T77-123[illegible] NMF
80-10196
JSC-13142
NASA CR-
160625

"AS-BUILT" DESIGN SPECIFICATION
OF THE
CAMS/CAS INTERFACE TAPE REPORT
GENERATION PROGRAM FOR LACIE 6A

Job Order 71-695

(TIRF 77-0040)

(This document supersedes LEC-9882)

Prepared By

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

Contract NAS 9-15200

For

EARTH OBSERVATIONS DIVISION
SCIENCE AND APPLICATIONS DIRECTORATE

National Aeronautics and Space Administration
LYNDON B. JOHNSON SPACE CENTER
Houston, Texas

October 1977

LEC-11292

JSC-13142

"AS-BUILT" DESIGN SPECIFICATION
OF THE
CAMS/CAS INTERFACE TAPE REPORT
GENERATION PROGRAM FOR LACIE 6A
JOB ORDER 71-695
(TIRF 77-0040)

PREPARED BY

W. a. Holley
W. A. Holley
Applications Software Section

APPROVED BY

Philip L. Krumm
Philip L. Krumm, Supervisor
Applications Software Section

Prepared By
Lockheed Electronics Company, Inc.
For
Earth Observations Division

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

October 1977

LEC-11292

CONTENTS

Section Page

Section Page

Appendices

1. SCOPE

1.1 GENERAL

This document is the "as-built" design specification of the CAMS/CAS Interface Tape Report Generation Program for LACIE 6A.

2. APPLICABLE DOCUMENTS

- TIRF 77-0040
- Specification for the CAMS/CAS Interface Tape Report Generation Program - LEC-9151
- CAMS/CAS Interface Control Tape format specification in Earth Resources Data Format Control Book Volume 1 (PHO-TR543, Rec. A. Change 3)
- "As-Built" Design Specification of the CAMS/CAS Interface Tape Report Generation Program - LEC 9882
- Acceptance Test Specification for CAMS/CAS Interface Tape Report Generation Program for LACIE 6A - LEC 11028
- CAMS/CAS Interface Tape Printout after LACIE 6A JSC Memorandum, SF4-77-7-13, 7/21/77.

3. SYSTEM DESCRIPTION

3.1 HARDWARE DESCRIPTION

N/A

3.2 SOFTWARE DESCRIPTION

The purpose of this program is to produce CAMS reports from data on the CAMS/CAS interface tape.

3.2.1 SOFTWARE COMPONENT NO. 1 (BIASRP)

The main program BIASRP reads control cards, locates segment data on the input tape and calls subroutines to generate requested reports.

3.2.1.1 Linkages

BIASRP calls subroutines CDRED, CAMREC, SEGEXT, CLURES, STDATA, and DOTRAY.

3.2.1.2 Interfaces

N/A

3.2.1.3 Inputs

BIASRP control cards are: SEGMENT XXXX, RECORD ID XXXXXX XXXXXX, ALL, END. CAM/CAS interface tape records are inputs to BIASRP. See reference 3 in section 2, for record formats.

3.2.1.4 Outputs

An error message is output indicating a bad data card. If a requested segment is not on the input tape, the program writes a message to that effect.

3.2.1.5 Storage Requirements

Total space allocated is 2330 bytes.

3.2.1.6 Description

BIASRP is the CAMRPT main program. The program sets the printout option indicator PRTOUT to 0 initially. In this mode the output of some reports is conditional. On the first call to tape read subroutine CDRED, the program reads two data cards specifying the device code (M or X) and the unit number (0 or 1). BIASRP next reads a program control card and tests the first non-blank character for one of the following: S, R, A, or E. If the card is blank or the first character is not one of the above, the program prints an error message on the line printer ans stops.

The action taken for each control card is given below. Note that if a control card other than E is read in, the printout option indicator PRTOUT is set to 1. This is the option to output all reports, including conditional reports.

- S - Option indicator PRTOUT Is set to 1. The program obtains the segment number from the input card. The program searches the input tape for a recognition segment record whose segment number matches the control card segment number. If a match is not found, a message is printed and the program goes to read the next control card. If a match is found, the CAMRPT reports for the segment are generated. The program then reads the next control card.

- R - The action taken is the same as for the S card above, except that the record identification number is used instead of the segment number.

- A - Option indicator PRTOUT is set to 1. Beginning with the segment on the the tape at which the tape is currently positioned, the program generates CAMRPT reports for that segment and all the following segments. When the second tape end of file, indicating end of data, is reached, the program rewinds the tape and returns to read the next control card.

- E - If the printout option indicator PRTOUT is 0, the program generates reports in the limited printout mode, rewinds the tape, and then stops. If PRTOUT is 1, the tape is rewound and the program stops.

To generate reports for a segment the program first calls CAMREC to read and process recognition segment records. Report output takes place in two stages. The standard report output is written to the line printer (unit 6). Conditional report output is written to disk (unit 3). The standard output from CAMREC is headed CAMS Interface Report, and the conditional output, Classification Data.

After CAMREC, the program reads the segment summary record and calls SEGEXT. Data from the segment summary record is saved in COMMON block CBIAS. This data includes bias correction factors which are used in DOTRAY calculations. BIASRP next calls CLURES to process the cluster match records and to generate the conditional Cluster Report. Next the Statistics Report, also conditional, is generated from statistics records. Finally DOT subset records are read in and processed by subroutine DOTRAY. The standard report output from DOTRAY consists of DOT Label/Classification tables and the Bias Correction Report.

In the limited printout mode the decision as to whether or not to output CAMRPT conditional reports is not made until the percentages of correctly classified Dots (PCC-1 and PCC-2) are calculated in subroutine DOTRAY. If either PCC-1 or PCC-2 is less than 80% then the conditional reports are read from disk and written to the line printer. In the full printout mode (PRTOUT = 1) the conditional reports are always retrieved from disk and printed out.

3.2.1.7 Flowcharts

See Flow Diagram 1.

3.2.1.8 Listing

See Appendix A

3.2.2 SOFTWARE COMPONENT NO. @ (CAMREC)

This program processes classification results contained in recognition segment records, outputs the CAMS Interface Report and the conditional Classification Data report.

3.2.2.1 Linkages

CAMREC is called by BIASRP and calls subroutines CAMHDG. CPIPO, MV, and CDRED.

3.2.2.2 Interfaces

N/A

3.2.2.3 Inputs

Recognition segment records, containing subclass a priori and threshold values, and subclass related classification results.

3.2.2.4 Outputs

CAMS Interface Report and a Classification Data report.

3.2.2.5 Storage Requirements

Total space allocated is 1636 bytes.

3.2.2.6 Description

CAMREC is called with the first recognition segment record for the segment to be processed residing in array IBUF. CAMREC first calls CAMHDG to print out the report heading, the segment number, record ID, and acquisition dates. Title and column headings for the classification section of the report are written out by CAMREC. Processing of classification results begins by setting the location in array IBUF of the first subfield containing subclass related results. Subfield contents are accessed by calling CPIPO. CPIPO returns the class portion of the subclass name and the counts PI and PO of pixels classified into, and thresholded out of the subclass. If the first character of the class name is X, PI is added to the X category pixel count. If the category is W, for wheat, then the count for the first wheat class is set to PI and the wheat class name is saved in CLIST. PO is added to the total of pixels thresholded, TC, in the COMMON block CBIAS.

In processing for the second, and subsequent subclasses, the program calls CPIPO to get the next class name, checks to see if it is wheat, and, if so, compares it to the last class name in CLIST. If it is not the same, the new name is saved in CLIST and the class index is incremented by 1. This causes wheat class pixel count PI to be tallied in the next results array location.

3.2.2.7 Flowcharts

N/A

3.2.2.8 Listing

See Appendix A.

3.2.3 SOFTWARE COMPONENT NO. 3 (SEGEXT)

This program saves data contained in the segment summary record.

3.2.3.1 Linkages

SEGEXT is called by BIASRP.

3.2.3.2 Interfaces

N/A

3.2.3.3 Inputs

Segment summary record.

3.2.3.4 Outputs

None.

3.2.3.5 Storage Requirements

Total space allocated is 248 bytes.

3.2.3.6 Description

SEGEXT is called if the main program reads a segment summary record during report generation. SEGEXT saves the five possible category labels in COMMON array LABEL. Population data is saved in NI. Bias correction factors and percentages are decoded and saved in other COMMON block CBIAS arrays.

3.2.3.7 Flowcharts

N/A

3.2.3.8 Listing

See Appendix A.

3.2.4 SOFTWARE COMPONENT NO. 4 (CLURES)

This program processes cluster match and distance records and generates the conditional cluster report.

3.2.4.1 Linkages

CLURES is called by BIASRP. It calls CDRED and BNT.

3.2.4.2 Interfaces

N/A

3.2.4.3 Inputs

Cluster match and distance records.

3.2.4.4 Outputs

Cluster report.

3.2.4.5 Storage Requirements

Total space allocated is 587 bytes.

3.2.4.6 Description

CLURES is called if the main program reads a cluster match record during report generation. The program first sets the clustering channel list array. The variables ALSETS, total number of clusters, and SETRK, number of clusters in the record, are decoded. ALSETS is written out as clusters generate. For each cluster the program writes out the cluster name, the subclass/dot match name and the L2 match distance. If there are more the 40 cluster sets, the program reads a second cluster match record. The clustering channel list is written at the end of the report.

3.2.4.7 Flowcharts

See Flow Diagram 2.

3.2.4.8 Listing

See Appendix A.

3.2.5 SOFTWARE COMPONENT NO. 5 (STDATA)

This subroutine formats and outputs field and subclass statistics data.

3.2.5.1 Linkages

STDATA is called by CAMRPT. STDATA calls subroutines KNT, MDTTL, MEAN, POP, CDRED, SNAME, FNAME, STDMP, and BNT.

3.2.5.2 Interfaces

N/A

3.2.5.3 Inputs

The statistics record, containing, for fields or for subclasses, the population and values of the mean and standard deviation by channel.

3.2.5.4 Outputs

The statistics report.

3.2.5.5 Storage Requirements

Space allocated, including subroutines, is 2197 bytes.

3.2.5.6 Description

STDATA is called from BIASRP. By means of decode statements, the program converts several variables from input character format in IBUF to integers. The variables are ALSETS, the total number of statistics sets, SETSR, the number of sets in the current record, and NCH, the number of channels. STDATA calls subroutines to move data from input record subfields to print buffers. SNAME and FNAME move name data and insert SUBCL and FIELD designations in the print buffer. POP is called to move population data. MDTTL is called to supply column headings for means and standard deviations, which are transferred to a print buffer by MEAN. MEAN also puts decimal points where needed. The variable DSETS, set to 5, controls the number of statistics sets to be accumulated before outputting the print buffers. When the current record statistics sets counter reaches SETSR, and ALSETS sets have not yet been processed, STDATA calls CDRED to read the next statistics record from tape.

3.2.5.7 Flowcharts

N/A

3.2.5.8 Listing

See Appendix A.

3.2.6 SOFTWARE COMPONENT NO. 6 (DOTRAY)

This program processes DOT subset records and generates DOT label/classification tables and the bias correction report.

3.2.6.1 Linkages

DOTRAY is called by BIASRP. It calls CDRED.

3.2.6.2 Interfaces

N/A

3.2.6.3 Inputs

DOT subset records

3.2.6.4 Outputs

Type 1 and 2 label/classification tables and the bias correction report.

3.2.6.5 Storage Requirements

Total space allocated is 2642 bytes.

3.2.6.6 Description

DOTRAY is called by BIASRP after reading in the first DOT subset record for the current segment. DOTRAY obtains category labels for C1 and C2 (as W for Wheat) from the common block CBIAS. The program locates the first DOT entry in the input buffer IBUF and sets variables TYPE (1), LBLED (1), and CLASFY (1) to the first DOT type, user supplied label, and classification label. The variables are set for the second DOT from data in the next IBUF Dot Entry, and so on. After DOT entries for the first record (15 DOTS) have been accessed the program calls CDRED to read in the next DOT record. The process is repeated until the information for all 209 DOTS has been read in.

Next DOTRAY generates an 11 by 19 matrix of user label/classification entries for type 1 and type 3 DOTS. A similar Matrix is generated for type 2 DOTS which also includes type 0 DOTS. DOTS with a classification label of DV or DO do not appear in either matrix.

After varibles listed below have been initialized to 0, the program tests the data for each DOT for the condition indicated, and increments the variable if the condition is satisfied. It is understood in every case that the conditions apply to DOTS which are labeled, that is, LBLED (I) is not blank, and to DOTS not classified as DU or DO.

- NTYP1 - The number of DOTS which are either type 1 or type 3.
- NTYP2 - The number of type 2 DOTS
- NAII - The number of type 1 type 3 DOTS whose label and classification are the same.
- NGIJ - The number of type 1 and type 3 DOTS whose label and classification are not the same.
- NOLC - The number of type 2 DOTS which are both labeled and classified.
- NBII - The number of type 2 DOTS whose label and classification are the same.
- WHIJ - The number of type 2 DOTS whose label and classification are not the same.
- LN(I) - For I = 1,2,3, the number of type 2 DOTS classified in category CI.
- LBC1 - The number of type 2 DOTS labeled C1.
- LBC2 - The number of type 2 DOTS labeled C2.
- LM1 - The number of type 2 DOTS labeled C1 and classified in either C1 or C2.
- LM2 - The number of type 2 DOTS labeled C2 and classified in either C1 or C2.
- LM3 - The number of type 2 DOTS both labeled and classified C3.

