28234-6028-RU-00

(NASA-CR-147756) LACIE PERFORMANCE PREDICTOR FINAL OPERATIONAL CAPABILITY PROGRAM DESCRIPTION, VOLUME 2 (TRW SYSTEMS GROUP) 394 P HC $10.75 CSCL 02C

N76-25631

NASA CR-147756

UNCLAS
G3/43 42117

LACIE PERFORMANCE PREDICTOR FINAL OPERATIONAL CAPABILITY PROGRAM DESCRIPTION

VOLUME II

MAY 1976

Prepared for

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Lyndon B. Johnson Space Center
Houston, Texas

Contract Number NAS-9-14547

TRW
SYSTEMS GROUP
ONE SPACE PARK · REDONDO BEACH · CALIFORNIA 90278

This document, in three volumes, describes the FOC version of the LACIE Performance Predictor produced under Contract NAS9-14547. NASA/JSC Contract Technical Monitor is I. D. Browne, Earth Observations Division.

Table of Contents

		Page
Volume I		
Book I	EPHEMS	1
Book II	GRID	111
Book III	LUMP	137
Book IV	SEE	254
Volume II		
Book V	SAGE	1
Book VI	SACS	107
Book VII	POUT	202
Volume III		
Book VIII	LEM	1

SAGE BOOK V

Table of Contents

		Page
Part I	Problem Description	1
Part II	Common Block Definitions	31
Part III	List of Subroutines and Subroutine Call Structure	43
Part IV	Subroutine Descriptions and Flowcharts	47
Part V	Subroutine Listings	70

PART I

PROBLEM DESCRIPTION

Problem Description for SAGE Program

1.0 SCOPE

1.1 PROGRAM CAPABILITIES

Given the swath table files, the segment set for one country and cloud cover data, this program determines how many times and under what conditions each segment is accessed by satellites as defined on the swath table. The program writes a record for each segment on the segment reference data file which contains the pertinent acquisition data. On option, a utility function can be performed to generate the weather data file from a NASA supplied tape. This option can be exercised prior to the segment processing or as a stand-alone utility.

1.2 METHOD OF PROGRAM DEVELOPMENT

This program will be developed in FORTRAN and use of machine dependent routines will be kept to a minimum. The direct access file processing is the only known problem in this area. Modular programming techniques will be used throughout to make program development, modifications and debugging easier.

1.3 OPERATIONAL ASSUMPTIONS

- A maximum of 426 days can be specified in NODAY--the number of days in the run.
- It is assumed that only 1 year of data, starting January 1st, is to be generated on the weather data file (366 days).
- The use of the weather file will be such that the look-up date will be determined modulo 1 year, i.e., given a run day of 400, then the look-up day will be 400 - 366 + 1 = 35.
- Only 1 case is run at a time.
- Only 1 country may be run at a time.
- The segment reference data file and the weather file are regenerated each time the program is run. There is no update capability.
- A maximum of 150 acquisitions for any one segment is allowed.

2.0 INPUT

There are four files and 1 card set required for input to the program if the segment reference data file is to be generated. If just the weather file is to be generated, then only the header card and NASA weather tape is required. If both modes of operation are exercised, the NASA weather tape would replace the standard weather file as an input file.

Mode 1 Inputs - Generate weather file only.

1. Header card with option set to 1.
2. NASA weather tape.

Mode 2 Inputs - Generate segment reference data file only.

1. Card set as defined below with option set to 2.
2. Swath table file.
3. Segment ID file.
4. Swath reference file.
5. Weather file.

Mode 3 Inputs - Generate both files.

The same as for Mode 2 except Item 5 is replaced with the NASA weather tape and the option flag is set to 3.

2.1 CARDS

2.1.1 List of Data Quantities

See Input Data Description sheet on Pages 4 and 5.

2.1.2 Card Formats

"SAGE" is punched in card columns 75-78 of all cards. A sequence number is punched in card columns 79-80.

See Data Card Formats sheet on Page 6 for details.

2.1.3 Deck Set Up

1. Header card - sequence 01
2. Data card - sequence 02

List of Data Quantities

Name	Symbol	Dimension	Nominal Value	Range	Units	Description
IHEAD	-	18	Blanks	-	-	72 character case header which prints out at the top of every page.
ICASE	-	1	0	0-9999	-	4 digit case no. to identify the printed output and the segment reference data file.
ICSESW	-	1	0	0-9999	-	Case number identifying the swath table and reference files.
ICSESG	-	1	0	0-9999	-	Case number identifying the segment ID file.
ISTIME	-	3	0	-	-	ISTIME(1) - (Year - 1900) ISTIME(2) - month no. ISTIME(3) - day no. Run start date
NDAYS	-	1	426	1-426	Days	No. of days to process in the run.
IOPT	-	1	0	0-2	-	Program run option. 0 - run sage only 1 - run weather file generation utility only 2 - run both the utility and SAGE
IVEH	-	2		0-2	-	List of vehicle numbers to process in this run.
IREPT	-	1	F	T or F	-	Flag to indicate whether access report is to be produced. T - yes, F - no.
DECR	-	1	0	0-100	Kilom.	Swath decrement
RAND	-	1	1.0	-	-	Random no. seed used to obtain daily weather data. Must be odd integer.
IGRDN	-	1	1600	1-16000	-	No. of grid points to be written on the weather file (debug only)

List of Data Quantities (cont'd)

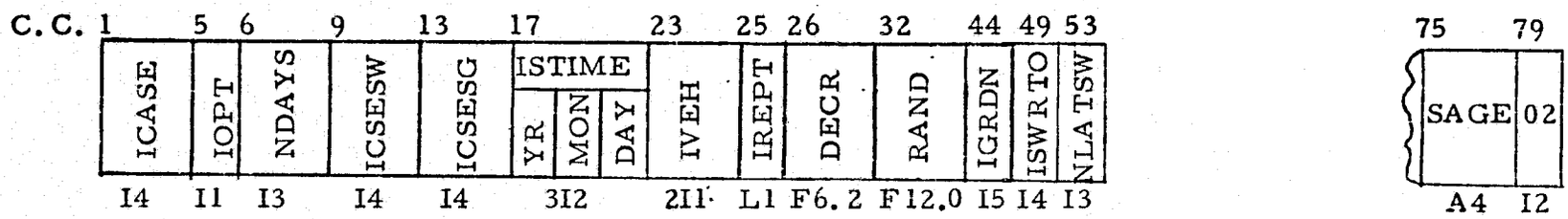
Name	Symbol	Dimension	Nominal Value	Range	Units	Description
ISWRTO*	-	1	1101	4-1101	-	The number of records written on the swath table file. This is equal to (NO OF DAYS FOR EACH VEHICLE) + 2 + NO OF VEHICLES
NLATSW	-	1	100	1-100	-	The number of latitude points written on the swath table record

*Not needed for UNIVAC

Data Card Formats

Header Card

The header information is entered in C. C. 1-72; "SAGE" is entered in C. C. 75-78, and 01 is entered in C. C. 79-80.



2.1.4 Rules for Entering Data on Cards

2.1.4.1 General

1. Integers must be entered right justified.
2. F format numbers must have the decimal point present, i.e., F6.2 - XXX.XX.
3. The card sequence numbers in C.C. 79-80 must be present in all data cards.

2.1.4.2 Rules for Specific Fields

- ICSESW must match the case number on the swath reference and swath table files.
- ICSESG must match the case number on the segment ID file.
- The start time in ISTIME must not be less than the earliest vehicle start date on the swath table file. (Note, if only 1 of the vehicles is to be processed, then ISTIME will be checked against that vehicle's start time only. If ISTIME is not input, it will be assumed that the earliest vehicle start date is the run start date.
- NDAYS must be ≥ 1 and ≤ 426.
- LVEH must have entries of 0, 1 or 2.

2.2 FILES

When running the SAGE option alone, the following five files must be present:

1. Swath Table - SWATH

   See Section 2.4 of the Users Manual.

2. Swath Reference - SWATHR

   See Section 2.4 of the Users Manual.

3. Segment ID File - SEGID

   See Section 2.4 of the Users Manual for a description.

4. Weather Data File - WEATHR

   Required as input if IOPT = 0 and produced as output if IOPT ≠ 0.

   See Section 2.4 of the Users Manual for a description.

5. NASA Weather Tape - WEATAP

   This tape is required input if IOPT = 0.

   See Section 2.4 of the Users Manual for a description.

3. Segment ID File

This file is generated from NASA data cards for use in the SAGE program. This file is only valid for DAPTS and IOC.

Access Method: Sequential with fixed length records

Status: Permanent during life of IOC

Sort: Country, then region, then zone, then strata, then substrata, then segment, 4800 records

Media: Disk

Record Formats:

Header Record:

Name (2) - 8 bytes, file name - 'SEGMENT'

ICASE - case no. 20 bytes filler

Detail Record:

Country ID - 4 bytes, 4 alpha characters

Region ID - 2 bytes, 3 digit no., set to 1

Zone ID - 2 bytes, 3 digit no., 1 to 400

Strata ID - 2 bytes, 4 digit no., 1 to 1600

Substrata ID - 2 bytes, 1 to 4800

Segment ID - 2 bytes, 4 digit no., 1 to 4800

Training Seg Ind - 2 bytes, 0 - normal, 1 - training

Latitude - 4 bytes, in radians, $\pm \pi/2$

Longitude - 4 bytes, in radians, 0 - 2π

Grid No. - 2 bytes, 1-16,000

Crop Proportion Table

Crop Cat % mixed crop - 4 bytes, %, 0 to 100

Crop Cat % winter wheat - 4 bytes, %, 0 to 100
Set to a value or 0 for DAPTS.

Crop Cat % spring wheat - 4 bytes, %, 0 to 100
Set to value or 0 for DAPTS
(opposite from Crop Cat %
winter wheat).

Last detail record has 'ZZZZ' in the first 4 bytes followed by 0's in the remainder of the record.

Total: 32 bytes, 13 words

Block Factor: 20

File Size: 153, 800 bytes, 62, 400 words

Usage: Used by SAGE to get lat/lon and crop data to write on segment reference data file.

4. Weather Data File

This file contains cloud cover % for 366 days and 16000 grid points. This file is generated from a NASA supplied weather tape.

Access Method: Direct with fixed length records

Status: Semi-permanent. Can be regenerated for each use of SAGE or kept as a permanent file for IOC studies.

Sort: Grid point, then day
To get to a particular record N which represents 5 grid points:

$$N = 1 + \frac{GRID}{5} + 1 \text{ (if remainder)} + 0 \text{ (if no remainder)}$$

Media: Disk

Record Formats:

Header Record:

Name (2) - 8 bytes, file name = 'WEATHER '

$\overline{N}$MAX - No. of days in a record, 2 bytes, 1 to 366

Total: 12 bytes or 4 words
171 byte filler

Detail Record - For 1 grid - all days
Each day entry is a value 0 to 8 respresenting 8th's of 100%. 8-4 bit entries are stored right justified in a word; for 366 days it would require $\frac{366*4}{8}$ = 183 bytes or 46 words.

Blocking Factor: 5

File Size: 2,928,000 bytes, 736,000 words

Usage: This file is used by SAGE to obtain the cloud cover data. For FOC on the PDP, the file will be stored in terms of 16 bit words. There will be three entries per PDP word. This will result in a 25% increase in space requirements.

When running the weather file generation utility alone, the NASA weather tape must be present for input.

NASA Weather Tape Format

There is one file on tape written in binary mode (odd parity) with sequential I/O routines. The tape is 7 track written at 800 BPI density. There will be 1600 physical records on tape and each record will be 500 words long. There will be 10 logical index point records per physical record. Each logical record will be 50 words long and will contain BCD information in the following format:

INDEX	⊢GRID⊣	LAT	N or S	LONG	E or W	YEARS	% FREQ	MEAN %
(I5, 2X,	I3, 2X, I3,	3X, 3I2,	A1,	IX, I3,	2I2, A1,	I2,	12 (I3, 8I2),	12I3)

If both the utility and SAGE are run, then the NASA weather tape would replace the weather data file as input. The weather data file would be computed prior to SAGE execution.

3.0 PROCESSING

3.1 OVERVIEW

The total program is divided into 2 phases. The phase which would be executed first (if selected) would be the weather data file creator. SAGE would be executed second if both phases are selected. The NASA weather tape must be supplied by NASA in the format shown in Section 2.2. In order to generate a weather data file, most of the input required by SAGE will come from the input data files. The segment location file will control SAGE since each segment identified on the file will be processed for acquisitions. In Section 3.2 a program block diagram illustrates the logic flow and the order in which the basic functions are performed. In Section 3.3 an ordered list of the procedures and equations is given.

3.2 PROGRAM FLOW

See Page for a block diagram giving an overview of program functions to be performed, and their order. Segment no. is the key control item for SAGE.

3.3 PROCEDURES AND EQUATIONS

3.3.1 Creation of the Weather Data File

1. Create Weather Data File

    1. A record for 1 grid point is read from the NASA weather tape to get grid no - GRID and f_j - IFREQ (frequency data set).

    2. The following computations are performed on f_j set for each month Jan through December.

        Form set $F_k = \sum_{j=0}^{k} f_j$ where $0 \leq k \leq 8$

        such that $F_0 = f_0$, $F_1 = f_0 + f_1$, etc.

    3. Test F_8 for 0; if it is then Step 4 is not done. Instead, a constant normal cumulative distribution probability table for the current month will be used for F_0-F_8. A message will print if F_8 is not within 99 to 101.

4. Form set $Y_K = \frac{F_k}{F_8}$. where $0 \leq k \leq 8$ and $K = k + 1$

   $Y_0 = 0.$

5. The following computations are performed for each day of each month.

   Form Ω - random number using RAND as the seed. Determine which pair of Y_K's $0 \leq K \leq 9$ Ω lies between. If Ω lies between Y_K and Y_{K+1} , then PC - cloud cover in eighths is set to K. In this manner the PC value for NMAX days of the year (assume 366 day max) is computed.

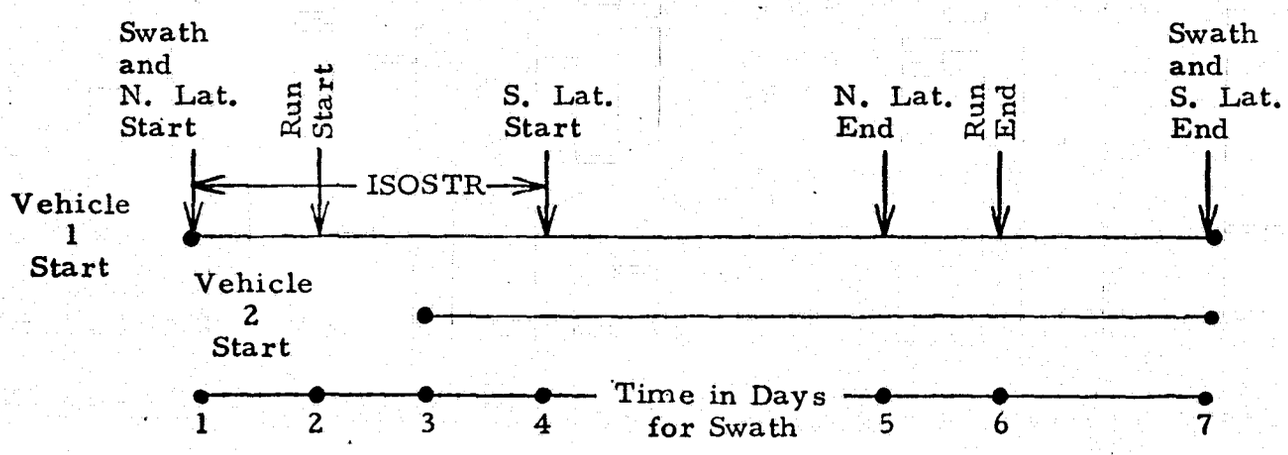
6. The PC table for IGRID is then written on the weather data file as one logical record.

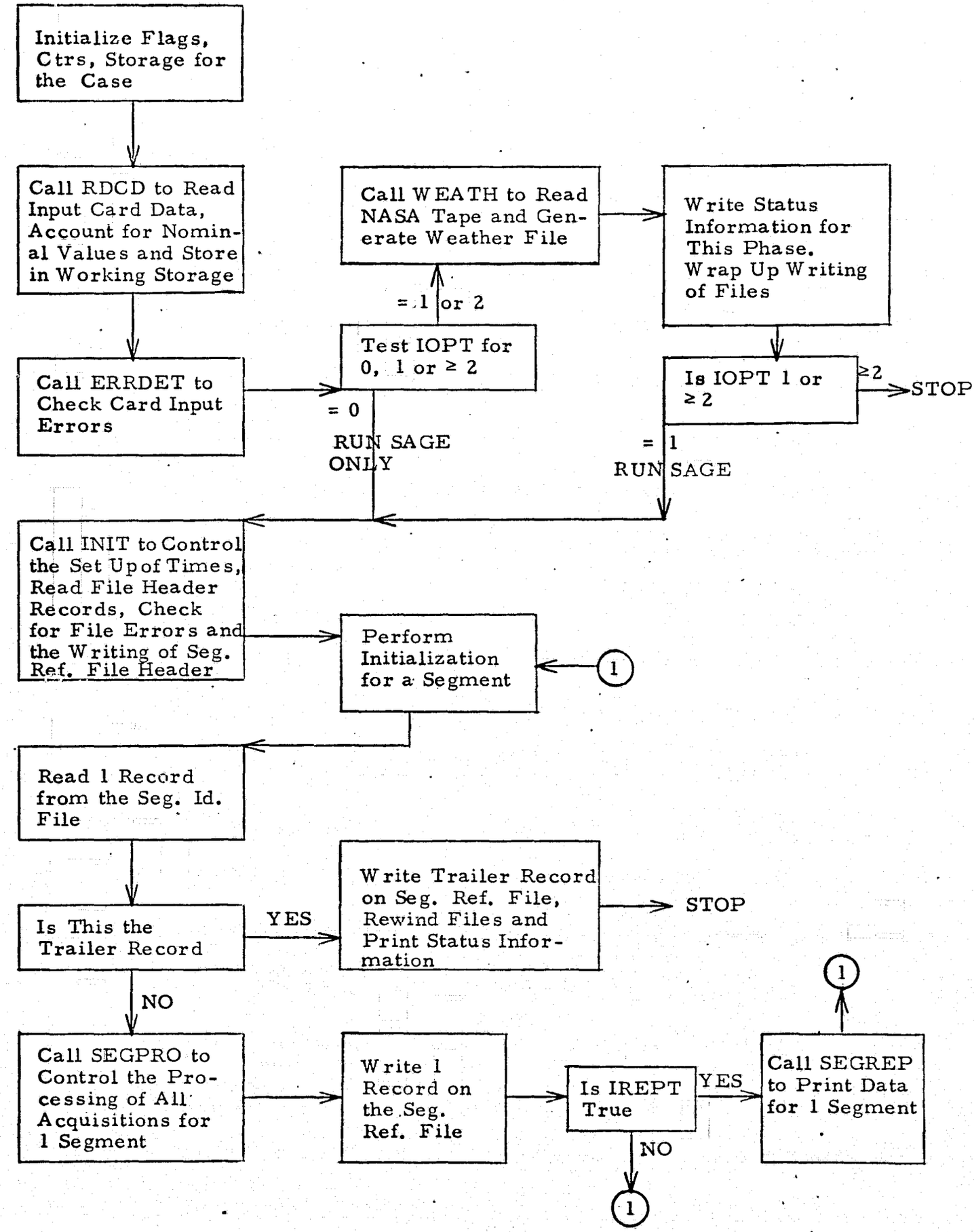
3.3.2 SAGE

1. Input Procedures

   - Read in card data and check fields for errors.
   - Read the header records of all input files, store required data in working storage and check for correct files.
   - Determine the number of vehicles and if only 1, which one.
   - Determine the key time points in the run as shown in the example below:

   Data Required: IV1TIM, IV2TIM, ISOSTR and NODAY from the swath table file.
   ISTIME, NDAYS from cards.




