMFIL                                                  DAM00290
C                                                                          DAM00300
C***  CALL TAPE HEADER READ PROGRAM WITH UNIT AND FILE                     DAM00310
C                                                                          DAM00320
      IDATU = IDATTP(I)                                                    DAM00330
      IDATF = IDATFL(I) - 1                                                DAM00340
      CALL TAPHDR(IDATU,IDATF)                                             DAM00350
C                                                                          DAM00360
C***  NUMBER OF CHANNELS FOR THIS FILE                                     DAM00370
C                                                                          DAM00380
      NF = NFEAT(I)                                                        DAM00390
      CALL FLDINT(FLDINF(1,I),FETVEC(1,I),NF)                              DAM00400
C                                                                          DAM00410
C***  SET FEATURE COUNTER                                                  DAM00420
C                                                                          DAM00430
      IF(I.GT.1) N1 = N1 + NFEAT(I - 1)                                    DAM00440
C                                                                          DAM00450
C***  SET SCALAR FIELD DESCRIPTION FOR THIS FILE                           DAM00460
C                                                                          DAM00470
      SAMSTR = FLDINF(4,I)                                                 DAM00480
      SAMINC = FLDINF(6,I)                                                 DAM00490
      SAMEND = FLDINF(5,I)                                                 DAM00500
      LINSTR = FLDINF(1,I)                                                 DAM00510
      LININC = FLDINF(3,I)                                                 DAM00520
      LINEND = FLDINF(2,I)                                                 DAM00530
C                                                                          DAM00540
C***  NO. SAMPLES/LINE FOR FILE I COMPUTED IN FLDINT                       DAM00550
C                                                                          DAM00560
      NS = NSAMP                                                           DAM00570
C                                                                          DAM00580
C***  TOTAL NUMBER OF RADIANCE VALUES PER LINE                             DAM00590
      NV = NS * NF                                                         DAM00600
C                                                                          DAM00610
C***    WRITE FIELD INFORMATION FOR FILE I                                 DAM00620
C                                                                          DAM00630
      WRITE (PRTUNT,1060) I                                                DAM00640
 1060 FORMAT(10    INPUT FIELD DESCRIPTION FOR FILE',I8)                   DAM00650
      WRITE (PRTUNT,1070)                                                  DAM00660
 1070 FORMAT(10    START LINE   END LINE    LINE INC  START PIXEL  END    DAM00670
     *PIXEL PIXEL INC')                                                    DAM00680
      WRITE (PRTUNT,1080) (FLDINF(J,I), J=1,6)                             DAM00690
 1080 FORMAT(10  ',6(4X,I4,4X))                                            DAM00700
C                                                                          DAM00710
C***  STORE SUN ANGLES                                                     DAM00720
C                                                                          DAM00730
      IF(ISUNT.NE.0)GO TO 610                                              DAM00740
      IF(ISUNC.EQ.0)GO TO 620                                              DAM00750
C                                                                          DAM00760
C***  SUN ANGLES FROM CARDS                                                DAM00770
C                                                                          DAM00780
      DO 600 J = 1,8                                                       DAM00790
```

FILE: DAMRG FORTRAN A

```
      ID(16 * J) = ISUN(J,I)                                                    DAM00800
      SUNANG(J) = ISUN(J,I)                                                     DAM00810
  600 CONTINUE                                                                  DAM00820
      GO TO 620                                                                 DAM00830
C                                                                               DAM00840
C***  SUN ANGLES FROM TAPE HEADER                                               DAM00850
C                                                                               DAM00860
  610 DO 615 J = 1,8                                                            DAM00870
      ISUN(J,I) = SUNANG(J)                                                     DAM00880
  615 CONTINUE                                                                  DAM00890
      CALL SUNFAC(SUNCOR,SUNANG,FETVEC(1,I),NF,ISUNT,ISUNC)                     DAM00900
  620    CONTINUE                                                               DAM00910
       IF (FORMM.NE.1) GO TO 660                                                DAM00920
       WRITE(PRTUNT,1000)I                                                      DAM00930
 1000  FORMAT('0DATA FOR INPUT FILE',I4)                                        DAM00940
       WRITE(PRTUNT,1010)IDL(100),IDL(104),IDL(108),IDL(111),IDL(112)           DAM00950
 1010  FORMAT('0INPUT FILE DATE AND SITE',5(3X,Z2))                             DAM00960
C                                                                               DAM00970
C***  LOAD VARIAB WITH EXTRA HEADER INFORMATION                                 DAM00980
C                                                                               DAM00990
C***  LOAD DATE AND SITE FROM FIRST FILE                                        DAM01000
C                                                                               DAM01010
      IF(I.NE.1)GO TO 625                                                       DAM01020
      VARIAB(73) = IDL(100)                                                     DAM01030
      VARIAB(74) = IDL(104)                                                     DAM01040
      VARIAB(75) = IDL(108)                                                     DAM01050
      VARIAB(79) = IDL(111)                                                     DAM01060
      VARIAB(80) = IDL(112)                                                     DAM01070
C                                                                               DAM01080
C***  FOR PURPOSES OF UNIVERSAL HEADER WRITE LOAD VARIAB WITH SUN ANGLES        DAM01090
C***  AND GAINS AND BIASES ONLY IF CHANNEL MERGE OPTION                         DAM01100
C                                                                               DAM01110
  625 IF(IMOPT.NE.1)GO TO 660                                                   DAM01120
      DO 650 J = 1,NF                                                           DAM01130
      IDUM = (FETVEC(J,I) - 1) * 2                                              DAM01140
      I1 = 112 + IDUM                                                           DAM01150
      I2 = 112 + N1 * 2 + (J - 1) * 2                                           DAM01160
      VARIAB(I2) = IDL(I1 + 3)                                                  DAM01170
      VARIAB(I2 + 1) = IDL(I1 + 4)                                              DAM01180
      I1 = 240 + IDUM                                                           DAM01190
      I2 = 240 + N1 * 2 + (J - 1) * 2                                           DAM01200
      VARIAB(I2) = IDL(I1 + 3)                                                  DAM01210
      VARIAB(I2 + 1) = IDL(I1 + 4)                                              DAM01220
      I1 = 368 + IDUM                                                           DAM01230
      I2 = 368 + N1 * 2 + (J - 1) * 2                                           DAM01240
      VARIAB(I2) = IDL(I1 + 3)                                                  DAM01250
      VARIAB(I2 + 1) = IDL(I1 + 4)                                              DAM01260
      I1 = 496 + IDUM                                                           DAM01270
      I2 = 496 + N1 * 2 +(J - 1) * 2                                            DAM01280
      VARIAB(I2) = IDL(I1 + 3)                                                  DAM01290
      VARIAB(I2 + 1) = IDL(I1 + 4)                                              DAM01300
  650 CONTINUE                                                                  DAM01310
      KS = 0                                                                    DAM01320
      DO 655 J = 1,NF                                                           DAM01330
      IDUM = FETVEC(J,I)                                                        DAM01340
      IDUM = (IDUM - 1) / NCHPAS                                                DAM01350
      IF(ISUNT.EQ.0.AND.J.EQ.1) KS = IDUM                                       DAM01360
      I1 = (IDUM - KS) * 4 + 3                                                  DAM01370
      I2 = 2201 + (N1 + J - 1) * 2                                              DAM01380
      VARIAB(I2) = LOGSUN(I1)                                                   DAM01390
      VARIAB(I2 + 1) = LOGSUN(I1 + 1)                                           DAM01400
  655 CONTINUE                                                                  DAM01410
  660 CONTINUE                                                                  DAM01420
C                                                                               DAM01430
C***    THE NEXT LINE WAS ADDED OCT. 23,1978 AS AN AD HOC ADDITION              DAM01440
C***    TO ADD SOIL LINES TO THE UNIVERSAL HEADER                               DAM01450
C                                                                               DAM01460
          VARIAB(2246 + 8*I)  =  IDL(640)                                       DAM01470
C                                                                               DAM01480
C                                                                               DAM01490
C***  INITIALIZATION FOR LINE EXTRACTION PARAMETERS NEEDED FOR                  DAM01500
C***  SPATIAL MERGE                                                             DAM01510
C                                                                               DAM01520
      LOC = (I - 1) / NACROS                                                    DAM01530
       NS = 0                                                                   DAM01540
       IF (LOC.EQ.0) GO TO 666                                                  DAM01550
       DO 661 J = 1,LOC                                                         DAM01560
  661  NS = NS + NLINES(J)                                                      DAM01570
  664  CONTINUE                                                                 DAM01580
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: DAMRG FORTRAN A

```
      LREM = (I - 1) - LOC * NACROS                                    DAM01590
      N2 = 0                                                           DAM01600
      IF(LREM.EQ.0)GO TO 663                                           DAM01610
      DO 662 J = 1,LREM                                                DAM01620
      N2 = N2 + NSS(J)                                                 DAM01630
  662 CONTINUE                                                         DAM01640
  663 ICT = 0                                                          DAM01650
C                                                                      DAM01660
C***  PARAMETERS NEEDED IF PSEUDO MERGE OPTION                         DAM01670
C                                                                      DAM01680
      LPTR = LINPTR(I)                                                 DAM01690
      NL = NLINES(I)                                                   DAM01700
      NLM = NL + LPTR - 1                                              DAM01710
C                                                                      DAM01720
C***  EXTRACT FIELD FOR THIS FILE LINE BY LINE                         DAM01730
C                                                                      DAM01740
      DO 690 II = LINSTR,LINEND,LININC                                 DAM01750
      ICT = ICT + 1                                                    DAM01760
       CALL LINERD(ARRAY(1),ENDTAP)                                    DAM01770
       IF (ENDTAP.EQ.-1) CALL CMERR                                    DAM01780
      IF(IMOPT.NE.3)GO TO 670                                          DAM01790
C                                                                      DAM01800
C***  LOOK FOR LINE MATCH IF PSEUDO MERGE                              DAM01810
C                                                                      DAM01820
      DO 665 J = LPTR,NLM                                              DAM01830
      IF(II.EQ.LINES(J))GO TO 670                                      DAM01840
  665 CONTINUE                                                         DAM01850
      GO TO 690                                                        DAM01860
  670 CONTINUE                                                         DAM01870
          ICCT = ICCT + 1                                              DAM01880
C                                                                      DAM01890
      IF(IPPPP.EQ.1)WRITE(PRTUNT,1020)(ARRAY(K),K=1,NV)                DAM01900
 1020 FORMAT(/1017)                                                    DAM01910
      IF(IMOPT.NE.1)GO TO 675                                          DAM01920
C                                                                      DAM01930
C***  CHANNEL MERGE MODE WRITE NV VALUES TO DIRECT ACCESS FILE         DAM01940
C                                                                      DAM01950
      IF(ISOPT.EQ.0)GO TO 672                                          DAM01960
C                                                                      DAM01970
C***  DO SUN ANGLE CORRECTION                                          DAM01980
C                                                                      DAM01990
      DO 671 J = 1,NF                                                  DAM02000
      DO 671 JJ = 1,NS                                                 DAM02010
      ITEMP = (JJ + (J - 1) * NS)                                      DAM02020
      DUM = SUNCOR(J) * FLOAT(ARRAY(ITEMP))                            DAM02030
      ARRAY(ITEMP) = IFIX(DUM)                                         DAM02040
  671 CONTINUE                                                         DAM02050
  672     ADDRES = DRUMAD + N1*NS + (ICT - 1)*NS*NOFEAT                DAM02060
      CALL RWRITE(ADDRES,ARRAY(1),NV,STATUS)                           DAM02070
      GO TO 690                                                        DAM02080
  675 IF(IMOPT.NE.2)GO TO 680                                          DAM02090
C                                                                      DAM02100
C***  SPATIAL MERGE MODE WRITE NSS(I)*NF VALUES TO DIRECT ACCESS FILE  DAM02110
C                                                                      DAM02120
      IF(ISOPT.EQ.0)GO TO 677                                          DAM02130
      DO 676 J = 1,NF                                                  DAM02140
      DO 676 JJ = 1,NS                                                 DAM02150
      ITEMP = (JJ + (J - 1) * NS)                                      DAM02160
      DUM = SUNCOR(J) * FLOAT(ARRAY(ITEMP))                            DAM02170
      ARRAY(ITEMP) = IFIX(DUM)                                         DAM02180
  676 CONTINUE                                                         DAM02190
  677     N4 = NSS(I)                                                  DAM02200
          DO 679 J=1,NF                                                DAM02210
          ADDRES =DRUMAD +(NS * ICT-1)*NOSAMP*NF+NOSAMP*(J-1)*N2       DAM02220
          ITEMP = 1 + (J-1)*N4                                         DAM02230
          CALL RWRITE(ADDRES,ARRAY(ITEMP),N4,STATUS)                   DAM02240
  679     CONTINUE                                                     DAM02250
      GO TO 690                                                        DAM02260
C                                                                      DAM02270
C***  PSEUDO MERGE OPTION                                              DAM02280
C                                                                      DAM02290
  680 IF(ISOPT.EQ.0)GO TO 682                                          DAM02300
      DO 681 J = 1,NF                                                  DAM02310
      DO 681 JJ = 1,NS                                                 DAM02320
      ITEMP = (JJ + (J - 1) * NS)                                      DAM02330
      DUM = SUNCOR(J) * FLOAT(ARRAY(ITEMP))                            DAM02340
      ARRAY(ITEMP) = IFIX(DUM)                                         DAM02350
  681 CONTINUE                                                         DAM02360
  682 ADDRES = DRUMAD + (ICCT - 1) * NV                                DAM02370
```

```
FILE: DAMRG    FORTRAN  A

      CALL RWRITE(ADDRES,ARRAY(1),NV,STATUS)                              DAM02380
  690 CONTINUE                                                            DAM02390
C                                                                         DAM02400
C***  LINE LOOP COMPLETE                                                  DAM02410
C                                                                         DAM02420
  700 CONTINUE                                                            DAM02430
C                                                                         DAM02440
C***  LOOP FOR FILE I COMPLETE                                            DAM02450
C                                                                         DAM02460
C***  WRITE OUTPUT FILE .                                                 DAM02470
C                                                                         DAM02480
      DATFI = DATFIL - 1                                                  DAM02490
C                                                                         DAM02500
C***  POSITION OUTPUT FILE                                                DAM02510
C                                                                         DAM02520
      CALL FSFMFL(DATAPE,DATFI,ISTAT)                                     DAM02530
C                                                                         DAM02540
C***  SET OUTPUT CHANNELS 1,2,....,NOFEAT                                 DAM02550
C                                                                         DAM02560
      DO 800 I = 1,NOFEAT                                                 DAM02570
      FETVEC(I,1) = I                                                     DAM02580
  800 CONTINUE                                                            DAM02590
C                                                                         DAM02600
C***  WRITE HEADER OF OUTPUT FILE                                         DAM02610
C                                                                         DAM02620
      CALL WRTHED(NOFEAT,FETVEC(1,1),NOSAMP,FORMM,DATAPE)                 DAM02630
C                                                                         DAM02640
C***  EXTRACT SCAN LINES ONE AT A TIME WRITE TO OUTPUT FILE               DAM02650
C                                                                         DAM02660
      LSTLIN = 0                                                          DAM02670
      NV = NOSAMP * NOFEAT                                                DAM02680
      DO 850 I = 1,NOLINE                                                 DAM02690
      IF(I.EQ.NOLINE)LSTLIN = - 1                                         DAM02700
      ADDRES = DRUMAD + (I - 1) * NV                                      DAM02710
      CALL RREAD(ADDRES,ARRAY(1),NV,ISTAT)                                DAM02720
      CALL WRTLN(ARRAY(1),LSTLIN)                                         DAM02730
      IF(IPPPP.EQ.1)WRITE(PRTUNT,1030)I                                   DAM02740
 1030 FORMAT(' OUTPUT LINE',I6)                                           DAM02750
      IF(IPPPP.EQ.1)WRITE(PRTUNT,1020)(ARRAY(K),K=1,NV)                   DAM02760
  850 CONTINUE                                                            DAM02770
C                                                                         DAM02780
C***  RETURN TO MONTOR                                                    DAM02790
C                                                                         DAM02800
C                                                                         DAM02810
C***  OUTPUT FILE COMPLETED                                               DAM02820
C                                                                         DAM02830
      RETURN                                                              DAM02840
      END                                                                 DAM02850
```