The program computes the corrected percentages, variances, and other parameters for categories C1 and C2, and for the combined C1 and C2, or grain category. The equations for these calculations are contained in the requirements document, JSC Memorandum SF4-77-7-13. The calculations are output as the Bias Correction Report.

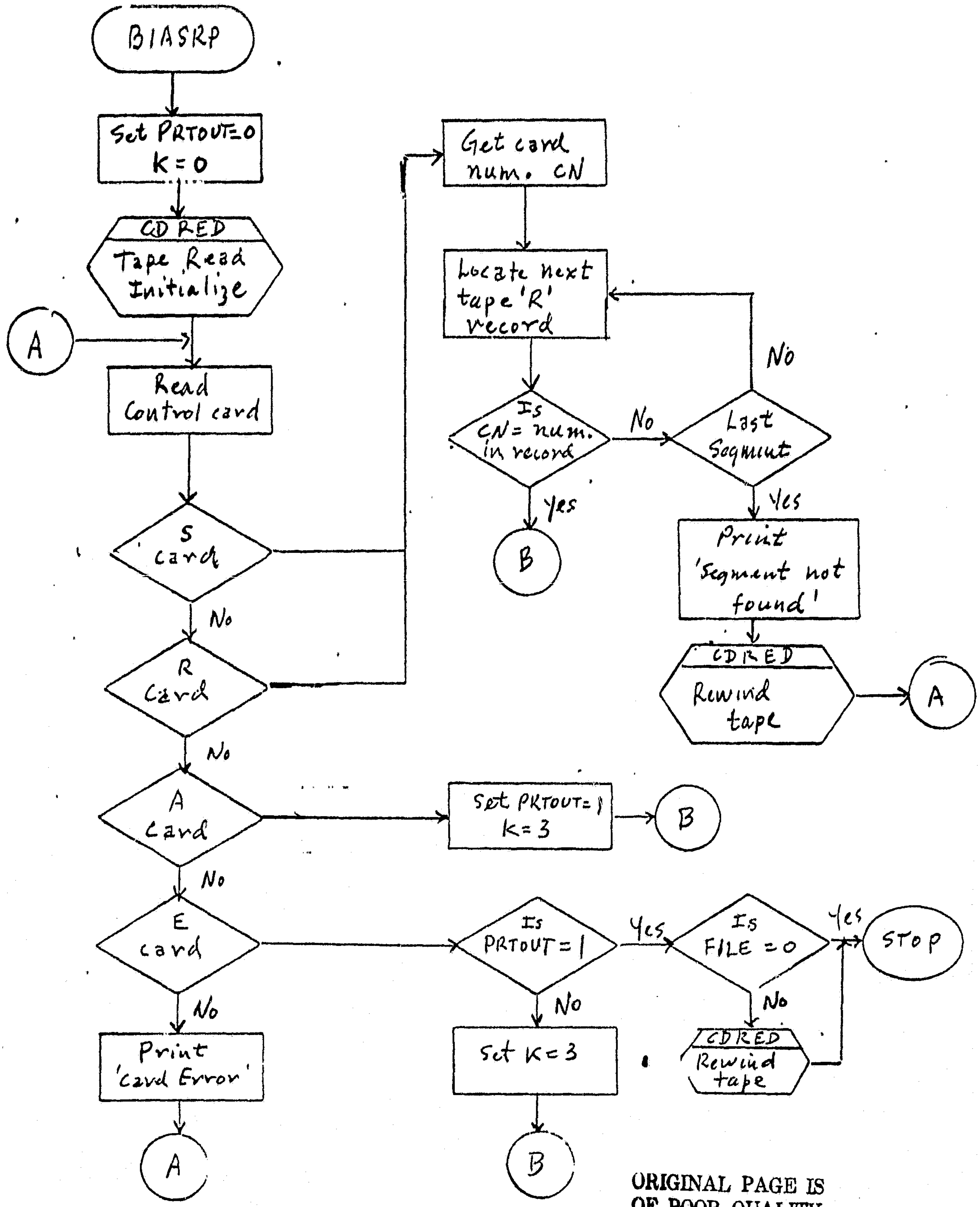
After the Report has been generated, if the printout option argument PRTOUT is 1, then a return is made to the main program. If PRTOUT is not 1, then PCC-1 and PCC-2 Values (percentages of type 1 and type 2 correctly classified DOTS) are tested. If either value is less than 80%, then PRTOUT is set to 2, to indicate to the main program that the conditional CAMRPT reports are to be read from disk and output to the line printer.

3.2.6.7 Flowcharts

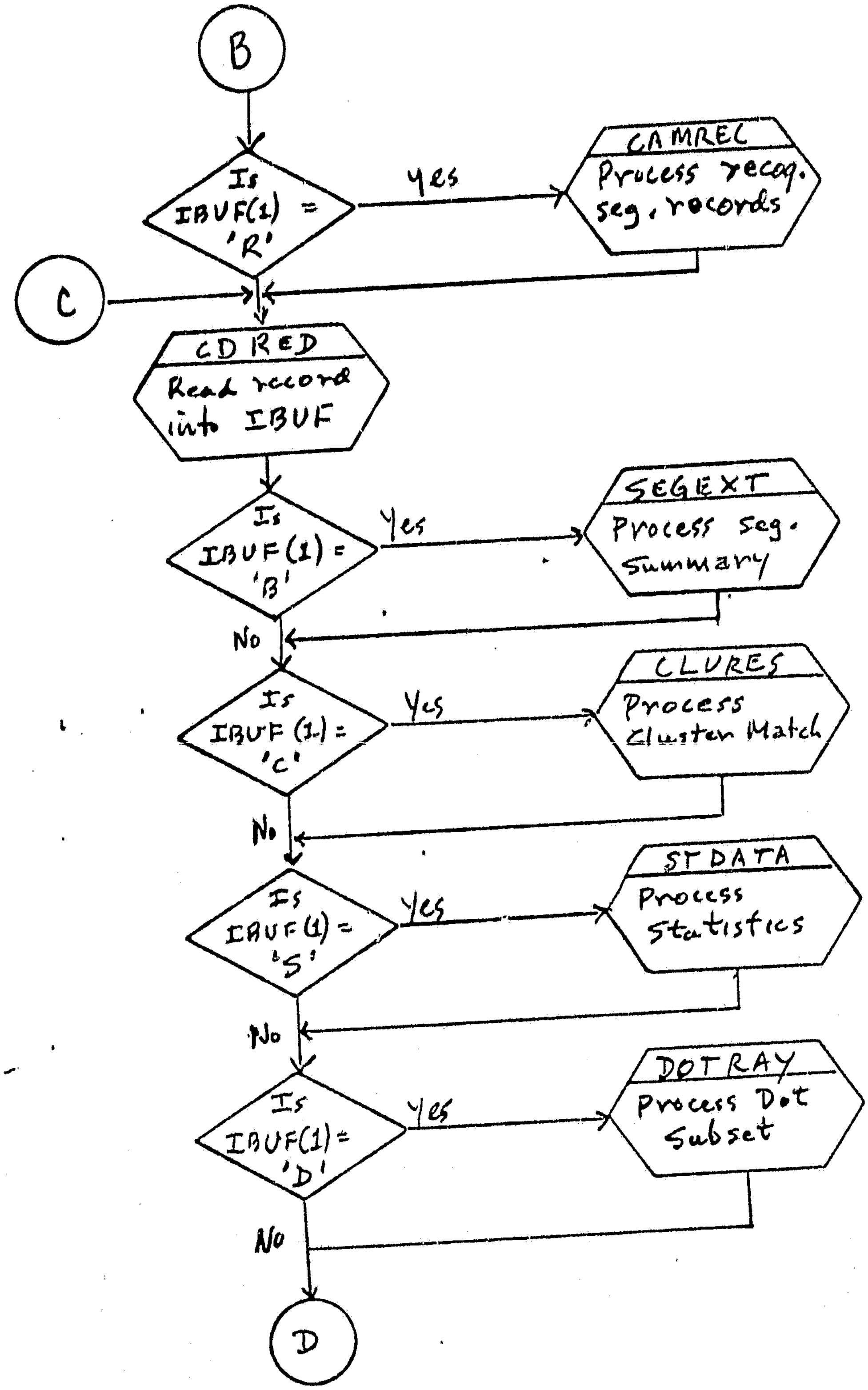
See Flow Diagram 3.

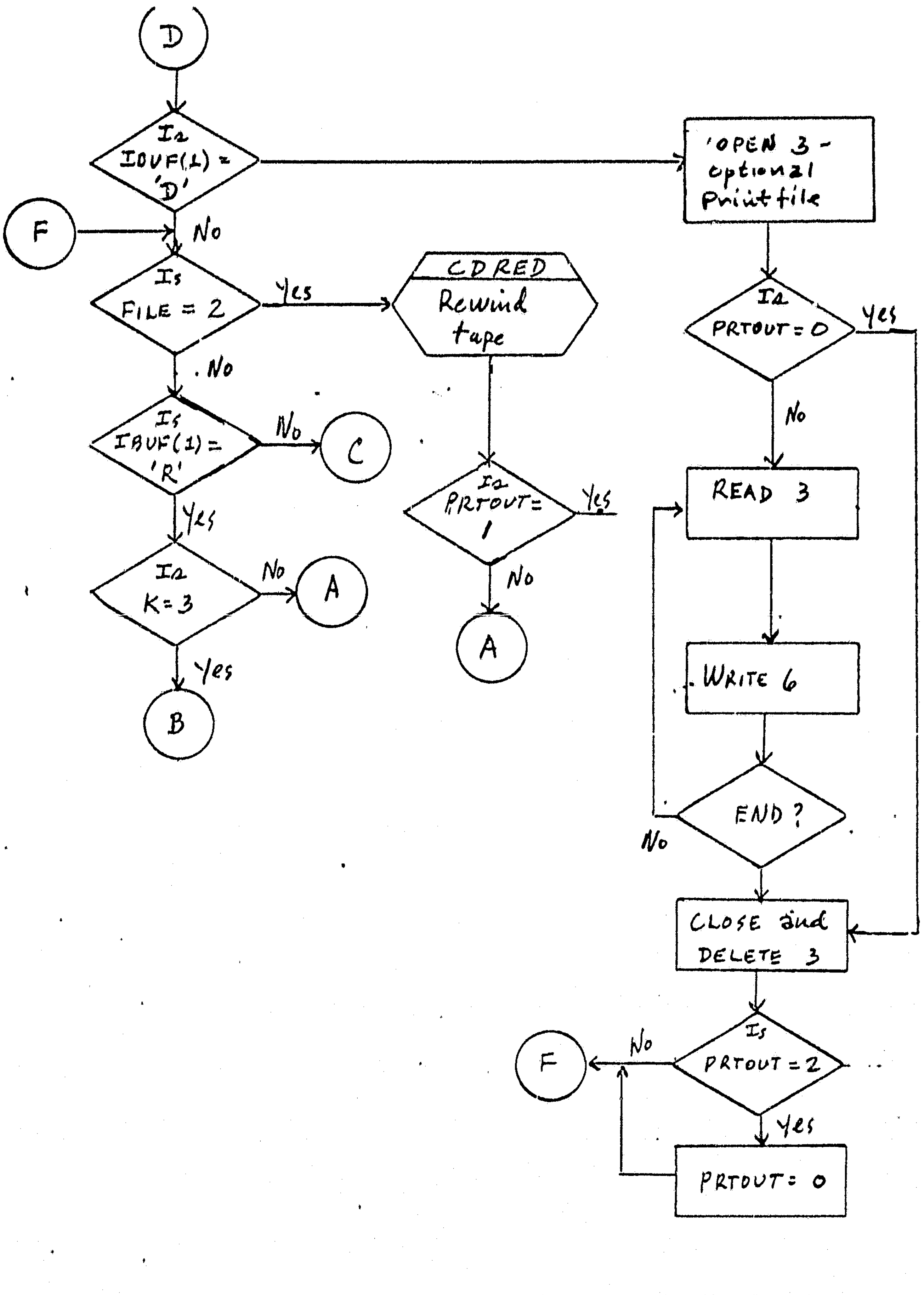
3.2.6.8 Listing

See Appendix A.