Initialize Flags, Ctrs, Storage for the Case
Call RDCD to Read Input Card Data, Account for Nominal Values and Store in Working Storage
Call ERRDET to Check Card Input Errors
Test IOPT for 0, 1 or ≥ 2
= 1 or 2
Call WEATH to Read NASA Tape and Generate Weather File
Write Status Information for This Phase. Wrap Up Writing of Files
Is IOPT 1 or ≥ 2
≥2
STOP
= 0
RUN SAGE ONLY
= 1
RUN SAGE
Call INIT to Control the Set Up of Times, Read File Header Records, Check for File Errors and the Writing of Seg. Ref. File Header
Perform Initialization for a Segment
1
Read 1 Record from the Seg. Id. File
Is This the Trailer Record
YES
Write Trailer Record on Seg. Ref. File, Rewind Files and Print Status Information
STOP
NO
Call SEGPRO to Control the Processing of All Acquisitions for 1 Segment
Write 1 Record on the Seg. Ref. File
Is IREPT True
YES
Call SEGREP to Print Data for 1 Segment
1
NO
1

Explanation of diagram above:

- Time 1 and time 7 represent the duration of the swath table and reference files.
- Time 2 is ISTIME and represents the desired run start date for access.
- Time 1 and time 3 are IV1TIM and IV2TIM.
- ISOSTR represents the delay in the utilization of the southern latitude band data which starts at time 4 = time 1 + ISOSTR.
- Time 5 represents the last day of northern latitude band day of northern latitude band day utilization and is computed by time 5 = time 7 - ISOSTR.
- Time 6 is = to ISTIME + NDAYS - 1.
- Time 7 is = to the smallest of IV1TIM and IV2TIM + (NODAY - 1).
- In this example, accesses for the northern latitudes and vehicle 1 would start at time 2. Accesses for northern latitudes and both vehicles would start at time 3. Both southern and northern latitudes would be processed starting at time 4. The processing for northern latitudes would stop at time 5 and the run would end at time 6.

2. Processing for 1 Segment

The following procedure is performed for all segments as read from the segment location file:

1) Determine if current segments LAT-LON will be accessed by 1 or 2 vehicles and whether it is in an active hemisphere. If not in an active segment, the remainder of the steps are skipped and the next segment is initiated. Determine day search limits on swath files.

2) If the THETSG is in the band about the equator not in LATNO table, an information message is printed and the next segment is initiated. Given the grid point for the current segment lat/lon, the cloud cover data is read from the weather file for 1 year of daily data, this is stored into ICLDCV array.

3) Establish Potential Access

The following steps are done for each day within the specified time limits and each vehicle:

a. Compute $\Delta\Omega_{MIN}$ and $\Delta\Omega_{MAX}$

Determine latitude j and j + 1 from LATNO table such that $LATNO(j) \geq \theta_T \geq LATNO(j+1)$

$\Delta\Omega_{MIN} = MIN\ [\Delta\Omega_1(j),\ \Delta\Omega_1(j+1)]$

$\Delta\Omega_{MAX} = MAX\ [\Delta\Omega_2(j),\ \Delta\Omega_2(j+1)]$

LAT = j; LAT1 = j+1

b. Find a day no. - JDAY, rev no. - IREV, vehicle no. - IVEH on the segment reference file such that

$$\Delta\Omega_{MIN} \leq (\phi_{NODE} - \phi_{SEG})_{MOD\ 2\pi} \leq \Delta\Omega_{MAX}$$

If $(\phi_{NODE} - \phi_{SEG})$ is not within these bounds, then no access is possible. The processing for the next day is initiated. Otherwise, step 4 is next.

4) Access Verification

a. The record on the swath table for day - JDAY and vehicle - IVEH is read to get TIME, IALT and DLONG for latitudes LAT and LAT1. IDAY and IZDAY are computed from JDAY.

b. Letting j = LAT and j+1 = LAT1, the following equation set is computed:

DELTA = THETSG - LATNO(j)

Compute following equations for i = 1, 3:

XTIME(i) = TIME(i, 1) + DELTA * [TIME(i, 2) - TIME(i, 1)]

XDLONG(i) = DLONG(i, 1) + DELTA * [DLONG(i, 2) - DLONG(i, 1)]

$$XALT = ALT_1 + \frac{ALT_2 - ALT_1}{TIME(2,2) - TIME(2,1)} * (XTIME(2) - TIME(2,1))$$

$$SIGMA = ARCSIN\left[\left(\frac{RADIUS + XALT}{RADIUS}\right) * SIN(SA(3))\right] - SA(3)$$

$$DDLNGL = \frac{-DECR}{RADIUS * SIGMA} * (XDLONG(3) - XDLONG(2))$$

$$DDLNGR = \frac{-DECR}{RADIUS * SIGMA} * (XDLONG(2) - XDLONG(1))$$

$$CHECKL = XDLONG(3) + DDLNGL$$

$$CHECKR = XDLONG(1) - DDLNGR$$

$$CHLONG = (PHISEG - PHINDE) \text{ MOD } 360\ (2\pi)$$

Check for valid access:

If CHLONG < than CHECKL, then no access is possible and step 3 is retried for the next day.

If CHLON > than CHECKR, then no access is possible and step 3 is retried for the next day.

If neither of these conditions exist, then there is an access and step 5 is performed.

5) Compute imaging conditions:

If (CHLONG - XDLONG(2)) is -

< 0 access is on left side, execute a.

= 0 access on ground track, execute b.

> 0 access is on right side, execute c.

a. $$TS = XTIME(2) + \left[\frac{CHLONG - XDLONG(2)}{XDLONG(3) - XDLONG(2)}\right] * [XTIME(3) - XTIME(2)] + TMNODE$$

$$CTA = +TAN^{-1}\left[\left(\frac{CHLONG - XDLONG(2)}{XDLONG(3) - XDLONG(2)}\right) * TAN(SA(3))\right]$$

$$YALT = ALT(1) + \left[\frac{TS - TIME(2,1)}{TIME(2,2) - TIME(2,1)}\right] * [ALT(2) - ALT(1)]$$

b. $$TS = XTIME(2) + TMNODE$$

$$CTA = 0$$

$$YALT = XALT$$

c. $$TS = XTIME(2) + \left[\frac{CHLONG - XDLONG(2)}{XDLONG(1) - XDLONG(2)}\right] * [XTIME(1) - XTIME(2)] + TMNODE$$

$$CTA = -TAN^{-1}\left[\left(\frac{CHLONG - XDLONG(2)}{XDLONG(1) - XDLONG(2)}\right) TAN(SA(3))\right]$$

$$YALT = ALT(1) + \left[\frac{TS - TIME(2,1)}{TIME(2,2) - TIME(2,1)}\right] * [ALT(2) - ALT(1)]$$

6) Compute sun angle - EL

This calculation is performed by using an existing subroutine ALPHA.

Store in ICLOD the % value from the cloud cover day table for IDAY.

The following data is stored by acquisition day for each acquisition from the computed quantities above. I represents the count on number of acquisitions, $1 \leq I \leq 150$.

IZDAYT(I) = IZDAY, ICLODT(I) = ICLOD

IREVT(I) = IREV, TST(I) = TS

IVEHT(I) = IVEH, YALTT(I) = YALT

CTAT(I) = CTA

ELT(I) = EL

7) The data from segment location file is then combined with the data from step 6 and a record on the segment reference data file is written.

3. If IREPT is true, the segment acquisition data report is generated.

4. Steps 2 and 3 are repeated for each segment on the segment location file.

SAGE Symbol Table

Name	Symbol	Description	Source	Used	Range	Units
DLOMMN (100)	$\Delta\Omega_1$	The minimum Δ longitude at a given latitude from the ascending node.	Sw. Tab.	3		Radians
DLONMX (100)	$\Delta\Omega_2$	The maximum Δ longitude at a given latitude from the ascending node.	Sw. Tab.	3		Radians
ICLDCV (366)	--	% * 10 of cloud cover for each day of a year.	Weath. File	2	0-1000	Radians
PHINOD (17)	ϕ_{NODE}	Longitude of ascending node for all revs for a day.	Sw. Ref.	3		Radians
PHISEG	ϕ_{SEG}	Longitude of segment.	Seg. Loc.	1, 2, 3, 4, 7	$0-2\pi$	Radians
THETSG	θ_T	Latitude of segment.	Seg. Loc.	1, 2, 3, 4, 7	$0-\pi/2$	Radians
LATNO (100)	--	Table of latitude no.'s to be studied.	Sw. Tab.	3	$0-\pm 65$	Deg
IZDAY	--	Zulu day no. for which there is an acquisition	4	Seg. Ref. File		Day
IREV	--	Day rev no. of acquisition.	3	7		--
IVEH	--	Veh no. associated with acquisition day.	3	7		--
IGRID	--	Grid point no. for current segment latitude-longitude	Seg. Loc. File	2		--
LAT	--	Latitude ptr. into LATNO which gives upper bound latitude on θ_T.	3	4		--
LAT1	--	Latitude ptr. into LATNO which gives lower bound latitude on θ_T.	3	4		--
TMNODE	t_{NODE}	Time ascending node crosses equator for ZDAY, IREV and IVEH.	Sw. Ref.	5	0	Sec
		*New				

SAGE Symbol Table

Name	Symbol	Description	Source	Used	Range	Units
DLNMIN	$\Delta\Omega_{MIN}$	The conservative minimum Δ longitude which bounds the segment.	3	3	0-2π	Radians
DLNMAX	$\Delta\Omega_{MAX}$	The conservative maximum Δ longitude which bounds the segment.	3	3	0-2π	Radians
DLONG (3, 2)	$\Delta\phi$	Vehicle and swath Δ longitudes for LAT and LAT1.	Sw. Tab.	3, 4	0-2π	Radians
TIME (3, 2)	--	Times associated with DLONG	Sw. Tab.	3, 4, 5	1-84,600	Sec
ALT (2)	--	Vehicle altitude at LAT and LAT1	Sw. Tab., 3	4, 5		
PHINDE	ϕ_{NODE}	Longitude of ascending node for IREV, IDAY and IVEH.	3	4		
SA (3)	--	Reference scan angle.	Sw. Ref.	4, 5		
DELTA	--	Latitude interpolation difference.	4	4		Radians
XTIME (3)	--	Vehicle time at latitude of interest for i^{th} swath position: left, center right.	4	4, 5		Sec
XDLONG (3)	--	Delta longitude at latitude of interest for i^{th} swath position: i = left, center, right.	4	5, 4		Radians
XALT	--	Vehicle altitude at latitude of interest.	4	4		Kilom.
SIGMA	--	Half earth centered angle for swath.	4	4		Radians
DDLNGL	--	Longitude decrement for left side.	4	4		Radians
DDLNGR	--	Longitude decrement for right side.	4	4		Radians
CHECKL	--	Relative longitude for left side of swath.	4	4		Radians
CHECKR	--	Relative longitude for right side of swath.	4	4		Radians

*New

SAGE Symbol Table

Name	Symbol	Description	Source	Used	Range	Units
DECR	--	Swath decrement input from cards.	Cd. In.	4	1 to 100	Kilom.
TS	T_S	Image time.	5	5, 7		Sec
CTA	--	Crosstrack angle.	5	5, 7		Radians
YALT	--	Altitude of vehicle at time of acquisition.	5	5, 7		Kilom.
RADIUS	--	Radius of earth - 6376.436	Const.	4		
EL	--	Sun elevation angle.	6	7		Deg
CIS	--	Obliquity of the ecliptic - .409280.	Const.	6		
CNUV	--	Constant-true anomaly of the earth at vernal equinox - 1.35695.	Const.	6		
ECCE	--	Constant-eccentricity of the earth's orbit - .167263E-1.	Const.	6		
JDAY	--	Day no. from 1 to 549 to compute swath table record no.	Sw. Ref.	4		
IDAY	--	Day no. of the current year (current date).	4	6		
PI	π	3.1415926	Const.	6		
PI2	$\pi/2$	1.5707963	Const.	6		
PI32	$3/2\pi$	4.712389	Const.	6		
RADIAN	--	57.29578	Const.	6		
TWOPI	2π	6.2831852	Const.	6		
CHLONG	--	Relative longitude difference of segment of interest.	4	4, 5		Radians
ICLOD	--	Cloud cover % * 10 for day of acquisition.	6	7		

4.0 OUTPUT

Where appropriate, the output discussion will refer to SAGE and to the weather data file utility. If no mention is made of the utility or SAGE, then it can be assumed the information refers to SAGE only.

4.1 PRINTED DATA

4.1.1 Reports

There is only one report and it is printed by segment no. The report lists the pertinent data for each time a given segment is acquired by one of the vehicles. The format of the report is as follows:

Segment Access Data Report Country AAAA

Segment nnnn

ACCESS COUNT	ACQUISI MO/DY/YR	REV	VEH	CTA	EL	CLOUD COVER	ACQUISI TIME	ALT.
1	nn/nn/nn	nn	n	nnn nn nn	nnn nn nn	nnn.n	nnnnn	nnnnn
2	↓	↓	↓	↓	↓	↓	↓	↓
3	↓	↓	↓	↓	↓	↓	↓	↓
↓	↓	↓	↓	↓	↓	↓	↓	↓
150	nn/nn/nn	nn	n	nnn nn nn	nnn nn nn	nnn.n	nnnnn	nnnnn

There is a maximum of 150 acquisitions possible requiring 4 pages of print, but in general, the total acquisitions will fit on 1 page.

4.1.2 Intermediate Debug

There is no debug output in this program.

4.1.3 Status Information

Weather Data File Utility

The following information is printed upon successful completion of the utility:

***WEATHER DATA FILE UTILITY COMPLETED

WEATHER DATA FILE:

NO OF DAYS IN A RECORD: NNN

NO OF RECORDS READ: NNNN

LAST RANDOM NO SEED: ± NNNNNNNNNN

SAGE Program

The following information prints out upon successful completion of the SAGE program:

***SAGE HAS SUCCESSFULLY WRITTEN THE SEGMENT REFERENCE DATA FILE

SEGMENT REFERENCE DATA FILE:

REFERENCE DATE: NN/NN/NN

SWATH FILES CASE USED: NNNN

NO OF RECORDS WRITTEN: NNNN

SEGMENT ID FILE:

REFERENCE DATE: NN/NN/NN

SEGMENT ID FILE CASE USED: NNNN

NO OF RECORDS READ: NNNN

4.1.4 Echo Print of Input Control Cards

The print format is as follows:

HEADER

AAA ——————————→ AAA

ICASE	IOPT	NDAYS	ICSESW	ICSESG	ISTIME	LVEH	IREPT	DECR
NNNN	N	NNN	NNNN	NNNN	NNNNNN	NN	L	N.NN

RAND	IGRDN	ISWRTO	NLATSW
.NNNNNNNNNN±NNN	NNNNN	NNNN	NNN

4.2 FILES

There are two possible output files. The weather data file utility produces the weather data file - WEATHR. This file is defined already in Section 2.2. SAGE generates the segment reference file - SEGREF. See Section 2.4 of the Users Manual for a description.

SEGMENT REFERENCE FILE

Access Method: Sequential with fixed length records

Status: Semi-permanent. Regenerated normally less than once per week. Will normally be kept on tape.

Sort: Country, then region, then zone, then strata, then substrata, and then segment. There are 4800 segments for DAPTS and IOC; 10000 for FOC.

Media: Disk or tape, tape for IOC

Record Formats:

Header Record:

Name (2) - 8 bytes, file name: 'SEG REFE1'

ICASE - 2 bytes, case no. - 4 digits

Reference Date - No. of days since 1900 (2 bytes)
Used in conjunction with acquisition date.

NMAX - 2 bytes, no. of days in study; range 1 to 426

No. of Crops - 3 for IOC, max of 26 for FOC

ICSESW - 2 bytes, swath files case no. - 4 digits

ICSESG - 2 bytes, segment ID file case no. - 4 digits

Total: 20 bytes or 8 words + 2360 bytes filler for IOC

Detail Record:

Country ID - 4 bytes, 4 alpha characters

Region ID - 2 bytes, 3 digit no., 1 to 100

Zone ID - 2 bytes, 3 digit no., 1 to 400

Strata ID - 2 bytes, 4 digit no., 1 to 1600

Substrata ID - 2 bytes, 4 digit no., 1 to 4800

Segment ID - 2 bytes, 5 digit no., 1 to 10000

Training Seg. Ind. - 2 bytes, 0 - normal, 1 - training

Training Seg. Ptr. - 2 bytes, segment no. of associated training segment (FOC)

Latitude - 4 bytes, in radians, $\pm\pi$

Longitude - 4 bytes, in radians, $0\text{-}2\pi$

Atmospheric Type - 2 bytes, 2 alphanumeric characters (FOC)

Group No. - 2 bytes (FOC)

Field Size - 2 bytes, kilometers^2 * 10, range 10-20000 (FOC)

Cropping Practice Flag - 2 bytes (FOC)

Crop Proportion Table - IOC (FOC has 26 entries)

- Crop Cat % mixed crop - 4 bytes, %, 0 to 100
- Crop Cat % winter wheat - 4 bytes, %, 0 to 100
- Crop Cat % spring wheat - 4 bytes, %, 0 to 100

Acquisition Data - 150 entries each

- Acquisition Day - 2 bytes, Zulu date (no. of days since 1900)
- Revolution No. - 2 bytes, range 1 to 17
- Vehicle No. - 2 bytes, range 1 to 2
- Crosstrack Angle - 4 bytes, Radians, range $\pm 10^{o}$
- Sun Elevation Angle - 4 bytes, Radians, range $\pm 90^{o}$
- Cloud Cover % - 2 bytes, % * 10
- Time of Acquisition - 4 bytes, sec, 1 to 86,400
- Vehicle Altitude - 2 bytes, kilometers * 10, 10 to 10000
- Station Contact Ind. - 2 bytes (FOC)
- Atmosphere Attenuation Factor - 2 bytes (FOC)
- Snow Cover % - 2 bytes (FOC)

Last detail record has 'ZZZZ' in the first 4 bytes followed by 0's in the remainder of the record.