FILE: SET10 FORTRAN A

```
      SUBROUTINE SET10(ARRAY,TOP)                                                SET00010
      IMPLICIT INTEGER (A-Z)                                                     SET00020
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,                  SET00030
     *HISFIL,HISKEY,TMFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,                          SET00040
     *DRIMAP,DRM,DS,BAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL,                            SET00050
     *RMSTUN,RMSTFI,SCTRUN,MAPFIL,                                               SET00060
     *DOTUNT,DOTFIL,NCMPAS,TRNSFL,RMTRFL,HISTFL,PCHUNT,                          SET00070
     *CRDUNT,PRTUNT,RAND10                                                       SET00080
      COMMON/TAPERD/IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,                 SET00090
     *LINC,ID(200),DSL,LBUF(30),JREC(30),IBYTE(30),NBUFS,FILENO,LINENO,         SET00100
     *LININC,NSAMP,NOCHAN,FORMT                                                 SET00110
      COMMON/WRTAP/ICOUNT,DUMMY,UNIT,VARBL(600),IREMD                           SET00120
      COMMON/ISUNR/SUNANG(8),ISUNT,ISUNC,SMSTR,SMINC,LINSKP                     SET00130
      COMMON/MRGDAT/IMOPT,ISOPT,NUMFIL,IDATTP(6),IDATFL(6),                     SET00140
     *NOFEAT,NFEAT(4),FETVEC(30,4),ISUN(8,6),SUNCOR(30),                        SET00150
     *FLDINF(6,6),NOSAMP,NOLINE,NSS(6),NACROS,NLINES(6),LINPTR(7),              SET00160
     *LINES(600),FORMM                                                          SET00170
      DIMENSION ARRAY(1)                                                        SET00180
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15),EQUVEC(2),SLASH(2)         SET00190
      EQUIVALENCE (HED1,HEAD(4)),(DATE(1),HEAD(22)),(HED2(1),HEAD(30)),         SET00200
     *(COMENT(1),HEAD(48))                                                      SET00210
      DATA EQUVEC/1,'='/,SLASH/1,'/'/                                           SET00220
      DATA IRCD/'I'/,ORCD/'O'/,BLANK/' '/,URCD/'U'/,                            SET00230
     *LRCD/'L'/,CRCD/'C'/,SRCD/'S'/,PRCD/'P'/,                                  SET00240
     *ARCD/'A'/,TRCD/'T'/,FRCD/'F'/                                             SET00250
      DIMENSION INVEC(13)                                                       SET00260
      DATA INVEC/'NCPA','FORM','DATE','HED1','HED2','NLIN','OPTI',              SET00270
     *'NACR','SUNA','DATA','LINE','CHAN','*END'/                                SET00280
      DIMENSION CARD(52),ACARD(20)                                              SET00290
C                                                                               SET00300
CC***  CHANNEL CARD COUNTER                                                     SET00310
C                                                                               SET00320
      ICHNCT = 0                                                                SET00330
C                                                                               SET00340
CC***  SUN ANGLE COUNTER - CONTROL CARDS                                        SET00350
C                                                                               SET00360
      ISUNCT = 0                                                                SET00370
C                                                                               SET00380
CC***  INPUT DATA TAPE COUNTER                                                  SET00390
C                                                                               SET00400
      NUMFIL = 0                                                                SET00410
      NOLINE = 0                                                                SET00420
C                                                                               SET00430
CC***  DEFAULT SETTINGS                                                         SET00440
C                                                                               SET00450
      IMOPT = 1                                                                 SET00460
      ISUNT = 0                                                                 SET00470
      ISOPT = 0                                                                 SET00480
      ISUNC = 0                                                                 SET00490
      NACROS = 1                                                                SET00500
      IDATTP(1) = 22                                                            SET00510
      IDATFL(1) = 1                                                             SET00520
      NOFEAT = 4                                                                SET00530
      DO 10 I = 1,6                                                             SET00540
      DO 10 J = 1,4                                                             SET00550
      NFEAT(I) = 4                                                              SET00560
      FETVEC(J,I) = J                                                           SET00570
   10 CONTINUE                                                                  SET00580
      DO 20 I = 1,6                                                             SET00590
      DO 20 J = 1,8                                                             SET00600
      ISUN(J,I) = 50                                                            SET00610
   20 CONTINUE                                                                  SET00620
      FORMM = 1                                                                 SET00630
      PRUNIT = 30                                                               SET00640
   80 REWIND PRUNIT                                                             SET00650
      READ(CRDUNT,1000)(ACARD(I),I = 1,20)                                      SET00660
 1000 FORMAT(20A4)                                                              SET00670
      WRITE(PRUNIT,1000)(ACARD(I),I = 1,20)                                     SET00680
      REWIND PRUNIT                                                             SET00690
      READ(PRUNIT,1010)CODE,CARD                                                SET00700
 1010 FORMAT(A4,5X,62A1)                                                        SET00710
      REWIND PRUNIT                                                             SET00720
      WRITE(PRTUNT,1020)CODE,CARD                                               SET00730
 1020 FORMAT(7X,A4,5X,62A1)                                                     SET00740
      COL = 0                                                                   SET00750
      ISTART = 0                                                                SET00760
      DO 90 I = 1,13                                                            SET00770
      IF(CODE.EQ.INVEC(I))GO TO(100,150,200,250,300,350,400,450,500,            SET00780
     *550,600,650,700),I                                                        SET00790
```

```
FILE: SET18    FORTRAN  A

   90 CONTINUE                                                          SET00800
      WRITE(PRTUNT,1030)CODE,CARD                                       SET00810
 1030 FORMAT(' INVALID CARD - IGNORED'/TS,A4,6X,62A1)                   SET00820
      GO TO 80                                                          SET00830
C                                                                       SET00840
C***  NUMBER OF CHANNELS PER PASS CARD IMAGE                            SET00850
C                                                                       SET00860
  100 J = NXTCHR(CARD,COL)                                              SET00870
      IF(J.EQ.BLANK)GO TO 80                                            SET00880
      J = NUMBER(CARD,COL,NCHPAS,ISTART)                                SET00890
      GO TO 80                                                          SET00900
C                                                                       SET00910
C***  FORMAT CARD IMAGE - OUTPUT FILE                                   SET00920
C                                                                       SET00930
  150 J = NXTCHR(CARD,COL)                                              SET00940
      IF(J.EQ.LRCD) FORMM = 2                                           SET00950
      GO TO 80                                                          SET00960
C                                                                       SET00970
C***  DATE CARD                                                         SET00980
C                                                                       SET00990
  200 READ(RRUNIT,1040)DATE                                             SET01000
 1040 FORMAT(10X,3A4)                                                   SET01010
      REWIND RRUNIT                                                     SET01020
      GO TO 80                                                          SET01030
C                                                                       SET01040
C***  MED1 CARD                                                         SET01050
C                                                                       SET01060
  250 READ(RRUNIT,1050)MED1                                             SET01070
 1050 FORMAT(10X,15A4)                                                  SET01080
      REWIND RRUNIT                                                     SET01090
      GO TO 80                                                          SET01100
C                                                                       SET01110
C***  MED2 CARD                                                         SET01120
C                                                                       SET01130
  300 READ(RRUNIT,1050)MED2                                             SET01140
      REWIND RRUNIT                                                     SET01150
      GO TO 80                                                          SET01160
C                                                                       SET01170
C***  NLINES (6) CARD FOR PSEUDO OPTION                                 SET01180
C                                                                       SET01190
  350 J = NUMBER(CARD,COL,ARRAY(1),ISTART)                              SET01200
      IF(J.GT.6) J = 6                                                  SET01210
      DO 360 JJ = 1,J                                                   SET01220
  360 NLINES(JJ) = ARRAY(JJ)                                            SET01230
      GO TO 80                                                          SET01240
C                                                                       SET01250
C***  OPTION CARD IMAGE                                                 SET01260
C                                                                       SET01270
  400 J = NXTCHR(CARD,COL)                                              SET01280
      IF(J.EQ.CRCD) IMOPT = 1                                           SET01290
      IF(J.EQ.SRCD) IMOPT = 2                                           SET01300
      IF(J.EQ.PRCD) IMOPT = 3                                           SET01310
      IF(J.EQ.ARCD) ISOPT = 1                                           SET01320
      GO TO 80                                                          SET01330
C                                                                       SET01340
C***  SPATIAL OPTION ... NUMBER FIELDS TO BE JOINED ACROSS              SET01350
C                                                                       SET01360
  450 J = NUMBER(CARD,COL,NACROS,ISTART)                                SET01370
      GO TO 80                                                          SET01380
C                                                                       SET01390
C***  SUN ANGLE CARD IMAGES                                             SET01400
C                                                                       SET01410
  500 J = NXTCHR(CARD,COL)                                              SET01420
      IF(J.NE.TRCD)GO TO 510                                            SET01430
      ISUNT = 1                                                         SET01440
      GO TO 80                                                          SET01450
  510 ISUNCT = ISUNCT + 1                                               SET01460
      COL = 0                                                           SET01470
      ISUNC = 1                                                         SET01480
      J = NUMBER(CARD,COL,ARRAY(1),ISTART)                              SET01490
      DO 520 JJ = 1,J                                                   SET01500
      ISUN(JJ,ISUNCT) = ARRAY(JJ)                                       SET01510
  520 CONTINUE                                                          SET01520
      GO TO 80                                                          SET01530
C                                                                       SET01540
C***  DATA TAPE CARD IMAGES                                             SET01550
C                                                                       SET01560
  550 J = NXTCHR(CARD,COL)                                              SET01570
      IF(J.EQ.IMCD)GO TO 570                                            SET01580
```

```
FILE: SET1A    FORTRAN  A

      IF(J.NE.ORCD)GO TO 595                                          SET01590
C                                                                     SET01600
C***  OUTPUT FILE                                                     SET01610
C                                                                     SET01620
      J = FIND12(CARD,COL,SLASH)                                      SET01630
      IF(J.NE.2)GO TO 595                                             SET01640
      J = NXTCHR(CARD,COL)                                            SET01650
      IF(J.EQ.FRCD)GO TO 555                                          SET01660
C                                                                     SET01670
C***  UNIT NUMBER OF OUTPUT FILE, THEN FILE                           SET01680
C                                                                     SET01690
      J = FIND12(CARD,COL,EQUVEC)                                     SET01700
      IF(J.NE.2)GO TO 595                                             SET01710
      J = NUMBER(CARD,COL,DATAPE,ISTART)                              SET01720
      J = FIND12(CARD,COL,EQUVEC)                                     SET01730
      IF(J.NE.2)GO TO 595                                             SET01740
      ISTART = 0                                                      SET01750
      J = NUMBER(CARD,COL,DATFIL,ISTART)                              SET01760
      GO TO 80                                                        SET01770
C                                                                     SET01780
C***  FILE NUMBER, THEN UNIT                                          SET01790
C                                                                     SET01800
  555 J = FIND12(CARD,COL,EQUVEC)                                     SET01810
      IF(J.NE.2)GO TO 595                                             SET01820
      J = NUMBER(CARD,COL,DATFIL,ISTART)                              SET01830
      J = FIND12(CARD,COL,EQUVEC)                                     SET01840
      IF(J.NE.2)GO TO 595                                             SET01850
      ISTART = 0                                                      SET01860
      J = NUMBER(CARD,COL,DATAPE,ISTART)                              SET01870
      GO TO 80                                                        SET01880
C                                                                     SET01890
C***  INPUT FILES                                                     SET01900
C                                                                     SET01910
  570 NUMFIL = NUMFIL + 1                                             SET01920
      IF(NUMFIL.GT.6)GO TO 590                                        SET01930
      J = FIND12(CARD,COL,SLASH)                                      SET01940
      IF(J.NE.2)GO TO 590                                             SET01950
      J = NXTCHR(CARD,COL)                                            SET01960
      IF(J.EQ.FRCD)GO TO 575                                          SET01970
C                                                                     SET01980
C***  UNIT NUMBER OF INPUT FILE, THEN FILE NUMBER                     SET01990
C                                                                     SET02000
      J = FIND12(CARD,COL,EQUVEC)                                     SET02010
      IF(J.NE.2)GO TO 590                                             SET02020
      J = NUMBER(CARD,COL,IDATTP(NUMFIL),ISTART)                      SET02030
      J = FIND12(CARD,COL,EQUVEC)                                     SET02040
      IF(J.NE.2)GO TO 590                                             SET02050
      ISTART = 0                                                      SET02060
      J = NUMBER(CARD,COL,IDATFL(NUMFIL),ISTART)                      SET02070
      GO TO 80                                                        SET02080
  575 J = FIND12(CARD,COL,EQUVEC)                                     SET02090
      IF(J.NE.2)GO TO 590                                             SET02100
      J = NUMBER(CARD,COL,IDATFL(NUMFIL),ISTART)                      SET02110
      J = FIND12(CARD,COL,EQUVEC)                                     SET02120
      IF(J.NE.2)GO TO 590                                             SET02130
      ISTART = 0                                                      SET02140
      J = NUMBER(CARD,COL,IDATTP(NUMFIL),ISTART)                      SET02150
      GO TO 80                                                        SET02160
C                                                                     SET02170
C***  ERRORS                                                          SET02180
C                                                                     SET02190
  590 NUMFIL = NUMFIL - 1                                             SET02200
  595 WRITE(PRTUNT,1060)                                              SET02210
 1060 FORMAT(' ERROR ON ABOVE INPUT CONTROL CARD')                    SET02220
      GO TO 80                                                        SET02230
C                                                                     SET02240
C***  LINES IN PSEUDO OPTION                                          SET02250
C                                                                     SET02260
  600 NOLINE = NUMBER(CARD,COL,LINES,NOLINE)                          SET02270
      GO TO 80                                                        SET02280
C                                                                     SET02290
C***  CHANNELS CARD                                                   SET02300
C                                                                     SET02310
  650 ICHNCT = ICHNCT + 1                                             SET02320
      IF(ICHNCT.GT.6)GO TO 80                                         SET02330
      J = NUMBER(CARD,COL,ARRAY(1),ISTART)                            SET02340
      IF(J.GT.30)J = 30                                               SET02350
      DO 660 JJ = 1,J                                                 SET02360
      FETVEC(JJ,ICHNCT) = ARRAY(JJ)                                   SET02370
```

```
FILE: SET18     FORTRAN  A

  660 CONTINUE                                                          SET02380
      NFEAT(ICHNCT) = J                                                 SET02390
      GO TO 80                                                          SET02400
C                                                                       SET02410
C***  *END CARD                                                         SET02420
C                                                                       SET02430
  700 LIMIT = 1                                                         SET02440
      IF(IMOPT.EQ.2)LIMIT = NUMFIL                                      SET02450
      DO 710 J = 1,LIMIT                                                SET02460
      ARRAY(2) = 0                                                      SET02470
      JJ = LAREAD(ARRAY(1),ARRAY(3),FLDINF(1,J),ARRAY(2))               SET02480
      JSAVE = J                                                         SET02490
      IF(JJ.NE.1)GO TO 750                                              SET02500
  710 CONTINUE                                                          SET02510
      JJ = LAREAD(ARRAY(1),ARRAY(3),ARRAY(4),ARRAY(2))                  SET02520
      IF(JJ.NE.0)GO TO 750                                              SET02530
      GO TO 770                                                         SET02540
C                                                                       SET02550
C***  ERROR IN FIELD CARDS                                              SET02560
C                                                                       SET02570
  750 WRITE(PRTUNT,1070)JSAVE                                           SET02580
 1070 FORMAT(' ERROR IN FIELD',I10,' OR SEND CARD MISSING')             SET02590
      CALL CMERR                                                        SET02600
C                                                                       SET02610
C***  PROCESS INFORMATION                                               SET02620
C                                                                       SET02630
  770 CONTINUE                                                          SET02640
      IF(ICHNCT.EQ.NUMFIL)GO TO 775                                     SET02650
      WRITE(PRTUNT,1080)ICHNCT,NUMFIL                                   SET02660
 1080 FORMAT(' NUMBER OF CHANNEL CARDS',I5,' DOES NOT MATCH NUMBER OF ',SET02670
     *'DATA FILES',I5)                                                  SET02680
      ICHNCT = NUMFIL                                                   SET02690
  775 IF(ISUNC.EQ.1.AND.ISUNCT.NE.NUMFIL)GO TO 780                      SET02700
      GO TO 785                                                         SET02710
  780 WRITE(PRTUNT,1090)ISUNCT,NUMFIL                                   SET02720
 1090 FORMAT(' NUMBER OF SUN ANGLE CARDS',I5,' DOES NOT MATCH NUMBER',  SET02730
     *'OF DATA FILES',I5)                                               SET02740
      ISUNCT = NUMFIL                                                   SET02750
  785 IF(IMOPT.EQ.1)GO TO 795                                           SET02760
C                                                                       SET02770
C***  CHECK NO. CHANNELS EQUAL ON SPATIAL OR PSEUDO OPTION              SET02780
C                                                                       SET02790
      NOFEAT = NFEAT(1)                                                 SET02800
        IF (NUMFIL.EQ.1) GO TO 795                                      SET02810
      DO 790 J = 2,NUMFIL                                               SET02820
      IF(NFEAT(J).EQ.NFEAT(1))GO TO 790                                 SET02830
      WRITE(PRTUNT,1100)J,NFEAT(J),NFEAT(1)                             SET02840
 1100 FORMAT(' NUMBER OF FEATURES OF',I5,' FILE',I5,                    SET02850
     *' IS NOT EQUAL TO FIRST',I5)                                      SET02860
      NFEAT(J) = NFEAT(1)                                               SET02870
  790 CONTINUE                                                          SET02880
      GO TO 806                                                         SET02890
C                                                                       SET02900
C***        FEATURES IN CHANNEL MERGE                                   SET02910
C                                                                       SET02920
  795 IDUM = 0                                                          SET02930
      DO 800 I = 1,NUMFIL                                               SET02940
      IDUM = IDUM + NFEAT(I)                                            SET02950
  800 CONTINUE                                                          SET02960
      IF(IDUM.LE.30)GO TO 805                                           SET02970
      WRITE(PRTUNT,1105)(NFEAT(I),I = 1,NUMFIL)                         SET02980
 1105 FORMAT(' FEATURES ADD UP TO A NUMBER GREATER THAN 30',6I5,        SET02990
     *' EXITING')                                                       SET03000
      CALL CMERR                                                        SET03010
  805 NOFEAT = IDUM                                                     SET03020
C                                                                       SET03030
C***  SET NOSAMP AND NOLINE                                             SET03040
C                                                                       SET03050
  806     IF (IMOPT.NE.2) NACROS = 1                                    SET03060
      NOSAMP = 0                                                        SET03070
      DO 810 J = 1,NACROS                                               SET03080
      NSS(J) = (FLDINF(5,J) - FLDINF(4,J))/FLDINF(6,J) + 1              SET03090
      NOSAMP = NOSAMP + NSS(J)                                          SET03100
  810 CONTINUE                                                          SET03110
      NDOWN = NUMFIL / NACROS                                           SET03120
      IF(IMOPT.NE.2)NDOWN = 1                                           SET03130
      NOLINE = 0                                                        SET03140
        ICT = 0                                                         SET03150
        NDOWN1 = NDOWN*NACROS                                           SET03160
```

ORIGINAL PAGE IS
OF POOR QUALITY


```
        DO 815 J = 1,NDOWN1,NACROS                                      SET03170
        ICT = ICT + 1                                                   SET03180
       IDUM = (FLDINF(2,J) - FLDINF(1,J))/FLDINF(3,J) + 1               SET03190
       NOLINE = NOLINE + IDUM                                           SET03200
       IF (IMOPT.EQ.2) NLINES(ICT) = IDUM                               SET03210
  815 CONTINUE                                                          SET03220
      IF(NDOWN.EQ.1)GO TO 830                                           SET03230
      NDOWN1 = NDOWN - 1                                                SET03240
      DO 820 J = 1,NDOWN1                                               SET03250
      DO 820 JJ = 1,NACROS                                              SET03260
      ITEMP = JJ + J * NACROS                                           SET03270
      NSS(ITEMP) = NSS(JJ)                                              SET03280
  820 CONTINUE                                                          SET03290
C                                                                       SET03300
C***  SET LINPTR                                                        SET03310
C                                                                       SET03320
  830 LINPTR(1) = 1                                                     SET03330
C                                                                       SET03340
C***   STORE FLDINF                                                     SET03350
C                                                                       SET03360
      IF(IMOPT.EQ.2)GO TO 833                                           SET03370
      DO 832 J = 2,6                                                    SET03380
      DO 832 JJ = 1,4                                                   SET03390
      FLDINF(JJ,J) = FLDINF(JJ,1)                                       SET03400
  832 CONTINUE                                                          SET03410
  833 IF(IMOPT.NE.3)GO TO 840                                           SET03420
      DO 835 J = 1,NUMFIL                                               SET03430
      LINPTR(J + 1) = LINPTR(J) + NLINES(J)                             SET03440
  835 CONTINUE                                                          SET03450
  840 RETURN                                                            SET03460
      END                                                               SET03470
```

21. GTDDM PROCESSOR

FILE: GTDDM FORTRAN A

```
C       WRITTEN BY C J AHLERS                                          GTD00010
C                                                                      GTD00020
C       GROUND TRUTH TAPE DUMP ROUTINE                                 GTD00030
C                                                                      GTD00040
      SUBROUTINE GTDDM(ARRAY,TOP)                                      GTD00050
      IMPLICIT INTEGER (A-Z)                                           GTD00060
      DIMENSION ARRAY(1)                                               GTD00070
      CALL SETIO                                                       GTD00080
      CALL DDM(ARRAY,TOP)                                              GTD00090
      RETURN                                                           GTD00100
      END                                                              GTD00110
```

FILE: ALPHA FORTRAN A

```
      FUNCTION ALPHA(S)                                                 ALP00010
      IMPLICIT INTEGER (A-Z)                                            ALP00020
      DIMENSION A(26)                                                   ALP00030
      DATA A/'A','B','C','D','E','F','G','H','I','J','K','L','M',      ALP00040
     *  'N','O','P','Q','R','S','T','U','V','W','X','Y','Z'/            ALP00050
      DO 10 I=1,26                                                      ALP00060
      ALPHA=I                                                           ALP00070
      IF(S.EQ.A(I)) RETURN                                              ALP00080
   10 CONTINUE                                                          ALP00090
      WRITE(6,20) S                                                     ALP00100
   20 FORMAT(1H ,5X,'THE SYMBOL ',A1,' CAN NOT BE USED.')               ALP00110
      RETURN                                                            ALP00120
      END                                                               ALP00130
```

FILE: DDM FORTRAN A

```
C       WRITTEN BY C W AHLERS                                                   DDM00010
        SUBROUTINE DDM(ARRAY,TOP)                                               DDM00020
        IMPLICIT INTEGER (A-Z)                                                  DDM00030
      DIMENSION ARRAY(1)                                                        DDM00040
      DIMENSION DMTX(11,19)                                                     DDM00050
      DIMENSION FETVEC(30)                                                      DDM00060
      DATA FETVEC/30*0/                                                         DDM00070
        DIMENSION IDATA(3060)                                                   DDM00080
      DIMENSION BLOCK(6)                                                        DDM00090
      COMMON /TAPERD/ IUNIT,IFWST,FSCAN,SAMEND,SAMINC,READY,NSCAN,              DDM00100
     *  LINC,ID(200),DSL,LBUF(30),JREC(30),IBYTE(30),NBUFS,FILENO,LINENODDM00110
     *  ,LININC,NSAMP,NOCHAN,IFORMT                                            DDM00120
        COMMON /GTPR/NROR,NPRT,PRTKEY,VLB(6),GTRDU,GTRDF,                       DDM00130
     *    GTWRU,GTWRF,GTNOF                                                     DDM00140
                                                                                DDM00150
C       EQUIVALENCE (IDATA(1),ARRAY(1))                                         DDM00160
C       EQUIVALENCE (IDATA1(1),ARRAY(3061))                                     DDM00170
      UNIT=GTWRU                                                                DDM00180
      IUNIT=GTRDU                                                               DDM00190
      FORMT=IFORMT                                                              DDM00200
        NEOF=0                                                                  DDM00210
        FILE=0                                                                  DDM00220
      NOFEAT=NOCHAN                                                             DDM00230
      FETVEC(1)=1                                                               DDM00240
      DAY=0                                                                     DDM00250
      MON=0                                                                     DDM00260
      YEAR=0                                                                    DDM00270
      SITE=0                                                                    DDM00280
      PX=196                                                                    DDM00290
        BLOCK(1)=1                                                              DDM00300
      BLOCK(2)=117                                                              DDM00310
      BLOCK(3)=1                                                                DDM00320
      BLOCK(4)=1                                                                DDM00330
      BLOCK(5)=196                                                              DDM00340
      BLOCK(6)=1                                                                DDM00350
 10     CONTINUE                                                                DDM00360
        REWIND GTRDU                                                            DDM00370
        REWIND GTWRU                                                            DDM00380
        RDF=GTRDF-1                                                             DDM00390
        WRF=GTWRF-1                                                             DDM00400
C  10   CONTINUE                                                               DDM00410
        NRPOS=1                                                                 DDM00420
        MAXREC=3060                                                             DDM00430
      ICOUNT=0                                                                  DDM00440
        NREC=0                                                                  DDM00450
        FILE=FILE+1                                                             DDM00460
      LINE=0                                                                    DDM00470
      WRITE(NPRT,951)                                                           DDM00480
 951    FORMAT(1H1)                                                             DDM00490
      CALL TAPRDR(GTRDU,RDF)                                                    DDM00500
      CALL FSFMFL(GTWRU,WRF,ISTATW)                                             DDM00510
        WRITE(NPRT,601) GTRDF,GTWRF                                             DDM00520
  601   FORMAT( //,' TAPE FILE ',I4,5X,'BEING DUMPED TO DOT FILE ',I4)          DDM00530
      DAY=ID(25)                                                                DDM00540
      MON=ID(26)                                                                DDM00550
      YEAR=ID(27)                                                               DDM00560
      SITE=ID(28)                                                               DDM00570
        WRITE(NPRT,306)SITE,DAY,MON,YEAR                                        DDM00580
        IF(PRTKEY.EQ.1) WRITE(GTWRU,356)SITE,DAY,MON,YEAR                       DDM00590
        IF(PRTKEY.EQ.2) WRITE(GTWRU,366)SITE,DAY,MON,YEAR                       DDM00600
        IF(PRTKEY.EQ.3) WRITE(GTWRU,376)SITE,DAY,MON,YEAR                       DDM00610
  306   FORMAT(' SITE = ',I5,5X,'DAY=',I5,5X,'MONTH=',I5,5X,                   DDM00620
     *          'YEAR=',I5)                                                     DDM00630
  356   FORMAT('SITE = ',I5,1X,'DAY=',I5,1X,'MONTH=',I5,1X,                    DDM00640
     *          'YEAR=',I5,' MASK = TY ',' TYPE = GT')                          DDM00650
  366   FORMAT('SITE = ',I5,1X,'DAY=',I5,1X,'MONTH=',I5,1X,                    DDM00660
     *          'YEAR=',I5,' MASK = PHASE 3 ',' TYPE = GT')                     DDM00670
  376   FORMAT('SITE = ',I5,1X,'DAY=',I5,1X,'MONTH=',I5,1X,                    DDM00680
     *          'YEAR=',I5,' MASK = INPUT ',' TYPE = GT')                       DDM00690
        WRITE(NPRT,600)                                                         DDM00700
      CALL FLOINT(BLOCK,FETVEC,NOFEAT)                                          DDM00710
  600   FORMAT(///,' THE 200 OUT LABELS')                                       DDM00720
   20   CONTINUE                                                                DDM00730
      IF(NREC.GE.BLOCK(2)) GO TO 50                                             DDM00740
      CALL LINERD(IDATA,ENDTAP)                                                 DDM00750
        IF(ENDTAP.EQ.-1) GO TO 55                                               DDM00760
        NEOF=0                                                                  DDM00770
        NREC=NREC+1                                                             DDM00780
C     IF(NREC.LE.2) WRITE(NPRT,850) (IDATA(K),K=1,63)                           DDM00790
```

FILE: DDM FORTRAN A

```
 850   FORMAT(1H ,15I4)                                                  DDM00800
       N10=MOD(NREC,10)                                                  DDM00810
C      IF(NREC.EQ.1) WRITE(NPRT,850) (IDATA(K),K=1,GTWEC)                DDM00820
       IF(N10.EQ.0) WRITE(NPRT,836)                                      DDM00830
     * (IDATA(I),I=10,190,10)                                           DDM00840
 836   FORMAT(1H ,5X,19I5)                                               DDM00850
       IF(N10.EQ.0) CALL GTDOTS                                          DDM00860
     *  (IDATA,DMTX,LINE)                                               DDM00870
 839    FORMAT(1H ,5X,19I5)                                              DDM00880
       GO TO 20                                                          DDM00890
   50  CONTINUE                                                          DDM00900
       CALL GTDTL(DMTX,NSYM)                                             DDM00910
       TYPE=PRTKEY                                                       DDM00920
       CALL GTDWR(DMTX,TYPE,NSYM)                                        DDM00930
       WRITE(NPRT,700) NREC                                              DDM00940
  700  FORMAT(///,' NUMBER OF SCAN LINES READ=',I5)                      DDM00950
       WRITE(NPRT,750) GTRDF,GTWRF                                       DDM00960
  750  FORMAT(' FILE ',I5,5X,'WAS DUMPED TO FILE ',I5)                   DDM00970
       GTRDF=GTRDF+1                                                     DDM00980
       GTWRF=GTWRF+1                                                     DDM00990
       RDF=RDF+1                                                         DDM01000
       WRF=WRF+1                                                         DDM01010
       ENDFILE GTWRU                                                     DDM01020
       IF(FILE.LT.GTNOF) GO TO 10                                        DDM01030
 55    CONTINUE                                                          DDM01040
       REWIND GTRDU                                                      DDM01050
       REWIND GTWRU                                                      DDM01060
       WRITE(NPRT,250)                                                   DDM01070
  250  FORMAT(' PROGRAM GTDDM RUN COMPLETED')                            DDM01080
       RETURN                                                            DDM01090
       END                                                               DDM01100
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: GTDOTS FORTRAN A

```
      SUBROUTINE GTDOTS(/IDATA/,DMTX,LINE)                              GTD00010
      IMPLICIT INTEGER (A-Z)                                            GTD00020
      COMMON /GTPRM/NPDR,NPRT,PRTKEY,VLB(6),GTRDU,                      GTD00030
     1          GTPDF,GTWRU,GTWPF,GTNOF                                 GTD00040
      DIMENSION DMTX(11,19),IDATA(1)                                    GTD00050
C     WRITE(6,901)                                                      GTD00060
C901    FORMAT(1H ,10X,'GTDOTS')                                        GTD00070
      LINE=LINE+1                                                       GTD00080
      DO 10 I=10,190,10                                                 GTD00090
      K=I/10                                                            GTD00100
      DMTX(LINE,K)=IDATA(I)                                             GTD00110
 10   CONTINUE                                                          GTD00120
      RETURN                                                            GTD00130
      END                                                               GTD00140
```

FILE: GTOTL FORTRAN A