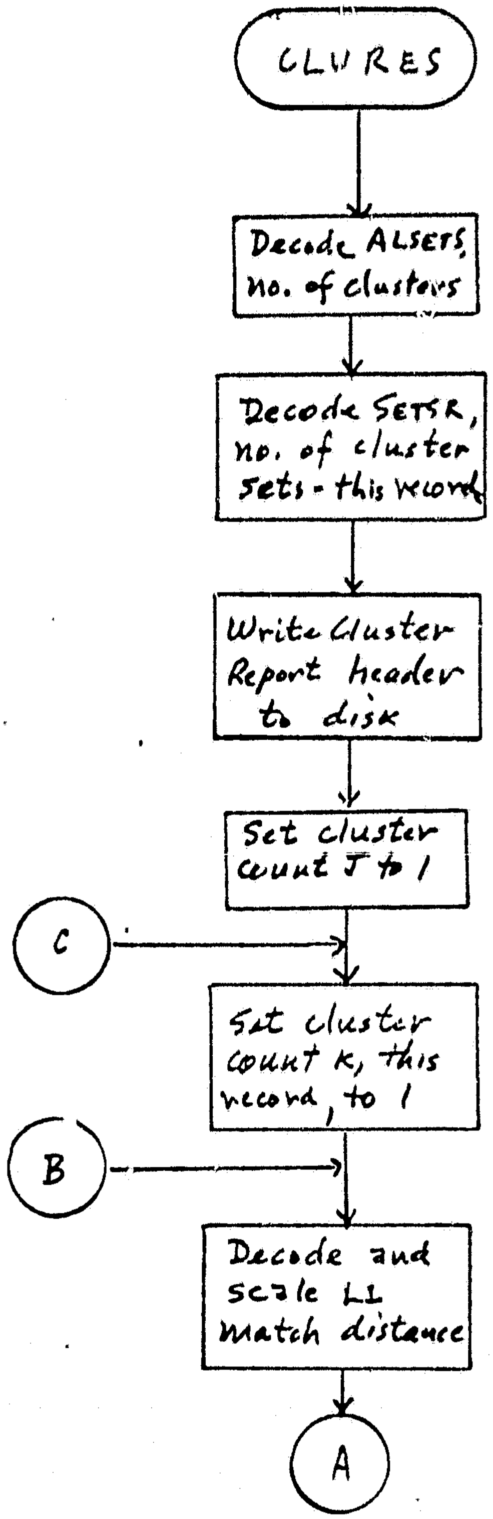


Flow Diagram 1 (page 1 of 3)

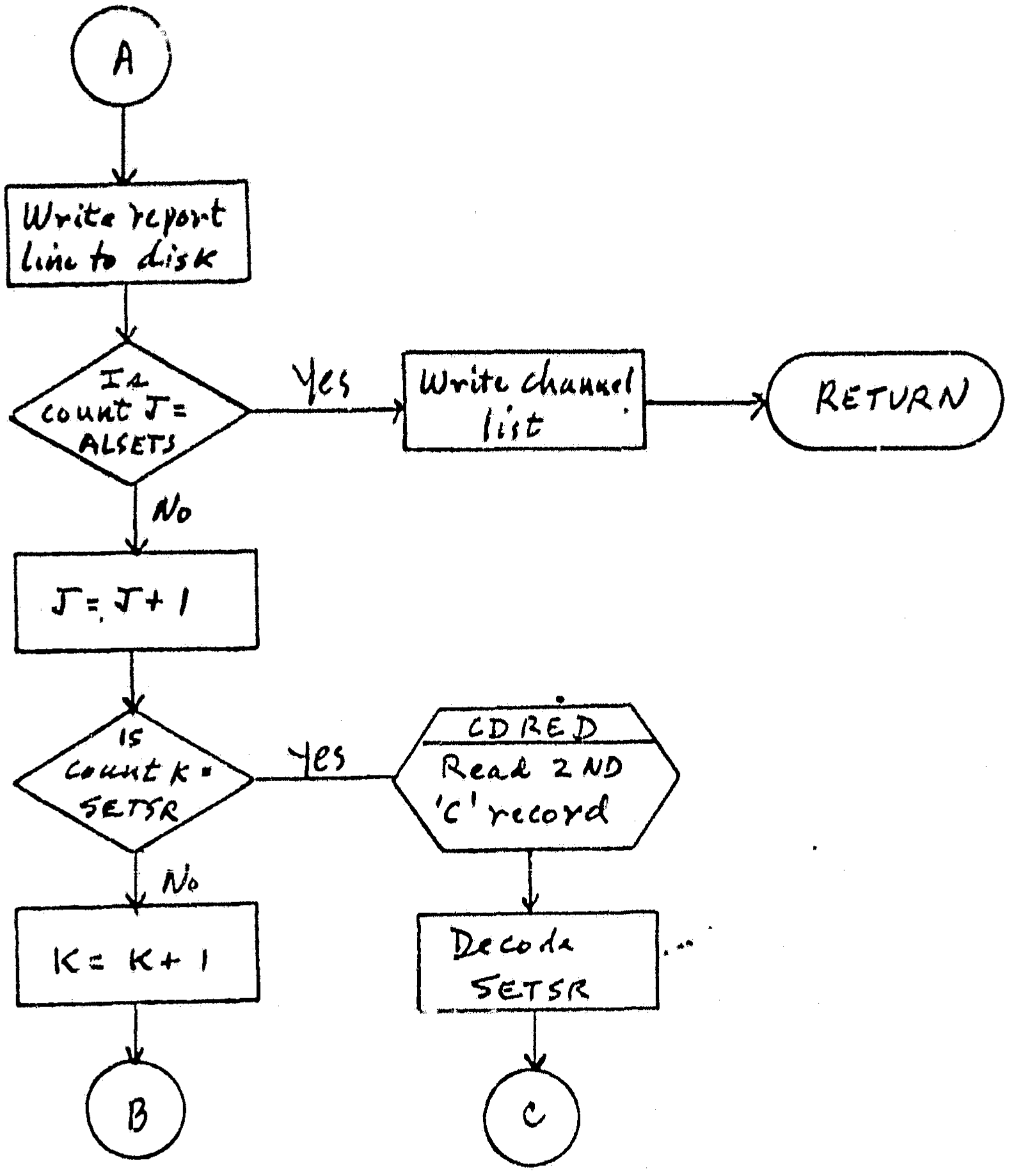

B
Is IBUF(1) = 'R'
yes
CAMREL
Process recog. seg. records
C
CDRED
Read record into IBUF
Is IBUF(1) = 'B'
Yes
SEGEXT
Process seg. summary
No
Is IBUF(1) = 'C'
Yes
CLURES
Process Cluster Match
No
Is IBUF(1) = 'S'
Yes
STDATA
Process Statistics
No
Is IBUF(1) = 'D'
Yes
DOTRAY
Process Dot Subset
No
D



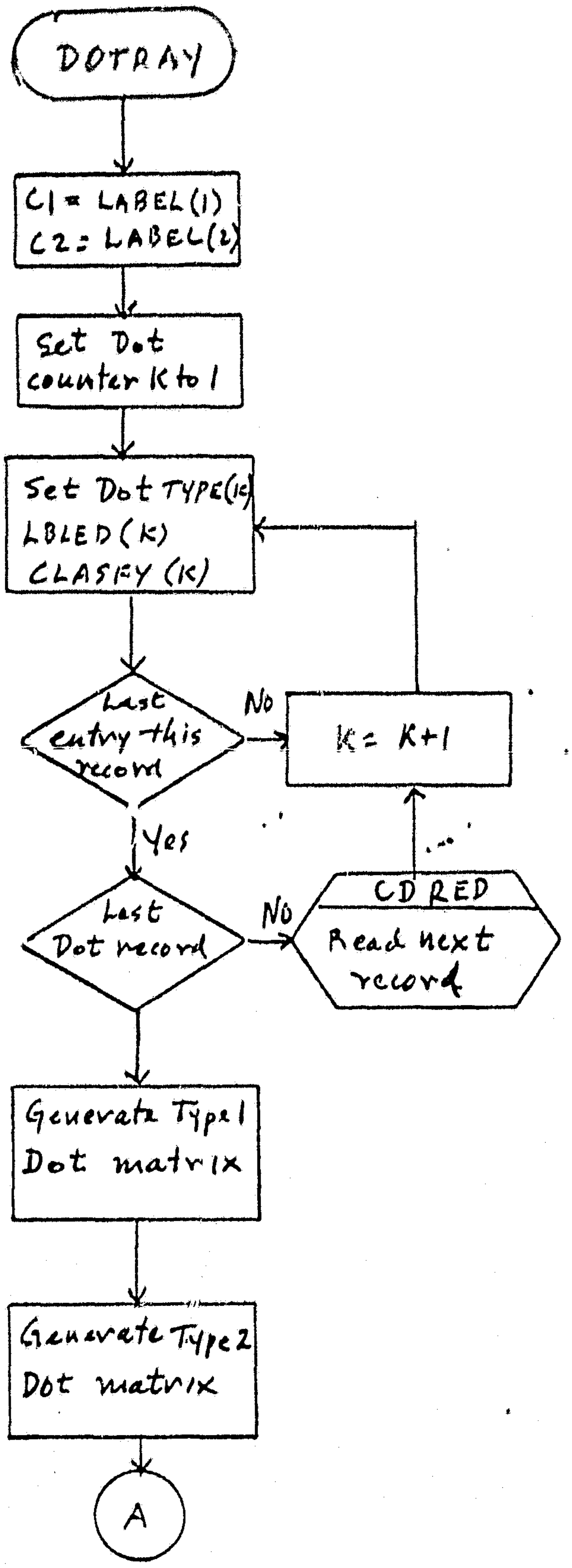
Flow Diagram I 17 (page 3 4


CLURES
Decode ALSETS, no. of clusters
Decode SETSR, no. of cluster sets - this record
Write Cluster Report header to disk
Set cluster count J to 1
C
Set cluster count K, this record, to 1
B
Decode and scale L1 match distance
A