Total Length: IOC - 2,740 bytes, 1,217 words
FOC - 3,686 bytes, 1,690 words

Block Factor: 1

File Size: IOC - 2,425,212 words
FOC - 14,700,008 words

Usage: This file will be used as follows:

1. DAPTS

   Used by SACS in conjunction with the crop window file to determine when accesses occur and conditions of access. This data is then represented on the segment acquisition file.

2. IOC

   Same usage as above. In addition, it will be used in conjunction with the data acquisition file by the CAMS module.

3. FOC

   Same usage as IOC except crop calendar information is obtained from substrata properties truth file.

5.0 ERROR PROCESSING

5.1 GENERAL

The program will attempt to find as many errors during the input card processing as possible. The program will continue checking for further input errors upon detecting any input error. There are 2 levels of error. These are:

Level 1 - continue processing

Level 2 - job fatal

When a level 1 error occurs, the program will print an informative message and continue. When a level 2 error occurs, the program will print an informative message and return control back to the computer system.

5.2 INPUT DATA ERRORS

.Level 2

1. A check is made to see if ICSESW matches the case no. on the swath files. Message:

   ***ICSESW DOES NOT MATCH THE CASE NO ON THE SWATH INPUT FILES OR SWATH FILES NOT MOUNTED.

2. A check is made to see if ICSESG matches the case no. on the segment ID file. Message:

   ***ICSESG DOES NOT MATCH THE CASE NO. ON THE SEGMENT ID FILE OR THIS FILE HAS NOT BEEN MOUNTED.

3. A check is made to make sure that ISTIME is not less than the earliest vehicle start time as specified on the swath table file. Message:

   ***ISTIME IS LESS THAN THE EARLIEST VEHICLE START TIME IV1TIM OR IV2TIM.

4. A check is made to make sure NDAYS is between 1 and 426. Message:

   ***NDAYS IS NOT BETWEEN 1 AND 426.

5. A check is made to make sure that the vehicle no. list in IVEH is between 0 and 2 and that there is at least one NONZERO entry. Message:

   *** IVEH HAS AN ENTRY NOT BETWEEN 0 AND 2 OR DOES NOT HAVE AT LEAST 1 NONZERO ENTRY.

6. A check is made to make sure that a weather data file has been mounted. Message

   ***THE WEATHER DATA FILE HAS NOT BEEN MOUNTED.

7. A check is made to make sure that the input quantity IVEH does not specify vehicles not on the swath files. Message.

   *** IVEH LIST IS NOT COMPATIBLE WITH NVEH AS SPECIFIED ON THE SWATH FILES.

8. A check is made to make sure IGRDN is between 1 and 16000

   ***IGRDN DOES NOT HAVE A VALUE BETWEEN 1 AND 16000

5.3 PROCESSING ERRORS

Level 1

20. A check is made while forming the acquisition list for a segment that no more than 150 acquisitions occur. Message:

    ***IN PROCESSING SEGMENT NNNN MORE THAN 150 ACQUISITIONS HAVE OCCURRED. NO MORE ACQUISITIONS WILL BE PROCESSED.

21. A check will be made to make sure at least 1 acquisition of a segment occurs. Message:

    ***IN PROCESSING SEGMENT NNNN NO VALID ACQUISITION OCCURRED.

22. A check is made to see if a segment's latitude is in the band about the equator which is not accounted for in LATNO table. Message:

    ***SEGMENT NNNN LATITUDE NN NN NN IS NOT IN THE LATITUDE BAND SPECIFIED BY LATNO TABLE.

23. A check is made in the weather file utility to make sure $99 \leq F_8 \leq 101$. Message.

    ***THE CUMULATIVE FREQUENCY DISTRIBUTION -F SUB 8 IS NOT BETWEEN 99 AND 101.

5.4 INPUT/OUTPUT ERRORS

For sequential I/O the FORTRAN system on the UNIVAC or PDP takes control and prints a message identifying the problem and will either continue processing or abandon the job. If processing continues, the system counts the number of times this error re-occurs and if it happens a certain number of times, the system will abandon the job.

For direct access I/O, the UNIVAC or PDP D.A. I/O package prints out an informative message, sets an error flag and allows processing to continue. In this program the swath table file is the only direct access file. Immediately after the informative message, the following message will print:

40. ***AN IRRECOVERABLE I/O ERROR HAS OCCURRED IN READING/WRITING A RECORD ON THE _____ FILE. THE JOB IS BEING ABANDONED.

The blanks will be filled in with 'SWATH TB', or 'WEATHER' depending on the direct access file involved.

PART II

COMMON BLOCK DEFINITIONS

COMMON STORAGE ALLOCATION

Name FLAG Size 56 Page 1 of

Function

Name	Dimension	Format	Description	Symbol	Units
HEADER	18	R	Contains 72 CHAR HEADER		
ICASE	1	I	Case no of access run		
JCSESW	1	I	Swath files case no.		
ICSESG	1	I	Segment ID file case no.		
ISTIME	3	I	Run start date		
ISDAY	1	I	Start day≡ISTIME only in Zulu time		
NDAYS	1	I	No. of days in the run set to 426		
IOPT	1	I	Program run option -0- run SAGE, 1-Run weather, run both		
LVEH	2	I	List of vehicle no.'s to process set to 1, 2		
IREPT	1	L	Report print indicator F-No Print, T-Print		
NVEH	1	I	No. of vehicles on swath files		
NODAY	1	I	No. of days in swath files		
IVITIM	(3)	I	Vehicle 1 swath start date		
IGRID	1	I	Current weather grid no.		
IV2TIM	(3)	I	Vehicle 2 swath calen, start date		
LYR	1	I	Current 2 digit years		
LMO	1	I	Current month no.		

COMMON STORAGE ALLOCATION

Name FLAG Size Page 2 of

Function

Name	Dimension	Format	Description	Symbol	Units
LDA	1	I	Current day of month no.		
ISWDST	1	I	Zulu start day of swath table ≡ earliest swath day		
JDAY	1	I	Swath table file day no used to get record no.		
IDAY	1	I	Day no. of current year		
IPDAY	1	I	Previous day no. of current year		
IYEAR	1	I	Computed from LYR modulo4 = 0-leap year, ≠ 0 - not leap yr.		
IYEARB	1	I	Previous year including 1950		
ICDAY	1	I	Day no. of run relative to 1950- current Zulu		
ISWVST	2	I	Start day of vehicle 1 swath - Zulu - subscript 1		
			Start day of vehicle 2 swath - Zulu - subscript 2		
ISLATS	1	I	Start day of southern latitude - Zulu		
INLATE	1	I	End day of northern latitude - Zulu		
IRUNE	1	I	Run end day - Zulu must be ≤ ISWATE		
ISWATE	1	I	Swath end day - Zulu		
NLAT	1	I	No. of latitudes in LATNO band and in swath file rec.		
NMAX	1	I	No. of days in a weather data record		
ISOSTR	1	I	Delay for southern hemisphere start relative to north lat months		

COMMON STORAGE ALLOCATION

Name WORK Size 414 Page 1 of

Function Contains Data Pertinent to the Acquisition Calculations

Name	Dimension	Format	Description	Symbol	Units
ICLDCV	366	I	%*10 Cloud cover for each day of year		
IZDAY	1	I	Zulu day no. for which there is an acquisition		
IREV	1	I	Rev no. on acquisition day IZDAY		
IVEH	1	I	Veh no. on acquisition day IZDAY		
LAT	1	I	Latitude pointer into LATNO which gives upper bound latitude on θ_T		
LAT1	1	I	Latitude PTR into LATNO which gives lower lower bound lat on θ_T		
TMNODE	1	R	Time ascending node cross's equater for IDAY, IREV and IVEH	t_{node}	
DLNMIN	1	R	The conservative minimum Δ longitude which bounds the seg.	$\Delta\Omega_{min}$	
DLNMAX	1	R	The conservative maximum Δ longitude which bounds the seg.	$\Delta\Omega_{max}$	
DLONG	3, 2	R	Vehicle and swath Δ longitudes for lat and lat1	$\Delta\phi$	
TIME	3, 2	R	Times associated with DLONG		
ALT	2	R	Vehicle altitude at lat and lat1		KILO.
PHINDE	1	R	Longitude of ascending node for IREV, IDAY and IVEH	ϕ Node	RAD.
SA	3	R	Reference scan angle		RAD.
DELTA	1	R	See problem definition		
XTIME	3	R	See problem definition		

COMMON STORAGE ALLOCATION

Name WORK Size Page 2 of

Function

Name	Dimension	Format	Description	Symbol	Units
XDLONG	3	R	See Problem Definition		
XALT	1	R	See Problem Definition		
SIGMA	1	R	See Problem Definition		
DDLNGL	1	R	See Problem Definition		
DDLNGR	1	R	See Problem Definition		
CHECKL	1	R	See Problem Definition		
CHECKR	1	R	See Problem Definition		
DECR	1	R	See Problem Definition		
RAND	1	D.P	Random Number generator input seed (set to 1.0)		
TS	1	R	See Problem Definition	T_s	
CTA	1	R	See Problem Definition		
YALT	1	R	See Problem Definition		
ICLOD	1	I	Cloud Cover % See Problem Definition		
EL	1	R	Sun elevation angle		
CHLONG	1	R	See Problem Definition		RAD
IACCT	1	I	Count of no. of acquisitions for a Segment		

COMMON STORAGE ALLOCATION

Name CONS Size 28 Page 1 of

Function Contains constants, file unit numbers and page control data for printing

Name	Dimension	Format	Description	Symbol	Units
INP	1	I	Card reader = 5		
KO	1	I	Printer = 6		
NLINE	1	I	Current line count = 40 initially		
LINMAX	1	I	Max lines on a page = 39		
NPAGE	1	I	Page count = 0 - initially		
ISEGF	1	I	Segment file = 1 ID		
ISWR	1	I	Swath reference file = 9		
ISWTB	1	I	Swath Table file = 8		
IREFF	1	I	Seg. reference file = 10		
IWTHF	1	I	Weather data file = 11		
IWTAPE	1	I	NASA weather tape = 33		
RADIUS	1	R	6376.436 radius of the earth		KILOM
CIS	1	R	.409280 obliquity of the ecliptic		
CNUV	1	R	1.35695 - See Problem Definition		
ECCE	1	R	.167263 E-1 See Problem Definition		
PI	1	R	3.1415926		
PI2	1	R	1.5707963		
PI32	1	R	4.712389		

COMMON STORAGE ALLOCATION

Name CONS Size Page 2 of

Function

Name	Dimension	Format	Description	Symbol	Units
RADIAN	1	R	57.29578		
TWOPI	1	R	6.2831852		
NAME	2	R	'SEGb REFER'		
NOCROP	1	I	≡ 3 No. of crops IOC		
IBLKF	1	I	Blocking factor for weather tape = 5		
IPCK	1	I	No. of weather items packed in a word = 8		
NPREC	1	I	= 230 - No. of words in physical rec. of IWTHF		
NLREC	1	I	= 46 - No. of words in logical rec. of IWTHF		
ISWRTO	1	I	Total no. of records on swath table file - nominal 1101		
IVCT	1	I	Count on no. of vehicles		
NLATSW	1	I	No. of latitudes in the (input data) swath table - nominal 100		
IGRDN	1	I	No. of grid points to read and write on weather file		
			nominal = 16000- can be changed for debug		

COMMON STORAGE ALLOCATION

Name DIRAC Size 5378 Page ____ of ____

Function Direct access file storage

Name	Dimension	Format	Description	Symbol	Units
IDEXSW	1104	I	Swath table file index CDC only		
IA	800	-	Swath buffer storage		
IDEXW	3201	I	Weather file index CDC only		
IB	230	-	Weather file storage		
IPC	46	I	Contains the weather % cloud cover in eights - 4 bits pack		
			8 per word right justified		

COMMON STORAGE ALLOCATION

Name WEATDT Size 477 Page ____ of ____

Function Contains weathe r data from weather tape

Name	Dimen-sion	For-mat	Description	Sym-bol	Units
IA	300	I	Contains 10 logical recrods of data direct for weather tape, UNIVAC format		
IB	50	I	Contains 1 logical record converted to BCD code right justified with		
			24 leading 0 bits, 6 characters		
IGRDPT	1	I	Grid point no.		
IPERWT	19, 12	I	% frequency table for 9 eights entries 0-8 for 12 months		
IC	30	I	Contains 1 logical record ≡ IB only packed 10 char per word for decode		

COMMON STORAGE ALLOCATION

Name ACQUIS Size 1200 Page ____ of ____

Function Contains acquisition and weather data

Data for writing on seg. ref. file

Name	Dimension	Format	Description	Symbol	Units
IZDAYT	150	I	See IZDAY def. in work com.		
IREVT	150	I	See IREV def. in work com.		
CTAT	150	R	See CTA def. in work com.		
ELT	150	R	See ELT def. in work com.		
ICLODT	150	I	See ICLOD def. in work com.		
TST	150	R	See TS def. in work com.		
YALTT	150	R	See YALT def. in work com.		
IVEHT	150	I	See IVEH def. in work com.		

COMMON STORAGE ALLOCATION

Name SEGFIL Size 13 Page ____ of ____

Function Contains data for one segment record segment ID file

Name	Dimension	Format	Description	Symbol	Units
COUNTR	2	R	Alphs country ID		
IREG	1	I	Region no. 1 to 100		
IZONE	1	I	Zone no. 1 to 400		
ISTRAT	1	I	Strata no. 1 to 1600		
ISUBST	1	I	Substrata 1 to 4800		
SEGMNT	1	I	Segment ID 1 to 4800		
THETSG	1	R	Segment latitude $\pm\pi/2$	θ SEG	RAD
PHISEG	1	R	Segment longitude 0 - 2π	ϕ SEG	RAD
ICROP	3	I	Crop proportion table		
ITRSEG	1	I	Training segment indicator		

COMMON STORAGE ALLOCATION

Name SWATH Size 1041 Page ____ of ____

Function Contains swath data working area

Name	Dimension	Format	Description	Symbol	Units
LATNO	100	I	List of latitudes from north to south ± 65		DEG
IALT	100	I	Vehicle altitudes vs LATNO KILOM * 10		
TIME1	100, 3	R	Vehicle and swath edge times 0 -86, 400 vs LATNO		SEC
DLONG1	100, 3	R	Vehicle and swath edge Δ longitude 0 -2π vs LATNO	$\Delta\phi$	RAD
DLONMN	100	R	Minimum Δ longitude at a given latitude vs LATNO, 0 - 2π	$\Delta\Omega_1$	RAD
DLONMX	100	R	Maximum Δ longitude at a given latitude vs LATNO, 0 - 2π	$\Delta\Omega_2$	RAD
KVEH	1	I	Current vehicle no. from swath ref 1 - 2		
KDAY	1	I	Current day no. 1-549 from swath ref.		
NREV	1	I	No. of revolutions for current day and vehicle - from swath ref.		
TIMNOD	17	R	Time ascending node cross's equator for each rev swath ref.	t_{node}	SEC
PHINOD	17	R	Longitude of ascending node for each rev swath ref	ϕ_{node}	RAD
INLAT	2	I	North latitude band (1) -low, (2) high		DEG
ISLAT	2	I	South latitude band Positive entries (1) - high (2) - low		DEG

PART III

LIST OF SUBROUTINES
AND
SUBROUTINE CALL STRUCTURE

LIST OF ROUTINES IN SAGE

NAME	FUNCTIONS
1. MAIN	Main control
2. RDCD	Controls reading of data cards, checking and storing
3. WEATH	Reads NASA weather tape and writes weather data file
4. INIT	Performs initialization of data and reads header records from input files, and writes header on seg. ref. file
5. READSW	Reads a record on swath file
6. WEATF	Reads a record from the weather file
7. LFPA	Computes days (Zulu Time) "common change"
8. DAY	Computes IDAY-DAY no. of year
9. FZULU	Given Zulu day, compute month, day and year
10. SOL	Compute EL sun elevation angle
11. POTEN	Determines potential access for a day
12. ACCVER	Verify access for a day
13. IMAGE	Compute image conditions and sun elevation angle
14. SEGPRO	Controls the processing for acquisitions for all days for one segment
15. SEGREP	Prints the output report for one segment
16. WRITWT	Write records on the weather file
17. RDMIA	Random number generator
18. UNPKPK	Data from or to IPC-file storage form

LIST OF ROUTINES IN SAGE
(CONTINUED)

NAME	FUNCTIONS
19. EJECT	Restore page, print header
20. PAGER	Keep line count and restore page automatically
21. PIMOD	Convert an angle to be between 0-2π
22. DEGMOD	Convert an angle from radians to deg., min., sec.
23. RDWETR	Read NASA weather tape data into storage (need a routine for UNIVAC and CDC)

SUBROUTINE CALL STRUCTURE

```
SAGE (MAIN)
    RDCD
        PAGER
        EJECT
    INIT
        WEATF
            PAGER
        READSW
            PAGER
        PAGER
        LFPA
    SEGPRO
        WEATF
        UNPKPK
        READSW
        DAY
        POTEN
            PIMOD
        ACCVER
            READSW
            PIMOD
        IMAGE
            ALPHA
        PAGER
        DEGMOD
    SEGREP
        EJECT
        PAGER
        FZULU
        DEGMOD
    WEATF
    READSW
    WEATH
        WRITWT
        RDWETR
            PAGER
        RDMIA
        UNPKPK
```

PART IV

SUBROUTINE DESCRIPTIONS AND FLOWCHARTS

SUPPLIED UTILITY ROUTINES

Routine Day

Call Day (IYMD, IDAY)

Given IYMD (3) where

IYMD (1)	IS Day No.
IYMD (2)	IS Month No.
IYMD (3)	IS Year No.

