JSC-12560

8.0-10204

NASA CR-160619

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

Job Order 71-983

(TIRF 76-0053)

(E80-10204) AS-BUILT DESIGN SPECIFICATION OF THE CAMS/CAS INTERFACE TAPE REPORT GENERATION PROGRAM (Lockheed Electronics Co.) 69 p HC A04/MF A01 CSCL 05B

N80-29783

Unclas

G3/43 00204

Prepared By

Lockheed Electronics Company, Inc.

Systems & 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

January 1977

LEC-9882

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

Job Order 71-983

(TIRF 76-0053)

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

February 19 7

LEC-9882

CONTENTS

1. SCOPE

1.1 GENERAL

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

2. APPLICABLE DOCUMENTS

- TIRF 76-0053
- 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 (PHO-TR543, Rev. A, Change 2)

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 (CAMRPT)

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

3.2.1.1 Linkages

CAMRPT calls subroutines CAMPER, CAPRNT, FLDPER, REDHED, and STDATA.

3.2.1.2 Interfaces

N/A

3.2.1.3 Inputs

CAMRPT control cards are: SEGMENT XXXX, RECORD ID XXXXXX XXXXXX, PRINT TAPE, ALL SEGMENTS, END. CAM/CAS interface tape records are inputs to CAMRPT. 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 2094 bytes.

3.2.1.6 Description

Upon reading a control card CAMRPT tests the first non-blank character for one of the following: S, R, P, A, or E. If the card is blank or if the character is not one of the above, the program prints an error message on the line printer and stops. The action taken for each control card is as follows:

S - The program obtains the segment number from the input card. Records from the CAMS/CAS interface tape are read until a recognition segment record is found whose segment number matches the card input number, or, if not found, an error message is printed out. After the segment is located, CAMPER is called to generate the classification summary. STDATA is called to output the statistics report, and then FLDPER to output the accuracy report.

R - The action taken for this control card is the same as in the S case above, except that the record ID number is used in place of the segment number.

P - Beginning with the first record, the program reads the input tape, calls CAPRNT to print the contents of the record, until an end of file indicating the end of data is reached.

A - For each segment on the input tape, the program calls in succession CAMPER, STDATA and FLDPER.

E - The program rewinds the input tape and stops.

With the exception of the END option, the program returns to read another control card after completing an option.

3.2.1.7 Flowcharts

N/A

3.2.1.8 Listing

See Appendix B.

3.2.2 SOFTWARE COMPONENT NO. 2 (CAMPER)

This program processes classification results contained in recognition segment records, computes category and class percentages, and outputs a classification summary report.

3.2.2.1 Linkages

CAMPER is called by CAMRPT and calls subroutines CAMHDG. CPIPO, MV, and REDHED.

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. See Appendix A.

3.2.2.5 Storage Requirements

Total space allocated is 1471 bytes.

3.2.2.6 Description

CAMPER is called with the first recognition segment record for the segment to be processed residing in array IBUF. CAMPER 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 CAMPER. 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.

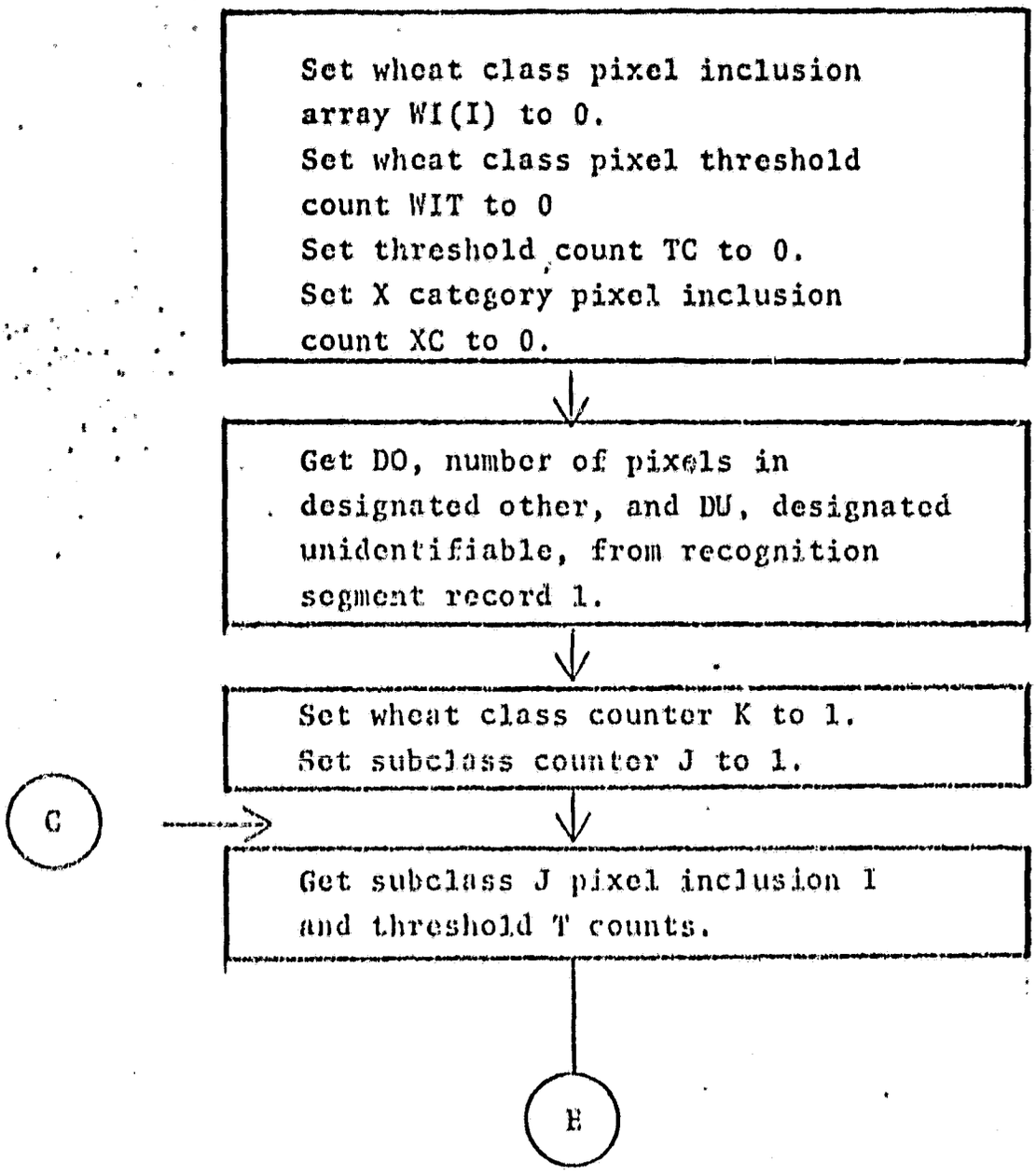
In processing for the second, and subsequent subclasses, the program calls CPIPO to get the next class name, checks to see if it is wheat, and, if so, compares it to the last class name in CLIST. If it is not the same, the new name is saved in CLIST and the class index is incremented by 1. This causes wheat class pixel count PI to be tallied in the next results array location. CAMPER output is in terms of percentages calculated according to equations in the requirements document LEC-9151.

3.2.2.7 Flowcharts

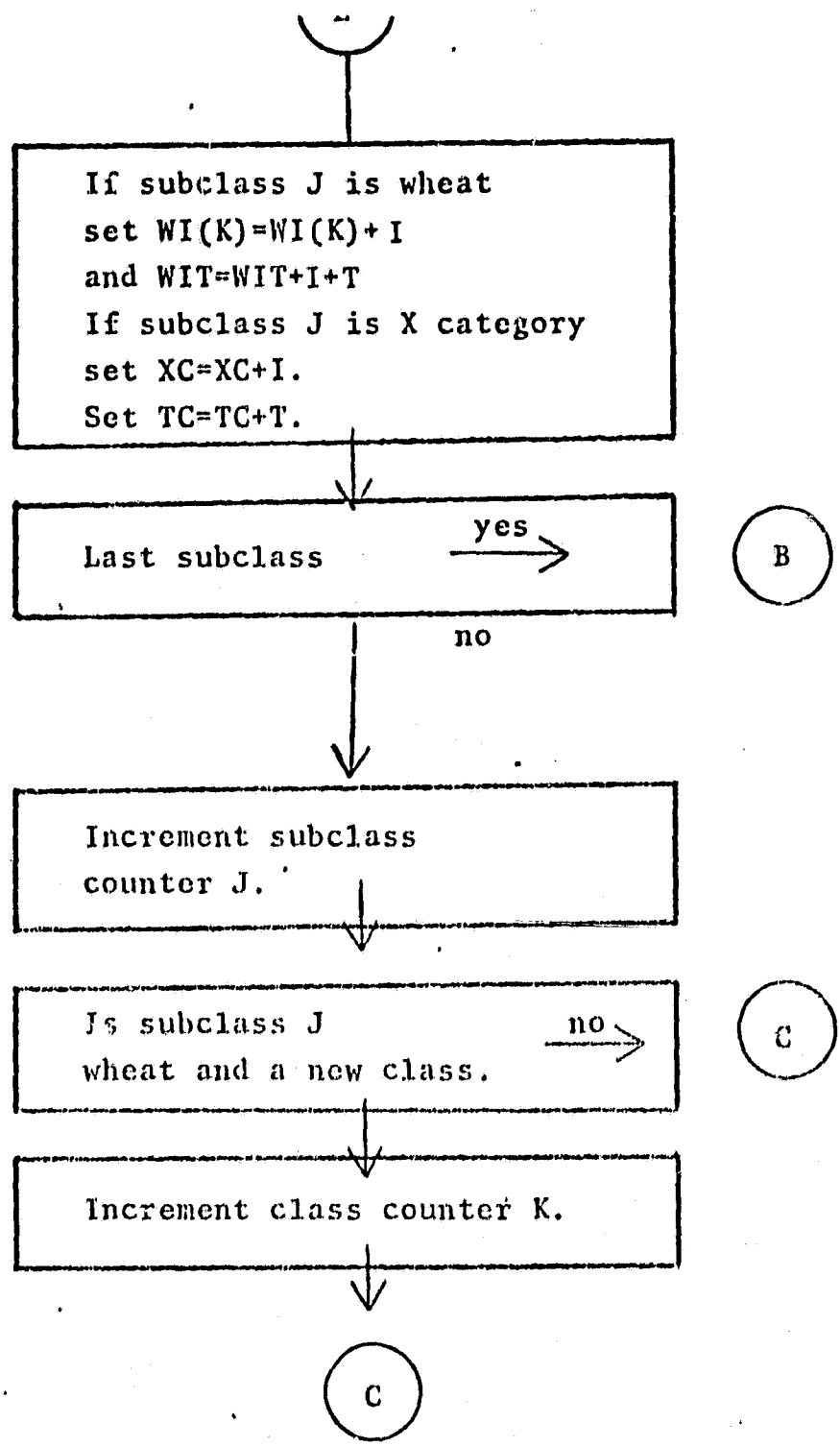
See Flow Diagram 1.

3.2.2.8 Listing

See Appendix B.

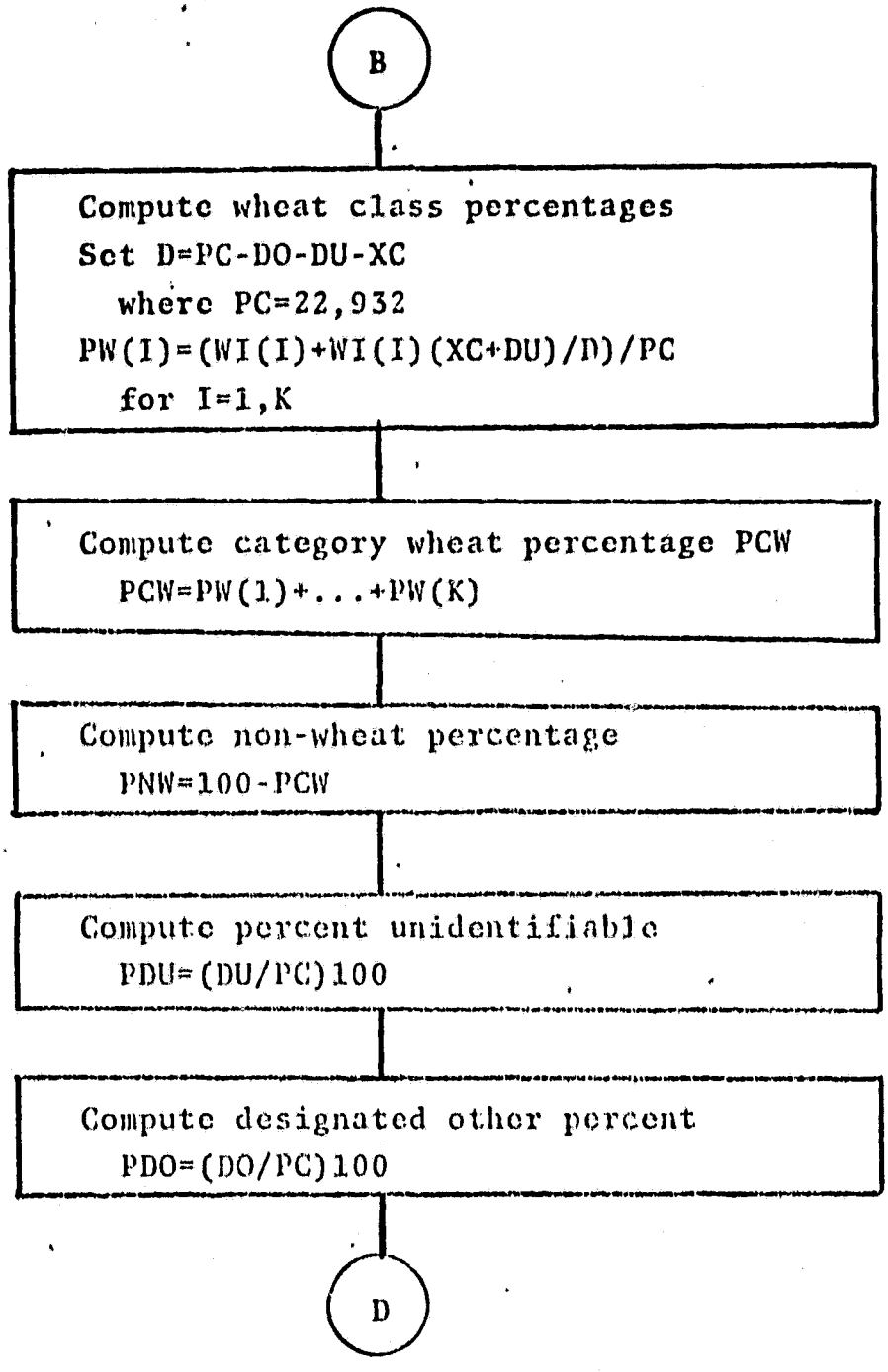


Flow Diagram 1 (Page 1 of 4)