Flow Diagram 2 (page 1 of 2,



Flow Diagram 2 (page 2 of 2

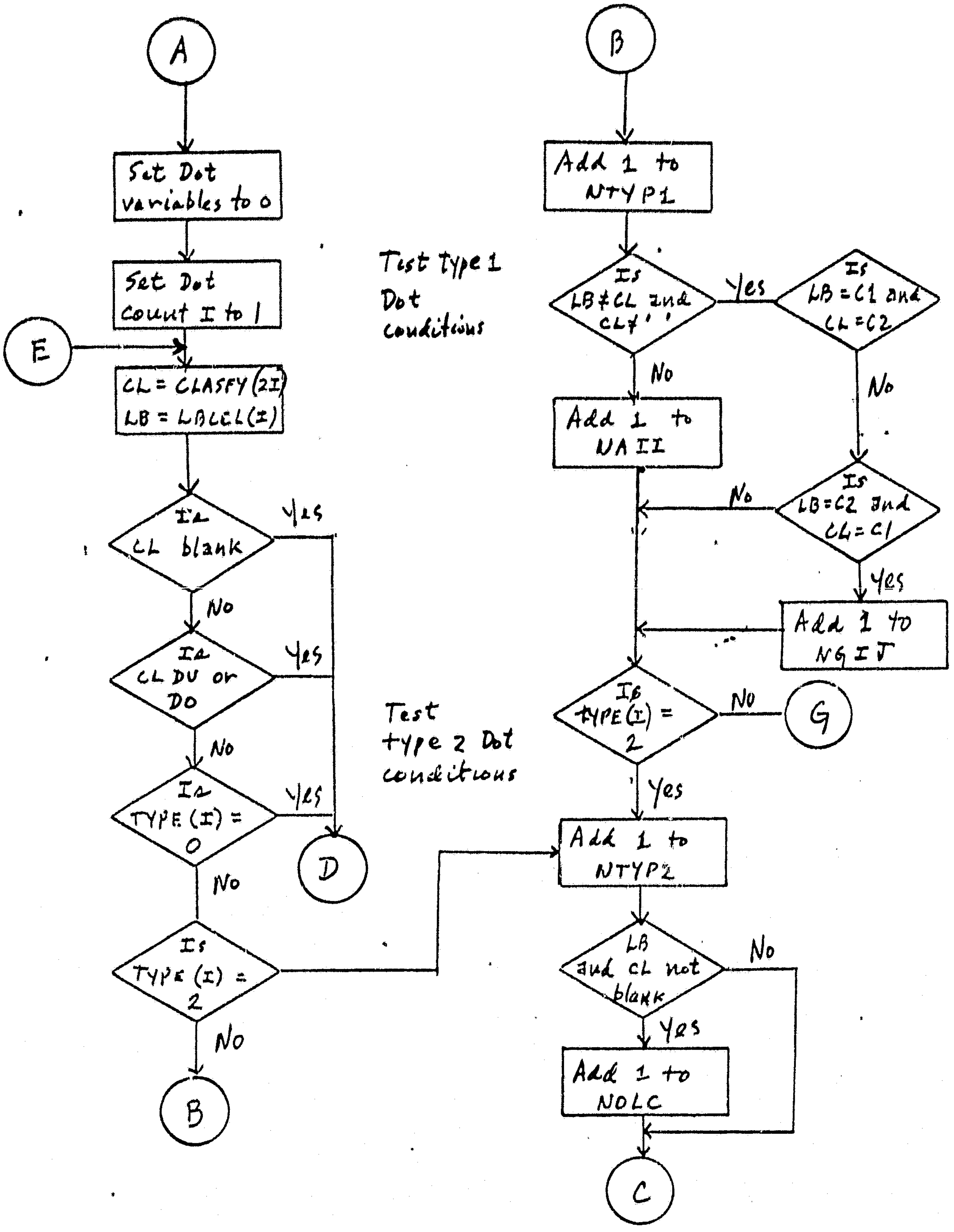


Set user symbols for categories 1 & 2

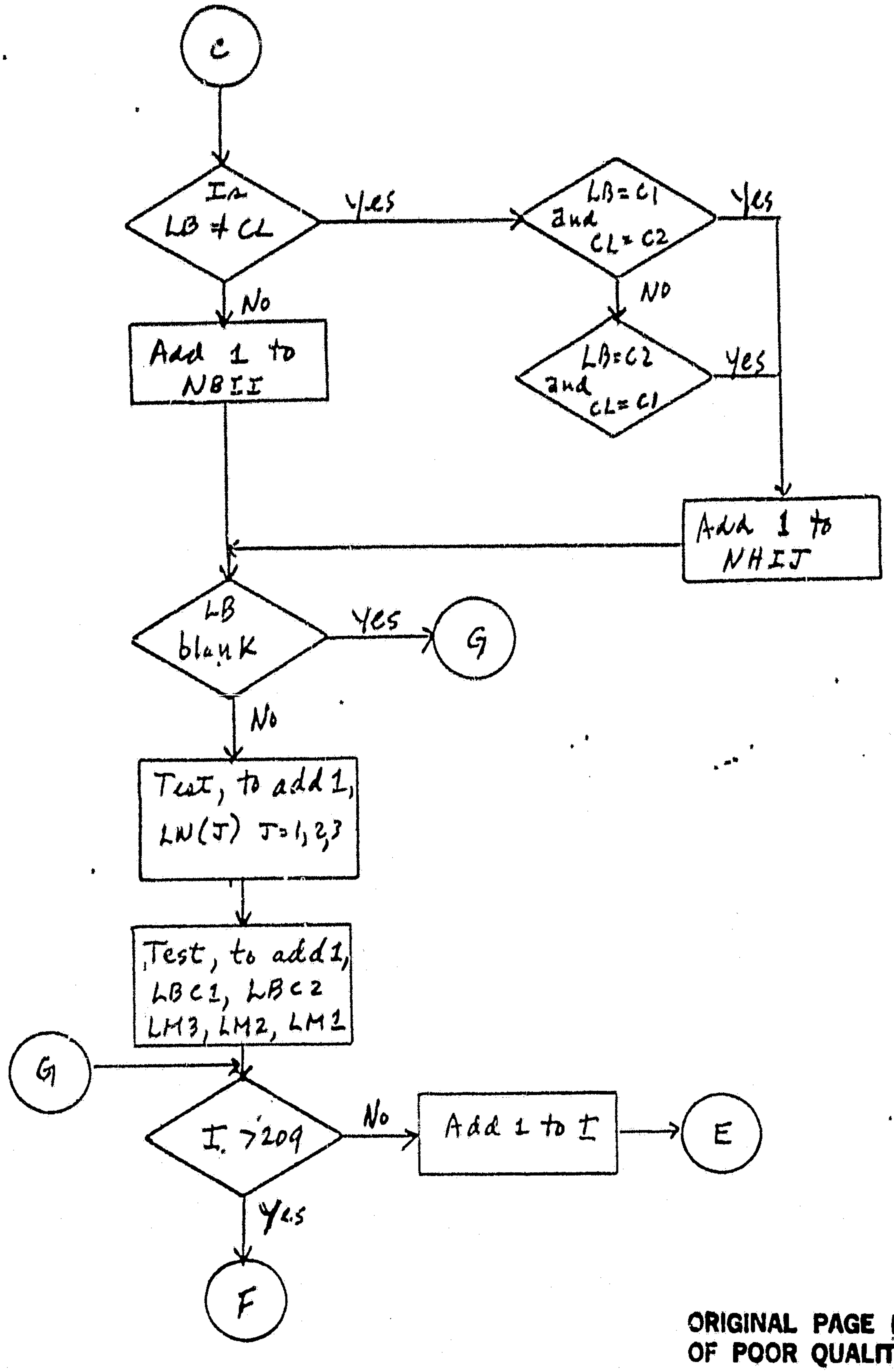
Process Dot Tape records

Flow Diagram 3

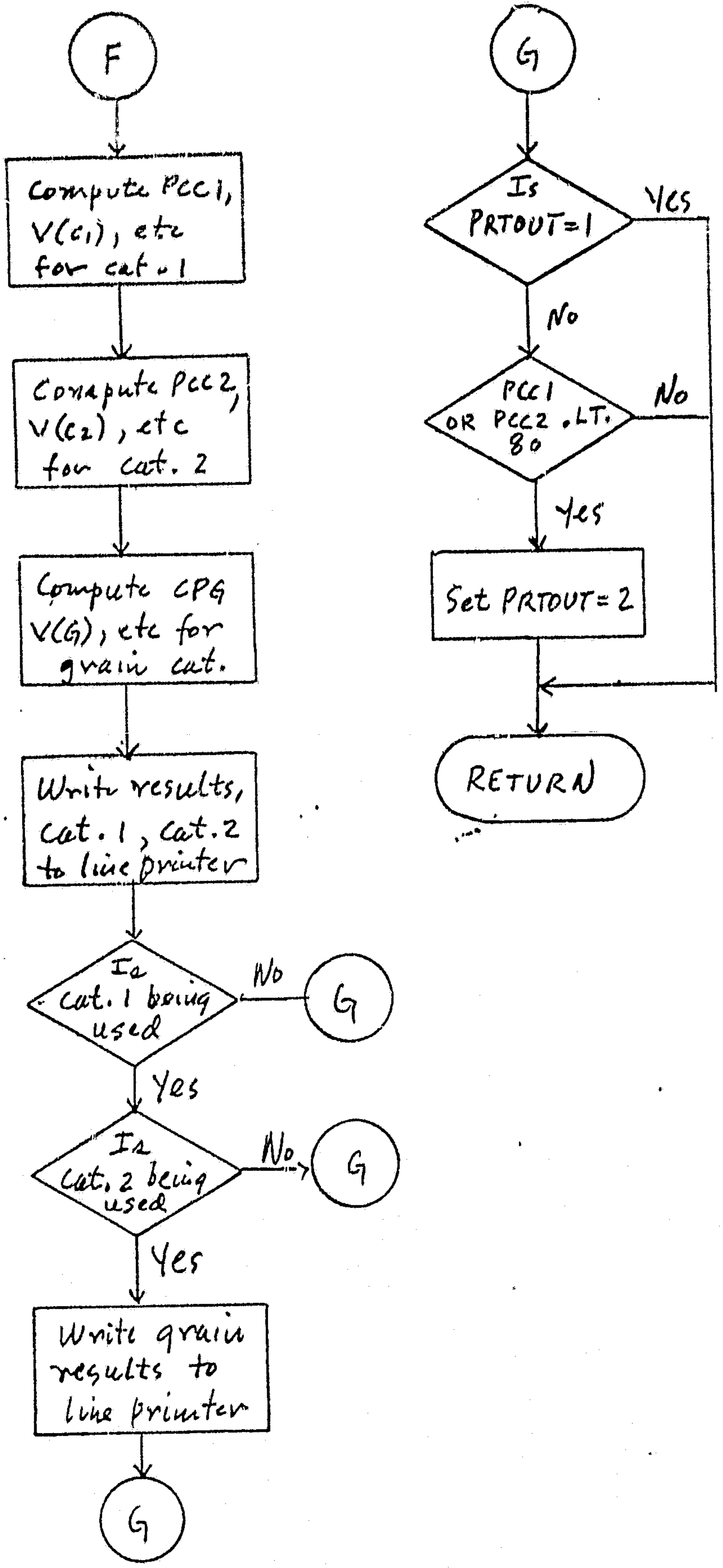
(page 1 of 4)



Flow Diagram 3 (page 2 of 4)



Flow Diagram 3 (page 3 of 4)
~~3-20~~ 22


F
Compute PCC1, V(C1), etc for cat. 1
Compute PCC2, V(C2), etc for cat. 2
Compute CPG V(G), etc for grain cat.
Write results, cat. 1, cat. 2 to line printer
Is cat. 1 being used
No
G
Yes
Is cat. 2 being used
No
G
Yes
Write grain results to line printer
G
G
Is PRTOUT=1
Yes
No
PCC1 OR PCC2 .LT. 80
No
Yes
Set PRTOUT=2
RETURN

Flow Diagram 3 (page 4 of 4)

4. OPERATING PROCEDURE

4.1 GENERAL

CAMRPT is an RSX-11D Batch program which reads a CAMS/CAS Interface tape and generates a series of reports which are output on the line printer. It requires data card inputs.

4.2 DECK SET UP

Two data cards define the input tape drive and unit number. The card formats are:

```
M or X (tape drive)
0 or 1 (tape unit)
```

Entries are keypunched in column 1. To execute the CAMRPT default option for a limited printout of reports, an END card must follow the two data cards above. If the option for a full output of all reports is desired, the control card sequence is:

```
A         (for all reports)
END
```

If the user desires to obtain the output for only a single segment on the input tape, segment 9681 for example, the control card sequence is:

```
S    9681
END
```

To obtain the output for segment 9681, and all segments following 9681, the control card sequence is:

```
S    9681
A
END
```

The above option is used when there is a bad segment on the input tape, to obtain the output for segments following the bad segment.

The Batch deck set up for the CAMRPT default option using input tape unit MTO is as follows:

```
$JOB/NAME=CAMRPT/MCR/LIMIT=99/ACCOUNT=5050
$DATA
M
O
END
$EOD
$RUN CAMRPT
$EOJ
```

To run the program, mount the CAMS/CAS Interface tape and enter a mount message.

For MTO the message would be:

```
MCR > MOU MTØ:/CHA=[FOR]      (CR)
```

Then load the card reader with the CAMRPT Batch deck and enter BAT CR:, to read in the deck.

