

N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM
MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT
CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED
IN THE INTEREST OF MAKING AVAILABLE AS MUCH
INFORMATION AS POSSIBLE

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

778-10807NMZ
80-10203
JSC-13917
NASA CR-
160629

"AS-BUILT" DESIGN SPECIFICATION
OF THE
CAMS/CAS INTERFACE TAPE REPORT
GENERATION PROGRAM FOR LACIE 7

Job Order 71-593

(TIRFs 77-0052 & 78-0010)

(E80-10203) AS-BUILT DESIGN SPECIFICATION
OF THE CAM/CAS INTERFACE TAPE REPORT
GENERATION PROGRAM FOR LACIE 7 (Lockheed
Electronics Co.) 123 p HC A06/MF A01
CSCS 05B G3/43

N80-29782

Unclas
00203

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

March 1978

LEC- 12022

JSC-13917

"AS-BUILT" DESIGN SPECIFICATION
OF THE
CAMS/CAS INTERFACE TAPE REPORT
GENERATION PROGRAM FOR LACIE 7

Job Order 71- 593

(TIRFs 77-0052 & 78-0010)

PREPARED BY

K. P. Eckel
Applications Software Section

APPROVED BY

for James A. Wilkinson
Philip L. Krumm, Manager
Data Systems Department

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

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

March 1978

LEC- 12022

CONTENTS

Section	Page
1. SCOPE	1-1
1.1 <u>GENERAL</u>	1-1
2. APPLICABLE DOCUMENTS.	2-1
3. SYSTEM DESCRIPTION	3-1
3.1 <u>HARDWARE DESCRIPTION</u>	3-1
3.2 <u>SOFTWARE DESCRIPTION</u>	3-1
3.2.1 SOFTWARE COMPONENT NO. 1 (CAMRPT)	3-1
3.2.1.1 <u>Linkages</u>	3-1
3.2.1.2 <u>Interfaces</u>	3-1
3.2.1.3 <u>Inputs</u>	3-1
3.2.1.4 <u>Outputs</u>	3-1
3.2.1.5 <u>Storage Requirements</u>	3-2
3.2.1.6 <u>Description</u>	3-2
3.2.1.7 <u>Flowcharts</u>	3-4
3.2.1.8 <u>Listing</u>	3-4
3.2.2 SOFTWARE COMPONENT NO. 2 (CAMREC)	3-5
3.2.2.1 <u>Linkages</u>	3-5
3.2.2.2 <u>Interfaces</u>	3-5
3.2.2.3 <u>Inputs</u>	3-5
3.2.2.5 <u>Storage Requirements</u>	3-5
3.2.2.6 <u>Description</u>	3-5

Section	Page
3.2.2.7 <u>Flowcharts</u>	3-6
3.2.2.8 <u>Listing</u>	3-6
3.2.3 SOFTWARE COMPONENT NO. 3 (BAUEXT)	3-7
3.2.2.1 <u>Linkages</u>	3-7
3.2.2.2 <u>Interfaces</u>	3-7
3.2.2.3 <u>Inputs</u>	3-7
3.2.2.4 <u>Outputs</u>	3-7
3.2.2.5 <u>Storage Requirements</u>	3-7
3.2.2.6 <u>Description</u>	3-7
3.2.2.7 <u>Flowcharts</u>	3-8
3.2.2.8 <u>Listing</u>	3-8
3.2.4 SOFTWARE COMPONENT NO. 4 (CLURES)	3-9
3.2.4.1 <u>Linkages</u>	3-9
3.2.4.2 <u>Interfaces</u>	3-9
3.2.4.3 <u>Inputs</u>	3-9
3.2.4.4 <u>Outputs</u>	3-9
3.2.4.5 <u>Storage Requirements</u>	3-9
3.2.4.6 <u>Description</u>	3-9
3.2.4.7 <u>Flowcharts</u>	3-10
3.2.4.8 <u>Listing</u>	3-10
3.2.5 SOFTWARE COMPONENT NO. 5 (STDATA)	3-11
3.2.5.1 <u>Linkages</u>	3-11
3.2.5.2 <u>Interfaces</u>	3-11
3.2.5.3 <u>Inputs</u>	3-11

Section	Page
3.2.5.4 <u>Outputs</u>	3-11
3.2.5.5 <u>Storage Requirements</u>	3-11
3.2.5.6 <u>Description</u>	3-11
3.2.5.7 <u>Flowcharts</u>	3-12
3.2.5.8 <u>Listing</u>	3-12
3.2.6 SOFTWARE COMPONENT NO. 6 (DOTRAY).	3-13
3.2.6.1 <u>Linkages</u>	3-13
3.2.6.2 <u>Interfaces</u>	3-13
3.2.6.3 <u>Inputs</u>	3-13
3.2.6.4 <u>Outputs</u>	3-13
3.2.6.5 <u>Storage Requirements</u>	3-13
3.2.6.6 <u>Description</u>	3-13
3.2.6.7 <u>Flowcharts</u>	3-14
3.2.6.8 <u>Listing</u>	3-14
3.2.7 SOFTWARE COMPONENT 7 (SEPRPT)	3-15
3.2.7.1 <u>Linkage</u>	3-15
3.2.7.2 <u>Interfaces</u>	3-15
3.2.7.3 <u>Inputs</u>	3-15
3.2.7.4 <u>Outputs</u>	3-15
3.2.7.5 <u>Storage Requirements</u>	3-15
3.2.7.6 <u>Description</u>	3-15
3.2.7.7 <u>Flowcharts</u>	3-15
3.2.7.8 <u>Listing</u>	3-15

Section	Page
3.2.8 SOFTWARE COMPONENT NO. 8 (CALC).	3-16
3.2.8.1 <u>Linkage</u>	3-16
3.2.8.2 <u>Interface</u>	3-16
3.2.8.3 <u>Inputs</u>	3-16
3.2.8.4 <u>Output</u>	3-16
3.2.8.5 <u>Storage Requirements</u>	3-16
3.2.8.6 <u>Description</u>	3-16
3.2.8.7 <u>Flowchart</u>	3-18
3.2.8.8 <u>Listing</u>	3-18
3.2.9 SOFTWARE COMPONENT NO. 9 (BIAPRT).	3-19
3.2.9.1 <u>Linkage</u>	3-19
3.2.9.2 <u>Interface</u>	3-19
3.2.9.3 <u>Inputs</u>	3-19
3.2.9.4 <u>Outputs</u>	3-19
3.2.9.5 <u>Storage Requirements</u>	3-19
3.2.9.6 <u>Description</u>	3-19
3.2.9.7 <u>Flowcharts</u>	3-20
3.2.9.8 <u>Listing</u>	3-20
4. OPERATING PROCEDURE	
4.1 <u>GENERAL</u>	4-1
4.2 <u>TEST PROCEDURE</u>	4-2

Appendices

A-1 PROGRAM LISTINGS	A-1
B-1 DW & DS FORMUL AE	B-1
C-1 FORMAT FOR DW & DS CURVE FITTING CONSTANTS	C-1

1. SCOPE

1.1 GENERAL

This document is the "as-built" design specification of the CAMS/CAS Interface Tape Report Generation Program for LACIE 7.

11
/

2. APPLICABLE DOCUMENTS

- TIRF 77-0052
- 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)
- TIRF 78-0010
- CAMS/CAS Interface Tape Printout after LACIE 6A JSC Memorandum, SF4-77-7-13, 7/21/77.
- "As-Built" Design Specification of the CAMS/CAS interface to tape Report Generation Program LEC-11292
- Acceptance Test Specification For CAMS/CAS Interface tape report generation Program for LACIE 7-LEC 11787
- Clarification and Prioritization of LACIE 7 CCIT Report JSC memorandum, SF4-77-11-8, 11/4/77.

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 (CAMRPT)

The main program CAMRPT 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 CDRED, CAMREC, BAUEXT, CLURES, STDATA, CONRED, and DOTRAY.

3.2.1.2 Interfaces

N/A

3.2.1.3 Inputs

CAMRPT control cards are: SEGMENT XXXX, RECORD ID XXXXXX XXXXXX, ALL, END. CAMS/CAS interface tape records are inputs to CAMRPT. See reference 4 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 2390 bytes.

3.2.1.6 Description

CAMPRT is the CAMPRT 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 to 1), then calls CONRED to read constants to be used in the calculations for DW & DS. CAMPRT 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 and 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 reports for the segment requested 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 tape at which the tape is currently positioned, the program generates 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.

The tape read subroutine CDRED tests all records for valid characters. If any invalid data is encountered an error message is output to the line printer to inform the user that the data for that segment or record is questionable. Any invalid characters in the record are converted to ones (1) and normal processing is resumed.

To generate reports for a segment the program first calls CAMREC to process recognition segment records. The processing entails generation of the optional classification Data report, the standard Separability Report and the standard header sheet for the report.

After CAMREC, the programming calls BAUEXT to process the Bias correction results records. Data from the Bias correction results records is saved in common blocks BIAS and Dummy. CAMRPT next calls CLURES to process the clustering results records and to generate the conditional cluster report. In addition CLURES saves cluster dot data in common block CLDOT for later use in the optional Dot report. Next STDATA is called to process the statistics records and generate the optional Statistics Report. Finally DOTRAY is called to process the Dot subset records and generate the standard Dot Label/Classification, Bias Correction Classification, Dot Label/Cluster and Bias Correction Cluster Reports.

In the limited printout mode the decision as to whether or not to output the optional reports is not made until the percentage of

correctly classified Dots (PCC-1 & PCC-2 for Bias Correction Classification report) or percentage of correctly clustered Dots (PCC-1 & PCC-2 for Bias Correction Cluster Report) are calculated in subroutine DOTRAY. If any of the values are less than 80%, PRTOUT is set to 2 in DOTRAY and the conditional reports are read from disc and written to the line printer. In the full printout mode (PRTOUT=1) the conditional reports are always retrieved from disc and printed out. If PRTOUT was = 2 it is reset to 0 after the report for a segment has been butput.

3.2.1.7 Flowcharts

See Flow Diagram 1.

3.2.1.8 Listing

See Appendix A.

3.2.2 SOFTWARE COMPONENT NO. 2 (CAMREC)

This program processes classification results contained in recognition segment records, outputs the conditional Classification Data report, saves data from the recognition segment records for use in generating the standard Separability Report and the standard CAMS Interface Report Header sheet.

3.2.2.1 Linkages

CAMREC is called by CAMPRT and calls subroutines BIAPRT, CPIPO, MV, CDRED, BNT and SEPRPT.

3.2.2.2 Interfaces

N/A

3.2.2.3 Inputs

Recognition segment records, containing subclass a priori and threshold values, subclass related classification results, and feature selection Bhattacharyya separability data for available acquisitions.

3.2.2.4 Outputs

CAMS Interface Report Header and a Classification Data report.

3.2.2.5 Storage Requirements

Total space allocated is 2845 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 BIAPRT with PASS=1 to have the report heading, segment

number, record ID, and acquisition dates output. Next CAMREC saves the number of channels used in classification and the Bhattacharyya separability data from the first recognition segment record for later use in generation of the Separability Report.

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 threshold, TC, in the COMMON blocks 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.

After all classification data has been processed the feature selection Bhattacharyya separability data is saved from the last recognition segment record and SEPRPT is called to generate the normal Separability Report.

3.2.2.7 Flowcharts

N/A

3.2.2.8 Listing

See Appendix A.

3.2.3 SOFTWARE COMPONENT NO. 3 (BAUEXT)

This program saves data contained in the clustering bias correction and classification bias correction results records.

3.2.3.1 Linkages

BAUEXT is called by CAMRPT. It calls CDRED.

3.2.3.2 Interfaces

N/A

3.2.3.3 Inputs

Clustering Bias correction and Classification Bias Correction results records.

3.2.3.4 Outputs

None.

3.2.3.5 Storage Requirements

Total space allocated is 2306 bytes.

3.2.3.6 Description

BAUEXT is called when the main program reads the first clustering Bias correction result record. BAUEXT saves the following data from both the Clustering Bias Correctin and Classification Bias Correction results records for up to 8 categories of interest plus the "designated other" and "unclassified" category"

Pixel pouplation

Bias corrected estimator

Machine estimate

Random estimate

Variable of bias corrected estimate

Variance

In addition the number of categories of interest and the character used for the categories of interest are saved. All data is saved in common blocks CBIAS arrays. This data is used by CALC for certain calculations and by BIAPRT for output of the normal Bias Correction reports.

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 process the cluster results records and generates the conditional cluster report.

3.2.4.1 Linkages

CLURES is called by CAMRPT. It calls CDRED.

3.2.4.2 Interfaces

N/A

3.2.4.3 Inputs

Cluster results records.

3.2.4.4 Outputs

The conditional Cluster report.

3.2.4.5 Storage Requirements

Total space allocated is 5632 bytes.

3.2.4.6 Description

CLURES is called when the main routine reads the first cluster results record. The program decodes ALSETS, the total number of clusters, and SETSR, the number of clusters in the current record. The routine then outputs the cluster report header, ALSETS as clusters generated and any options used. Next CLURES saves all data for each cluster for later output. When all clusters in the current record have been processed, another cluster results record is read in and processed as above.

After all cluster have been processed and if the cluster/dot report option is set, four additional cluster results records need to be processed. The processing consists of saving all dot

information in an array called DOTBUF for later output. In addition each dots cluster assignment is transferred to the common blocks CLCOM for later use in a different report. Finally when all cluster results records are processed in the above manner the cluster information is output as follows. For each cluster the program outputs the cluster name, the labeling dot match name, Ll distance, categorie used, brightness and greenness numbers for all Acquisitions used and information on all dots in the cluster. The clustering channel list is written at the end of the report.

3.2.4.7 Flowchart

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, FANME, 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 conditional statistics report.

3.2.5.5 Storage Requirements

Space allocated is 3792 bytes.

3.2.5.6 Description

STDATA is called from CAMRPT. 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. In addition, the pixel population and classified percentage for the category "unassigned" in the bias correction cluster report is calculated and saved in the common blocks CBIAS.

3.2.5.7 Flowcharts

N/A

3.2.5.8 Listing

See Appendix A.

3.2.6 SOFTWARE COMPONENT NUMBER 6 (DOTRAY)

This program processes Dot Data records and generator the conditional Dot Report.

3.2.6.1 Linkage

DOTRAY is called by CAMRPT. It calls BIAPRT, CALC and CDRED.

3.2.6.2 Interfaces

Dot Data records.

3.2.6.3 Inputs

N/A

3.2.6.4 Outputs

The Conditional Dot report.

3.2.6.5 Storage Requirements

Total space allocated is 2306 Bytes.

3.2.6.6 Description

DOTRAY is called by CAMPRT after reading the first Dot Data record. DOTRAY next outputs the report header to the top of the next page and processess the Dot Data records until all 209 dots are processed. DOTRAY also saves the dot Lable, dot type and classification for each dot in the common block CBIAS for later use by BIAPRT & CALC.

The processing involves outputting the following for each of 209 dots:

Dot number, line and pixel number for the dot, type and label (if any) for the dot, cluster and classification as well as the greenness and brightness of up to 4 acquisition for the Dot. Each record contains data for 15 dots. After the 15 dots in the record has been processed and output the next record is read in via CDRED and processed. After all dots are processed DOTRAY

calls CALC to perform calculation for the bias correction reports then calls BIAPRT to output the bias correction reports. Finally, DOTRAY check the value PRTOUT. If PRTOUT is 1 DOTRAY returns to the main program. If PRTOUT is not 1 all valid PCC values (percentages of TYPE 1 & TYPE 2 correctly classified or clustered dots) are tested. If any of the valid PCC's are less than 80%, then PRTOUT is set to 2, to indicate to the main program that the conditional reports are to be read from the disk and output to the line printer.

3.2.6.7 Flowcharts

N/A

3.2.6.8 Listing

See Appendix A

3.2.7 SOFTWARE COMPONENT 7 (SEPRPT)

This program processes the separability data which was saved for it by CAMREC and outputs the separability report.

3.2.7.1 Linkage

SEPRPT is called by CAMREC.

3.2.7.2 Interfaces

N/A

3.2.7.3 Inputs

Segment Recognition record number 1.

3.2.7.4 Outputs

The normal Separability report.

3.2.7.5 Storage Requirements

Total space allocated in 588 bytes.

3.2.7.6 Description

SEPRPT is called by CAMREC after all separability data has been saved. SEPRPT then outputs the separability report which consists of channel combinations for up 4 (16 channels) and selection Battacharyya separability data.

3.2.7.7 Flowcharts

N/A

3.2.7.8 Listings

See Appendix A.

3.2.8 SOFTWARE COMPONENT NO. 8 (CALC)

This subroutine calculates data necessary for the output of the Bias Correction Classification and Bias Correction Cluster reports.

3.2.8.1 Linkage

CALC is called by DOTRAY.

3.2.8.2 Interface

N/A

3.2.8.3 Inputs

Dot Data from common blocks CBIAS.

3.2.8.4 Output

None

3.2.8.5 Storage

Total space allocated is 4031 bytes.

3.2.8.6 Description

CALC performs 2 identical calculations on different sets of data. The first data set involves Classification data and the second involves Cluster data. CALC first sets all needed variables to zero. If the flag CLADUM is equal to 1 no classification calculations are performed and CALC goes directly to the cluster calculations. Otherwise CALC checks "category 8 interest" for the presence of an "S", "N", or "W", and sets pointers and indication as necessary. Next certain arrays are zeroed.

Following is the variables calculated for each of 209 dots. 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.
- NTYP 2 - The number of type 2 DOTS
- NAIJ - The number of type 1 type 3 DOTS whose label and classification are the same.
- NGIJ - The number of type 1 and type 3 "Grain TYPE" DOTS whose label and classification are not the same.
- NOCL - The number of type 2 DOTS which are both labeled and classified.

In addition when calculating the above for all dots a bias correction vector table is calculated. This table consists of a two dimensional array and contains summations of dots that have valid labels as the first index and valid classification as the second index.

In addition the following variables are calculated: It is understood that classified does not mean "threshold".

- ALGT - The number of TYPE 2 dots labeled either "W" or "S" and classified either "W" or "S".
- ALBG - The number of TYPE 2 dots labeled with any "category used" and classified either "W" or "S".
- ALNT - The number of TYPE 2 dots labeled in any "category used" other than "W" or "S" and classified in any "category used" other than "W" or "S".
- ALNT - The number TYPE 2 dots labeled in any "category used" and classified in any "category used" other than "W" or "S".

The subroutines computes the corrected percentages, uncorrected populations, variances, uncorrected percentages and random

sample data for all "categories used" plus "grain category".
The equations for these calculations are shown in Attachment B.

CALC now performs the identical calculations on the cluster data
after checking CLUDUM as outlined above.

3.2.8.7 Flowchart

See Flow diagram No. 3.

3.2.8.8 Listing

See Appendix A.

3.2.9 SOFTWARE COMPONENT NO. 9 (BIAPRT)

This program outputs the report header sheet, Label/Classification table, Label/Cluster table and the Bias Correction reports.

3.2.9.1 Linkage

BIAPRT is called by CAMREC & DOTRAY. It calls BNT.

3.2.9.2 Interface

N/A

3.2.9.3 Inputs

Segment recognition records and common blocks CBIAS, Dummy and CLCOM.

3.2.9.4 Outputs

Report header sheet, TYPE1 and TYPE2 Dot Label/Classification report, Bias Correction Classification Report, TYPE1 and TYPE2 Dot Label/Cluster report and the Bias Correction Cluster reports.

3.2.9.5 Storage Requirements

Total space allocated in 3712 bytes.

3.2.9.6 Description

When CAMREC calls BIAPRT it sets PASS=1. This causes BIAPRT to output the report header which contains tape number, DPAR No., record ID, segment number and all acquisition dates. This information is retrieved from the segment recognition record which had been read into IBUF.

When DOTRAY calls BIAPRT it sets PASS=2. In this mode of operation up to 2 similar reports can be generated. If CLADUM is equal to 1 no classification report is generated.

Otherwise 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 DU or DO do not appear in either matrix.

Next the bias correction classification report is output using data calculated by CALC and stored in common blocks CBIAS. The report consists entries for all "categories of interest", DO, TH, UN and grain as follows:

Pixel population, classified and corrected percentages, variance and random sample estimate.

Also the alpha value matrix is output, and PCC values, DW and DS and lastly the Bias correction vectors and totals.

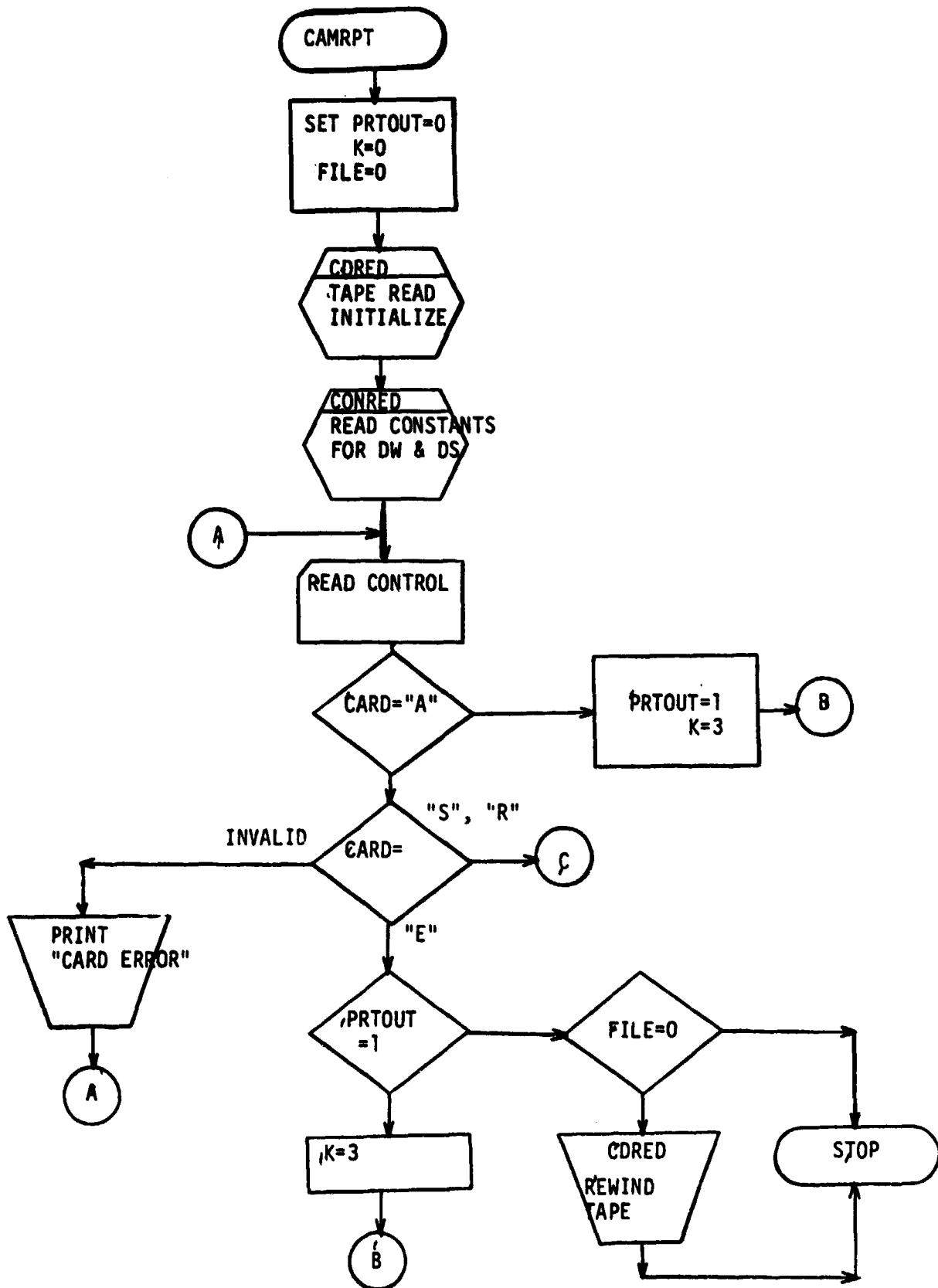
A similar report is generated for the cluster data if CLUDUM is not equal to 1.