Compute year day no. in IDAY

Routine PIMOD

Call PIMOD (A)

Convert ±A in radians to an angle 0-2π

Routine SOL (Entry ALPHA)

Call ALPHA (IFLAG)

For emphemeris usage as called by hector

computes ALPHAM and ALPHAT and IFLAG = 1

Routine PAGER (Entry Eject)

Call PAGER (NLINES)

Updates line count in NLINE with NLINES

NPAGE = 0 causes page to be restored prior to print.

NPAGE - page no.

HEADER- 80 char. 20A5

ICASE- case no.

KO - 6 print unit

INMAX is max no. of lines allowed

Initially NLINE should be set > LINMAX and NPAGE = 0

SUPPLIED UTILITY ROUTINES
(CONTINUED)

Call EJECT (NLINES)

Causes page to be restored automatically and then prints headers.

Routine CLDAY

Call CLDAY

Given IDAY-DAY no. of the year compute in LMO-the month and in LDA the day no.

Need: IYEAR = 0 - Leap Year, ≠ 0 not Leap Year

Routine KEPLER

Call KEPLER (XM, XECC, XE, ERROR)

Given XM - Mean anomaly, XECC - eccentricity

Compute: E-eccentric anomaly, error = 0 means OK

Routine LFPA

Call LFPA [FLDA, LMO, LYR, ALFGM (can be dummy), DAYS]

Given: FLDA - day of month no., LMO - month no.,
LYR - year no. compute ALFGM - right ascension and
DAYS - Zulu day no.

Routine DEGMOD

Call DEGMOND (RAD, IDEG)

Given: angle rad in radians store the angle in deg., min., sec., in IDEG(1) - (3).

Routine FZULU

Call FZULU (IOATE, IOUT)

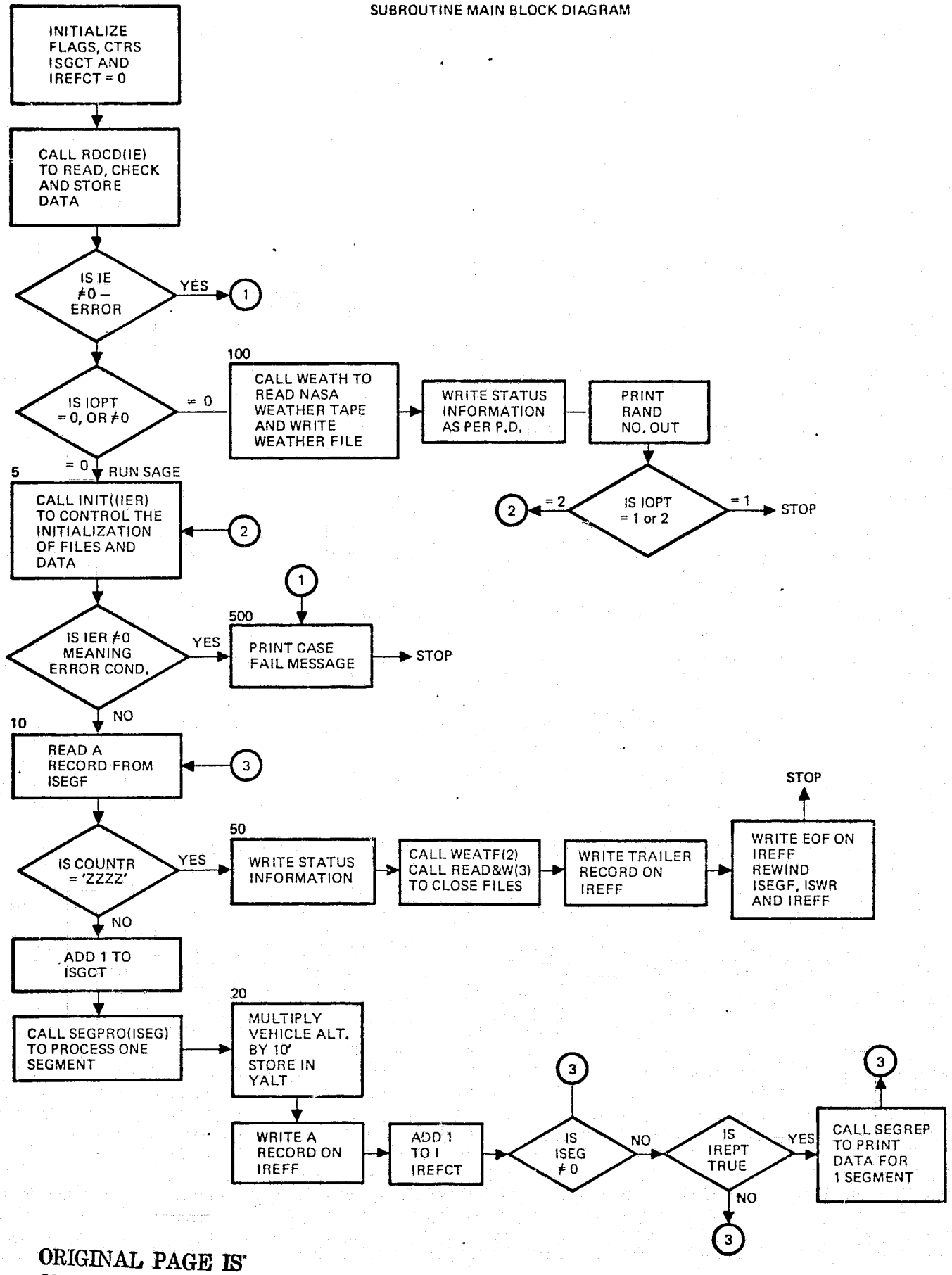
Given Zulu date in IDATE, compute year, month and day in IOUT(1) - IOUT(3).

Routine RDMIA

Call RDMIA(FL, U)

Given double precision random no. seed in FL, comoute random no. U (0-1) based on uniform distribution.

SUBROUTINE MAIN BLOCK DIAGRAM



Subroutine RDCD

CALLING SEQUENCE:

Call RDCD (IE)

IE = 0 - OK, IE ≠ 0 - ERROR

PURPOSE:

The purpose of this routine is to read the header card and data card into working storage and then perform error checks on the input data.

INPUT: CONS COMMONS

INP, KO

OUTPUT; FLAG COMMON

HEADER, ICASE, ICSESW, ICSESG, ISTIME, NDAYS, IOPT, LVEH, IREPT, NMON,

WORK COMMON

DECR, RAND

CONS: IVCT, IGRON, NLATSW

SUBROUTINES USED:

PAGER, EJECT

METHOD/PROCEDURE

1. Read header and store in HEADER, set IE = 0
2. Read data card ISWRTO, IGRDN, NLATSW, NDAYS, LVEH, NMON, ISWRTO, IGRDN, NLATSW and RAND are read to temporary locations. If these quantities are 0 then set NDAYS = 426, LVEH=1, 2; NMON=12; and RAND=1.0, ISWRTO= 1101, IGRDN= 16000, NLATSW=100
3. Call EJECT (2)
4. Perform checks 4 and 5, 8 and set IE=0 if any errors.
5. Return

Subroutine WEATH

CALLING SEQUENCE:

Call WEATH

PURPOSE:

This routine controls the reading of the NASA weather tape and writing of the weather file.

INPUT:

WEATDT COMMON
IGRIDPT, IPERWT
CONS COMMON
IWRAPE
FLAG COMMON
NMON
WORK COMMON: RAND

OUTPUT:

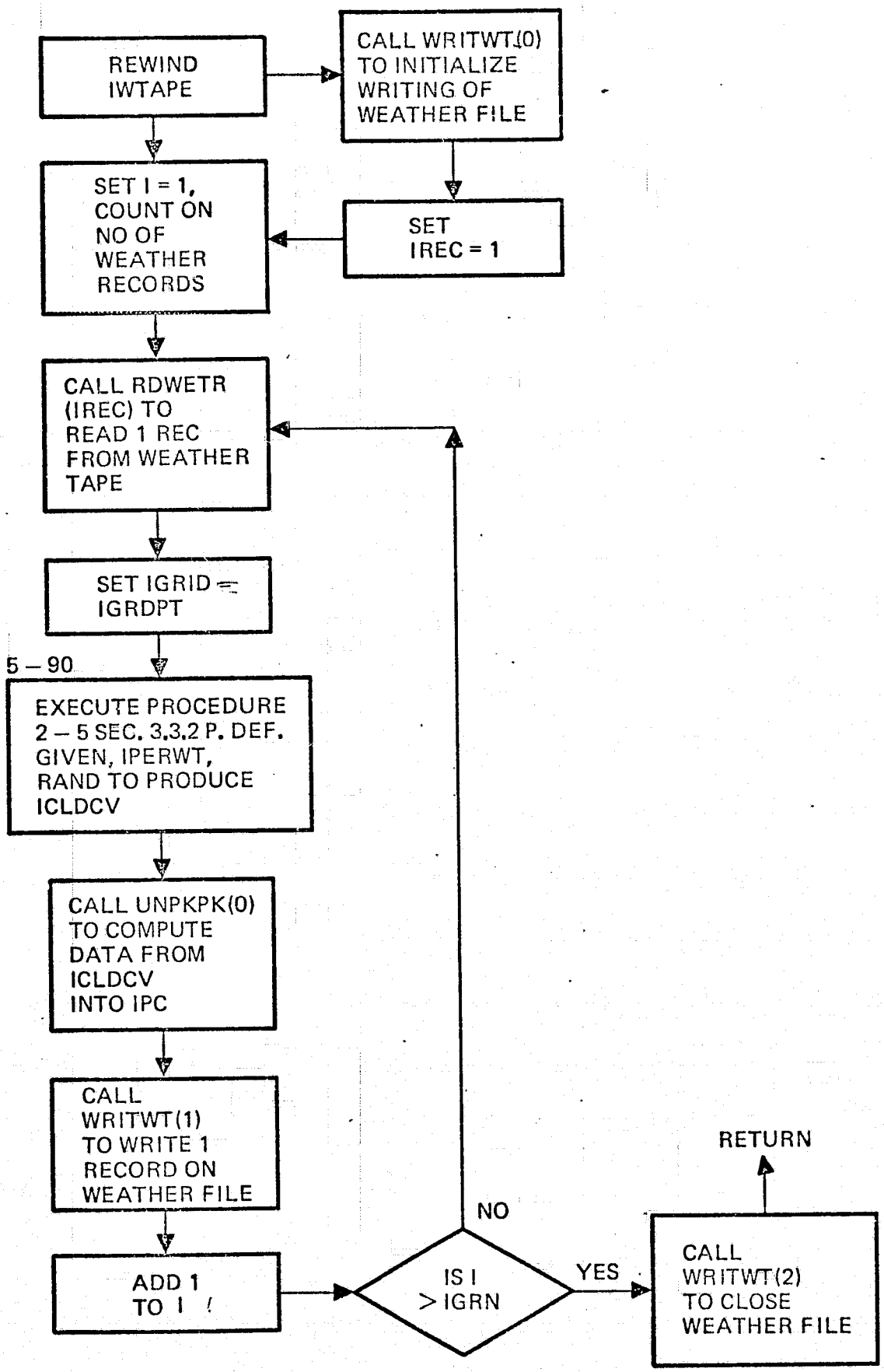
COMMON FLAG
NMAX, IGRID
DIRAC COMMON
IPC
COMMON WORK ICLDCV - Intermediate storage of PC data by day

SUBROUTINES USED: RDWETR, WRITWT, UNPKPK, RDMIA

METHOD/PROCEDURE

See flow chart.

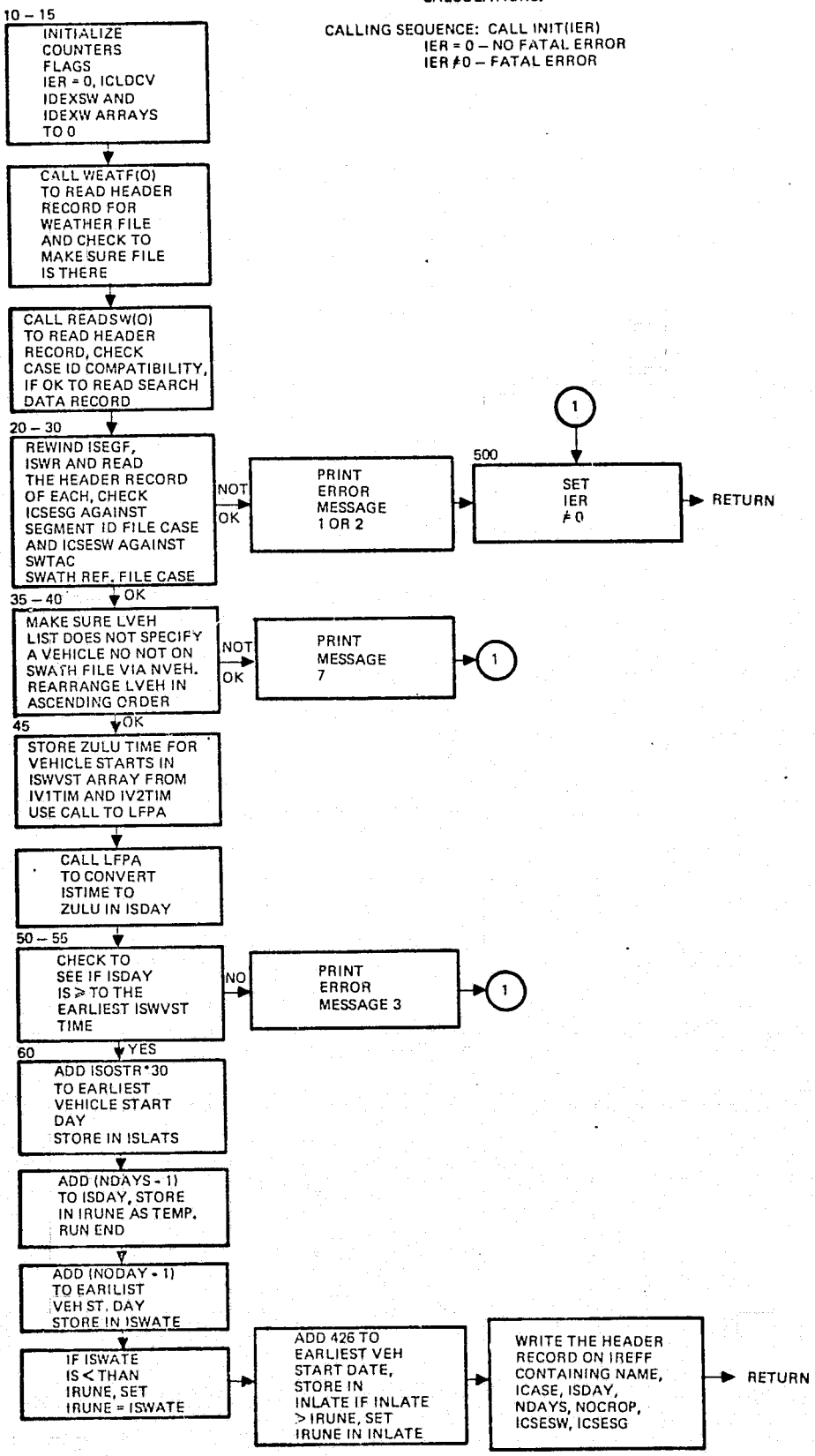
SUBROUTINE WEATH BLOCK DIAGRAM



SUBROUTINE INIT BLOCK DIAGRAM

PURPOSE: THIS ROUTINE INITIALIZES CORE, FILES AND DOES SOME PRELIMINARY CALCULATIONS.

CALLING SEQUENCE: CALL INIT(IER)
IER = 0 – NO FATAL ERROR
IER ≠ 0 – FATAL ERROR



Subroutine READSW

CALLING SEQUENCE:

Call READSW (IND)

IND = 0 - initial, read header record and check for correct file

IND = 1 - normal read 1 data record into working storage

IND = 3 - termination - close file activity

IND = 2 - read search table data record

PURPOSE:

This routine reads the header record or data record from the swath table file - ISWTB and stores the data into working storage. ISWTB is a direct access file

INPUT:

CONS COMMON
KO, ISWTB, NLATSW, ISWRTO

DIRAC COMMON
IDEXSW, A

WORK COMMON
IREV, IVEH

FLAG COMMON
JDAY, JCSESW, NODAY

ISWTB FILE - SWATH TABLE

OUTPUT:

FLAG COMMON
ISOSTR, NLAT, NVEH, IV1TIM, IV2TIM, NODAY

SWATH COMMON
LATNO, IALT, TIME1, DLONG1, DLONMX, DLONMN, INLAT, ISLAT

Subroutine READSW (Continued)

METHOD/PROCEDURE

For options 0, 1, 2 array A is cleared

IND = 0 -define file is given then the header record is read into A, the case no. is compared against ICSCSN and the first 2 words check to be = 'SWATH TB' if not, an error message is printed with a STOP. ISOSTR, NLAT, INLAT, and ISLAT are stored and check LVEH vs. NVEH.

IND = 1 -IREC - record no. = (NODAY + 1)* (IVEH -1) + JDAY+ 1.IREC is read into A and then using NLAT the data is moved from A to SWATH COMMON,NOWDS = 8*NLAT.

IND = 2 -IREC = record no. = (NODAY -1) * (IVEH-1) + NODAY +2. Read search record from ISWTB (NOWDS = 8*NLAT) into A and move the search data to LATNO DLONMX and DLONMN using NLAT

IND = 3 - Return

Subroutine WEATF

CALLING SEQUENCE:

Call WEATF (IND)

IND = 0 - read header record and check to make sure the file is weather file.

IND = 1 - normal read a physical record and move desired logical rec to IPC

IND = 2 - termination

PURPOSE:

This routine, given the grid no., will fetch the weather data for that grid from file IWTHF and store it into IPC for use by the application program.

INPUT: CONS COMMON

IWTHF, IBLKF, NPREC, NLREC, KO

FLAG COMMON

IGRID

DIRAC COMMON

INDEXC, IB

The file as defined in sec. 2.2, item 4.