APPENDIX A

LISTING

```
            IMPLICIT INTEGER(A-Z)
            BYTE PBUF(120)
            LØGICAL*1 IBUF(3060)
            LØGICAL*1 CCHAR
            LØGICAL*1 CD(80)
            LØGICAL*1 CØNTR(4)
            LØGICAL*1 SEGNØ(4),RECØRD(12)
            CØMMØN/SEG/SEGNØ
           CØMMØN/PCNT/LINE,PAGE
            DATA CØNTR/1HS,1HR,1HA,1HE/
            PRTØUT=0
            K=0
            R=0
            FILE=0
            TAPNØ=0
            CALL CDRED(IBUF,R,FILE)
            R=1
            CALL CDRED(IBUF,R,FILE)
   C        DECØDE(4,100,IBUF(44))  TAPNØ
     100    FØRMAT(I4)
   15       CØNTINUE
            R=1
            READ(1,99)  CD
   99       FØRMAT(80A1)
            DØ 20 I=1,72
            IF(CD(I).NE.' ')   GØ TØ 21
   20       CØNTINUE
            GØ TØ 9
   21       CCHAR=CD(I)
            DØ  1  K=1,4
            IF(CØNTR(K).EQ.CCHAR)  GØ TØ 2
   1        CØNTINUE
   22       CØNTINUE
            WRITE(6,88)
   88       FØRMAT('1',10X,'  BAD DATA CARD ')
            STØP
   18       FILE=0
            R=-1
            CALL CDRED(IBUF,R,FILE)
            IF(PRTØUT.EQ.0)  STØP
            GØ TØ 15
     2      CØNTINUE
            IF(TAPNØ.EQ.0) DECØDE(4,100,IBUF(44))  TAPNØ
            GØ TØ(31,31,34,35)K
   35       CØNTINUE
            IF(PRTØUT.EQ.1)  GØ TØ 5
            K=3
            GØ TØ 6
     5      IF(FILE.EQ.0)  STØP
            R=-1
            CALL CDRED(IBUF,R,FILE)
            STØP
   31       CØNTINUE
            PRTØUT=1
            F=0
            I=1
```

```
        4       IF(CD(I).NE.' ')  GØ TØ 3
                I=I+1
                IF(I.GT.72)  GØ TØ 22
                GØ TØ 4
        3       F=F+1
                IF(F.EQ.1)  GØ TØ 7
                IF(K.EQ.1)  GØ TØ 8
                IF(F.EQ.2)  GØ TØ 7
                GØ TØ 8
        7       I=I+1
                IF(CD(I).EQ.' ')  GØ TØ 4
                IF(I.GT.72)  GØ TØ 22
                GØ TØ 7
        C       WRITE(6,102)  (CD(J),J=I,I+3)
        8       CØNTINUE
                IF(K.EQ.2)  GØ TØ 12
                JJ=1
                DØ  11  J=I,I+3
                SEGNØ(JJ)=CD(J)
                JJ=JJ+1
        11      CØNTINUE
                GØ TØ 13
        12      CØNTINUE
                JJ=1
                DØ  14  J=I,I+12
                DIF=J-I+1
                IF(DIF.EQ.7)  GØ TØ 14
                RECØRD(JJ)=CD(J)
                JJ=JJ+1
        14      CØNTINUE
        13      CØNTINUE
                GØ TØ 47
        41      CØNTINUE
                CALL CDRED(IBUF,R,FILE)
                IF(FILE.LT.2)  GØ TØ 47
                WRITE(6,89)
        89      FØRMAT(1H0,10X,'SEGMENT NØ, ØR RECØRD ID NØT FØUND')
                GØ TØ 18
        47      CØNTINUE
                R=R+1
        C       WRITE(6,42) R
        42      FØRMAT(10X,' R= ',I4)
                IF(IBUF(1).NE.'R')  GØ TØ 41
                IF(IBUF(2).GT.'1')  GØ TØ 41
                IF(K.EQ.2)  GØ TØ 16
                DØ  43  I=1,4
                IF(SEGNØ(I).NE.IBUF(147+I))  GØ TØ 41
        43      CØNTINUE
                SFLG=1
        C       WRITE(6,45)
        45      FØRMAT(10X,'  FØUND SEGMENT  ')
                GØ TØ 46
        48     CALL CDRED(IBUF,R,FILE)
        34      CØNTINUE
                PRTØUT=1
         6      R=R+1
```

```
0109       46   CØNTINUE
0110            IF(IBUF(1),EQ,'R')  CALL CAMREC(IBUF,R,FILE,TAPNØ)
0111       49   CØNTINUE
0112            CALL CDRED(IBUF,R,FILE)
0113            IF(IBUF(1),EQ,'B') CALL SEGEXT(IBUF,R,FILE)
0114            IF(IBUF(1),EQ,'C') CALL CLURES(IBUF,R,FILE)
0115            IF(IBUF(1),EQ,'S')  CALL SIDATA(IBUF,R,FILE,DFLG)
0116            IF(IBUF(1),EQ,'D') CALL PRIMAY(IBUF,R,FILE,PRTØUT)
0117            IF(IBUF(1),EQ,'D')  GØ TØ 25
      C         IF(IBUF(1),EQ,'F')  CALL F DPER(IBUF,R,FILE)
0118       19   IF(FILE,EQ,2)  GØ TØ 18
0119            IF(IBUF(1),NE,'R')  GØ TØ 49
0120            IF(K,NE,3) GØ TØ 15
0121            GØ TØ 46
0122      44    CØNTINUE
0123      16    CØNTINUE
          C     WRITE(6,999)  (RECØRD(LL),LL=1,12),(IBUF(708+LN),LN=1,12)
0124      999   FØRMAT(12X,12A1,4X,12A1)
0125            DØ 17  I=1,12
0126            IF(RECØRD(I),NE,IBUF(708+I))  GØ TØ 41
0127      17    CØNTINUE
          C     WRITE(6,104)
0128      104   FØRMAT(10X,' FØUND RECØRD ')
0129            GØ TØ 16
0130      102   FØRMAT(10X,4A1)
0131      9     WRITE(6,98)
0132      98    FØRMAT('1',10X,'  BLANK DATA CARD  ')
0133            GØ TØ 18
0134       25   CØNTINUE
0135            CLØSE(UNIT=3)
0136            ØPEN(UNIT=3,TYPE='ØLD',NAME='XXXX,DAT')
0137            IF(PRTØUT,EQ,0)  GØ TØ 10
0138       26   READ(3,500,END=10)  PBUF
0139            WRITE(6,500)  PBUF
0140       500  FØRMAT(120A1)
0141            GØ TØ 26
0142       10   CLØSE(UNIT=3,DISPØSE='DELETE')
0143            IF(PRTØUT,EQ,2) PRTØUT=0
0144            GØ TØ 19
0145            END
```

```
CAMREC.FTN     /TR:BLOCKS/WR
               SUBRØUTINE CAMREC(IBUF,R,FILE,TAPNØ)
               IMPLICIT INTEGER(A-Z)
               LØGICAL*1  IBUF(1),CLIST(249),CLASS(4)
                BYTE PB(132),BLANK,PT,LABEL(5)
                LØGICAL*1 SEGNØ(4)
                DIMENSIØN CT(16)
                REAL A(3,2),PER(6)
                INTEGER NI(5)
               CØMMØN/PCNT/LINE,PAGE
                CØMMØN/CBIAS/A,NI,PER,DØ,DU,TC,LABEL
               INTEGER WI(60)
               REAL PW(60),X,Y,PCW,PNW
               REAL  PDU,PDØ,PTH,PX,PWTH
                CØMMØN/SEG/SEGNØ
                DATA BLANK/1H /
                DATA PT/1H,/
                DØ 11  I=1,132
       11       PB(I)=BLANK
                CALL CAMHDG(IBUF,TAPNØ)
                ØPEN(UNIT=3,TYPE='NEW',NAME='XXXX.DAT')
                LINE=70
                CALL BNT
                WRITE(3,99)
       99      FØRMAT('0',28X,'CLASSIFICATIØN DATA')
                WRITE(3,90)
       94       FØRMAT(2H0 )
                WRITE(3,98)
       98       FØRMAT(1H ,53X,'NØRMALIZED')
                II=0
                DØ 95 JJ=1,16
                IF(IBUF(83+JJ).EQ.'1')  GØ TØ 96
                GØ TØ 95
       96       II=II+1
                CT(II)=JJ
       95       CØNTINUE
                IIMAX=II
                WRITE(3,97)
       97       FØRMAT(1H ,16X,' CLUSTER NAME     THRESHØLD VALUE     ',
             *'APRIØRI VALUE     ')
                CALL BNT
                WRITE(3,90)
         90     FØRMAT('  ')
               LINE=LINE+5
               K=0
               J=1
               N=1
               RJ=249
               SETSR=14
               LFLD=32
               WIT=0
               XC=0
               TC=0
               DØ  3 II=1,60
        3      WI(II)=0
               DECØDE(4,100,IBUF(56))  NØSUB
      100      FØRMAT(I4)
```

```
                DECODE(5,101,IBUF(64))  DØ
      101       FORMAT(I5)
                DECODE(5,101,IBUF(69))  DU
                GØ TØ 2
      1         CALL CDRED(IBUF,R,FILE)
                RJ=3
                SETSR=22
                N=1
      2         CONTINUE
                CALL CFIPP(IBUF(RJ),CLASS,PI,PØ)
                 CALL MV(IBUF(RJ),PB(28),6)
                 CALL MV(IBUF(RJ+15),PB(45),3)
                 PB(49)=PT
                 PB(50)=IBUF(RJ+18)
                 CALL MV(IBUF(RJ+19),PB(66),3)
                 WRITE(3,111)  (PB(PJ),PJ=21,79)
                CALL BNT
      111        FØRMAT(20X,60A1)
                TC=TC+PØ
                IF(CLASS(1).EQ.'X')  GØ TØ 10
                IF(CLASS(1).EQ.'W')  GØ TØ 20
                GØ TØ 30
      10        XC=XC+PI
      30        IF(J.EQ.NØSUB)  GØ TØ 40
                J=J+1
                IF(N.EQ.SETSR)  GØ TØ 1
                N=N+1
                RJ=RJ+LFLD
                GØ TØ 2
      20        IF(K.EQ.0)  GØ TØ 5
                LK=4*(K-1)
                DØ  4 II=1,4
                IF(CLASS(II).NE.CLIST(LK+II))  GØ TØ 5
      4         CØNTINUE
                WI(K)=WI(K)+PI
                 WIT=WIT+PØ+PI
                GØ TØ 30
      5         K=K+1
                LK=4*(K-1)
                DØ  6 II=1,4
      6         CLIST(LK+II)=CLASS(II)
                WI(K)=WI(K)+PI
                 WIT=WIT+PØ+PI
                GØ TØ 30
      40        CØNTINUE
                 IF(1.EQ.1)  RETURN
                 WRITE(3,94)
                CALL BNT
                CALL BNT
                 WRITE(6,93)  (CT(II),II=1,IIMAX)
                CALL BNT
      93         FØRMAT(25X,'CHANNELS USED: ',16(1X,I2))
                PC=22932
                D=PC-DØ-DU-XC
                Y=FLØAT(XC)+FLØAT(DU)
                DØ  41 JJ=1,K
                X=FLØAT(WI(JJ))
                PW(JJ)=(X+(X*Y)/FLØAT(D))/FLØAT(PC)
                 PW(JJ)=PW(JJ)*100.
        41      CØNTINUE
                PCW=0.
                DØ 42  JJ=1,K
                PCW=PCW+PW(JJ)
        42      CØNTINUE
                PNW=100.-PCW
                PDU=(FLØAT(DU)/FLØAT(PC))*100.
                PDØ=(FLØAT(DØ)/FLØAT(PC))*100.
                D=PC-DØ-DU
                PTH=(FLØAT(TC)/FLØAT(D))*100.
                PX=(FLØAT(XC)/FLØAT(PC))*100.
                X=FLØAT(WIT)
                PWTH=(X+(X*Y)/FLØAT(D))/FLØAT(PC)
                PWTH=PWTH*100.
                WRITE(3,200)
                CALL BNT
                CALL BNT
        200     FØRMAT('0',21X,'SEGMENT PERCENTAGES')
                 WRITE(3,94)
                CALL BNT
                CALL BNT
                WRITE(6,201)  PCW
                CALL BNT
        201     FØRMAT(1H ,21X,'WHEAT CATEGØRY      = ',F5.1)
                 WRITE(6,202)  PNW
                CALL BNT
        202      FØRMAT(1H ,21X,'NØN WHEAT CATEGØRY = ',F5.1)
                 WRITE(6,203)  PDØ
                CALL BNT
        203      FØRMAT(1H ,21X,'DESIGNATED ØTHER   = ',F5.1)
                 WRITE(6,204)  PTH
                CALL BNT
        204      FØRMAT(1H ,21X,'THRESHØLD          = ',F5.1)
                 WRITE(6,205)  PDU
                CALL BNT
        205      FØRMAT(1H ,21X,'DESIGNATED UNIDENT = ',F5.1)
                 WRITE(6,206)  PX
                CALL BNT
        206      FØRMAT(1H ,21X,'X CATEGØRY         = ',F5.1)
                 WRITE(6,94)
                CALL BNT
                CALL BNT
                 DØ  207  I=1,K
                 J1=1
                 J2=4
                 WRITE(6,208)  (CLIST(JJ),JJ=J1,J2),PW(I)
                 J1=J1+4
                 J2=J2+4
        207      CØNTINUE
        208      FØRMAT(1H ,21X,'WHEAT CLASS ',4A1,'    = ',F5.1)
                 WRITE(6,210)  PWTH
                CALL BNT
                CALL BNT
```

```
0168    210       FØRMAT(1H0,21X,'WHEAT, 0 THRESHØLD:  ',F5.1)
0169              RETURN
0170              END
```

```
FØRTRAN IV-PLUS V02-04                      10:12:46    07-OCT-77
SEGEXT.FTN        /TR:BLØCKS/WR
                  SUBRØUTINE SEGEXT(IBUF,N,FILE)
                  IMPLICIT INTEGER(A-Z)
                  BYTE IBUF(1),LABEL(5)
                  REAL A(3,2),PER(6),X
                  INTEGER NI(5)
                  CØMMØN/CBIAS/A,NI,PER,DØ,DU,TC,LABEL
                  CAT=1
                  RJ=7
                  LFLD=47
                  DECØDE(2,102,IBUF(6)) CATS
          102     FØRMAT(I2)
          1       CØNTINUE
                  LABEL(CAT)=IBUF(RJ+1)
                  DECØDE(5,105,IBUF(RJ+2)) PØP
                  NI(CAT)=PØP
                  IF(CAT.GT.3) GØ TØ 7
                  DECØDE( 3,103,IBUF(RJ+7)) UNVAL
                  PER(CAT)=UNVAL
                  DECØDE(3,103,IBUF(RJ+10)) BIVAL
                  PER(CAT+3)=BIVAL
          105     FØRMAT(I5)
                  IF(CAT.GT.2)  GØ TØ 7
                  ØFF=27
                  DØ 8 I=1,3
                  DECØDE(3,103,IBUF(RJ+ØFF)) IV
          103     FØRMAT(I3)
                  X=IV
                  A(I,CAT)=X/100.
                  ØFF=ØFF+3
          8       CØNTINUE
          7       IF(CAT.EQ.CATS) GØ TØ 2
                  CAT=CAT+1
                  RJ=RJ+LFLD
                  GØ TØ 1
          2       CØNTINUE
        D         WRITE(6,100) A,NI
          100     FØRMAT('1',10X,'A:  ',6(F4.2,2X),'NI:  ',5I6)
                  RETURN
                  END
```

```
HFØRTRAN IV-PLUS V02-04                         10:12:52    07-OCT-77                PA
CLURES,FTN          /TRIBLØCKS/WR

                    SUBRØUTINE CLURES(IBUF,R,FILE)
                    IMPLICIT INTEGER(A-Z)
                     INTEGER*4 L2DIST
                    REAL XL2
                    BYTE IBUF(1)
                    DIMENSIØN CT(16)
                     CØMMØN/PCNT/LINE,PAGE
                    DECØDE(2,102,IBUF(4))  ALSETS
                     IF(ALSETS.EQ.0)  RETURN
                    DECØDE(2,102,IBUF(6))  SETSR
                     IF(SETSR.EQ.0)  RETURN
              102   FØRMAT(I2)
                    I=0
                    DØ 10 J=1,16
                    IF(IBUF(7+J).EQ.'1')  GØ TØ 12
                    GØ TØ 10
              12    I=I+1
                    CT(I)=J
              10    CØNTINUE
                    CHMAX=I
        C
        C WRITE CLUSTER RESULTS HEADER
        C
                     IF(LINE.GT.45) LINE=69
                     CALL BNT
                     IF(LINE.EQ.0)  GØ TØ 3
                     CALL BNT
                    WRITE(3,98)
              98    FØRMAT('0 ')
              3      CØNTINUE
                     CALL BNT
                     CALL BNT
                     WRITE(3,100)
              100    FØRMAT('0',28X,'CLUSTER REPØRT ')
                     CALL BNT
                     WRITE(3,99)
                     CALL BNT
                     CALL BNT
                     WRITE(3,110)  ALSETS
              110    FØRMAT('0',16X,'CLUSTERS GENERATED: ',I2)
                     CALL BNT
                     CALL BNT
                    WRITE(3,91)
              91    FØRMAT('0',16X,'CLUSTER NAME',2X,'SUBCLASS/DØT',2X,'L2  MAT
                     CALL BNT
                    WRITE(3,92)
              92    FØRMAT(34X,'MATCH',6X,'DISTANCE')
                     CALL BNT
                    WRITE(3,99)
              99    FØRMAT('  ')
                    RJ=23
                    LFLD=17
                    J=1
              1     CØNTINUE
                    K=1
              2     CØNTINUE
```