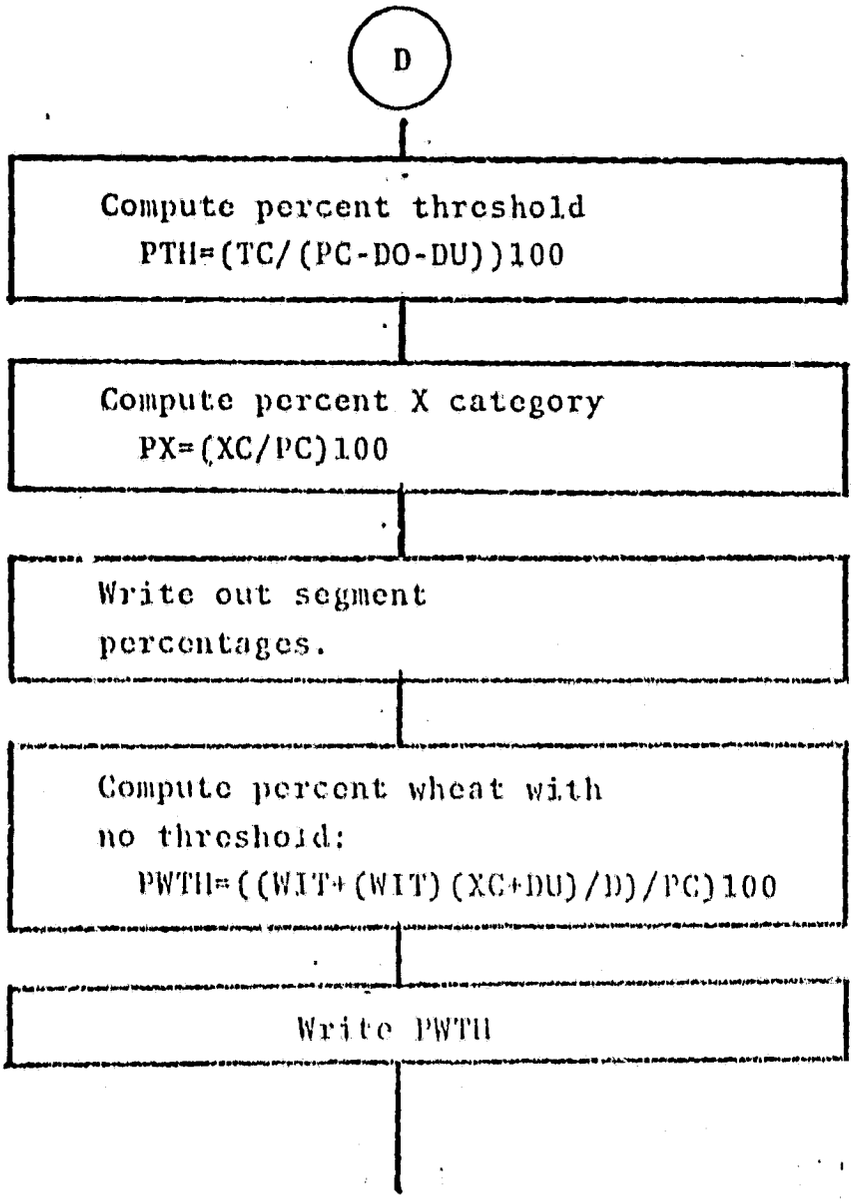
Flow Diagram 1 (Page 2 of 4)

3 of 9



Flow Diagram 1 (Page 3 of 4)

3-8
10



Flow Diagram 1 (Page 4 of 4)

3.2.3 SOFTWARE COMPONENT NO. 3 (STDATA)

This subroutine formats and outputs field and subclass statistics data.

3.2.3.1 Linkages

STDATA is called by CAMRPT and by CAPRNT. STDATA calls subroutines KNT, MDTTL, MEAN, POP, REDHED, FNAME, FNAME, and STDMP.

3.2.3.2 Interfaces

N/A

3.2.3.3 Inputs

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

3.2.3.4 Outputs

The statistics report. See Appendix A.

3.2.3.5 Storage Requirements

Space allocated, including subroutines, is 2811 bytes.

3.2.3.6 Description

STDATA is called from CAMRPT or from CAPRNT. The argument DFLG is set to indicate the calling program. If it is CAPRNT, STDATA calls STDMP to output the CAMS/CAS tape printout heading and to print initial data from the statistics record. If called by CAMRPT, STDATA prints a statistics report title. 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 calss REDHED to read the next statistics record from tape.

3.2.3.7 Flowcharts

N/A

3.2.3.8 Linting

See Appendix B.

3.2.4 SOFTWARE COMPONENT NO. 4 (FLDPER)

FLDPER generates an accuracy report from field data record classification results.

3.2.4.1 Linkages

FLDPR is called by CAMRPT, and calls FPIPO and REDHED.

3.2.4.2 Interfaces

N/A

3.2.4.3 Inputs

The field data record, containing the field name, the field population, the associated subclass name of the field, the number of subclasses, and subclass related classification results.

3.2.4.4 Outputs

The accuracy report for requests. See Appendix A.

3.2.4.5 Storage Requirements

Space allocated to FLDPER is 1783 bytes.

3.2.4.6 Description

FLDPER is called after the first field data record for the current segment has been read into array IBUF. The program converts the number of field data record subclasses, NOSUB, from character code format in IBUF to integer format. The class portion of the field associated subclass name is saved in FASC. FLDPER initializes parameters and arrays for accumulating classification results on the three classification levels, category, class, and field. To begin subclass processing, the program sets the location in the array IBUF

of the first subfield containing subclass related classification results. Subfield contents are accessed by calling subroutine FPIPO, which returns the class portion of the subclass name, and PI, the subclass inclusion pixel count. The first character of the class name, which specifies the category, is tested for categories W, N, and X. If a match is found, the corresponding category pixel inclusion count on the field level is increased by PI. The first subfield class name is saved in CLIST and the pixel count for that class is set to PI. In processing for the next subfield, after calling FPIPO, the class name is compared to the last class name in CLIST. If it is a new one, it is saved in CLIST and the class counter is increased by 1. This causes the pixel count PI to be added into the next array location for summing results. After NOSUB subclass results have been processed, the field level classification percentages are computed for the categories W, N. X and for all classes in CLIST. The percentages are output along with the field name and field population. On the class level, where class now refers to the field associated class name in FASC, the just compiled field totals are added to class level totals. This completes processing of the current field record. A new field data record is read into IBUF. The class portion of its field associated subclass name is compared to the name for the previous field in FASC. If it is the same class, the program goes to initialize field level variables and to process the current field record as before.

If a new class is indicated, then class level percentages are computed and output with the FASC class name. Next the first character of the new class name, and the old class name in FASC, are compared to see if the category has also changed. If so, then category level percentages are computed

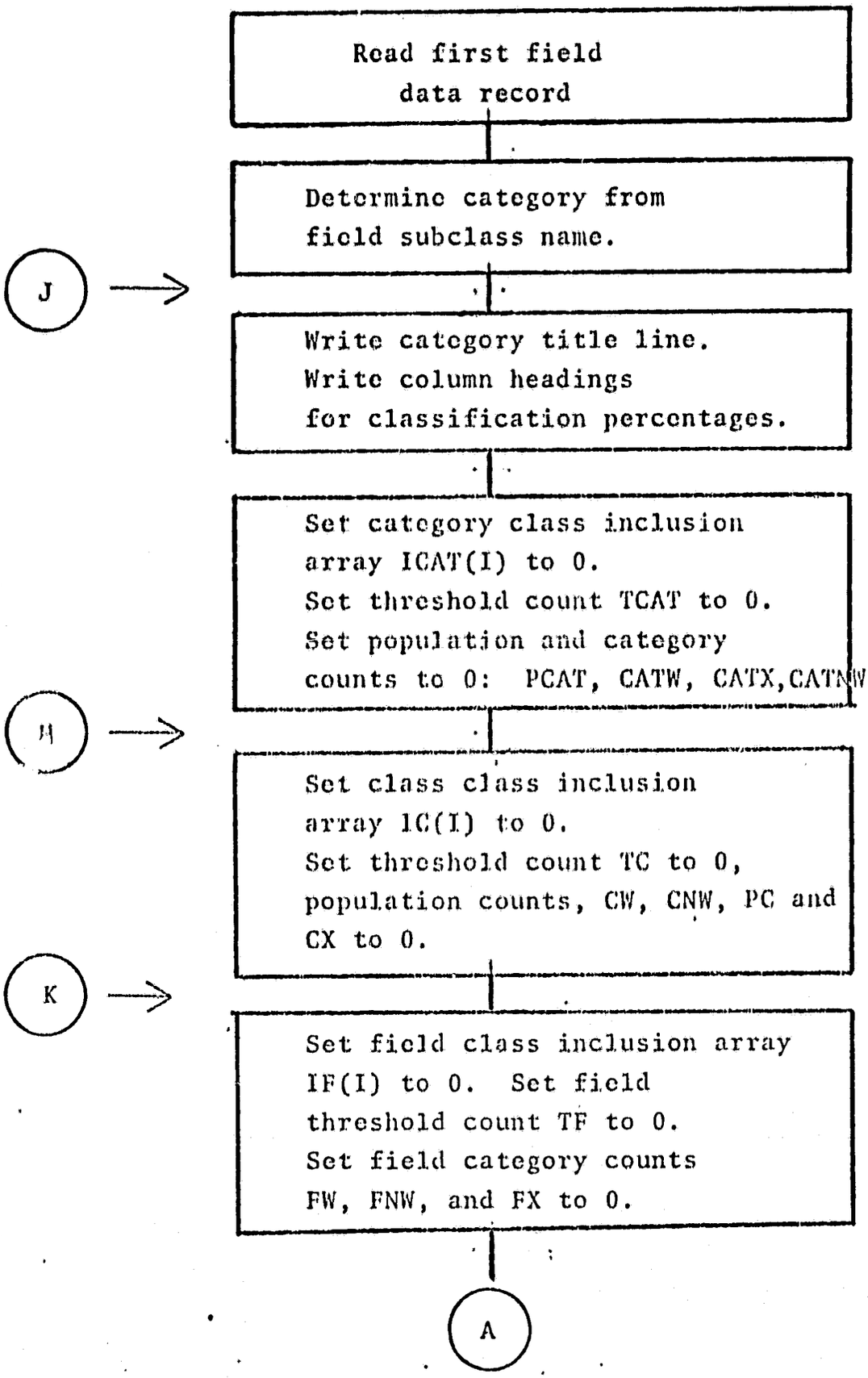
and output. In any case, the new class level name is saved in FASC, and a return is made to initialize either category or class level variables, as required, before processing the new field data record.