3.2.9.7 Flowcharts

See flow diagram 4.

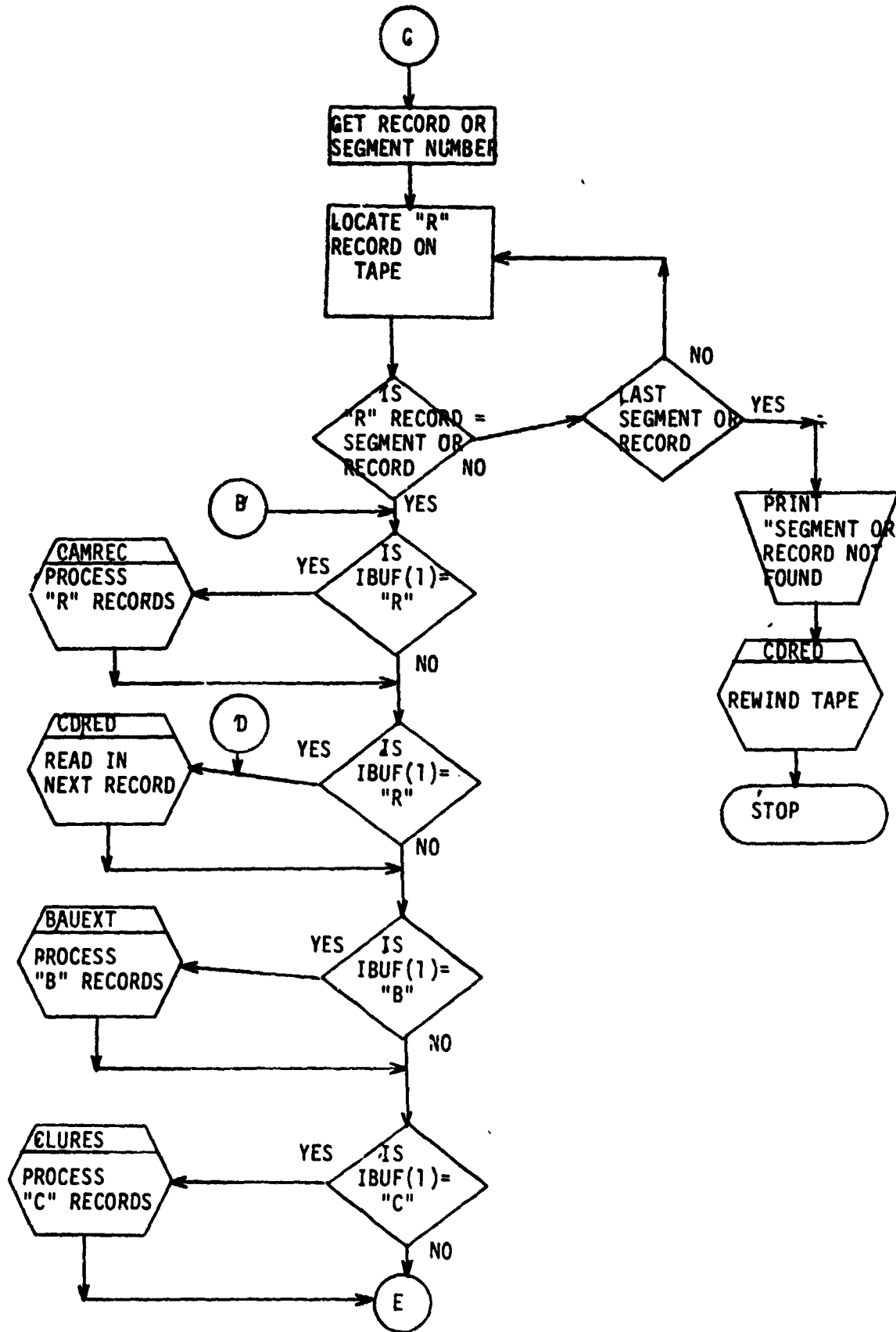
3.2.9.8 Listing

See Appendix A.