~~A-9~~ 35.

```
                C
                C PRØCESS A CLUSTER DATA SET
                C
0054                  DECØDE(5,105,IBUF(RJ+13)) L2DIST
0055            105    FØRMAT(I5)
0056                  XL2=L2DIST
0057                  XL2=XL2/100.
0058                   CALL BNT
0059                  WRITE(3,93) (IBUF(RJ+L),L=1,6),(IBUF(RJ+6+M),M=1,6),XL2
0060            93    FØRMAT(20X,6A1,6X,6A1,6X,F6.2)
                C
                C TEST FØR END ØF DATA
                C
0061                  IF(J.EQ.ALSETS)  GØ TØ 4
0062                  J=J+1
0063                  IF(K.EQ.SETSR)  GØ TØ 13
0064                  K=K+1
0065                  RJ=RJ+LFLD
0066                  GØ TØ 2
                C
                C READ 2ND CLUSTERING RECØRD
                C
0067            13    CALL CDRED(IBUF,R,FILE)
0068                  RJ=23
0069                  DECØDE(2,102,IBUF(6))   SETSR
0070                  GØ TØ 1
0071            4     CØNTINUE
0072                   CALL BNT
0073                   CALL BNT
0074                   WRITE(3,50)
0075            50     FØRMAT('0',18X,'CLUSTERING CHANNEL LIST:')
0076                   CALL BNT
0077                   CALL BNT
0078                   WRITE(3,51)  (CT(I),I=1,CHMAX)
0079            51     FØRMAT('0',10X,16(1X,I2))
0080                   RETURN
0081                  END
```