3.2.4.7 Flowcharts

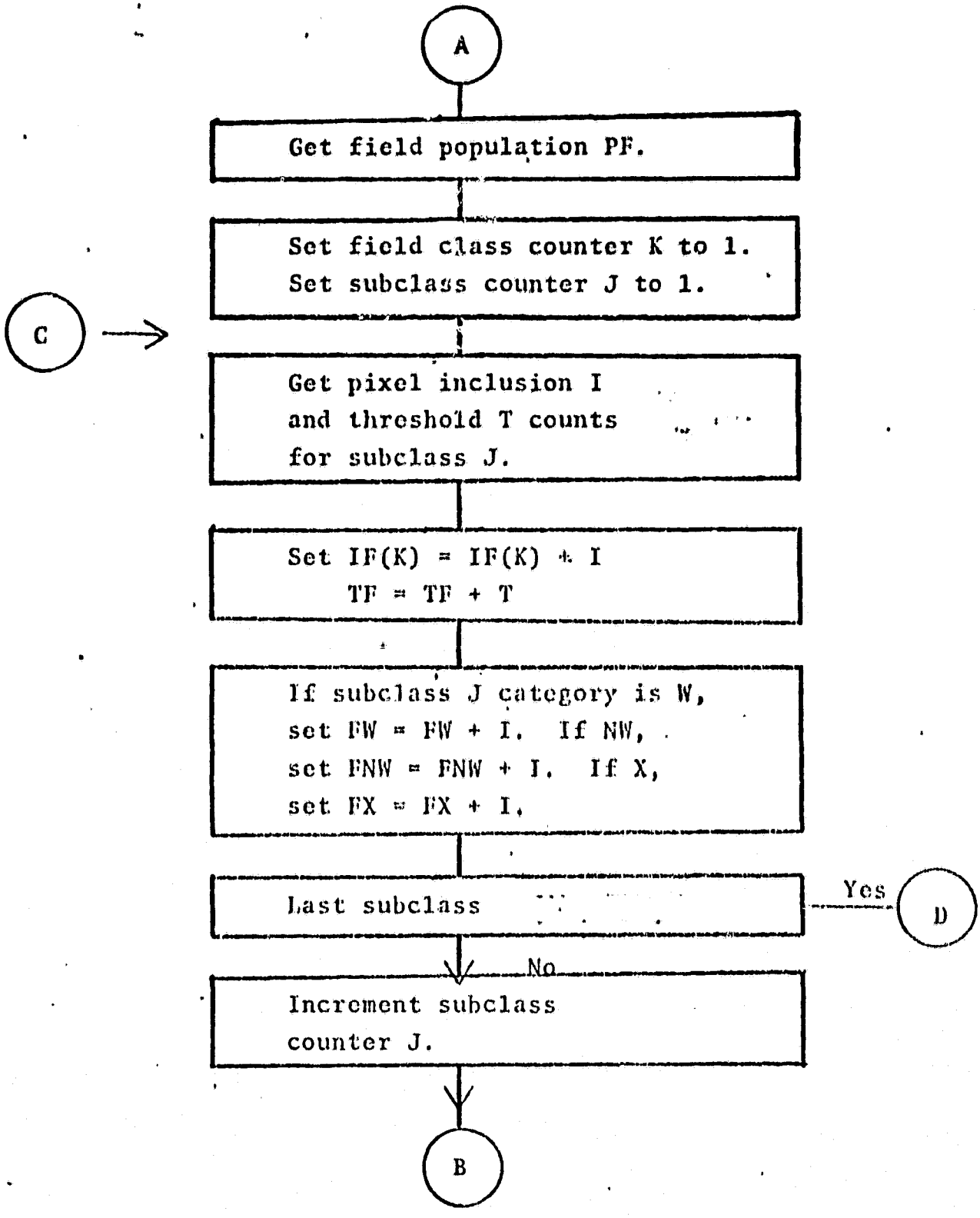
See Flow Diagram 2.

3.2.4.8 Listing

See Appendix B.



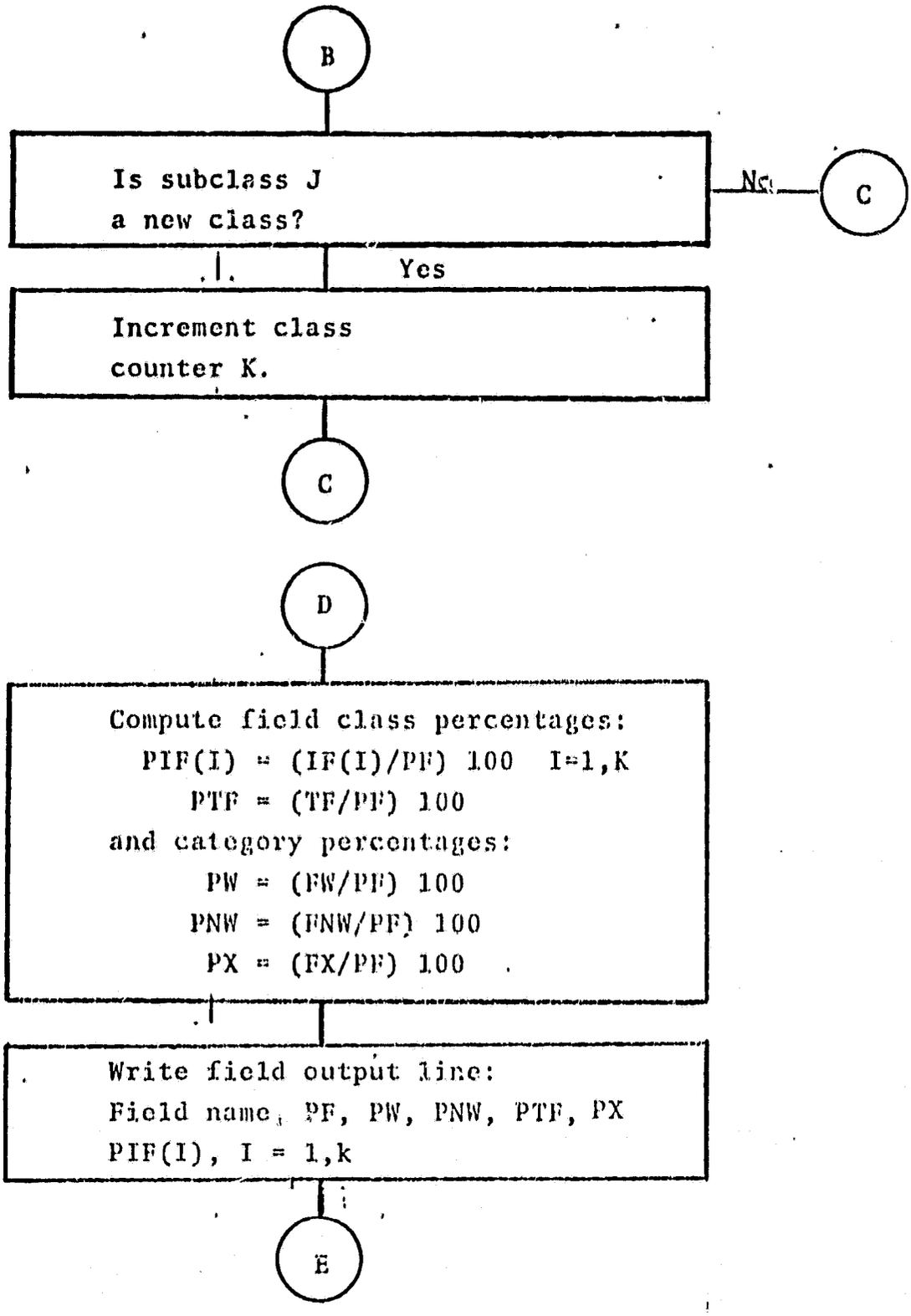
Flow Diagram 2 (Page 1 of 6)



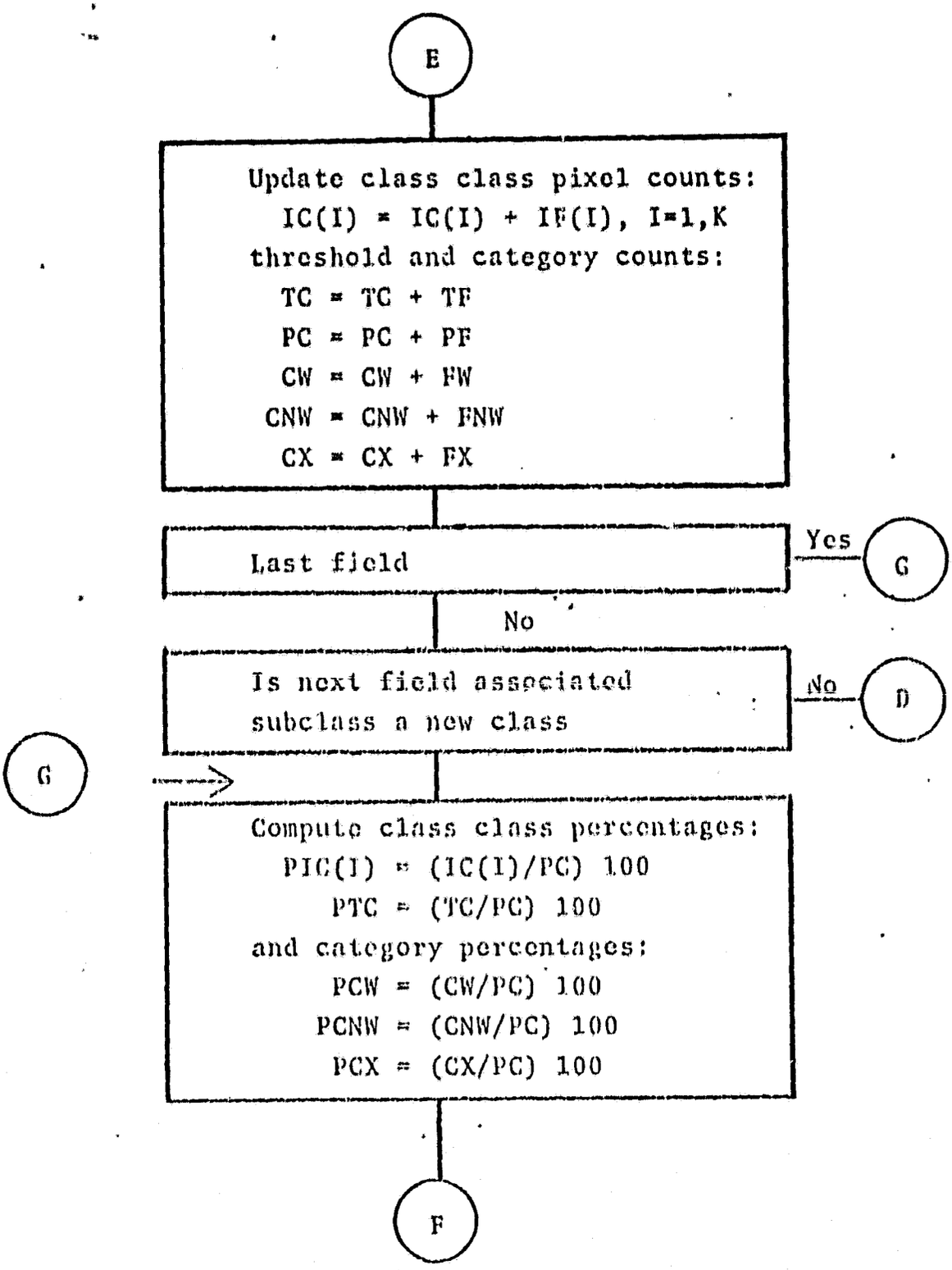
Flow Diagram 2 (Page 2 of 6)