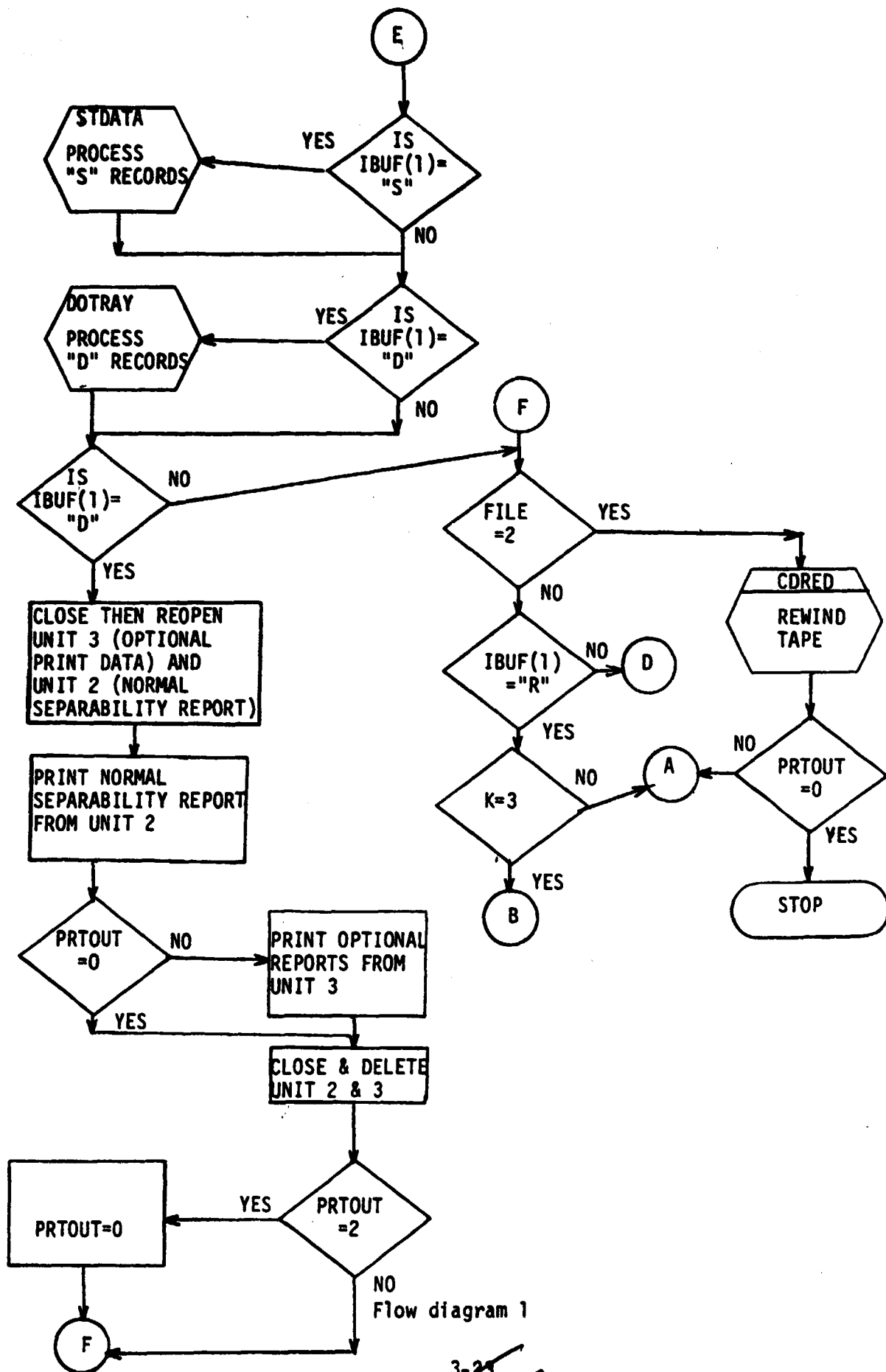
Flow diagram 1

3-21
23



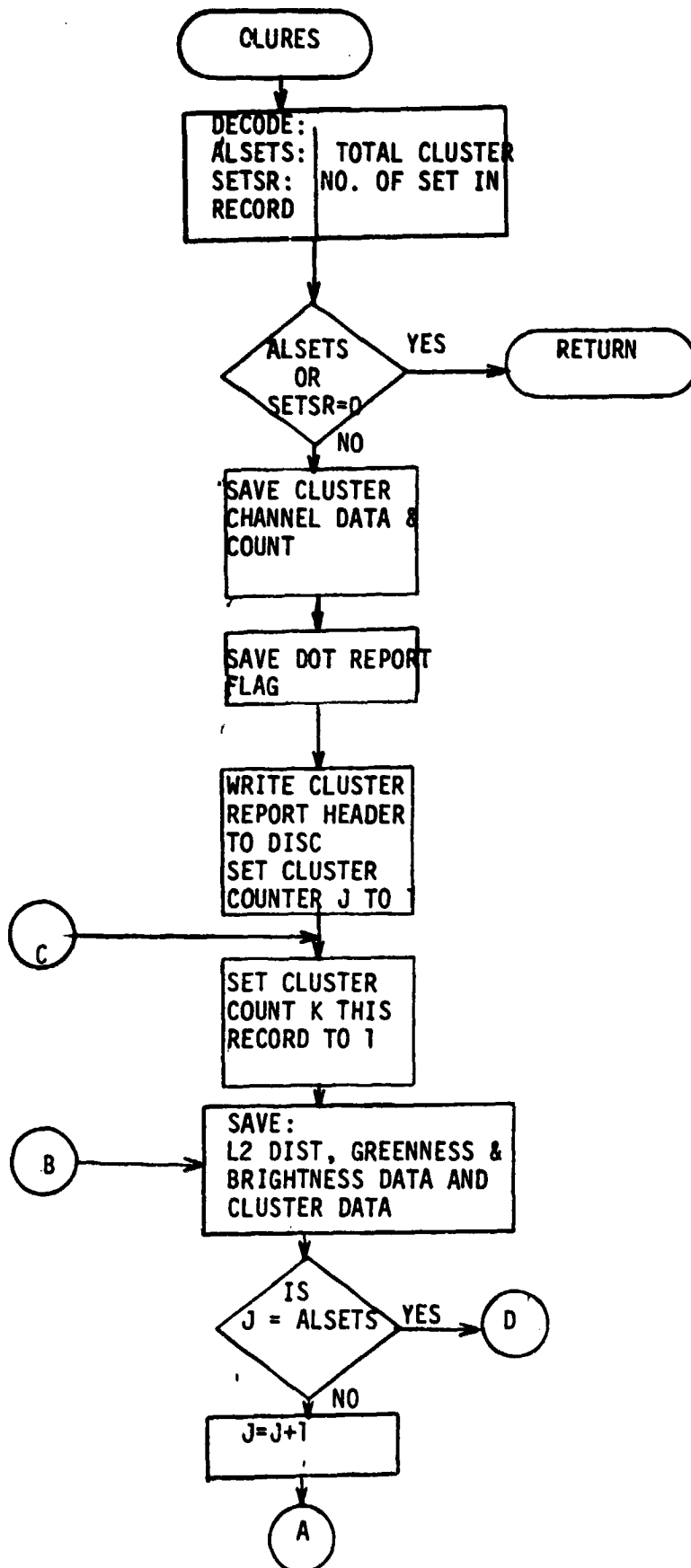
Flow diagram 1

3-2284



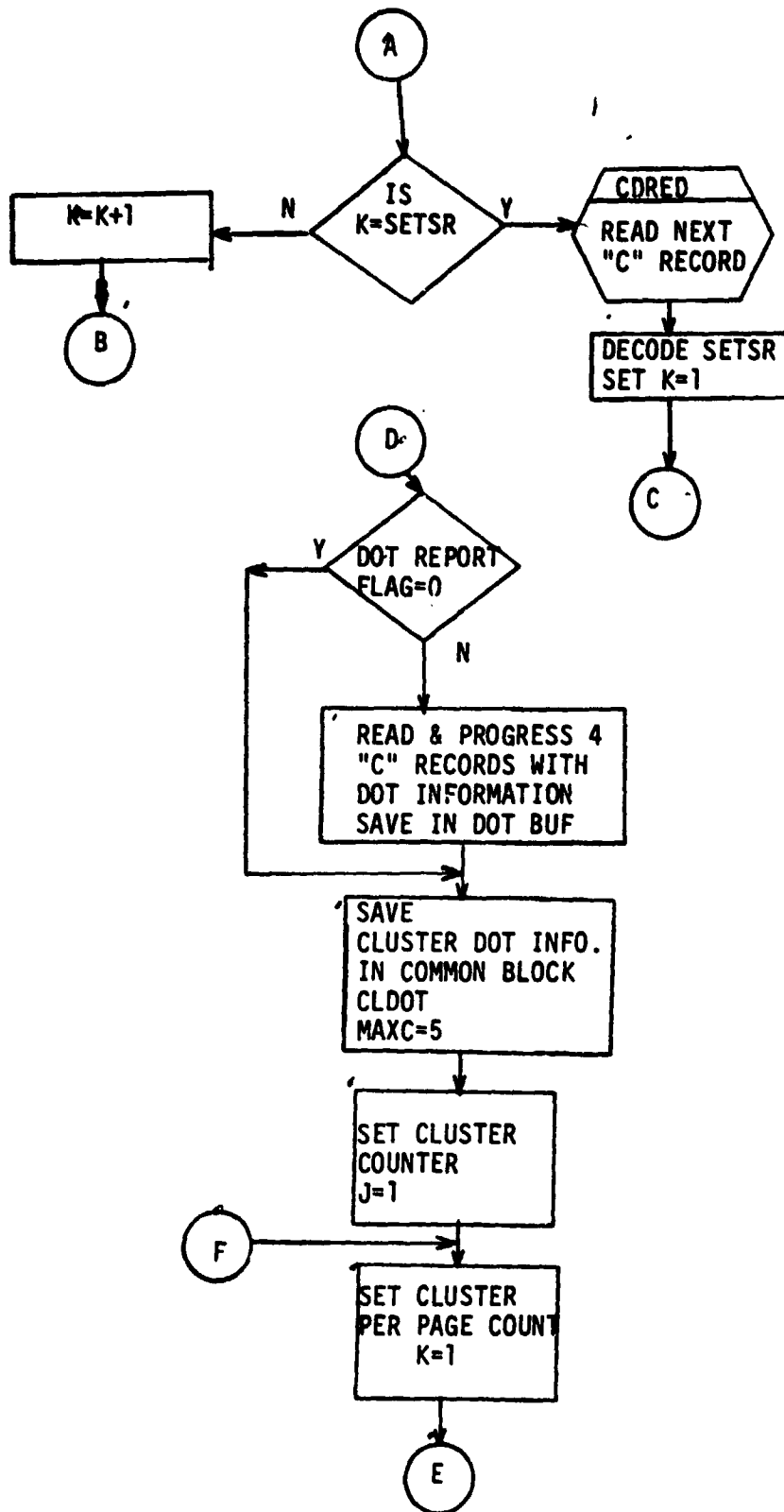
Flow diagram 1

3-25
25



Flow diagram 2

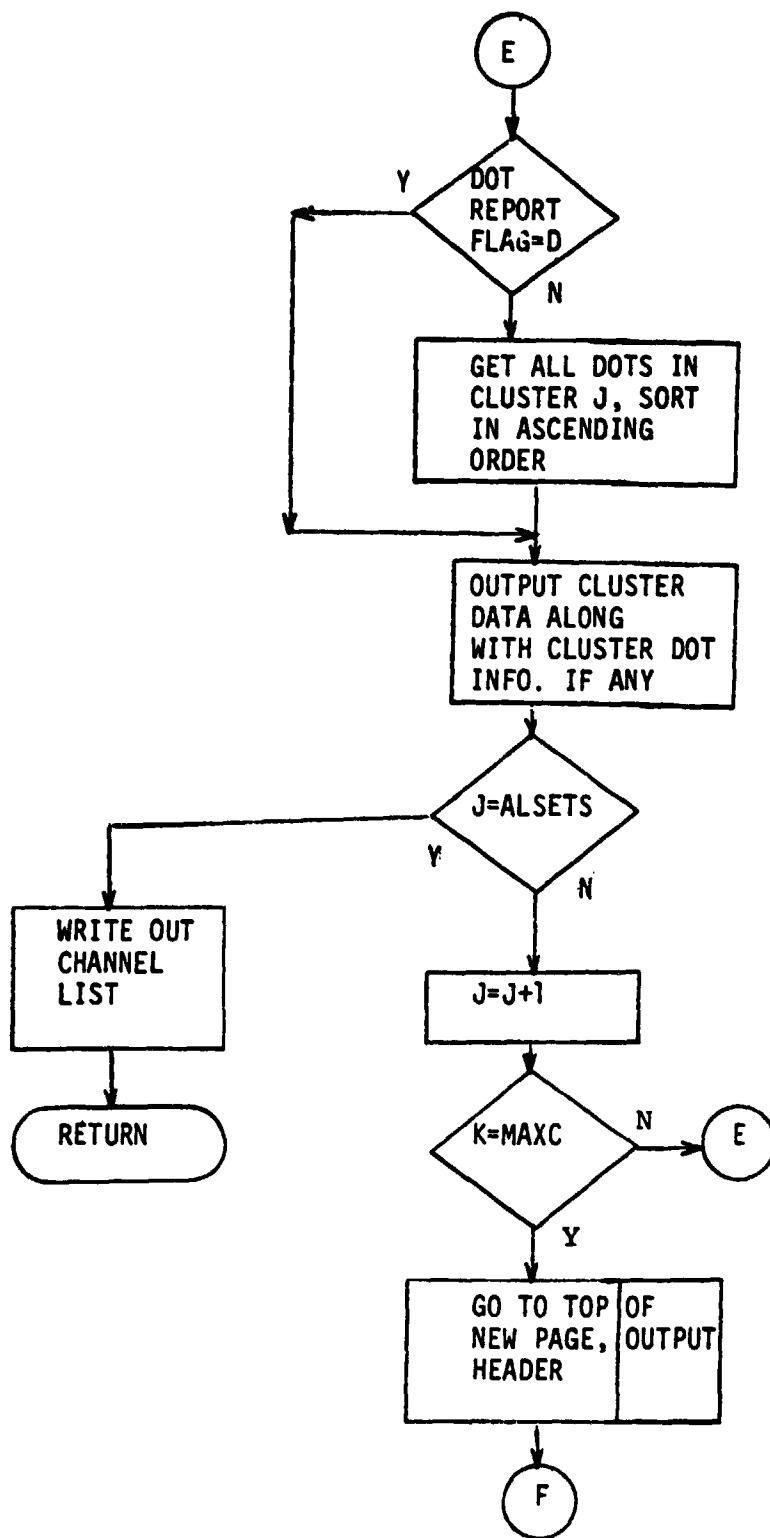
3-24-26



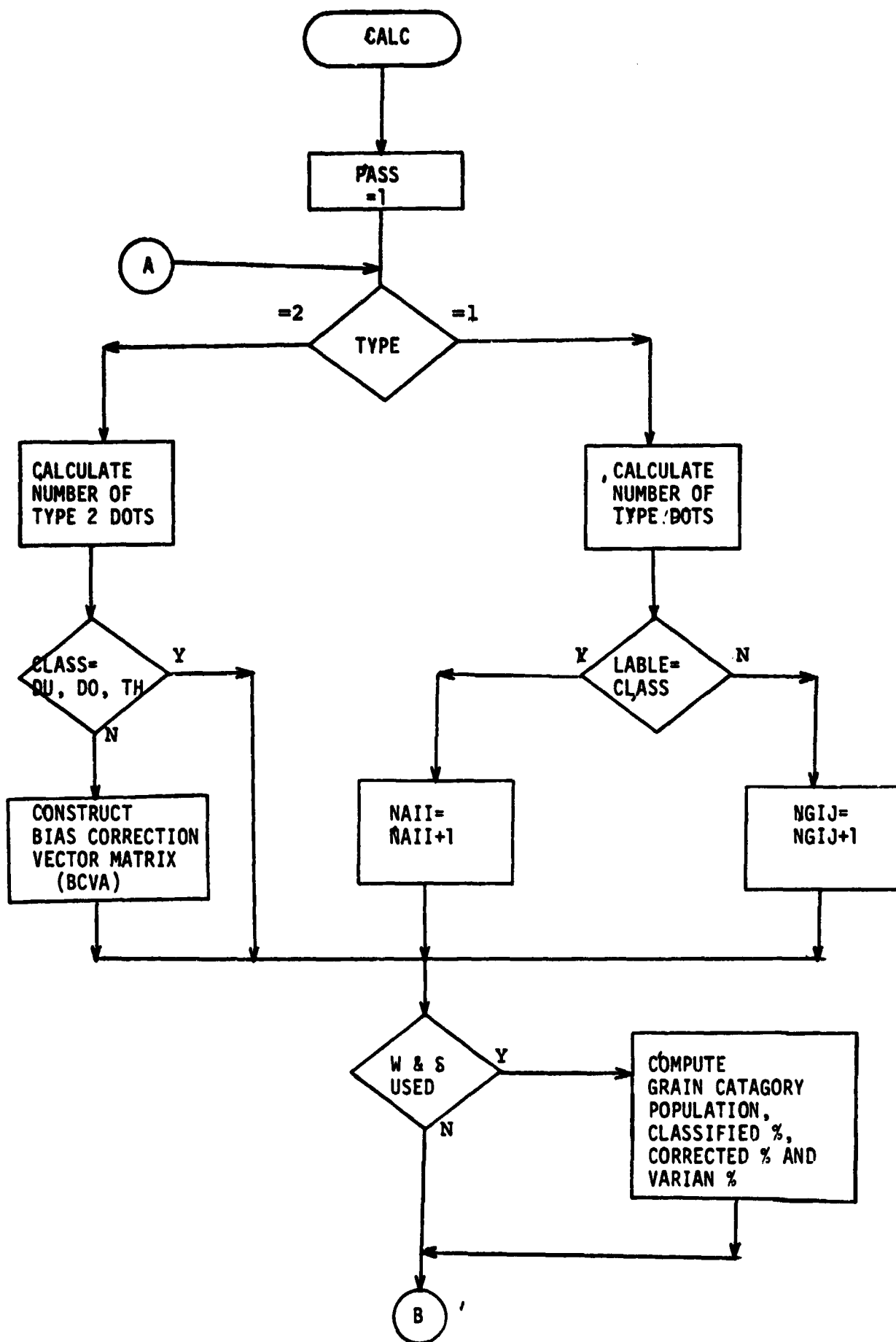
Flow diagram 2

3-25

27

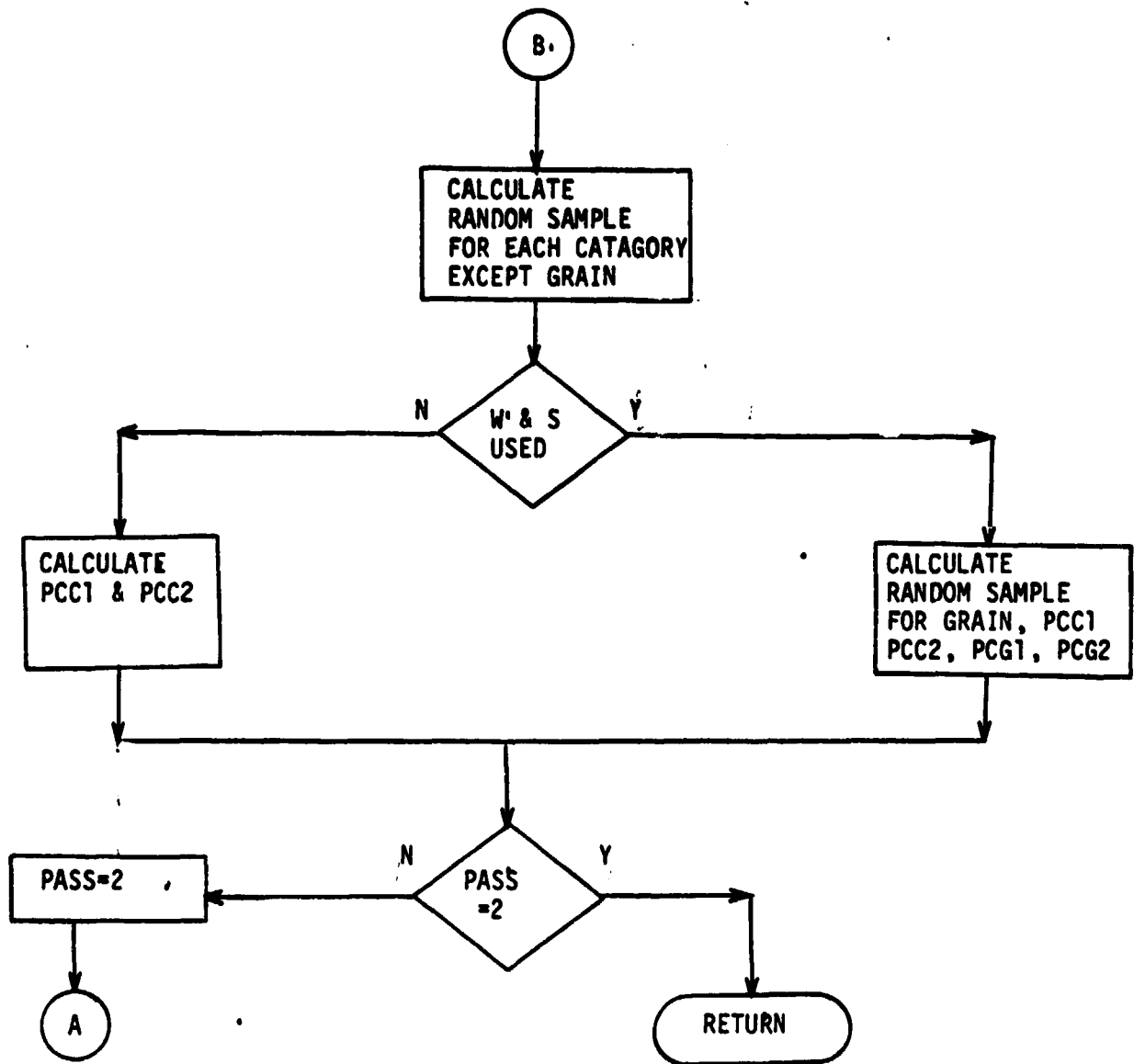


Flow diagram 2



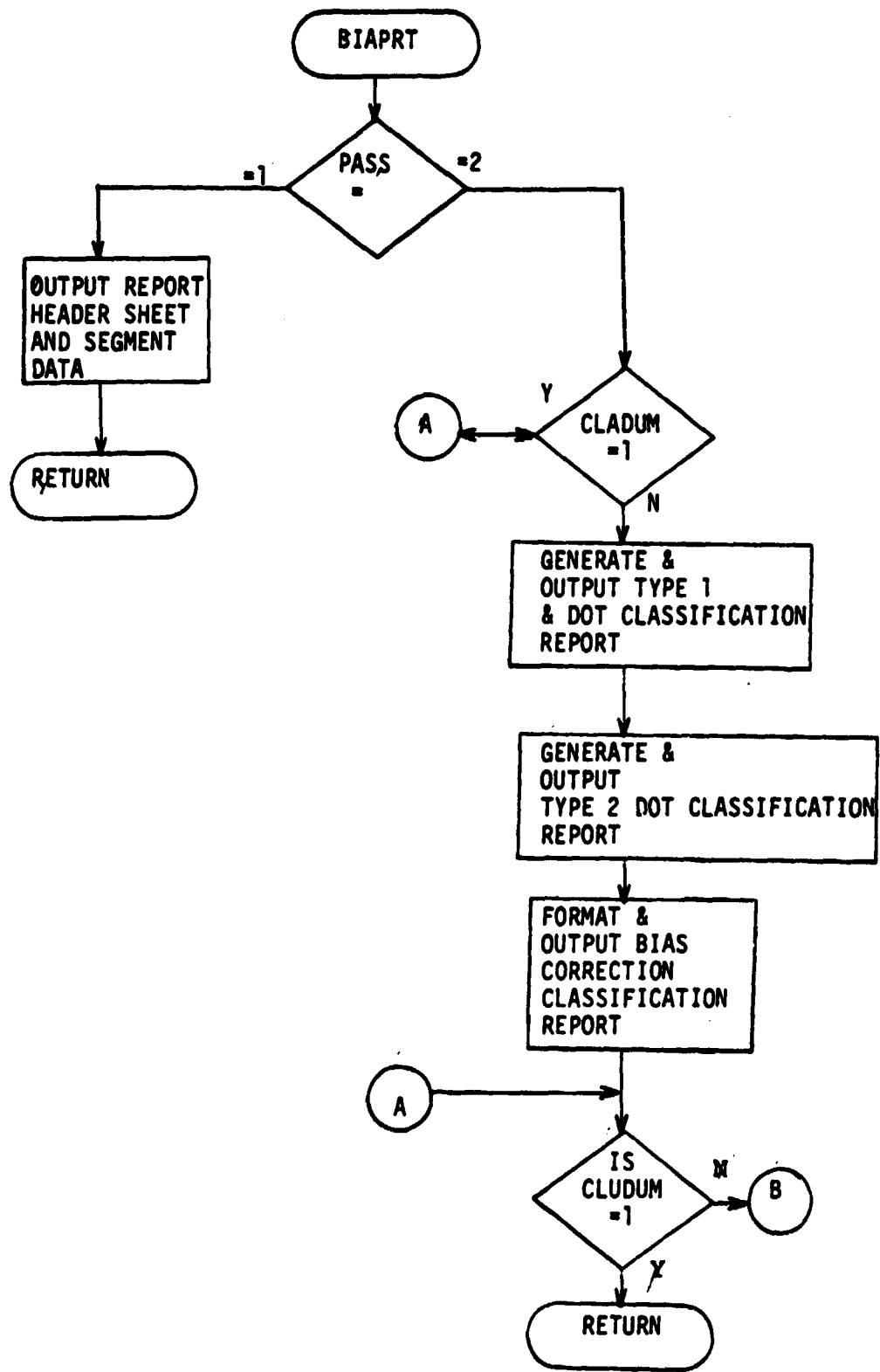
Flow diagram 3

3-27 29

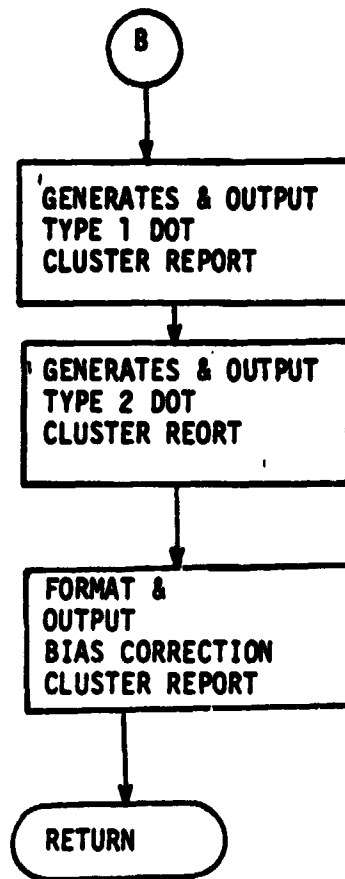


THE LOGIC FOR SECOND PASS IS THE SAME AS THE FLOW CHART ONLY THE VARIABLES ARE CHANGED.

Flow diagram 3



Flow diagram 4



Flow ddiagram 4

~~3-20~~
3-21

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

The first 22 data cards define the input tape drive, input tape unit and curve fitting constants for DW & DS calculations. The DS & DS formular are shown in Appendix B. The card formats are:

M or X (tape drive)

0 or 1 (tape unit)

10 DW curve fitting constant cards as described in Appendix C1

10 DS curve fitting constant cards as described in Appendix C2

Entries always start 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
10 DW constant cards
10 DS constant cards
END
$EOD
$MCR REM RSXBAT
$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 MTO:/CHA=[FOR] (CR)
```

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

~~42~~
34

APPENDIX A

```

CAPRPT,FTM /TR19LYCS/M3
0001 IMPLICIT INTEGR(A=2)
0002 BYTE PROF(120)
0003 LOGICALS1 IRUF(3000)
0004 LOGICALS1 CCHAR
0005 LOGICALS1 CD(80)
0006 LOGICALS2 CPYTR(4)
0007 LOGICALS1 SERG(4),RECORD(12)
0008 CPMCNV/SEG/SEGNE
0009 CMMN/PCNT/LINE,PAGE
0010 DATA CNTR/1HS,147.1MA,1HE/
0011 PRZOUT=0
0012 K=0
0013 R=0
0014 FILE=0
0015 TAPN=0
0016 CALL CDRED(IRUF,R,FILE)
0017 R=1
0018 CALL CDRED(IRUF,R,FILE)
0019 FPMAT(14)
C READ CONSTANTS FHM DW AND CM CALCULATIONS
0020
0021 15 CONTINUE
0022 R=1
0023 C TYPE 1000
C FPMAT(14),5X,'INPUT RUN TYPE >')
ACCEPT 99,CD
0024 READ(1,90,(ND=97),CD)
0025 FPMAT(8,11)
0026 DZ 24 15,172
0027 IF(CD(1).NE.' ') GO TO 21
0028 CONTINUE
0029 GO TO 9
0030 - 21 - CCHAR=CD(1)
0031 DW 1 K=1,4
0032 IF(CCHAR.EQ.CCHAR) GO TO 2
0033 CONTINUE
0034 22 CONTINUE
0035 WRITE(6,R0)
0036 88 FPMAT(11,10X,'PAD DATA CARD ')
0037 97 STOP
0038 18 FILE=0
0039 R=1
0040 CALL CDRED(IRUF,R,FILE)
0041 IF(PRTPUT.EQ.0) STOP
0042 GO TO 15
0043 CONTINUE
0044 2
IF(TAPN.EQ.0) DECDS(4,100,IRUF(44)) TAPN0
0045 50 T(31,31,34,35)K
0046 CONTINUE
0047 35
IF(PRTZUT.EQ.1) GO TO 5
0048 K=3
0049 GO TO 6
0050 5
IF(FILE.EQ.0) STOP
0051 R=1
0052 CALL CDRED(IRUF,R,FILE)
0053 STOP

```

ORIGINAL PAGE IS OF POOR QUALITY

36

```

CAMRPT,FTN /TP:BLZCKS/WR
0054 31 CONTINUE
0055 PRTOUT=1
0056 F=0
0057 I=1
0058 4 IF(CD(I).NE.' ') GZ TO 3
0059 I=I+1
0060 IF(I.GT.72) GZ TO 22
0061 GZ TO 4
0062 3 F=F+1
0063 IF(F.EQ.1) GZ TO 7
0064 IF(K.EQ.1) GZ TO 8
0065 IF(F.EQ.2) GZ TO 7
0066 GZ TO 8
0067 7 I=I+1
0068 IF(CD(I).EQ.' ') GZ TO 4
0069 IF(I.GT.72) GZ TO 22
0070 GZ TO 7
0071 8 CONTINUE
0072 IF(K.EQ.2) GZ TO 12
0073 JJ=1
0074 DZ 11 J=I,I+3
0075 SEGNO(JJ)=CD(J)
0076 JJ=JJ+1
0077 11 CONTINUE
0078 GZ TO 13
0079 12 CONTINUE
0080 JJ=1
0081 DZ 14 J=I,I+12
0082 DIF=J-I+1
0083 IF(DIF.EQ.7) GZ TO 14
0084 RECORD(JJ)=CD(J)
0085 JJ=JJ+1
0086 14 CONTINUE
0087 13 CONTINUE
0088 GZ TO 47
0089 41 CONTINUE
0090 CALL CDRED(IBUF,B,FILE)
0091 IF(FILE.LT.2) GZ TO 47
0092 WRITE(6,R9)
0093 89 FORMAT(1H0,10X,'SEGMENT NO. ON RECORD ID NOT FOUND')
0094 GZ TO 18
0095 47 CONTINUE
0096 R=R+1
0097 IF(IBUF(1).NE.'R') GZ TO 41
0098 IF(IBUF(2).GT.'1') GZ TO 41
0099 C IF(K.EQ.2) GZ TO 15
0100 DZ 43 I=1,4
0101 43 IF(SEGNO(I).NE.IBUF(147+I)) GZ TO 41
0102 SFLG=1
0103 GZ TO 46
0104 48 CALL CDRED(IBUF,B,FILE)
0105 34 CONTINUE
0106 PRTOUT=1
0107 6 R=R+1
0108 46 CONTINUE

```

ORIGINAL PAGE IS
OF POOR QUALITY

```

CAMRPT,FTN
0109 IF(IBUF(1).EQ.'R') CALL CAMREC(IBUF,R,FILE,TAPNO)
0110 IF(IBUF(1).NE.'R') GO TO 51
0111 49 CONTINUE
0112 50 CALL CDRED(IBUF,R,FILE)
0113 51 CONTINUE
0114 IF(IBUF(1).EQ.'B') CALL BAUEXT(IBUF,R,FILE)
0115 IF(IBUF(1).EQ.'C') CALL CLURES(IRUF,R,FILE)
0116 IF(IBUF(1).EQ.'S') CALL STDATA(IBUF,R,FILE,DPLG)
0117 IF(IBUF(1).EQ.'D') CALL D2TRAY(IRUF,R,FILE,PRTOUT)
0118 IF(IBUF(1).EQ.'O') GO TO 25
0119 19 IF(FILE.EQ.2) GO TO 19
0120 IF(IBUF(1).NE.'R') GO TO 49
0121 IF(K.NE.3) GO TO 15
0122 GO TO 46
0123 9 WRITE(6,98)
0124 98 FORMAT('1',10X,' BLANK DATA CARD ')
0125 GO TO 18
0126 25 CONTINUE
0127 27 CLOSE(UNIT=2,DISPOSE='SAVE')
0128 CLOSE(UNIT=3,DISPOSE='SAVE')
0129 OPEN(UNIT=3,TYPE='OLD',NAME='XXXX.DAT')
0130 OPEN(UNIT=2,TYPE='OLD',NAME='YYYY.DAT')
0131 23 READ(2,500,END=24) PBUF
0132 WRITE(6,500) PBUF
0133 GO TO 23
0134 24 IF(PRTOUT.FO.0) GO TO 10
0135 26 READ(3,500,END=10) PBUF
0136 WRITE(6,500) PBUF
0137 500 FORMAT(120A1)
0138 GO TO 26
0139 10 CLOSE(UNIT=3,DISPOSE='DELETE')
0140 CLOSE(UNIT=2,DISPOSE='DELETE')
0141 IF(PRTOUT.FO.2) PRTOUT=0
0142 GO TO 19
0143 END
    
```

ORIGINAL PAGE IS
OF POOR QUALITY

PROGRAM SECTIONS

ATTRIBUTES

NUMBER	NAME	SIZE	ATTRIBUTES
1	SCROCI	00272	605
2	SPDATA	00034	15
3	SDATA	00394	114
4	SVARS	00334	1651
5	STEMPS	00002	1
6	SEC	00014	2
7	PCPT	00004	2

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
CHAR	L01	4-006354	DIFS	L02	4-006340	F	L02	4-006312
	L02	4-006330	J	L02	4-006334	K	L02	4-006320
PAGE	L02	7-000002	PRTOUT	L02	4-006322	SFL1	L02	4-006312
				L02	4-006316	R	L02	4-006326

ARRAYS

DIMENSIONS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
CD	L01	4-006125	00120	40 (60)
CMTR	L01	4-006275	00204	2 (4)
ISUF	L01	4-000170	00764	1530 (3060)
PROF	L01	4-000000	000170	60 (120)
RECORD	L01	4-006301	000014	6 (12)
SEGE	L01	6-000000	000004	2 (4)

39

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
1	**	2	1-000426	3	1-000675	4	1-000434
6	1-001440	7	1-000742	8	1-001004	9	1-001714
11	**	12	1-001102	13	1-001234	14	1-001212
19	1-000360	20	1-001144	21	**	22	1-000232
23	1-002014	24	1-002112	25	1-001744	26	1-002124
31	1-000610	34	1-001824	35	1-000512	41	1-001244
46	1-001452	47	1-001316	48	**	49	1-001510
51	1-001526	60	3-000010	69	3-000040	97	3-000344
99	3-000004	100	3-000006	503	3-000144	1000	**

ORIGINAL PAGE IS
 OF POOR QUALITY

FUNCTIONS AND SUBROUTINES REFERENCED

BAUXT	CANREC	CODED	CLASS	CIURES	CZARED	DPTRAY	OPENUS	STDATA

TOTAL SPACE ALLOCATED = 011254 2300

F0RTRAN IV-PLUS V02-51
CAMRPT.FTN /TRIBL0CKS/HR
NO FPP INSTRUCTIONS GENERATED

08102102

09-MAR-78

PAGE 5

CAMRPT.LPI=CAMRPT

```

CAMREC.FTN /TRIBLZCKS/W9
0001 SUBROUTINE CAMREC(IGUF,R,FILE,TAP,N0)
0002 IMPLICIT INTEGER(I=2)
0003 LOGICAL*1 IBUF(1),TAPSEP(6),SCATA(112)
0004 INCLUDE 'CAM*01.CAT'
0005 BYTE CLASSY(419),CLZ(21416)
0006 LOGICAL*1 TYPE(209),LBLEP(209)
0007 LOGICAL*1 RELABEL(6),ACLAB(6),BLCLAB(6),BULAPL(6)
0008 REAL PCC1,PCC2,PCC3,PCC4,PCC5,PCL1,PCL2,PCL3,PCL4,PCL5,PCL6
0009 REAL CLAD,CLUD,TCNS,ACNS,CCNS
0010 DIMENSION CLAD(2),CLUD(2),TCNS(2,8),CCNS(2),ACNS(2,8)
0011 DIMENSION ACVAP(8),PCV(8,8)
0012 DIMENSION RAPP(12),RUPP(12)
0013 COMMON/CPAS/PA,P,SB,P,EA,NCC,QUINCC,BLOC,R,BUC,R,BACAIN,BUCAIN,
    * BULAPL,BALANL,CLASY,TYPE,LBLED,BACATS,BACLAB,
    * PCC1,PCC2,PCC3,PCC4,PCL1,PCL2,PCL3,PCL4,PCL5,PCL6,
    * ACVA,SCV,ARAT,BLBT,CLAD,CLUD,TCNS,ACNS,CCNS
0014 * COMMON/DUMMY/CLADUM,CLUDUM
0015 * COMMON/CLCPM/CLCPM
0016 REAL A(3,2)
0017 DIMENSION RAINC(12),BLUNCR(12),CACOR(12),RUCER(12)
0018 REAL BACAIN(8),BUCAIN(6,8),BAVAR(12),HUVAR(12),HARANS(12),HURANS(12)
0019 COMMON/PCNT/LINE,PAGE
0020 COMMON/SEG/SEG%

0021 LOGICAL*1 CLIST(240),CLASS(4),CATLAB(6)
0022 BYTE PR(132),BLANK,PTLABEL(5)
0023 LOGICAL*1 SEM(4)
0024 DIMENSION CI(15)
0025 INTEGER NI(5)
0026 INTEGER NI(40)
0027 REAL P(60),T,X,Y,FCM,PM
0028 REAL PDU,PM,PTH,PX,PTH
0029 DATA BLANK/1H /
0030 DATA PT/1H, /
0031 D(1) 1=1,132
0032 P(1)=BLANK
0033 PASS=1
0034 CALL RIAPRT(IGUF,TAPN,PASS)
0035 OPEN(UNIT=3,TYPE='TEXT',NAME='XXX.DAT')
0036 OPEN(UNIT=2,TYPE='TEXT',NAME='YYY.DAT')
C SAVE DATA FOR SEPARABILITY REPORT
TCHANG=0
0037
0038 DE 7 J=64,99
0039 IF(IGUF(J).EQ.1) TCHAN=TCHAN*1
0040 CONTINUE
DE 8 J=1,8
0042 TAPSEP(J)=IBUF(99+J)
0043 CONTINUE
0044 DECODE(4,10,IBUF(54)) NPSUF
0045 IF(LNSUR.EQ.0) GOTO 41
0046 FOR=AT(14)
0047 LINE=70
0048 CALL BNT
0049 WRITE(3,99)
0050 FORMAT('0,2EX,'CLASSIFICATION: DATA')

```


0051

WRITE(3,90)

0052

94

FORMAT(2H0)

0053

WRITE(3,98)

0054

98

FORMAT(1H ,53X,'NORMALIZED')

0055

II=0

0056

DO 95 JJ=1,16

0057

IF(1BUF(R3+JJ).EQ.'1') GO TO 96

0058

GO TO 95

ORIGINAL PAGE IS
OF POOR QUALITY

0059

96

II=II+1

0060

CT(II)=JJ

0061

95

CONTINUE

0062

IIMAX=II

0063

WRITE(3,97)

0064

97

FORMAT(1H ,15X,' CLUSTER NAME THRESHOLD VALUE ',

'APRIORY VALUE ')

0065

CALL RNT

0066

WRITE(3,90)

0067

90

FORMAT(' ')

0068

LINE=LINE+5

0069

K=0

0070

J=1

0071

N=1

0072

RJ=249

0073

POP=0

0074

SETSR=14

0075

LFLD=32

0076

WIT=0

0077

XC=0

0078

TC=0

0079

DO 3 II=1,60

0080

3

WI(II)=0

0081

DECODE(5,101,1BUF(64)) DU

0082

101

FORMAT(15)

0083

DECODE(5,101,1BUF(69)) DU

0084

GO TO 2

0085

1

CALL CORED(1BUF,K,FILE)

0086

RJ=3

0087

SETSR=22

0088

N=1

0089

2

CONTINUE

0090

DECODE(5,102,1BUF(RJ+27)) DUM1

0091

POP=POP+DUM1

0092

CALL CRIP2(1BUF(RJ),CLASS,PI,PE)

0093

CALL MV(1BUF(RJ),PB(28),6)

0094

CALL MV(1BUF(RJ+15),PB(46),3)

0095

PB(49)=PT

0096

PR(50)=1BUF(RJ+18)

0097

CALL MV(1BUF(RJ+19),PB(66),3)

0098

WRITE(3,111) (PR(RJ),PJ=27,79)

0099

CALL RNT

0100

111

FORMAT(20X,60A1)

0101

TC=TC+PR

0102

IF(CLASS(1).EQ.'X') GO TO 10

0103

IF(CLASS(1).EQ.'1') GO TO 20

0104

GO TO 30

0105

10

XC=XC+PI

CAMREC,FTN

/TR:BLOCKS/WR