```
C       WRITTEN C W AHLERS                                                GTD00010
        SUBROUTINE GTOTL(DMTX,NSYM)                                       GTD00020
        IMPLICIT INTEGER (A-Z)                                            GTD00030
      COMMON /TR/TRNS1(256),TRNS2(26),TRNS3(26),TY(11,19)                 GTD00040
      DATA B/' '/                                                         GTD00050
      DIMENSION DMTX(11,19)                                               GTD00060
      DO 5  I=1,26                                                        GTD00070
       TRNS2(I)=B                                                         GTD00080
      TRNS3(I)=B                                                          GTD00090
    5  CONTINUE                                                           GTD00100
      DO 10 I=1,11                                                        GTD00110
      DO 10 J=1,19                                                        GTD00120
      GO=DMTX(I,J)                                                        GTD00130
       SYM=TRNS1(GO)                                                      GTD00140
      NUM=ALPHA(SYM)                                                      GTD00150
      TRNS2(NUM)=SYM                                                      GTD00160
      DMTX(I,J)=SYM                                                       GTD00170
   10  CONTINUE                                                           GTD00180
C     WRITE(6,60) (TRNS2(I),I=1,26)                                       GTD00190
   60   FORMAT(1H ,5X,26A1)                                              GTD00200
      NSYM=0                                                              GTD00210
      DO 100 I=1,26                                                       GTD00220
      IF(TRNS2(I).NE.B) NSYM=NSYM+1                                       GTD00230
      IF(TRNS2(I).NE.B) TRNS3(NSYM)=TRNS2(I)                              GTD00240
  100 CONTINUE                                                            GTD00250
      WRITE(6,110) (TRNS3(I),I=1,26)                                      GTD00260
  110  FORMAT(///,5X,'CATEGORIES FOUND: ',26A1)                           GTD00270
      WRITE(6,20)                                                         GTD00280
   20  FORMAT(1H1,//,5X,'THE 209 DOTS TRANSFORMED')                       GTD00290
       DO 40 I=1,11                                                       GTD00300
       WRITE(6,30) (DMTX(I,J),J=1,19)                                     GTD00310
   40 CONTINUE                                                            GTD00320
   30 FORMAT(1H ,5X,19(4X,A1))                                            GTD00330
      RETURN                                                              GTD00340
      END                                                                 GTD00350
```

FILE: GTDWR FORTRAN A

```
C       WRITTEN BY C W AHLERS                                                   GTD00010
C         THIS PROGRAM WRITES LACIE FORMAT DOT FILES                            GTD00020
C         GIVEN A MATRIX OF DOT LABELS & A TYPE MASK MATRIX                     GTD00030
        SUBROUTINE GTDWR(DMTX,TYPE,NSYM)                                        GTD00040
        IMPLICIT INTEGER (A-Z)                                                  GTD00050
        DIMENSION DMTX(11,19)                                                   GTD00060
        DIMENSION DWR(14)                                                       GTD00070
        COMMON /TP/TRNS1(256),TRNS2(26),TRNS3(26),TY(11,19)                     GTD00080
        COMMON /GTHK/NAQR,NPRT,PRTKEY,VLB(6),GTRDU,GTRDF,                       GTD00090
     *   GTWRU,GTWRF,GTNDF                                                      GTD00100
        WRITE(NPRT,11)                                                          GTD00110
 11     FORMAT(1H1,//,10X,'LACIE FORMAT DOT LABELS')                            GTD00120
        WRITE(NPRT,111) GTWRU                                                   GTD00130
 111    FORMAT(1H ,10X,'WRITTEN TO UNIT ',I5)                                   GTD00140
        IF(TYPE.EQ.1) WRITE(NPRT,12)                                            GTD00150
        IF(TYPE.EQ.2) WRITE(NPRT,13)                                            GTD00160
        IF(TYPE.EQ.3) WRITE(NPRT,14)                                            GTD00170
 12     FORMAT(//,10X,'TYPES BASED ON A TRANSITION YEAR MASK')                  GTD00180
 13     FORMAT(//,10X,'TYPES BASED ON A PHASE THREE MASK')                      GTD00190
 14     FORMAT(//,10X,'TYPES BASED ON AN INPUT MASK')                           GTD00200
        NTYPES=2                                                                GTD00210
        DO 30 TT=1,NTYPES                                                       GTD00220
        DO 30 SS=1,NSYM                                                         GTD00230
        COUNT=0                                                                 GTD00240
        DO 25 I=1,11                                                            GTD00250
        DO 25 J=1,19                                                            GTD00260
        IJ=J+19*(I-1)                                                           GTD00270
        IF(TY(I,J).EQ.TT.AND.DMTX(I,J).EQ.TRNS3(SS)) COUNT=COUNT+1              GTD00280
        IF(TY(I,J).EQ.TT.AND.DMTX(I,J).EQ.TRNS3(SS)) DWR(COUNT)=IJ              GTD00290
        IF(COUNT.EQ.15) WRITE(NPRT,33) TT,TRNS3(SS),(DWR(K),K=1,COUNT)          GTD00300
        IF(COUNT.EQ.15) WRITE(GTWRU,43) TT,TRNS3(SS),(DWR(K),K=1,COUNT)         GTD00310
        IF(COUNT.EQ.15) COUNT=0                                                 GTD00320
 25     CONTINUE                                                                GTD00330
        IF(COUNT.GT.0) WRITE(NPRT,33) TT,TRNS3(SS),(DWR(K),K=1,COUNT)          GTD00340
        IF(COUNT.GT.0) WRITE(GTWRU,43) TT,TRNS3(SS),(DWR(K),K=1,COUNT)         GTD00350
 33     FORMAT(1H ,5X,'DOT ',I1,1X,A1,3X,15I4)                                  GTD00360
 43     FORMAT('DOT ',I1,1X,A1,3X,15I4)                                         GTD00370
 30     CONTINUE                                                                GTD00380
        WRITE(GTWRU,53)                                                         GTD00390
 53     FORMAT('SEND')                                                          GTD00400
        RETURN                                                                  GTD00410
        END                                                                     GTD00420
```

FILE: GTTRNS FORTRAN A

```
      SUBROUTINE GTTRNS"                                                GTT00010
        IMPLICIT INTEGER (A-Z)                                          GTT00020
      DATA W,S,M,N,F,O,N/'W','S','R','H','F','O','N'/                   GTT00030
        COMMON /T3/TRNS1(256),TRNS2(256),TRNS3(256)                     GTT00040
          DO 10 I=1,256                                                 GTT00050
          TRNS1(I)=N                                                    GTT00060
   10     CONTINUE                                                      GTT00070
        TRNS1(99)=W                                                     GTT00080
        TRNS1(124)=W                                                    GTT00090
        TRNS1(100)=S                                                    GTT00100
        TRNS1(126)=S                                                    GTT00110
        TRNS1(101)=S                                                    GTT00120
        TRNS1(125)=R                                                    GTT00130
        TRNS1(102)=R                                                    GTT00140
        TRNS1(127)=R                                                    GTT00150
        TRNS1(103)=F                                                    GTT00160
        TRNS1(128)=F                                                    GTT00170
        TRNS1(104)=O                                                    GTT00180
        TRNS1(129)=O                                                    GTT00190
      DO 20 I=1,15                                                      GTT00200
      II=I+15                                                           GTT00210
      TRNS1(I)=W                                                        GTT00220
      TRNS1(II)=S                                                       GTT00230
   20   CONTINUE                                                       GTT00240
      RETURN                                                            GTT00250
      END                                                               GTT00260
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: SET19 FORTRAN A

```
C       WRITTEN BY C W AHLERS                                                  SET00010
        SUBROUTINE SET19                                                       SET00020
        IMPLICIT INTEGER (A-Z)                                                 SET00030
        DIMENSION CODE(10),CARD(67),EQUCOM(3),ACARD(20)                        SET00040
        DIMENSION SLASH(2)                                                     SET00050
        DATA SLASH /1H,,1H/                                                    SET00060
        DATA CODE/1HTRAN,1HREAD,1HWRIT,                                        SET00070
     * 1HMASK,1HDATE,1HCOMM,1HED1,1HED2,1HEND,1HDUMP/                          SET00080
        DATA EQUCOM/2,1H=,1H,/                                                 SET00090
        DATA V/1HV/,BLNK/1H /,U/1HU/,FF/1HF/,ZZ/1HZ/,P/1HP/                    SET00100
        DATA T/1HT/                                                            SET00110
        DATA QQ/1H?/,II/1HI/                                                   SET00120
C         INCLUDE COMRK1,LIST                                                  SET00130
CCC       INCLUDE COMRK4,LIST                                                  SET00140
CCC     INCLUDE COMRK5,LIST                                                    SET00150
C       INCLUDE CMRK14,LIST                                                    SET00160
        COMMON/INFORM/NOCLS2,NOSUM2,NOFET2,VARSZ2,TOTVT2,NOFLD2,               SET00170
     *                AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLDSV2,VERTX2,        SET00180
     *                FETVC2(30),SUBVC2(75),SUMPTR(75),CLSVC2(60),            SET00190
     *                REPPTS(60),NOGRP,GRPNAM(60),GRPDES(61),                 SET00200
     *                GRPCHR(61),GROUPS(124)                                  SET00210
        DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                         SET00220
        EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                      SET00230
     2              (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                   SET00240
        COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,AMFILE,AMKEY,              SET00250
     *                MISFIL,MISKEY,TRFORM,ENIPTP,ERRKEY,MAPUNT,NOFILE,       SET00260
     *    OBUMAD,OBMXOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                       SET00270
     *  ,NMSTUN,NMSTFI,SCTRUN,MAPFIL                                          SET00280
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,AMTPFL,HISTFL,PCHUNT,                    SET00290
     *  CRDUNT,PRTUNT,RANDIO                                                  SET00300
        COMMON /RTBK/NRDR,NPRT,MSKKEY,VLN(6),GTRDU,GTRDF,                      SET00310
     *               GTWDU,GTWRF,GTNOF                                         SET00320
        COMMON /TAPEHD/IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,            SET00330
     *  LISC,IO(200),OBSL,LHUF(30),JREC(30),IBYTE(30),NBUFS,FILENO,LINENO      SET00340
     *  ,LINUF,LSAMP,NOCHAN,IFORMT                                            SET00350
        COMMON /TR/TRNS1(256),TRNS2(26),TRNS3(26),TY(11,10)                    SET00360
C$END                                                                          SET00370
        DIMENSION PM31(19),PM32(19),PM33(19),PM34(19),PM35(19)                 SET00380
        DIMENSION PM36(19),PM37(19),PM38(19),PM39(19),PM310(19)                SET00390
        DIMENSION PM311(19)                                                    SET00400
        DATA PM31/5*1,2,2,1,1,2,2,8*1/                                         SET00410
        DATA PM32/1,2,1,2,1,2,2,1,2,2,1,2,1,1,2,1,1,2,1/                       SET00420
        DATA PM33/1,2,2,3*1,2,1,2,3*1,3*2,1,2,1,2/                             SET00430
        DATA PM34/2,2,1,1,2,1,2,2,3*1,2,1,2,2,1,2,2,2/                         SET00440
        DATA PM35/1,2,2,3*1,3*2,1,1,2,1,1,2,2,3*1/                             SET00450
        DATA PM36/1,2,2,1,4*2,1,1,4*2,1,2,1,2/                                 SET00460
        DATA PM37/1,2,2,1,2,3*1,2,2,1,2,1,3*2,1,1,2/                           SET00470
        DATA PM38/3*1,1,6*2,1,4*2,1/                                           SET00480
        DATA PM39/1,2,1,2,1,1,2,2,1,1,2,1,2,1,4*2,1/                           SET00490
        DATA PM310/4*1,4*2,1,1,2,3*1,2,1,1,2,2/                                SET00500
        DATA PM311/1,2,1,1,2,5*1,2,1,1,2,2,3*1,2/                              SET00510
        DIMENSION NVEC(30)                                                     SET00520
C                                                                              SET00530
CCC                                                                            SET00540
C                                                                              SET00550
        ZERO = 0                                                               SET00560
        GOOF=0                                                                 SET00570
        GTRDU=12                                                               SET00580
        GTWDU=23                                                               SET00590
        GTRDF=1                                                                SET00600
        GTWRF=1                                                                SET00610
        GTNOF=1                                                                SET00620
        LINE=0                                                                 SET00630
         NPRT=PRTUNT                                                           SET00640
         NRDR=CRDUNT                                                           SET00650
        NOLAB = 0                                                              SET00660
        MSKKEY = 1                                                             SET00670
        NPUT=10                                                                SET00680
        GTNOF=1                                                                SET00690
         NOCHAN=1                                                              SET00700
        IFORMT=1                                                               SET00710
        DO 50 I=1,256                                                          SET00720
 50      TRNS1(I)=00                                                           SET00730
C                                                                              SET00740
        DO 55 I=1,11                                                           SET00750
        DO 55 J=1,10                                                           SET00760
        TY(I,J)=-100                                                           SET00770
 55      CONTINUE                                                              SET00780
        WRITE(NPRT,100)                                                        SET00790
```

```
FILE: SET10    FORTRAN  A

  100 FORMAT(/11X,'INPUT SUMMARY'//)                                    SET00800
C                                                                       SET00810
C     SET UP REREAD BUFFER                                              SET00820
C                                                                       SET00830
      RRUNIT = 30                                                       SET00840
      CALL REREAD(RRUNIT,80)                                            SET00850
C                                                                       SET00860
C    PUT CARD IN BUFFER                                                 SET00870
C                                                                       SET00880
  105 READ(NRDR,103)(ACARD(I),I=1,20)                                   SET00890
  103 FORMAT(20A4)                                                      SET00900
      WRITE(RRUNIT,103)(ACARD(I),I=1,20)                                SET00910
      REWIND RRUNIT                                                     SET00920
C                                                                       SET00930
      READ(RRUNIT,110)CODE1,CARD                                        SET00940
      REWIND RRUNIT                                                     SET00950
      COL= 0                                                            SET00960
      WRITE(NPRT,120)CODE1,CARD                                         SET00970
  120 FORMAT(1X,A4,6X,62A1)                                             SET00980
  110 FORMAT(A4,5X,62A1)                                                SET00990
      DO 130  I=1,NPUT                                                  SET01000
      IF (CODE1.EQ.CODE(I)) GO TO (150,180,210,330,370,                 SET01010
     *                             390,400,410,420,500),I               SET01020
  130    CONTINUE                                                       SET01030
  135 WRITE(NPRT,140)                                                   SET01040
  140 FORMAT( ' INVALID CONTROL CARD - IGNORED ')                       SET01050
      GO TO 105                                                         SET01060
C                                                                       SET01070
C      TRANSFORMATION                                                   SET01080
C                                                                       SET01090
  150 M = NXTCHR(CARD,COL)                                              SET01100
      IF (M.EQ.BLNK) GO TO 105                                          SET01110
      GO TO 155                                                         SET01120
  152 WRITE(NPRT,153)                                                   SET01130
  153 FORMAT(' ERROR ON TRANSFORMATION CARDS')                          SET01140
      GO TO 105                                                         SET01150
  155 J = FIND12(CARD,COL,EQUCOM)                                       SET01160
      IF (J .NE. 2) GO TO 152                                           SET01170
      NOLAB=0                                                           SET01180
      NOLAB = NUMBER(CARD,COL,VLR,NOLAB)                                SET01190
      COL =COL -1                                                       SET01200
      IF(NOLAB.GT.2) GO TO 152                                          SET01210
      IF(VLR(1).GT.VLR(2)) GO TO 152                                    SET01220
      IF(VLR(2).GT.256) GO TO 152                                       SET01230
      NB=VLR(1)                                                         SET01240
      NE=VLR(2)                                                         SET01250
      DO 156 I=NB,NE                                                    SET01260
       TRNSL(I)=M                                                       SET01270
  156    CONTINUE                                                       SET01280
      GO TO 105                                                         SET01290
C                                                                       SET01300
C      READ TAPE                                                        SET01310
C                                                                       SET01320
  180 M = NXTCHR(CARD,COL)                                              SET01330
      IF (M .EQ. BLNK ) GO TO 105                                       SET01340
      IF (M.EQ.U) GO TO 190                                             SET01350
      IF (M.EQ.FF) GO TO 200                                            SET01360
  185 WRITE(NPRT,187)                                                   SET01370
  187 FORMAT(' ERROR ON READ TAPE CARD')                                SET01380
      GO TO 105                                                         SET01390
  190 J = FIND12(CARD,COL,EQUCOM)                                       SET01400
      IF (J .NE. 2) GO TO 185                                           SET01410
      M = NUMBER(CARD,COL,GTRDU,ZERO)                                   SET01420
      COL = COL - 1                                                     SET01430
      GO TO 180                                                         SET01440
  200 J = FIND12(CARD,COL,EQUCOM)                                       SET01450
      IF ( J .NE. 2 ) GO TO 185                                         SET01460
      M = NUMBER(CARD,COL,GTRDF,ZERO)                                   SET01470
      COL = COL - 1                                                     SET01480
      GO TO 180                                                         SET01490
C                                                                       SET01500
C      WRITE FILE                                                       SET01510
C                                                                       SET01520
  210 CONTINUE                                                          SET01530
  214 M = NXTCHR(CARD,COL)                                              SET01540
      IF (M .EQ. BLNK  ) GO TO 105                                      SET01550
       IF (M.EQ.U) GO TO 230                                            SET01560
      IF (M.EQ.FF) GO TO 240                                            SET01570
  215 WRITE(NPRT,220)                                                   SET01580
```

```
FILE: SET19    FORTRAN  A

  220 FORMAT(' ERROR ON WRITE FILE CARD')                                  SET01590
      GO TO 105                                                            SET01600
  230 J = FIND12(CARD,COL,EQUCOM)                                          SET01610
      IF ( J .NE. 2)  GO TO 215                                            SET01620
      M = NUMBER(CARD,COL,GTWRU,ZERO)                                      SET01630
      COL = COL - 1                                                        SET01640
      GO TO 214                                                            SET01650
  240 J = FIND12(CARD,COL,EQUCOM)                                          SET01660
      IF (J .NE.  2) GO TO 215                                             SET01670
      M = NUMBER(CARD,COL,GTWRF,ZERO)                                      SET01680
      COL = COL - 1                                                        SET01690
      GO TO 214                                                            SET01700
C                                                                          SET01710
CC    MASK CARD                                                            SET01720
C                                                                          SET01730
  330 M = NXTCHR(CARD,COL)                                                 SET01740
      IF (M .EQ. BLNK ) GO TO 105                                          SET01750
      IF (M.EQ.P) GO TO 340                                                SET01760
      IF (M.EQ.T) GO TO 341                                                SET01770
      IF(M.EQ.II) GO TO 342                                                SET01780
  333 WRITE(NPRT,335)                                                      SET01790
  335 FORMAT(' ERROR ON MASK CARD --- TRANSITION YEAR MASK USED')          SET01800
      GO TO 341                                                            SET01810
  340 MSKKEY = 2                                                           SET01820
      DO 20 J=1,19                                                         SET01830
      TY(1,J)=PH31(J)                                                      SET01840
      TY(2,J)=PH32(J)                                                      SET01850
      TY(3,J)=PH33(J)                                                      SET01860
      TY(4,J)=PH34(J)                                                      SET01870
      TY(5,J)=PH35(J)                                                      SET01880
      TY(6,J)=PH36(J)                                                      SET01890
      TY(7,J)=PH37(J)                                                      SET01900
      TY(8,J)=PH38(J)                                                      SET01910
      TY(9,J)=PH39(J)                                                      SET01920
      TY(10,J)=PH310(J)                                                    SET01930
      TY(11,J)=PH311(J)                                                    SET01940
   20   CONTINUE                                                          SET01950
      GO TO 105                                                            SET01960
  341   MSKKEY=1                                                           SET01970
      DO 10 I=1,11                                                         SET01980
      DO 10 J=1,19                                                         SET01990
      IJ=J+19*(I-1)                                                        SET02000
      TY(I,J)=1                                                            SET02010
      IF(IJ/2*2.EQ.IJ) TY(I,J)=2                                           SET02020
   10   CONTINUE                                                          SET02030
      IF(GOOF.EQ.1) GO TO 461                                              SET02040
      GO TO 105                                                            SET02050
  342 MSKKEY=3                                                             SET02060
      LINE=LINE+1                                                          SET02070
      IF(LINE.GT.11) GO TO 333                                             SET02080
      J=FIND12(CARD,COL,EQUCOM)                                            SET02090
      IF(J.NE.2) GO TO 333                                                 SET02100
      NOMSK=0                                                              SET02110
      NOMSK=NUMBER(CARD,COL,MVEC,NOMSK)                                    SET02120
      IF(NOMSK.GT.19) GOOF=1                                               SET02130
      IF(NOMSK.GT.19) GO TO 105                                            SET02140
      DO 345 J=1,NOMSK                                                     SET02150
      TY(LINE,J)=MVEC(J)                                                   SET02160
  345   CONTINUE                                                           SET02170
      GO TO 105                                                            SET02180
C                                                                          SET02190
CC    DATE CARD                                                            SET02200
C                                                                          SET02210
  370 M = NXTCHR(CARD,COL)                                                 SET02220
      IF ( M .EQ. BLNK ) GO TO 105                                         SET02230
      READ(RDUNIT,380)DATE                                                 SET02240
  380 FORMAT(10A,15A4)                                                     SET02250
      REWIND RDUNIT                                                        SET02260
      GO TO 105                                                            SET02270
C                                                                          SET02280
CC    COMMENT CARD                                                         SET02290
C                                                                          SET02300
  390 M = NXTCHR(CARD,COL)                                                 SET02310
      IF (M .EQ. BLNK ) GO TO 105                                          SET02320
      READ(RDUNIT,380)COMENT                                               SET02330
      REWIND RDUNIT                                                        SET02340
      GO TO 105                                                            SET02350
C                                                                          SET02360
C                                                                          SET02370
```

FILE: SET19 FORTRAN A

```
C       HED1                                                                    SET02380
C                                                                               SET02390
  400 M = NXTCHR(CARD,COL)                                                      SET02400
      READ(RRUNIT,380) HED1                                                     SET02410
      REWIND RRUNIT                                                             SET02420
      GO TO 105                                                                 SET02430
C                                                                               SET02440
C       HED2                                                                    SET02450
C                                                                               SET02460
  410 M = NXTCHR(CARD,COL)                                                      SET02470
      READ(RRUNIT,380) HED2                                                     SET02480
      REWIND RRUNIT                                                             SET02490
      GO TO 105                                                                 SET02500
C                                                                               SET02510
C       CONVERT CARD                                                            SET02520
C                                                                               SET02530
  500  M=NXTCHR(CARD,COL)                                                       SET02540
       IF(M.EQ.BLNK) GO TO 105                                                  SET02550
       IF(M.EQ.FF) GO TO 510                                                    SET02560
  530   WRITE(NPRT,520)                                                         SET02570
  520   FORMAT(' ERROR ON CONVERT CARD')                                        SET02580
       GO TO 105                                                                SET02590
  510  J=FIND12(CARD,COL,EQUCOM)                                                SET02600
                                                                                SET02610
       M=NUMBER(CARD,COL,GTNOF,ZERO)                                            SET02620
       COL=COL-1                                                                SET02630
       GO TO 500                                                                SET02640
C                                                                               SET02650
C       *END*                                                                   SET02660
C                                                                               SET02670
  420 CONTINUE                                                                  SET02680
      DO 430 I=1,256                                                            SET02690
      IF(TRNS1(I).EQ.QQ) GO TO 450                                              SET02700
  430    CONTINUE                                                               SET02710
       GO TO 440                                                                SET02720
  450   WRITE(NPRT,153)                                                         SET02730
      WRITE(NPRT,451)                                                           SET02740
  451   FORMAT(//,10X,'DEFAULT TRANSFORMATION USED')                            SET02750
       CALL GTTRNS                                                              SET02760
C                                                                               SET02770
  440  CONTINUE                                                                 SET02780
C                                                                               SET02790
      IF(GOOF.EQ.1) GO TO 333                                                   SET02800
       DO 460 I=1,11                                                            SET02810
       DO 460 J=1,19                                                            SET02820
       IF(TY(I,J).EQ.-100) GOOF=1                                               SET02830
       IF(GOOF.EQ.1) GO TO 333                                                  SET02840
  460  CONTINUE                                                                 SET02850
  461   CONTINUE                                                                SET02860
C                                                                               SET02870
      WRITE(NPRT,1000)                                                          SET02880
      WRITE(NPRT,1100) GTRDU,GTRDF                                              SET02890
       WRITE(NPRT,1200) GTWRU,GTWRF                                             SET02900
       WRITE(NPRT,1300) GTNOF                                                   SET02910
      IF (MSKKEY .EQ. 1) WRITE(NPRT,1030)                                       SET02920
      IF (MSKKEY .EQ. 2) WRITE(NPRT,1031)                                       SET02930
      IF (MSKKEY .EQ. 3) WRITE(NPRT,1032)                                       SET02940
 1000 FORMAT(//' USER HAS REQUESTED THE FOLLOWING OPTIONS :'/)                  SET02950
 1030 FORMAT(' TRANSITION YEAR MASK')                                           SET02960
 1031 FORMAT(' PHASE THREE MASK')                                               SET02970
 1032 FORMAT(' INPUT MASK')                                                     SET02980
 1100  FORMAT(' READ UNIT = ',I3,'  READ FILE = ',I3)                           SET02990
 1200   FORMAT(' WRITE UNIT = ',I3,' WRITE FILE = ',I3)                        SET03000
 1300  FORMAT(' THE NUMBER OF TAPE FILES TO BE DUMPED TO DOT FILES =',         SET03010
     *  I4)                                                                    SET03020
C                                                                               SET03030
      WRITE(NPRT,630)                                                           SET03040
  630   FORMAT(//,' THE CROP CODE TO SYMBOL TRANSFORMATION')                    SET03050
      FIRST=1                                                                   SET03060
      SYM=TRNS1(1)                                                              SET03070
      DO 631 I=1,256                                                            SET03080
      LAST=I-1                                                                  SET03090
      IF(TRNS1(I).NE.SYM)  WRITE(NPRT,632) SYM,FIRST,LAST                       SET03100
      IF(TRNS1(I).NE.SYM) FIRST=I                                               SET03110
      IF(TRNS1(I).NE.SYM) SYM=TRNS1(I)                                          SET03120
       IF(I.EQ.256) WRITE(NPRT,632) SYM,FIRST,I                                 SET03130
  631   CONTINUE                                                                SET03140
  632   FORMAT(1H ,5X,A1,' = ',I3,' , ',I3)                                     SET03150
      WRITE(NPRT,640)                                                           SET03160
```