3-16

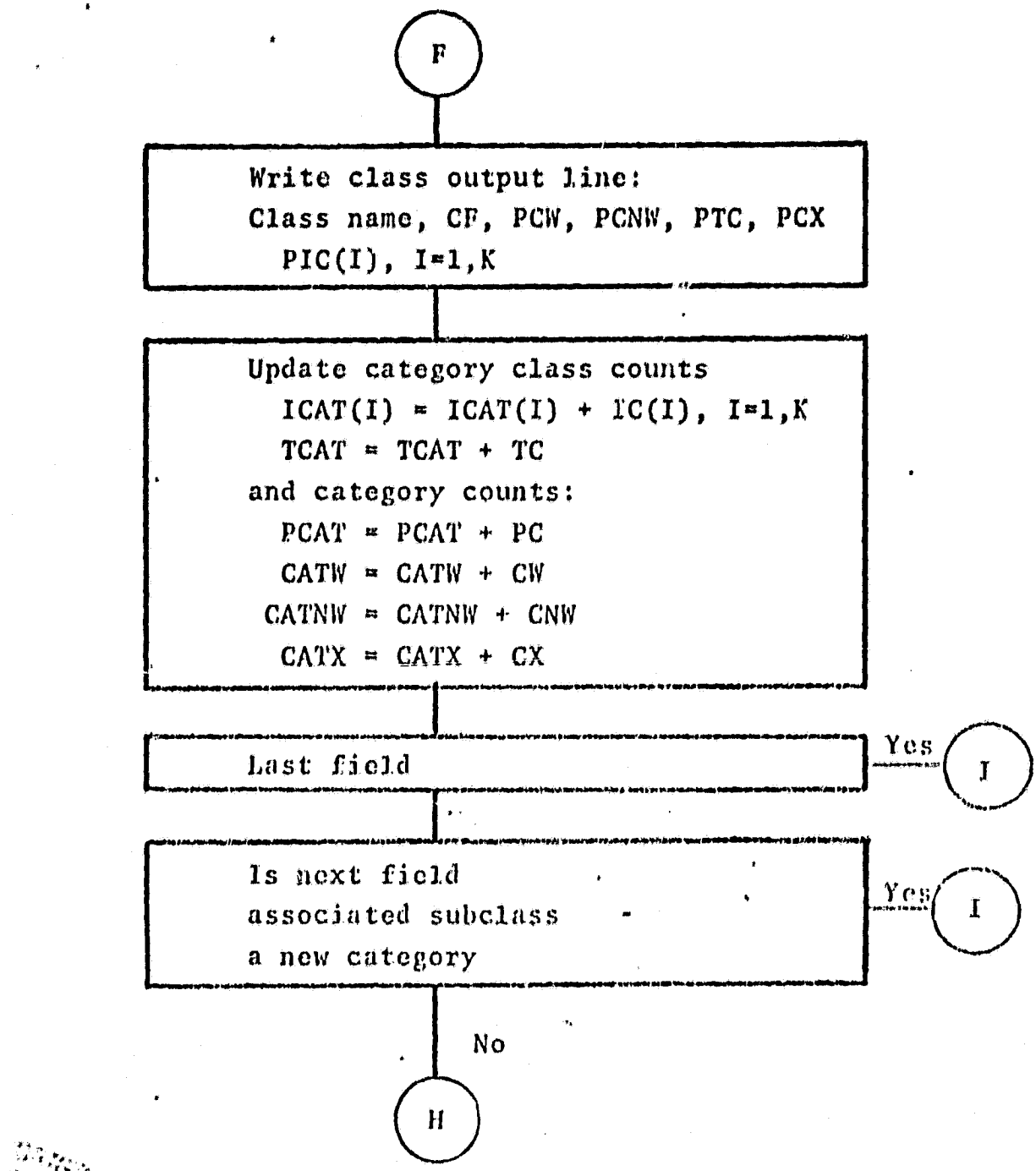
18



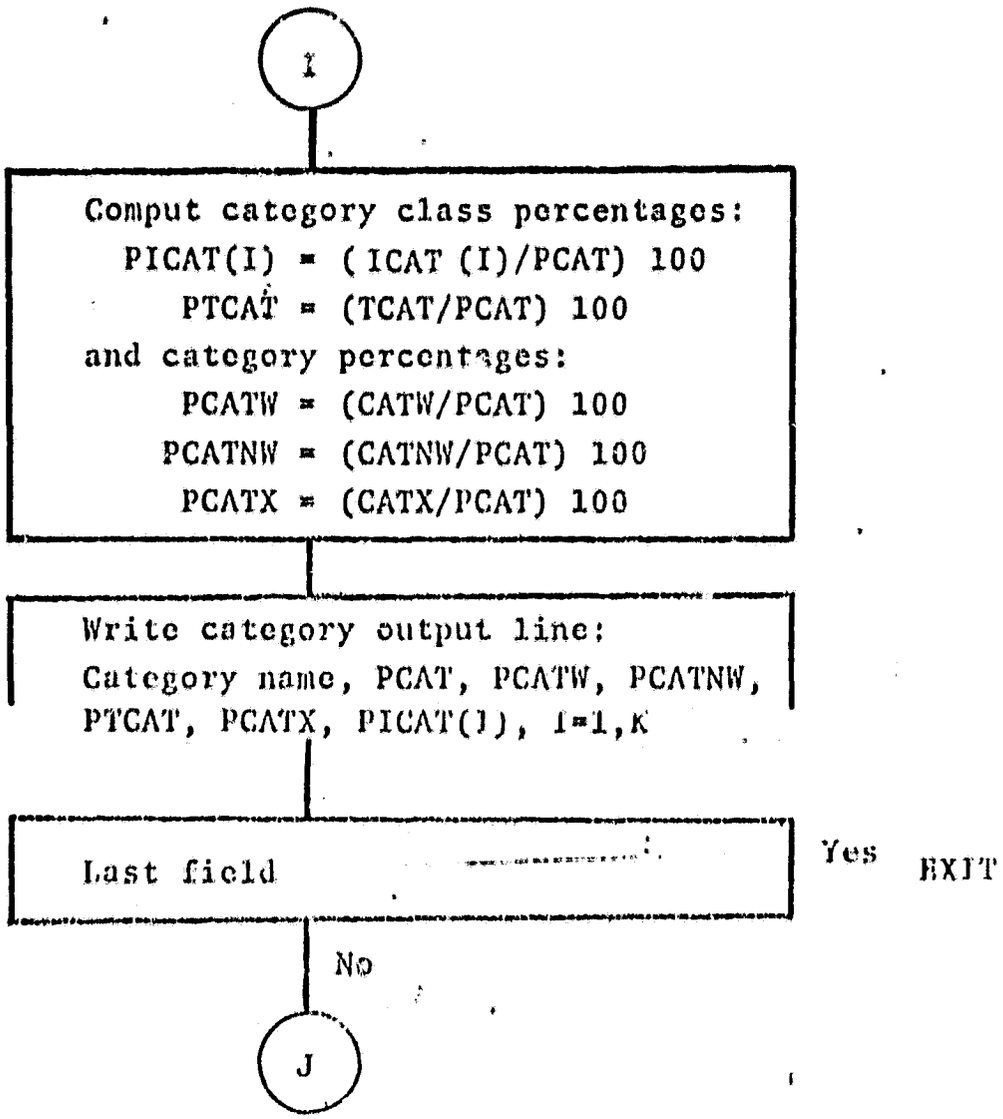
Flow Diagram 2 (Page 3 of 6)



Flow Diagram 2 (Page 4 of 6)



Flow Diagram 2 (Page 5 of 6)



Flow Diagram 2 (Page 6 of 6)

3.2.5 SOFTWARE COMPONENT NO. 5 (CAPRNT)

This program prints out the contents, record by record, of the CAMS/CAS interface tape.

3.2.5.1 Linkages

CAPRNT is called by CAMRPT and calls subroutines KNT, STDATA, and REDHED.

3.2.5.2 Interfaces

N/A

3.2.5.3 Inputs

CAMS/CAS interface tape records.

3.2.5.4 Outputs

The CAMS/CAS interface tape printout.

3.2.5.5 Storage Requirements

Total space allocated is 2204 bytes.

3.2.5.6 Description

CAPRNT is called by CAMRPT after an initializing call to tape routine REDHED has been made and the first input tape record read into array IBUF. The record type is determined from the first byte of the record. The program transfers to a portion of the program which prints out the contents of that record type. A return is made to main program CAMRPT to read in the next record. The program calls STDATA to format and output statistics data. Subroutine KNT is called to print a page title and page number.

3.2.5.7 Flowcharts

N/A

3.2.5.8 Listing

See Appendix B.

4. OPERATING PROCEDURE

4.1 GENERAL

This procedure illustrates how a Bldg. 30 provided CAMS/CAS Interface tape is processed so that the required reports, within user desired options, are being produced. The test is accomplished by mounting a Bldg. 30 tape and executing the process program through various parameter cards and keyboard controls. The control cards available are:

PRINT TAPE (optional)
ALL SEGMENTS (optional)
SEGMENT XXXX (optional)
RECORD ID XXXXXX XXXXXX (optional)
END (mandatory)

All optional commands can be used independent of each other or collectively, in any order.

4.2 TEST PROCEDURE

a. Have the operator set the line printer to non-spool.

b. Log on to the system.
MCR>HEL [50,50]

c. Mount CAMS/CAS Interface tape #A20203 on unit MTO and enter mount message.
MCR>MOU MTØ:/CHA=[FOR] (CR)

d. Load card reader with the following reports and program control cards and start card reader.
PRINT TAPE
SEGMENT 9691
SEGMENT 9679
RECORD ID 020414 B D001
END

e. Execute CAMS/CAS Interface Tape Report Generation program by keyboard input of:

```
MCR>RUN:CAMRPT$
```

f. Remove tape #A20203 from unit MTO and mount tape #A20204.

g. Load card reader with following report and program control cards and start card reader.

```
ALL SEGMENTS
END
```

h. Execute CAMS/CAS Tape Report Generation program by keyboard input of:

```
MCR>RUN:CAMRPT$
```

i. Unload mounted tape, remove cards from card reader and collect reports from the system line printer.

APPENDIX A

CAMS REPORTS CONTENTS

CAMS INTERFACE REPORT

DPAE NO. = AAAAAAAAAAAAAAAAAAAAAAAA
RECORD ID = AAAAAA AAAAAA

	ACQUISITION DATES 1	2	3	4	SEGMENT TYPE
RECOGNITION SEG. NUMBER = NNNN	YYDDD	YYDDD	YYDDD	YYDDD	A
TRAINING SEGMENT NO. 1 = NNNN	YYDDD	YYDDD	YYDDD	YYDDD	A
TRAINING SEGMENT NO. 2 = NNNN	YYDDD	YYDDD	YYDDD	YYDDD	A
TRAINING SEGMENT NO. N = NNNN	YYDDD	YYDDD	YYDDD	YYDDD	A

SUBCLASS STATISTICS REPORT

CHANNEL NUMBER	FIELD AAAAAA (NNNNN) SUBCL AAAAAA MEAN	ST. DEV	FIELD AAAAAA (NNNNN) SUBCL AAAAAA MEAN	ST. DEV	SUBCL AAAAAA MEAN	ST. DEV	SUBCL AAAAAA MEAN	ST. DEV	SUBCL AAAAAA MEAN	ST. DEV
1	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN
.	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN
.	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN
16	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN

SUBCLASS MANIPULATION STATISTICS REPORT

CHANNEL NUMBER	SUBCL AAAAAA (NNNNNN) MEAN	ST. DEV	SUBCL AAAAAA (NNNNNN) MEAN	ST. DEV	SUBCL AAAAAA (NNNNNN) MEAN	ST. DEV	SUBCL AAAAAA (NNNNNN) MEAN	ST. DEV	SUBCL AAAAAA (NNNNNN) MEAN	ST. DEV
1	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN
.	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN
.	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN
16	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN	NNN.NN	NN.NN

FEATURE SELECTION OPTION EXERCISED:

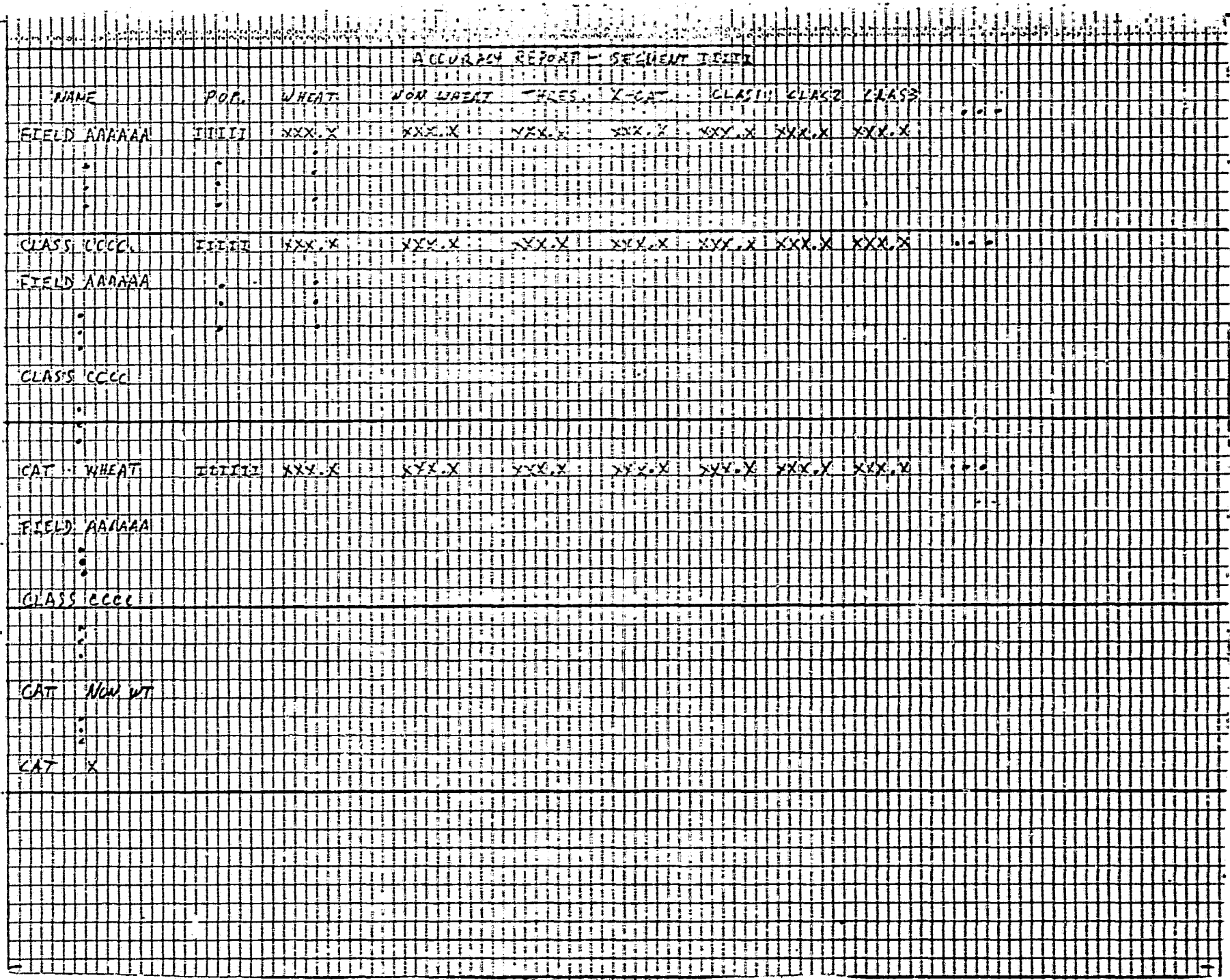
CLASSIFICATION SUMMARY REPORT

SUBCLASS NAME	THRESHOLD VALUE	NORMALIZED A PRIORI VALUE	CHANNELS USED
AAAAAA		NNN	1
		NNN	.
		NNN	.
			.
			16

SEGMENT PERCENTAGES
WHEAT CATEGORY = NNN.N %

NON-WHEAT CATEGORY = NNN.N %
DESIGNATED OTHER = NNN.N %
THRESHOLD = NNN.N %
DESIGNATED DIVIDENT = NNN.N %
X CATEGORY = NNN.N %