OUTPUT: FLAG COMMON

NMAX

DIRAC COMMON

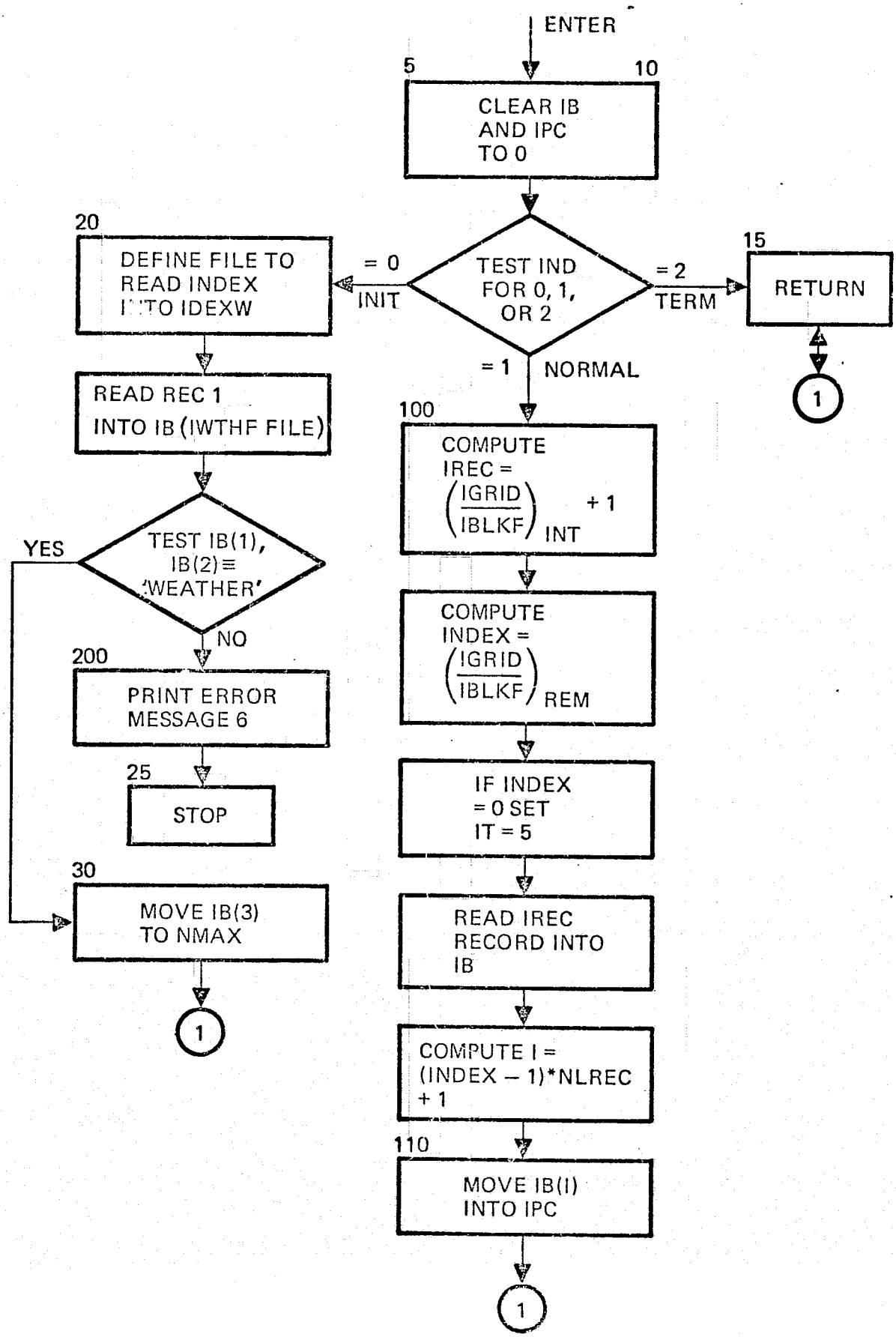
IPC

METHOD/PROCEDURE

See flow chart

SUBROUTINES USED:

SUBROUTINE WEATF BLOCK-DIAGRAM



Subroutine POTEN

CALLING SEQUENCE:

Call POTEN (IND)

IND = 0 - potential access, IND $\neq$ 0 no access

PURPOSE:

This routine determines for a given day and vehicle whether there is a potential access of a given segment. It does this by reading the segment reference file and executing equation set 3.3.2 item 2.3.

INPUT: SWATH COMMON

DLONMN - $\Delta\Omega_2$, LATNO, IALT

SEGFIL COMMON

THETSG - θ_{SEG}, PHISEG - ϕ_{SEG}

CONS COMMON

RADIAN, ISWR

SWATH REFERENCE FILE RECORD WITH FOLLOWING DATA FOR JDAY AND IVEH IN SWATH COMMON

KVEH, KDAY, NREV, TIMNOD - τ_{NODE}, PHINOD - ϕ_{NODE}

FLAG COMMON

JDAY

WORK COMMON

IVEH

OUTPUT: WORK COMMON

IREV, LAT, LAT1, PHINDE, DLNMIN, DLNMAX

METHOD/PROCEDURE

1. Set IND = 0
2. Read data record from ISWR file
3. Compute DLNMIN, DLNMAX, LAT, LAT1
4. Compute IREV where there is an access if no access set IND $\neq$ 0 return.
5. If access, store PHINOD(IREV) into PHINDE and TIMNOD(IREV) into TMNODE
6. Return

SUBROUTINES CALLED: PIMOD, AMAX1, AMIN1

Subroutine ACCVER

CALLING SEQUENCE:

Call ACCVER (IND)

IND = 0 - Access verified, IND ≠ no access

PURPOSE:

This routine verifies that a segment is actually accessed on JDAY for IVEH and IREV. It reads the record from ISWTB file specified by JDAY and IVEH, determines if an access is possible then computes IZDAY. See section 3.3.2 set 4 for equations.

INPUT:

FLAG COMMON
JDAY, ICDAY
WORK COMMON
LAT, LAT1, PHINDE, SA, IVEH, DECR
CONS COMMON
RADIAN
SEGFIL COMMON
THETSG
ISWTB SWATH FILE
ORIGINALLY IN SWATH COMMON
IALT, TIME1, DLONG1

OUTPUT:

WORK COMMON
DELAT, DLONG, TIME, XTIME, XDLONG, XALT, SIGMA, DDLNGL, IZDAY, DDLNGR, CHECKL, CHECKR, CHLONG, TMNODE

METHOD/PROCEDURE

1. IND is set = 0
2. A call to READSW is made to get IALT, TIME1, DLONG1
3. DLONG1, TIME1, and IALT are moved into DLONG, TIME and ALT for latitude subscripts LAT and LAT1. Note IALT is integer KILOM * 10
4. Equation set 4b is executed
5. If no access set IND ≠ 0 and return
6. ICDAY is stored in IZDAY

Subroutine ACCVER (Continued)

METHOD/PROCEDURE (continued)

7. Return

SUBROUTINES CALLED: READSW

Subroutine IMAGE

CALLING SEQUENCE:

Call IMAGE

PURPOSE:

This routine computes the imaging conditions of the access which are to be written out on the segment reference file. See section 3.3.2 of the problem description equation set 5, 6 for equations.

INPUT:

WORK COMMON

SA, ALT, XTIME, XDLONG, CHLONG, TMNODE

OUTPUT:

WORK COMMON

TS, YALT, CTA, EL

METHOD/PROCEDURE

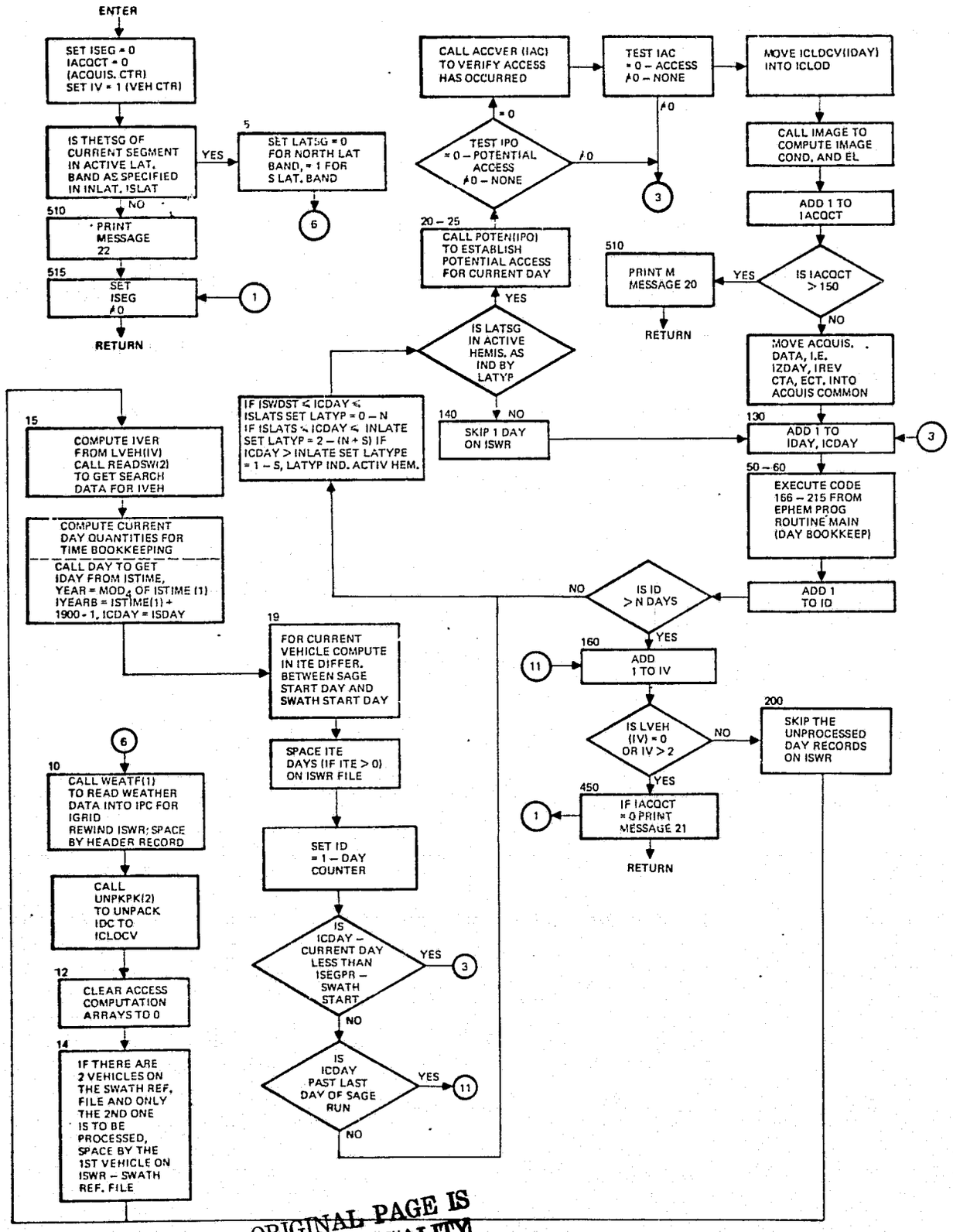
1. Execute equation set 5.
2. Call ALPHA to compute EL.

SUBROUTINES CALLED: ALPHA

SUBROUTINE SEGPRO BLOCK DIAGRAM

PURPOSE: THIS IS A ROUTINE WHICH CONTROLS ALL PROCESSING FOR ONE SEGMENT.

CALLING SEQUENCE: CALL SEGPRO(ISEG)
ISEG, = 0 – OK, ISEG ≠ 0 NO GOOD



Subroutine SEGREP

CALLING SEQUENCE:

Call SEGREP

PURPOSE:

This routine prints the data for all acquisitions for 1 segment. See problem description for the format.

INPUT:

WORK COMMON
IACQCT
ACQUIS COMMON
IZDAYT, IREVT, CTAT, ELT, ICLODT, ICEHT, TST, YALTT
SEGFIL COMMON
COUNTR, SEGMNT
CONS COMMON
KO

OUTPUT: See problem description Section 4.1.1

METHOD/PROCEDURE

1. Call EJECT(7) to restore page print run header
2. Print report headers
3. Print column headers
4. For each acquisition - IACQCT
   Call PAGER(1)
   Call FZULU to convert IZDAYT to YR MON DAY
   Call DEGMOD to convert EL and CTA to DEG, MIN, SEC
   Print 1 line of data

SUBROUTINES CALLED: DEGMOD, EJECT, PAGER, FZULU

Subroutine WRITWT

CALLING SEQUENCE:

Call WRITWT (IND)

IND = 0 - write header record, IND = 1-normal, IND = 2, - termination

PURPOSE: This routine accumulates logical weather data records into a record block and writes the record on the direct access file IWTHF. This routine is called for each grid point.

INPUT: CONS COMMON

IWTHF, IBLKF, NPREC, NLREC

FLAG COMMON

IGRID, NMAX

DIRAC COMMON

IDEXW, IB, IPC

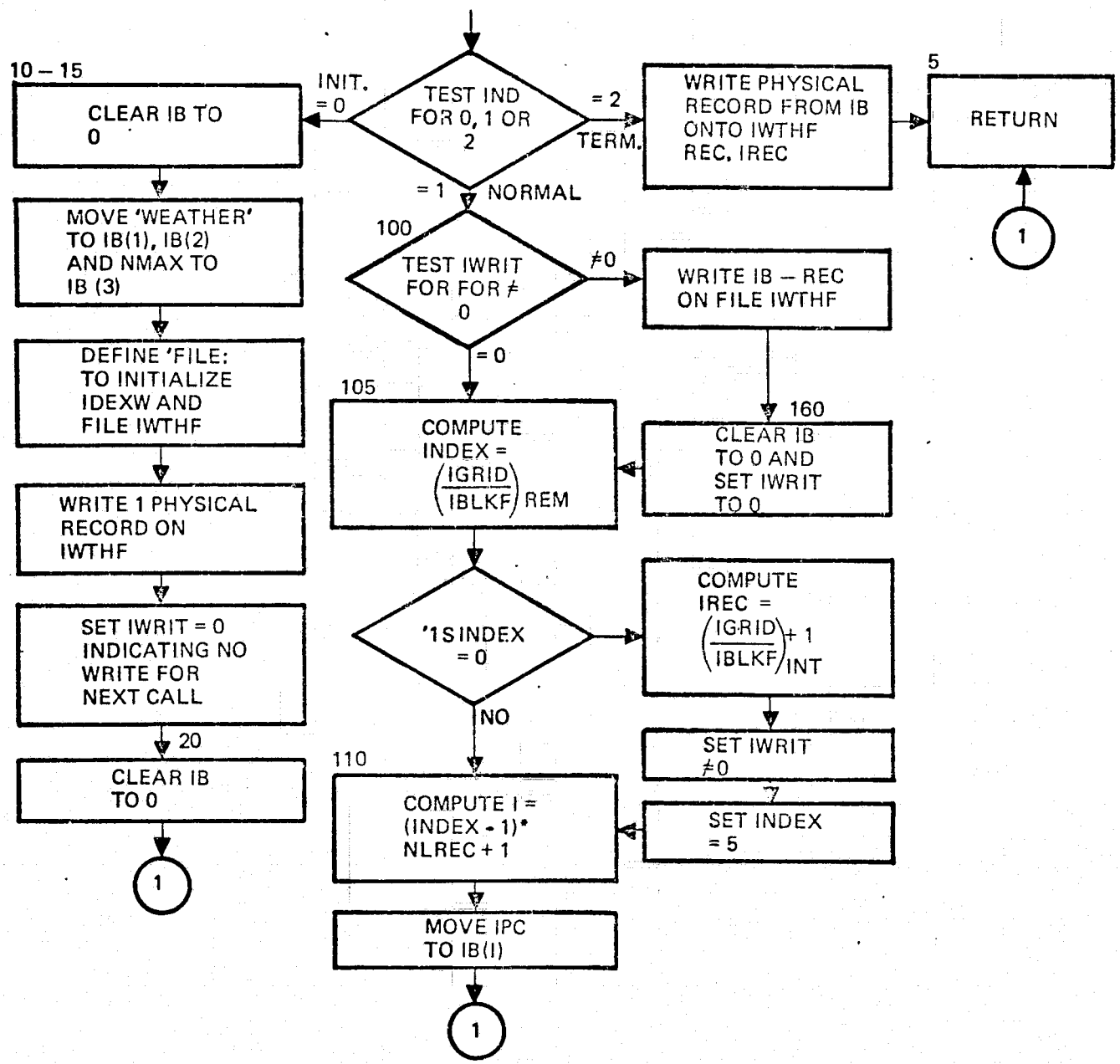
OUTPUT: Either 1 logical record IPC moved into B or B written out on the IWTHF file, see sec. 2.2 item 4. for format details in problem description.

SUBROUTINES USED:

METHOD/PROCEDURE

See flow chart.