ORIGINAL PAGE
OF POOR QUALITY

```
FØRTRAN IV-PLUS V02-04                  10:13:01    07-OCT-77          PA
DØTRAY.FTN          /TR:BLOCKS/WR
                    SUBRØUTINE DØTRAY(IBUF,R,FILE,PRTØUT)
                    REAL S(6),VC(2),A(3,2)
                    REAL PER(6)
                    INTEGER PAGE,FILE,R,REC,RJ,SETSR,PRTØUT
                    INTEGER MI(5),LA(3),CØL(19),DØ,DU,TC
                    BYTE IBUF(1),TYPE(209),LBLED(209),CLASFY(418)
                    BYTE BLK,ØNE,ØNETØ,SLASH,TWØ,LABEL(5)
                    BYTE LB,CL,C1,C2,C3,C4
                    BYTE FMATR(120)
                    CØMMØN/PØNT/LINE,PAGE
                    CØMMØN/CØIAS/A,MI,PER,DØ,DU,TC,LABEL
                    DATA ØNE/'1'/,SLASH/'/'/,TWØ/'2'/
                    DATA C1/'W'/,C2/'S'/,C3/'N'/,C4/'X'/
                    DATA BLK/' '/
                    C1=LABEL(1)
                    C2=LABEL(2)
                    LFLD=44
                    K=1
                    REC=1
                    SETSR=15
           2        CØNTINUE
                    RJ=12
                    N=1
           1        CØNTINUE
                    TYPE(K)=IBUF(RJ+12)
                    LBLED(K)=IBUF(RJ+11)
                    CLASFY(2*K)=IBUF(RJ+10)
                    CLASFY(2*K-1)=IBUF(RJ+9)
                    K=K+1
                    IF(N.EQ.SETSR)  GØ TØ 13
                    N=N+1
                    RJ=RJ+LFLD
                    GØ TØ 1
          13        IF(REC.EQ.14) GØ TØ 14
                    CALL CDRED(IBUF,R,FILE)
                    REC=REC+1
                    IF(REC.EQ.14) SETSR=14
                    GØ TØ 2
          14        CØNTINUE
      C             WRITE(6,100)
         100        FØRMAT('1 ')
                    WRITE(6,95)
                    WRITE(6,95)
                    WRITE(6,95)
          95        FØRMAT('0  ')
                    ØNETØ=ØNE
                    DØ 6 I=1,19
           6        CØL(I)=I*10
           4        K=1
                    NLINE=10
                    WRITE(6,102)  ØNETØ
         102        FØRMAT('0',42X,'TYPE ',A1,' DØT LABEL/CLASSIFICATIØN')
                    WRITE(6,101)
                    WRITE(6,103)
         103        FØRMAT(' ',1X,'PIXEL/')
                    WRITE(6,104)  (CØL(I),I=1,19)
      104     FØRMAT(' ',1X,'LINE',2X,19(I3,3X))
              DØ 11 J=1,11
              L=6
              DØ 10 I=1,120
              PMATR(I)=BLK
       10     CØNTINUE
              WRITE(6,101)
      101     FØRMAT('  ')
              DØ 12 K=1,19
              IF(TYPE(K).EQ.ØNETØ)  GØ TØ 7
              IF((TYPE(K).EQ.'3').AND.(ØNETØ.EQ.'1')) GØ TØ 7
              IF((TYPE(K).EQ.'0').AND.(ØNETØ.EQ.'2')) GØ TØ 7
              GØ TØ 5
        7     CØNTINUE
    C         CL=CLASFY(2*K)
    C         IF((CL.EQ.'0').ØR.(CL.EQ.'U'))  GØ TØ 5
              PMATR(L)=LBLED(K)
              PMATR(L+1)=SLASH
              PMATR(L+2)=CLASFY(2*K)
              IF(CLASFY(2*K-1).EQ.' ')  GØ TØ 5
              PMATR(L+2)=CLASFY(2*K-1)
              PMATR(L+3)=CLASFY(2*K)
        5     CØNTINUE
              K=K+1
              L=L+6
       12     CØNTINUE
              WRITE(6,99) NLINE,(PMATR(I),I=6,120)
       99     FØRMAT(' ',2X,I3,2X,115A1)
              NLINE=NLINE+10
       11     CØNTINUE
              IF(ØNETØ.EQ.TWØ) GØ TØ 3
              ØNETØ=TWØ
              WRITE(6,100)
              GØ TØ 4
        3     CØNTINUE
              NTYP1=0
              NTYP2=0
              NAII=0
              NGIJ=0
              NBII=0
              NHIJ=0
              LN(1)=0
              LN(2)=0
              LN(3)=0
              LM3=0
              LM2=0
              LM1=0
              LBC1=0
              LBC2=0
              NØLC=0
              DØ 39 I=1,209
              LB=LBLED(I)
              CL=CLASFY(2*I)
              IF(LB.EQ.' ')  GØ TØ 39
              IF((CL.EQ.'0').ØR.(CL.EQ.'U'))  GØ TØ 39
              IF(TYPE(I).EQ.'0')  GØ TØ 39
```

```
0110                IF(TYPE(I),EQ,'2')  GØ TØ 35
0111                NTYP1=NTYP1+1
0112                IF((LB,NE,CL),AND,(CL,NE,' '))  GØ TØ 33
0113                NAII=NAII+1
0114                GØ TØ 35
0115         33     IF((LB,EQ,C1),AND,(CL,EQ,C2)) NGIJ=NGIJ+1
0116                IF((LB,EQ,C2),AND,(CL,EQ,C1)) NGIJ=NGIJ+1
           C
           C
0117         35     IF(TYPE(I),NE,'2')  GØ TØ 39
0118                NTYP2=NTYP2+1
0119                IF((LB,NE,' '),AND,(CL,NE,' ')) NØLC=NØLC+1
0120                IF(LB,NE,CL)  GØ TØ 36
0121                NBII=NBII+1
0122                GØ TØ 37
0123         36     IF((LB,EQ,C1),AND,(CL,EQ,C2)) NHIJ=NHIJ+1
0124                IF((LB,EQ,C2),AND,(CL,EQ,C1)) NHIJ=NHIJ+1
0125         37     IF(LB,EQ,' ') GØ TØ 39
0126                IF(CL,EQ,C1) LN(1)=LN(1)+1
0127                IF(CL,EQ,C2) LN(2)=LN(2)+1
0128                IF(CL,EQ,C3) LN(3)=LN(3)+1
0129                IF(LB,EQ,C1)  LBC1=LBC1+1
0130                IF(LB,EQ,C2)  LBC2=LBC2+1
0131                IF((CL,EQ,C3),AND,(LB,EQ,C3)) LM3=LM3+1
0132                IF(LB,NE,C2)  GØ TØ 40
0133                IF((CL,EQ,C1),ØR,(CL,EQ,C2)) LM2=LM2+1
0134         40     IF(LB,NE,C1)  GØ TØ 39
0135                IF((CL,EQ,C1),ØR,(CL,EQ,C2)) LM1=LM1+1
0136         39     CØNTINUE
           D        WRITE(6,800) NTYP1,NTYP2,NAII,NBII,NGIJ,NØLC
0137        800     FØRMAT('1',9X,6I5)
           D        WRITE(6,900) NHIJ,LN(1),LN(2),LN(3),LM1,LM2,LM3,LBC1,LBC2
0138        900     FØRMAT('0',9X,9I5)
           D        WRITE(6,899)  (NI(I),I=1,5),DØ,DU,TC
0139        899     FØRMAT('0',10X,'NI : ',5I6,10X,3I5)
           D        WRITE(6,898)  LABEL
0140        898     FØRMAT('0',10X,'LABELS : ',5(2X,A1))
0141                PCC1=0.
0142                PCC2=0.
0143                PCCG1=0.
0144                PCCG2=0.
0145                IF(NTYP1,EQ,0) GØ TØ 45
0146                PCC1=(FLØAT(NAII)/FLØAT(NTYP1))*100.
0147                X=NGIJ+NAII
0148                PCCG1=(X/FLØAT(NTYP1))*100.
0149         45     IF(NTYP2,EQ,0)  GØ TØ 46
0150                PCC2=(FLØAT(NBII)/FLØAT(NTYP2))*100.
0151                X=NHIJ+NBII
0152                PCCG2=(X/FLØAT(NTYP2))*100.
           C
           C
0153         46     CØNTINUE
0154                IX=NI(4)
0155                BASE=22932-NI(4)
           D        WRITE(6,801) BASE
0156        801     FØRMAT(10X,'BASE= ',F8.1)
```

```
0157              DØ 47 K=1,6
0158        47    S(K)=0,
0159              K=1
0160              DØ 53 J=1,2
0161              DØ 52 I=1,3
0162              IF(LN(I),LE,1)  GØ TØ 49
0163              S(K)=(A(I,J)*(1,-A(I,J)))/FLØAT(LN(I)-1)
0164        49    K=K+1
0165        52    CØNTINUE
0166        53    CØNTINUE
         D        WRITE(6,808) S
0167        808   FØRMAT(10X,' S= ',6F8,4)
         C
0168              VC(1)=0,
0169              VC(2)=0,
0170              K=1
0171              DØ 51 J=1,2
0172              DØ 50 I=1,3
0173              X=(FLØAT(NI(I))/BASE)*100,
0174              VC(J)=VC(J)+X*X*S(K)
0175              K=K+1
0176        50    CØNTINUE
0177        51    CØNTINUE
         C
0178              PG1=0,
0179              PG2=0,
0180              RS1=0,
0181              RS2=0,
0182              NSUM=NI(1)+NI(2)+NI(3)
0183              PSUM=FLØAT(NSUM)/BASE
0184              IF(NØLC,EQ,0)  GØ TØ 70
0185              RS1=(FLØAT(LRC1)/FLØAT(NØLC))*PSUM
0186              RS2=(FLØAT(LRC2)/FLØAT(NØLC))*PSUM
0187        70    CØNTINUE
0188              PRS1=RS1*100,
0189              PRS2=RS2*100,
0190              NØLNI=LN(1)+LN(2)+LN(3)-1
         C
0191              VRS1=(RS1*(PSUM-RS1))/FLØAT(NØLNI)
0192              VRS1=VRS1*10000,
0193              VRS2=(RS2*(PSUM-RS2))/FLØAT(NØLNI)
0194              VRS2=VRS2*10000,
         D        WRITE(6,334) RS1,RS2,PSUM,VRS1,VRS2
0195        334   FØRMAT(10X,'RS1 : ',5F8,4)
         C
0196              IF(VC(1),EQ,0,)  GØ TØ 55
0197              PG1=((VRS1/VC(1))-1,)*100,
0198        55    IF(VC(2),EQ,0,)  GØ TØ 56
0199              PG2=((VRS2/VC(2))-1,)*100,
         C
0200        56    CG12=PER(1)+PER(2)
0201              M=LM1+LM2
0202              N=LN(1)+LN(2)
0203              ALG=0,
0204              IF(N,EQ,0)  GØ TØ 57
0205              ALG=FLØAT(M)/FLØAT(N)
```

```
      57   CØNTINUE
           N=NI(1)+NI(2)
           CPG=ALG*(FLØAT(N)/BASE)
           X=1.
           IF(LN(3).EQ.0)  GØ TØ 58
           X=1.-(FLØAT(LM3)/FLØAT(LN(3)))
      58   CØNTINUE
           X=X*(FLØAT(NI(3))/BASE)
           CPG=CPG+X
           CPG=CPG*100.
           ALN=0.
           IF(LN(3).EQ.0)  GØ TØ 59
 C
 C
           ALN=FLØAT(LM3)/FLØAT(LN(3))
      59   CØNTINUE
           X=(FLØAT(N)/BASE)*100.
           VG=0.
           IF((LN(1)+LN(2)-1).LE.0)  GØ TØ 60
           VG=X*X*((ALG*(1.-ALG))/FLØAT(LN(1)+LN(2)-1))
      60   CØNTINUE
 D         WRITE(6,333) ALG,ALN,X,VG
     333   FØRMAT(10X,'ALG... ',4F8.2)
           X=0.
           IF((LN(3)-1).LE.0)  GØ TØ 61
           X=(FLØAT(NI(3))/BASE)*100.
           X=X*X*((ALN*(1.-ALN))/FLØAT(LN(3)-1))
      61   CØNTINUE
           VG=VG+X
 C
           LSUM=LN(1)+LN(2)+LN(3)
           RSG=0.
           IF(LSUM.EQ.0)  GØ TØ 62
           RSG=(FLØAT(LBC1+LBC2)/FLØAT(LSUM))*PSUM
      62   CØNTINUE
           PRSG=100.*RSG
           VRSG=0.
           IF((LSUM-1).LE.0)  GØ TØ 63
           VRSG=(RSG*(PSUM-RSG))/FLØAT(LSUM-1)
           VRSG=VRSG*10000.
      63   CØNTINUE
           PGG=0.
           IF(VG.EQ.0.)  GØ TØ 64
           PGG=((VRSG/VG)-1.)*100.
      64   CØNTINUE
 C
 C
           WRITE(6,95)
           WRITE(6,499)
     499   FØRMAT('0',33X,'BIAS CØRRECTIØN REPØRT')
 C
           WRITE(6,500)  PCC1,PCC2
     500   FØRMAT('0',5X,'PCC1  ',F5.1,4X,'PCC2  ',F5.1)
 C
           WRITE(6,501)  PCCG1,PCCG2
     501   FØRMAT(' ',5X,'PCCG1 ',F5.1,4X,'PCCG2 ',F5.1)
```

```
      C
                  PDO=(FLØAT(NI(5))/BASE)*100.
                  WRITE(6,491) PDØ
            491   FØRMAT(' ',5X,'DESIGNATED ØTHER',3X,F5.1)
      C
                  PTH=(FLØAT(TC)/22932.)*100.
                  WRITE(6,492)  PTH
            492   FØRMAT(' ',5X,'THRESHØLD',10X,F5.1)
      C
                  PDU=(FLØAT(NI(4))/22932.)*100.
                  WRITE(6,493)  PDU
            493   FØRMAT(' ',5X,'UNIDENTIFIABLE',5X,F5.1)
      C
      C
                  WRITE(6,502)
            502   FØRMAT('0',8X,'CATEGØRY 1',18X,'CATEGØRY 2',19X,'CATEGØR'
      C
                  WRITE(6,503)  (PER(I),I=1,3)
            503   FØRMAT('0',4X,'CLASSIFIED PERCENTAGE ',F5.1,2X,'CLASSIFI
                 *' PERCENTAGE ',F5.1,2X,'CLASSIFIED PERCENTAGE ',F5.1)
      C
                  WRITE(6,504)  (PER(I),I=4,6)
            504   FØRMAT(' ',4X,'CØRRECTED PERCENTAGE  ',F5.1,2X,'CØRRECTE
                 *' PERCENTAGE  ',F5.1,2X,'CØRRECTED PERCENTAGE  ',F5.1)
      C
                  WRITE(6,505)  VC(1),VC(2)
            505   FØRMAT(' ',4X,'VARIANCE',14X,F5.2,2X,'VARIANCE',14X,F5.2
      C
                  WRITE(6,506)  PRS1,PRS2
            506   FØRMAT(' ',4X,'RANDØM SAMPLE EST.',4X,F5.2,2X,'RANDØM',
                 *' SAMPLE EST.',4X,F5.2)
      C
                  WRITE(6,507)  VRS1,VRS2
            507   FØRMAT(' ',4X,'VARIANCE',13X,F6.2,2X,'VARIANCE',13X,F6.2
                  WRITE(6,508)  PG1,PG2
            508   FØRMAT(' ',4X,'PERCENT GAIN',8X,F7.1,2X,'PERCENT GAIN',8
      C
      C
                  WRITE(6,101)
                  IF((LABEL(1).EQ.' ').ØR.(LABEL(2).EQ.' '))  GØ TØ 90
                  WRITE(6,510)
            510   FØRMAT('0',8X,'GRAIN CATEGØRY')
                  WRITE(6,511)  CG12
            511   FØRMAT('0',4X,'CLASSIFIED PERCENNTAGE',F5.1)
      C
                  WRITE(6,512)  CPG
            512   FØRMAT(' ',4X,'CØRRECTED PERCENTAGE  ',F5.1)
      C
                  WRITE(6,513) VG
            513   FØRMAT(' ',4X,'VARIANCE',13X,F6.2)
      C
                  WRITE(6,514)  PRSG
            514   FØRMAT(' ',4X,'RANDØM SAMPLE EST.',4X,F5.2)
                  WRITE(6,516)  VRSG
            516   FØRMAT(' ',4X,'VARIANCE',14X,F5.2)
      C
```

```
0291               WRITE(6,515)  PGG
0292        515    FØRMAT(' ',4X,'PERCENT GAIN',8X,F7.1)
0293        90     CØNTINUE
0294               IF(PRTØUT.EQ.1)  RETURN
0295               IF((PCC1.LT.80.).ØR.(PCC2.LT.80.))  PRTØUT=2
0296               RETURN
0297               END
```

```
              SUBRØUTINE CDRED(IBUF,R,FILE)
              IMPLICIT INTEGER(A-Z)
              INTEGER*2 ISTAT(2),IPRM(6)
               EQUIVALENCE(ISTAT(1),IP)
               BYTE IP(2)
              LØGICAL*1 IBUF(1)
              DIMENSIØN IHDAT(1)
              DIMENSIØN IA(37)
              DATA XDEV/2HXT/
              DATA MDEV/2HMT/
              RECØRD=R
              IF(RECØRD.GT.0)  GØ TØ 10
               ERCNT=0
               IF(RECØRD.EQ.-1)  GØ TØ 6
        12    CØNTINUE
              FILE=0
       D      WRITE(5,49)
        49    FØRMAT(10X,' TYPE M ØR X FØR TAPE DEVICE CØDE'/)
              READ(1,51,END=99)  IA
              IF(IA(1).EQ.'M')  GØ TØ 13
              IF(IA(1).EQ.'X')  GØ TØ 14
        99    CØNTINUE
              WRITE(6,97)
        97    FØRMAT(10X,' CARD ERRØR  '//)
              WRITE(6,97)
              STØP
        13    IDEV=MDEV
              GØ TØ 15
        14    IDEV=XDEV
        15    CØNTINUE
       D      WRITE(5,50)
        50    FØRMAT(10X,' TYPE TAPE UNIT NUMBER - 0 ØR 1')
              READ(1,51,END=99)  IA
        51    FØRMAT(37A2)
               IF(IA(1).EQ.'0')  GØ TØ 18
               IF(IA(1).EQ.'1')  GØ TØ 19
               GØ TØ 99
        18     IUNT=0
               GØ TØ 20
        19     IUNT=1
        20     CØNTINUE
              ILUN=9
              IDS=0
              ISR=0
              CALL ASNLUN(ILUN,IDEV,IUNT,IDS)
              IF(IDS.LT.0)  GØ TØ 1
              CALL GETADR(IPRM,IBUF)
        6      CØNTINUE
              CALL QIØ("2400,ILUN,1,,ISTAT,IPRM,ISR)
              IF(ISR.LT.0)  GØ TØ 2
               IF(RECØRD.EQ.-1)  GØ TØ 11
              IPRM(2)=3600
        10    CØNTINUE
              ISW=0
              RECØRD=RECØRD+1
              CALL QIØ("1000,ILUN,1,,ISTAT,IPRM,ISW)
```

```
0055                 IF(ISW.LT.0)  GØ TØ 3
0056                 CALL WAITFR(1,IDS)
0057                 IF(IDS.LT.0)  GØ TØ 4
          C           WRITE(6,101) (IB(I),I=1,2),ISTAT(2)
0058      101         FØRMAT(1H0,10X,2I4,3X,I5,2X,'BYTES TRANSFERED')
0059                  ERCØDE=IB(1)
0060                 IF(ERCØDE.LT.0)  GØ TØ 5
0061                 RETURN
0062      1          CØNTINUE
0063                 WRITE(6,100)  IDS
0064      100        FØRMAT(1H ,'  ASLUN CALL DSW=  ',I6)
0065                 STØP
0066      2          CØNTINUE
0067                 WRITE(6,200) ISR
0068      200        FØRMAT(1H ,'REWIND DSW = ',I6)
0069                 STØP
0070      3          CØNTINUE
0071                 WRITE(6,300)  ISW
0072      300        FØRMAT(1H ,'  READ QIØ DSW = ',I6)
0073                 STØP
0074      4          CØNTINUE
0075                 WRITE(6,400)  IDS
0076      400        FØRMAT(1H ,'  WAIT DSW = ',I6)
0077                 STØP
0078      5          CØNTINUE
0079                 IF(ERCØDE.EQ.-10)  FILE=FILE+1
0080                 IF(ERCØDE.EQ.-10)  RETURN
0081                  IF(ERCØDE.NE.-4)  GØ TØ 7
0082                  ERCNT=ERCNT+1
0083                  RETURN
0084      11          CØNTINUE
0085                  WRITE(6,700)
0086      700         FØRMAT(' ')
0087                  IF(ERCNT.EQ.0)  RETURN
0088                  WRITE(6,600)  ERCNT
0089      600         FØRMAT('1',10X,' TAPE ERRØRS ENCØUNTERED = ',I5)
0090                  RETURN
0091      7           CØNTINUE
0092                 WRITE(6,500)  ERCØDE
0093      500        FØRMAT(1H0,'  I/Ø STATUS BLØCK ERRØR CØDE= ',I6)
0094                 STØP
0095                 END
```

```
HFPRTRAN IV-PLUS V02-04                          10:13:52     07-ØCT-77
STDATA,FTN          /TR:BLOCKS/WR
                    SUBRØUTINE STDATA(IBUF,R,FILE,DFLG)
                    IMPLICIT INTEGER(A-Z)
                    LØGICAL*1 IBUF(1),PFN(132),PP(132),PN(132)
                    LØGICAL*1 PTL(132),PM(2112)
                    LØGICAL*1 BLANK
                    LØGICAL*1 PF(132)
                    CØMMØN/PCNT/LINE,PAGE
                    DATA BLANK/1H /
                    IF(DFLG.EQ.0)  GØ TØ 12
                    CALL STDMP(IBUF)
                    GØ TØ 13
      12            CØNTINUE
      13            CØNTINUE
                    NMAX=110
                    K=1
                    I=1
                    PI=11
                    DECØDE(3,99,IBUF(5))  ALSETS
                    IF(ALSETS.EQ.0)  RETURN
      99            FØRMAT(I3)
                    DECØDE(2,98,IBUF(10))  NCH
      98            FØRMAT(I2)
                    IF(IBUF(2).EQ.'S')  LINE=LINE+2
                    NLINS=59-LINE
                    LSAV=LINE
                    XL=NCH+6
                    IF(DFLG.EQ.1) NLINS=NLINS-7
                    LSETS=NLINS/XL
                    IF(LSETS.GT.0) WRITE(3,100)
          100       FØRMAT('0 ')
                    IF(LSETS.GT.0)  GØ TØ 8
                    LINE=66
                    CALL PNT
                    LSETS=60/XL
      8             CØNTINUE
                    IF(IBUF(2).EQ.'F')  GØ TØ 16
                    WRITE(3,97)
          97        FØRMAT(' ',49X,'STATISTICS REPØRT')
          16        CØNTINUE
                    ØSETS=5
                    PSETS=0
                    LFLD=11+9*NCH
                    IF(IBUF(2).EQ.'F')  LFLD=18+9*NCH
      20            J=1
                    RJ=12
                    IF(K.EQ.1)  GØ TØ 6
                    CALL CDRED(IBUF,R,FILE)
      4             DECØDE(2,98,IBUF(8))  SETSR
                    IF(SETSR.EQ.0)  RETURN
      1             CØNTINUE
                    IF(IBUF(2).EQ.'S')  GØ TØ 22
                    CALL FNAME(IBUF(RJ),PFN(PI))
      22            CALL PØP(IBUF(RJ),PP(PI),IBUF(2))
                    CALL SNAME(IBUF(RJ),PN(PI),IBUF(2))
                    CALL HDTTL(PTL(PI))
                    CALL MEAN(IBUF(RJ),PM(PI),NCH,IBUF(2))
```

```
              IF(K.EQ.ALSETS)  GØ TØ 10
              K=K+1
              IF(I.EQ.ØSETS)  GØ TØ 9
              I=I+1
              PI=PI+19
      2       IF(J.EQ.SETSR)  GØ TØ 20
              J=J+1
              RJ=RJ+LFLD
              GØ TØ 1
      9        K=K-1
     10       CØNTINUE
               PSETS=PSETS+1
               IF(PSETS.LE.LSETS)  GØ TØ 11
               PSETS=1
               IF(DFLG.EQ.1)  GØ TØ 14
               LINE=66
               CALL BNT
               LSETS=60/XL
               GØ TØ 11
     14        CØNTINUE
               LINE=66
               CALL KNT
     11        CØNTINUE
               WRITE(3,90)
     90        FØRMAT(1H0,' ')
              IF(IBUF(2).EQ.'S')  GØ TØ 3
              WRITE(3,101)  (PFN(N),N=1,NMAX)
    101       FØRMAT(1H ,131A1)
              WRITE(3,102)  (PP(N),N=10,NMAX)
    102       FØRMAT(1H ,'CHANNEL ',122A1)
              WRITE(3,101)  (PN(N),N=1,NMAX)
              GØ TØ 7
      3        CØNTINUE
              WRITE(3,101)  (PN(N),N=1,NMAX)
              WRITE(3,102)  (PP(N),N=10,NMAX)
      7       WRITE(3,103)  (PTL(N),N=8,NMAX)
    103       FØRMAT(1H ,'NUMBER ',122A1)
              IN=6
              IM=NMAX
              DØ 5  II=1,NCH
              WRITE(3,104)  (II,(PM(N),N=IN,IM))
    104       FØRMAT(1H ,2X,I2,128A1)
              IN=IN+132
              IM=IM+132
      5       CØNTINUE
      6       CØNTINUE
              DØ  30 N=1,132
              PF(N)=BLANK
               PFN(N)=BLANK
              PN(N)=BLANK
              PTL(N)=BLANK
              PP(N)=BLANK
              DØ  40  NN=1,NCH
              PM(N+(NN-1)*132)=BLANK
     40       CØNTINUE
     30       CØNTINUE
```

```
      IF(K.EQ.1)  GØ TØ 4
      IF(K.EQ.ALSETS)  GØ TØ 15
      K=K+1
      I=1
      PI=11
      GØ TØ 2
 15   CØNTINUE
      IF(LINE.EQ.0) LSAV=0
      LINE=PSETS*XL+LSAV
      CALL BNT
      RETURN
      END
```

```
FØRTRAN IV-PLUS V02-04                    10:14:09    07-ØCT-77       PA
FNAME.FTN          /TR:BLØCKS/WR
0001               SUBRØUTINE FNAME(FLD,PN)
0002               IMPLICIT INTEGER(A-Z)
0003               LØGICAL*1  FLD(1),PN(1),FIELD(6)
0004               DATA FIELD/1HF,1HI,1HE,1HL,1HD,1H /
0005               DØ 11  II=1,6
0006               PN(II)=FIELD(II)
0007               PN(II+6)=FLD(II)
0008       11      CØNTINUE
0009               RETURN
0010               END
```

```
FORTRAN IV-PLUS V02-04                       10:14:12    07-OCT-77          PA
MDTTL.FTN          /TR:BLOCKS/WR
0001               SUBROUTINE MDTTL(PTL)
0002               IMPLICIT INTEGER(A-Z)
0003               LOGICAL*1  MSD(12),PTL(1)
0004                DATA MSD/1HM,1HE,1HA,1HN,1H ,1H ,1HS,1HT,1H ,1HD,1HE,1HV/
0005               DO 1  I=1,13
0006        1      PTL(I)=MSD(I)
0007               RETURN
0008               END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
HFØRTRAN IV-PLUS V02-04                              10:14:15    07-ØCT-77         P
MEAN.FTN            /TR:BLØCKS/WR
                    SUBRØUTINE MEAN(FLD,PM,NCH,FLG)
                    IMPLICIT INTEGER(A-Z)
                    LØGICAL*1 FLD(1),PM(1),FLG(1),PT
                     LØGICAL*1 BLK
                    DATA PT/1H./
                     DATA BLK/1H /
                    ØFF=11
                    IF(FLG(1).EQ.'F')  ØFF=18
                    K=1
                     DØ  3  J=1,NCH
                    N=(J-1)*132
                     PM(N+1)=FLD(ØFF+1)
                     PM(N+2)=FLD(ØFF+2)
                     IF(PM(N+1).NE.'0')  GØ TØ 10
                     PM(N+1)=BLK
                     IF(PM(N+2).EQ.'0')  PM(N+2)=BLK
          10         PM(N+3)=FLD(ØFF+3)
                     PM(N+4)=PT
                     PM(N+5)=FLD(ØFF+4)
                     PM(N+6)=FLD(ØFF+5)
                     PM(N+7)=BLK
                     PM(N+8)=FLD(ØFF+6)
                     PM(N+9)=FLD(ØFF+7)
                     IF(PM(N+8).NE.'0')  GØ TØ 11
                     PM(N+8)=BLK
                     IF(PM(N+9).EQ.'0')  PM(N+9)=BLK
          11         PM(N+10)=FLD(ØFF+8)
                     PM(N+11)=PT
                     PM(N+12)=FLD(ØFF+9)
                     ØFF=ØFF+9
          3         CØNTINUE
                    RETURN
                     END
```