WHEAT CLASS 1 = NNN.N %
WHEAT CLASS 2 = NNN.N %



A-39

13. COMMENTS	CAMS / CAS INTERFACE TAPE PRINTOUT - PAGE 11
	RECORD 1:
	VOL
	1
Volume Serial No.	AAAAAA
1 OR BLANK	1
	1
	RECORD 2:
	HDR
	1
FILE ID	AAA ... A
	1
DATE	YYDDD
	1
	RECORD 3:
	HDR
	2
	F
BLOCK LENGTH	1111
RECORD LENGTH	1111
BUFFER OFFSET	11

11. COMMENTS	CARS / CES INTERFACE TAPE PRINTOUT PAGE 2
	RECOGNITION SEGMENT
	RECORD 2
SEQ. NO.	1
DPAR NO	AAA . . . A
RUN ID	AAA . . . A
Bh. No	XX.XXXXXX
TAPE NO	AAAAAAA
NO. SUB. CL.	IIII
NO. MERGED ELMS	AAAA
PIXELS D/O	IIIII
PIXELS D/U	IIIII
DATE	YYDDD
	I
	IIII IIII IIII IIII
Bh No.	XX XXXX
BIO. WINDOWS	YYDDD

16 A4

13. COMMENTS

LAMS / LAS INTERFACE TAPE PRINTOUT PAGE 3

SEG. NO. AAAAA

S OR W A

T OR blank A

DATES A YYDDD

SEG. NO AAAAA

T A.P. P. INTO P. OUT

SUB. NAME AAAAAA I AAAAAAAA XX.XX AAA I:II:I I:I:I

" "

" "

(SEQ. NO. 2,3,4 if present)

RECORD n

AAAAAA = AAAAAAAAA XX.XX I:I:I:I I:I:I:I

13. COMMENTS

```
                                              CAMS / CAS INTERFACE TAPE PRINTOUT                PAGE N

                        FIELD DATA

                        RECORD  1

Field Name              AAAAAA

Subclass Name           AAAAAA

Type                    A

Population              AAAAA

No. Subclasses          AA
                                      S.A.        M.E.        P. INTO      P. OUT
Sub. Name               AAAAAA          A           A          IIIII        IIIII
   .                       .            .           .
   .                       .            .           .

(SEQ. No. 1             RECORD  2     S.A.        M.E.        P. INTO      P. OUT
 if present)            AAAAAA          A           A          IIIII        IIIII
                           .            .           .
                           .            .           .

                        RECORD  N
```

APPENDIX B

PROGRAM LISTINGS

```
              IMPLICIT INTEGER(A-Z)
              LOGICAL*1 IBUF(3060)
              LOGICAL*1 CCHAR
              LOGICAL*1 CD(80)
              LOGICAL*1 CONTR(5)
              LOGICAL*1 SEGNO(4),RECORD(12)
              COMMON/SEG/SEGNO
              DATA CONTR/1HS,1HR,1HP,1HA,1HE/
      15      CONTINUE
      C       WRITE(6,100)
      100     FORMAT(10X,'**********?????$#$#+##')
              READ(5,99)  CD
      99      FORMAT(80A1)
              DO 20 I=1,72
              IF(CD(I).NE.' ').  GO.TO.21
      20      CONTINUE
              GO TO 22
      21      CCHAR=CD(I)
              DO  1  K=1,5
              IF(CONTR(K).EQ.CCHAR)  GO TO 2
      1       CONTINUE
      22      CONTINUE
              WRITE(6,88)
      88      FORMAT(10X,'  BAD DATA CARD ')
              STOP
      2       GO TO(31,31,33,34,35)K
      33      CONTINUE
              FILE=0
              R=0
      331     CONTINUE
              CALL REDHED(IBUF,R,FILE)
              R=R+1
              IF(FILE.EQ.3)  GO TO 15
              CALL CAPRNT(IBUF,R,FILE)
              GO TO 331
      35      CONTINUE
              R=0
              CALL REDHED(IBUF,R,FILE)
              STOP
      31      CONTINUE
              F=0
              I=1
      4       IF(CD(I).NE.' ')  GO TO 3
              I=I+1
              IF(I.GT.72)  GO TO 9
              GO TO 4
      3       F=F+1
              IF(F.EQ.1)  GO TO 7
              IF(K.EQ.1)  GO TO 8
              IF(F.EQ.2)  GO TO 7
              GO TO 8
      7       I=I+1
              IF(CD(I).EQ.' ')  GO TO 4
              IF(I.GT.72)  GO TO 9
              GO TO 7
      C       WRITE(6,102)  (CD(J),J=I,I+3)
```