SUBROUTINE WRITWT BLOCK DIAGRAM



SUBROUTINE UNPKPK BLOCK DIAGRAM (PAGE 1 OF 2)

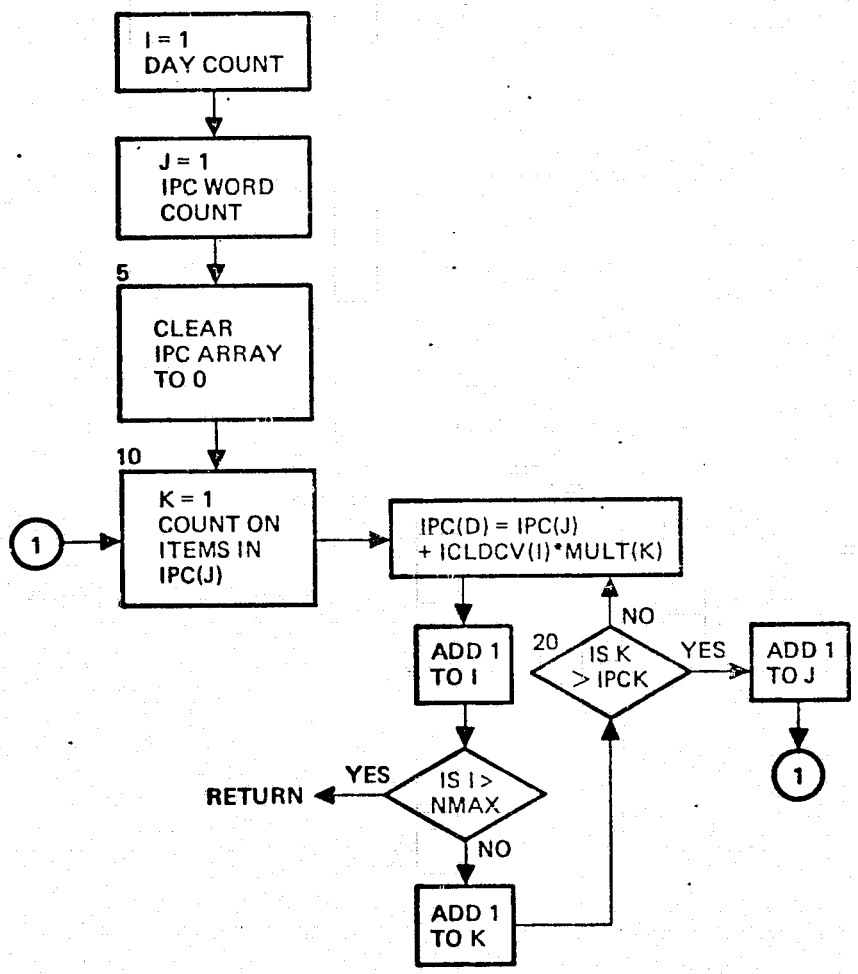
CALLING SEQUENCE:
CALL UNPKPK(IND)
IND = 0 – PACK CLOUD DATA FROM ICLDCV INTO IPC
IND ≠ 0 – UNPACK CLOUD COVER DATA FROM IPC AND STORE IN ICLDCV

PURPOSE: DEPENDING ON IND THIS ROUTINE EITHER CONVERTS INTEGERS 0-8 IN ICLDCV AND THEN PACKS THESE 4 BIT NUMBERS 8 TO A IPC WORD OR CONVERTS AND UNPACKS THE INTEGERS FROM IPC INTO %*10 AND STORES THEM INTO A ICLDCV WORD.

INPUT: WORK COMMON
ICLDCV
DIRAC COMMON
IPC
FLAG COMMON
NMAX
CONS COMMON
NLREC, IPCK

OUTPUT: SAME AS ABOVE EXCEPT FOR NLREC AND IPCK

METHOD/PROCEDURE
IND = 0



METHOD/PROCEDURE (CONTD)

IND ≠0

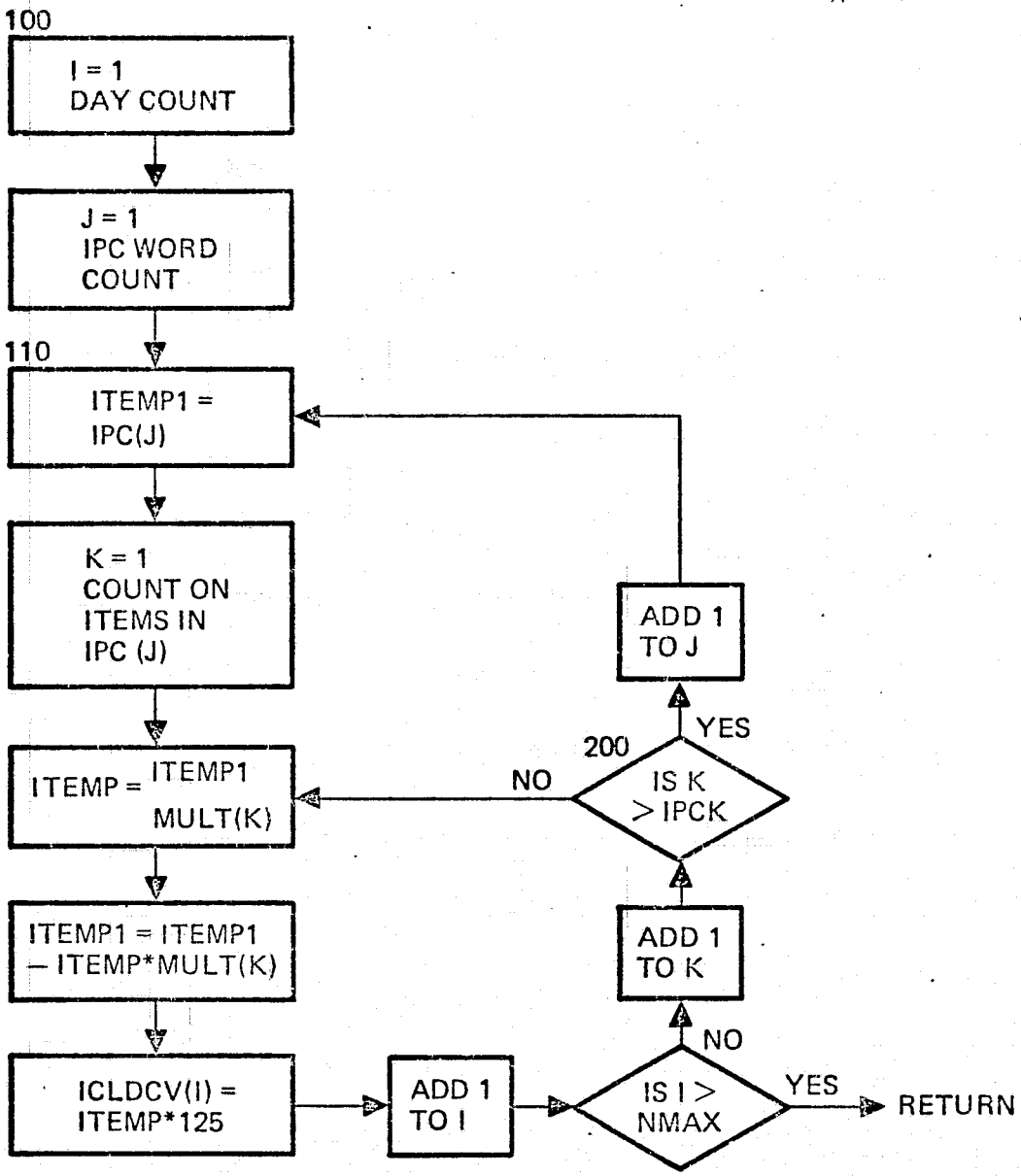


TABLE OF MULTIPLIERS/DIVISORS MULT

K	VALUE
1	2^{28}
2	2^{24}
3	2^{20}
4	2^{16}
5	2^{12}
6	2^{8}
7	2^{4}
8	2^{0}

Subroutine RDWETR

CALLING SEQUENCE:

Call RDWETR (IRC)

IRC is logical record count 1 to 10 initialized to 1 by calling routine for first call only. Routine then adds to counter and resets it as needed to 1.

PURPOSE:

To read one physical record from the NASA tape unit IWTAPE and move 1 logical record into working storage

INPUT:

CONS COMMON NASA WEATHER TAPE
IWTAPE
WEATDT
IA - See Section 2.2 of problem description for format of each logical record

OUTPUT:

WEATDT COMMON
IGROPT, IPERWR

METHOD/PROCEDURE

Use NTRAN to perform the read from the weather tape.

SUBROUTINES CALLED: NTRAN, PAGER

PART V

SUBROUTINE LISTINGS

```
* FLT ACCVER,1,760331, 59935   , 1
000001                SUBROUTINE ACCVER(IND)                                    ACCVER
000002          C     THIS ROUTINE DETERMINES IF AN ACCESS HAS ACTUALLY OCCURED ACCVER
000003                COMMON /FLAG/                                             FLAG
000004               1HEADER(18),ICASE,ICSLSW,ICSESG,ISTIME(5),ISDAY,NDAYS,IOPT,LVEH(2),FLAG
000005               2NVEH,NODAY,IV1TIM(5),IV2TIM(5),IGRID,ISNDST,JDAY,IDAY,IYEAR,IYEARPFLAG
000006               3,ICDAY,ISWVST(2),ISLATS,INLATF,IRUNE,ISWATE,NLAT,NMAX,ISUSTR,IREPTFLAG
000007                LOGICAL IRETT                                             FLAG
000008                COMMON /WORK/      RAND.                                  WORK
000009               1TMNODF,DLNMIN,DLNMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3),  WORK
000010               2DELTA,XTIME(3),XDLONG(3),XALT,SIGMA,DDLNGL,CHECKL,CHECKR,DECR,TS,  WORK
000011               3CTA,YALT,FL,CHLONG,DDLNGR,ICLOCV(366),IZDAY,IREV,IVEH,LAT,LAT1,  WORK
000012               4ICLOD,JACOCT                                             WORK
000013                DOUBLE PRECISION RAND                                     WORK
000014                COMMON /CONS/                                             CONS
000015               1RADIUS,CIS,CNUV,ECCE,PI,PI2,PI32,RADIAN,IROPI,NAME(2),INP,KO,NLINECONS
000016               2,LIGMAX,NPAGE,ISEOF,ISWTB,ISWR,IRLEF,INTNF,IWTAPE,NOCROP,IULKF,  CONS
000017               3IPLK,NPREC,LREC,ISWRTO,IVCT,NLATSW,IGRDN                 CONS
000018                REAL NAME                                                 CONS
000019                COMMON /SEGFIL/                                           SEGFIL
000020               1COUNTR(2),THETSG,PHISEG,IREG,IZONE,ISTRAT,ISUBST,SEGMNT,ITRPRL(6)      - *NEW
000021               2,ITRSEG,ISPRW                                            *NEW
000022                INTEGER SEGMNT                                            SEGFIL **-2
000023                REAL ICROP                                                SET1
000024                COMMON /SWATH/                                            SWATH
000025               1TIME1(100,3),DLONG1(100,3),DLONMN(100),DLONMX(100),TIMNOD(17),  SWATH
000026               2PHIMOD(17),LATNO(100),IALT(100),KVEH,KDAY,NPEV,INLAT(2),ISLAT(2)  SWATH
000027                IND = 0                                                   ACCVER
000028                CALL READSW(1)                                            ACCVER
000029                DELTA = THETSG - LATNO(LAT)/RADIAN                        ACCVER
000030                DELTA = DELTA*RADIAN /(LATNO(LAT1)-LATNO(LAT))            SET2
000031                DO 10 I=1,3                                               ACCVER
000032                DLONG(I,1) =DLONG1(LAT,I)                                 ACCVER
000033                DLONG(I,2) = DLONG1(LAT1,I)                               ACCVER
000034                TIME(I,1) = TIME1(LAT,I)                                  ACCVER
000035                TIME(I,2) = TIME1(LAT1,I)                                 ACCVER
000036             10 CONTINUE                                                  ACCVER
000037                ALTEM  = IALT(LAT)                                        ACCVER
000038                ALT(1) = ALTEM/10.                                        ACCVER
000039                ALTEM  = IALT(LAT1)                                       ACCVER
000040                ALT(2) = ALTEM/10.                                        ACCVER
000041                DO 20 I=1,3                                               ACCVER
000042                XTIME(I) = TIME(I,1) + DELTA*(TIME(I,2) - TIME(I,1))      ACCVER
000043                XDLONG(I) = DLONG(I,1) + DELTA*(DLONG(I,2) - DLONG(I,1))  ACCVER
000044             20 CONTINUE                                                  ACCVER
000045                TEMP = (ALT(2) - ALT(1))/(TIME(2,2) - TIME(2,1))          ACCVER
000046                XALT = ALT(1) + TEMP*(XTIME(2) - TIME(2,1))               ACCVER
000047                SIGMA = ASIN(((RADIUS + XALT)/RADIUS)*SIN(SA(3)))- SA(3)  ACCVER
000048                TEMP1 = - DECR/(RADIUS*SIGMA)                             ACCVER
000049                DDLNGL = TEMP1*(XDLONG(3) - XDLONG(2))                    ACCVER
000050                DDLNGR = TEMP1*(XDLONG(2) - XDLONG(1))                    ACCVER
000051                CHECKL = XDLONG(3) + DDLNGL                               ACCVER
000052                CHECKR = XDLONG(1) - DDLNGR                               ACCVER
000053                CHLONG = PIMOD(PHISEG - PHINDF)                           ACCVER
000054                IF(CHLONG .LT. CHECKL)GO TO 100                           ACCVER
000055                IF(CHLONG .GT. CHECKR)GO TO 100                           ACCVER
000056                IZDAY = ICDAY                                             ACCVER
000057                GO TO 110                                                 ACCVER
000058             100 IND = 1                                                  ACCVER
```