```
0106 30 IF(J.EQ.NOSUB) GO TO 40
0107     J=J+1
0108     IF(N.EQ.SETSR) GO TO 1
0109     N=N+1
0110     RJ=RJ+LFLN
0111     GO TO 2
0112 20 IF(K.EQ.0) GO TO 5
0113     LK=4*(K-1)
0114     DO 4 II=1,4
0115     IF(CLASS(II).NE.CLIST(LK+II)) GO TO 5
0116 4 CONTINUE
0117     WI(K)=WI(K)+PI
0118     WIT=WIT+PP*PI
0119     GO TO 30
0120 5 K=K+1
0121     LK=4*(K-1)
0122     DO 6 II=1,4
0123     6 CLIST(LK+II)=CLASS(II)
0124     WI(K)=WI(K)+PI
0125     WIT=WIT+PP*PI
0126     GO TO 30
0127 40 CONTINUE
0128     CALL RNT
0129     CALL RNT
0130     WRITE(3,93) (CT(II),II=1,IIMAX)
0131 93 FORMAT(25X,'CHANNELS USED: ',16(1X,12))
0132 102 FORMAT(15)
0133 41 BAPP(10)=PP
0134     T1=PP
0135     T1=(T1/22932.)*100.
0136     BAJNC(10)=T1
0137     DO 35 II=1,112
0138     SDATA(II)=IBUF(II+596)
0139 35 CONTINUE
0140     CALL CORED(IBUF,P,FILE)
0141     IF(IBUF(1).NE.'R') GO TO 50
0142     DO 45 II=1,112
0143     SDATA(II)=IBUF(II+596)
0144 45 CONTINUE
0145 50 CALL SEPRPT(TCHAP,TOTSEP,SDATA)
0146     RETURN
0147     END
```

ORIGINAL PAGE IS
OF POOR QUALITY

CARREC.FTN /TRIPBLOCKS/HR
PROGRAM SECTIONS

NUMBER	NAME	SIZE	ATTRIBUTES
1	SCZDF1	002462	645 R01,CEA,LCL
2	SPDATA	000040	16 R01,CEA,LCL
3	SIDATA	000400	128 R01,CEA,LCL
4	SVARS	001744	496 R01,CEA,LCL
5	STEMPS	000004	2 R01,CEA,LCL
6	GBIAS	005122	1321 R01,CEA,LCL
7	DUMMY	000004	2 R01,CEA,LCL
8	CLCEM	020642	709 R01,CEA,LCL
9	PCNT	000004	2 R01,CEA,LCL
10	SEC	000004	2 R01,CEA,LCL

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
CARREC		1-000000						

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
BACATS	102	6-003314	BAT3T	102	6-003766	BLANK	101	4-000770
CAADUM	102	7-000070	CLUDUM	102	7-001002	DA	102	6-001550
EJLE	102	F-000006	I	102	4-001472	IL	102	4-001704
JJ	102	4-001706	K	102	4-001712	LELD	102	9-001774
N	102	4-001714	KMSUB	102	4-001702	PAGE	102	9-000002
PCCG2	104	6-003342	PCC1	104	6-003326	PCC2	104	6-003332
PCL1	104	6-003346	PCL2	104	6-003352	PCW	104	4-001642
PI	102	4-001734	PJ	102	4-001740	PWH	104	4-001642
PT	101	4-001771	PTH	104	4-001656	PIWM	104	4-001666
RJ	102	4-001716	SETSR	102	4-001722	TAPNB	102	F-000010
T1	104	4-001622	W1Y	102	4-001726	X	104	4-001626

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
BACAIN	104	6-001520	000030	12 (3,2)
BACLAB	101	6-003316	000010	4 (2)
BACER	102	6-000140	000030	12 (12)
BALABL	101	6-001609	000010	4 (2)
BAPP	102	6-000000	000030	12 (12)
BARPS	104	6-001360	000050	24 (12)
BAUGA	102	6-000050	000030	12 (12)
BAVAR	104	6-001220	000060	24 (12)
BCVA	102	6-003366	000200	64 (8,8)
BCVU	102	6-003566	000200	64 (8,8)
BUCAIN	104	6-000620	000400	128 (8,8)

CAMREC.PTH	JFK12126<5/4R	4	(8)
BUCLAM L01	6-001560 000010	4	(8)
BUCLAM L02	6-000170 000010	12	(12)
BULABL L01	6-001570 000010	4	(8)
BUPRP L02	6-000030 000010	12	(12)
BURANS R04	6-501440 000060	24	(12)
BUMNCP L02	6-000110 000030	12	(12)
BUMNCP L04	6-001300 000040	24	(12)
CATLAB L01	4-001554 000010	4	(8)
CC34S R04	6-001512 000010	4	(8)
CLAD R04	6-002772 000010	4	(8)
CLASSY L01	6-001610 000042	20	(416)
CLASS L01	4-001550 000014	2	(4)
CLDRT L01	6-000000 000042	20	(416)
CLIST L01	4-000170 000040	120	(240)
CLUD R04	6-000002 000010	4	(8)
CT L02	4-001000 000040	16	(16)
ISUF L01	F-000002 000001	0	(1)
LABEL L01	4-000772 000015	2	(5)
LOLED L01	6-002773 000021	104	(209)
NI L02	4-001040 000012	5	(5)
PH L01	4-000564 000214	46	(132)
PH R04	4-001242 000340	120	(80)
RCENS R04	6-004112 000000	256	(2,560)
SDATA L01	4-000010 000150	56	(112)
SEGM L01	10-000000 000014	2	(4)
TCMS R04	6-004012 000100	32	(2,0)
TRYSRP L01	4-001000 000010	4	(8)
TYPE L01	6-002452 000321	104	(205)
MI L02	4-001052 000170	60	(60)

LABELS	LABEL ADDRESS	LABEL ADDRESS	LABEL ADDRESS	LABEL ADDRESS	LABEL ADDRESS	LABEL ADDRESS	
1	1-001064	2	1-001152	3	1-001250	4	1-001350
20	1-001610	30	1-001536	40	1-001460	50	1-001380
45	1-000930	50	1-002436	60	1-00174	70	1-001660
100	3-000000	101	3-000164	102	3-000232	103	3-000300

FUNCTIONS AND SUBROUTINES REFERENCED

BIAPRT 9MT CORED CQIP0 MV OPENS SEPRPT

TOTAL SPACE ALLOCATED = 013072 2845

CAMREC.LPT=CAMREC

BAUEXT,FTN

/TR:BLOCKS/WR

0053

CLUFLG=0

0054

CLAEFG=1

0055

GO TO 2

0056

1

CLUFLG=1

0057

2

DECODE(2,100,IRUF(3)) SEQNO

0058

IF(SEQNO.GT.1) GO TO 7

0059

DO 3 I=5,30

0060

IF(IRUF(I).NE.' ') GO TO 5

0061

3

CONTINUE

0062

IF(CLUFLG.EQ.1) GO TO 4

0063

CLADUM=1

0064

RETURN

0065

4

CLUDUM=1

0066

CALL CDHED(IRUF,C,FILE)

0067

GO TO 6

0068

5

DECODE(2,100,IRUF(31)) TEMP2

0069

IF(CLUFLG.EQ.1) GO TO 22

0070

BACATS=TEMP2

0071

DO 21 I=1,BACATS

0072

BACLAR(I)=IRUF(I+4)

0073

21

CONTINUE

0074

GO TO 24

0075

22

BUCATS=TEMP2

0076

DO 23 I=1,BUCATS

0077

BUCLAR(I)=IRUF(I+4)

0078

23

CONTINUE

0079

24

IF(SEQNO.EQ.0) CPR=TEMP2

0080

7

CAT=CAT+1

0081

IF(CLUFLG.NE.1) GO TO 19

0082

MULAR(CAT)=IRUF(RJ)

0083

GO TO 20

0084

19

BALAR(CAT)=IRUF(RJ)

0085

20

CONTINUE

0086

DECODE(5,101,IRUF(RJ+1)) CUP1

0087

IF(CLUFLG.NE.1) GO TO 8

0088

RUPAR(CAT)=DUM1

0089

GO TO 9

0090

8

RUPAR(CAT)=DUM1

0091

9

DECODE(3,102,IRUF(RJ+6)) CUP1

0092

IF(CLUFLG.NE.1) GO TO 10

0093

RUMAR(CAT)=DUM1

0094

GO TO 11

0095

10

RUMAR(CAT)=DUM1

0096

11

DECODE(3,102,IRUF(RJ+9)) CUP1

0097

IF(CLUFLG.NE.1) GO TO 12

0098

RUCAR(CAT)=DUM1

0099

GO TO 13

0100

12

RUCAR(CAT)=DUM1

0101

13

DECODE(4,103,IRUF(RJ+12)) CUP1

0102

TEMP1=DUM1

0103

TEMP1=TEMP1/100.

0104

IF(CLUFLG.NE.1) GO TO 25

0105

BUVAR(CAT)=TEMP1

0106

GO TO 26

0107

25

BAVAR(CAT)=TEMP1

0108

24

CJRJ=26

ORIGINAL PAGE IS
OF POOR QUALITYORIGINAL PAGE IS
OF POOR QUALITY

```

BAUEXT,FTN /TRIRLOCKS/WR
0109 02 14 I=1,A
0110 DECIDE(3,102,IPUF(CJ)) DUM1
0111 TEMP1=DUM1
0112 ATR=1
0113 IF(TEMP1.FU.0) GO TO 17
0114 TEMP1=TEMP1/1000
0115 17 IF(CLUFLG.NE.1) GO TO 35
0116 RUCAT1(CAT,1)=TEMP1
0117 GO TO 10
0118 15 BACAIN(CAT,1)=TEMP1
0119 16 CJ=CJ+3
0120 10 CONTINUE
0121 IF(CAT.EQ.TEMP2) GO TO 30
0122 IF(CTR.EQ.CPD) GO TO 18
0123 CTR=CTR+1
0124 RJEHV=LELD
0125 GO TO 7
0126 10 CALL CDRED(IHUF,P,FILE)
0127 GO TO 6
0128 30 CAT=0
0129 DECIDE(5,101,IPUF(670)) T2
0130 DECIDE(5,101,IPUF(679)) T1
0131 DECIDE(3,102,IPUF(675)) T4
0132 DECIDE(3,102,IPUF(664)) T3
0133 IF(CLUFLG.NE.1) GO TO 31
0134 RUPAP(9)=T1
0135 RUPAP(11)=T2
0136 RUPAP(9)=T3
0137 RUPAP(11)=T4
0138 GO TO 10
0139 31 RUPAP(9)=T1
0140 RUPAP(11)=T2
0141 RUPAP(9)=T3
0142 RUPAP(11)=T4
0143 CALL CDRED(IHUF,P,FILE)
0144 RETURN
0145 END
    
```

ORIGINAL PAGE IS
OF POOR QUALITY

BAUEXT/PTN	TRIPLE/LOCKS/HP	ADDRESS	ADDRESS	ADDRESS	ADDRESS	ADDRESS	ADDRESS
CC3MS R04	6-005112	00010	4	(2)			
CLAD R04	6-003772	00010	4	(2)			
CLASFY L01	6-001610	00042	209	(41E)			
CLD2Y L01	6-000000	00042	209	(41E)			
CLUD R04	6-004002	00010	4	(2)			
LAUF L01	6-000002	00000	0	(1)			
LBLED L01	6-002773	00021	104	(209)			
RCANS R04	6-004112	00100	255	(2,6,8)			
TCM'S R04	6-004012	00130	32	(2,6)			
TYPE L01	6-002452	00021	104	(209)			

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
1	1-000366	2	1-000402	3	1-000554	4	1-000630
6	1-000306	7	1-001062	8	1-001254	9	1-001400
11	1-001422	12	1-001524	13	1-001546	14	1-002144
16	1-002204	17	1-002072	18	1-002272	19	1-001124
21	1-001730	22	1-000762	23	1-001040	24	1-001700
26	1-000004	27	1-002342	28	1-002612	29	1-001200
101	3-000004	102	3-000010	103	3-000016		

FUNCTIONS AND SUBROUTINES REFERENCED

CORED

TOTAL SPACE ALLOCATED = 011012 2309

BAUEXT/PTN=BAUEXT

67

CLURES,FTN /TP:RLCKS/KR

```

0050 200  FORMAT('0',1AX,'OPTIONS USED')
0051      IF(IBUF(704),EQ,'1') WRITE(3,1000)
0052      IF(IBUF(705),EQ,'1') WRITE(3,1003)
0053      IF(IBUF(706),EQ,'1') WRITE(3,1001)
0054      IF(IBUF(707),EQ,'1') WRITE(3,1002)
0055 1000  FORMAT(' ',20X,'SUN ANGLE NORMALIZATION ')
0056 1001  FORMAT(' ',20X,'DC EXCLUSION')
0057 1002  FORMAT(' ',20X,'EN EXCLUSION')
0058 1003  FORMAT(' ',20X,'MIN/MAX RADIANCE EXCLUSION')
0059      J=1
0060      RJ=24
0061      LFLD=45
0062      DFLD=17
0063 19    K=RJ
0064      PT=K
0065      DO 20 I=1,SETS
0066      DECIDE(2,102,IBUF(K+4)) NUM
0067      CNAME(NUM)=IBUF(K)
0068      DO 21 JJ=1,6
0069      CNAME(JJ,J)=IBUF(K)
0070      SCLASS(JJ,J)=IBUF(K+6)
0071      K=K+1
0072 21    CONTINUE
0073      DECIDE(5,105,IBUF(PT+12)) L2DIST(J)
0074      DECIDE(3,204,IBUF(PT+17)) A1E(J)
0075      DECIDE(4,202,IBUF(PT+20)) A1G(J)
0076      DECIDE(3,204,IBUF(PT+24)) A2E(J)
0077      DECIDE(4,202,IBUF(PT+27)) A2G(J)
0078      DECIDE(3,204,IBUF(PT+31)) A3E(J)
0079      DECIDE(4,202,IBUF(PT+34)) A3G(J)
0080      DECIDE(3,204,IBUF(PT+38)) A4E(J)
0081      DECIDE(4,202,IBUF(PT+41)) A4G(J)
0082      J=J+1
0083      PT=PT+1,FLD
0084      K=PT
0085 20    CONTINUE
0086 204   FORMAT(I3)
0087      IF(J.GT.ALSETS) GO TO 30
0088      CALL CORREC(IBUF,K,FILE)
0089      DECIDE(2,102,IBUF(6)) SETS
0090      RJ=8
0091      GO TO 19
C SAVE THE DAT DATA
C
0092 30    IF(DRPT,EQ,0) GO TO 43
0093      K=1
0094      DO 40 I=1,3
0095      CALL CORREC(IBUF,K,FILE)
0096      DO 39 JJ=4,499
0097      DSTRUF(K)=IBUF(JJ)
0098      K=K+1
0099 39    CONTINUE
0100 40    CONTINUE
0101      CALL CORREC(IBUF,K,FILE)
0102      DO 41 JJ=4,423
0103      DSTRUF(K)=IBUF(JJ)

```

ORIGINAL PAGE IS
OF POOR QUALITY

CLURES,FTN /TR:BLCKS/WR

0104 K=K+1

0105 41 CONTINUE

0106 43 MAXC=4

C SAVE THE CLUSTER ASSIGNMENT FOR LOT REPORT

0107 IJ=1

0108 D= 45 II=1,209

0109 NDX=(II-1)*DFLD+1

0110 IF(D%TRUF(NDX),EQ,'0') GO TO 47

0111 IF(D%TRUF(NDX),EQ,'1') GO TO 47

0112 IF(D%TRUF(NDX),EQ,'2') GO TO 47

0113 DECODE(2,102,D%TRUF(NDX)) NUM

0114 CLDCT(IJ)=CDNAME(NUM)

0115 CLDCT(IJ+1)=BLANK

0116 GO TO 46

0117 47 CLDCT(IJ)=D%TRUF(NDX)

0118 CLDCT(IJ+1)=D%TRUF(NDX+1)

0119 46 IJ=IJ+2

0120 45 CONTINUE

0121 J=1

0122 1 CONTINUE

0123 K=1

0124 2 CONTINUE

C

C PROCESS A CLUSTER DATA SET

C

0125 IF(DRPT,EQ,0) GO TO 44

C PROCESS THE LOT REPORT HERE SO CATEGORY DATA CAN BE OUTPUT,

C PROCESSING LOT DATA FOR CLUSTER 'J'

0126 D= 42 I=1,8

0127 CAT(I)=' '

0128 42 CONTINUE

0129 N1=CDNAME(5,J)

0130 N2=CDNAME(6,J)

0131 N=0

0132 DC=1

0133 D= 250 II=1,209

0134 IF(D%TRUF(DC),NE,'1')OR(D%TRUF(DC+1),NE,N2) GO TO 249

0135 N=N+1

0136 TRUF(1,N)=D%TRUF(DC+5)

0137 TRUF(2,N)=D%TRUF(DC+2)

0138 TRUF(3,N)=D%TRUF(DC+3)

0139 TRUF(4,N)=D%TRUF(DC+4)

0140 TRUF(5,N)=D%TRUF(DC+6)

0141 DECODE(5,103,D%TRUF(DC+7)) DBUF(N)

0142 103 FORMAT(I5)

0143 DSBUF(N)=DBUF(N)

0144 249 DC=DC+DFLD

0145 250 CONTINUE

0146 IF(DRPT,EQ,0) GO TO 44

C NOW CHECK FOR CATEGORIES

0147 IF(N,EQ,0) GO TO 44

0148 NC=1

0149 CAT(NC)=TRUF(1,1)

0150 IF(N,EQ,1) GO TO 271

0151 D= 270 I=2,N

0152 D= 260 JJ=1,NC

```

FORTRAN IV-PLUS V02-51          08102154      09-MAR-78          PAGE 4
CLURES,FTN      /TRIBLOCKS/WR
0153             IF (TRUF(1,N).EQ.CAT(NC)) GO TO 270
0154      260    CONTINUE
0155             NC=NC+1
0156             CAT(NC)=TRUF(1,N)
0157      270    CONTINUE
                C NOW SORT THE ENTRIES BY NET DISTANCE
0158      271    DO 280 II=1,N
0159             T=9999
0160             DO 275 JJ=1,N
0161             IF (T.LT.DRUF(JJ)) GO TO 275
0162             PTAB(II)=JJ
0163             T=DRUF(JJ)
0164      275    CONTINUE
0165             JK=PTAB(II)
0166             DRUF(JK)=9999
0167      280    CONTINUE
0168      44      CONTINUE
0169      105    FORMAT(15)
0170             XL2=L2DIST(J)
0171             XL2=XL2/100.
0172      210    FORMAT(12)
0173             WRITE(3,201)
0174      201    FORMAT('0',70X,'BRIGHTNESS GREEN NUMBER')
0175      202    FORMAT(14)
0176             WRITE(3,203) (NAME(L,J),L=1,6),A1B(J),A1G(J)
0177      203    FORMAT(' ',16X,'CLUSTER NAME:',2X,6A1,18X,'ACQUISITION 1',6
                214,8X,13)
0178             WRITE(3,211) (CLASS(L,J),L=1,6),A2R(J),A2G(J)
0179      211    FORMAT(' ',16X,'LABELING DOTS',2X,6A1,18X,'ACQUISITION 2',6
                214,8X,13)
0180             WRITE(3,212) YL2,A3P(J),A3G(J)
0181      212    FORMAT(' ',16X,'L1 DISTANCE',2X,F6,2,18X,'ACQUISITION 3',
                214,8X,13)
0182             WRITE(3,213) CAT,A4P(J),A4G(J)
0183      213    FORMAT(' ',16X,'CATAGORIES',2X,8(X,A1),8X,'ACQUISITION 4
                214,8X,13)
0184             WRITE(3,99)
                C
                C NOW OUTPUT DOTS IF ANY
0185             IF (N.EQ.0) GO TO 106
0186             WRITE(3,214)
0187      214    FORMAT(' ',10X,'DPTS      DISTANCE      DOTS      DISTANCE      D
                2'  DISTANCE      DPTS      DISTANCE      DPTS      DISTANCE')
0188             WRITE(3,99)
0189             DO 216 II=1,N,5
0190             FF=II+4
0191             LIM=5
0192             IF (FF.LE.N) GO TO 220
0193             FF=N
0194             LIM=N-II+1
0195      220    LL=1
0196             DO 217 JJ=II,FF
0197             NDX=PTAB(JJ)
0198             DO 218 KK=1,5
0199             DD(KK,LL)=TRUF(KK,NDX)
0200      218    CONTINUE

```

CLURES,FTN /TR:RI,0CKS/WP

ORIGINAL PAGE IS
OF POOR QUALITY