ORIGINAL PAGE IS OF POOR QUALITY

```
HFØRTRAN IV-PLUS V02-04                          10:14:21    07-OCT-77              PA
PØP.FTN               /TR:BLOCKS/WR
                      SUBRØUTINE PØP(FLD,PP,FLG)
                      IMPLICIT INTEGER(A-Z)
                      LØGICAL*1 FLD(1),PP(1),PAR(2),FLG(1)
                      DATA PAR/1H(,1H)/
                      ØFF=6
                      IF(FLG(1).EQ.'F')  ØFF=13
                      PP(3)=PAR(1)
                      DØ  1  I=1,5
                      PP(I+3)=FLD(I+ØFF)
              1       CØNTINUE
                      PP(9)=PAR(2)
                      RETURN
                      END
```

```
HFØRTRAN IV-PLUS V02-04                      10:14:25    07-OCT-77            PAG
SNAME.FTN          /TR:BLOCKS/WR
                   SUBRØUTINE SNAME(FLD,PN,FLG)
                   IMPLICIT INTEGER(A-Z)
                   LØGICAL*1  FLD(1),PN(1),SUBCL(6),FLG(1)
                   DATA SUBCL/1HS,1HU,1HB,1HC,1HL,1H /
                   ØFF=0
                   IF(FLG(1).EQ.'F')  ØFF=7
                   DØ 11  II=1,6
                   PN(II)=SUBCL(II)
                   PN(II+6)=FLD(II+ØFF)
         11        CØNTINUE
                   RETURN
                   END
```

```
HFØRTRAN IV-PLUS V02-04                        10:14:28    07-ØCT-77              PA
STDMP.FTN              /TRIBLØCKS/WR
                       SUBRØUTINE STDMP(IBUF)
                       IMPLICIT INTEGER(A-Z)
                       LØGICAL*1 IBUF(1)
                       CØMMØN /PCNT/LINE,PAGE
                       LINE=66
                       CALL KNT
                       IF(IBUF(2).EQ.'F')  GØ TØ 8
                       WRITE(6,200)
          200          FØRMAT(5X,'SUBCLASS STATISTICS RECØRD')
                       GØ TØ 9
          8            WRITE(6,201)
          201          FØRMAT(5X,'FIELD STATISTICS RECØRD')
          9            CØNTINUE
                       WRITE(6,202)  (IBUF(N),N=3,4)
          202          FØRMAT(5X,'RECØRD SEQ. NØ.   ',2A1)
                       WRITE(6,203)  (IBUF(N),N=5,7)
          203          FØRMAT(5X,'NØ. SUBCLASSES   '3A1)
                       WRITE(6,204)  (IBUF(N),N=8,9)
          204          FØRMAT(5X,'NØ. SUBCLASS SETS ',2A1)
                       WRITE(6,205)  (IBUF(N),N=10,11)
          205          FØRMAT(5X,'NØ. ØF CHANNELS   ',2A1)
                       LINE=5
                       RETURN
                       END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
              SUBRØUTINE CAMHDG(IBUF,TAPNØ)
              IMPLICIT INTEGER(A-Z)
              LØGICAL*1 IBUF(1),SEGNØ(4)
              LØGICAL*1 PB(132),BLANK
              CØMMØN/SEG/SEGNØ
              DATA BLANK/1H /
              DØ  1  I=1,4
     1        SEGNØ(I)=IBUF(147+I)
              WRITE(6,400)
     400      FØRMAT('1',49X,'CAMS INTERFACE REPØRT')
              WRITE(6,401)  (IBUF(I),I=3,26),TAPNØ
     401      FØRMAT(1H0,3X,'DPAR NØ.   = ',24A1,57X,'TAPE NUMBER: ',I5,'(
              WRITE(6,402)  (IBUF(I),I=709,714),(IBUF(J),J=715,720)
     402      FØRMAT(1H0,3X,'RECØRD ID = ',6A1,1X,6A1)
              WRITE(6,403)
     403      FØRMAT(1H0,50X,'ACQUISITIØN DATES')
              WRITE(6,404)
     404      FØRMAT(1H ,47X,'1',6X,'2',6X,'3',6X,'4',15X,
           *'SEGMENT TYPE')
              DØ  11  N=1,132
     11       PB(N)=BLANK
              RJ=148
              CALL MV(IBUF(RJ),PB(31),4)
              IP=47
              PB(91)=IBUF(RJ+4)
              RI=RJ+6
              DØ  9  J=1,4
              IF(IBUF(RI).EQ.'0')  GØ TØ 10
              CALL MV(IBUF(RI),PB(IP),5)
              RI=RI+5
              IP=IP+7
     9        CØNTINUE
     10       WRITE(6,301)  (PB(K),K=30,95)
     301      FØRMAT(1H ,4X,'****** SEGMENT NUMBER =',66A1)
              RJ=174
              DØ  2  I=1,3
              DØ  12  N=1,132
     12       PB(N)=BLANK
              IF(IBUF(RJ).EQ.'0')  GØ TØ 4
              CALL MV(IBUF(RJ),PB(31),4)
              PB(91)=IBUF(RJ+4)
              IP=47
              RI=RJ+5
              DØ  3  J=1,4
              IF(IBUF(RI).EQ.'0')  GØ TØ 4
              CALL MV(IBUF(RI),PB(IP),5)
              RI=RI+5
              IP=IP+7
     3        CØNTINUE
              WRITE(6,300)  I,(PB(K),K=30,95)
     300      FØRMAT(1H ,'TRAINING SEGMENT NØ. ',I1,' =',66A1)
              RJ=RJ+25
     2        CØNTINUE
     4        CØNTINUE
              RETURN
              END
```

```
HFØRTRAN IV-PLUS V02-04                          10:14:45    07-ØCT-77          PA
CPIPØ.FTN             /TR:BLØCKS/WR
                      SUBRØUTINE CPIPØ(FLD,CLASS,PI,PØ)
                      IMPLICIT INTEGER(A-Z)
                      LØGICAL*1  FLD(1),CLASS(1)
                      DØ 2  I=1,4
                      CLASS(I)=FLD(I)
              2       CØNTINUE
                      DECØDE(5,100,FLD(23))  PI
            100       FØRMAT(I5)
                      DECØDE(5,100,FLD(28))  PØ
                      RETURN
                      END
```

```
HFØRTRAN IV-PLUS V02-04                          10:14:49    07-ØCT-77          PA
MV.FTN                 /TR:BLOCKS/WR
0001                   SUBROUTINE MV(FLD,PB,NC)
0002                   IMPLICIT INTEGER(A-Z)
0003                   LOGICAL*1 FLD(1),PB(1)
0004                   DO  8  J=1,NC
0005        8          PB(J)=FLD(J)
0006                   RETURN
0007                   END
```

```
CAMRPT=BIASRP,CAMREC,CDRFD,DØTRAY,SEGEXT
CLURES,STDATA,FNAME,KNT,MDTTL,MEAN,PØP,SNAME
STDMP
BNT
CAMHDG,CPIPØ,MV
[1,1]F4PØTS/LB
/
UNITS=9
ASG=LP16
//
```