FILE: SET19 FORTRAN A

```
 640   FORMAT(//,' THE MASK')                                          SET03170
       DO 641 I=1,11                                                   SET03180
       WRITE(NPRT,642)(TY(I,J),J=1,19)                                 SET03190
 641   CONTINUE                                                        SET03200
 642   FORMAT(1H ,5X,19I5)                                             SET03210
       RETURN                                                          SET03220
C                                                                      SET03230
       END                                                             SET03240
```

22. GTTCN PROCESSOR

FILE: GTTCN FORTRAN A

```
C       WRITTEN BY C W AHLERS                                          GTT00010
C                                                                      GTT00020
C       GROUND TRUTH TAPE CONVERSION ROUTINE                           GTT00030
C                                                                      GTT00040
      SUBROUTINE GTTCN(ARRAY,TOP)                                      GTT00050
      IMPLICIT INTEGER (A-Z)                                           GTT00060
      DIMENSION ARRAY(1)                                               GTT00070
      CALL SET17                                                       GTT00080
      CALL TCN(ARRAY,TOP)                                              GTT00090
      RETURN                                                           GTT00100
      END                                                              GTT00110
```

FILE: GTCRPL FORTRAN A

```
C       WRITTEN BY C W AHLERS                                          GTC00010
        SUBROUTINE GTCRPL(CROP,MT,NC)                                  GTC00020
        IMPLICIT INTEGER (A-Z)                                         GTC00030
        DIMENSION MT(6)                                                GTC00040
       COMMON /GTRK/NROP,NPRT,PRTKEY,VLB(6),GTROU,GTROF                GTC00050
      *  ,GTWRU,GTWRF,GTNOF                                            GTC00060
C       WRITE(NPRT,901)                                                GTC00070
 901    FORMAT(1H ,10X,'GTCRPL')                                       GTC00080
        NC=0                                                           GTC00090
        DO 10 I=1,6                                                    GTC00100
        IF(VLB(I).LT.1) GO TO 10                                       GTC00110
        CC=MT(I)                                                       GTC00120
        N=0                                                            GTC00130
        DO 20 J=1,6                                                    GTC00140
        IF(VLB(J).LT.1) GO TO 20                                       GTC00150
        IF(CC.EQ.MT(J)) N=N+1                                          GTC00160
   20   CONTINUE                                                       GTC00170
        IF(N.LE.NC) GO TO 10                                           GTC00180
        NC=N                                                           GTC00190
        CROP=CC                                                        GTC00200
        IF(NC.GE.3) RETURN                                             GTC00210
   10   CONTINUE                                                       GTC00220
        RETURN                                                         GTC00230
        END                                                            GTC00240
```

FILE: GTUNPK FORTRAN A

```
      SUBROUTINE GTUNPK(IDATA,N1,N2,OS,NRPDS,LENGTH)                    GTU00010
      IMPLICIT INTEGER (A-Z)                                            GTU00020
      DIMENSION IDATA(1)                                                GTU00030
C     WRITE(3,900)                                                      GTU00040
C     WRITE(6,900)                                                      GTU00050
  900 FORMAT(1H ,10X,'GTUNPK')                                          GTU00060
      DO 10 I=1,NRPDS                                                   GTU00070
      DO 20 J=N1,N2                                                     GTU00080
      JJ=J+OS                                                           GTU00090
      IF(JJ.GT.30A0) WRITE(6,901) JJ                                    GTU00100
      IF(JJ.GT.30A0) STOP                                               GTU00110
  901   FORMAT(1H ,10X,'JJ=',I10)                                       GTU00120
      II=IDATA(JJ)                                                      GTU00130
      IF(II.GT.128)  IDATA(JJ)=IDATA(JJ)-128                            GTU00140
      IF(II.LE.128)  IDATA(JJ)=128+IDATA(JJ)                            GTU00150
   20 CONTINUE                                                          GTU00160
      OS=OS+LENGTH                                                      GTU00170
   10 CONTINUE                                                          GTU00180
      RETURN                                                            GTU00190
      END                                                               GTU00200
```

~~223~~
501

FILE: LINLAB FORTRAN A

```
      SUBROUTINE LINLAB(/IDATA/,MAXREC)                                 LIN00010
      IMPLICIT INTEGER (A-Z)                                            LIN00020
      DIMENSION MT(6),IDATA(3060)                                       LIN00030
C     WRITE(3,901)                                                      LIN00040
901   FORMAT(1H ,10X,'LINLAB')                                          LIN00050
      OS=72                                                             LIN00060
      DO 30 PIX=1,196                                                   LIN00070
      OS=OS+2                                                           LIN00080
      M=0                                                               LIN00090
      DO 40 LL=1,3                                                      LIN00100
      DO 50 SS=1,2                                                      LIN00110
      M=M+1                                                             LIN00120
      SSS=OS+SS-2                                                       LIN00130
      IF(SSS.GT.3060) WRITE(6,903) SSS                                  LIN00140
      IF(SSS.GT.3060) STOP                                              LIN00150
C     WRITE (6,903) SSS                                                 LIN00160
903   FORMAT(1H ,10X,'SSS=',I10)                                        LIN00170
      IF (M.GT.6) WRITE(6,911) M                                        LIN00180
      IF (M.GT.6) STOP                                                  LIN00190
911     FORMAT(1H ,10X,'M=',I5)                                         LIN00200
      MT(M)=IDATA(SSS)                                                  LIN00210
 50   CONTINUE                                                          LIN00220
      OS =OS+MAXREC                                                     LIN00230
 40   CONTINUE                                                          LIN00240
      CALL GTCRPL(CROP,MT,NCROP)                                        LIN00250
C     PP=PIX+72                                                         LIN00260
      PP=PIX                                                            LIN00270
      IDATA(PP)=CROP                                                    LIN00280
      OS=72+2*PIX                                                       LIN00290
 30   CONTINUE                                                          LIN00300
      DO 60 I=269,3060                                                  LIN00310
 60   IDATA(I)=0                                                        LIN00320
C     WRITE(6,902) (IDATA(K),K=1,196)                                   LIN00330
902     FORMAT(1H ,25I4)                                                LIN00340
      RETURN                                                            LIN00350
      END                                                               LIN00360
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FILE: SET17    FORTRAN  A

C         WRITTEN BY C. W. AHLERS                                              SET00010
      SUBROUTINE SET17                                                         SET00020
      IMPLICIT INTEGER (A-Z)                                                   SET00030
      DIMENSION CODE(10),CARD(62),EQUCOM(3),ACARD(20)                          SET00040
      DIMENSION SLASH(2)                                                       SET00050
      DATA SLASH /1H/,1H/                                                      SET00060
      DATA CODE/4HLABE,4HREAD,4HWRIT,                                          SET00070
     * 4HOPTI,4HDATE,4HCOMM,4HED1',4HED2',4HEND',4HCONV'/                      SET00080
      DATA  EQUCOM/2H=',1H,/                                                   SET00090
      DATA V/1HV/,BLNK/1H /,U/1HU/,FF/1HF/,OO/1H0/,P/1HP/                      SET00100
C       INCLUDE COMMK1,LIST                                                    SET00110
CCC     INCLUDE COMMK4,  LIST                                                  SET00120
CCC     INCLUDE COMMK4,LIST                                                    SET00130
C       INCLUDE CMMK14,LIST                                                    SET00140
      COMMON/INFORM/NOCLS2,NUSUB2,NOFET2,VARSZ2,TOTVT2,NOFLD2,                 SET00150
     *              AVAR2,COVAR2,CLSID2,SUBNO2,SUBOS2,FLDSV2,VERTX2,           SET00160
     *              FETVC2(30),SUBVC2(75),SUBPTR(75),CLSVC2(60),               SET00170
     *              KEPPTS(60),NOGRP,GRPNAM(60),GRPDEX(61),                    SET00180
     *              GRPCHK(61),GROUPS(124)                                     SET00190
      DIMENSION HED1(15),HED2(15),DATE(3),COMENT(15)                           SET00200
      EQUIVALENCE (HED1(1),HEAD(4)),(DATE(1),HEAD(22)),                        SET00210
     2            (HED2(1),HEAD(30)),(COMENT(1),HEAD(48))                      SET00220
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,RMFILE,RMKEY,                SET00230
     *              HISFIL,HISKEY,INFORM,EMIPTP,ERPKEY,MAPUNT,NOFILF,          SET00240
     *    DRUMAD,DRMPOS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                       SET00250
     *  ,NHSTUN,NHSTFI,SCTHUN,MAPFIL                                           SET00260
     *  ,DOTUNT,DOTFIL,NCMPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                     SET00270
     *  CRUNT,PRTUNT,HANDIO                                                    SET00280
      COMMON /GTRK/NRDR,NPRT,PRTKEY,VLB(6),GTRDU,GTRDF,                        SET00290
     *             GTWRU,GTWRF,GTNOF                                           SET00300
      COMMON /TAPEWD/IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,              SET00310
     *  LINC,ID(200),DSL,LHUF(30),JREC(30),IBYTE(30),NHUFS,FILENO,LININDSET00320
     *  ,LININC,NSAMP,NOCHAN,IFORMT                                            SET00330
C$END                                                                          SET00340
      ZERO = 0                                                                 SET00350
       NPRT=PRTUNT                                                             SET00360
       NRDR=CRDUNT                                                             SET00370
      GTRDU=11                                                                 SET00380
      GTWRU=12                                                                 SET00390
      GTRDF=1                                                                  SET00400
      GTWRF=1                                                                  SET00410
      GTNOF=1                                                                  SET00420
      NOLAB = 0                                                                SET00430
      PRTKEY = 0                                                               SET00440
      NPUT=10                                                                  SET00450
      GTNOF=1                                                                  SET00460
       NOCHAN=1                                                                SET00470
      IFORMT=1                                                                 SET00480
C                                                                              SET00490
      WRITE(NPRT,100)                                                          SET00500
  100 FORMAT(/11X,'INPUT SUMMARY'//)                                           SET00510
C                                                                              SET00520
CC    SET UP REREAD BUFFER                                                     SET00530
CC                                                                             SET00540
      RRUNIT = 30                                                              SET00550
      CALL REREAD(RRUNIT,80)                                                   SET00560
C                                                                              SET00570
CC    PUT CARD IN BUFFER                                                       SET00580
CC                                                                             SET00590
  105 READ(NRDR,103)(ACARD(I),I=1,20)                                          SET00600
  103 FORMAT(20A4)                                                             SET00610
      WRITE(RRUNIT,103)(ACARD(I),I=1,20)                                       SET00620
      REWIND RRUNIT                                                            SET00630
C                                                                              SET00640
      READ(RRUNIT,110)CODE1,CARD                                               SET00650
      REWIND RRUNIT                                                            SET00660
      COL= 0                                                                   SET00670
      WRITE(NPRT,120)CODE1,CARD                                                SET00680
  120 FORMAT(1X,A4,2X,62A1)                                                    SET00690
  110 FORMAT(A4,A2,62A1)                                                       SET00700
      DO 130  I=1,NPUT                                                         SET00710
      IF (CODE1.EQ.CODE(I)) GO TO (150,160,210,330,370,                        SET00720
     *                             390,400,410,420,500),I                      SET00730
  130    CONTINUE                                                            SET00740
  135 WRITE(NPRT,140)                                                          SET00750
  140 FORMAT(' INVALID CONTROL CARD - IGNORED ')                               SET00760
      GO TO 105                                                                SET00770
C                                                                              SET00780
C       LABEL VECTOR                                                           SET00790
```

```
FILE: SET17    FORTRAN  A

C                                                                       SET00H00
  150 M = NXTCHR(CARD,COL)                                              SET00H10
      IF (M.EQ.V) GO TO 155                                             SET00H20
      IF (M.EQ.BLNK) GO TO 105                                          SET00H30
  152 WRITE(NPRT,153)                                                   SET00H40
  153 FORMAT(' ERROR ON LABEL CARD')                                    SET00H50
      GO TO 105                                                         SET00H60
  155 J = FIND12(CARD,COL,EQUCOM)                                       SET00H70
      IF (J .NE. 2) GO TO 152                                           SET00H80
      NOLAB = NUMBER(CARD,COL,VLR,NOLAB)                                SET00H90
      GO TO 105                                                         SET00900
C                                                                       SET00910
CCC    READ TAPE                                                        SET00920
C                                                                       SET00930
  180 M = NXTCHR(CARD,COL)                                              SET00940
      IF (M .EQ. BLNK ) GO TO 105                                       SET00950
      IF (M.EQ.U) GO TO 190                                             SET00960
      IF (M.EQ.FF) GO TO 200                                            SET00970
  185 WRITE(NPRT,187)                                                   SET00980
  187 FORMAT(' ERROR ON READ TAPE CARD')                                SET00990
      GO TO 105                                                         SET01000
  190 J = FIND12(CARD,COL,EQUCOM)                                       SET01010
      IF (J .NE. 2) GO TO 185                                           SET01020
      M = NUMBER(CARD,COL,GTRDU,ZERO)                                   SET01030
      COL = COL - 1                                                     SET01040
      GO TO 180                                                         SET01050
  200 J = FIND12(CARD,COL,EQUCOM)                                       SET01060
      IF ( J .NE. 2 ) GO TO 185                                         SET01070
      M = NUMBER(CARD,COL,GTRDF,ZERO)                                   SET01080
      COL = COL - 1                                                     SET01090
      GO TO 180                                                         SET01100
C                                                                       SET01110
CCC    WRITE TAPE                                                       SET01120
C                                                                       SET01130
  210 CONTINUE                                                          SET01140
  214 M = NXTCHR(CARD,COL)                                              SET01150
      IF (M .EQ. BLNK ) GO TO 105                                       SET01160
      IF (M.EQ.U) GO TO 230                                             SET01170
      IF (M.EQ.FF) GO TO 240                                            SET01180
  215 WRITE(NPRT,220)                                                   SET01190
  220 FORMAT(' ERROR ON WRITE TAPE CARD')                               SET01200
      GO TO 105                                                         SET01210
  230 J = FIND12(CARD,COL,EQUCOM)                                       SET01220
      IF ( J .NE. 2)  GO TO 215                                         SET01230
      M = NUMBER(CARD,COL,GTWRU,ZERO)                                   SET01240
      COL = COL - 1                                                     SET01250
      GO TO 214                                                         SET01260
  240 J = FIND12(CARD,COL,EQUCOM)                                       SET01270
      IF (J .NE.  2) GO TO 215                                          SET01280
      M = NUMBER(CARD,COL,GTWRF,ZERO)                                   SET01290
      COL = COL - 1                                                     SET01300
      GO TO 214                                                         SET01310
C                                                                       SET01320
CCC    OPTION CARD                                                      SET01330
C                                                                       SET01340
  330 M = NXTCHR(CARD,COL)                                              SET01350
      IF (M .EQ. BLNK ) GO TO 105                                       SET01360
      IF (M.EQ.P) GO TO 340                                             SET01370
  333 WRITE(NPRT,335)                                                   SET01380
  335 FORMAT(' ERROR ON OPTION CARD')                                   SET01390
      GO TO 105                                                         SET01400
  340 PRTKEY = 1                                                        SET01410
      GO TO 105                                                         SET01420
C                                                                       SET01430
CCC    DATE CARD                                                        SET01440
C                                                                       SET01450
  370 M = NXTCHR(CARD,COL)                                              SET01460
      IF ( M .EQ. BLNK ) GO TO 105                                      SET01470
      READ(NRUNIT,380)DATE                                              SET01480
  380 FORMAT(10X,1S4)                                                   SET01490
      REWIND NRUNIT                                                     SET01500
      GO TO 105                                                         SET01510
C                                                                       SET01520
CCC    COMMENT CARD                                                     SET01530
C                                                                       SET01540
  390 M = NXTCHR(CARD,COL)                                              SET01550
      IF (M .EQ. BLNK ) GO TO 105                                       SET01560
      READ(NRUNIT,400)COMENT                                            SET01570
      REWIND NRUNIT                                                     SET01580
```

```
FILE: SET17    FORTRAN  A

      GO TO 105                                                          SET01590
C                                                                        SET01600
CCCC  HED1                                                               SET01610
C                                                                        SET01620
  400 M = NXTCHR(CARD,COL)                                               SET01630
      READ(RRUNIT,340) HED1                                              SET01640
      REWIND RRUNIT                                                      SET01650
      GO TO 105                                                          SET01660
C                                                                        SET01670
CCC   HED2                                                               SET01680
C                                                                        SET01690
  410 M = NXTCHR(CARD,COL)                                               SET01700
      READ(RRUNIT,340) HED2                                              SET01710
      REWIND RRUNIT                                                      SET01720
      GO TO 105                                                          SET01730
C                                                                        SET01740
CCC    CONVERT CARD                                                      SET01750
C                                                                        SET01760
 500   M=NXTCHR(CARD,COL)                                                SET01770
       IF(M.EQ.BLNK) GO TO 105                                           SET01780
       IF(M.EQ.FF) GO TO 510                                             SET01790
 530    WRITE(NPRT,520)                                                  SET01800
 520    FORMAT(' ERROR ON CONVERT CARD')                                 SET01810
       GO TO 105                                                         SET01820
 510   J=FIND12(CARD,COL,EQUCOM)                                         SET01830
       M=NUMBER(CARD,COL,GTNOF,ZERO)                                     SET01840
       COL=COL-1                                                         SET01850
       GO TO 500                                                         SET01860
C                                                                        SET01870
CCC    *END*                                                             SET01880
C                                                                        SET01890
  420 CONTINUE                                                           SET01900
      IF (NOLAB .NE. 0) GO TO 440                                        SET01910
      DO 430 I=1,6                                                       SET01920
      VLA(I) = 1                                                         SET01930
  430    CONTINUE                                                        SET01940
      NOLAB = 6                                                          SET01950
       GO TO 440                                                         SET01960
 450    WRITE(NPRT,153)                                                  SET01970
       NOLAB=0                                                           SET01980
       GO TO 420                                                         SET01990
C                                                                        SET02000
 440   CONTINUE                                                          SET02010
       IF(NOLAB.LT.6) GO TO 450                                          SET02020
C                                                                        SET02030
C                                                                        SET02040
      WRITE(NPRT,1000)                                                   SET02050
      WRITE(NPRT,1100) GTRDU,GTRDF                                       SET02060
       WRITE(NPRT,1200) GTWRU,GTWRF                                      SET02070
       WRITE(NPRT,1300) GTNOF                                            SET02080
       WRITE(NPRT,1010) (VLA(I),I=1,6)                                   SET02090
      IF (PRTKEY .EQ. 1) WRITE(NPRT,1030)                                SET02100
 1000 FORMAT(//' USER HAS REQUESTED THE FOLLOWING OPTIONS :'/)           SET02110
 1010 FORMAT(' THE LABEL VECTOR IS = ',6I3)                              SET02120
 1030 FORMAT(' PRINT THE 209 DOT LABELS')                                SET02130
 1100 FORMAT(' READ UNIT = ',I3,'  READ FILE = ',I3)                     SET02140
 1200  FORMAT(' WRITE UNIT = ',I3,' WRITE FILE = ',I3)                   SET02150
 1300   FORMAT(' THE NUMBER OF FILES TO BE CONVERTED =' ,I3)             SET02160
C                                                                        SET02170
      RETURN                                                             SET02180
C                                                                        SET02190
      END                                                                SET02200
                                                                         SET02210
                                                                         SET02220
```

```
FILE: TCN      FORTRAN  A

C       WRITTEN BY C W AHLERS                                                  TCN00010
        SUBROUTINE TCN(ARRAY,TOP)                                              TCN00020
        IMPLICIT INTEGER (A-Z)                                                 TCN00030
       DIMENSION ARRAY(1)                                                      TCN00040
       LOGICAL*1 VAL(2400),IDL(800)                                            TCN00050
        DIMENSION IDATA1(3060)                                                 TCN00060
       DIMENSION IBUF(765)                                                     TCN00070
       DIMENSION FETVEC(30)                                                    TCN00080
       DATA FETVEC/30*0/                                                       TCN00090
        DIMENSION IDATA(3060)                                                  TCN00100
       DIMENSION BLOCK(6)                                                      TCN00110
       COMMON /TAPEMO/ IUNIT,IFRST,FSCAN,SAMEND,SAMINC,READY,NSCAN,            TCN00120
      *  LINC,ID(200),DSL,LBUF(30),JREC(30),IBYTE(30),NBUFS,FILENO,LINENDTCN00130
      *  ,LININC,NSAMP,NOCHAN,IFORMT                                          TCN00140
       COMMON /WRTAP/ICOUNT,FORMT,UNIT,VARBL(600),IREMD                        TCN00150
       EQUIVALENCE (VAL,VARBL)                                                 TCN00160
        EQUIVALENCE (ID,IDL)                                                   TCN00170
C      LOGICAL*1 DY(4),MN(4),YR(4),ST(4)                                       TCN00180
CC     EQUIVALENCE (DAY,DY(1)),(MON,MN(1)),(YEAR,YR(1)),(SITE,ST(1))           TCN00190
CC     EQUIVALENCE (DAY,DY),(MON,MN),(YEAR,YR),(SITE,ST)                       TCN00200
        COMMON /GTAK/NHDR,NPRT,PRTKEY,VLB(6),GTRDU,GTRDF,                      TCN00210
      *   GTWRU,GTWRF,GTNOF                                                    TCN00220
                                                                               TCN00230
C       EQUIVALENCE (IDATA(1),ARRAY(1))                                        TCN00240
C       EQUIVALENCE (IDATA1(1),ARRAY(3061))                                    TCN00250
       UNIT=GTWRU                                                              TCN00260
       IUNIT=GTRDU                                                             TCN00270
       FORMT=IFORMT                                                            TCN00280
        NEOF=0                                                                 TCN00290
        FILE=0                                                                 TCN00300
       NOCHAN=1                                                                TCN00310
       NOFEAT=NOCHAN                                                           TCN00320
       FETVEC(1)=1                                                             TCN00330
       ZERO=0                                                                  TCN00340
       OUTPX=1096                                                              TCN00350
       INPX=392                                                                TCN00360
       OSG=-2                                                                  TCN00370
       ONE=1                                                                   TCN00380
       GTNOFN=3                                                                TCN00390
        BLOCK(1)=1                                                             TCN00400
       BLOCK(2)=351                                                            TCN00410
       BLOCK(3)=1                                                              TCN00420
       BLOCK(4)=1                                                              TCN00430
       BLOCK(5)=392                                                            TCN00440
       BLOCK(6)=1                                                              TCN00450
   10   CONTINUE                                                              TCN00460
        REWIND GTRDU                                                           TCN00470
        REWIND GTWRU                                                           TCN00480
        RDF=GTRDF-1                                                            TCN00490
        WRF=GTWRF-1                                                            TCN00500
        CALL FSEMFL(GTWRU,WRF,ISTATW)                                          TCN00510
C  10   CONTINUE                                                              TCN00520
        JREC=0                                                                 TCN00530
       ICOUNT=0                                                                TCN00540
        NREC=0                                                                 TCN00550
        FILE=FILE+1                                                            TCN00560
       DAY=0                                                                   TCN00570
       MON=0                                                                   TCN00580
       YEAR=0                                                                  TCN00590
       SITE=0                                                                  TCN00600
       CALL TCNHDR(GTRDU,RDF)                                                  TCN00610
       DAY=ID(25)                                                              TCN00620
       MON=ID(26)                                                              TCN00630
       YEAR=ID(27)                                                             TCN00640
       SITE=ID(28)                                                             TCN00650
       VAL(73)=IDL(100)                                                        TCN00660
       VAL(74)=IDL(101)                                                        TCN00670
       VAL(75)=IDL(102)                                                        TCN00680
       VAL(79)=IDL(111)                                                        TCN00690
       VAL(80)=IDL(112)                                                        TCN00700
       CALL WRTHFD(NOFEAT,FETVEC,OUTPX,FORMT,GTWRU)                            TCN00710
        GTVEC=540                                                              TCN00720
        WRITE(NPRT,601) GTRDF,GTWRF                                            TCN00730
  601   FORMAT( //,' FILE ',I4,5X,'BEING CONVERTED TO FILE ',I4)               TCN00740
        WRITE(NPRT,306)SITE,DAY,MON,YEAR                                       TCN00750
  306   FORMAT(' SITE = ',I5,5X,'DAY=',I5,5X,'MONTH=',I5,5X,                  TCN00760
      *          'YEAR=',I5)                                                  TCN00770
        IF(PRTKEY.EQ.1) WRITE(NPRT,600)                                        TCN00780
       CALL FLDINT(BLOCK,FETVEC,NOFEAT)                                        TCN00790
```

```
FILE: TCN      FORTRAN  A

  800  FORMAT(' THE 209 DOT LABELS')                                          TCN00800
   20  CONTINUE                                                               TCN00810
       DO 801 J=1,3                                                           TCN00820
       IREC=IREC+1                                                            TCN00830
       IF(IREC.GT.BLOCK(2)) GO TO 50                                          TCN00840
       CALL LINERD(IDATA1,ENDTAP)                                             TCN00850
C      IF(ENDTAP.EQ.-1) GO TO 50                                              TCN00860
       DO 800 I=1,GTREC                                                       TCN00870
       JJ=(J-1)*72                                                            TCN00880
       IDATA(JJ)=IDATA1(I)                                                    TCN00890
  800  CONTINUE                                                               TCN00900
  801  CONTINUE                                                               TCN00910
       NEOF=0                                                                 TCN00920
       NREC=NREC+1                                                            TCN00930
       OSGG=OSG                                                               TCN00940
       CALL GTUNPK(IDATA,ONE,OUTPX,OSGG,GTNRPD,GTREC)                         TCN00950
       G3=GTREC*3                                                             TCN00960
C      IF(NREC.LE.2) WRITE(NPRT,850) (IDATA(K),K=1,G3)                        TCN00970
  850  FORMAT(1H ,15I4)                                                       TCN00980
       CALL LINLAR(IDATA,GTREC)                                               TCN00990
       N10=MOD(NREC,10)                                                       TCN01000
C      IF(NREC.EQ.1) WRITE(NPRT,850) (IDATA(K),K=1,GTREC)                     TCN01010
       CALL WOTLN(IDATA,ZERO)                                                 TCN01020
       IF(PRTKEY.EQ.1.AND.N10.EQ.0) WRITE(NPRT,839) (IUATA(I),I=10,190        TCN01030
     *  ,10)                                                                  TCN01040
  839  FORMAT(1H ,5X,19I5)                                                    TCN01050
       GO TO 20                                                               TCN01060
   50  CONTINUE                                                               TCN01070
       ENDFILE GTWDU                                                          TCN01080
       WRITE(NPRT,700) NREC                                                   TCN01090
  700  FORMAT(' NUMBER OF SCAN LINES WRITTEN=',I5)                            TCN01100
       WRITE(NPRT,750) GTROF,GTWPF                                            TCN01110
  750  FORMAT(' FILE ',I5,5X,'WAS CONVERTED ONTO FILE ',I5)                   TCN01120
       GTROF=GTROF+1                                                          TCN01130
       ROF=ROF+1                                                              TCN01140
       GTWPF=GTWPF+1                                                          TCN01150
       IF(FILE.LT.GTNOF) GO TO 10                                             TCN01160
       ENDFILE GTWDU                                                          TCN01170
       REWIND GTRDU                                                           TCN01180
       REWIND GTWDU                                                           TCN01190
       WRITE(NPRT,250)                                                        TCN01200
  250  FORMAT(' PROGRAM GTTCN RUN COMPLETED')                                 TCN01210
       RETURN                                                                 TCN01220
       END                                                                    TCN01230
```