ORIGINAL PAGE IS OF POOR QUALITY

```
0055     8      CONTINUE
0056            IF(K.EQ.2) GO TO 12
0057            JJ=1
0058            DO  11  J=I,I+3
0059            SEGNO(JJ)=CD(J)
0060            JJ=JJ+1
0061    11      CONTINUE
0062            GO TO 13
0063    12      CONTINUE
0064            JJ=1
0065            DO  14  J=I,I+12
0066            DIF=J-I+1
0067            IF(DIF.EQ.7)  GO TO 14
0068            RECORD(JJ)=CD(J)
0069            JJ=JJ+1
0070    14      CONTINUE
0071    13      CONTINUE
0072            R=0
0073            FILE=0
0074    41      CONTINUE
0075            CALL REDHED(IBUF,R,FILE)
0076            IF(FILE.LT.2)  GO TO 47
0077            WRITE(6,89)
0078    89      FORMAT(1H0,10X,'SEGMENT NO. OR RECORD ID NOT FOUND!')
0079            GO TO 15
0080    47      CONTINUE
0081            R=R+1
        C       WRITE(6,42) R
0082    42      FORMAT(10X,' R= ',I4)
0083            IF(IBUF(1).NE.'R')  GO TO 41
0084            IF(IBUF(2).GT.'1')  GO TO 41
0085            IF(K.EQ.2)  GO TO 16
0086            DO  43  I=1,4
0087            IF(SEGNO(I).NE.IBUF(147+I)) GO TO 41
0088    43      CONTINUE
0089            SFLG=1
        C       WRITE(6,45)
0090    45      FORMAT(10X,'  FOUND SEGMENT  ')
0091            GO TO 46
0092    34      R=0
0093            FILE=0
0094    48      CALL REDHED(IBUF,R,FILE)
0095            R=R+1
0096            IF(IBUF(1).NE.'R') GO TO 48
0097    46      CALL CAMPER(IBUF,R,FILE)
0098            CALL REDHED(IBUF,R,FILE)
0099            DFLG=0
0100            CALL STDATA(IBUF,R,FILE,DFLG)
0101            CALL REDHED(IBUF,R,FILE)
0102            CALL STDATA(IBUF,R,FILE,DFLG)
0103            CALL REDHED(IBUF,R,FILE)
0104            CALL FLDPER(IBUF,R,FILE)
0105            IF(FILE.EQ.2)  GO TO 15
0106            IF(K.EQ.4)  GO TO 46
0107            GO TO 15
0108    44      CONTINUE
```

```
0001            SUBROUTINE CAMPER(IBUF,R,FILE)
0002            IMPLICIT INTEGER(A-Z)
0003            LOGICAL*1  IBUF(1),CLIST(240),CLASS(4)
0004             LOGICAL*1 PB(132),BLANK,PT
0005            LOGICAL*1 SEGNO(4)
0006             DIMENSION CT(16)
0007            INTEGER WI(60)
0008            REAL PW(60),X,Y,PCW,PNW
0009            REAL  PDU,PDO,PTH,PX,PWTH
0010             COMMON/SEG/SEGNO
0011            DATA BLANK/1H /
0012             DATA PT/1H./
0013            DO 11  I=1,132
0014    11       PB(I)=BLANK
0015             CALL CAMHOG(IBUF)
0016             WRITE(6,99)
0017    99       FORMAT(1H1,45X,'CLASSIFICATION SUMMARY REPORT')
0018             WRITE(6,94)
0019    94       FORMAT(1H0)
0020             WRITE(6,98)
0021    98       FORMAT(1H ,62X,'NORMALIZED')
0022             II=0
0023            DO 95 JJ=1,16
0024             IF(IBUF(83+JJ).EQ.'1')  GO TO 96
0025             GO TO 95
0026    96       II=II+1
0027             CT(II)=JJ
0028    95       CONTINUE
0029            IIMAX=II
0030             WRITE(6,97)
0031    97       FORMAT(1H ,25X,'SUBCLASS NAME    THRESHOLD VALUE   ';
                *'APRIORI VALUE    ')
0032             K=0
0033             J=1
0034             N=1
0035            RJ=249
0036            SETSR=14
0037            LFLD=32
0038            WIT=0
0039            XC=0
0040            YC=0
0041            DO  3 II=1,60
0042    3       WI(II)=0
0043            DECODE(4,100,IBUF(56))  NOSUB
0044    100     FORMAT(I4)
0045            DECODE(5,101,IBUF(64))  DO
0046    101     FORMAT(I5)
0047            DECODE(5,101,IBUF(69))  DU
0048            GO TO 2
0049    1       CALL REDHED(IBUF,R,FILE)
0050            RJ=3
0051            SETSR=22
0052            N=1
0053    2       CONTINUE
0054            CALL CPIPO(IBUF(RJ),CLASS,PI,PO)
0055             CALL MV(IBUF(RJ),PB(28),6)
```

```
0056              CALL MV(IBUF(RJ+15),PB(46),3)
0057              PB(49)=PT
0058              PB(50)=IBUF(RJ+18)
0059              CALL MV(IBUF(RJ+19),PB(66),3)
0060              WRITE(6,111)  (PB(PJ),PJ=27,79)
0061         111  FORMAT(29X,60A1)
0062              TC=TC+PO
0063              IF(CLASS(1).EQ.'X')  GO TO 10
0064              IF(CLASS(1).EQ.'W')  GO TO 20
0065              GO TO 30
0066         10   XC=XC+PI
0067         30   IF(J.EQ.NOSUB)  GO TO 40
0068              J=J+1
0069              IF(N.EQ.SETSR)  GO TO 1
0070              N=N+1
0071              RJ=RJ+LFLD
0072              GO TO 2
0073         20   IF(K.EQ.0)  GO TO 5
0074              LK=4*(K-1)
0075              DO  4 II=1,4
0076              IF(CLASS(II).NE.CLIST(LK+II))  GO TO 5
0077         4    CONTINUE
0078              WI(K)=WI(K)+PI
0079              WIT=WIT+PO+PI
0080              GO TO 30
0081         5    K=K+1
0082              LK=4*(K-1)
0083              DO  6 II=1,4
0084         6    CLIST(LK+II)=CLASS(II)
0085              WI(K)=WI(K)+PI
0086               WIT=WIT+PO+PI
0087              GO TO 30
0088         40   CONTINUE
0089               WRITE(6,94)
0090               WRITE(6,93)  (CT(II),II=1,IIMAX)
0091         93    FORMAT(25X,'CHANNELS USED! ',16(1X,I2))
0092              PC=22932
0093              D=PC-DO-DU-XC
0094              Y=FLOAT(XC)+FLOAT(DU)
0095              DO  41 JJ=1,K
0096              X=FLOAT(WI(JJ))
0097              PW(JJ)=(X+(X*Y)/FLOAT(D))/FLOAT(PC)
0098               PW(JJ)=PW(JJ)*100.
0099         41   CONTINUE
0100              PCW=0.
0101              DO 42  JJ=1,K
0102              PCW=PCW+PW(JJ)
0103         42   CONTINUE
0104              PNW=100.-PCW
0105              PDU=(FLOAT(DU)/FLOAT(PC))*100.
0106              PDO=(FLOAT(DO)/FLOAT(PC))*100.
0107              D=PC-DO-DU
0108              PTH=(FLOAT(TC)/FLOAT(D))*100.
0109              PX=(FLOAT(XC)/FLOAT(PC))*100.
0110              X=FLOAT(WIT)
0111              PWTH=(X+(X*Y)/FLOAT(D))/FLOAT(PC)
```

```
0112            PWTH=PWTH*100.
0113            WRITE(6,200)
0114    200     FORMAT('1',21X,'SEGMENT PERCENTAGES')
0115            WRITE(6,94)
0116            WRITE(6,201)  PCW
0117    201     FORMAT(1H ,21X,'WHEAT CATEGORY      = ',F5.1)
0118            WRITE(6,202)  PNW
0119    202     FORMAT(1H ,21X,'NON WHEAT CATEGORY  = ',F5.1)
0120            WRITE(6,203)  PDO
0121    203     FORMAT(1H ,21X,'DESIGNATED OTHER    = ',F5.1)
0122            WRITE(6,204)  PTH
0123    204     FORMAT(1H ,21X,'THRESHOLD           = ',F5.1)
0124            WRITE(6,205)  PDU
0125    205     FORMAT(1H ,21X,'DESIGNATED UNIDENT. = ',F5.1)
0126            WRITE(6,206)  PX
0127    206     FORMAT(1H ,21X,'X CATEGORY          = ',F5.1)
0128            WRITE(6,209)
0129    209     FORMAT(1H0)
0130            DO  207  I=1,K
0131            J1=1
0132            J2=4
0133            WRITE(6,208) (CLIST(JJ),JJ=J1,J2),PW(I)
0134            J1=J1+4
0135            J2=J2+4
0136    207     CONTINUE
0137    208     FORMAT(1H ,21X,'WHEAT CLASS ',4A1,'  = ',F5.1)
0138            WRITE(6,210)  PWTH
0139    210     FORMAT(1H0,21X,'WHEAT, 0 THRESHOLD: ',F5.1)
0140            RETURN
0141            END
```

```
 FORTRAN IV-PLUS V02-04                    15:32:58    08-FEB-77          PAGE 1
 CPIPO.FIN          /IB:BLOCKS/NR

0001            SUBROUTINE CPIPO(FLD,CLASS,PI,PO)
0002            IMPLICIT INTEGER(A-Z)
0003            LOGICAL*1  FLD(1),CLASS(1)
0004            DO 2  I=1,4
0005            CLASS(I)=FLD(I)
0006     2      CONTINUE
0007            DECODE(5,100,FLD(23))  PI
0008     100    FORMAT(I5)
0009            DECODE(5,100,FLD(28))  PO
0010            RETURN
0011            END
```

```
 FORTRAN IV-PLUS V02-04                  15133112     08-FEB-77          PAGE 1
 CAMHDG.FIN       /TR:BLOCKS/WR

0001                SUBROUTINE CAMHDG(IBUF)
0002                IMPLICIT INTEGER(A-Z)
0003                LOGICAL*1 IBUF(1),SEGNO(4)
0004                LOGICAL*1 PB(132),BLANK
0005                COMMON/SEG/SEGNO
0006                DATA BLANK/1H /
0007                DO 1 I=1,4
0008           1    SEGNO(I)=IBUF(147+I)
0009                WRITE(6,400)
0010           400  FORMAT('1',49X,'CAMS INTERFACE REPORT')
0011                WRITE(6,401) (IBUF(I),I=3,26)
0012           401  FORMAT(1H0,3X,'OPAR NO.   = ',24A1)
0013                WRITE(6,402)  (IBUF(I),I=709,714),(IBUF(J),J=715,720)
0014           402  FORMAT(1H0,3X,'RECORD ID = ',6A1,1X,6A1)
0015                WRITE(6,403)
0016           403  FORMAT(1H0,50X,'ACQUISITION DATES')
0017                WRITE(6,404)
0018           404  FORMAT(1H ,47X,'1',6X,'2',6X,'3',6X,'4',15X,
                    *'SEGMENT TYPE')
0019                DO 11 N=1,132
0020           11   PB(N)=BLANK
0021                RJ=148
0022                CALL MV(IBUF(RJ),PB(31),4)
0023                IP=47
0024                PB(91)=IBUF(RJ+4)
0025                RI=RJ+6
0026                DO 9 J=1,4
0027                IF(IBUF(RI).EQ.'0') GO TO 10
0028                CALL MV(IBUF(RI),PB(IP),5)
0029                RI=RI+5
0030                IP=IP+7
0031           9    CONTINUE
0032           10   WRITE(6,301)  (PB(K),K=30,95)
0033           301  FORMAT(1H ,3X,'RECOGNITION SEG. NUMBER #',66A1)
0034                RJ=170
0035                DO 2 I=1,3
0036                DO 12 N=1,132
0037           12   PB(N)=BLANK
0038                IF(IBUF(RJ).EQ.'0') GO TO 4
0039                CALL MV(IBUF(RJ),PB(31),4)
0040                PB(91)=IBUF(RJ+4)
0041                IP=47
0042                RI=RJ+5
0043                DO 3 J=1,4
0044                IF(IBUF(RI).EQ.'0') GO TO 4
0045                CALL MV(IBUF(RI),PB(IP),5)
0046                RI=RI+5
0047                IP=IP+7
0048           3    CONTINUE
0049                WRITE(6,300)  I,(PB(K),K=30,95)
0050           300  FORMAT(1H ,'TRAINING SEGMENT NO. ',I1,' =',66A1)
0051                RJ=RJ+25
0052           2    CONTINUE
0053           4    CONTINUE
0054                RETURN
0055                END
```

```
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
```

```
FORTRAN IV-PLUS V02-04                    15:33:31    08-FEB-77          PAGE 1
STDATA.FTN        /IP:BLOCKS/WR

0001            SUBROUTINE STDATA(IBUF,R,FILE,DFLG)
0002            IMPLICIT INTEGER(A-Z)
0003            LOGICAL*1 IBUF(1),PFN(132),PP(132),PN(132)
0004            LOGICAL*1 PTL(132),PM(2112)
0005            LOGICAL*1 BLANK
0006             LOGICAL*1 PF(132)
0007             COMMON/PCNT/LINE,PAGE
0008            DATA BLANK/1H /
0009             IF(DFLG.EQ.0) GO TO 12
0010             CALL STOMP(IBUF)
0011            GO TO 13
0012     12      CONTINUE
0013             WRITE(6,97)
0014     97      FORMAT('1',49X,'STATISTICS REPORT')
0015     13      CONTINUE
0016             NMAX=110
0017            K=1
0018            I=1
0019             PI=11
0020             DECODE(3,99,IBUF(5))  ALSETS
0021     99      FORMAT(I3)
0022             DECODE(2,98,IBUF(10))  NCH
0023     98      FORMAT(I2)
0024             NLINS=62
0025             XL=NCH+6
0026             IF(DFLG.EQ.1) NLINS=NLINS-7
0027             LSETS=NLINS/XL
0028            OSETS=5
0029             PSETS=0
0030            LFLD=11+9*NCH
0031             IF(IBUF(2).EQ.'F')  LFLD=18+9*NCH
0032     20      J=1
0033             RJ=12
0034             IF(K.EQ.1)  GO TO 6
0035             CALL REDHED(IBUF,R,FILE)
0036     4       DECODE(2,98,IBUF(8))  SETSR
0037     1        CONTINUE
0038             IF(IBUF(2).EQ.'S')  GO TO 22
0039             CALL FNAME(IBUF(RJ),PFN(PI))
0040     22      CALL POP(IBUF(RJ),PP(PI),IBUF(2))
0041             CALL SNAME(IBUF(RJ),PN(PI),IBUF(2))
0042             CALL MOTTL(PTL(PI))
0043              CALL MEAN(IBUF(RJ),PM(PI),NCH,IBUF(2))
0044             IF(K.EQ.ALSETS)  GO TO 10
0045             K=K+1
0046             IF(I.EQ.OSETS)  GO TO 9
0047             I=I+1
0048             PI=PI+19
0049     2       IF(J.EQ.SETSR)  GO TO 20
0050             J=J+1
0051             RJ=RJ+LFLD
0052             GO TO 1
0053     9        K=K-1
0054     10      CONTINUE
0055              PSETS=PSETS+1
0056              IF(PSETS.LE.LSETS)  GO TO 11
```

```
0057            PSETS=1
0058            IF(DFLG.EQ.1)  GO TO 14
0059            WRITE(6,96)