ACCVER
ACCVER

110 RETURN
END

000059
000060

```
FLT ALPHA,1,760331, 59936    * 1

000001            SUBROUTINE ALPHA(IFLAG)                                        ALPHA
000002       C                                                                   ALPHA
000003       C                    CALL ALPHA(IFLAG)                              ALPHA
000004       C                                                                   ALPHA
000005       C                          INPUT --  IFLAG-- =1 COMPUTE RT. ASC. OF TALPHA
000006       C                                                                   ALPHA
000007       C                                                                   ALPHA
000008            COMMON /CONS/                                                  CONS
000009           1RADIUS,CIS,CNUV,ECCE,PI,PI2,PI32,RADIAN,TWOPI,NAME(2),INP,KO,NLINECONS
000010           2,LINMAX,NPAGE,ISEOF,ISWTB,ISWR,IREFF,IWTHF,IWTAPE,NOCROP,IBLKF,  CONS
000011           3IPCK,NPREC,NLREC,ISWRTO,IVCT,NLATSW,IGRDN                        CONS
000012            REAL NAME                                                      CONS
000013            COMMON /FLAG/                                                  FLAG
000014           1HEADER(18),ICASE,ICSESW,ICSESG,ISTIME(3),TSDAY,NDAYS,IOPT,LVEN(2),FLAG
000015           2LVEN,NODAY,IV1TIM(3),IV2TIM(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARRFLAG
000016           3,ICDAY,ISWVS(2),ISLATS,INLATE,IRUNE,ISWATE,NLAT,NMAX,ISUSTR,IRLPTFLAG
000017            LOGICAL IPEPI                                                  FLAG
000018            COMMON /WORK/       RAND,                                      WORK
000019           1TWOON,DLNMTH,DLNMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3),  WORK
000020           2DELTA,XTIME(3),XDLONG(3),XALT,SIGMA,DDLNGL,CHECKL,CHECKR,DECR,TS,WORK
000021           3CTA,YALT,FL,CHLONG,DDLNGR,ICLDCV(366),IZDAY,IMEV,IVFH,LAT,LAII,  WORK
000022           4ICLOD,IALOCT                                                  WORK
000023            DOUBLE PRECISION RAND                                         WORK
000024            COMMON /SEGFIL/                                               SEGFIL
000025           1COUNTP(2),THETSG,PHISEG,IREG,IZONE,ISTRAT,ISUBST,SEGMNT,ITRPRL(6)      *NEW
000026           2,ITRSEG,ISPRM                                                          *NEW
000027            INTEGER SEGMNT                                               SEGFIL **-2
000028            REAL ICROP                                                   SEGFIL
000029       C                                                                   ALPHA
000030       C                                                                   ALPHA
000031       C  COMPUTE DAY OF YEAR (SINCE NOT AVAILABLE IN PREDICTOR)           ALPHA
000032       C                                                                   ALPHA
000033       C  ERROR TERM DUE TO VARIABLE SPEED OF ROTATION OF EARTH            ALPHA
000034            ER = 0.                                                        ALPHA
000035       C  CALCULATE MEAN ANOMALY OF EARTH                                  ALPHA
000036            DAY = IDAY                                                     ALPHA
000037            EM = (DAY-2.)*TWOPI/365.25                                     ALPHA
000038       C  CALCULATE ECC. ANOMALY OF EARTH                                  ALPHA
000039            EP = .001/RADIAN                                               ALPHA
000040            E1 = EM                                                        ALPHA
000041           10 E2 = E1-((E1-EM-ECCE*SIN(E1))/(1.-ECCE*COS(E1)))             ALPHA
000042            ED = ABS(E2-E1)                                                ALPHA
000043            E1 = E2                                                        ALPHA
000044            IF(ED-EP) 20,20,10                                             ALPHA
000045           20 IF(E1-TWOPI) 40,50,30                                        ALPHA
000046           30 E1 = E1-TWOPI                                                ALPHA
000047           40 IF(ABS(E1-PI)-EP) 41,41,45                                   ALPHA
000048           45 IF(E1) 46,55,60                                              ALPHA
000049           46 E1 = E1+TWOPI                                                ALPHA
000050              GO TO 40                                                     ALPHA
000051       C  TRUE ANOMALY OF EARTH                                            ALPHA
000052           41 XNUE = PI                                                    ALPHA
000053              GO TO 70                                                     ALPHA
000054           50 E1 = 0.                                                      ALPHA
000055           55 XNUE = 0.                                                    ALPHA
000056              GO TO 70                                                     ALPHA
000057           60 XNUE = 2.*ATAN(SQRT((1.+ECCE)/(1.-ECCE))*TAN(E1/2.))         ALPHA
000058              IF(XNUE) 65,70,70                                            ALPHA
```

```
000059          65 XNUE = TWOPI+XNUE                                                ALPHA
000060        C  DECLINATION OF SUN                                                 ALPHA
000061          70 THES = ASIN(SIN(CIS)*SIN(XNUE-CNUV))                             ALPHA
000062             DIF = XNUE-CNUV                                                  ALPHA
000063             COSDIF = COS(DIF)                                                ALPHA
000064             SINDIF = SIN(DIF)                                                ALPHA
000065             IF(COSDIF) 90,75,90                                              ALPHA
000066          75 IF(SINDIF) 85,80,80                                              ALPHA
000067        C  RT.ASC. OF TRUE SUN                                                ALPHA
000068          80 ALPHAT = PI2                                                     ALPHA
000069             GO TO 150                                                        ALPHA
000070          85 ALPHAT = 3.*PI/2.                                                ALPHA
000071             GO TO 150                                                        ALPHA
000072          90 ALPHAT = ATAN(COS(CIS)*SINDIF/COSDIF)                            ALPHA
000073             IF(COSDIF) 105,95,95                                             ALPHA
000074          95 IF(ALPHAT) 100,150,150                                           ALPHA
000075         100 ALPHAT = TWOPI+ALPHAT                                            ALPHA
000076             GO TO 150                                                        ALPHA
000077         105 ALPHAT = ALPHAT+PI                                               ALPHA
000078        C  RT. ASC. OF MEAN SUN                                               ALPHA
000079         150 ALPHAM = (279.69668+360./365.25*DAY)/RADIAN                      ALPHA
000080        C                                                                     ALPHA
000081             PHIS = (43200. - TS)*TWOPI/86400. - ALPHAM+ ALPHAT - EM          ALPHA
000082             ARG1 = COS(THES + THETSG)                                        ALPHA
000083             ARG2 = COS(PHIS - PHISEG)                                        ALPHA
000084             ARG3 = COS(THES - THETSG)                                        ALPHA
000085             EL = PI2 - ACOS(.5*(ARG1*(ARG2-1.)+ (ARG3*(ARG2+1.))))          ALPHA
000086         1000 RETURN                                                          ALPHA
000087             END                                                              ALPHA
```

```
...T BLKDAT,1,751007, 50490    , 1

000001                BLOCK DATA                                                    BLOCK
000002                COMMON /CONS/                                                 CONS
000003               1RADIUS,CIS,CNUV,ECCE,PI,PI2,PI32,RADIAN,TWOPI,NAME(2),INP,KO,NLINECONS
000004               2,LINMAX,NPAGE,ISEGF,ISWTB,ISWR,IREFF,INTHF,IWTAPE,NOCROP,IBLKF,   CONS
000005               3IPCK,NPREC,NLREC,ISWRTO,IVCT,NLATSW,IGRDN                        CONS
000006                REAL NAME                                                     CONS
000007                COMMON /FLAG/                                                 FLAG
000008               1HEADER(16),TCASE,ICSESN,ICSFSC,ISTIME(3),ISDAY,NDAYS,IOPT,LVER(2),FLAG
000009               2NVER,MODAY,IV1TIM(3),IV2TIM(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARPFLAG
000010               3,ICDAY,ISWVST(2),ISLATS,INLATF,IRUNF,ISWATE,NLAT,NMAX,ISUSTR,IKCPTFLAG
000011                LOGICAL IBLPI                                                 FLAG
000012                COMMON /WORK/      RAND,                                      WORK
000013               1IMMODE,DLNMIN,DLNMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3),   WORK
000014               2DELTA,XTIME(3),XDLONG(3),YALT,SIGMA,DDLNGL,CHECKL,CHECKR,DECK,IS,WORK
000015               3CTA,YALT,FL,CHLONG,DDINCR,ICLDCV(366),IZDAY,IREV,IYEN,LAT,LAT1,  WORK
000016               4ICLOD,IACCCT                                                 WORK
000017                DOUBLE PRECISION RAND                                        WORK
000018                DATA READER/18*0H     / ,NDAYS/426/,RAND/1.0/,INP/5/,KO/6/,   BLOCK
000019               1NLINE/40/,LINMAX/39/,NPAGE/0/,ISEGF/1/,ISWTB/8/,ISWR/9/,IREFF/10/,BLOCK
000020               2INTHF/11/,IWTAPE/15/,RADIUS/6376.436/,CIS/.409280/,CNUV/1.55695/,      *NEW
000021               3ECCE/.167263E-1/,PI/3.141592/,PI2/1.5707963/,PI32/4.712389/,     BLOCK **-1
000022               4RADIAN/57.29578/,TWOPI/6.2831852/,NAME/4HSEG ,4HREFE/,NOCROP/4/,  BLOCK
000023               5IBLKF/5/,IIPCK/0/,NPREC/250/,NLREC/46/,ISWRTO/1101/,NLATSW/100/,  BLOCK
000024               6IGRDN/16000/,NMAX/366/                                       BLOCK
000025                END
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
      ...LT DAY,1,751001, 67870      , 1
000001                SUBROUTINE DAY(IYMD,IDAY)                                 DAY
000002          C                                                               DAY
000003          C  CONVERTS MONTH, DAY TO DAY NUMBER                            DAY
000004          C                                                               DAY
000005                DIMENSION IYMD(3),IMONTH(12),IMON(12)                     DAY
000006          C                                                               DAY
000007                DATA IMONTH/0,31,59,90,120,151,181,212,243,273,304,334/   DAY
000008                DATA IMON/0,31,60,91,121,152,182,213,244,274,305,335/     DAY
000009          C                                                               DAY
000010                MONTH=IYMD(2)                                             DAY
000011                IDAY=IYMD(3)                                              DAY
000012                IYEAR=MOD(IYMD(1),4)                                      DAY
000013                IF(IYEAR .EQ. 0)GO TO 10                                  DAY
000014          C                                                               DAY
000015          C                                                               DAY
000016          C                                                               DAY
000017                IF (MONTH.NE.0) IDAY=IDAY+IMONTH(MONTH)                   DAY
000018          C                                                               DAY
000019              5 RETURN                                                    DAY
000020             10 IF(MONTH .NE. 0) IDAY = IDAY + IMON(MONTH)                DAY
000021                GO TO 5                                                   DAY
000022          C                                                               DAY
000023                END                                                       DAY
```

```
*LT DEGMOD,1,751001, 67876    , 1
000001                SUBROUTINE DEGMOD(RAD,IDEG)                                DEGMOD
000002          C                                                                DEGMOD
000003          C                                                                DEGMOD
000004          C     DEGMOD CONVERTS RADIANS INTO DEGREES, MINUTES, AND SECONDS DEGMOD
000005          C             AND MODS  THE DEGREES BETWEEN -360 TO +360         DEGMOD
000006          C                                                                DEGMOD
000007          C     IDEG(1)=DEGREES                                            DEGMOD
000008          C     IDEG(2)=MINUTES                                            DEGMOD
000009          C     IDEG(3)=SECONDS                                            DEGMOD
000010          C     RAD    = RADIANS                                           DEGMOD
000011          C     RADIAN = DEGREES PER RADIAN                                DEGMOD
000012          C     RDUM   = R*8 TEMPORARY STORAGE                             DEGMOD
000013          C                                                                DEGMOD
000014                INTEGER IDEG(3)                                            DEGMOD
000015          C                                                                DEGMOD
000016                DATA RADIAN/57.29578/                                      DEGMOD
000017          C     CONVERT RADIANS TO DEGREES, MINUTES, AND SECONDS           DEGMOD
000018          C                                                                DEGMOD
000019                RDUM = RAD*RADIAN                                          DEGMOD
000020                IDEG(1)=RDUM                                               DEGMOD
000021                RDUM=ABS(RDUM-IDEG(1))*60.                                 DEGMOD
000022                IDEG(2)=RDUM                                               DEGMOD
000023                IDEG(3)=(RDUM-IDEG(2))*60.+.5                              DEGMOD
000024          C                                                                DEGMOD
000025                IF(IDEG(3).LT.60) GO TO 10                                 DEGMOD
000026                IDEG(3)=IDEG(3)-60                                         DEGMOD
000027                IDEG(2)=IDEG(2)+1                                          DEGMOD
000028          C                                                                DEGMOD
000029           10   IF(IDEG(2).LT.60) GO TO 20                                 DEGMOD
000030                IDEG(2)=IDEG(2)-60                                         DEGMOD
000031          C                                                                DEGMOD
000032                ISIGN=1                                                    DEGMOD
000033                IF(IDEG(1).LT.0) ISIGN=-1                                  DEGMOD
000034                IDEG(1)=IDEG(1)+ISIGN                                      DEGMOD
000035          C                                                                DEGMOD
000036           20   CONTINUE                                                   DEGMOD
000037          C                                                                DEGMOD
000038                IDEG(1)=MOD(IDEG(1),360)                                   DEGMOD
000039                RETURN                                                     DEGMOD
000040                END                                                        DEGMOD
```

```
* FLT EJECT,1,760331, 59946     , 1

000001                SUBROUTINE EJECT(NLINES)                                  EJECT
000002          C     RESTORE PAGE AND PRINT PAGE HEADER                        EJECT
000003                COMMON /FLAG/                                             FLAG
000004               1HEADER(18),ICASE,ICSESH,ICSESC,ISTIME(3),ISDAY,NDAYS,IOPT,LVEH(2),FLAG
000005               2NVEH,NODAY,IV1IN(3),IV2IN(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARUFLAG
000006               3,ICDAY,ISWVST(2),ISLATS,INLATF,IRUNF,ISRATE,NLAT,NMAX,ISUSIR,IREPTFLAG
000007                LOGICAL IREPT                                             FLAG
000008                COMMON /CONS/                                             CONS
000009               1RADIUS,CIS,CONV,ECCF,PI,PI2,PI32,RADIAN,IWOPT,NAME(2),INP,KO,NLINECONS
000010               2,LINMAX,NPAGE,ISEGF,ISHTO,ISWR,IREFF,IWTHF,IWTAPE,NOCROP,IBLKF,  CONS
000011               3IPCK,IPREC,NLREC,ISWRTO,IVCT,NLATSW,IGRDN                CONS
000012                REAL NAME                                                 CONS
000013                IF(NLINES .EQ. 0) RETURN                                  EJECT
000014                NLINE = NLINES                                            EJECT
000015                NPAGE = NPAGE + 1                                         EJECT
000016                WRITE(KO,9010)                                            EJECT
000017                WRITE(KO,9020)HEADER,ICASE,NPAGE                          EJECT
000018                RETURN                                                    EJECT
000019           9010 FORMAT(1H1)                                               EJECT
000020           9020 FORMAT(1X,18A4,18H  LPP SIMULATION  ,5HCASE ,I4,2X,4HPAGE,I4)      *NEW
000021                END                                                       EJECT   **-1
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
LT FZULU,1,751007, 50496     , 1

000001            SUBROUTINE FZULU(IDATE,IOUT)                                    FZULU
000002          C ************************************************************FZULU
000003          C                                                                 FZULU
000004          C ROUTINE       FZULU                                             FZULU
000005          C                                                                 FZULU
000006          C PURPOSE       TO CONVERT ZULU DATE TO YEAR MONTH AND DAY        FZULU
000007          C                                                                 FZULU
000008          C LINKAGE       CALL FZULU (IDATE,IOUT)                           FZULU
000009          C                    IDATE,  BINARY INTEGER ZULU DATE             FZULU
000010          C                    IOUT,   CALENDER DATE,DIMENSION 3  FOR INTEGER FZULU
000011          C                            YEAR, MONTH AND DAY RESPECTIVELY     FZULU
000012          C                                                                 FZULU
000013          C ROUTINES CALLED   NONE                                          FZULU
000014          C                                                                 FZULU
000015          C LOCAL VARIABLES:                                                FZULU
000016          C                                                                 FZULU
000017          C                    IDAYS      VECTOR CONTAINING NO. DAYS OF YEAR ON FZULU
000018          C                               LAST DAY OF MONTHS 0 THROUGH 12 FOR NORMAL FZULU
000019          C                               YEAR FOLLOWED BY 13 MONTHS OF LEAP YEAR FZULU
000020          C                                                                 FZULU
000021          C                    NLEAP,     NO. OF LEAP YEARS SINCE 1900 TO SPECIFIED FZULU
000022          C                               DATE                              FZULU
000023          C                    NFLG,      SUBSCRIPT INTO IDAYS VECTOR       FZULU
000024          C                                    START=1, NOT LEAP YEAR       FZULU
000025          C                                    START=14, LEAP YEAR          FZULU
000026          C                    JDAY,      JULIAN DAY OF YEAR                FZULU
000027          C                    I,         LOCAL USE                         FZULU
000028          C COMMENTS           NONE                                         FZULU
000029          C************************************************************FZULU
000030                DIMENSION IOUT (3)                                          FZULU
000031                DIMENSION IDAYS(26)                                               *NEW
000032                DATA IDAYS/0,31,59,90,120,151,181,212,243,273,304,334,365,  FZULU **-1
000033               1            0,31,60,91,121,152,182,213,244,274,305,335,366/ FZULU
000034          C           FIND NO. LEAP YEARS SINCE 1900 AND GET CURRENT YEAR   FZULU
000035                IDATE = IDATE + 18263                                       SET1
000036                DO 5 I=1,3                                                  SET1
000037                IOUT(I) = 0                                                 SET1
000038               5 CONTINUE                                                   SET1
000039                NLEAP=(1096+IDATE)/1461                                     FZULU
000040                IOUT(1) = (IDATE - NLEAP)/365                                     *NEW
000041          C           SEE IF CURRENT YEAR ISLEAP YEAR-SET NFLG=1 OR 14      FZULU **-1
000042                I=IOUT(1)/4                                                       *NEW
000043                I=I*4                                                       FZULU **-1
000044                NFLG=1                                                      FZULU
000045                IF(IOUT(1) - I) 10,10,20                                          *NEW
000046          10    NFLG=14                                                     FZULU **-1
000047          C           JULIAN DAY = ZULU DAY-NO YEARS*365-NO.LEAP YEARS      FZULU
000048           20   JDAY = IDATE - (IOUT(1) * 365) - NLEAP + 1                        *NEW
000049          C           GET DAY MONTH FROM TABLE SEARCH                       FZULU **-1
000050          C                                                                 FZULU
000051          50    IF(JDAY-IDAYS(NFLG))50,50,40                                FZULU
000052          40    IOUT(2)=IOUT(2)+1                                           FZULU
000053                NFLG=NFLG+1                                                 FZULU
000054                GOTO 30                                                     FZULU
000055          50    NFLG=NFLG-1                                                 FZULU
000056                IOUT(3)=JDAY-IDAYS(NFLG)                                    FZULU
000057                RETURN                                                      FZULU
000058                END                                                         FZULU
```

```
* cLT IMAGE,1,751001, 67864     , 1

000001                SUBROUTINE IMAGE                                          IMAGE
000002          C     THIS ROUTINE COMPUTES THE IMAGING CONDITIONS OF AN ACCESS IMAGE
000003                COMMON /WORK/       RAND,                                 WORK
000004               1INROUT,DENOIN,DLNMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3), WORK
000005               2DELTA,XTIME(3),XDLONG(3),XALT,SIGMA,DDENGL,CHECKL,CHECKR,DECR,IS, WORK
000006               3CTA,YALT,FL,CHLONG,DDINGR,ICLOCV(366),I7DAY,IREV,IVEH,LAT,LAT1, WORK
000007               4ICLOD,IALOCT                                              WORK
000008                DOUBLE PRECISION RAND                                     WORK
000009                COMMON /CONS/                                             CONS
000010               1RADIUS,CIS,CNUV,ECCF,PI,PI2,PI32,RADIAN,IWOPI,NAME(2),INP,KO,NLINF CONS
000011               2,LINMAX,NPAGE,ISEGF,ISWTB,ISWR,IREPF,IWTHF,IWTAPE,NOCROP,IULKF, CONS
000012               3IPCR,NPFIC,NIREC,ISWRTO,IVCT,NLATSW,IGRDN                 CONS
000013                REAL NAME                                                 CONS
000014                COMMON /FLAG/                                             FLAG
000015               1HEADER(18),RELEASE,ICSESW,ICSESG,ISTIME(3),ISDAY,NDAYS,IOPT,LVEH(2),FLAG
000016               2NVEH,NODAY,IV1TIM(3),IV2TIM(3),IGRID,ISWDSI,JDAY,IDAY,IYEAR,IYEARBFLAG
000017               3,ICDAY,ISRVST(2),ISLATS,INLATF,IRUNE,ISWATE,NLAT,NMAX,ISUSIR,IREPTFLAG
000018                LOGICAL IREPT                                             FLAG
000019                IF(CHLONG .NE. XDLONG(2))GO TO 10                         IMAGE