```
0201 DIS(LL)=DSBUF(NDX)
0202 DIS(LL)=DIS(LL)/100
0203 LL=LL+1
0204 217 CONTINUE
0205 WRITE(3,215) ((NF(KK,JJ),KK=1,5),DIS(JJ),JJ=1,LIM)
0206 216 CONTINUE
0207 215 FORMAT(' ',9X,5(A1,1X,3A1,1X,A1,3X,F6.2,5X))
C
C TEST FOR END OF DATA
C
0208 106 IF(J.EQ.ALSETS) GO TO 4
0209 J=J+1
0210 IF(K.EQ.MAXC) GO TO 15
0211 K=K+1
0212 GO TO 2
C
0213 15 FINISHED 1 PAGE. ON TOP OF FORM AND HEADER
0214 WRITE(3,98)
0215 WRITE(3,100)
0216 MAXC=5
0217 GO TO 1
C
0217 4 CONTINUE
0218 WRITE(3,50)
0219 50 FORMAT('0',1PX,'CLUSTERING CHANNEL LIST:')
0220 WRITE(3,51) (CT(I),I=1,C+MAX)
0221 51 FORMAT(' ',10X,14(1X,I2))
0222 RETURN
0223 END
```

CLURES,FTN /TRIOLECKS/NO PROGRAM SECTIONS

NUMBER	NAME	SIZE	ATTRIBUTES
1	SCDEF1	009076	1311 RNFJLCCN.LCL
3	SIDATA	001214	326 RNFJLCCN.LCL
4	SVARS	011464	2496 RNFJLCCN.LCL
5	STEMPS	000012	5 RNFJLCCN.LCL
6	CBTAS	005122	1321 RNFJLCCN.LCL
7	DUMMY	000014	2 RNFJLCCN.LCL
8	CLC04	000642	209 RNFJLCCN.LCL

ENTRY PRINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------	------	------	---------

CLURES	1-000020
--------	----------

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
ALSETS	102	4-011376	BIGCATS	102	5-003314	BATPT	102	6-003766	ELANK	102	4-011374
BUTZT	102	6-003770	CMWAX	102	4-011406	CLARUM	102	7-000000	CLJUM	102	7-000002
DFLO	102	4-011416	DR	102	7-001550	DRPT	102	4-011410	DU	102	6-011542
FILE	102	F-000006	I	102	4-011402	II	102	4-011434	IJ	102	4-011432
JJ	102	4-011426	JK	102	4-011420	K	102	4-011420	KX	102	4-011442
LFLD	102	4-011414	LIM	102	4-011456	LL	102	4-011450	MAX	102	4-011438
NC	102	4-011444	NPA	102	4-011436	NUM	102	4-011424	N1	102	4-007306
PCCG1	104	6-003336	PCCG2	104	6-003342	PCC1	104	6-003326	PCC2	104	6-013332
PCLG2	104	6-003362	PCL1	104	6-003346	PCL2	104	6-003352	PT	102	4-011422
RJ	102	4-011412	SETSR	102	4-011400	T	102	4-011446	TC	102	6-001554
X0	104	4-000010	XL2	104	6-000000						

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
A	104	6-001520	000030	12 (3,2)
A10	102	4-007400	000170	60 (60)
A10	102	4-007230	000270	60 (60)
A20	102	4-007760	000170	60 (60)
A20	102	4-007570	000170	60 (60)
A30	102	4-010340	000170	60 (60)
A30	102	4-010150	000270	60 (60)
A40	102	4-010720	000170	60 (60)
A40	102	4-010530	000170	60 (60)
BACATN	104	6-000220	000170	128 (6,8)
BACLAB	101	6-003316	000010	4 (4)
BACPR	102	6-000140	000010	12 (12)
BALAB	101	6-001600	000010	4 (4)
BAPAP	102	6-000010	000030	12 (12)
BARANS	104	6-001360	000060	24 (12)

FORTRAN IV-PLUS V02-51

CH102154

09-MAR-78

PAGE 8

CLURES,FTN /TRIBLOCKS/WR
CDRED

TOTAL SPACE ALLOCATED = 026000 5632

CLURES,LPI=CLURES

58

FORTRAN IV-PLUS	V02-51	08103126	09-MAR-78	PAGE 2
STDATA,FTN	/TRIBL DCKS/WP			
0053	LSETS=60/XL			
0054	CONTINUE			
0055	IF(IBUF(2),EQ,'F') GO TO 14			
0056	WRITE(3,97)			
0057	97 FORMAT(' ',49X,'STATISTICS REPORT')			
0058	16 CONTINUE			
0059	OSETS=5			
0060	PSETS=0			
0061	LFLD=11*9*NCH			
0062	IF(IBUF(2),EQ,'F') LFLD=18*9*NCH			
0063	20 J=1			
0064	RJ=12			
0065	IF(K,EQ,1) GO TO 6			
0066	CALL CDRED(IBUF,R,FILE)			
0067	4 DECODE(2,98,IBUF(R)) SETSR			
0068	IF(SETSR,EQ,0) RETURN			
0069	1 CONTINUE			
0070	IF(IBUF(2),EQ,'S') GO TO 22			
0071	CALL FNAME(IBUF(R),PFN(PI))			
0072	22 CALL POP(IBUF(RJ),PP(PI),IBUF(2))			
0073	CALL SNAME(IBUF(R),PN(PI),IBUF(2))			
0074	CALL MDTL(PTL(PI))			
0075	CALL MEAN(IBUF(R),PM(PI),NCH,IBUF(2))			
0076	DECODE(5,96,IBUF(2J+6)) DUM1			
0077	96 FORMAT(15)			
0078	DP2P=DP2P+DIUM1			
0079	IF(K,EQ,ALSETS) GO TO 10			
0080	K=K+1			
0081	IF(I,EQ,OSETS) GO TO 9			
0082	I=I+1			
0083	PI=PI+19			
0084	2 IF(J,EQ,SETSR) GO TO 20			
0085	J=J+1			
0086	RJ=RJ+LFLD			
0087	GO TO 1			
0088	9 K=K-1			
0089	10 CONTINUE			
0090	PSETS=PSETS+1			
0091	IF(PSETS,LE,LSETS) GO TO 11			
0092	PSETS=1			
0093	IF(DFLG,EQ,1) GO TO 14			
0094	LINE=66			
0095	CALL BNT			
0096	LSETS=60/XL			
0097	GO TO 11			
0098	14 CONTINUE			
0099	LINE=66			
0100	CALL KNT			
0101	11 CONTINUE			
0102	WRITE(3,90)			
0103	90 FORMAT(140,' ')			
0104	IF(IRUF(2),EQ,'S') GO TO 3			
0105	WRITE(3,101) (PFN(N),N=1,AMAX)			
0106	101 FORMAT(1H ,131A1)			
0107	WRITE(3,102) (PP(N),N=10,AMAX)			
0108	102 FORMAT(1H ,122A1)			

STDATA,FTN

/TRIBLOCKS/HR

```

0109 WRITE(3,101) (PN(N),N=1,NMAX)
0110 GO TO 7
0111 3 CONTINUE
0112 WRITE(3,101) (PN(N),N=1,NMAX)
0113 WRITE(3,102) (PP(N),N=10,NMAX)
0114 7 WRITE(3,103) (PTI(N),N=8,NMAX)
0115 103 FORMAT(1H,'NUMREX',122A1)
0116 IN=6
0117 IM=NMAX
0118 DO 5 J=1,NCH
0119 WRITE(3,104) (I,(PM(N),N=J,IM))
0120 104 FORMAT(1H,2X,12,12BA1)
0121 IN=IN+132
0122 IM=IM+132
0123 5 CONTINUE
0124 6 CONTINUE
0125 DO 30 N=1,132
0126 PF(N)=BLANK
0127 PFM(N)=BLANK
0128 PN(N)=BLANK
0129 PTL(N)=BLANK
0130 PP(N)=BLANK
0131 DO 40 NN=1,NCH
0132 PM(N+(NN-1)*132)=BLANK
0133 40 CONTINUE
0134 30 CONTINUE
0135 IF(K.EQ.1) GO TO 4
0136 IF(K.EQ.ALSETS) GO TO 15
0137 K=K+1
0138 I=1
0139 PI=11
0140 GO TO 2
0141 15 CONTINUE
0142 IF(LINE.EQ.0) LSAV=0
0143 LINE=PSETS*XL+LSAV
0144 CALL BNT
0145 BUPOP(10)=22932-DPOP
0146 BUUNC(10)=((22932,-DPOP)/22932,)*100
0147 RETURN
0148 END

```

NUMBER	NAME	SIZE	ATTRIBUTES	
1	SCPEDE1	003012	773	R01,C0A,LCL
2	SPDATA	000034	2	R01,C0A,LCL
3	SIDATA	000210	60	R01,C0A,LCL
4	SVARS	005410	1412	R01,C0A,LCL
5	STEMPS	000006	3	R01,C0A,LCL
6	CMIAS	005122	1321	R01,EVR,GBL
7	DUMNY	000004	2	R01,EVR,GBL
8	CLC00	000642	209	R01,EVR,GBL
9	PCMT	000004	2	R01,EVR,GBL

ENTRY PRINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
STDATA		1-000000						

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
ALSETS	102	4-005344	BACATS	102	4-005314	BLANK	101	4-005376
OUTOT	102	6-003770	GLADUM	102	7-000000	DFLG	102	7-000002
DP0P	102	4-005334	DV	102	6-001532	FILE	102	7-0010000
J1	102	4-005404	IM	102	4-005402	J	102	4-005366
LFLD	102	4-005364	LIME	102	9-000000	LSETS	102	4-005346
MCH	102	4-005346	MLNS	102	4-005350	M	102	4-005370
PAGE	102	9-000002	PC01	102	4-005336	MM	102	4-015406
PCL01	102	4-005356	PCL02	102	4-005362	PCC2	102	4-005332
PSETS	102	4-005362	R	102	5-000004	PCL7	102	4-005352
T1	102	4-005326	XL	102	4-005354	SETSR	102	4-005372
						TC	102	4-005394

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
BACAIN	R04	6-001520	000030	12 (1,2)
BACLAB	L01	6-003316	000010	120 (0,0)
BAC0R	102	6-000140	000030	4 (0)
BALABL	L01	6-001600	000010	12 (1,2)
BAP0P	102	6-000000	000030	4 (0)
BARANS	R04	6-001360	000060	24 (1,2)
BAUNCO	102	6-000000	000030	12 (1,2)
BAVAR	R04	6-001220	00'040	24 (1,2)
BCVA	102	6-003360	001200	64 (0,0)
BCVU	102	6-003560	000200	64 (0,0)
BUCAIN	R04	6-000620	000400	120 (0,0)
BUCLAB	L01	6-001960	000010	4 (0)
BUC0R	102	6-000170	000030	12 (1,2)

SYMBOL	ADDRESS	SYMBOL	ADDRESS	SYMBOL	ADDRESS	SYMBOL	ADDRESS
DATA.FTN	0001000	SYMBOL	0001000	DATA.FTN	0001000	SYMBOL	0001000
BURABL L01	6-001570	000010	4	(R)			
BURABP L02	6-000030	000030	12	(R)			
BURANS R04	6-001440	000060	24	(R)			
BUNYCD J02	6-000110	000030	12	(R)			
BUNYAR R04	6-001300	000060	24	(R)			
CCONS R04	6-005112	000010	4	(R)			
CLAD R04	6-003772	000010	4	(R)			
CLASFY L01	6-001610	000042	209	(R)			
CLDNY L01	6-000000	000042	209	(R)			
CLUD R04	6-004002	000010	4	(R)			
LOUF L01	F-0000020	000001	0	(R)			
LBLCD L01	6-002773	000321	104	(R)			
PF L01	4-005121	000204	66	(R)			
PFM L01	4-000000	000204	66	(R)			
PM L01	4-001020	004100	1056	(R)			
PP L01	4-000410	000204	66	(R)			
PTL L01	4-000014	000204	66	(R)			
RCONS R04	6-004112	001000	256	(R)			
TCNS R04	6-004012	000120	32	(R)			
TYPE L01	6-002452	000321	104	(R)			

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
1	1-001004	2	1-001426	3	1-002104	4	1-000724
5	1-002406	7	1-002240	8	1-000500	9	1-001466
11	1-001632	12	1-000072	13	1-000100	14	1-001602
16	1-000542	20	1-000030	22	1-001072	30	00
90	3-000052	91	3-000046	97	3-000016	99	3-000000
100	3-000010	101	3-000000	102	3-000006	103	3-000010
				104	3-000010	104	3-000010

FUNCTIONS AND SUBROUTINES REFERENCED

ONT	CODED	FRAME	KWT	MDTTL	MEAN	POP	SMAME	STOMP
TOTAL SPACE ALLOCATED = 016640 3792								
STDATA.LPIOSTDATA								

```

/TRIBLOCK$NR
SUBROUTINE DOTRAY(IBUF,R,FILE,PRINT)
IMPLICIT INTEGER (*=2)
BYTE IBUF(1)
INCLUDE 'COMMON1.DAT'
BYTE CLASY(418),CLDOT(418)
LOGICAL*1 TYPE(209),LBLE(209)
LOGICAL*1 RALAR(8),RACLAB(8),BLCLAB(8),BULABL(8)
REAL PCC1,PCC2,PCCG1,PCCG2,PCL1,PCL2,PCLG1,PCLG2
REAL CLAD,CLUD,ICONS,RCONS,CCONS
DIMENSION CLAD(?),CLUD(?),ICONS(2,8),RCONS(2,8,8)
DIMENSION RCVA(8,8),RCVU(8,8)
DIMENSION BAP(12),BUPAP,BALNCO
COMMON/CRIAS/BAP,BUPAP,BALNCO,BUNCO,BACPR,BUC,R,BACAIN,BUCAIN,
* BULABL,BALABL,CLASY,TYPE,LBLE,BACATS,BACLAB,
* PCC1,PCC2,PCCG1,PCCG2,PCL1,PCL2,PCLG1,PCLG2,
* RCVA,RCVU,BATBT,BLTTI,CLAD,CLUD,ICONS,RCONS,CCONS
COMMON/DUMMY/CLADUM,CLUDUP
COMMON/CLCOM/CLDOT
REAL A(3,2)
DIMENSION RAUNCO(12),BUNCO(12),BACOR(12),BUCOR(12)
REAL BACAIN(8,R),BUCAIN(6,8),BAVAR(12),BUVAR(12),BARANS(12),BURANS(12)
COMMON/PENT/LINE,PAGE
REAL S(6),VC(2)
INTEGER PAGE,FILE,R,REC,RJ,SETSR,PRINT
INTEGER *I(5),LN(3),COL(19),PASS
BYTE ALK,ONE,ONETO,SLASH,THE,LABEL(5)
BYTE LB,DY(4)
DATA ONE/1//,SLASH/1//,THE/2//
DATA DT/ 'L',RT/ 'S'
DATA TH1/1//,TH2/1//
DATA RLK/ ' '
LFLD=44
K=1
REC=1
WRITE(3,100)
WRITE(3,106) (I,I=1,4)
WRITE(3,101)
WRITE(3,102)
WRITE(3,103)
WRITE(3,105)
LCNT=6
SETSR=15
CONTINUE
RJ=12
N=1
CONTINUE
TYPE(K)=IBUF(RJ+12)
LBLE(K)=IBUF(RJ+11)
IF(IBUF(RJ+13).EQ.'1') GO TO 6
CLASY(2*K)=IBUF(RJ+10)
CLASY(2*K+1)=IBUF(RJ+9)
G=TR 9
CLASY(2*K)=TH2
CLASY(2*K+1)=TH1
IF(LCNT.LY.60) G=TR 10

```

```

0053 WRITE(3,100)
0054 WRITE(3,104) (1,101,4)
0055 WRITE(3,101)
0056 WRITE(3,102)
0057 WRITE(3,103)
0058 WRITE(3,105)
0059 LCNT=6
0060 30 DECODE(1,200,TYPE(K)) ID
0061 10=10+1
0062 1WRITE(3,104) (IRUF(RJ+1),L=1,5),DT(1D),L=LED(K),CLDRT(20K-1),
1CLDRT(2K),CLASY(20K-1),CLASY(20K), (IRUF(RJ+15+M),M=1,3),
2(IRUF(RJ+27+M),M=1,4), (IRUF(RJ+10+0),M=1,3),
3(IRUF(RJ+31 P),P=1,4), (IRUF(RJ+21+0),O=1,3),
4(IRUF(RJ+35+R),R=1,4), (IRUF(RJ+24+1),U=1,3), (IRUF(RJ+39+7),T=1,4)
LCNT=LCNT+1
K=K+1
0065 IF(N.EO.SETSR) GO TO 13
0066 N=21
0067 R=21-LFLD
0068 GO TO 1
0069 13 IF(REC.EQ.14) GO TO 3
0070 CALL CORED(IRUF,R,FILE)
0071 REC=REC+1
0072 IF(13C.EQ.14) SETSR=14
0073 GO TO 2
0074 3 PASS=2
0075 CALL CALC
0076 CALL RIAPRT(IRUF,TAPNO,PASS)
0077 IF(PRTOUT.EQ.1) RETURN
0078 IF(13CLDUM.EQ.1) GO TO 20
0079 IF(13CCL.LT.80),M,(PCC2.LT.80,)) PRTOUT=2
0080 20 IF(13CLDUM.EQ.1) RETURN
0081 IF(13CCL.LT.80),M,(PCL2.LT.80,)) PRTOUT=2
0082 REY=1
0083 100 FPRAT(1M,54X,10GT REPRAT)
0084 104 FPRAT(1,54X,4,1) PRIGHT GREEN(1)
0085 102 FPRAT(1,9X,10CT LINE PIXEL TYPE LABEL CLUSTER(1)
0086 103 FPRAT(1M,49X,1CLASS,4,1) PRESS NO,1)
0087 104 FPRAT(1,6X,3(3X,3A1),1X,2(4X,A1),1X,2(6X,A1,A1))
11X,41X,3A1,4X,4A1)
0088 105 FPRAT(1,10X)
0089 106 FPRAT(10,50X,4('ACQUISITION',12,2X))
0090 200 FPRAT(12)
0091 END

```


FORTAN IV-PLUS V02-51
 D0TRAY.FTN /TRIGLOCKS/MR
 PROGRAM SECTIONS

ATTRIBUTES

NUMBER NAME SIZE

1 SCDF1 002210 590
 3 S1DATA 000350 119
 4 SVARS 000220 72
 5 STFMPS 000002 1
 6 CBTAS 000122 1321
 7 DUMY 000004 2
 8 CLCBM 000642 209
 9 PC1 000004 2

ENTRY POINTS

NAME TYPE ADDRESS NAME TYPE ADDRESS NAME TYPE ADDRESS NAME TYPE ADDRESS

D0TRAY 1-000000

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
BACATS	102	6-003314	BATRT	102	4-003760	BLK	101	4-000136	BUCATS	102	6-001420
CLADUM	102	7-000000	CLUDUM	102	7-000002	DB	102	6-001550	DU	102	6-001552
LCNT	102	4-000170	ID	102	4-000174	K	102	4-000164	L	102	4-000176
LN	102	4-000202	LFLD	102	4-000162	LINE	102	9-000000	M	102	4-000200
PAGE	102	9-000002	PASS	102	4-000134	PCCG1	104	6-003336	PCC2	104	6-003342
PRTOUT	102	F-000010*	PC1G1	102	6-003356	PCLG2	104	6-003362	PCL1	104	6-003346
SETSR	102	4-000044	SLASH	101	4-000210	R	102	F-000004*	REC	102	4-000340
TH1	102	4-000156	TM2	102	4-000141	T	102	4-000214	TAPAC	102	4-000216
						TWO	101	4-000142	U	102	4-000212

ARRAYS

DIMENSIONS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
A	104	6-001520	000030	12 (3,2)
BACAIN	104	6-000220	000400	128 (8,8)
BACLAG	101	6-003316	000010	4 (8)
BACER	102	6-000140	000030	12 (12)
BALABL	101	6-001600	000010	4 (8)
BAPPP	102	6-000000	000030	12 (12)
BARAUS	104	6-001346	000640	24 (12)
BAUNCB	102	6-000060	000030	12 (12)
BAVAR	104	6-001220	000060	24 (12)
BCVA	102	6-003566	000200	64 (8,8)
BCVU	102	6-003566	000200	64 (8,8)
BUCAIN	104	6-000620	000400	128 (8,8)
BUCLAG	101	6-001560	000010	4 (8)
BUCAR	102	6-000170	000030	12 (12)
BULABL	101	6-001570	000010	4 (8)

DATA,FTN	TRIPLCRS/HP	12	(12)
BUPOP 1e2	6-000030	000030	12 (12)
BUQNS 0e4	6-001440	000000	24 (12)
BUWCP 1e2	6-000110	000030	12 (12)
BUVAR 0e4	6-001300	000000	24 (12)
CCMS 0e4	6-005112	000010	4 (2)
CLAD 0e4	6-003772	000010	4 (2)
CLASFY 1e1	6-001610	000642	209 (418)
CLDPT 1e1	8-000000	000642	209 (418)
CLUD 0e4	6-004002	000010	4 (2)
CPL 1e2	4-000066	000046	19 (19)
DT 1e1	4-000151	000004	2 (4)
IR:F 1e1	F-000002	000001	0 (1)
LABEL 1e1	4-000143	000005	2 (5)
LEAD 1e1	6-002773	000321	104 (209)
LN 1e2	4-000060	000006	3 (3)
NI 1e2	4-000046	000012	5 (5)
RCNS 0e4	6-004112	001000	256 (2,818)
S 0e4	4-000000	000030	12 (4)
TCNS 0e4	6-004012	000190	32 (2,8)
TYPE 1e1	6-002452	000321	104 (209)
VC 0e4	4-000030	000010	4 (2)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
1	1-000320	2	1-000776	3	1-002002	8	1-000410
10	1-000664	13	1-001700	20	1-002132	100	3-000000
102	3-000054	103	3-000132	104	3-000174	105	3-000224
200	3-000316						

FUNCTIONS AND SUBROUTINES REFERENCED

BIAPRT	CALC	CDRED
TOTAL SPACE ALLOCATED = 011004 2306		
DATA,FTN		

ORIGINAL PAGE IS OF POOR QUALITY