ORIGINAL PAGE IS
OF POOR QUALITY

23. TESTSP PROCESSOR

FILE: TESTSP FORTRAN A

```
      SUBROUTINE TESTSP(ARRAY,TOP)                                          TES00010
C*                                                                          TES00020
C***********************************************************************    TES00030
C*                                                                          TES00040
C*    THIS PROGRAM PERFORMS A MODIFIED VERSION OF THE CLUSTERING            TES00050
C*    ALGORITHM (ISODATA) ORIGINALLY DEVELOPED BY BALL AND HALL OF          TES00060
C*    STANFORD RESEARCH INSTITUTE.  THE ALGORITHM HAS BEEN MODIFIED         TES00070
C*    ON THE RECOMMENDATIONS OF ED KAN (LEC).                               TES00080
C*                                                                          TES00090
C*    THE PROGRAM EXPECTS MULTISPECTRAL SCANNER DATA                        TES00100
C*    IN EITHER THE LARSYS 22 OR THE UNIVERSAL                              TES00110
C*    FORMAT.  THE DATA TAPE SHOULD BE ASSIGNED TO FORTRAN UNIT 3.          TES00120
C*                                                                          TES00130
C***********************************************************************    TES00140
C*                                                                          TES00150
      IMPLICIT INTEGER (A-K)                                                TES00160
C     INCLUDE COMMKS.LIST                                                   TES00170
C     INCLUDE CONNTS.LIST                                                   TES00180
C     INCLUDE COMMKA.LIST                                                   TES00190
C     INCLUDE CMMK16.LIST                                                   TES00200
      COMMON/PASS/STOP,LNCAT,NMIN,KMN,STDMAX,DLMIN,SEP,                     TES00210
     *                  MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,           TES00220
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IHEGIN,REGIN1,                          TES00230
     *              REGIN2,REGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,           TES00240
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT             TES00250
     * ,NXTCLS,NOFEAT,MARCLS,FETVEC(30),SYMMTX(62)                          TES00260
     *,VANS12,STATKY,ISOKEY,MAPENT,MAPKEY,SEQUEN(20),PERCEN,SIMERR         TES00270
     *,IOWDS,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)          TES00280
     * ,NOCHAN,PRCOMP,NOSEQ,MEANDO,MEANDU,                                 TES00290
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                    TES00300
     *  DUFLAG,DODO,STDOTS(40),NSDOTS,SUNCOR(30),LLNCAT,                    TES00310
     * DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)                TES00320
     *  ,MXFFTI,MAXPOP                                                      TES00330
      REAL SUNCOR                                                           TES00340
C*                                                                          TES00350
C*    COMMON BLOCK 'PASS' IS USED ONLY BY THE ISOCLS PROCESSOR.             TES00360
C*                                                                          TES00370
C*    ISOCLS USES THE RANDOM ACCESS DRUM FILE AS FOUR DISTINCT FILES.       TES00380
C*    SEE DEFINITIONS OF IHEGIN,REGIN1,REGIN2,REGIN3 BELOW                  TES00390
C*                                                                          TES00400
C*    DEFINITIONS                                                           TES00410
C*                                                                          TES00420
C*       ISTOP  - MAX. NO. OF ITERATIONS FOR THE CLUSTERING PROCEDURE       TES00430
C*                SET IN SETUP7 ROUTINE. (USER INPUT)                       TES00440
C*       LNCAT  - CURRENT NO. OF CLUSTERS. SET INITIALLY IN PROFILE OR IS   TES00450
C*                ISOCLS, THEN ONLY IN ISODAT.                              TES00460
C*       NMIN   - MIN. NO. OF POINTS TO ALLOW PER CLUSTER                   TES00470
C*                SET IN SETUP7 ROUTINE. (USER INPUT)                       TES00480
C*       KMN    - PRINT CLUSTER SUMMARY EVERY 'KMN' ITERATION(S)            TES00490
C*                SET IN SETUP7 ROUTINE. (USER INPUT)                       TES00500
C*       STDMAX - STANDARD DEVIATION FOR SPLITTING CLUSTERS                 TES00510
C*                SET IN SETUP7 ROUTINE. (USER INPUT)                       TES00520
C*       DLMIN  - MIN. DISTANCE BETWEEN CLUSTERS FOR COMBINING.             TES00530
C*       SEP    - DISTANCE TO SEPARATE CLUSTERS.  SET EITHER IN SETUP7,     TES00540
C*                BY USER INPUT, OR IN IO                                   TES00550
C*                BY USER INPUT, OR IN ISODAT.                              TES00560
C*       MAP    - PRINT A CLUSTER MAP EVERY 'MAP' ITERATION(S)- SETUP7      TES00570
C*       SPTRIG - TRIGGER TELLING WHETHER OR NOT 'SEP' WAS INPUT. -SETUP7   TES00580
C*       IRD    - NO. OF RECORDS TO READ FROM DATA FILE. COMPUTED IN        TES00590
C*                ISOCLS                                                    TES00600
C*       NOPTS  - NO. OF POINTS IN EACH RECORD. COMPUTER IN ISOCLS          TES00610
      CONTINUE                                                              TES00620
C*       KPTS   - NO. OF POINTS IN LAST RECORD. COMPUTER IN ISOCLS          TES00630
C*       PUNCH  - TRIGGER TELLING WHETHER OR NOT TO PUNCH THE MODULE        TES00640
C*                STAT DECK. - SETUP7                                       TES00650
C*       ICHN   - TRIGGER TELLING WHETHER OR NOT CHAINING IS TO BE DONE     TES00660
C*       CHNTHS - MIN. DISTANCE BETWEEN CLUSTERS FOR CHAINING - SETUP7      TES00670
C*       ICHAIN - ARRAY CONTAINING CHAINED CLUSTER NUMBERS. SET IN          TES00680
C*                'CHAIN' ROUTINE.                                          TES00690
C*       NWDS   - TOTAL NO. OF WORDS AVAILABLE FOR DRUM STORAGE OF          TES00700
C*                IMAGE DATA TO BE CLUSTERED - SET IN ISOCLS                TES00710
C*       IHEGIN - BEGINNING DRUM FILE ADDRESS FOR INPUT INITIAL CLUSTER     TES00720
C*                CENTERS - SET IN ISOCLS                                   TES00730
C*       REGINS - BEGINNING DRUM FILE ADDRESS FOR TEMPORARY STORAGE OF      TES00740
C*                CLASS STATISTICS - SET IN ISOCLS ROUTINE                  TES00750
C*       REGIN1 - BEGINNING DRUM FILE ADDRESS FOR IMAGE DATA                TES00760
C*       REGIN2 - BEGINNING DRUM FILE ADDRESS FOR 'IMLACE' .(CLUSTER TO     TES00770
C*                WHICH CORRESPONDING POINT BELONGS.)                       TES00780
C*       CLSNAM - NAME OF CLASS CURRENTLY BEING PROCESSED. - ADDATA         TES00790
```

FILE: TESTSP FORTRAN A

```
C*       NOFLD  - NO. OF FIELDS INPUT FOR THIS CLASS - RDDATA                TES00800
C*       IPT    - NO. OF WORDS OF STORAGE USED IN 'ARRAY' FOR FIELD AND      TES00810
C*                CLASS INFORMATION FOR THIS CLASS. - RDDATA                 TES00820
C*       TOTWRD - TOTOL WORDS WRITTEN ON DRUM FILE BEGINNING AT ADDRESS      TES00830
C*                BEGIN1 - RDDATA                                            TES00840
C*       TOTPTS - TOTOL POINTS TO BE CLUSTERED FOR CURRENT CLASS - RDDATTES00850
C*       NCLASS - NO. OF CLASSES TO BE CLUSTERED FOR CURRENT CALL TO         TES00860
C*                ISOCLS. USER INPUT - SETUP7.                               TES00870
C*       NOCLS  - CURRENT CLASS NO. - ISOCLS                                 TES00880
C*       TOTSUB - TOTAL CLUSTERS FOR THIS CALL TO ISOCLS                     TES00890
C*       TOTFLD - TOTAL FIELDS FOR ALL CLASSES - ISOCLS                      TES00900
C*       TOTVRT - TOTAL VERTICES FOR ALL FIELDS - ISOCLS                     TES00910
C*       NOCL   - NO. OF CLASSES SINCE LAST CALL TO SETUP - RDDATA           TES00920
C*       NVRT   - NO.OF VERTICES FOR ALL FIELDS IN CURRENT CLASS             TES00930
C*       NXTCLS - NAME OF NEXT CLASS  (RDDATA)                               TES00940
C*       MAPFMT - FORMAT OF OUTPUT MAP TAPE                                  TES00950
C*       NOFEAT - NO. OF CHANNELS USED IN CLUSTERING                         TES00960
C*       MAXCLS - MAX. NO. OF CLUSTERS PER CLASS                             TES00970
C*       FETVEC - CHANNELS USED IN CLUSTERING                                TES00980
C*       SYMMTX - SYMBOLS FOR PRINTING MAP                                   TES00990
C*       VARSIZ - SIZE OF EACH CLUSTER'S COVARIANCE MATRIX                   TES01000
C*       STATKY - FLAG FOR PRINTING COVARIANCES.                             TES01010
C      MAPKEY-INDICATES WHETHER TO USE MEAN VECTOR OR CLUSTER NUMBER         TES01020
C            IN MAPTAPE                                                      TES01030
C      SEQUEN-ARRAY CONTAINING SEQUENCE OF 'S' AND 'C' CHARACTERS FOR        TES01040
C            SPLIT-COMBINE ITERATION CONTROL.                                TES01050
C      PERCEN-CUT OFF PERCENTAGE FOR STABILIZED CLUSTERS IN INITIAL          TES01060
C            SEQUENCE.                                                       TES01070
C      SIMERP-TRIGGER FOR SIMULATING ERIPS CLUSTERING ALGORITHM.             TES01080
C            SIMERP=1    SIMULATE ERIPS-SIMERP=0    STANDARD ISOCLS         TES01090
C      ORDER--TRIGGER FOR ORDERING THE COLOR KEYS BASE ON THE RANKING        TES01100
C            OF GREENNESS OF EACH CLUSTER AS COMPUTED BY THE SUBROUTINETES01110
C            RANK.SET IN SETUP7.                                             TES01120
C*     INUNIT-UNIT NUMBER FOR INPUT STATISTICS FILE.                         TES01130
C*     INFILE-FILE NUMBER FOR INPUT STATISTICS FILE.                         TES01140
C*     INITM--TRIGGER FOR INITIAL MEANS FROM STATISTICS FILE.                TES01150
C*     PMIN---POPULATION MINIMUM FOR EACH CLUSTER.                           TES01160
C*           NOFEAT.                                                         TES01170
C*                                                                           TES01180
C                                                                            TES01190
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,             TES01200
     *          HISFIL,HISKEY,TRFORM,ERIPTP,ERPKEY,MAPUNT,NOFILE,           TES01210
     *    DRUMAD,DRM.DS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                     TES01220
     *  ,NHSTUN,NHSTFI,SCTPUN,MAPFIL                                         TES01230
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                   TES01240
     *  CRDUNT,PRTUNT,RANDIO                                                 TES01250
      COMMON/ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP           TES01260
C$END                                                                        TES01270
      DIMENSION KVAR(11500)                                                  TES01280
      KVARDM = 11500                                                         TES01290
      DIMENSION ARRAY(1)                                                     TES01300
      DIMENSION COVAR(465)                                                   TES01310
      DIMENSION NN(60)                                                       TES01320
      DATA SYMDA /'*   '/,SYMDB /'#   '/                                     TES01330
      MAXPOP=62                                                              TES01340
      MXFET1=30                                                              TES01350
      IBEGIN=DRUMAD                                                          TES01360
C                                                                            TES01370
C*    RESERVE ENOUGH DRUM STORAGE FOR MAXIMUM INITIAL MEANS                  TES01380
C*                                                                           TES01390
      BEGIN3=IBEGIN + MAXPOP*MXFET1 + MXFET1 + 2                             TES01400
C                                                                            TES01410
C*    CALL SETUP TO READ CARD INPUT AND INITIALIZE DEFAULT VALUES            TES01420
C*                                                                           TES01430
      ITIME=1                                                                TES01440
      NOCLS = 0                                                              TES01450
      TOTFLD = 0                                                             TES01460
      TOTVRT = 0                                                             TES01470
      TOTSUB = 0                                                             TES01480
      CORRAS=1                                                               TES01490
      ITRIGU = 0                                                             TES01500
      ITRIGO=0                                                               TES01510
      SYMDO = SYMDA                                                          TES01520
      SYMDU = SYMDB                                                          TES01530
      MEANDO = 0                                                             TES01540
      MEANDU = 255                                                           TES01550
    1 CALL SETUP7(ARRAY(CORRAS),TOP,ITIME)                                   TES01560
      IDUM = MAXCLS                                                          TES01570
      IF(ITIME.GT.1)GO TO 2                                                  TES01580
```

FILE: TESTSP FORTRAN A

```
      VARSIZ=NOFEAT*(NOFEAT+1)/2                                        TES01590
      BEGIN1 = BEGIN3 + NCLASS*MAXPOP*(VARSIZ + NOFEAT + 1)             TES01600
      NWDS=DRMWDS-(BEGIN1-DRUMAD)                                       TES01610
    2 ITIME=ITIME+1                                                     TES01620
      NOCL=0                                                            TES01630
C*                                                                      TES01640
C     CALL RDDATA  TO COORDINATE READING OF DATA                        TES01650
C*                                                                      TES01660
    5 MAXDIM = TOP-COPBAS                                               TES01670
      FD1=COPBAS                                                        TES01680
      CALL RDDPAT(FD1,MAXDIM,KVAR,KVARDM,LAST)                          TES01690
      MAXCLS = IDUM + DDDU                                              TES01700
      WRITE(6,210) NDDU(1),NDDU(2)                                      TES01710
  210 FORMAT(1X,//'  DD/DU CLUSTER POP FOR THIS CLASS ',2I7)            TES01720
          BEGIN2 = BEGIN1 + (TOTWRD/4) + 2                              TES01730
      N1 = FD1 + IPT                                                    TES01740
      MEANS1=N1 + MAXCLS                                                TES01750
      STDEV1=MEANS1 + MAXCLS*NOFEAT                                     TES01760
      TTOP = STDEV1 + MAXCLS*NOFEAT                                     TES01770
      MAXDIM=TOP-TTOP                                                   TES01780
      NOPTS = MAXDIM/(NOFEAT+1)                                         TES01790
         NOPTS = (NOPTS/4)*4                                            TES01800
C        PART OF PACKING CHG SEPT 1978                                  TES01810
      IDAT1 = TTOP                                                      TES01820
      IF (NSDOTS.EQ.0) GO TO 4                                          TES01830
      DOTDMF = NOCHAN                                                   TES01840
      TYPSWT = 1                                                        TES01850
      CALL RDDOTS(ARRAY(MEANS1),STDOTS,NSDOTS,                          TES01860
     *  TYPSWT,DOTDMF,DOTDMC,DOTDUM,COVAR,                              TES01870
     *  NOCHAN,CHNVC,DTDM,COVAR,                                        TES01880
     * DOTDM,DOTDM,DOTDM,DOTDM,DOTDM,DOTDM,KVAR)                        TES01890
      LNCAT = NSDOTS                                                    TES01900
      DO 500 I = 1,NSDOTS                                               TES01910
      DO 500 K = 1,NOFEAT                                               TES01920
      III = (I-1)*NOFEAT + K                                            TES01930
      II = III + MEANS1 - 1                                             TES01940
  500  ARRAY(II) = KVAR(III)                                            TES01950
      IF (NOCHAN.EQ.NOFEAT) GO TO 8                                     TES01960
      WRITE(6,110)                                                      TES01970
  110 FORMAT(1H ,'NO CHANNELS FOR STARTING NOT EQUAL THAT FOR CLUSTER') TES01980
      GO TO 9                                                           TES01990
    4  CONTINUE                                                         TES02000
       IF(ISOKEY.EQ.1)GO TO 7                                           TES02010
C*     SUBVEC-SUBCLASSES FROM STATISTICS FILE FOR INTIAL MEANS.         TES02020
C*     NOSUB2-NUMBER OF INITIAL MEANS.                                  TES02030
C*     CHNVEC-NUMBER OF CHANNELS FROM STATISTICS FILE. NOCHAN MUST EQUAL TES02040
       IF(INITM.EQ.1)GO TO 6                                            TES02050
       LNCAT=1                                                          TES02060
       GO TO 8                                                          TES02070
    6  LNCAT=NOSUB2                                                     TES02080
      CALL GETST(INUNIT,INFILE,ARRAY(MEANS1),DUM,NOSUB2,SUBVEC,NOCHAN   TES02090
     * ,CHNVC,ARRAY(TTOP),COVAR,0)                                      TES02100
      LNCAT = NOSUB2                                                    TES02110
       GO TO 8                                                          TES02120
    7  CONTINUE                                                         TES02130
      IF(ISOKEY.EQ.1)CALL RDFILE(ARRAY(MEANS1),ARRAY(TTOP))             TES02140
    8  CONTINUE                                                         TES02150
      IF(NOPTS.GT.0)GOTO 10                                             TES02160
      WRITE(6,100)MAXDIM                                                TES02170
  100 FORMAT(' DIMENSION LIMITS EXCEEDED IN ISOCLS BY',I6,              TES02180
     *  ' REDUCE CHANNELS OR MAX.CLUSTERS')                            TES02190
    9  CALL CMERR                                                       TES02200
   10 CONTINUE                                                          TES02210
      IRD=TOTPTS/NOPTS                                                  TES02220
      IF(MOD(TOTPTS,NOPTS).EQ.0)GO TO 20                                TES02230
      KPTS=MOD(TOTPTS,NOPTS)                                            TES02240
      IRD=IRD+1                                                         TES02250
      IF(IRD.EQ.1)NOPTS=KPTS                                            TES02260
      GO TO 25                                                          TES02270
   20 KPTS=NOPTS                                                        TES02280
   25 CONTINUE                                                          TES02290
C*                                                                      TES02300
C*    CALL ISODAT TO PERFORM CLUSTERING                                 TES02310
C*                                                                      TES02320
      A1=1                                                              TES02330
      A2=A1+ MAXCLS*NOFEAT                                              TES02340
      CLD1=A2 + MAXCLS*NOFEAT                                           TES02350
      KPLCE = NOPTS*NOFEAT + IDAT1                                      TES02360
      CALL ISOPAT(IDAT1,ARRAY(KPLCE),ARRAY(MEANS1),ARRAY(N1),           TES02370
```

```
FILE: TESTSP   FORTRAN  A

     *              ARRAY(STDEV1),KVAR (CLD1),ARRAY(FD1),KVAR(A1),          TES02380
     *              KVAR(A2))                                               TES02390
C*                                                                           TES02400
C*    CHAIN CLUSTERS WHOSE DISTANCES ARE LESS THAN DLMIN                     TES02410
C*                                                                           TES02420
      LNCAT=LNCAT+0DDU                                                       TES02430
      IF(ICHN.GT.0)CALL CHAIN(KVAR(CLD1))                                    TES02440
C*                                                                           TES02450
C*    PRINT FINAL RESULTS                                                    TES02460
C*                                                                           TES02470
      CALL PRINT(-1,ARRAY(KPLCE),ARRAY(MEANS1),ARRAY(STDEV1),                TES02480
     *           KVAR (CLD1),ARRAY(FD1),ARRAY(N1))                           TES02490
C*                                                                           TES02500
C*    CREATE MAP OUTPUT TAPE FOR PMIS DAS IF DESIRED                         TES02510
C*                                                                           TES02520
      IF(MAPFMT.GT.0)CALL DSTAPE(ARRAY(KPLCE),KVAR(1),ARRAY(MEANS1),         TES02530
     *                           ARRAY(FD1))                                 TES02540
      LNCAT=LNCAT-0DDU                                                       TES02550
C*                                                                           TES02560
C*    CALCULATE COVARIANCE MATRIX FOR EACH CLUSTER                           TES02570
C*                                                                           TES02580
      IF(VARSIZ*LNCAT.GT.KVARDM)GO TO 30                                     TES02590
      CALL COVPAT(KVAR,IDAT1,ARRAY(KPLCE),ARRAY(MEANS1),                     TES02600
     *            ARRAY(N1))                                                 TES02610
C*                                                                           TES02620
      DO 26  II=1,LNCAT                                                      TES02630
   26 NN(TOTSUB+II)    = ARRAY(N1+II-1)                                      TES02640
      TOTSUB = TOTSUB + LNCAT                                                TES02650
      NOCLS = NOCLS + 1                                                      TES02660
      TOTFLD = TOTFLD + NOFLD                                                TES02670
      TOTVRT = TOTVRT + NVRT                                                 TES02680
      ARRAY(FD1+1)=IPT + FD1                                                 TES02690
      ARRAY(FD1+2)=LNCAT                                                     TES02700
      ARRAY(FD1+3)=NOFLD                                                     TES02710
C*                                                                           TES02720
C*    WRITE STATS FOR THESE CLUSTERS ON SCRATCH FILE 18                      TES02730
C*                                                                           TES02740
      IF(NOCLS.EQ.1) ADRES=BEGIN3                                            TES02750
      IN=NOFEAT*LNCAT                                                        TES02760
      CALL RWRITE(ADRES,ARRAY(MEANS1),IN,JSTAT)                              TES02770
      ADRES=ADRES+IN                                                         TES02780
      IN=VARSIZ*LNCAT                                                        TES02790
      CALL RWRITE(ADRES,KVAR,IN,LSTAT)                                       TES02800
      ADRES=ADRES+IN                                                         TES02810
C*    WAIT FOR I/O COMPLETION                                                TES02820
   60 IF(LSTAT.EQ.1) GO TO 60                                                TES02830
C*                                                                           TES02840
C*    GO READ IN ANOTHER CLASS                                               TES02850
C*                                                                           TES02860
      CORBAS=CORBAS+IPT                                                      TES02870
      IF(LAST.NE.1)GO TO 5                                                   TES02880
      IF(NOCLS.LT.NCLASS)GO TO 1                                             TES02890
C*                                                                           TES02900
C*    NOW READ SCRATCH FILE AND STORE ON SAVTAP FILE AND PUNCH ON            TES02910
C*    CARDS IF REQUESTED.                                                    TES02920
C*                                                                           TES02930
      FLD1 = 1                                                               TES02940
      VERTX1 = FLD1 + TOTFLD*4                                               TES02950
      CLSNM1 = VERTX1 + TOTVRT*2                                             TES02960
      NOSUB1 = CLSNM1 + NOCLS                                                TES02970
      SUBNM1 = NOSUB1 + NOCLS                                                TES02980
C                                                                            TES02990
C     RETRIEVE INFORMATION FROM 'ARRAY'                                      TES03000
C                                                                            TES03010
      CALL GETINF(ARRAY(1),KVAR(FLD1),KVAR(VERTX1),KVAR(CLSNM1),             TES03020
     *     KVAR(NOSUB1),KVAR(SUBNM1),NOCLS,TOTSUB)                          TES03030
C                                                                            TES03040
      SWTCH = 1                                                              TES03050
C                                                                            TES03060
C     OUTPUT STATS                                                           TES03070
C                                                                            TES03080
      CALL LARMAN(SAVTAP,STAFIL,NOCLS,TOTSUB,NOFEAT,TOTFLD,TOTVRT,           TES03090
     *   FETVEC,KVAR(FLD1),KVAR(VERTX1),KVAR(CLSNM1),KVAR(NOSUB1),          TES03100
     *  KVAR(SUBNM1),NN,BEGIN3,VARSIZ,PUNCH,DUMMY,STATKY,SWTCH)               TES03110
      RETURN                                                                 TES03120
   30 KV=KVARDM                                                              TES03130
      WRITE(6,200)KV                                                         TES03140
      CALL CMFSR                                                             TES03150
  200 FORMAT(' DIMENSION LIMIT OF',I6,' FOR COVARIANCES EXCEEDED')           TES03160
```