0060    96      FORMAT('1')
0061            GO TO 11
0062    14      CONTINUE
0063            LINE=66
0064            CALL KNT
0065    11      CONTINUE
0066            WRITE(6,90)
0067    90      FORMAT(1H0)
0068            IF(IBUF(2).EQ.'S')  GO TO 3
0069            WRITE(6,101)  (PFN(N),N=1,NMAX)
0070    101     FORMAT(1H ,131A1)
0071            WRITE(6,102)  (PP(N),N=10,NMAX)
0072    102     FORMAT(1H ,'CHANNEL ',122A1)
0073            WRITE(6,101)  (PN(N),N=1,NMAX)
0074            GO TO 7
0075    3       CONTINUE
0076            WRITE(6,101)  (PN(N),N=1,NMAX)
0077            WRITE(6,102)  (PP(N),N=10,NMAX)
0078    7       WRITE(6,103)  (PTL(N),N=8,NMAX)
0079    103     FORMAT(1H ,'NUMBER ',122A1)
0080            IN=6
0081            IM=NMAX
0082            DO 5  II=1,NCH
0083            WRITE(6,104)  (II,(PM(N),N=IN,IM))
0084    104     FORMAT(1H ,2X,I2,128A1)
0085            IN=IN+132
0086            IM=IM+132
0087    5       CONTINUE
0088    6       CONTINUE
0089            DO  30 N=1,132
0090            PF(N)=BLANK
0091             PFN(N)=BLANK
0092            PN(N)=BLANK
0093            PTL(N)=BLANK
0094            PP(N)=BLANK
0095            DO  40  NN=1,NCH
0096            PM(N+(NN-1)*132)=BLANK
0097    40      CONTINUE
0098    30      CONTINUE
0099            IF(K.EQ.1)  GO TO 4
0100            IF(K.EQ.ALSETS)  RETURN
0101             K=K+1
0102            I=1
0103            PI=11
0104            GO TO 2
0105            END
```

```
 FORTRAN IV-PLUS V02-04                    15133150    08-FEB-77          PAGE 1
 MEAN.FTN          /TR:BLOCKS/WR

0001            SUBROUTINE MEAN(FLD,PM,NCH,FLG)
0002            IMPLICIT INTEGER(A-Z)
0003            LOGICAL*1 FLD(1),PM(1),FLG(1),PT
0004             LOGICAL*1 BLK
0005            DATA PT/1H./
0006             DATA BLK/1H /
0007            OFF=11
0008            IF(FLG(1).EQ.'F')  OFF=18
0009            K=1
0010             DO  3  J=1,NCH
0011            N=(J-1)*132
0012             PM(N+1)=FLD(OFF+1)
0013             PM(N+2)=FLD(OFF+2)
0014             IF(PM(N+1).NE.'0')  GO TO 10
0015             PM(N+1)=BLK
0016             IF(PM(N+2).EQ.'0')  PM(N+2)=BLK
0017       10    PM(N+3)=FLD(OFF+3)
0018             PM(N+4)=PT
0019             PM(N+5)=FLD(OFF+4)
0020             PM(N+6)=FLD(OFF+5)
0021             PM(N+7)=BLK
0022             PM(N+8)=FLD(OFF+6)
0023             PM(N+9)=FLD(OFF+7)
0024             IF(PM(N+8).NE.'0')  GO TO 11
0025             PM(N+8)=BLK
0026             IF(PM(N+9).EQ.'0')  PM(N+9)=BLK
0027       11    PM(N+10)=FLD(OFF+8)
0028             PM(N+11)=PT
0029             PM(N+12)=FLD(OFF+9)
0030             OFF=OFF+9
0031        3    CONTINUE
0032             RETURN
0033             END
```

```
 FORTRAN IV-PLUS V02-04                    15134101     08-FEB-77          PAGE 1
 POP.FIN          /TR:BLOCKS/WR

0001            SUBROUTINE POP(FLD,PP,FLG)
0002            IMPLICIT INTEGER(A-Z)
0003            LOGICAL*1 FLD(1),PP(1),PAR(2),FLG(1)
0004            DATA PAR/1H(,1H)/
0005            OFF=6
0006            IF(FLG(1).EQ.'F')  OFF=13
0007            PP(3)=PAR(1)
0008            DO  1  I=1,5
0009            PP(I+3)=FLD(I+OFF)
0010          1 CONTINUE
0011            PP(9)=PAR(2)
0012            RETURN
0013            END
```

```
 FORTRAN IV-PLUS V02-04          15:34:08    08-FEB-77          PAGE 1
 FNAME.FTN       /TR:BLOCKS/HR

0001            SUBROUTINE FNAME(FLD,PN)
0002            IMPLICIT INTEGER(A-Z)
0003            LOGICAL*1  FLD(1),PN(1),FIELD(6)
0004            DATA FIELD/1HF,1HI,1HE,1HL,1HD,1H /
0005            DO 11 II=1,6
0006            PN(II)=FIELD(II)
0007            PN(II+6)=FLD(II)
0008     11     CONTINUE
0009            RETURN
0010            END
```

```
              SUBROUTINE SNAME(FLD,PN,FLG)
              IMPLICIT INTEGER(A-Z)
              LOGICAL*1  FLD(1),PN(1),SUBCL(6),FLG(1)
              DATA SUBCL/1HS,1HU,1HB,1HC,1HL,1H /
              OFF=0
              IF(FLG(1).EQ.'F')  OFF=7
              DO 11  II=1,6
              PN(II)=SUBCL(II)
              PN(II+6)=FLD(II+OFF)
      11      CONTINUE
              RETURN
              END
```

```
 FORTRAN IV-PLUS V02-04                    15:34:20    08-FEB-77          PAGE 1
 MDTTL.FTN                /TR:BLOCKS/WR

0001          SUBROUTINE MDTTL(PTL)
0002          IMPLICIT INTEGER(A-Z)
0003          LOGICAL*1  MSG(12),PTL(1)
0004           DATA MSG/1HM,1HE,1HA,1HN,1H ,1H ,1HS,1HT,1H ,1HD,1HE,1HV/
0005          DO 1 I=1,13
0006        1 PTL(I)=MSG(I)
0007          RETURN
0008          END
```

```
 FORTRAN IV-PLUS V02-04          15:34:39    08-FEB-77          PAGE 1
 FLOPER.FTN      /IR:BLOCKS/WR
0001            SUBROUTINE FLOPER(IBUF,R,FILE)
0002            IMPLICIT INTEGER(A-O,Q-Z),REAL (P)
0003            INTEGER  PI,PO
0004            LOGICAL*1  IBUF(1),CLIST(240),FASC(4)
0005            DIMENSION ICAT(60),IC(60),IF(60)
0006            DIMENSION PIF(60),PIC(60),PICAT(60)
0007            LOGICAL*1  FIELD(6),CLASS(4)
0008             LOGICAL*1 SEGNO(4)
0009            COMMON/SEG/SEGNO
0010            LFLD=18
0011            CFILE=FILE
0012            SETSR=38
0013            RJ=23
0014            DECODE(2,100,IBUF(21))  NOSUB
0015     100    FORMAT(I2)
                C
                C SET CATEGORY NUMBERS TO ZERO
                C
0016     1      CONTINUE
0017             HDFLG=0
0018            DO  11 I=1,60
0019     11     ICAT(I)=0
0020            TCAT=0
0021            IPCAT=0
0022            CATW=0
0023            CATX=0
0024            CATNW=0
                C
                C SET CLASS NUMBERS TO ZERO
                C
0025     2      CONTINUE
0026            DO  111 I=1,60
0027     111    IC(I)=0
0028            IPC=0
0029            TC=0
0030            CW=0
0031            CNW=0
0032            CX=0
                C
                C SE FIELD NUMBERS TO ZERO
                C
0033     3      CONTINUE
0034            DO 112  I=1,60
0035     112    IF(I)=0
0036            TF=0
0037            FW=0
0038            FNW=0
0039            FX=0
                C
                C SAVE FIELD ASSOCIATED SUBCLASS NAME
                C
0040            DO  119 I=1,4
0041     119    FASC(I)=IBUF(I+8)
0042            DO 77  I=1,6
0043     77     FIELD(I)=IBUF(2+I)
0044            DECODE(5,101,IBUF(16)) IPF
```

```
0045     101    FORMAT(I5)
0046            K=0
0047            N=1
0048            J=1
0049            EXFLG=0
0050     22     CONTINUE
                C
                C OBTAIN SUBCLASS PIXEL COUNTS FOR CURRENT FIELD
                C
0051            CALL FPIPO(IBUF(RJ),CLASS,PI,PO)
0052            IF(CLASS(1).EQ.'W') FW=FW+PI
0053            IF(CLASS(1).EQ.'NW') FNW=FNW+PI
0054            IF(CLASS(1).EQ.'X') FX=FX+PI
0055            IF(K.EQ.0) GO TO 5
0056            LK=4*(K-1)
0057            DO 4  II=1,4
0058            IF(CLASS(II).NE.CLIST(LK+II))  GO TO 5
0059     4      CONTINUE
0060            GO TO 7
0061     5      K=K+1
0062            LK=4*(K-1)
0063            DO 6  II=1,4
0064     6      CLIST(LK+II)=CLASS(II)
0065     7      CONTINUE
0066            IF(K)=(F(K)+PI
0067            TF=TF+PO
                C
                C TEST FOR LAST SUBCLASS
                C
0068            IF(J.EQ.NOSUB)  GO TO 40
0069            J=J+1
0070            IF(N.EQ.SETSR)  GO TO 33
0071            N=N+1
0072            RJ=RJ+LFLD
0073            GO TO 22
0074     33     CALL REDHED(IBUF,R,CFILE)
0075            N=1
0076            RJ=3
0077            GO TO 22
0078     40     CONTINUE
                C
                C COMPUTE FIELD CLASS PERCENTAGES
                C
0079            D=FLOAT(IPF)
0080            DO 9  II=1,K
0081     9      PTF(II)=(FLOAT(IF(II))/D)*100.
0082            PTF=(FLOAT(TF)/D)*100.
0083            PW=(FLOAT(FW)/D)*100.
0084            PNW=(FLOAT(FNW)/D)*100.
0085            PX=(FLOAT(FX)/D)*100.
0086            IF(HDFLG.EQ.1)  GO TO 441
0087             WRITE(6,698)  (SEGNO(I),I=1,4)
0088     698     FORMAT('1',36X,'ACCURACY REPORT - SEGMENT ',4A1)
0089            HDFLG=1
0090            KK=4*K
0091            WRITE(6,699)  (CLIST(I),I=1,KK)
```

```
0092     699    FORMAT(1H0,4X,'NAME',9X,'POP.    WHEAT      NONWHEAT    TH',
               *'RES.',2X,'X-CAT',6X,5(4A1,4X)/(64X,5(4A1,4X)))
0093            WRITE(6,703)
0094     441    CONTINUE
0095            WRITE(6,700)  FIELD,IPF,PW,PNW,PTF,PX,(PIF(I),I=1,K)
0096     700    FORMAT(1H ,2X,'FIELD ',6A1,2X,I5,2X,4(F5.1,5X),5(F5.1,3X)/
               *(64X,5(4A1,4X)))
C
C UPDATE CLASS CLASS PIXEL COUNTS
C
0097            DO 10  I=1,K
0098     10     IC(I)=IC(I)+IF(I)
0099            TC=TC+TF
0100            IPC=IPC+IPF
0101            CW=CW+FW
0102            CNW=CNW+FNW
0103            CX=CX+FX
C
C READ NEXT FIELD DATA RECORD
C
0104            CALL REDHED(IBUF,R,CFILE)
0105            IF(CFILE.GT.FILE)  GO TO 41
0106            IF(IBUF(1).NE.'F')  GO TO 41
0107            RJ=23
0108            DO 42  I=1,4
0109            IF(FASC(I).NE.IBUF(8+I))  GO TO 43
0110     42     CONTINUE
0111            GO TO 3
0112     41     EXFLG=1
0113            FILE=CFILE
C
C COMPUTE CLASS CLASS PERCENTAGES
C
0114     43     D=FLOAT(IPC)
0115            DO 44  I=1,K
0116            PIC(I)=(FLOAT(IC(I))/D)*100.
0117     44     CONTINUE
0118            PTC=(FLOAT(TC)/D)*100.
0119            PCW=(FLOAT(CW)/D)*100.
0120            PCNW=(FLOAT(CNW)/D)*100.
0121            PCX=(FLOAT(CX)/D)*100.
0122            WRITE(6,701)  FASC,IPC,PCW,PCNW,PTC,PCX,(PIC(I),I=1,K)
0123     701    FORMAT(1H0,' CLASS ',4A1,5X,I5,2X,4(F5.1,5X),5(F5.1,3X)/
               *(64X,5(4A1,4X)))
0124            WRITE(6,703)
C
C UPDATE CATEGORY CLASS COUNTS
C
0125            DO  45  I=1,K
0126     45     ICAT(I)=ICAT(I)+IC(I)
0127            TCAT=TCAT+TC
0128            IPCAT=IPCAT+IPC
0129            CATW=CATW+CW
0130            CATNW=CATNW+CNW
0131            CATX=CATX+CX
0132            IF(EXFLG.EQ.1)  GO TO 47
```

```
0133            IF(FASC(1).NE.IBUF(9))  GO TO 47
0134            GO TO 2
                C
                C  COMPUTE CATEGORY CLASS PERCENTAGES
                C
0135      47    D=FLOAT(IPCAT)
0136            DO 48  I=1,K
0137            PICAT(I)=(FLOAT(ICAT(I))/D)*100.
0138      48    CONTINUE
0139            PTCAT=(FLOAT(TCAT)/D)*100.
0140            PCATW=(FLOAT(CATW)/D)*100.
0141            PCATNW=(FLOAT(CATNW)/D)*100.
0142            PCATX=(FLOAT(CATX)/D)*100.
0143            WRITE(6,702) FASC(1),IPCAT,PCATW,PCATNW,PTCAT,PCATX,
               *(PICAT(I),I=1,K)
0144      702   FORMAT(1H ,'CAT.  ',1A1,9X,I5,2X,4(F5.1,5X),5(F5.1,3X)/
               *(64X,5(4A1,4X)))
0145            WRITE(6,703)
0146      703   FORMAT(1H0)
0147            IF(EXFLG.EQ.1) GO TO 49
0148            GO TO 1
0149      49    CONTINUE
0150            FILE=CFILE
0151            RETURN
0152            END
```

```
 FORTRAN IV-PLUS V02-04                  15133104     08-FEB-77          PAGE 1
 FPIPO.FTN          /TR:BLOCKS/WR

0001          SUBROUTINE FPIPO(FLD,CLASS,PI,PO)
0002          IMPLICIT INTEGER(A-Z)
0003          LOGICAL*1  FLD(1),CLASS(1)
0004          DO 2  I=1,4
0005          CLASS(I)=FLD(I)
0006     2    CONTINUE
0007          DECODE(5,100,FLD(9))  PI
0008   100    FORMAT(I5)
0009          DECODE(5,100,FLD(14))  PO
0010          RETURN
0011          END
```

```
FORTRAN IV-PLUS V02-04                  15:35:12    08-FEB-77          PAGE 1
CAPRNT.FTN          /TP:BLOCKS/WR

0001          SUBROUTINE CAPRNT(IBUF,R,FILE)
0002          IMPLICIT INTEGER(A-Z)
0003          LOGICAL*1 IBUF(1)
0004          COMMON/PCNT/LINE,PAGE
0005          COMMON/NSUB/NS,NCLS
0006          IF(IBUF(1).EQ. 'V') GO TO 1
0007          IF(IBUF(1).EQ. 'H') GO TO 2
0008          IF(IBUF(1).EQ. 'R') GO TO 3
0009          IF(IBUF(1).EQ. 'S') GO TO 6
0010          IF(IBUF(1).EQ. 'F') GO TO 5
0011          IF(IBUF(1).EQ. 'E') GO TO 7
0012          GO TO 7
0013        1 PAGE=0
0014          LINE=66
0015          CALL KNT
0016          WRITE(6,203) (IBUF(I),I=1,3)