```

SEPRPT,PT4 7P130C/S/MP
0001 SUPRPUTIVE SEOPT(TCMAN,ITSEF,SPATA)
0002 IMPLICIT INTEGER(A-Z)
0003 LOGICAL SDATA(1),ITSEF(1)
0004 WRITE(2,200)
0005 IF(TCMAN.NE.0) G= TP 20
0006 WRITE(2,199)
0007 F20MAT(0),DATA FILE EMPTY)
0008 RETURN
0009 WRITE(2,201) (I,181,16)
0010 WRITE(2,202)
0011 WRITE(2,203) (ITSEF(J),J=1,8)
0012 DATA=0
0013 TP 50 I=1,14
0014 04 47 KK=1,6
0015 IF(SDATA(KK+DPTR).NE.0) G= TP 48
0016 CONTINUE
0017 G= TP 49
0018 G= TP (1,2,3,4,5,6,7,8,9,10,11,12,13,14),1
0019 1 WRITE(2,201)
0020 G= TP 46
0021 2 WRITE(2,202)
0022 G= TP 46
0023 3 WRITE(2,203)
0024 G= TP 46
0025 4 WRITE(2,104)
0026 G= TP 46
0027 5 WRITE(2,105)
0028 G= TP 46
0029 6 WRITE(2,106)
0030 G= TP 46
0031 7 WRITE(2,107)
0032 G= TP 46
0033 8 WRITE(2,108)
0034 G= TP 46
0035 9 WRITE(2,109)
0036 G= TP 46
0037 10 WRITE(2,110)
0038 G= TP 46
0039 11 WRITE(2,111)
0040 G= TP 46
0041 12 WRITE(2,112)
0042 G= TP 46
0043 13 WRITE(2,113)
0044 G= TP 46
0045 14 WRITE(2,114)
0046 46 WRITE(2,203) (SDATA(DPTR+KK),KK=1,6)
0047 49 DPTR=DPTR+8
0048 50 CONTINUE
0049 RETURN
0050 F20MAT(0),SEPARABILITY REPORT)
0051 201 F20MAT(0),10X,CHANNELS',78,1613,6X,SEPARABILITY')
0052 202 F20MAT(0),19X,AVAILABLE',28X,(TCMANS(2X,1X))
0053 203 F20MAT(14),8X,21,1,1,641)
0054 101 F20MAT(10),19X,(A-C),11X,(12X,1Y))
0055 102 F20MAT(10),46X,412X,1X))
0056 103 F20MAT(10),597,412V,1X))

```

ORIGINAL PAGE IS OF POOR QUALITY

ORIGINAL PAGE IS
OF POOR QUALITY

0057 104 F0RMAT('0',70X,4(2X,'X'))
0058 105 F0RMAT('0',19X,'2ACQ',11X,8(2X,'X'))
0059 106 F0RMAT('0',34X,4(2X,'X'),12X,4(2X,'X'))
0060 107 F0RMAT('0',34X,4(2X,'X'),24X,4(2X,'X'))
0061 108 F0RMAT('0',46X,8(2X,'X'))
0062 109 F0RMAT('0',44X,4(2X,'X'),12X,4(2X,'X'))
0063 110 F0RMAT('0',58X,8(2X,'X'))
0064 111 F0RMAT('0',19X,'3ACQ',11X,12(2X,'X'))
0065 112 F0RMAT('0',36X,8(2X,'X'),12X,4(2X,'X'))
0066 113 F0RMAT('0',36X,4(2X,'X'),12X,8(2X,'X'))
0067 114 F0RMAT('0',46X,12(2X,'X'))
0068 END

SEPRPT.FTM /TRIBLOCKS/NR
 PROGRAM SECTIONS

NUMBER	NAME	SIZE	ATTRIBUTES
1	SCODFI	001322	361
2	SPDATA	000036	15
3	SIDATA	000640	208
4	SVARS	020010	4

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
SEPRPT		1-000030						

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
DCTR	102	4-000004	J	102	4-000000	KK	102	4-000002
						TCMIX	102	3-000036

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
SDATA L01	F-000006	000001	0	(1)
TATSEP L01	F-000006	000001	0	(1)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
1	1-000444	2	1-000276	3	1-000530	4	1-000567	5	1-000412
6	1-000642	7	1-000672	8	1-000722	9	1-000752	10	1-001002
11	1-001032	12	1-001062	13	1-001112	14	1-001142	21	1-001034
46	1-001170	47	1-001200	48	1-001222	49	1-001254	50	1-001284
101	3-000174	102	3-000222	103	3-000240	104	3-000296	105	3-000274
106	3-000322	107	3-000352	108	3-000402	109	3-000420	110	3-000450
111	3-000466	112	3-000514	113	3-000544	114	3-000574	199	3-000000
200	3-000024	201	3-000056	202	3-000122	203	3-000154		

TOTAL SPACE ALLOCATED * 002230 508

NB FPP INSTRUCTIONS GENERATED

SEPRPT.LPISEPRPT

CALC,FTN

/TRIAL LOCKS/WR

0053

GO TO 15

0054 12

IF (RACLAR(I),NE,'N') GO TO 15

0055

N=1

0056 15

CONTINUE

0057

DO 40 I=1,11

0058

LCAT(I)=0

0059 40

CONTINUE

0060

DO 401 J=1,8

0061

DO 401 I=1,8

0062

BCVA(J,I)=0

0063 401

CONTINUE

0064

DO 41 I=1,12

0065

PARAMS(I)=0

0066 41

CONTINUE

0067

DO 39 I=1,209

0068

LR=LRLED(I)

0069

CL=CLASFY(2*I-1)

0070

CL2=CLASFY(2*I)

0071

IF (LB,EQ,' ') GO TO 39

0072

IF (CL,EQ,'0',.7H,CL,EQ,' ') GO TO 39

0073

IF (TYPE(I),EQ,'2',OR,TYPE(I),EQ,'0') GO TO 35

C

C COMPUTE TYPE 1 DPT DATA

C

0074

NTYP1=NTYP1+1

0075

IF (LR,NE,CL,AND,CL,NE,' ') GO TO 32

0076

NATI=NATI+1

0077

GO TO 35

0078 32

IF (CHECK,LT,2) GO TO 35

0079 33

IF (LR,EQ,BACLAR(W),AND,CL,EQ,BACLAR(S)) NGIJ=NGIJ+1

0080

IF (LR,EQ,BACLAR(S),AND,CL,EQ,BACLAR(W)) NGIJ=NGIJ+1

C

C COMPUTE TYPE 2 DPT

C

0081 35

IF (TYPE(I),NE,'2') GO TO 39

0082

IF (CL2,NE,' ') GO TO 39

0083

NTYP2=NTYP2+1

0084

IF (LR,NE,' ') AND (CL,NE,' ') NRCL=NRCL+1

C

C CONSTRUCT BIAS CORRECTION VECTORS(LB)

C

0085

Z1=0

0086

Z2=0

0087

DO 43 K=1,BACATS

0088

IF (LR,EQ,BACLAR(K)) Z1=K

0089

IF (CL,EQ,BACLAR(K)) Z2=K

0090 43

CONTINUE

0091

IF (Z1,EQ,0,OR,Z2,EQ,0) GO TO 39

0092

BCVA(Z1,Z2)=BCVA(Z1,Z2)+1

0093 39

CONTINUE

C

C COMPUTE GRAIN POPULATION, CLASSIFIED %, CORRECTED %

C

0094

IF (CHECK,LT,2) GO TO 25

0095

BAPOP(12)=BAPOP(W)+BAPOP(S)

0096

BAUNCP(12)=BAUNCP(W)+BAUNCE(S)

C
C COMPUTE GRAIN CORRECTED X AND VARIANCE X

0097 47 ALG \bar{T} =BCVA(W,W)+BCVA(W,S)+BCVA(S,W)+BCVA(S,S)

0098 ALG \bar{B} =0

0099 DO 155 I=1,BACATS

0100 ALG \bar{B} =ALG \bar{B} +BCVA(I,W)+BCVA(I,S)

0101 155 CONTINUE

0102 ALGTF=FL \bar{T} AT(ALG \bar{T})

0103 ALGBF=FL \bar{B} AT(ALG \bar{B})

0104 ALG=ALGTF/ALGBF

0105 ALNB=0

0106 ALNT=0

0107 DO 156 I=1,BACATS

0108 IF(I.EQ.W.OR.I.EQ.S)GO TO 156

0109 DO 156 J=1,BACATS

0110 IF(J.EQ.W.OR.J.EQ.S)GO TO 156

0111 ALNT=ALNT+BCVA(I,J)

0112 156 CONTINUE

0113 DO 158 I=1,BACATS

0114 DO 158 J=1,BACATS

0115 IF(J.EQ.W.AND.J.EQ.S)GO TO 158

0116 ALNB=ALNB+BCVA(I,J)

0117 158 CONTINUE

0118 BASEF=FL \bar{T} AT(BASE=BAP \bar{P} (11))

0119 BASE1=0.

0120 IF(BASEF.LE.0) BASEF=1

0121 BAPF=FL \bar{B} AT(BAP \bar{P} (12))

0122 BASE1=BAPF/BASEF

0123 ALNTF=FL \bar{T} AT(ALNT)

0124 ALNBF=FL \bar{B} AT(ALNB)

0125 IF(ALNBF.LE.0) ALNBF=1

0126 AL \bar{N} =ALNTF/ALNBF

0127 N3F=0.

0128 DO 159 I=1,BACATS

0129 IF(I.EQ.W.OR.I.EQ.S)GO TO 159

0130 N3F=N3F+BAP \bar{P} (I)

0131 159 CONTINUE

0132 BAP \bar{P} (12)=(ALG \bar{B} +BASE1*(1.-ALG \bar{B}))*N3F/BASEF*100.

0133 BASE2=(N3F/BASEF*100.)**2

0134 Z4=0

0135 DO 21 J=1,BACATS

0136 Z4=Z4+BCVA(J,W)+BCVA(J,S)

0137 21 CONTINUE

0138 VAR=0.

0139 Z4F=FL \bar{B} AT(Z4-1)

0140 IF(Z4F.LT.1)GO TO 24

0141 VAR=((BASE1*100.)**2*(ALG \bar{B} *(1.-ALG \bar{B})))/Z4F

0142 24 IF(ALNBF.GT.1.)GO TO 22

0143 BAVAR(12)=VAR

0144 GO TO 25

0145 22 BAVAR(12)=VAR+BASE2*AL \bar{N} *(1.-AL \bar{N})/(ALNBF-1.)

C
C COMPUTE THE RANDOM SAMPLE FOR EACH CATEGORY EXCEPT FOR GRAIN

0146 25 BATO \bar{T} =0

0147

PSUM=0,
DO 160 I=1,BACATS

0148

PSUM=PSUM+FL0AT(:AP0P(I))
CONTINUE

0149

160

PSUM=PSUM/(22932.*HAP0P(11))
N0CLF=FL0AT(N0CL)

0152

DO 165 I=1,BACATS

0153

BCVT(I)=0

0154

DO 168 J=1,BACATS
BCVT(I)=BCVT(I)+CVA(I,J)

0155

168

CONTINUE
PARANS(I)=FL0AT(BCVT(I))/N0CLF*PSUM*100.

0156

BAT0T=BAT0T+PCVT(I)

0157

IF(I,NE,4,AND,I,NE,5)GO TO 165

0158

GWS=GWS+BCVT(I)

0159

165

CONTINUE

C

C COMPUTE THE RANDOM SAMPLE FOR GRAIN

C

0163

IF(CHECK,LT,2)GO TO 100

0164

GWSF=FL0AT(GWS)

0165

PARANS(12)=GWSF/N0CLF*PSUM*100.

C

C COMPUTE PCC1, PCC2, PCCG1, PCCG2

0166

100

RATAT=0

0167

X=0

0168

IF (NTYP1,EG,0) GO TO 45

0169

PCC1=FL0AT(NAT1)*100./FL0AT(NTYP1)

0170

45

IF (NTYP2,EG,0) GO TO 469

0171

X=PCVA(1,1)+PCVA(2,2)+PCVA(3,3)+PCVA(4,4)+PCVA(5,5)

0172

X=X+PCVA(6,6)+PCVA(7,7)+PCVA(8,8)

0173

PCC2=FL0AT(X)*100./FL0AT(NTYP2)

0174

469

IF(CHECK,LT,2)GO TO 470

0175

RATAT=PCVA(W,S)+PCVA(S,W)

0176

DO 175 I=1,BACATS

0177

RATAT=BATAT+PCVA(I,I)

0178

175

CONTINUE

0179

IF (NTYP1,EG,0)GO TO 468

0180

PCCG1=(FL0AT(NAT1)+FL0AT(NG1J))*100./FL0AT(NTYP1)

0181

468

IF (NTYP2,EG,0)GO TO 470

0182

PCCG2=FL0AT(:ATAT)*100./FL0AT(NTYP2)

0183

470

CONTINUE

C CLADⁿ CALCULATIONS FIRST

0184

DO 280 M=1,2

0185

IF(MM,EG,1) KK=

0186

IF(MM,EG,2) KK=S

0187

SLW=0.

0188

CLW=0.

0189

IF(KK,EG,0) GO TO 270

0190

WARY(7)=0.

0191

WARY(1)=RAC0P(KK)

0192

WARY(2)=BAUNCO(KK)

0193

WARY(3)=PARANS(KK)

0194

WARY(4)=RAVAN(KK)

0195

WARY(5)=RACAIN(KK, KK)

0196

WARY(6)=PCC1

FORTRAN IV-PI US		V02-51	08104125	09-PAR=78	PAGE 5
			TRIBLOCK/WR		
0197			IF(V,NE,0) WARY(7)=BACA(N,N)		
0198			WARY(8)=PCC2		
0199			DO 250 I=1,8		
0200			SLI=SLW+WARY(I)*TCONS(MM,I)		
0201			DO 240 J=1,8		
0202			CLW=CLW+WARY(I)*WARY(J)*RCONS(MI,I,J)		
0203	240		CONTINUE		
0204	250		CONTINUE		
0205			SLW=SLW+CCONS(MM)		
0206	270		CLAD(MM)=CLW+SLW		
0207	280		CONTINUE		
0208			NTYP2=0		
0209			NAII=0		
0210			NBII=0		
0211			NGIJ=0		
0212			RASE=22932		
0213			GWS=0		
0214			RUPDP(12)=0		
0215			BUVAN(12)=0		
0216			RUC2R(12)=0		
0217			BUUNC0(12)=0		
0218			RURANS(12)=0		
0219			W=0		
0220			S=0		
0221			CHECK=0		
0222			PCL1=0		
0223			N0FL=0		
0224			NTYP1=0		
0225			PCL2=0		
0226			PCLG1=0.		
0227			PCLG2=0.		
0228			IF(CLUDDU=,EG,1) RETURN		
0229			DO 915 I=1,BUCATS		
0230			IF(BUCLAR(I),NF,0) GO TO 910		
0231			CHECK=CHECK+1		
0232			W=I		
0233			GO TO 915		
0234	910		IF(BUCLAR(I),NF,0) GO TO 915		
0235			CHECK=CHECK+1		
0236			S=I		
0237	915		CONTINUE		
0238			DO 940 I=1,11		
0239			LCAT(I)=0		
0240	940		CONTINUE		
0241			DO 9402 J=1,8		
0242			DO 9401 I=1,8		
0243			BCVJ(J,I)=0		
0244	9401		CONTINUE		
0245			DO 941 I=1,12		
0246			RURANS(I)=0		
0247	941		CONTINUE		
0248			DO 939 I=1,209		
0249			LB=LPLFD(I)		
0250			CL=CLDNT(2*I-1)		
0251			CL2=CLDNT(2*I)		
0252			IF(LB=EQ,0) GO TO 939		

```

CALC.FTN /TR:HLRCKS/WR
0253 IF (CL.EQ.'0'.OR.CL.EQ.' ') GO TO 930
0254 IF (TYPE(I).EQ.'2'.OR.TYPE(I).EQ.'0') GO TO 935
    
```

C
C COMPUTE TYPE 1 DDT DATA

```

0255 NTYP1=NTYP1+1
0256 IF (LR.NE.CL.AND.CL.NE.' ') GO TO 932
0257 NALL=NALL+1
0258 GO TO 935
0259 932 IF(CHECK.LT.2)GO TO 939
0260 933 IF (LR.EQ.BUCLAB(W).AND.CL.EQ.BUCLAB(S)) NGIJ=NGIJ+1
0261 IF (LR.EQ.BUCLAB(S).AND.CL.EQ.EUCLAB(W)) NGIJ=NGIJ+1
    
```

C
C COMPUTE TYPE 2 DDT

```

0262 935 IF (TYPE(I).NE.'2')GO TO 939
0263 IF (CL2.NE.' ') GO TO 939
0264 NTYP2=NTYP2+1
0265 IF (LR.NE.' '.AND.CL.NE.' ') NZCL=NZCL+1
    
```

C
C CONSTRUCT BIAS CORRECTION VECTORS(L)

ORIGINAL PAGE IS
OF POOR QUALITY

```

0266 Z1=0
0267 Z2=0
0268 D2 943 I=1, IUCATS
0269 IF (LR.EQ.BUCLAB(W))Z1=K
0270 IF (CL.EQ.BUCLAB(W))Z2=K
0271 943 CONTINUE
0272 IF (Z1.EQ.0.OR.Z2.EQ.0)GO TO 939
0273 HCVU(Z1,Z2)=HCVU(Z1,Z2)+1
0274 939 CONTINUE
    
```

C
C COMPUTE GRAIN POPULATION, CLASSIFIED %, CORRECTED %

```

0275 IF(CHECK.LT.2)GO TO 925
0276 HUPPP(12)=HUPPP(I)+HUPPP(S)
0277 BUUNCP(12)=BUUNCP(W)+BUUNCP(S)
    
```

C
C COMPUTE GRAIN CORRECTED % AND VARIANCE %

```

0278 947 ALGT=HCVU(W,W)+HCVU(W,S)+HCVU(S,W)+HCVU(S,S)
0279 ALGR=0
0280 D2 9155 I=1, IUCATS
0281 ALGR=ALGR+HCVU(I,W)+HCVU(I,S)
0282 9155 CONTINUE
0283 ALGTF=FLPAT(ALGT)
0284 ALGPF=FLPAT(ALGR)
0285 ALG=ALGTF/ALGPF
0286 ALNR=0
0287 ALNT=0
0288 D2 9156 I=1, IUCATS
0289 IF (I.EQ.W.OR.I.EQ.S)GO TO 9156
0290 D2 9156 J=1, IUCATS
0291 IF (J.EQ.W.OR.J.EQ.S)GO TO 9156
0292 ALNT=ALNT+HCVU(I,J)
0293 9156 CONTINUE
    
```

```

CALC,FTN /TRIPLOCKS/WR
0294 DO 9158 J=1,NUCATS
0295 DO 9158 J=1,NUCATS
0296 IF(J.EQ.4,OR(J.EQ.5))GO TO 9158
0297 ALNB=ALNR*PCVU(I,J)
0298 9158 CONTINUE
0299 BASEF=FLOAT(BASE*BUPOP(11))
0300 BASE1=0.
0301 IF(BASEF.LE.0) BASEF=1
0302 BAPF=FLOAT(BUPOP(12))
0303 BASE1=BAPF/BASEF
0304 ALNBF=FLOAT(ALNB)
0305 ALNBF=FLOAT(ALNBF)
0306 IF(ALNBF.LE.0) ALNBF=1
0307 ALNB=ALNBF/ALNBF
0308 NSF=0.
0309 DO 9159 I=1,NUCATS
0310 IF(I.EQ.4,OR(I.EQ.5))GO TO 9159
0311 NSF=NSF+FLOAT(BUPOP(11))
0312 9159 CONTINUE
0313 BUCNR(12)=(ALGO*BASE1+(1.-ALN)*NSF/BASEF)*100.
0314 BASE2=(NSF/BASEF*100.)*.02
0315 Z4=0
0316 DO 921 J=1,NUCATS
0317 Z4=Z4+BCVU(J,4)*PCVU(J,5)
0318 921 CONTINUE
0319 VAR=0.
0320 Z4F=FLOAT(Z4-1)
0321 IF(Z4.LT.1)GO TO 924
0322 VAR=((BASE1*100.)*.02*(ALGO*(1.-ALG)))/Z4F
0323 924 IF(ALNBF.GT.1.)GO TO 922
0324 BUVAR(12)=VAR
0325 GO TO 925
0326 922 BUVAR(12)=VAR*BASE2*ALN*(1.-ALN)/(ALNBF-1.)
C
C COMPUTE THE RANDOM SAMPLE FOR EACH CATEGORY EXCEPT FOR GRAIN
C
0327 925 BUT2T=0
0328 PSUM=0.
0329 DO 9160 I=1,NUCATS
0330 PSUM=PSUM+FLOAT(BUPOP(11))
0331 9160 CONTINUE
0332 PSUM=PSUM/(22932.-BUPOP(11))
0333 NRCLF=FLOAT(NRCL)
0334 DO 9165 I=1,NUCATS
0335 RCVT(I)=0
0336 DO 9166 J=1,NUCATS
0337 RCVT(I)=RCVT(I)+PCVU(I,J)
0338 9166 CONTINUE
0339 BURANS(I)=FLOAT(RCVT(I))/NRCLF*PSUM*100.
0340 BUT2T=BU2T+RCVT(I)
0341 IF(I.EQ.4,OR(I.EQ.5))GO TO 9165
0342 GWS=GWS+RCVT(I)
0343 9165 CONTINUE
C
C COMPUTE THE RANDOM SAMPLE FOR GRAIN
C

```

ORIGINAL PAGE IS
OF 2002 CI

CALC,FTN /TRIPBLOCKS/WR

0344 IF(CHECK,LT,2)GOTO 9100

0345 GWSF=FLOAT(GWS)

0346 RUPANS(12)=GWSF/NOCLF*PSUM*100.

C

C COMPUTE PCC1, PCC2, PCCG1, PCCG2

0347 9100 BATAT=0

0348 X=0

0349 IF (NTYP1.EQ.0) GOTO 945

0350 PCL1=FLOAT(MA(I))*100./FLOAT(NTYP1)

0351 945 IF(NTYP2.EQ.0) GOTO 9469

0352 X=RCVU(1,1)+RCVU(2,2)+RCVL(3,3)+RCVU(4,4)+RCVU(5,5)

0353 X=X+RCVU(6,6)+RCVU(7,7)+RCVL(8,8)

0354 PCL2=FLOAT(X)*100./FLOAT(NTYP2)

0355 9469 IF(CHECK,LT,2)GOTO 9470

0356 BATAT=RCVU(W,S)+RCVU(S,W)

0357 DO 9175 I=1,NUCATS

0358 HATAT=BATAT+RCVU(I,1)

0359 9175 CONTINUE

0360 IF (NTYP1.EQ.0)GOTO 9468

0361 PCLG1=(FLOAT(MA(I))+FLOAT(MG(I,J)))*100./FLOAT(NTYP1)

0362 9468 IF(NTYP2.EQ.0)GOTO 9470

0363 PCLG2=FLOAT(BATAT)*100./FLOAT(NTYP2)

0364 9470 CONTINUE

C CLUDW CALCULATIONS NEW

0365 DO 9280 MM=1,2

0366 IF(MM.EQ.1) KK=S

0367 IF(MM.EQ.2) KK=S

0368 SLW=0.

0369 CLW=0.

0370 IF(KK.EQ.0) GOTO 9270

0371 WARY(7)=0.

0372 WARY(1)=RUCOR(KK)

0373 WARY(2)=RUDIMP2(KK)

0374 WARY(3)=RUPARS(KK)

0375 WARY(4)=RUVAR(KK)

0376 WARY(5)=RUCAIN(KK, KK)

0377 WARY(6)=PCL1

0378 IF(N.NE.0) WARY(7)=RUCAIN(N,N)

0379 WARY(8)=PCL2

0380 DO 9250 I=1,N

0381 SLW=SLW+WARY(I)*TCONS(MM, I)

0382 DO 9240 J=1,N

0383 CLW=CLW+WARY(I)*WARY(J)*RCONS(MM, I, J)

0384 9240 CONTINUE

0385 9250 CONTINUE

0386 SLW=SLW+CCONS(MM)

0387 9270 CLUD(MM)=CLW+SLW

0388 9280 CONTINUE

0389 CONTINUE

0390 RETURN

0391 END

PROGRAM SECTIONS

ATTRIBUTES

NUMBER	NAME	SIZE	ATTRIBUTES
1	SCDCE1	01150	2356
2	SPATA	00204	2
4	SVARS	20062	153
5	STFAS	00034	14
6	CSIAS	00527	1321
7	DUMV	20004	2
8	CLC04	00042	209

ENTRY PRINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------	------	------	---------

9 CALC 1-000000

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
ALG	RO4	4-000037	ALGP	RO4	4-000442	ALGRF	RO4	4-000122	ALGT	RO4	4-000440
ALK	RO4	4-000056	ALND	RO4	4-000444	ALNRF	RO4	4-000112	ALV	RO4	4-000116
BACATS	RO4	6-003314	HAFV	RO4	4-000126	BASE	RO4	4-000410	BASEF	RO4	4-000192
BASER	RO4	4-000072	SATAT	RO4	4-000432	BAYCT	RO4	6-003746	EUCATS	RO4	4-000186
CHECK	RO4	4-000422	CL	RO4	4-000327	CLADUM	RO4	7-000070	CLDUM	RO4	7-000102
CL2	RO4	4-000374	DP	RO4	4-000150	DV	RO4	6-000152	GV	RO4	6-000132
J	RO4	4-000426	J	RO4	4-000430	K	RO4	4-000436	KK	RO4	4-000440
N	RO4	4-000456	N	RO4	4-000420	MAIL	RO4	4-000402	MAIL	RO4	4-000404
NCLL	RO4	4-000424	NCLF	RO4	4-000142	NTYP1	RO4	4-000400	NTYP2	RO4	4-000376
PC091	RO4	6-003362	PC032	RO4	6-003342	PC01	RO4	6-003326	PC03	RO4	6-003356
PC032	RO4	6-003362	PC01	RO4	6-003346	PC02	RO4	6-003322	PSUM	RO4	4-000186
SLV	RO4	4-000226	TC	RO4	6-001554	VAR	RO4	4-000052	Z	RO4	4-000414
Z1	RO4	4-000432	Z2	RO4	4-000434	Z4	RO4	4-000450	ZAF	RO4	4-000116

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
------	------	---------	------	------------

A	RO4	6-001520	00030	12 (3,2)
BACATN	RO4	5-000220	00040	12 (6,8)
BACLAR	LO1	6-003316	00010	4 (8)
BACR	LO2	6-000140	00030	12 (12)
BALBL	LO1	6-001600	00030	12 (8)
BAPER	LO2	6-000000	00030	12 (12)
BAPPF	RO4	4-000276	00060	24 (12)
BARANS	RO4	6-001356	00060	24 (12)
BAUNCE	LO2	6-000060	00030	12 (12)
BAVAN	RO4	6-001221	00060	24 (12)
BCVA	LO2	6-003366	00020	64 (8,8)
BCVT	LO2	4-000026	00020	6 (8)
BCVU	LO2	6-003566	00020	64 (8,8)

ORIGINAL PAGE IS OF POOR QUALITY