FILE: TESTSP FORTRAN A

```
      RETURN                                                        TES03170
      END                                                           TES03180
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
FILE: COVPAT   FORTRAN  A

      SUBROUTINE COVPAT(COVAR,IDAT1,IPLACE,MEANS,N)                          COV00010
      IMPLICIT INTEGER (A-Z)                                                 COV00020
           COMMON ARRAY(10600)                                               COV00030
           LOGICAL*1 LARRAY(42400)                                           COV00040
           EQUIVALENCE (ARRAY,LARRAY)                                        COV00050
           LOGICAL*1 LPACK(4)                                                COV00060
           EQUIVALENCE (LPACK,IPACK)                                         COV00070
C*                                                                           COV00080
C*    SUBROUTINE COVARM CALCULATES AND PRINTS THE COVARIANCE MATRIX FOR       COV00090
C*    EACH CLUSTER                                                           COV00100
C*                                                                           COV00110
C     INCLUDE COMMKS.LIST                                                    COV00120
C     INCLUDE COMMK6.LIST                                                    COV00130
      COMMON/PASS/STOP,LNCAT,UMIN,KRN,STDMAX,DLMIN,SEP,                      COV00140
     *                        MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,       COV00150
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                           COV00160
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,            COV00170
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT              COV00180
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                           COV00190
     *,VARSIZ,STATKY,ISUKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP          COV00200
     *,IORDER,IUNIT,INFILE,INITM,PMIN,SURVEC(62),NOSUB2,CHNVC(30)           COV00210
     * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                                  COV00220
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                     COV00230
     *  DUFLAG,DODO,STOPTS(60),NSOPTS,SUNCOR(30),LLNCAT,                     COV00240
     * OVERT(2=0,2),OVECT(60,2),OVPNT(11,2),IDCNT(2),NDOU(2)                 COV00250
     *  ,MXFEIL,MAXPCP                                                       COV00260
      REAL SUNCOR                                                            COV00270
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,              COV00280
     *              HISFIL,HISKEY,TRFORM,EPIPTP,EPPKEY,MAPUNT,NOFILE,        COV00290
     *    DRUMAD,DRMODS,PASSI/,DATFIL,STAFIL,ASAV,ASAVFL                     COV00300
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                         COV00310
     *  ,DOTUNT,DOTFIL,NCHRAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                   COV00320
     *  CRDUNT,PRTUNT,RANDIO                                                 COV00330
C$END                                                                        COV00340
      REAL MEANS,COVAR                                                       COV00350
      DIMENSION COVAR(VARSIZ,LNCAT),IPLACE(NOPTS)                            COV00360
      DIMENSION MEANS(NOFEAT,MAXCLS),N(MAXCLS)                               COV00370
      DATA CH/'CH'/                                                          COV00380
      DO 10 I=1,LNCAT                                                        COV00390
      DO 10 J=1,VARSIZ                                                       COV00400
   10 COVAR(J,I)=0.0                                                         COV00410
C*                                                                           COV00420
          REAL IDA                                                           COV00430
          IB = (IDAT1 - 1)*4                                                 COV00440
          IPACK = 0                                                          COV00450
      ADRES1=BEGIN1                                                          COV00460
      ADRES2= BEGIN2                                                         COV00470
      ICCT=NOPTS                                                             COV00480
      IRC=IRD                                                                COV00490
   20 IF(IRC.LE.1)ICCT=KPTS                                                  COV00500
      IF (IPO.EQ.0) GO TO 30                                                 COV00510
      IWPOS=ICCT*NOFEAT                                                      COV00520
          IWD4 = IWPOS/4                                                     COV00530
          IF (4*IWD4.NE.IWPOS) IWD4 = IWD4 + 1                               COV00540
          CALL RREAD(ADRES1,ARRAY(IDAT1),IWD4,ISTAT)                         COV00550
      ADRES1=ADRES1+IWD4                                                     COV00560
   22 IF(ISTAT.EQ.1)GO TO 22                                                 COV00570
      CALL RREAD(ADRES2,IPLACE,ICCT,ISTAT)                                   COV00580
      ADRES2=ADRES2+ICCT                                                     COV00590
   25 IF(ISTAT.EQ.1)GO TO 25                                                 COV00600
C*                                                                           COV00610
C*    SINCE THE COVARIANCE MATRIX IS SYMMETRICAL ONLY THE LOWER              COV00620
C*    TRIANGULAR PORTION OF THE MATRIX IS CALCULATED.                        COV00630
C*                                                                           COV00640
   30 DO 45 I = 1,ICCT                                                       COV00650
          IBASE = IB + (I - 1)*NOFEAT                                        COV00660
      KK=0                                                                   COV00670
      ICLS=IPLACE(I)                                                         COV00680
      IF(ICLS.GT.LNCAT) GO TO 45                                             COV00690
      DO 40 J=1,NOFEAT                                                       COV00700
      DO 40 K=1,J                                                            COV00710
         LPACK(4) = LARRAY(IBASE + J)                                        COV00720
         IPACK1 = IPACK                                                      COV00730
         LPACK(4) = LARRAY(IBASE + K)                                        COV00740
         IPACK1 = IPACK1*IPACK                                               COV00750
         DUM = IPACK1                                                        COV00760
      KK=KK+1                                                                COV00770
      COVAR(KK,ICLS)=COVAR(KK,ICLS)+DUM                                      COV00780
   40 CONTINUE                                                               COV00790
```

FILE: COVPAT FORTRAN A

```
 45   CONTINUE                                                          COV00800
      IRC=IRC-1                                                         COV00810
      IF (IRC.GT.0) GO TO 20                                            COV00820
      DO 50 I=1,LNCAT                                                   COV00830
      KK=0                                                              COV00840
      DO 50 J=1,NOFEAT                                                  COV00850
      DO 50 K=1,J                                                       COV00860
      KK=KK+1                                                           COV00870
      COVAR(KK,I)=COVAR(KK,I)/N(I) - MEANS(K,I)*MEANS(J,I)              COV00880
   50 CONTINUE                                                          COV00890
      RETURN                                                            COV00900
C     IF(STATKY.NE.1)RETURN                                             COV00910
   55 WRITE (6,HEAD)                                                    COV00920
      WRITE(6,150)CLSNAM                                                COV00930
      DO 80 I=1,LNCAT                                                   COV00940
      WRITE (6,90)I                                                     COV00950
      DO 70 LOC=1,NOFEAT,12                                             COV00960
      ISTOP=LOC+11                                                      COV00970
      IF(ISTOP.GT.NOFEAT)ISTOP=NOFEAT                                   COV00980
      WRITE(6,140)(CH,FETVEC(J),J=LOC,ISTOP)                            COV00990
      II=1                                                              COV01000
      KINC=1                                                            COV01010
      DO 60 J=LOC,NOFEAT                                                COV01020
      K=J*(J+1)/2-II+1                                                  COV01030
      JK=K+KINC-1                                                       COV01040
      WRITE(6,100)(COVAR(M,I),M=K,JK)                                   COV01050
      II=II+1                                                           COV01060
   60 IF(KINC.LT.ISTOP.AND.KINC.LT.12)KINC=KINC+1                       COV01070
      WRITE (6,110)                                                     COV01080
   70 CONTINUE                                                          COV01090
   80 CONTINUE                                                          COV01100
      RETURN                                                            COV01110
   90 FORMAT(//' COVARIANCE MATRIX FOR CLUSTER',I4/)                    COV01120
  100 FORMAT(/6X,12F9.2)                                                COV01130
  110 FORMAT(///)                                                       COV01140
  120 FORMAT(1H1)                                                       COV01150
  140 FORMAT(9X,12(A3,I2,')',3X))                                       COV01160
  150 FORMAT(/' COVARIANCES FOR CLASS',2X,A4//)                         COV01170
      END                                                               COV01180
```

FILE: ISOPAT FORTRAN A

```
      SUBROUTINE ISOPAT(IDAT1,IPLACE,MEANS,N,STDEV,CLD,FLDINF,AVP,AMN)   ISO00010
       IMPLICIT INTEGER (A-Z)                                            ISO00020
        IMPLICIT INTEGER (A-Z)                                           ISO00030
C       INCLUDE COMMKS.LIST                                              ISO00040
C        INCLUDE COMAK6.LIST                                             ISO00050
CMS360                                                                   ISO00060
CMS360                                                                   ISO00070
C        INCLUDE CMAK16.LIST                                             ISO00080
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                  ISO00090
     *                    MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,       ISO00100
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,HEGIN),                       ISO00110
     *              HEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,        ISO00120
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT          ISO00130
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                       ISO00140
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERP       ISO00150
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)       ISO00160
     * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                               ISO00170
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                 ISO00180
     *  DUFLAG,DODU,STOOTS(60),NSOOTS,SUNCOR(30),LLNCAT,                 ISO00190
     * DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)             ISO00200
     *  ,MXFFTI,MAXPOP                                                   ISO00210
      REAL SUNCOR                                                        ISO00220
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,          ISO00230
     *              HISFIL,HISKEY,TRFORM,TRIPTP,ERPKEY,MAPUNT,NOFILF,    ISO00240
     *     DRUMAD,DRMWDS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                ISO00250
     *   ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                    ISO00260
     *   ,DDTUNT,DDTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,              ISO00270
     *   CRDUNT,PRTUNT,RANDIO                                            ISO00280
      COMMON/ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP       ISO00290
C$END                                                                    ISO00300
      EQUIVALENCE (SGMIN,STDMAX)                                         ISO00310
       REAL MEANS,STDEV,STDMAX,SEP,AVP, AMN,SGMA,RND,                    ISO00320
     *TEST,DMIN,DLMIN,CLD,TIME,PERCEN,DIJ                                ISO00330
       REAL FSUM,ESQT,MEAN(30,62),SDIJ                                   ISO00340
      LOGICAL DEL                                                        ISO00350
      DIMENSION AVP(NOFEAT,MAXCLS),ISGMA(62)                             ISO00360
      DIMENSION AMN(NOFEAT,MAXCLS),SGMA(62)                              ISO00370
        DIMENSION IPLACE(NOPTS)                                          ISO00380
      DIMENSION MEANS(NOFEAT,MAXCLS),N(MAXCLS)                           ISO00390
      DIMENSION STDEV(NOFEAT,MAXCLS),CLD(MAXCLS,MAXCLS)                  ISO00400
      DIMENSION FLDINF(1)                                                ISO00410
       REAL SDUM                                                         ISO00420
      DIMENSION PTR(62)                                                  ISO00430
      DATA SS/'S'/,CC/'C'/                                               ISO00440
      EQUIVALENCE (XDIM,NOFEAT),(LNCAT,INCAT)                            ISO00450
      DEL=.FALSE.                                                        ISO00460
       ISEQ=0                                                            ISO00470
      MAXCL = MAXCLS - DODU                                              ISO00480
      IDUM=LNCAT+DODU-MAXCLS                                             ISO00490
      IF(IDUM.GT.0) LNCAT=LNCAT-IDUM                                     ISO00500
      ISTOP=STOP                                                         ISO00510
      SPLFIN=0                                                           ISO00520
      KKT=1                                                              ISO00530
       DO 5 I = 1,30                                                     ISO00540
 5     SUNCOR(I) = 1.                                                    ISO00550
      IF (ISUNC.NE.0.OR.ISUNT.NE.0) CALL SUNFAC(SUNCOR,SUNANG,           ISO00560
     *  FETVEC,NOFEAT,ISUNC,ISUNT)                                       ISO00570
C*                                                                       ISO00580
       LK=K                                                              ISO00590
C*    ASSIGN DATA TO CLUSTERS                                            ISO00600
C*                                                                       ISO00610
   10 CONTINUE                                                           ISO00620
      LLNCAT = LNCAT + DODU                                              ISO00630
      IF(DOFLAG.EQ.0) GO TO 12                                           ISO00640
      DO 11 J=1,NOFEAT                                                   ISO00650
 11   MEANS(J,LNCAT+1)=MEANDO                                            ISO00660
 12   IF(DUFLAG.EQ.0) GO TO 14                                           ISO00670
      DO 13 J=1,NOFEAT                                                   ISO00680
 13   MEANS(J,LLNCAT) = MEANDU                                           ISO00690
 14    CONTINUE                                                          ISO00700
      DO 15 K=1,LLNCAT                                                   ISO00710
      DO 15 J=1,NOFEAT                                                   ISO00720
   15 MEAN(J,K) = MEANS(J,K)                                             ISO00730
      IF (LNCAT.LE.1.AND.KKT.GT.1) GO TO 530                             ISO00740
      CALL PSPPAT(MEANS,STDEV,N,CLD,IDAT1,IPLACE,AVP,AMN,MEANS)          ISO00750
C     CALL CLOCK(TIME)                                                   ISO00760
C     IF(MOD(KKT,KRN).EQ.0)WRITE(6,120)KKT,TIME                          ISO00770
C 120 FORMAT(' CUMULATIVE TIME AFTER ASSIGNING DATA TO CLUSTERS FOR ITER ISO00780
C    *ATION',I4,' IS',F10.6)                                             ISO00790
```

```
FILE: ISOPAT   FORTRAN  A

        IF(ERCOMP.NE.1)GO TO 135                                          ISO00800
        FSUM=0.0                                                          ISO00810
        DO 132 J=1,NOFEAT                                                 ISO00820
        DO 132 K=1,LNCAT                                                  ISO00830
        FSUM=FSUM+N(K)*(STDEV(J,K))**2/TOTPTS                             ISO00840
 132    CONTINUE                                                          ISO00850
      FSQT=SQRT(FSUM/NOFEAT)                                              ISO00860
      WRITE(6,133) FSQT,PERCEN,STDMAX                                     ISO00870
 133  FORMAT(1X,///' ERCOMP= ',F7.3,' PERCEN = ',F5.3,' STDMAX = ',      ISO00880
     *F7.3/)                                                              ISO00890
C*                                                                        ISO00900
C*     CALCULATE DISTANCES BETWEEN CLUSTER CENTERS                        ISO00910
C*                                                                        ISO00920
 135    CALL CLDIST(CLD,STDEV,MEANS)                                      ISO00930
C*                                                                        ISO00940
C*     IF STOP EQUALS ZERO RETURN AFTER INITIAL CLASSIFICATION            ISO00950
C*                                                                        ISO00960
      IF (ISTOP .EQ. 0) RETURN                                            ISO00970
      LNCAT=LLNCAT                                                        ISO00980
      IF (MOD(KKT,MAP)) 150,140,150                                       ISO00990
  140 CALL PRINT(KKT,IPLACE,MEANS,STDEV,CLD,FLDINF,N)                     ISO01000
      GO TO 161                                                           ISO01010
  150 IF (MOD(KKT,KRN))161,160,161                                        ISO01020
  160 CONTINUE                                                            ISO01030
      CALL PRINT(KKT,IPLACE,MEANS,STDEV,CLD,FLDINF,N)                     ISO01040
  161 CONTINUE                                                            ISO01050
      LNCAT=LLNCAT-0000                                                   ISO01060
C                                                                         ISO01070
CC     FOR ITERATION N CHECK N(K) AGAINST PMIN * NOFEAT                   ISO01080
C                                                                         ISO01090
      IF (ISEQ .NE. NOSEQ) GO TO 169                                      ISO01100
      ISTOP = 0                                                           ISO01110
 162    DO 163 K = 1,LNCAT                                                ISO01120
        IF (N(K) - (PMIN * NOFEAT)) 167,163,163                           ISO01130
 163    CONTINUE                                                          ISO01140
        IF (.NOT.DEL) RETURN                                              ISO01150
        DO 164 KK=1,LNCAT                                                 ISO01160
        DO 164 KKK=1,NOFEAT                                               ISO01170
 164    MEANS(KKK,KK) = MEAN(KKK,KK)                                      ISO01180
        CALL PSPPAT(MEANS,STDEV,N,CLD,IDAT1,IPLACE,AVP,AMN,MEANS)         ISO01190
        DO 165 KK=1,LNCAT                                                 ISO01200
        DO 165 KKK=1,NOFEAT                                               ISO01210
 165    MEAN(KKK,KK) = MEANS(KKK,KK)                                      ISO01220
        CALL CLDIST(CLD,STDEV,MEANS)                                      ISO01230
      RETURN                                                              ISO01240
  167 WRITE(6,168)K,N(K),PMIN,NOFEAT                                      ISO01250
  168 FORMAT(/' CLUSTER',I3,' REMOVED FOR HAVING ONLY',I6,' POINTS.'/     ISO01260
     *' MIN. POINTS IS  (',I4,' * ',I2,')')                               ISO01270
      RETF=1                                                              ISO01280
        LK=K                                                              ISO01290
      GO TO 970                                                           ISO01300
 171    K=LK                                                              ISO01310
      DEL = .TRUE.                                                        ISO01320
      GO TO 162                                                           ISO01330
  169 CONTINUE                                                            ISO01340
  170 CONTINUE                                                            ISO01350
C                                                                         ISO01360
CC     ON ITERATIONS 1 THRU N-1 CHECK N(K) AGAINST NMIN                   ISO01370
C                                                                         ISO01380
      DO 180 K=1,LNCAT                                                    ISO01390
      IF (N(K)-NMIN) 190,180,180                                          ISO01400
  180 CONTINUE                                                            ISO01410
      IF(DEL)CALL CLDIST(CLD,STDEV,MEANS)                                 ISO01420
      GO TO 220                                                           ISO01430
  190 IF (MOD(KKT,KRN)) 200,195,200                                       ISO01440
  195    WRITE (6,210)K,N(K),NMIN                                         ISO01450
 200    RETF=2                                                            ISO01460
      LK=K                                                                ISO01470
        GO TO 970                                                         ISO01480
 201    K=LK                                                              ISO01490
        DEL=.TRUE.                                                        ISO01500
      GO TO 170                                                           ISO01510
  210 FORMAT('0 CLUSTER ',I2,' REMOVED FOR HAVING  ONLY ',I6,             ISO01520
     2       ' ELEMENTS. MIN. NO. ELEMENTS IS ',I6)                       ISO01530
  220 CONTINUE                                                            ISO01540
C                                                                         ISO01550
C                                                                         ISO01560
                                                                          ISO01570
C*     SPLIT ITERATION                                                    ISO01580
```