0017          WRITE(6,99)
0018          WRITE(6,103) IBUF(4)
0019      103 FORMAT(5X,A1)
0020          WRITE(6,104) (IBUF(I),I=5,10)
0021      104 FORMAT(5X,6A1/)
0022          WRITE(6,103) IBUF(11)
0023          WRITE(6,104) IBUF(80)
0024       99 FORMAT(5X,'RECORD 1!')
0025          RETURN
0026        2 IF( IBUF(4) .EQ. '2') GO TO 21
0027          WRITE(6,201)
0028      201 FORMAT(5X,'RECORD 2!')
0029          WRITE(6,203) (IBUF(I),I=1,3)
0030          WRITE(6,104) IBUF(4)
0031          WRITE(6,203) (IBUF(I),I=5,21)
0032      203 FORMAT(5X,30A1/)
0033          WRITE(6,203) FSNO
0034          WRITE(6,203) (IBUF(I),I=49,53)
0035          WRITE(6,203)  IBUF(54)
0036          RETURN
0037       21 WRITE(6,206)
0038      206 FORMAT(5X,'RECORD 3!'/)
0039          WRITE(6,203) (IBUF(I),I=1,3)
0040          WRITE(6,104) IBUF(4)
0041          WRITE(6,203) IBUF(5)
0042          WRITE(6,203) (IBUF(I),I=6,10)
0043          WRITE(6,203) (IBUF(I),I=11,15)
0044          WRITE(6,210)(IBUF(I),I=51,52)
0045      210 FORMAT(5X,2A1)
0046          RETURN
0047        3 CONTINUE
0048      301 FORMAT(I1)
0049          DECODE(1,301,IBUF(2)) RSN
0050          IF(RSN .GT. 1) GO TO 31
0051          LINE=66
0052          CALL KNT
0053          IF(RSN.LT. 2) WRITE(6,999)
0054          CALL KNT
0055      999 FORMAT(5X,'RECOGNITION SEGMENT')
0056          IF(RSN.EQ.0) RSN=1
```

```
0057            CALL KNT
0058            WRITE(6,302) RSN
0059            CALL KNT
0060        302 FORMAT(5X,'RECORD:',I2/)
0061            CALL KNT
0062            CALL KNT
0063            WRITE(6,203) IBUF(2)
0064            CALL KNT
0065            WRITE(6,203) (IBUF(I),I=3,26)
0066            CALL KNT
0067            WRITE(6,203) (IBUF(I),I=27,40)
0068            CALL KNT
0069            WRITE(6,303) (IBUF(I),I=41,42),(IBUF(I),I=43,48)
0070        303 FORMAT(5X,2A1,',',6A1)
0071            CALL KNT
0072            WRITE(6,203) (IBUF(I),I=49,55)
0073            CALL KNT
0074        777 FORMAT(I4)
0075            DECODE(4,777,IBUF(56))NCLS
0076            WRITE(6,203) (IBUF(I),I=56,59)
0077            CALL KNT
0078            WRITE(6,203) (IBUF(I),I=60,63)
0079            CALL KNT
0080            WRITE(6,203) (IBUF(I),I=64,68)
0081            CALL KNT
0082            WRITE(6,203) (IBUF(I),I=69,73)
0083            CALL KNT
0084            WRITE(6,203) (IBUF(I),I=74,78)
0085            CALL KNT
0086            WRITE(6,203) IBUF(83)
0087            CALL KNT
0088            WRITE(6,203) (IBUF(I),I=84,99)
0089            CALL KNT
0090            WRITE(6,303) (IBUF(I),I=100,101),(IBUF(I),I=102,107)
0091            ST=108
0092            DO 304 J=1,8
0093            L=ST+4
0094            CALL KNT
0095            WRITE(6,203) (IBUF(I),I=ST,L)
0096            ST=ST+5
0097        304 CONTINUE
0098            CALL KNT
0099            WRITE(6,203) (IBUF(I),I=148,151)
0100            CALL KNT
0101            WRITE(6,203) IBUF(152)
0102            CALL KNT
0103            WRITE(6,203) IBUF(153)
0104            ST=154
0105            DO 305 J=1,4
0106            L=ST+4
0107            CALL KNT
0108            WRITE(6,203) (IBUF(I),I=ST,L)
0109            ST=ST+5
0110        305 CONTINUE
0111            ST=174
0112            DO 306 J=1,3
```

```
0113            L1=ST+3
0114            CALL KNT
0115            WRITE(6,203) (IBUF(I),I=ST,L1)
0116            ST=L1+1
0117            CALL KNT
0118            WRITE(6,203) IBUF(ST)
0119            ST=ST+1
0120            DO 307 JJ=1,4
0121            L2=ST+4
0122            CALL KNT
0123            WRITE(6,203) (IBUF(I),I=ST,L2)
0124            ST=ST+5
0125        307 CONTINUE
0126            ST=ST+1
0127        306 CONTINUE
0128            CALL KNT
0129            CALL KNT
0130            WRITE(6,308)
0131            CALL KNT
0132        308 FORMAT(/,19X,'FL',10X,'T',11X,'A.P.',10X,'P.INTO',9X,'P.OUT'/)
0133            ST1=249
0134        311 LOOP=14
0135            IF(NCLS.LT.14) LOOP=NCLS
0136            IF(RSN .EQ. 4) LOOP=2
0137            DO 309 J=1,LOOP
0138            L1=ST1+5
0139            ST2=L1+1
0140            ST3=ST2+1
0141            L3=ST3+7
0142            ST41=L3+1
0143            L41=ST41+1
0144            ST42=L41+1
0145            L42=ST42+1
0146            ST5=L42+1
0147            L5=ST5+2
0148            ST6=L5+1
0149            L6=ST6+4
0150            ST7=L6+1
0151            L7=ST7+4
0152            CALL KNT
0153            WRITE(6,310)(IBUF(I),I=ST1,L1),IBUF(ST2),(IBUF(I),I=ST3,L3),(IBUF(
               *I),I=ST41,L41),(IBUF(I),I=ST42,L42),(IBUF(I),I=ST5,L5),(IBUF(I),I=
               *ST6,L6),(IBUF(I),I=ST7,L7)
0154        310 FORMAT(5X,6A1,2X,A1,2X,8A1,5X,2A1,'.',2A1,9X,3A1,11X,5A1,10X,5A1)
0155            ST1=L7+1
0156        309 CONTINUE
0157            IF( RSN .EQ. 4) WRITE(6,203) (IBUF(I),I=709,720)
0158            CALL KNT
0159            RETURN
0160        31    WRITE(6,302)  RSN
0161            CALL KNT
0162            CALL KNT
0163            NCLS=NCLS-14
0164            IF(NCLS.LE.0) RETURN
0165            ST1=3
0166            GO TO 311
```

```
0167          5 DECODE(1,301,IBUF(2)) RSN
0168            IF(RSN .GT. 1) GO TO 51
0169            LINE=66
0170            CALL KNT
0171            IF(RSN.LT.2) WRITE(6,998)
0172            CALL KNT
0173        998 FORMAT(5X,'FIELD DATA')
0174            IF(RSN.EQ.0) RSN=1
0175            CALL KNT
0176            WRITE(6,302)RSN
0177            CALL KNT
0178            CALL KNT
0179            WRITE(6,203) (IBUF(I),I=3,8)
0180            CALL KNT
0181            WRITE(6,203) (IBUF(I),I=9,14)
0182            CALL KNT
0183            WRITE(6,203) IBUF(15)
0184            CALL KNT
0185            WRITE(6,203) (IBUF(I),I=16,20)
0186         52 FORMAT(I2)
0187            DECODE(2,52,IBUF(21)) NS
0188            CALL KNT
0189            WRITE(6,203) (IBUF(I),I=21,22)
0190            LOOP=38
0191            IF( NS .LT. 38) LOOP=NS
0192            CALL KNT
0193            CALL KNT
0194            WRITE(6,53)
0195            CALL KNT
0196         53 FORMAT(/,22X,'S,A,',8X,'N,L,',11X,'P,INTO',10X,'P,OUT'//)
0197            ST1=23
0198         56 DO 54 J=1,LOOP
0199            L1=ST1+5
0200            ST2=L1+1
0201            ST3=ST2+1
0202            ST4=ST3+1
0203            L4=ST4+4
0204            ST5=L4+1
0205            L5=ST5+4
0206            CALL KNT
0207            WRITE(6,55) (IBUF(I),I=ST1,L1),IBUF(ST2),IBUF(ST3),(IBUF(I),I=ST4,
               *L4),(IBUF(I),I=ST5,L5)
0208         55 FORMAT(5X,6A1,12X,A1,11X,A1,13X,5A1,11X,5A1)
0209            ST1=L5+1
0210         54 CONTINUE
0211            RETURN
0212         51 NS=NS-38
0213            IF (NS.LE.0) RETURN
0214            LOOP=NS
0215            ST1=7
0216            GO TO 56
0217      6       DFLG=1
0218              CALL STDATA(IBUF,R,FILE,DFLG)
0219          7 RETURN
0220            END
```

```
 FORTRAN IV-PLUS V02-04                  15134126     08-FEB-77          PAGE 1
 STOMP.FTN          /IR:1BLOCKS/WR

0001                SUBROUTINE STOMP(IBUF)
0002                IMPLICIT INTEGER(A-Z)
0003                LOGICAL*1 IBUF(1)
0004                COMMON /PCNT/LINE,PAGE
0005                LINE=66
0006                CALL KNT
0007                IF(IBUF(2).EQ.'F') GO TO 8
0008                WRITE(6,200)
0009           200  FORMAT(5X,'SUBCLASS STATISTICS RECORD')
0010                GO TO 9
0011           8    WRITE(6,201)
0012           201  FORMAT(5X,'FIELD STATISTICS RECORD')
0013           9    CONTINUE
0014                WRITE(6,202)  (IBUF(N),N=3,4)
0015           202  FORMAT(5X,'RECORD SEQ. NO. ',2A1)
0016                WRITE(6,203)  (IBUF(N),N=5,7)
0017           203  FORMAT(5X,'NO. SUBCLASSES ',3A1)
0018                WRITE(6,204)  (IBUF(N),N=8,9)
0019           204  FORMAT(5X,'NO. SUBCLASS SETS ',2A1)
0020                WRITE(6,205)  (IBUF(N),N=10,11)
0021           205  FORMAT(5X,'NO. OF CHANNELS ',2A1)
0022                LINE=5
0023                RETURN
0024                END
```

```
 FORTRAN IV-PLUS V02-04                    15:34:33    08-FEB-77          PAGE 1
 KNT.FIN          /IR:BLOCKS/NR

0001            SUBROUTINE KNT
0002            IMPLICIT INTEGER(A-Z)
0003            COMMON/PCNT/LINE,PAGE
0004            LINE=LINE+1
0005            IF(LINE.LT.60) GO TO 10
0006            PAGE=PAGE+1
0007            LINE=0
0008            WRITE(6,100) PAGE
0009         10 RETURN
0010        100 FORMAT('1',26X,'CAMS/CAS INTERFACE TAPE PRINTOUT',14X,'PAGE',I4/)
0011            END
```

```
0001            SUBROUTINE REDHED(IBUF,R,FILE)
0002            IMPLICIT INTEGER(A-Z)
0003           INTEGER*2  ISTAT(2),IPRM(6)
0004            LOGICAL*1 IBUF(1)
0005           DIMENSION IHDAT(1)
0006            COMMON/MESUR/U1,V1,U2,V2
0007           DIMENSION IA(37)
0008          DATA XDEV/2HXT/
0009           DATA MDEV/2HMT/
0010            RECORD=R
0011           IF(RECORD.GT.0) GO TO 10
0012    12      CONTINUE
0013            FILE=0
              C     WRITE(5,49)
              C49   FORMAT(10X,' TYPE M OR X FOR TAPE DEVICE CODE!/)
              C     READ(5,51) IA
              C     IF(IA(1).EQ.'M') GO TO 13
              C     IF(IA(1).EQ.'X')  GO TO 14
              C     GO TO 12
0014    13      IDEV=MDEV
              C     GO TO 15
              C14   IDEV=XDEV
              C15   CONTINUE
              C      WRITE(5,50)
              C50   FORMAT(10X,' TYPE TAPE UNIT NUMBER - 0 OR 1!/)
              C      READ(5,51) IA
              C51    FORMAT(37A2)
              C      I=0
              C      CALL INTFF(I,IA,79,IUNT)
0015           ILUN=9
0016            IUNT=0
0017           IDS=0
0018           ISR=0
0019            CALL ASNLUN(ILUN,IDEV,IUNT,IDS)
0020            IF(IDS.LT.0)  GO TO 1
0021           CALL GETADR(IPRM,IBUF)
0022            CALL QIO("2400,ILUN,1,,ISTAT,IPRM,ISR)
0023            IF(ISR.LT.0) GO TO 2
0024           IPRM(2)=3600
0025    10      CONTINUE
0026            ISW=0
0027           DO  11 I=1,1800
0028    11      IBUF(I)=0
0029           RECORD=RECORD+1
0030            CALL QIO("1000,ILUN,1,,ISTAT,IPRM,ISW)
0031            IF(ISW.LT.0) GO TO 3
0032           CALL WAITFR(1,IDS)
0033           IF(IDS.LT.0)  GO TO 4
0034            CALL TOPRNT(ISTAT(1),ISTAT(2),ERCODE)
0035            IF(ERCODE.LT.0) GO TO 5
0036            IF(RECORD.EQ.1)  RETURN
0037            RETURN
0038    101     FORMAT(' TAPE ERR; RECORDS= ',I6)
0039    1       CONTINUE
0040           WRITE(6,100) IDS
0041    100    FORMAT(1H ,' ASLUN CALL DSW=  ',I6)
```

```
0042                STOP
0043      2         CONTINUE
0044                WRITE(6,200)  ISR
0045      200       FORMAT(1H ,'  REWIND DSW = ',I6)
0046                CALL IOPRNT(ISTAT(1),ISTAT(2))
0047                STOP
0048      3         CONTINUE
0049                WRITE(6,300)  ISW
0050      300       FORMAT(1H ,'  READ QIO DSW = ',I6)
0051                STOP
0052      4         CONTINUE
0053                WRITE(6,400)  IDS
0054      400       FORMAT(1H , '  WAIT  DSW = ',I6)
0055                STOP
0056      5         CONTINUE
0057                IF(ERCODE.EQ.-10)  FILE=FILE+1
0058                IF(ERCODE.EQ.-10)  RETURN
0059                WRITE(6,500) ERCODE
0060      500       FORMAT(1H0,' I/O STATUS BLOCK ERROR CODE= ',I6)
0061                STOP
0062                END
```

```
 FORTRAN IV-PLUS V02-04                    15:32:27    08-FEB-77          PAGE 1
 IOPRNT.FTN          /TR:BLOCKS/WR

0001                SUBROUTINE IOPRNT(IW0,IW1,ERCODE)
0002                IMPLICIT INTEGER(A-Z)
0003                LOGICAL*1 IW0(1)
          C         WRITE(6,100) (IW0(I),I=1,2),IW1
0004      100       FORMAT(1H0,2I5,O10)
0005                ERCODE=IW0(1)
0006                RETURN
0007                END
```

```
DK1:CAMRPT=DK1:CAMRPT,DK1:REDHED,DK1:IOPRNT
DK1:CAMPER,DK1:CPIPO,DK1:CAMHDG,DK1:MV
DK1:STDATA,DK1:MFAN,DK1:POP,DK1:FNAME
DK1:SNAME,DK1:MOTTL,DK1:STDMP,DK1:KNT
DK1:FLDPER,DK1:FPIPO,DK1:CAPRNT
DK1:[1,1]F4POTS/LB
[100,4]IMALIB/LB
/
UNITS=9
ASG=CR:5
ASG=LP:6
//
```