```

0001 /TPR/BLCKS/WF
0002 SUBROUTINE DIAPRT(UNIT,TAPE,PASS)
0003 IMPLICIT INTEGER (A-Z)
0004 LOGICAL UNIT, TAPE, PASS
0005 I=0
0006 TYPE=CLASSIFY(UNIT), CLDZT(4,10)
0007 LOGICAL UNIT, TAPE, PASS, LBLR(209)
0008 LPSICAL(1) = PALA(1), RACAL(1), ELCLAH(8), FUVAHL(6)
0009 REAL PCC1, PCC2, PCC3, PCC4, PCL1, PCL2, PCL3, PCL4, PCL5
0010 REAL CLAR, EL, PALA(1), RACAL(1), ELCLAH(8), FUVAHL(6)
0011 DIMENSION CLAR(2), CLDZT(2), TORNS(2,8), CCM(2), RCM(2), RCM(2,8,6)
0012 DIMENSION CVAL(1), CVAL(2), CVAL(3), CVAL(4)
0013 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0014 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0015 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0016 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0017 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0018 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0019 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0020 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0021 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0022 CVAL(1) = CVAL(2) = CVAL(3) = CVAL(4) = 0
0023 DATA BLA(1), BLA(2), BLA(3), BLA(4), BLA(5), BLA(6), BLA(7), BLA(8), BLA(9), BLA(10), BLA(11), BLA(12), BLA(13), BLA(14), BLA(15), BLA(16), BLA(17), BLA(18), BLA(19), BLA(20)
0024 DATA BLA(21), BLA(22), BLA(23), BLA(24), BLA(25), BLA(26), BLA(27), BLA(28), BLA(29), BLA(30), BLA(31), BLA(32), BLA(33), BLA(34), BLA(35), BLA(36), BLA(37), BLA(38), BLA(39), BLA(40)
0025 DATA BLA(41), BLA(42), BLA(43), BLA(44), BLA(45), BLA(46), BLA(47), BLA(48), BLA(49), BLA(50), BLA(51), BLA(52), BLA(53), BLA(54), BLA(55), BLA(56), BLA(57), BLA(58), BLA(59), BLA(60)
0026 DATA BLA(61), BLA(62), BLA(63), BLA(64), BLA(65), BLA(66), BLA(67), BLA(68), BLA(69), BLA(70), BLA(71), BLA(72), BLA(73), BLA(74), BLA(75), BLA(76), BLA(77), BLA(78), BLA(79), BLA(80)
0027 DATA BLA(81), BLA(82), BLA(83), BLA(84), BLA(85), BLA(86), BLA(87), BLA(88), BLA(89), BLA(90), BLA(91), BLA(92), BLA(93), BLA(94), BLA(95), BLA(96), BLA(97), BLA(98), BLA(99), BLA(100)
0028 DATA BLA(101), BLA(102), BLA(103), BLA(104), BLA(105), BLA(106), BLA(107), BLA(108), BLA(109), BLA(110), BLA(111), BLA(112), BLA(113), BLA(114), BLA(115), BLA(116), BLA(117), BLA(118), BLA(119), BLA(120)
0029 DATA BLA(121), BLA(122), BLA(123), BLA(124), BLA(125), BLA(126), BLA(127), BLA(128), BLA(129), BLA(130), BLA(131), BLA(132), BLA(133), BLA(134), BLA(135), BLA(136), BLA(137), BLA(138), BLA(139), BLA(140)
0030 DATA BLA(141), BLA(142), BLA(143), BLA(144), BLA(145), BLA(146), BLA(147), BLA(148), BLA(149), BLA(150), BLA(151), BLA(152), BLA(153), BLA(154), BLA(155), BLA(156), BLA(157), BLA(158), BLA(159), BLA(160)
0031 DATA BLA(161), BLA(162), BLA(163), BLA(164), BLA(165), BLA(166), BLA(167), BLA(168), BLA(169), BLA(170), BLA(171), BLA(172), BLA(173), BLA(174), BLA(175), BLA(176), BLA(177), BLA(178), BLA(179), BLA(180)
0032 DATA BLA(181), BLA(182), BLA(183), BLA(184), BLA(185), BLA(186), BLA(187), BLA(188), BLA(189), BLA(190), BLA(191), BLA(192), BLA(193), BLA(194), BLA(195), BLA(196), BLA(197), BLA(198), BLA(199), BLA(200)
0033 DATA BLA(201), BLA(202), BLA(203), BLA(204), BLA(205), BLA(206), BLA(207), BLA(208), BLA(209), BLA(210), BLA(211), BLA(212), BLA(213), BLA(214), BLA(215), BLA(216), BLA(217), BLA(218), BLA(219), BLA(220)
0034 DATA BLA(221), BLA(222), BLA(223), BLA(224), BLA(225), BLA(226), BLA(227), BLA(228), BLA(229), BLA(230), BLA(231), BLA(232), BLA(233), BLA(234), BLA(235), BLA(236), BLA(237), BLA(238), BLA(239), BLA(240)
0035 DATA BLA(241), BLA(242), BLA(243), BLA(244), BLA(245), BLA(246), BLA(247), BLA(248), BLA(249), BLA(250), BLA(251), BLA(252), BLA(253), BLA(254), BLA(255), BLA(256), BLA(257), BLA(258), BLA(259), BLA(260)
0036 DATA BLA(261), BLA(262), BLA(263), BLA(264), BLA(265), BLA(266), BLA(267), BLA(268), BLA(269), BLA(270), BLA(271), BLA(272), BLA(273), BLA(274), BLA(275), BLA(276), BLA(277), BLA(278), BLA(279), BLA(280)
0037 DATA BLA(281), BLA(282), BLA(283), BLA(284), BLA(285), BLA(286), BLA(287), BLA(288), BLA(289), BLA(290), BLA(291), BLA(292), BLA(293), BLA(294), BLA(295), BLA(296), BLA(297), BLA(298), BLA(299), BLA(300)
0038 DATA BLA(301), BLA(302), BLA(303), BLA(304), BLA(305), BLA(306), BLA(307), BLA(308), BLA(309), BLA(310), BLA(311), BLA(312), BLA(313), BLA(314), BLA(315), BLA(316), BLA(317), BLA(318), BLA(319), BLA(320)
0039 DATA BLA(321), BLA(322), BLA(323), BLA(324), BLA(325), BLA(326), BLA(327), BLA(328), BLA(329), BLA(330), BLA(331), BLA(332), BLA(333), BLA(334), BLA(335), BLA(336), BLA(337), BLA(338), BLA(339), BLA(340)
0040 DATA BLA(341), BLA(342), BLA(343), BLA(344), BLA(345), BLA(346), BLA(347), BLA(348), BLA(349), BLA(350), BLA(351), BLA(352), BLA(353), BLA(354), BLA(355), BLA(356), BLA(357), BLA(358), BLA(359), BLA(360)
0041 DATA BLA(361), BLA(362), BLA(363), BLA(364), BLA(365), BLA(366), BLA(367), BLA(368), BLA(369), BLA(370), BLA(371), BLA(372), BLA(373), BLA(374), BLA(375), BLA(376), BLA(377), BLA(378), BLA(379), BLA(380)
0042 DATA BLA(381), BLA(382), BLA(383), BLA(384), BLA(385), BLA(386), BLA(387), BLA(388), BLA(389), BLA(390), BLA(391), BLA(392), BLA(393), BLA(394), BLA(395), BLA(396), BLA(397), BLA(398), BLA(399), BLA(400)
0043 DATA BLA(401), BLA(402), BLA(403), BLA(404), BLA(405), BLA(406), BLA(407), BLA(408), BLA(409), BLA(410), BLA(411), BLA(412), BLA(413), BLA(414), BLA(415), BLA(416), BLA(417), BLA(418), BLA(419), BLA(420)
0044 DATA BLA(421), BLA(422), BLA(423), BLA(424), BLA(425), BLA(426), BLA(427), BLA(428), BLA(429), BLA(430), BLA(431), BLA(432), BLA(433), BLA(434), BLA(435), BLA(436), BLA(437), BLA(438), BLA(439), BLA(440)
0045 DATA BLA(441), BLA(442), BLA(443), BLA(444), BLA(445), BLA(446), BLA(447), BLA(448), BLA(449), BLA(450), BLA(451), BLA(452), BLA(453), BLA(454), BLA(455), BLA(456), BLA(457), BLA(458), BLA(459), BLA(460)
0046 DATA BLA(461), BLA(462), BLA(463), BLA(464), BLA(465), BLA(466), BLA(467), BLA(468), BLA(469), BLA(470), BLA(471), BLA(472), BLA(473), BLA(474), BLA(475), BLA(476), BLA(477), BLA(478), BLA(479), BLA(480)
0047 DATA BLA(481), BLA(482), BLA(483), BLA(484), BLA(485), BLA(486), BLA(487), BLA(488), BLA(489), BLA(490), BLA(491), BLA(492), BLA(493), BLA(494), BLA(495), BLA(496), BLA(497), BLA(498), BLA(499), BLA(500)
0048 DATA BLA(501), BLA(502), BLA(503), BLA(504), BLA(505), BLA(506), BLA(507), BLA(508), BLA(509), BLA(510), BLA(511), BLA(512), BLA(513), BLA(514), BLA(515), BLA(516), BLA(517), BLA(518), BLA(519), BLA(520)
0049 DATA BLA(521), BLA(522), BLA(523), BLA(524), BLA(525), BLA(526), BLA(527), BLA(528), BLA(529), BLA(530), BLA(531), BLA(532), BLA(533), BLA(534), BLA(535), BLA(536), BLA(537), BLA(538), BLA(539), BLA(540)
0050 DATA BLA(541), BLA(542), BLA(543), BLA(544), BLA(545), BLA(546), BLA(547), BLA(548), BLA(549), BLA(550), BLA(551), BLA(552), BLA(553), BLA(554), BLA(555), BLA(556), BLA(557), BLA(558), BLA(559), BLA(560)
0051 DATA BLA(561), BLA(562), BLA(563), BLA(564), BLA(565), BLA(566), BLA(567), BLA(568), BLA(569), BLA(570), BLA(571), BLA(572), BLA(573), BLA(574), BLA(575), BLA(576), BLA(577), BLA(578), BLA(579), BLA(580)

```


SIAPRT,FIN /TP19L2CK5/GR

0107 WRITE(6,100)

0108 WRITE(6,101)

0109 WRITE(6,102) (12,AHL(1),1,1,0)

0110 WRITE(6,126)

0111 WRITE(6,103) (2,APP(1),1,1,12)

0112 WRITE(6,104) (3,NGA(1),1,1,12)

0113 WRITE(6,105) (4,PT(1),1,1,16),EASR(12)

0114 WRITE(6,106) (5,VAR(1),1,1,6),SAVAR(12)

0115 WRITE(6,107) (6,AP S(1),1,1,3),FARNAS(12)

0116 WRITE(6,132)

0117 CALL BMT

0118 WRITE(6,109)

0119 WRITE(6,130) PCC1.PCC2

0120 WRITE(6,109)

0121 WRITE(6,131) PCC3,PCC4

0122 WRITE(6,110) (BACLAR(1),1,1,0)

0123 WRITE(6,111)

0124 WRITE(6,136) (CLD(1),1,1,2)

0125 DO 36 J=1,BACATS

0126 WRITE(6,112) (BACAL(J),BACAL(J),K),K=1,BACATS)

0127 96 CONTINUE

0128 WRITE(6,132)

0129 CALL BMT

0130 CALL BMT

0131 WRITE(6,113)

0132 DO 50 K=1,5ACATS

0133 WRITE(6,134) (BACAL(K),BACAL(L),BCVA(K,L),LL=1,BACATS)

0134 50 CONTINUE

0135 WRITE(6,135) (ATP

0136 IF(CLDUM.EQ.1) RETURN

0137 WRITE(6,129)

0138 CMTZANE

0139 DO 207 J=1,2

0140 K=1

0141 MLT=210

0142 WRITE(6,123) CMT

0143 WRITE(6,124)

0144 WRITE(6,125) (CPL(1),1,1,19)

0145 DO 208 J=1,11

0146 L=6

0147 DO 210 I=1,120

0148 PHATRI)=FLANK

0149 210 CONTINUE

0150 DO 212 N=1,19

0151 IF(CN.EQ.1) GO TO 206

0152 IF(TYPE(K),EQ,IN,LR,TYPE(K),EC,2) GO TO 207

0153 GO TO 205

0154 206 IF(TYPE(K),EQ,1,1,DR,TYPE(K),EC,3) GO TO 207

0155 GO TO 205

0156 PHATRI)=LRLC(K)

0157 PHATRI(1)=SLAS4

0158 PHATRI(2)=CLDST(20K)

0159 IF(CLDST(20K-1).EQ.1) GO TO 5

0160 PHATRI(2)=CLDST(20K-1)

0161 PHATRI(3)=CLDST(20K)

0162 205 CONTINUE

ORIGINAL PAGE IS OF POOR QUALITY

FORTAN IV-PLUS V02-51
01APRT.FTN /TRAISLECKX/NO

```

0163      KSK+1
0164      LSL*6
0165      CONTINUE
0166      WRITE(6,99) N(LINE,(P*ATS(1),1*6,127)
0167      N(LINE*LINE*10
0168      CONTINUE
0169      CRT*7*8
0170      CONTINUE
0171      WRITE(6,178)
0172      WRITE(6,161)
0173      WRITE(6,107) (PUI AEL(1),1*1,8)
0174      WRITE(6,127)
0175      WRITE(6,103) (PUPP(1),1*1,2)
0176      WRITE(6,104) (PUNX(1),1*1,12)
0177      WRITE(6,105) (PUCR(1),1*1,2),BLVAR(12)
0178      WRITE(6,106) (PUPA(1),1*1,5),BLVAR(12)
0179      WRITE(6,107) (PUPA(1),1*1,8),BLVANS(12)
0180      WRITE(6,112)
0181      CALL BNT
0182      WRITE(6,107)
0183      WRITE(6,110) PCL1,PCL2
0184      WRITE(6,109)
0185      WRITE(6,131) PCL61,PCL62
0186      WRITE(6,110) (BUCLAR(1),1*1,8)
0187      WRITE(6,111)
0188      WRITE(6,136) (CLUD(1),1*1,2)
0189      DO 20 J=1,BUCATS
0190      WRITE(6,112) PUI APL(J),(EUCAIN(J,K),K=1,BUCATS)
0191      CONTINUE
0192      WRITE(6,112)
0193      CALL BNT
0194      CALL BNT
0195      WRITE(6,113)
0196      DO 250 K=1,BUCATS
0197      WRITE(6,134) (BUCLAR(K),BUCLAR3(LL),RCVU(K,LL),LL=1,BUCATS)
0198      CONTINUE
0199      WRITE(6,135) RUT*7
0200      RETURN
0201      END

```

ORIGINAL PAGE IS
OF POOR QUALITY

84

PROGRAM SECTIONS

ATTRIBUTES

NUMBER NAME SIZE

1	SCDEF1	006000	1536
2	SPDATA	000004	3
3	SIDATA	001714	409
4	SVARS	000462	153
6	CBTAS	005122	1321
7	DUMMY	020004	2
8	CLCBM	000642	209
9	SEG	000204	2

	FRPL.CPA.LCL		
	FRPL.CBA.LCL		
	FRPL.CCA.LCL		
	FRPL.LEP.LCL		
	FRPL.VVF.GRL		
	FRPL.VVF.GRL		
	FRPL.VVF.GRL		

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------	------	------	---------

BIAPRT 1-000000

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
BACATS	102	6-003314	BATPT	102	6-003764	BLANK	101	4-000170	EUCATS	102	6-001556
CLADUM	102	7-000000	CLUCUM	102	7-003000	CHT	102	4-000450	CM	102	6-001550
L	102	4-003436	J	102	4-000443	JJ	102	4-000452	K	102	4-003442
LL	102	4-000440	H	102	4-000446	N	102	4-000456	NLIVE	102	4-000454
PASS	102	F-000000	PCCG1	104	6-003338	PCCG2	104	6-003342	PCC1	104	6-001326
PCLG1	104	6-003356	PCLG2	104	6-003362	PCL1	104	6-003346	PCL7	104	6-003352
TAPR0	102	F-000004	C	102	6-001554	TH3	102	4-000434	SLASH	102	6-000432

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
------	------	---------	------	------------

BACAIN	104	6-001570	000030	12 (3,2)
BACLAB	101	6-000220	000400	128 (8,8)
BACR2	102	6-003316	000010	4 (8)
BALABL	101	6-000142	000030	12 (12)
BAPP	102	6-001600	000010	4 (8)
BARANS	104	6-001020	000030	12 (12)
BAUNCA	102	6-001360	000060	24 (12)
BAVAR	104	6-001220	000060	24 (12)
BCVA	102	6-003366	000200	64 (8,8)
BCVJ	102	6-003566	000200	64 (8,8)
BUCAIN	104	6-000620	000400	128 (8,8)
BUCLAP	101	6-001550	000010	4 (8)
BUGER	102	6-000170	000030	12 (12)
BULABL	101	6-001570	000010	4 (8)
BUPP	102	6-000030	000010	12 (12)
BURANS	104	6-001440	000060	24 (12)
BUUNCA	102	6-000110	000030	12 (12)

FAPTRM IV-PLUS V02-51
 BIAPRT.FIN /BIAPRT.CCS/AR
 BVAR B04 6-00130C 00050 24 (12)
 CC3S B04 6-005112 00020 4 (2)
 CLAD B04 6-003772 00010 4 (2)
 CLASSY L01 6-001617 00042 209 (418)
 CLDPT L01 6-000070 00042 209 (416)
 CLUD F04 6-004002 00010 4 (2)
 CCL L07 4-000352 00046 19 (19)
 IPUF L01 6-000022 00021 3 (1)
 LPLED L01 6-002773 00021 164 (209)
 PR L01 4-000020 000170 60 (120)
 PVAR L01 6-000171 000170 60 (120)
 RC3S B04 6-002412 00000 256 (2,8,8)
 SEG-W L01 9-001000 00004 2 (4)
 TC3S P04 6-002012 00010 12 (2,8)
 TYPE LB: 6-002052 00021 184 (209)

LABELS

ADDRESS	ADDRESS	ADDRESS	ADDRESS	ADDRESS	ADDRESS	ADDRESS	ADDRESS	ADDRESS	ADDRESS
1-000054	2	1-000030	3	00	00	1-000144	5	00	1-001214
1-001294	10	00	12	00	00	00	20	00	00
3-000054	97	00	95	00	00	3-001454	201	00	3-002000
3-000094	102	3-000074	103	3-000119	3-000150	3-000150	204	3-001250	3-003204
3-000094	107	3-000014	108	3-000360	3-000410	3-000410	109	3-001450	3-001450
3-000094	112	3-000217	113	3-000534	00	00	114	3-000624	3-000624
3-000054	117	3-000230	118	3-000762	00	00	119	3-001034	3-001034
00	121	3-001114	123	3-001164	3-001190	3-001190	124	3-001254	3-001254
3-001302	127	3-001330	128	3-001374	00	00	129	3-001450	3-001450
3-001510	132	3-001510	133	00	00	3-001616	134	3-001642	3-001642
3-001550	212	1-003312	209	1-004030	00	00	206	1-003674	1-003734
00	212	00	250	00	00	00	297	00	00

FUNCTIONS AND SUBROUTINES REFERENCED

BMT

TOTAL SPACE ALLOCATED = 016400 X712

NO CPP INSTRUCTIONS GENERATED

BIAPRT.LPIBTPHY

0001 /TR16LGRS/LR
SUBROUTINE CPPEP(19UF,R,FILE)

0002 IMPLICIT INTEGER(2-7)

0003 INTEGER(2) ISTAT(2),IPRM(6)

0004 EQUIVALENCE(ISTAT(1),I8)

0005 BYTE I8(2)

0006 LOGICAL(1) IPUF(1)

0007 DIMENSION IMDAT(1)

0008 DIMENSION IA(37)

0009 DATA XDEV/2X17

0010 DATA XDEV/2XMT/

0011 PSXPDMR

0012 IF(RECORD(87,0) GE TO 10

0013 ERCT=0

0014 IF(RECORD(60,-1) GE TO 3

0015 CONTINUE

0016 FILE=C

D WRITE(5,149)

C TYPE 49

49 FORMAT(10X,' TYPE M OR K FOR TAPE DEVICE CODE.'/)

READ(1,51,END=99) IA

0018 ACCEPT 51,IA

0019 IF(IA(1).EQ.'M') GO TO 13.

0020 IF(IA(1).EQ.'K') GO TO 14

0021 CONTINUE

0022 WRITE(6,97)

0023 FORMAT(10X,' CARD ERROR '//)

0024 WRITE(6,97)

0025 STOP

0026 IDEV=MDEV

0027 GO TO 15

0028 IDEV=XDEV

0029 CONTINUE

D WRITE(5,150)

C TYPE 50

50 FORMAT(10X,' TYPE TAPE UNIT NUMBER = 0 OR 1.'/)

READ(1,51,END=99) IA

0031 ACCEPT 51,IA

0032 FORMAT(37A2)

0033 IF(IA(1).EQ.'0') GO TO 18

0034 IF(IA(1).EQ.'1') GO TO 19

0035 GO TO 99

0036 IUNT=0

0037 GO TO 20

0038 IUNT=1

0039 CONTINUE

0040 ILM=9

0041 IDS=0

0042 ISR=0

0043 CALL ASNUN(ILUN,IDEV,IUNT,ICD)

0044 IF(IDS.LY.0) GO TO 1

0045 CALL GETADR(IPRM,IBUF)

0046 CONTINUE

0047 CALL OI0(2400,ILUN,1,ISTAT,IPRM,ISR)

0048 IF(ISR.LY.0) GO TO 2

0049 IF(RECORD(60,-1) GE TO 11

0050 IPRM(2)=3600

ORIGINAL PAGE IS
OF POOR QUALITY

CDRED,PTN

77918BLOCKS/HR

0051 10

CONTINUE

0052

ISW=0

0053

RECORD=RECORD+1

0054

CALL Q10(*1000,ILUN,1,,ISTAT,IPR,ISW)

0055

IF(ISW.LT.0) GO TO 3

0056

CALL WAITER(1,IDS)

0057

IF(IDS.LT.0) GO TO 4

C

WRITE(6,101) (IB(I),IB(2),ISTAT(2))

0058 101

FORMAT(1H,10X,2I4,3X,15,2X,'BYTES TRANSFERED')

0059

ERCODE=IB(1)

0060

IF(ERCODE.LT.0) GO TO 5

0061

FF=0

0062

DO 800 JJ=1,720

0063 798

IF(IBUF(JJ).LT.0) GO TO 799

0064 800

CONTINUE

0065

RETURN

0066 799

IBUF(JJ)=11

0067

IF(FF.EQ.1) GO TO 798

0068

FF=1

0069

WRITE(6,1000)

0070 1000

FORMAT(1H '***** THIS SEGMENT CONTAINS BAD DATA *****')

0071

GO TO 798

0072 1

CONTINUE

0073

WRITE(6,100) IDS

0074 100

FORMAT(1H ' ASLUN CALL DSW = ',I6)

0075

STOP

0076 2

CONTINUE

0077

WRITE(6,200) ISP

0078 200

FORMAT(1H ' REWIND DSW = ',I6)

0079

STOP

0080 3

CONTINUE

0081

WRITE(6,300) ISW

0082 300

FORMAT(1H ' READ Q10 DSW = ',I6)

0083

STOP

0084 4

CONTINUE

0085

WRITE(6,400) IDS

0086 400

FORMAT(1H ' WAIT DSW = ',I6)

0087

STOP

0088 5

CONTINUE

0089

IF(ERCODE.EQ.-10) FILE=FILE+1

0090

IF(ERCODE.EQ.-10) RETURN

0091

IF(ERCODE.NE.-4) GO TO 7

0092

ERCNT=ERCNT+1

0093

RETURN

0094 11

CONTINUE

0095

WRITE(6,700)

0096 700

FORMAT(' ')

0097

IF(ERCNT.EQ.0) RETURN

0098

WRITE(6,600) ERCNT

0099 600

FORMAT('1',10X,' TAPE ERRORS ENCOUNTERED = ',I5)

0100

RETURN

0101 7

CONTINUE

0102

WRITE(6,500) ERCODE

0103 500

FORMAT(1H, ' I/O STATUS BLOCK ERROR CODE = ',I6)

0104

STOP

0105

END

ORIGINAL PAGE IS
OF POOR QUALITY

CORED.FTH /TRIPLOCKS/HP
PROGRAM SECTIONS

ATTRIBUTES

SIZE

NUMBER	NAME	SIZE	ATTRIBUTES
1	ICD01	392	FILE,DATA,LCL
2	DATA	0	FILE,DATA,LCL
3	DATA	197	FILE,DATA,LCL
4	SYMS	59	FILE,DATA,LCL

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
CORED		1-000000						

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
ENTRY	102	4-000142	ENDCDS	102	4-000160	FF	102	4-000162
IDS	102	4-000152	ILUN	102	4-000150	ISR	102	4-000154
JJ	102	4-000164	PREV	102	4-000136	R	102	4-000130
						SECED	102	4-000140
						YMET	102	4-000134

ARRAYS

SIMENSIPAS

NAME	TYPE	ADDRESS	SIZE
IA	102	4-000022	00112 (37)
IP	101	4-000009	00132 (2)
IPUF	101	4-000032	00061 (1)
JMATA	102	4-000020	00002 (1)
IPRM	102	4-000004	00004 (2)
ISTAT	102	4-000060	00004 (2)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
1	1-000774	2	1-00100C	3	1-001104	4	1-001154
6	1-000472	7	1-001354	10	1-000544	11	1-001204
13	1-000232	14	1-000250	15	1-000204	18	1-000152
20	1-000402	40	00	50	00	51	3-000024
90	1-000156	100	3-000016	171	00	200	3-000142
400	3-000216	500	3-000316	600	3-000250	700	3-000424
700	1-000712	800	00	1000	3-000030		

FUNCTIONS AND SUBROUTINES REFERENCED

ASNLUM	GETADR	OIP	WAITFR

FORTRAN IV-PLUS V02-51

08105155

US-PAR-7A

PAGE 4

CDREN,FTM

/TR1BLOCKS/42

TOTAL SPACE ALLCATED = 002314

614

•
•
NB FPP INSTRUCTIONS GENERATED

•
•
CDREN,LPICDREN

COINTEGRATION	7791-LOGS/5	SUBSCRIPTION	7791-LOGS/5
0001	IMPLICIT	17612	51216(12)
0002	EXPLICIT	17612	51216(12)
0003	IMPLICIT	17612	51216(12)
0004	EXPLICIT	17612	51216(12)
0005	IMPLICIT	17612	51216(12)
0006	EXPLICIT	17612	51216(12)
0007	IMPLICIT	17612	51216(12)
0008	EXPLICIT	17612	51216(12)
0009	IMPLICIT	17612	51216(12)
0010	EXPLICIT	17612	51216(12)
0011	IMPLICIT	17612	51216(12)
0012	EXPLICIT	17612	51216(12)
0013	IMPLICIT	17612	51216(12)
0014	EXPLICIT	17612	51216(12)
0015	IMPLICIT	17612	51216(12)
0016	EXPLICIT	17612	51216(12)
0017	IMPLICIT	17612	51216(12)
0018	EXPLICIT	17612	51216(12)
0019	IMPLICIT	17612	51216(12)
0020	EXPLICIT	17612	51216(12)
0021	IMPLICIT	17612	51216(12)
0022	EXPLICIT	17612	51216(12)
0023	IMPLICIT	17612	51216(12)
0024	EXPLICIT	17612	51216(12)
0025	IMPLICIT	17612	51216(12)
0026	EXPLICIT	17612	51216(12)
0027	IMPLICIT	17612	51216(12)
0028	EXPLICIT	17612	51216(12)
0029	IMPLICIT	17612	51216(12)
0030	EXPLICIT	17612	51216(12)

ORIGINAL PA... IS
OF POOR QUALITY

PROGRAM SECTIONS

ATTRIBUTES

NUMBER NAME SIZE

1	SCDF1	00034	110
3	3IPATA	00012	5
4	2VANS	00010	4
6	CHIAS	00122	1321
7	DUMMY	00004	2
8	CLC0V	00162	209

ENTRY POINTS

NAME TYPE ADDRESS NAME TYPE ADDRESS NAME TYPE ADDRESS

CZARED 1-00000

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
BACATS	102	6-003766	BUCATS	102	6-001536	WUTAT	102	6-003273
CLUDUM	102	7-000302	DU	102	6-001552	TV	102	6-000000
J	102	4-000304	PCC31	104	6-003310	PCC31	104	6-003326
PCC2	104	6-003332	PCL01	104	6-003356	PCL1	104	6-003392
TC	102	6-001554	FCL02	104	6-003346	3CLP	104	6-003392

ARRAYS

DIMENSIONS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
A	104	6-001520	00070	12 (3,2)
BACAIN	104	6-000220	00040	120 (8,8)
BACLAN	101	6-003316	00010	4 (8)
BAC0P	102	6-000140	00030	12 (12)
BALABL	101	6-001600	00010	4 (8)
BAPPP	102	6-000000	00030	12 (12)
BARANS	104	6-001360	00090	24 (12)
BAUN02	102	6-000060	00030	12 (12)
BAVAR	104	6-001220	00060	24 (12)
BCVA	102	6-003366	00020	64 (8,8)
BCVU	102	6-003366	00020	64 (8,8)
BUCAL4	104	6-000420	00040	120 (8,2)
BUCLAG	101	6-001560	00010	4 (8)
BUCR01	102	6-000170	00030	12 (12)
BULABL	101	6-001570	00010	4 (8)
BU0P0P	102	6-000030	00030	12 (12)
BURANS	104	6-001440	00060	24 (12)
BUVAC0	102	6-000110	00030	12 (12)
BUVAR	104	6-001300	00060	24 (12)
CC0NS	104	6-005112	00010	4 (2)
CLAD	104	6-003772	00010	4 (2)
CLASFY	101	6-001610	00062	209 (418)

C-2

92

05-PAR-78

PS106107

FORTRAN IV-PLUS V02-51

CLDZY	LOC	6-024000	003542	279	(418)
CLUC	004	6-004002	000010	4	(2)
LBLEC	LOC	6-032773	000321	304	(209)
RC2:5	304	6-004112	001000	256	(2.618)
TC2:5	404	6-004012	000100	32	(2.8)
TYPE	LOC	6-022452	000321	104	(209)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
-------	---------	-------	---------	-------	---------

9	00	10	00	100	3-000004
				101	3-000000

TOTAL SPACE ALLOCATED = 000346 1651

NO PFP INSTRUCTIONS GENERATED

COPYED,LP1=C1NRE7