FILE: ISOPAT FORTRAN A

```
C                                                                               ISO01590
C                                                                               ISO01600
      DO 225 I=1,INCAT                                                          ISO01610
  225  PTR(I)=I                                                                 ISO01620
      ISPLT=0                                                                   ISO01630
      DO 260 K=1,INCAT                                                          ISO01640
C*                                                                              ISO01650
C*    FIND MAXIMUM STANDARD DEVIATION PER CLUSTER                               ISO01660
C*                                                                              ISO01670
       SGMA(K) = 0.                                                             ISO01680
       DO 250 J = 1,NOFEAT                                                      ISO01690
       SDUM = STDEV(J,K)*SUNCOR(J)                                              ISO01700
       IF (SDUM - SGMA(K)) 250,240,240                                          ISO01710
  240  ISGMA(K) = J                                                             ISO01720
       SGMA(K) = SDUM                                                           ISO01730
  250 CONTINUE                                                                  ISO01740
      IF(SGMA(K).GE.STDMAX)ISPLT=ISPLT+1                                        ISO01750
  260 CONTINUE                                                                  ISO01760
      IF (2*LNCAT.GT.MAXCL) CALLDESCEN(SGMA,LNCAT,ISGMA,PTR)                    ISO01770
      TEST=FLOAT(ISPLT)/FLOAT(LNCAT)                                            ISO01780
       IF(TEST.LE.PERCEN)SPLFIN=1                                               ISO01790
      IF (KKT .GT. ISTOP) SPLFIN = 1                                            ISO01800
      IF (SPLFIN .EQ. 0) GO TO 270                                              ISO01810
      IF (MOD(KKT,KRN) .EQ. 0) WRITE(6,503)                                     ISO01820
  503 FORMAT( /)                                                                ISO01830
       IF(MOD(KKT,KRN).EQ.0)WRITE(6,502)                                        ISO01840
  502  FORMAT(1X,'USER INPUT*SPLIT-COMBINE SEQUENCE OF ITERATIONS')             ISO01850
       ISEQ=ISEQ+1                                                              ISO01860
      IF (SEQUEN(ISEQ).EQ.SS) GO TO 270                                         ISO01870
       IF (SEQUEN(ISEQ).EQ.CC) GO TO 410                                        ISO01880
C                                                                               ISO01890
CC    IS SPLITTING REQUIRED                                                     ISO01900
C                                                                               ISO01910
  270 K=1                                                                       ISO01920
      NCAT=INCAT                                                                ISO01930
  280 IF (K-NCAT) 290,290,500                                                   ISO01940
  290 IF (STDMAX-SGMA(K)) 300,300,310                                           ISO01950
  300 IF (N(K)-(NMIN+NMIN+2)) 310,310,320                                       ISO01960
  310 K=K+1                                                                     ISO01970
      GO TO 280                                                                 ISO01980
C                                                                               ISO01990
CC    SPLIT CLUSTER K                                                           ISO02000
C                                                                               ISO02010
  320 TRIG1=1                                                                   ISO02020
       DEL=.TRUE.                                                               ISO02030
      KX=ISGMA(K)                                                               ISO02040
  330 INCAT=INCAT+1                                                             ISO02050
      LLNCAT=LLNCAT+1                                                           ISO02060
      IF(LLNCAT.LE.MAXCLS) GO TO 350                                            ISO02070
      IF(MOD(KKT,KRN).EQ.0)WRITE(6,340)KKT                                      ISO02080
  340 FORMAT(/' MAXIMUM CLUSTERS ON ITERATION',I4/' SPLITTING REQUIRED H ISO02090
     *UT NOT PERFORMED'/)                                                       ISO02100
      LNCAT = MAXCL                                                             ISO02110
      LLNCAT=MAXCLS                                                             ISO02120
      GO TO 500                                                                 ISO02130
  350 INC=INCAT                                                                 ISO02140
       LL=PTR(K)                                                                ISO02150
  360 DO 370 I=1,KDIM                                                           ISO02160
  370  AMN(I,INC)=AMN(I,LL)                                                     ISO02170
  380  AMN(KX,LL)=AMN(KX,LL)+SEP*SGMA(K)                                        ISO02180
       AMN(KX,INC)=AMN(KX,INC)-SEP*SGMA(K)                                      ISO02190
      SGMA(K)=0.0                                                               ISO02200
      IF (MOD(KKT,KRN)) 400,401,400                                             ISO02210
  401  WRITE(6,390)LL,KX,INC                                                    ISO02220
  390 FORMAT('0 CLUSTER ',I2,' IS SPLIT IN THE ',I2,'TH PARAMETER INTO C ISO02230
     *LUSTER ',I2)                                                              ISO02240
  400 CONTINUE                                                                  ISO02250
      K=K+1                                                                     ISO02260
      GO TO 280                                                                 ISO02270
C                                                                               ISO02280
CC    EVEN ITERATION                                                            ISO02290
CC                                                                              ISO02300
CC    ARE CLUSTERS TO BE COMBINED                                               ISO02310
C                                                                               ISO02320
  410 CONTINUE                                                                  ISO02330
       DO 405 L=1,LNCAT                                                         ISO02340
  405  PTR(L)=1                                                                 ISO02350
C                                                                               ISO02360
       NOCOMB=0                                                                 ISO02370
```

FILE: ISOPAT FORTRAN A

```
      NOCLST=LNCAT-1                                                    ISO02380
      L=-1                                                              ISO02390
  404 L=L+2                                                             ISO02400
      IF (L.GT.NOCLST) GO TO 480                                        ISO02410
      NOCLTR = LNCAT - 1                                                ISO02420
      KK=0                                                              ISO02430
      DMIN=DLMIN                                                        ISO02440
      DO 430 I=1,NOCLTR                                                 ISO02450
C                                                                       ISO02460
      IF(PTR(I).EQ.0)GO TO 430                                          ISO02470
      II=I+1                                                            ISO02480
      DO 425 J=II,LNCAT                                                 ISO02490
      IF(PTR(J).EQ.0)GO TO 425                                          ISO02500
      SDIJ = 0.0                                                        ISO02510
      DO 420 JJ=1,KDIM                                                  ISO02520
      SDIJ=SDIJ+((AMN(JJ,I)-AMN(JJ,J))**2/(STDEV(JJ,I)*STDEV(JJ,J)))   ISO02530
  420 CONTINUE                                                          ISO02540
      DIJ=SQRT(SDIJ)                                                    ISO02550
C                                                                       ISO02560
      IF(DIJ.GT.DMIN)GO TO 425                                          ISO02570
      DMIN=DIJ                                                          ISO02580
      KK=I                                                              ISO02590
      KKK=J                                                             ISO02600
  425 CONTINUE                                                          ISO02610
  430 CONTINUE                                                          ISO02620
C                                                                       ISO02630
      IF(KK.EQ.0)GO TO 480                                              ISO02640
      PTR(KK)=0                                                         ISO02650
C                                                                       ISO02660
C       COMBINE CLUSTERS KK AND KKK                                     ISO02670
C                                                                       ISO02680
      DEL=.TRUE.                                                        ISO02690
      RND=1.0 /FLOAT(N(KK)+N(KKK))                                      ISO02700
C                                                                       ISO02710
      DO 460 K=1,KDIM                                                   ISO02720
  460 AMN(K,KK)=(N(KK)*AMN(K,KK)+N(KKK)*AMN(K,KKK))*RND                 ISO02730
C                                                                       ISO02740
      PETF=3                                                            ISO02750
      LK=KKK                                                            ISO02760
      GO TO 570                                                         ISO02770
  461 KKK=LK                                                            ISO02780
      IF (KKK.EQ.(LNCAT+1)) GO TO 435                                   ISO02790
C                                                                       ISO02800
C     MOVE POINTERS UP                                                  ISO02810
C                                                                       ISO02820
      DO 175 K=KKK,LNCAT                                                ISO02830
  175 PTR(K) = PTR(K+1)                                                 ISO02840
C                                                                       ISO02850
  435  IF(MOD(KKT,KRN))440,441,440                                      ISO02860
  441    WRITE(6,490)KK,KKK,KK                                          ISO02870
  440  IF (L.LT.NOCLST) GO TO 406                                       ISO02880
C                                                                       ISO02890
  480  CONTINUE                                                         ISO02900
  490 FORMAT(' CLUSTERS ',I2,' AND ',I2,' HAVE BEEN COMBINED INTO CLUST ISO02910
     2ER ',I2)                                                          ISO02920
C*                                                                      ISO02930
C*    REINITIALIZE                                                      ISO02940
C*                                                                      ISO02950
  500 CONTINUE                                                          ISO02960
      DO 510 J=1,MAXCLS                                                 ISO02970
      SGMA(J)=0.0                                                       ISO02980
      ISGMA(J)=0                                                        ISO02990
      DO 510 K=1,KDIM                                                   ISO03000
      AVP(K,J)=0.0                                                      ISO03010
      STDEV(K,J)=0.0                                                    ISO03020
      MEANS(K,J)=AMN(K,J)                                               ISO03030
      AMN(K,J)=0.0                                                      ISO03040
  510 CONTINUE                                                          ISO03050
      KKT=KKT+1                                                         ISO03060
      DEL=.FALSE.                                                       ISO03070
      GO TO 10                                                          ISO03080
C                                                                       ISO03090
  530 IF (KKT.NE.2) GO TO 550                                           ISO03100
      WRITE (6,540)                                                     ISO03110
  540 FORMAT(' THE ORIGINAL CLUSTER WAS NOT SPLIT - EXAMINE THE INPUT VA ISO03120
     *LUE FOR STUMAX'/)                                                 ISO03130
      KKT=1                                                             ISO03140
      ISTOP=0                                                           ISO03150
      GO TO 10                                                          ISO03160
```

FILE: ISOPAT FORTRAN A

```
  550 WRITE (6,560)KKT                                                    ISO03170
  560 FORMAT(///' AFTER ',I4,' ITERATIONS ALL DATA HAS BEEN ASSIGNED TO O ISO03180
     *NE CLUSTER'/)                                                       ISO03190
      KKT=1                                                               ISO03200
      ISTOP=0                                                             ISO03210
      GO TO 10                                                            ISO03220
  570 CONTINUE                                                            ISO03230
C*                                                                        ISO03240
C       ROUTINE TO DELETE A CLUSTER                                       ISO03250
C*                                                                        ISO03260
      INCAT=INCAT-1                                                       ISO03270
      LLNCAT=LLNCAT-1                                                     ISO03280
      IF (LK.EQ.(INCAT+1)) GO TO (171,201,461),RETF                       ISO03290
       DO 561 J=LK,INCAT                                                  ISO03300
       DO 562 L=1,KOUN                                                    ISO03310
      AMN(L,J)=AMN(L,J+1)                                                 ISO03320
      MEANS(L,J)=MEANS(L,J+1)                                             ISO03330
      MEAN(L,J) = MEAN(L,J+1)                                             ISO03340
  562  STDEV(L,J)=STDEV(L,J+1)                                            ISO03350
      N(J)=N(J+1)                                                         ISO03360
  561  CONTINUE                                                           ISO03370
       GO TO (171,201,461),RETF                                           ISO03380
      END                                                                 ISO03390
```

ORIGINAL PAGE IS
OF POOR QUALITY

FILE: PSPPAT FORTRAN A

```
      SUBROUTINE PSPPAT(MEANS,STDEV,N,CLD,IDAT1,IPLACE,AVP,AMN,MEN)      PSP00010
      IMPLICIT INTEGER (A-Z)                                             PSP00020
        COMMON ARRAY(10600)                                              PSP00030
        LOGICAL*1 LARRAY(42400)                                          PSP00040
        EQUIVALENCE (LARRAY,ARRAY)                                       PSP00050
        LOGICAL*1 LPACK(4)                                               PSP00060
        EQUIVALENCE (IPACK,LPACK)                                        PSP00070
C                                                                        PSP00080
C       INCLUDE COMRKS.LIST                                              PSP00090
C       INCLUDE CMRK16.LIST                                              PSP00100
      COMMON/PASS/STOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                  PSP00110
     *                          MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,  PSP00120
     *  ICHN,CHNTMS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                       PSP00130
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,        PSP00140
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NUCL,NVHT          PSP00150
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                       PSP00160
     *,VARS17,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR       PSP00170
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SURVEC(62),NOSUB2,CHNVC(30)       PSP00180
     * ,NOCHAN,ENCOMP,NOSEQ,MEANDO,MEANDU,                               PSP00190
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                 PSP00200
     *  DUFLAG,DODU,STOOTS(50),NSUOTS,SUNCUR(30),LLNCAT,                 PSP00210
     * DVERT(250,2),OBJECT(60,2),OVENT(11,2),IOCNT(2),NODU(2)            PSP00220
     *  ,MXFFTI,MAXPOP                                                   PSP00230
      REAL SUNCOM                                                        PSP00240
      COMMON/ISOLNK/SUNANG(8),ISUNT,ISUNC,SMSTR,SMSTP,SMINC,LINSKP       PSP00250
CSEND                                                                    PSP00260
      DIMENSION  IPLACE(NOPTS),AMN(NOFEAT,MAXCLS)                        PSP00270
      DIMENSION STDEV(NOFEAT,MAXCLS),CLD(MAXCLS,MAXCLS),N(MAXCLS)        PSP00280
      DIMENSION AVP(NOFEAT,MAXCLS),MEANS(NOFEAT,MAXCLS)                  PSP00290
      REAL MEN(NOFEAT,MAXCLS)                                            PSP00300
      REAL AMN,STDEV,AVP,SDIST,DIST,RND,MEANS                            PSP00310
      DIMENSION CSUN(30)                                                 PSP00320
      REAL CSUN                                                          PSP00330
        IB = (IDAT1 - 1)*4                                               PSP00340
        IPACK = 0                                                        PSP00350
        REAL DUM,DUMA                                                    PSP00360
        DUM = .00001                                                     PSP00370
        ICOUNT = 0                                                       PSP00380
        IDELET = 0                                                       PSP00390
 2      ICOUNT = ICOUNT + 1                                              PSP00400
        IDELET = 0                                                       PSP00410
C                                                                        PSP00420
      IF(DOFLAG.NE.0) N(LNCAT+1)=NODU(DOFLAG)                            PSP00430
      IF(DUFLAG.NE.0) N(LNCAT+DODU)=NODU(DODU)                           PSP00440
       DO 5 I=1,LNCAT                                                    PSP00450
       N(I)=0                                                            PSP00460
       DO 5 J=1,NOFEAT                                                   PSP00470
       AMN(J,I)=0.0                                                      PSP00480
      STDEV(J,I) = 0.0                                                   PSP00490
 5     AVP(J,I)=0.0                                                      PSP00500
C*                                                                       PSP00510
C*      ASSIGN DATA TO CLUSTERS                                          PSP00520
C*                                                                       PSP00530
       ADDRS1=BEGIN1                                                     PSP00540
       ADDRS2=BEGIN2                                                     PSP00550
       ICCT=NOPTS                                                        PSP00560
       IRC=IRD                                                           PSP00570
 20    IF(IRC.LE.1)ICCT=KPTS                                             PSP00580
       IF(IRD.EQ.0)GO TO 40                                              PSP00590
       IWDS=NOFEAT*ICCT                                                  PSP00600
         IW04 = IWDS/4                                                   PSP00610
         IF (4*IW04.NE.IWDS) IW04 = IW04 + 1                             PSP00620
         CALL MREAD(ADDRS1,ARRAY(IDAT1),IW04,ISTAT)                      PSP00630
       ADDRS1=ADDRS1+IW04                                                PSP00640
 25    IF(ISTAT.EQ.1)GO TO 25                                            PSP00650
       IF(ISTAT.EQ.0)GO TO 40                                            PSP00660
       IF(ISTAT.GT.0)GO TO 40                                            PSP00670
       WRITE(6,30)ISTAT                                                  PSP00680
 30    FORMAT(' ERROR READING DRUM---ISTAT=',I4)                         PSP00690
 40    CONTINUE                                                          PSP00700
      IF (ISUNT.EQ.0.AND.ISUNC.EQ.0) GO TO 50                            PSP00710
      DO 49 I=1,ICCT                                                     PSP00720
        IBASE = (I - 1)*NOFEAT + IB                                      PSP00730
      IF(DODU.EQ.0) GO TO 42                                             PSP00740
      DO 41 K=1,NOFEAT                                                   PSP00750
        LPACK(4) = LARRAY(IBASE + K)                                     PSP00760
        CDUM = IPACK                                                     PSP00770
      IF (CDUM.NE.MEANDO.AND.CDUM.NE.MEANDU) GO TO 42                    PSP00780
 41   CONTINUE                                                           PSP00790
```

FILE: PSPPAT FORTRAN A

```
      IF (CDUM.EQ.MEANDO) IPLACE(I) = LNCAT + 1                          PSP00800
      IF (CDUM.EQ.MEANDU) IPLACE(I) = LNCAT + DODU                       PSP00810
      GO TO 49                                                           PSP00820
   42 KK=1                                                               PSP00830
      SDIST=10.0E+20                                                     PSP00840
      DO 46 J=1,LNCAT                                                    PSP00850
      DIST=0.                                                            PSP00860
      DO 44 K=1,NOFEAT                                                   PSP00870
         LPACK(4) = LARRAY(IBASE + K)                                    PSP00880
         CSUN(K) = IPACK                                                 PSP00890
   44 DIST=DIST+ABS(MEANS(K,J)-CSUN(K))*SUNCOR(K)                        PSP00900
      IF (DIST - SDIST) 45,46,46                                         PSP00910
   45 KK=J                                                               PSP00920
      SDIST=DIST                                                         PSP00930
   46 CONTINUE                                                           PSP00940
   47 CONTINUE                                                           PSP00950
      N(KK)=N(KK)+1                                                      PSP00960
      IPLACE(I)=KK                                                       PSP00970
      DO 48 K=1,NOFEAT                                                   PSP00980
      AMN(K,KK)=AMN(K,KK)+CSUN(K)                                        PSP00990
      AVP(K,KK)=AVP(K,KK)+CSUN(K)**2                                     PSP01000
   48 CONTINUE                                                           PSP01010
   49 CONTINUE                                                           PSP01020
      GO TO 101                                                          PSP01030
   50 DO 100 I = 1,ICCT                                                  PSP01040
         IBASE = (I - 1)*NOFEAT + IB                                     PSP01050
      KK=1                                                               PSP01060
      IF(DODU.EQ.0) GO TO 52                                             PSP01070
      DO 51 K=1,NOFEAT                                                   PSP01080
         LPACK(4) = LARRAY(IBASE + K)                                    PSP01090
         CDUM = IPACK                                                    PSP01100
      IF (CDUM.NE.MEANDO.AND.CDUM.NE.MEANDU) GO TO 52                    PSP01110
   51 CONTINUE                                                           PSP01120
      IF (CDUM.EQ.MEANDO) IPLACE(I) = LNCAT + 1                          PSP01130
      IF (CDUM.EQ.MEANDU) IPLACE(I) = LNCAT + DODU                       PSP01140
      GO TO 100                                                          PSP01150
   52 CONTINUE                                                           PSP01160
      KK = 1                                                             PSP01170
       SDIST=10.0E+20                                                    PSP01180
       DO 70 J=1,LNCAT                                                   PSP01190
       DIST=0.0                                                          PSP01200
      DO 55 K = 1,NOFEAT                                                 PSP01210
         LPACK(4) = LARRAY(IBASE + K)                                    PSP01220
         CSUN(K) = IPACK                                                 PSP01230
   55 DIST = DIST + ABS(MEANS(K,J) - CSUN(K))                            PSP01240
       IF(DIST-SDIST)60,70,70                                           PSP01250
   60  KK=J                                                             PSP01260
       SDIST=DIST                                                       PSP01270
   70  CONTINUE                                                         PSP01280
   80  CONTINUE                                                         PSP01290
       N(KK)=N(KK)+1                                                    PSP01300
       IPLACE(I)=KK                                                     PSP01310
       DO 90 K=1,NOFEAT                                                 PSP01320
      AMN(K,KK) = AMN(K,KK) + CSUN(K)                                   PSP01330
      AVP(K,KK) = AVP(K,KK) + CSUN(K)**2                                PSP01340
   90  CONTINUE                                                         PSP01350
  100  CONTINUE                                                         PSP01360
  101 CONTINUE                                                          PSP01370
       IF(IRD.EQ.0)GO TO 110                                            PSP01380
       CALL WRITTF(ADDRS2,IPLACE,ICCT,ISTAT)                            PSP01390
       ADDRS2=ADDRS2+ICCT                                               PSP01400
  105  IF(ISTAT.EQ.1)GO TO 105                                          PSP01410
  110  IRD=IRD-1                                                        PSP01420
       IF(IRD.GT.0)GO TO 20                                             PSP01430
         KA = 1                                                         PSP01440
  115    CONTINUE                                                       PSP01450
       DO 130 K=KA,LNCAT                                                PSP01460
          KAA = K                                                       PSP01470
          IF (N(K).EQ.0) GO TO 140                                      PSP01480
       RNO=FLOAT(N(K))                                                  PSP01490
       DO 130 J=1,NOFEAT                                                PSP01500
       AMN(J,K)=AMN(J,K)/RNO                                            PSP01510
       MEANS(J,K)=AMN(J,K)                                              PSP01520
       STDEV(J,K)=SQRT(AVP(J,K)/RNO-AMN(J,K)*AMN(J,K))                  PSP01530
        DUMA = STDEV(J,K)                                               PSP01540
        IF (DUMA.LT.DUM)   STDEV(J,K) = DUM                             PSP01550
  130    CONTINUE                                                       PSP01560
         IF (ICOUNT.GT.2) RETURN                                        PSP01570
         IF (IDELF1.NE.0) GO TO 2                                       PSP01580
```

FILE: PSPPAT FORTRAN A

```
          RETURN                                                        PSP01590
 140      LNCAT = LNCAT - 1                                             PSP01600
          LLNCAT = LLNCAT - 1                                           PSP01610
          IDELET = IDELET + 1                                           PSP01620
          KAAA = KAA - 1                                                PSP01630
          IF (KAAA.EQ.LNCAT) GO TO 160                                  PSP01640
          DO 150 K = KAA,LNCAT                                          PSP01650
          N(K) = N(K+1)                                                 PSP01660
          DO 150 J=1,NOFEAT                                             PSP01670
          AMN(J,K) = AMN(J,K+1)                                         PSP01680
          AVP(J,K) = AVP(J,K+1)                                         PSP01690
  150      CONTINUE                                                     PSP01700
  160     IF (DOFLAG.NE.0) N(LNCAT+1) = NDOU(DOFLAG)                    PSP01710
          IF (DOUFLAG.NE.0) N(LNCAT + DODU) = NDOU(DODU)                PSP01720
          IF (DOFLAG.EQ.0) GO TO 170                                    PSP01730
          DO 165 J = 1,NOFEAT                                           PSP01740
  165     MEANS(J,LNCAT+1) = MEANDO                                     PSP01750
  170     IF (DOUFLAG.EQ.0) GO TO 180                                   PSP01760
          DO 175 J = 1,NOFEAT                                           PSP01770
  175     MEANS(J,LLNCAT) = MEANDU                                      PSP01780
  180     CONTINUE                                                      PSP01790
          KA = KAA                                                      PSP01800
          IF (KAAA.NE.LNCAT) GO TO 115                                  PSP01810
C*                                                                      PSP01820
C                                                                       PSP01830
          IF (ICOUNT.GT.2) RETURN                                       PSP01840
          IF (IDELET.NE.0) GO TO 2                                      PSP01850
      RETURN                                                            PSP01860
C                                                                       PSP01870
       END                                                              PSP01880
```

~~23-15~~

512

FILE: RDDPAT FORTRAN A

```
C*     THIS SUBROUTINE COORDINATES THE ROUTINES TO READ FIELDS OF DATA    RDD00010
C*     FROM THE IMAGE TAPE AND STORE IT ON A DRUM FILE FOR                RDD00020
C*     THE ISOCLS ROUTINES.                                               RDD00030
C*                                                                        RDD00040
       SUBROUTINE RDDPAT(FD1,TOP,IDATA,IDIM,LAST)                         RDD00050
      IMPLICIT INTEGER (A-Z)                                              RDD00060
      DIMENSION  FLDINF(6),IDATA(IDIM),FL(12),LSTAT(3)                    RDD00070
          COMMON ARRAY(10600)                                             RDD00080
          LOGICAL*1 LARRAY(42400)                                         RDD00090
          EQUIVALENCE (ARRAY,LARRAY)                                      RDD00100
          LOGICAL*1 LPACK(4)                                              RDD00110
          EQUIVALENCE (LPACK,IPACK)                                       RDD00120
C     INCLUDE COMMON LIST                                                 RDD00130
C     INCLUDE COMMON LIST                                                 RDD00140
      COMMON/PASS/ TOP,LNCAT,NMIN,KRN,STDMAX,DLMIN,SEP,                   RDD00150
     *                        MAP,SPTRIG,   IRD,  KPTS, NOPTS, PUNCH,    RDD00160
     *  ICHN,CHNTHS,ICHAIN(62),NWDS,IBEGIN,BEGIN1,                        RDD00170
     *              BEGIN2,BEGIN3,CLSNAM,NOFLD,IPT,TOTWRD,TOTPTS,         RDD00180
     *              NCLASS,NOCLS,TOTSUB,TOTFLD,TOTVRT,NOCL,NVRT           RDD00190
     * ,NXTCLS,NOFEAT,MAXCLS,FETVEC(30),SYMMTX(62)                        RDD00200
     *,VARSIZ,STATKY,ISOKEY,MAPFMT,MAPKEY,SEQUEN(20),PERCEN,SIMERR       RDD00210
     *,IORDER,INUNIT,INFILE,INITM,PMIN,SUBVEC(62),NOSUB2,CHNVC(30)       RDD00220
     * ,NOCHAN,ERCOMP,NOSEQ,MEANDO,MEANDU,                               RDD00230
     * SYMDO,SYMDU,ITRIGO,ITRIGU,DOFLAG,                                 RDD00240
     *  DUFLAG,DODU,STDOTS(60),NSDOTS,SUNCOR(30),LLNCAT,                  RDD00250
     * DVERT(250,2),DRECT(60,2),DVPNT(11,2),IDCNT(2),NDOU(2)              RDD00260
     *  ,MXFFTL,MAXPOP                                                    RDD00270
      REAL SUNCOR                                                         RDD00280
      COMMON/GLOBAL/HEAD(63),MAPTAP,DATAPE,SAVTAP,BMFILE,BMKEY,           RDD00290
     *              HISFIL,HISKEY,TRFORM,ERRPTP,ERRKEY,MAPUNT,NOFILE,     RDD00300
     *    DRUMAD,DWMODS,PAGSIZ,DATFIL,STAFIL,ASAV,ASAVFL                  RDD00310
     *  ,NHSTUN,NHSTFI,SCTRUN,MAPFIL                                      RDD00320
     *  ,DOTUNT,DOTFIL,NCHPAS,TRNSFL,BMTRFL,HISTFL,PCHUNT,                RDD00330
     *  CRDUNT,PRTUNT,RANDIO                                              RDD00340
C$END                                                                     RDD00350
      PRUNIT = 30                                                         RDD00360
      DIMENSION  CARD(20)                                                 RDD00370
      EQUIVALENCE (FLDINF(1),LINSTR), (FLDINF(4),SAMSTR),                 RDD00380
     *            (FLDINF(2),LINEND), (FLDINF(5),SAMEND),                 RDD00390
     *            (FLDINF(3),LININC), (FLDINF(6),SAMINC)                  RDD00400
      DATA LPRN/'('/                                                      RDD00410
      DIMENSION LND(2),LDOU(11,2),IDB(35),IDE(35),NDINT(11,2),            RDD00420
     *  DINT(120,2),DPINT(12,2),DIN(70)                                   RDD00430
      DATA DONAME/'OTHE'/                                                 RDD00440
      DATA DUNAME/'UNID'/                                                 RDD00450
      DIMENSION FLDSAV(4,10),VERTEX(220)                                  RDD00460
C*                                                                        RDD00470
C*     RESERVE 2000 LOCATIONS OF 'ARRAY' FOR FIELD DEFINITION INFORMATION.RDD00480
C*     THE REMAINDER OF 'ARRAY' IS USED FOR I/O BUFFERS.                  RDD00490
C*                                                                        RDD00500
C*     CLASS AND FIELD INFORMATION STORED AS FOLLOWS                      RDD00510
C*                                                                        RDD00520
C*                                                                        RDD00530
C*                                                                        RDD00540
C*        ARRAY(1)    =CLASS NAME                                         RDD00550
C*        ARRAY(2)    =RESERVED FOR INDEX POINTER TO NEXT CLASS NAME      RDD00560
C*        ARRAY(3)    =RESERVED FOR NO. OF CLUSTERS IN THIS CLASS         RDD00570
C*        ARRAY(4)    =NO. OF FIELDS FOR THIS CLASS                       RDD00580
C*        ARRAY(5)    =FIRST FIELD NAME FOR THIS CLASS                    RDD00590
C*             (6)    =NO. OF VERTICES FOR THIS FIELD  (NV)               RDD00600
C*             (7)-(7+NV*2)  = ACTUAL VERTEX NUMBERS                      RDD00610
C*             (8+NV*2)  =TOTAL PIXELS IN THIS FIELD                      RDD00620
C*             (9+NV*2)-(10+NV*2) = FLDINF BLOCK FOR THIS FIELD           RDD00630
      CALL TAPHDR(DATAPE,DATFIL)                                          RDD00640
    1 CONTINUE                                                            RDD00650
      RESERV=2000                                                         RDD00660
      ADDRES=BEGIN1                                                       RDD00670
      IBUF=1                                                              RDD00680
      IWRD=0                                                              RDD00690
      NVRT=0                                                              RDD00700
      LAST=0                                                              RDD00710
      TOTWRD=0                                                            RDD00720
      IDP = 0                                                             RDD00730
      DOFLAG = 0                                                          RDD00740
      DUFLAG = 0                                                          RDD00750
      DODU = 0                                                            RDD00760
      NDOU(1) = 0                                                         RDD00770
      NDOU(2) = 0                                                         RDD00780
    2 BFINDX=RESERV+1                                                     RDD00790
      NBUFS=1
```

FILE: RDDPAT FORTRAN A

```
      MAXDIM=TOP-RESERV                                                 RDD00800
      BUFSIZ= MAXDIM/(NBUFS*NOFEAT)  * NOFEAT                           RDD00810
            BUFSIZ = (BUFSIZ/4)*4                                       RDD00820
      IF(BUFSIZ.GT. 100)GO TO 3                                         RDD00830
      RESERV=RESERV-100                                                 RDD00840
      IF(RESERV.GT.30 )GO TO 2                                          RDD00850
      GO TO 70                                                          RDD00860
    3 CONTINUE                                                          RDD00870
      NOFLD=0                                                           RDD00880
      IPT=1 + FD1 - 1                                                   RDD00890
      TOTVT2=0                                                          RDD00900
      IF(NOCL.EQ.0)GO TO 5                                              RDD00910
    4 ARRAY(IPT)=NXTCLS                                                 RDD00920
      IPT=IPT+4                                                         RDD00930
      WRITE(6,HEAD)                                                     RDD00940
      WRITE(6,500)NXTCLS                                                RDD00950
C*                                                                      RDD00960
C*    READ A FIELD DESCRIPTION FROM CARDS.                              RDD00970
C*                                                                      RDD00980
    5 ICK = LAREAD(ARRAY(IPT),ARRAY(IPT+2),FLDINF,ARRAY(IPT+1) )        RDD00990
      IF(ICK.NE.-3) GO TO 1000                                          RDD01000
       WRITE (6,140)                                                    RDD01010
       READ (RRUNIT,150) (CARD(I), I=1,20)                              RDD01020
       WRITE (6,160)  (CARD(I), I=1,20)                                 RDD01030
  150   FORMAT(20A4)                                                    RDD01040
  160   FORMAT(1X,20A4)                                                 RDD01050
       REWIND RRUNIT                                                    RDD01060
      IDP=IDP+1                                                         RDD01070
      IDCNT(IDP)=0                                                      RDD01080
      DVPNT(1,IDP)=1                                                    RDD01090
      READ(30,100) DNAME                                                RDD01100
      REWIND 30                                                         RDD01110
      IF(DNAME.EQ.DONAME) ITRIGO=1                                      RDD01120
      IF(DNAME.EQ.DUNAME) ITRIGU=1                                      RDD01130
      IF(DNAME.EQ.DONAME) IS=2                                          RDD01140
      IF(DNAME.EQ.DUNAME) IS=1                                          RDD01150
      INDV=1                                                            RDD01160
      INDR=1                                                            RDD01170
      GO TO 5                                                           RDD01180
 1000 IF(ICK.LE.0.OR.IDP.LE.0) GO TO 1030                               RDD01190
      IF (IDCNT(IDP).LT.10) GO TO 1025                                  RDD01200
      WRITE(6,170)                                                      RDD01210
  170 FORMAT(// ' TOO MANY DO OR DU FIELDS THESE IGNORED')              RDD01220
      GO TO 5                                                           RDD01230
 1025 CONTINUE                                                          RDD01240
       READ (RRUNIT,150) (CARD(I), I=1,20)                              RDD01250
       WRITE (6,160)  (CARD(I), I=1,20)                                 RDD01260
       REWIND RRUNIT                                                    RDD01270
       DVERT(INDV,IDP) = ARRAY(IPT + 1)                                 RDD01280
      IDLIM = DVERT(INDV,IDP)*2                                         RDD01290
      DO 1010 I=1,IDLIM                                                 RDD01300
      INDV=INDV+1                                                       RDD01310
      VERTEX(TOTVT2+I)=ARRAY(IPT+I+1)                                   RDD01320
 1010 DVERT(INDV,IDP) = ARRAY(IPT + I + 1)                              RDD01330
      INDV = INDV + 1                                                   RDD01340
      TOTVT2=TOTVT2+IDLIM                                               RDD01350
      DO 1020 I=1,6                                                     RDD01360
      DRECT(INDR,IDP)=FLDINF(I)                                         RDD01370
 1020 INDR=INDR+1                                                       RDD01380
      IDCNT(IDP)=IDCNT(IDP)+1                                           RDD01390
      IDUM=IDCNT(IDP)+1                                                 RDD01400
      DVPNT(IDUM,IDP)=INDV                                              RDD01410
      GO TO 5                                                           RDD01420
C     FINISHED WITH DO/DU FIELD PROCESSING                              RDD01430
 1030   CONTINUE                                                        RDD01440
      IDP=0                                                             RDD01450
      IDPP=ITRIGO+ITRIGU                                                RDD01460
      IF(ICK.LT.0)GO TO 20                                              RDD01470
      IF(ICK.EQ.0)GO TO 30                                              RDD01480
      IF(NOCL.GT.0)GO TO 6                                              RDD01490
      WRITE(6,200)                                                      RDD01500
      CALL CMERR                                                        RDD01510
    6 CONTINUE                                                          RDD01520
      NV=ARRAY(IPT+1)                                                   RDD01530
      NVRT=NVRT+NV                                                      RDD01540
      NOFLD=NOFLD+1                                                     RDD01550
      NSAMP=(SAMEND-SAMSTR)/SAMINC+1                                    RDD01560
      FLDSAM=0                                                          RDD01570
      IB=IPT+2                                                          RDD01580
```

FILE: RDDPAT FORTRAN A

```
      NQ=NV-1                                                          RDD01590
      NR=NQ-5                                                          RDD01600
      IF(NQ.GT.5)NQ=5                                                  RDD01610
      IE=IP+NQ*2 - 1                                                   RDD01620
      WRITE(6,600)NOFLD,ARRAY(IPT),SAMINC,LININC,                      RDD01630
     *     (LPRN,ARRAY(I),ARRAY(I+1),I=IB,IE,2)                        RDD01640
      IF(NR.LE.0)GO TO 7                                               RDD01650
      IB=IE+1                                                          RDD01660
      IE=IB+NR*2 - 1                                                   RDD01670
      WRITE(6,650) (LPRN,ARRAY(I),ARRAY(I+1),I=IB,IE,2)                RDD01680
    7 CONTINUE                                                         RDD01690
      IF(NSAMP*NOFEAT.GT.IDIM)GO TO 90                                 RDD01700
C*                                                                     RDD01710
C*    POSITION TAPE FOR THIS FIELD                                     RDD01720
C*                                                                     RDD01730
      CALL FLDINT(FLDINF,FETVEC,NOFEAT)                                RDD01740
      FLDSAM=0                                                         RDD01750
      DO 10 LINE=LINSTR,LINEND,LININC                                  RDD01760
      LND(1)=0                                                         RDD01770
      LND(2)=0                                                         RDD01780
      IDBB = 2                                                         RDD01790
      IDEE=1                                                           RDD01800
      IF (IDPP.EQ.0) GO TO 1095                                        RDD01810
C     BOTH DO AND DU TRIGGERS OFF --- SKIP AROUND                      RDD01820
      DO 1060 IND=1,IDPP                                               RDD01830
      IDLIM=IDCNT(IND)                                                 RDD01840
      DO 1050 I=1,IDLIM                                                RDD01850
      IDUM=(I-1)*3                                                     RDD01860
      LDSTR=DIRECT(IDUM+1,IND)                                         RDD01870
      LDEND=DIRECT(IDUM+2,IND)                                         RDD01880
      LDINC=DIRECT(IDUM+3,IND)                                         RDD01890
      DO 1040 II = LDSTR,LDEND,LDINC                                   RDD01900
      IF(II.NE.LINE) GO TO 1040                                        RDD01910
      LND(IND)=LND(IND)+1                                              RDD01920
      IDUM=LND(IND)                                                    RDD01930
      LDDU(IDUM,IND) = I                                               RDD01940
 1040 CONTINUE                                                         RDD01950
 1050 CONTINUE                                                         RDD01960
 1060 CONTINUE                                                         RDD01970
      IF (LND(1).EQ.0.AND.LND(IDPP).EQ.0) GO TO 1095                   RDD01980
C    NO DO  OR   DU   FOR THIS LINE                                    RDD01990
      IF(LND(1).GT.0)IDBB=1                                            RDD02000
      IF(IDPP.EQ.2.AND.LND(2).GT.0) IDEE=2                             RDD02010
      DO 1090 IND=IDBB,IDEE                                            RDD02020
      IDLIM=LND(IND)                                                   RDD02030
      IF (IDLIM.EQ.0) GO TO 1090                                       RDD02040
      IDUM=0                                                           RDD02050
      DPINT(1,1) = 1                                                   RDD02060
      DPINT(1,2) = 1                                                   RDD02070
      DO 1080 I=1,IDLIM                                                RDD02080
      IDF=LDDU(I,IND)                                                  RDD02090
      DVP=DVPNT(IDF,IND)                                               RDD02100
      CALL FDLINT(DVERT(DVP+1,IND),DVERT(DVP,IND),FL,LINE,SAMPS,NI)    RDD02110
      NDINT(I,IND)=NI                                                  RDD02120
      IF (NI.EQ.0) GO TO 1080                                          RDD02130
      DO 1070 II=1,NI                                                  RDD02140
 1070 DINT(II+IDUM,IND)=FL(II)                                         RDD02150
      IDUM=IDUM+NI                                                     RDD02160
      DPINT(I+1,IND) = IDUM + 1                                        RDD02170
 1080 CONTINUE                                                         RDD02180
 1090 CONTINUE                                                         RDD02190
 1095 CONTINUE                                                         RDD02200
      CALL LINEND(IDATA,ENDTAP)                                        RDD02210
      IF(ENDTAP.EQ.-1)GO TO 80                                         RDD02220
C*                                                                     RDD02230
C*    FIND SAMPLE INTERSECTS FOR THIS LINE - NI=NO. OF INTERSECTS      RDD02240
C*                                                                     RDD02250
      CALL FDLINT(ARRAY(IPT+2),NV,FL,LINE,SAMPS,NI)                    RDD02260
C*                                                                     RDD02270
C*    STORE DATA ON THIS LINE INTO OUTPUT BUFFER                       RDD02280
C*                                                                     RDD02290
      MODSS=MOD(SAMSTR,SAMINC)                                         RDD02300
      DO 60 I=1,NI,2                                                   RDD02310
      IB=(FL(I)-SAMSTR)/SAMINC+1                                       RDD02320
      IE=(FL(I+1)-SAMSTR)/SAMINC+1                                     RDD02330
      IF(MODSS.NE.MOD(FL(I),SAMINC)) IB=IB+1                           RDD02340
      IF(IP.GT.IE) GOTO 60                                             RDD02350
      IF(IDPP.EQ.0) GOTO 2055                                          RDD02360
      IF(LND(IDBB).EQ.0.AND.LND(IDEE).EQ.0) GOTO 2055                  RDD02370
```

FILE: RDDPAT FORTRAN A

```
      DO 2050 IND=IDBB,IDEE                                                  RDD02380
      IDLIM=LND(IND)                                                         RDD02390
      IF(IDLIM.EQ.0) GOTO 2050                                               RDD02400
      IDSIT=1                                                                RDD02410
      MEANDD=MEANDO                                                          RDD02420
      IF(IDBB.EQ.IDEE) GOTO 2003                                             RDD02430
      IF(IND.EQ.2) IDSIT=2                                                   RDD02440
      IF(IND.EQ.2) MEANDD=MEANDU                                             RDD02450
      GOTO 2009                                                              RDD02460
 2003 IF(IDBB.EQ.1.AND.IDPP.EQ.2) GOTO 2009                                  RDD02470
      IF(ITRIGU.EQ.0) GOTO 2009                                              RDD02480
      IDSIT=2                                                                RDD02490
      MEANDD=MEANDU                                                          RDD02500
 2009 CONTINUE                                                               RDD02510
      DO 2040 K=1,IDLIM                                                      RDD02520
      NDIN=NDINT(K,IND)                                                      RDD02530
      IF(NDIN.EQ.0) GOTO 2040                                                RDD02540
      DPIN=DPINT(K,IND)                                                      RDD02550
      DO 2010 KK=1,NDIN                                                      RDD02560
 2010 DIN(KK)=DINT(DPIN+KK-1,IND)                                            RDD02570
      IDUM=0                                                                 RDD02580
      DO 2020 KK=1,NDIN,2                                                    RDD02590
      IDUM=IDUM+1                                                            RDD02600
      IDB(IDUM)=(DIN(KK)-SAMSTR)/SAMINC+1                                    RDD02610
      IDE(IDUM)=(DIN(KK+1)-SAMSTR)/SAMINC+1                                  RDD02620
      IF(MODSS.NE.MOD(DIN(KK),SAMINC)) IDB(IDUM)=IDB(IDUM)+1                 RDD02630
 2020 CONTINUE                                                               RDD02640
      DO 2030 KK=1,IDUM                                                      RDD02650
      IDS=IDB(KK)                                                            RDD02660
      IDE=IDE(KK)                                                            RDD02670
      IF(IDS.GT.IE.OR.IB.GT.IDE) GOTO 2030                                   RDD02680
      IF(IDS.LE.IB) IDS=IB                                                   RDD02690
      IF(IDE.GT.IE) IDE=IE                                                   RDD02700
      IF(IDS.GT.IDE) GOTO 2030                                               RDD02710
      DO 2025 KKK=IDS,IDE                                                    RDD02720
      NDOU(IND)=NDOU(IND)+1                                                  RDD02730
      DO 2023 KKKK=1,NOFEAT                                                  RDD02740
      DUMMY1=KKK+NSAMP*(KKKK-1)                                              RDD02750
 2023 IDATA(DUMMY1)=MEANDD                                                   RDD02760
 2025 CONTINUE                                                               RDD02770
 2030 CONTINUE                                                               RDD02780
 2040 CONTINUE                                                               RDD02790
      IF(IDSIT.EQ.1.AND.NDOU(IND).GT.0) DOFLAG=1                             RDD02800
      IF(IDSIT.EQ.2.AND.NDOU(IND).GT.0) DUFLAG=1                             RDD02810
 2050 CONTINUE                                                               RDD02820
 2055 DODU=DOFLAG+DUFLAG                                                     RDD02830
         IBASE = (BFINDX - 1)*4                                              RDD02840
         IBUF4 = BUFSIZ*4                                                    RDD02850
         NWDS4 = NWDS*4                                                      RDD02860
      DO 50 J=IB,IE                                                          RDD02870
      FLDSAM=FLDSAM+1                                                        RDD02880
      DO 50 K=1,NOFEAT                                                       RDD02890
      IWRD=IWRD+1                                                            RDD02900
      DUMMY2=J+NSAMP*(K-1)                                                   RDD02910
         IPACK = IDATA(DUMMY2)                                               RDD02920
         DUMMY3 = IBASE + IWRD                                               RDD02930
         LARRAY(DUMMY3) = LPACK(4)                                           RDD02940
         IF (IWRD.LT.IBUF4) GO TO 50                                         RDD02950
         TOTWRD = TOTWRD + IWRD                                              RDD02960
         IF (TOTWRD.GT.NWDS4) GO TO 35                                       RDD02970
         CALL RWRITE(ADDRES,ARRAY(BFINDX),BUFSIZ,LSTAT(1,))                  RDD02980
         ADDRES = ADDRES + BUFSIZ                                            RDD02990
   40 IF(LSTAT(IBUF).EQ.1) GOTO 40                                           RDD03000
      IWRD=0                                                                 RDD03010
   50 CONTINUE                                                               RDD03020
   60 CONTINUE                                                               RDD03030
   10 CONTINUE                                                               RDD03040
      IPT = IPT + NV*2 + 2                                                   RDD03050
      ARRAY(IPT)=FLDSAM                                                      RDD03060
      DO 15 I=1,6                                                            RDD03070
      IPT=IPT+1                                                              RDD03080
   15 ARRAY(IPT)=FLDINF(I)                                                   RDD03090
      IPT=IPT+1                                                              RDD03100
      IF(IPT+30 .GT. RESERV)GO TO 70                                         RDD03110
      GO TO 5                                                                RDD03120
C*                                                                           RDD03130
C*    CLASS NAME CARD ENCOUNTERED - REREAD PREVIOUS CARD TO GET NAME         RDD03140
C*                                                                           RDD03150
   20 NOCL=NOCL+1                                                            RDD03160
```

FILE: RDDPAT FORTRAN A

```
      IF(NOCL.GT.1)GO TO 25                                                  RDD03170
      READ(30,100)NXTCLS                                                     RDD03180
      REWIND 30                                                              RDD03190
      GO TO 4                                                                RDD03200
   25 CLSNAM=NXTCLS                                                          RDD03210
      READ(30,100)NXTCLS                                                     RDD03220
      REWIND 30                                                              RDD03230
      GO TO 31                                                               RDD03240
C*    EMPTY LAST BUFFER AND RETURN TO PROCESS DATA FOR THIS CLASS.           RDD03250
   30 CLSNAM=NXTCLS                                                          RDD03260
      LAST=1                                                                 RDD03270
C*                                                                           RDD03280
   31 TOTWRD=TOTWRD+IWRD                                                     RDD03290
      IF(TOTWRD.GT.NWDS4) GOTO 35                                            RDD03300
          IWRD4 = IWRD/4                                                     RDD03310
          IF (4*IWRD4.NE.IWRD) IWRD4 = IWRD4 + 1                             RDD03320
          CALL RWRITE(ADDRES,ARRAY(BFINDX),IWRD4,LSTAT(1))                   RDD03330
      TOTPTS=TOTWRD/NOFEAT                                                   RDD03340
          IPT = IPT - FD1 + 1                                                RDD03350
        IPAT = (TOTWRD/4) + 1                                                RDD03360
      IF(IPAT+TOTPTS.LE.NWDS)RETURN                                          RDD03370
      WRITE(6,200)NWDS                                                       RDD03380
          IPT = IPT - FD1 + 1                                                RDD03390
      RETURN                                                                 RDD03400
   35 WRITE(6,200)NWDS                                                       RDD03410
      CALL CMERR                                                             RDD03420
   70 WRITE(6,300)RESERV                                                     RDD03430
      CALL CMERR                                                             RDD03440
   80 WRITE(6,400)                                                           RDD03450
      CALL CMERR                                                             RDD03460
   90 WRITE(6,700)IDIM                                                       RDD03470
      CALL CMERR                                                             RDD03480
  100 FORMAT(10X,A4)                                                         RDD03490
  140 FORMAT(/// ' DESIGNATED OTHER OR UNIDENTIFIABLE FIELDS INPUT'/)        RDD03500
  200 FORMAT(' TOO MUCH DATA REQUESTED--PIXELS*(CHANNELS+1) CANNOT EXCEEDRDD03510
     *D',I10)                                                                RDD03520
  300 FORMAT(' STORAGE REQUIRED FOR FIELD DEFINITION INFORMATION EXCEEDSRDD03530
     * THE DIMENSION LIMIT OF',I5)                                          RDD03540
  400 FORMAT(' END-OF-TAPE REACHED BEFORE END OF FIELD')                     RDD03550
  500 FORMAT(//40X,'FIELDS TO BE CLUSTERED FOR CLASS',1X,A4//                RDD03560
     *   T36,'SAMPLE',T45,'LINE'/T20,'FIELD NAME',T36,'INC.',               RDD03570
     *   T45,'INC.',T73,'VERTICES  (SAMPLE,LINE)'/)                         RDD03580
  600 FORMAT(1X,T16,I3,T22,A4,T36,I4,T45,I4,T50,                             RDD03590
     * 5(A1,I4,',',I4,')',1X))                                             RDD03600
  650 FORMAT(1X,T60,5(A1,I4,',',I4,')',1X))                                  RDD03610
  700 FORMAT(' NO. OF PIXELS TO BE UNPACKED PER SCAN EXCEEDS THE DIMENSIRDD03620
     *ON LIMIT OF',I5)                                                       RDD03630
  800 FORMAT(// ' INPUT ERROR - A CLASSNAME CARD MUST BE INPUT BEFORE A GRDD03640
     *ROUP OF FIELDS'/)                                                      RDD03650
      RETURN                                                                 RDD03660
      END                                                                    RDD03670
```