```

0001 /TRIAL/CKS/JP
0002 S/BRZUTIME W/YYI (PIL)
0003 IMPLICIT INTEGER (A-Z)
0004 LOGICAL A-I, M-Z (12), P (11)
0005 DATA MSD/1MM, 1VE, 1HA, 1MN, 1M .1M, 1MS, 1HT, 1H, 1MD, 1MG, 1MV/
0006 D3 1 121, 13
0007 PVL (1)MSD (1)
0008 RETURN
0009 END

```

94

FORTRAN IV-PLUS V02-51
MOTIL.PT. /TP1818CK5/HR
PROGRAM SECTIONS

ATTRIBUTES

NUMBER	NAME	SIZE
1	SCODE1	000102 33
3	PIRATA	000012 5
4	SWARS	000016 7

NAME	ADDRESS	TYPE	ADDRESS	NAME	TYPE	ADDRESS
RA:ICRN.LCL						
RA:ICRN.LCL						
RA:ICRN.LCL						

ENTRY PRINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
MOTIL		1-000000									

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
I	I=2	4-000014						

ARRAYS

DIMENSIONS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
MSE	L=1	4-200000	000014	6 (12)
PTL	L=1	F-000002	000001	3 (1)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS

TOTAL SPACE ALLOCATED = 000132 45

NO FPP INSTRUCTIONS GENERATED

MOTIL.LP:MOTIL

PAGE 1

02160116 0922AS-7A

DESTRAY IV-PLUS V02-51

FRAME.FTN

791HL7CRS/M9

SUBROUTINE FMINC(FL,DM)

IMPLICIT INTEGER(*=2)

LOGICAL FL,FL(1),FL(2),FL(10)

DATA FIELC/MF,1M,1ME,1ML,1F,1B /

CE 11 11316

PRINT*,FIELC(1)

PRINT*,FIELC(1)

CONTINUE

RETURN

END

11

0001

0002

0003

0004

0006

0007

0009

0010

ORIGINAL PAGE IS
OF POOR QUALITY

96

7

FNAME,PT: /TRIPLOC/SNR
PRGRM SECTION:5

NUMBER	NAME	SIZE	ATTRIBUTES
1	ICODE1	900132	45 RPL,COR,LCL
3	SICATA	000224	10 RPL,COR,LCL
4	SVARS	00010	4 RPL,COR,LCL

ENTRY PRINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------	------	------	---------

FNAME		1-000000									
-------	--	----------	--	--	--	--	--	--	--	--	--

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------	------	------	---------

I1		102	4-000006								
----	--	-----	----------	--	--	--	--	--	--	--	--

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
------	------	---------	------	------------

FIELD	L01	4-000000	000006	3 (6)
FLO	L01	F-000002	000001	0 (1)
PH	L01	F-000004	000001	0 (1)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
-------	---------	-------	---------	-------	---------	-------	---------

I1							
----	--	--	--	--	--	--	--

TOTAL SPACE ALLOCATED = 700166 59

NO FOR INSTRUCTIONS GENERATED

FNAME,PT:FNAME

0001 /TR92/CRS/MP
 0002 SUPROUTINE KYT
 0003 IMPLICIT INTEGER(0-2)
 0004 CRX22/DCM7/L1:EPAGE
 0005 LINES(1,5-1
 0006 IF(1,5,11, 07) 5, 12 10
 0007 PAGEPAGE-1
 0008 LINES
 0009 WRITE(6,100) PAGE
 0010 11 RETURN
 0011 100 FORMAT(11,26X,'CRX22/DCM7/INTERFACE TAPE PRINTOUT',14X,'PAGE',11/)
 0012 END

98

FORTRAN IV-PLUS V02-51
KNTALPH /TRIPLECKS/LR

PROGRAM SECTIONS

ATTRIBUTES

NUMBER NAME SIZE

3	SCODE1	000076	31	SAJLCCAL.LCL
3	SIDATA	000092	25	SAJLCCAL.LCL
6	PC17	000004	2	SAJLCCAL.LCL

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------

KNT 1-000000

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------

LINE 102 6-000000 PAGE 102 6-000002

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
-------	---------	-------	---------	-------	---------	-------	---------

10 1-000074 100 3-000000

TOTAL SPACE ALLOCATED = 000164 58

NO FPP INSTRUCTIONS GENERATED

KNTALPHKNT

```

/TP1919CYS/19
SUPPORTIVE MEANS(D,P,=CHK,FLG)
EXPLICIT IVEVE(=2)
L3IGC(=1)FLD(=1),FLG(=1),PT
L2TICAL(=1)BLK
DATA DT/14 /
DATA BLK/14 /
OFF(=1)
IF(FL(=1),ED,=1) OFF(=1)
K21
D2 3 J21,NCB
M2(=1)J21
PM(=1)FLJ(=FF=1)
OM(=2)FLD(=FF=2)
IF(=1)M(=1),NE,=1) SE TP 10
PM(=1)BLK
IF(=1)M(=2),ED,=1) PM(=1)J21BLK
OM(=3)FLD(=FF=3)
PM(=4)BLKPT
PM(=5)FLC(=FF=4)
PM(=6)FLC(=FF=5)
PM(=7)BLK
PM(=8)FLC(=FF=6)
PM(=9)FLC(=FF=7)
IF(=1)M(=9),NE,=1) SE TP 11
PM(=10)BLK
IF(=1)M(=9),ED,=1) PM(=11)BLK
OM(=10)FLD(=FF=8)
PM(=11)BLKPT
PM(=12)FLD(=FF=9)
OFF(=1)M(=9)
GE(=1)LINE
NET(=9)
E22

```

22

FORTRAN IV-PLUS V02-51
MEAN.PFN /T/HL3C/S/20
PROGRAM SECTIONS

NUMBER	NAME	SIZE	ATTRIBUTES
1	SCDF1	00514	166
2	SI2VA	00536	18
3	IV4VC	00012	5
4	STEPS	00002	1

ENTRY PRINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
MEAN		1-000000									

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
BL1	L01	4-000001	J	L02	4-000006	K	L02	4-000014			
ZFC	L02	4-000012	PT	L01	4-000000						

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
FL0	L01	F-000020	00001	0 (1)
FL3	L01	F-000010	00001	0 (1)
PH	L01	F-000030	00002	2 (1)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
J	00	L1	1-000260	L1	1-000420		

TOTAL SPACE ALLOCATED = 00056 187

NO FPP INSTRUCTIONS GENERATED

MEAN.LPIMEAN

CFRTRAN IV-PLUS V22-51

0001 /791PL0645/42
 0002 SUBROUTINE OPDCL,PP,F,G)
 0003 I=POLICY INTRSE(1-7)
 0004 L=SIGCALO1 FIDIAL,PP(1),24(2),FLE(1)
 0005 DATA PAW/INT(1)/
 0006 IF(FLE(1).EQ.0) PFF=3
 0007 PPA3=PPAR(1)
 0008 D2 1 181.5
 0009 PPA3)=FID(1-DEF1
 0010 CPVINDL
 0011 PPA3=PPAR(2)
 0012 RETN
 0013 END

100

NUMBER	NAME	SIZE	ATTRIBUTES
1	ICD01	60216	001,CPL,LCL
2	SIDATA	02024	001,CPL,LCL
4	SVARS	20000	001,CPL,LCL

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
PP0		1-000020									

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
1	102	4-000034	001	102	4-000002						

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
FLD	101	F-000020	000001	0 (1)
FLG	101	F-000000	000001	0 (1)
PAR	101	F-000000	000002	1 (2)
PP	101	F-000000	000001	0 (1)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
1		00					

TOTAL SPACE ALLOCATED = 00262 10

NO FPP INSTRUCTIONS GENERATED

REP.LPT000

OPPRAM IV-PLUS 40-21

SWAG.FTV /PRIP120487.9

0001 SUBROUTINE SWAGE(FL,P,FLG)

0002 LOGICAL INTENT(IN)

0003 LOCALS FL(1),P(1),SWAG(10),FLG(1)

0004 DATA SWAG(1:5),P(1:5),SWAG(6:10),FLG(1)

0005 P(1:5)

0006 IF(FL(1).E.Q.0) P(1:5) = P(1:5)

0007 P(1:5) = P(1:5)

0008 P(1:5) = P(1:5)

0009 11 CONTINUE

0010 RETURN

0011 END

404

FORTRAN IV-PLUS V02-51
SPACELEFT / TOTAL DEFS/MR... *R1C6:12 09-PAR-79

PROGRAM SECTIONS

NUMBER	NAME	SIZE	ATTRIBUTES
1	SCDEF1	000202	65
3	LDATA	000036	15
4	SVARS	000012	5

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
SVAVE		1-000000						

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
11	102	4-030010	OFF	102	4-000006			

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
FLD	101	F-000000	200001	2 (1)
FLS	101	F-000000	000001	0 (1)
PH	101	F-000000	000001	0 (1)
SUBCL	101	4-000000	000005	3 (6)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
11	00						

TOTAL SPACE ALLOCATED = 000252 RS

NO FPP INSTRUCTIONS GENERATED

SPACELEFT/PROGRAM

ORIGINAL PAGE IS OF POOR QUALITY

105

QFCSTRAN IVOPLUS V02051

STAMP.FTN /FRIECLS/W

0001 SUPPLY TIME STAMP(1,2,F)

0002 IMPLICIT INTEGRALS(2)

0003 LOGICALS(1,1,JE(1))

0004 CPMEN /PC 1/LIST,PAGE

0005 LINES(6)

0006 CALL W/T

0007 IF(1,F(1),E(1,F)) GO TO 8

0008 WRITE(6,20)

0009 202 F2PVAL(5),S,NSL,SS STATISTICS RECORD)

0010 33 7 9

0011 X91E(6,2,1)

0012 201 F2PVAL(5) (FIELD STATISTICS RECORD)

0013 C,VALUE

0014 WRITE(6,202) (1,UF(1),N3,4)

0015 F2PVAL(5),RECON SEC,NSL,2A)

0016 X91F(6,203) (1,UF(1),N3,7)

0017 F2PVAL(5),S,NSL,SUBCLASSES,3A)

0018 WRITE(6,204) (1,UF(1),N3,9)

0019 F2PVAL(5),N,3,SECLASS SETS,1,2A)

0020 WRITE(6,205) (1,UF(1),N3,10,11)

0021 F2PVAL(5),N,3,CHARACTERS,1,2A)

0022 LINES

0023 RETURN

0024 END

FURTRAY IVAPUS V32-51
SYMP.PTN /TRISLACKS/LR
PROGRAM SECTIONS

19106117 09-PAR-70

NUMBER	NAME	SIZE	ATTRIBUTES
1	SCD51	300460	152
3	SDATA	300256	R7
4	SWAPS	300002	1
6	PCY	300004	2

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
SYMP		3-000000						

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
LINE	102	6-000000	N	102	4-000000	PAGE	102	5-000000

NAME	TYPE	ADDRESS	SIZE	TYPESTRTS
IRUF	L01	F-000002	000001	0 (1)

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
203		3-000125	204		3-000156	200		3-000200
						205		3-000210

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
203		3-000125	204		3-000156	200		3-000200
						205		3-000210

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
203		3-000125	204		3-000156	200		3-000200
						205		3-000210

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
203		3-000125	204		3-000156	200		3-000200
						205		3-000210

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
203		3-000125	204		3-000156	200		3-000200
						205		3-000210

08100142 09-MAR-79 PAGE 3

0001 SUBROUTINE INT

0002 IMPLICIT INTEGER(*=2)

0003 CHARACTER*20 LINEPAGE

0004 LINE=1

0005 IF (LINE.LT. 5) GO TO 1

0006 PAGEPAGE=1

0007 LINE=0

0008 WRITE(3,100)

0009 10 RETURN

0010 100 FORMAT('1.71X, ')

0011 END

10000000
10000000

10000000
10000000

108

ENTRPTN /TOTALLOCKS/HZ

PROGRAM SECTIONS

NUMBER	NAME	SIZE	ATTRIBUTES	
1	ICDDE1	00006	27	PAR,ICEN,LCL
3	SIGATA	00014	6	PAR,ICEN,LCL
6	PCNT	00004	2	PAR,ICEN,LCL

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------

BAT 1-00000

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------

LINE 102 6-00000 PAGE 102 6-00000

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
-------	---------	-------	---------	-------	---------

10 1-000064 1001 3-000100

TOTAL SPACE ALLOCATED = 00106 35

% FPP INSTRUCTIONS GENERATED

ENT,LP,IDENT

85

```

0001 SUBROUTINE CPIP((FLD,CLASS,PI,PC)
0002 EXPLICIT INTEGER(1-7)
0003 LOGICAL(1) FL(1),CLASS(1)
0004 D P 181.9
0005 CLASS(1)=FL(1)
0006 CONTINUE
0007 DECIDE(S,100,FL(1)) P1
0008 P=47(15)
0009 DECIDE(S,100,FL(1)) P2
0010 RETN(P)
0011 END

```

110

PROGRAM SECTIONS

NUMBER	NAME	SIZE	ATTRIBUTES
1	CODE1	00024	74
3	DATA	00001	12
4	SWRS	00002	1

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
CPIPV		1-00000									

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
1	102	4-00000	PI	102	F-000006	PR	102	F-000010			

ARRAYS

NAME	TYPE	ADDRESS	SIZE	DIMENSIONS
CLASS	101	F-000004	000001	3 (1)
FLD	101	F-000002	000001	3 (1)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
2	00	100	3-000010				

TOTAL SPACE ALLOCATED = 000250 87

V2 FPP INSTRUCTIONS GENERATED

CPIPV:PI:CP:PI

SECRETAN IV-PLUS X02-51

0001

71913LPCRS/MP

SUBROUTINE MV(FL,CH,MC)

IMPLICIT I-INTEGER(2)

LOGICAL FL,CH,MC,DEL

DE R JELSON

RETURN

END

112

F2PTRAJ IVAPLUS V09051
VAPLTX /TRIPLOCKS/HR
PARTIAL SECTIONS

SYMBOLS

NUMBER NAME SIZE

1	IC001	000124	42	PARALLEL
3	IC002	010024	10	PARALLEL
4	IC003	000002	1	PARALLEL
5	IC004	000002	1	PARALLEL

ENTRY POINTS

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------

1-000000

VARIABLES

NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS	NAME	TYPE	ADDRESS
------	------	---------	------	------	---------	------	------	---------

J I02 4-000000 NC I02 F-000000

ARRAYS

NAME	TYPE	ADDRESS	SIZE	SYMBOLS
------	------	---------	------	---------

FLD L01 F-000000 00001 U (1)
PR L01 F-000000 00001 S (1)

LABELS

LABEL	ADDRESS	LABEL	ADDRESS	LABEL	ADDRESS
-------	---------	-------	---------	-------	---------

0 00

TOTAL SPACE ALLOCATED = 000194 54

NO FPP INSTRUCTIONS GENERATED

APPENDIX B

The values D_w and D_s are computed as follows:

$$D_w = C_1 + \sum_{p=1}^a \sum_{q=1}^b R_{pq} W_p W_q + \sum_{p=1}^a T_p W_p \text{ or } 0 \text{ if Category } W \text{ not used}$$

$$D_s = C_2 + \sum_{p=1}^a \sum_{q=1}^b U_{pq} S_p S_q + \sum_{p=1}^a V_p S_p \text{ or } 0 \text{ if Category } S \text{ not used}$$

where $C_1, C_2, \{R_{pq}\}_{p=1}^a \{q=1}^b, \{T_p\}_{p=1}^a, \{U_{pq}\}_{p=1}^a \{q=1}^b, \text{ and } \{V_p\}_{p=1}^a$ are constants, supplied by user

W_1 = Bias corrected estimate for Category W

S_1 = Bias corrected estimate for Category S

W_2 = Machine estimate for Category W

S_2 = Machine estimate for Category S

W_3 = Random estimate for Category W

S_3 = Random estimate for Category S

W_4 = Variance of Bias corrected estimate of W

S_4 = Variance of Bias corrected estimate of S

$$W_5 = \frac{W:W}{W:W + S:W + N:W}$$

$$S_5 = \frac{S:S}{W:S + S:S + N:S}$$

$$W_6 = S_6 = \text{PCC1}$$

$$W_7 = S_7 = \frac{N:N}{W:N + S:N + N:N}$$

$$W_8 = S_8 = \text{PCC2}$$

APPENDIX C

a. Constants* for D_w calculations

Card 1 contains C_1 as follows:

blanks or XXX.XXXXX in first 10 columns only

Card 2 contains T_1 thru T_8 as follows:

blanks or XXX.XXXXX for each entry. A maximum of 10 columns wide starting in columns 1, 11, 21, etc.

Card 3 thru 10 contain $R_{1,1}$ to $R_{8,8}$. $R_{1,1}$ thru $R_{1,8}$ on card 3, $R_{2,1}$ thru $R_{2,8}$ on card 4, etc. Format same as card 2.

b. Constants* for D_s calculations

Card 1 contains C_2 as follows:

blanks or XXX.XXXXX in first 10 columns only

Card 2 contains V_1 thru V_8 as follows:

blanks or XXX.XXXXX for each entry. A maximum of 10 columns wide starting in columns 1, 11, 21, etc.

Card 3 thru 10 contain $U_{1,1}$ to $U_{8,8}$. $U_{1,1}$ thru $U_{1,8}$ on card 3, $U_{2,1}$ thru $U_{2,8}$ on card 4, etc. Format same as card 2.

*List of test cards on the following page.