000020                IS = XTIME(2) + TMNODE                                    IMAGE
000021                CTA = 0.0                                                 IMAGE
000022                YALT = XALT                                               IMAGE
000023                GO TO 50                                                  IMAGE
000024           10   IF(CHLONG .GT. XDLONG(2))GO TO 20                         IMAGE
000025                TEMP = (CHLONG - XDLONG(2))/(XDLONG(3) - XDLONG(2))       IMAGE
000026                IS = XTIME(2) + TEMP*(XTIME(3) - XTIME(2)) + TMNODE       IMAGE
000027                CTA = ATAN(TEMP*TAN(SA(3)))                               IMAGE
000028                TEMP2 = (IS - TMNODE - TIME(2,1))/(TIME(2,2)-TIME(2,1))   SET2
000029                YALT = ALT(1) + TEMP2*(ALT(2) - ALT(1))                   IMAGE
000030                GO TO 50                                                  IMAGE
000031           20   TEMP1 = (CHLONG - XDLONG(2))/(XDLONG(1) - XDLONG(2))      IMAGE
000032                IS = XTIME(2) + TEMP1*(XTIME(1) - XTIME(2)) + TMNODE      IMAGE
000033                CTA = -ATAN(TEMP1*TAN(SA(3)))                             IMAGE
000034                TEMP2 = (IS - TMNODE - TIME(2,1))/(TIME(2,2) - TIME(2,1)) SET2
000035                YALT = ALT(1) + TEMP2*(ALT(2) - ALT(1))                   IMAGE
000036           50   CALL ALPHA(IG)                                            IMAGE
000037                RETURN                                                    IMAGE
000038                END                                                       IMAGE
```

```
* FLT INIT,1,760429, 64008     , 1

000001                SUBROUTINE INIT(IER)                                          INIT
000002          C     THIS ROUTINE INITIALIZES CORE,FILES AND DOES SOME PRELIMINARY CA6,INIT
000003          C     ULATIONS                                                      INIT
000004                COMMON /FLAG/                                                 FLAG
000005               1HEADER(18),ICASE,ICSESW,ICSESG,ISTIME(3),ISDAY,NDAYS,IOP1,LVEH(2),FLAG
000006               2NVEH,NDCAY,IV1TIM(3),IV2TIM(3),IGRID,ISWDS1,JDAY,IDAY,IYEAR,IYEARB,FLAG
000007               3,ICDAY,ISWVST(2),JSLATS,INLATF,IRUNE,ISKATE,NLAT,NMAX,ISUSTR,IRLPT,FLAG
000008                LOGICAL IRLPT                                                 FLAG
000009                COMMON /CONS/                                                 CONS
000010               1RADIUS,GIS,CNUV,ECCE,PI,PI2,PI32,RADIAN,INOPI,NAME(2),INP,KO,NLINE,CONS
000011               2,IIRMAX,NPAGE,ISEGF,ISWTO,ISWR,IREEF,INTHF,INTAPE,NOGROP,IBLKF,  CONS
000012               3IPLK,LPREC,NLREC,ISKRTO,IVCT,NLATSW,IGRON                    CONS
000013                REAL LABL                                                     CONS
000014                COMMON /WORK/      RAND,                                      WORK
000015               1IMPROF,GLONMIN,GLONMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3),  WORK
000016               2DELTA,XTIPL(3),XDELAG(3),XALT,SIGMA,DDLNGL,CHECKL,CHECKR,DECR,TS,  WORK
000017               3CTA,YALT,EL,CHLONG,DDLNGR,ICLDCV(366),IZDAY,IREV,IVEH,LAT,LAT1,  WORK
000018               4ICLPD,IACCCT                                                 WORK
000019                DOUBLE PRECISION RAND                                        WORK
000020                COMMON /DIRAC/                                               DIRAC
000021               1IA(890),IB(230),IPC(46)                                      DIRAC
000022                COMMON /SWATH/                                               SWATH
000023               1TIME1(100,3),DLONG1(100,3),DLONMN(100),DLONMX(100),TIMNOD(17),  SWATH
000024               2PHINOD(17),LATNO(100),IALT(100),KVEH,KDAY,NREV,INLAT(2),ISLAT(2)  SWATH
000025                COMMON /SEGFIL/                                              SEGFIL
000026               1COUNTR(2),THLTSG,PHISEG,IREG,IZONE,ISTRAT,ISURST,SEGMNT,ITRPRL(6)
000027               2,IIRSEG,ISPRW
000028                INTEGER SEGMNT                                               SEGFIL
000029                REAL ICROP                                                   SETT
000030                DIMENSION XNAM(2),XNAM1(2),XNAM2(2),ITEM1(8),TEMP(10),XNAM3(2)  INIT
000031                DIMENSION ITEM(2)                                            INIT
000032                DATA XNAM1/4HSEGM,4HENT /,XNAM2/4HSWAT,4HH RE/,XNAM3/4HSEG ,4HREFE  INIT
000033               1/,NCROP/3/                                                   INIT
000034                IER = 0                                                      INIT
000035                DO 5 I=1,366                                                 INIT
000036                ICLDCV(I) = 0                                                INIT
000037              5 CONTINUE                                                     INIT
000038                CALL READTF(0)                                               INIT
000039                CALL READSW(0)                                               INIT
000040                REWIND ISEGF                                                 INIT
000041                REWIND ISWR                                                  INIT
000042                READ(ISEGF)(XNAM(I),I=1,2),ITEMP                             INIT
000043                IF(XNAM(1) .NE. XNAM1(1) .OR. XNAM(2) .NE. XNAM1(2))GO TO 20  INIT
000044                IF(ICSESG .NE. 0 .AND. ICSESG .NE. ITEMP)GO TO 20            INIT
000045                GO TO 25                                                     INIT
000046             20 CALL PAGLR(2)                                                INIT
000047                WRITE(KO,900)                                                INIT
000048                GO TO 500                                                    INIT
000049            900 FORMAT(1H0,94H*** ICSESG DOES NOT MATCH THE CASE NO ON THE SEGMENTINIT
000050               1 ID FILE OR THIS FILE HAS NOT BEEN MOUNTED)                  INIT
000051             25 READ(ISWR)(XNAM(I),I=1,2),ITEMP,(ITEM1(I),I=1,8),(TEMP(I),I=1,10),INIT
000052               1(SA(I),I=1,3)                                              INIT
000053                IF(XNAM(1) .NE. XNAM2(1) .OR. XNAM(2) .NE. XNAM2(2))GO TO 30  INIT
000054                IF(ICSESW .NE. 0 .AND. ICSESW .NE. ITEMP)GO TO 30            INIT
000055                GO TO 35                                                     INIT
000056             30 CALL PAGLR(2)                                                INIT
000057                WRITE(KO,901)                                                INIT
000058                GO TO 500                                                    INIT
```

```
000059          901 FORMAT(1H0,89H*** ICSFSW DOES NOT MATCH THE CASE NO ON THE SWATH IINIT
000060             1NPUT FILES OR SWATH FILES NOT MOUNTED)                          INIT
000061           35 IF(LVEH(1) .EQ. 0)GO TO 40                                      INIT
000062              IF(LVEH(2) .EQ. 0)GO TO 45                                      INIT
000063              IF(LVEH(1) .EQ. LVEH(2))GO TO 40                                INIT
000064              IF(LVEH(2) .GT. LVEH(1))GO TO 45                                INIT
000065              ITEM(1) = LVEH(2)                                               INIT
000066              ITEM(2) = LVEH(1)                                               INIT
000067              LVEH(1) = ITEM(1)                                               INIT
000068              LVEH(2) = ITEM(2)                                               INIT
000069              IF(NVEH .GT. 1)GO TO 45                                         INIT
000070           40 CALL PAGER(2)                                                   INIT
000071              WRITE(KO,902)                                                   INIT
000072          902 FORMAT(1H0,70H*** LVEH LIST IS NOT COMPATIBLE WITH NVEH AS SPECIFIINIT
000073             1ED OR SWATH TABLES)                                            INIT
000074              GO TO 500                                                       INIT
000075           45 XV1T = IV1TIM(3)                                                INIT
000076              CALL TEPA(XV1T     ,IV1TIM(2),IV1TIM(1),X,DAYS)                 INIT
000077              ISWVST(1) = DAYS                                                INIT
000078              XV2T = IV2TIM(3)                                                INIT
000079              CALL TEPA(XV2T,IV2TIM(2),IV2TIM(1),X,DAYS)                      SET1
000080              ISWVST(2) = DAYS                                                INIT
000081              STIME = ISTIME(3)                                               INIT
000082              CALL TEPA(STIME    ,ISTIME(2),ISTIME(1),X,DAYS)                 INIT
000083              ISDAY = DAYS                                                    INIT
000084              IF(IV2TIM(1) .EQ. 0)GO TO 50                                              *NEW
000085              IF(ISWVST(1) .LT. ISWVST(2))GO TO 50                            INIT
000086              ISWDST = ISWVST(2)                                              INIT
000087              GO TO 55                                                        INIT
000088           50 ISWDST = ISWVST(1)                                              INIT
000089           55 IF(ISDAY .GE. ISWDST)GO TO 60                                   INIT
000090              CALL PAGER(2)                                                   INIT
000091              WRITE(KO,903)                                                   INIT
000092          903 FORMAT(1H0,72H*** ISTIME IS LESS THAN THE EARLIEST VEHICLE START TINIT
000093             1IME IV1TIM OR IV2TIM)                                          INIT
000094              GO TO 500                                                       INIT
000095           60 ISLATS = ISWDST + ISUSTR*30                                     INIT
000096              IPUNE = ISDAY + NDAYS - 1                                       INIT
000097              ISWATE = ISWDST + NODAY - 1                                     SET1
000098              IF(ISWATE .LT. IRUNE)IRUNE = ISWATE                             INIT
000099              INLATE = ISWDST + 426                                           INIT
000100              IF(INLATE .GT. IRUNE)INLATE = IPUNE                             INIT
000101              REWIND IREF                                                     INIT
000102              IACCT = 75*NVEH
000103              IWDC = IACCT*4 - 2
000104              IFILL=0                                                         SET10
000105              WRITE(IREF)(XLAMS(I),I=1,2),ICASE,ISDAY,NDAYS,NCROP,ICSESW,ICSESGINIT
000106             1,NVEH,(IFILL,I=1,IWDC)
000107              GO TO 505                                                       INIT
000108          500 IER = 1                                                        INIT
000109          505 RETURN                                                         INIT
000110              END                                                            INIT
```

```
* LT 10.1.751008, 62582 .11

      UBUF
000001  FS 8    SMATH/DATA,BL=1000
000002  FS 9    SMATHR/DATA,BL=400
000003  FS 1    SLGID/FILE,BL=260
000004  FS 10   SEGRFF/DATA,BL=1300
000005  FS 11   WEATHR/DATA,BL=240
000006
```

```
> .LT LFPA,1,751001, 67872      , 1

000001          SUBROUTINE LFPA(FLDA,LMO,LYR,ALFGM,DAYS)                    LFPA
000002      C                                                               LFPA
000003      C                                                               LFPA
000004      C  GIVEN  DAY,MONTH,YEAR - SUBR. LFPA RETURNS THE RIGHT ASCENSION OF  LFPA
000005      C  GREENWICH AT MIDNIGHT OF A GIVEN DAY                         LFPA
000006      C                                                               LFPA
000007      C     REFERENCE EPOCH IS 0 HOUR 1JAN1950                        LFPA
000008          DATA RADIAN /57.29578/                                      LFPA
000009      C                                                               LFPA
000010      C     COMPUTE DAYS IN FULL YEARS FROM EPOCH TO LYR              LFPA
000011      C     DAYS TO 1JAN1963 IS 4748.                                 LFPA
000012      C                                                               LFPA
000013          DAYS=4748.                                                  LFPA
000014      C     NOTE--LYR MUST BE GREATER THAN OR EQUAL TO (19)64         LFPA
000015          LASTYR=LYR-1                                                LFPA
000016          DO 20 I=63,LASTYR                                           LFPA
000017          KREMD=MOD((I-60),4)                                         LFPA
000018          IF(KREMD.GT.0) GO TO 10                                     LFPA
000019          DAYS=DAYS+366                                               LFPA
000020          GO TO 20                                                    LFPA
000021       10 DAYS=DAYS+365                                               LFPA
000022       20 CONTINUE                                                    LFPA
000023          IF(LMO-1)30,40,30                                           LFPA
000024       30 IF(LMO-2)60,50,60                                           LFPA
000025       40 DAYS=DAYS+FLDA-1.                                           LFPA
000026          GO TO 270                                                   LFPA
000027       50 DAYS=DAYS+FLDA+30.                                          LFPA
000028          GO TO 270                                                   LFPA
000029       60 KOEL=MOD((LYR-60),4)                                        LFPA
000030          IF(KOEL.GT.0) GO TO 70                                      LFPA
000031          DAYS=DAYS+59.                                               LFPA
000032          GO TO 80                                                    LFPA
000033       70 DAYS=DAYS+58.                                               LFPA
000034       80 IF(LMO-5) 270,170,90                                        LFPA
000035       90 IF(LMO-6) 180,190,100                                       LFPA
000036      100 IF(LMO-7) 200,210,110                                       LFPA
000037      110 IF(LMO-8) 220,230,120                                       LFPA
000038      120 IF(LMO-11)240,250,260                                       LFPA
000039      170 DAYS=DAYS+FLDA                                              LFPA
000040          GO TO 270                                                   LFPA
000041      180 DAYS=DAYS+FLDA+31.                                          LFPA
000042          GO TO 270                                                   LFPA
000043      190 DAYS=DAYS+FLDA+61.                                          LFPA
000044          GO TO 270                                                   LFPA
000045      200 DAYS=DAYS+FLDA+92.                                          LFPA
000046          GO TO 270                                                   LFPA
000047      210 DAYS=DAYS+FLDA+122.                                         LFPA
000048          GO TO 270                                                   LFPA
000049      220 DAYS=DAYS+FLDA+153.                                         LFPA
000050          GO TO 270                                                   LFPA
000051      230 DAYS=DAYS+FLDA+184.                                         LFPA
000052          GO TO 270                                                   LFPA
000053      240 DAYS=DAYS+FLDA+214.                                         LFPA
000054          GO TO 270                                                   LFPA
000055      250 DAYS=DAYS+FLDA+245.                                         LFPA
000056          GO TO 270                                                   LFPA
000057      260 DAYS=DAYS+FLDA+275.                                         LFPA
000058      270 CONTINUE                                                    LFPA
```

```
000059      ALFGM=AMOD(SNGL((.1000755420 3+(.985647346D01*DAYS+      LFPA
000060     1(2.9015D-13)*DAYS**2)),360.)                             LFPA
000061        ALFGM=ALFGM/RADIAN                                     LFPA
000062        RETURN                                                 LFPA
000063        END                                                    LFPA
```

```
LT PAGER,1,760331, 59945    , 1
```

```
000001                SUBROUTINE PAGER(NLINES)                                  PAGER
000002          C                                                               PAGER
000003          C                                                               PAGER
000004                COMMON /CONS/                                             CONS
000005               1RADIUS,GIS,GNUV,ECCE,PI,PI2,PI32,RADIAN,TWOPI,NAME(2),INP,KO,NLINECONS
000006               2,LINMAX,NPAGE,ISEGF,ISWTB,ISWR,IREFF,IWTHF,IWTAPE,NOCROP,IBLKF,  CONS
000007               3IPCK,NPREC,NLREC,ISWRTO,IVCT,NLATSW,IGRDN                      CONS
000008                REAL NAME                                                 CONS
000009                COMMON /FLAG/                                             FLAG
000010               1HEADER(18),ICASE,ICSESW,ICSFSG,ISTIME(3),ISDAY,NDAYS,IOPT,LVER(2),FLAG
000011               2NVER,NODAY,IV1TIM(3),IV2TIM(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARFLAG
000012               3,ICDAY,ISWVST(2),ISLATS,INLATE,IRUNE,ISWATE,NLAT,NMAX,ISUSIR,IREPTFLAG
000013                LOGICAL IREPT                                             FLAG
000014                NLINE=NLINE+NLINES                                        PAGER
000015                IF (NLINE.LT.LINMAX) GO TO 300                            PAGER
000016                IF (NPAGE.EQ.0) GO TO 200                                 PAGER
000017                IMAX=LINMAX-NLINE+NLINES                                  PAGER
000018                GO TO 10                                                  PAGER
000019             10 CONTINUE                                                  PAGER
000020          C                                                               PAGER
000021          C  SPACE TO BOTTOM OF PAGE                                      PAGER
000022          C                                                               PAGER
000023                IF (NLINES.LT.0) RETURN                                   PAGER
000024             200 CONTINUE                                                 PAGER
000025                NLINE=NLINES                                              PAGER
000026                NPAGE=NPAGE+1                                             PAGER
000027                WRITE (KO,9010)                                           PAGER
000028                WRITE (KO,9020) HEADER,ICASE,NPAGE                        PAGER
000029             300 CONTINUE                                                 PAGER
000030                RETURN                                                    PAGER
000031          C                                                               PAGER
000032          C  FORMATS                                                      PAGER
000033          C                                                               PAGER
000034             9000 FORMAT (1H )                                            PAGER
000035             9010 FORMAT (1H1)                                            PAGER
000036             9020 FORMAT(1X,18A4,18H  LPP SIMULATION  ,5HCASE ,I4,2X,4HPAGE,I4)          *NEW
000037          C                                                               PAGER  **-1
000038                END                                                       PAGER
```

```
* ELT PIMOD,I,751001,.67871     , 1

000001                FUNCTION PIMOD(A)                                    PIMOD
000002          C                                                          PIMOD
000003          C  GET POSITIVE ARGUMENT OF AN ANGLE IN RADIANS BETWEEN 0 AND 2PI   PIMOD
000004          C                                                          PIMOD
000005                B=6.28318853                                         PIMOD
000006             10 IF(A) 20,50,30                                       PIMOD
000007             20 A=A+B                                                PIMOD
000008                GO TO 10                                             PIMOD
000009             30 IF(A-B) 50,50,40                                     PIMOD
000010             40 A=A-B                                                PIMOD
000011                GO TO 30                                             PIMOD
000012             50 PIMOD=A                                              PIMOD
000013                RETURN                                               PIMOD
000014                END                                                  PIMOD
```

```
* .LT POTEN,1,760331, 59937     , 1

000001                SUBROUTINE POTEN(IND)                                          POTEN
000002          C     THIS ROUTINUE DETERMINES WHETHER THERE IS A POTENTIAL ACCESS OR POTEN
000003          C     NOT                                                            POTEN
000004                COMMON /SWATH/                                                 SWATH
000005               1TIMF1(100,3),DLONG1(100,3),DLONMN(100),DLONMX(100),TIMNOD(17), SWATH
000006               2PHINOD(17),LATNO(100),IALT(100),KVEH,KDAY,NREV,INLAT(2),ISLAT(2) SWATH
000007                COMMON /SEGFIL/                                                SEGFIL
000008               1COUNTR(2),THETSG,PHISEG,IREG,IZONE,ISTRAT,ISURST,SEGNNT,ITRPRL(6)      *NEW
000009               2,TTRSEG,ISPRN                                                  *NEW
000010                INTEGER SEGNNT                                                 SEGFIL  **-2
000011                REAL ICROP                                                     SET1
000012                COMMON /CONS/                                                  CONS
000013               1RADIUS,CIS,CONV,ECCE,PI,PI2,PI32,RADIAN,IWOP1,NAME(2),INP,KO,NLINE CONS
000014               2,LINMAX,NPAGE,ISEGE,ISHTB,ISWR,IREFE,IWTHE,IHTAPE,NOCROP,IBLKE,  CONS
000015               3IPCK,NPREC,NLPEC,ISWRTO,IVCT,NLATSW,IGRDN                      CONS
000016                REAL NAME                                                      CONS
000017                COMMON /FLAG/                                                  FLAG
000018               1HEADER(18),ICASE,ICSESW,ICSESG,ISTIME(3),ISDAY,NDAYS,IOPT,LVEH(2),FLAG
000019               2NVEH,NODAY,IV1TIM(3),IV2TIM(3),IGPID,ISWDST,JDAY,IDAY,IYEAR,IYEARDFLAG
000020               3,ICDAY,ISEVST(2),ISLATS,INLATE,IRUNE,ISWATE,NLAT,NMAX,ISUSTR,IREPTFLAG
000021                LOGICAL IEEFI                                                  FLAG
000022                COMMON /WORK/      RAND,                                       WORK
000023               1TMNODE,DEMTIN,DLNMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3), WORK
000024               2DELTA,XTIME(3),XDLONG(3),XALT,SIGMA,DDLNGL,CHECKL,CHECKR,DECK,TS, WORK
000025               3CTA,YALT,TL,CHLONG,DDLNGR,ICLDCV(366),IZDAY,IREV,IVEH,LAT,LAT1,  WORK
000026               4ICLOD,IALOCT                                                  WORK
000027                DOUBLE PRECISION RAND                                          WORK
000028                IND = 0                                                        POTEN
000029                READ(ISWR)KVEH,KDAY,X,NREV,(TIMNOD(I),I=1,17),(PHINOD(I),I=1,17) POTEN
000030                DO 10 I=1,NLAT                                                 POTEN
000031                ATNO = LATNO(I)                                                SET1
000032                RLAT1 = ATNO/RADIAN                                            SET1
000033                I1 = I                                                         POTEN
000034                I2 = I+1                                                       POTEN
000035                ATNO = LATNO(I2)                                               SET1
000036                RLAT2 = ATNO/RADIAN                                            SET1
000037                IF(THETSG .LE. RLAT1 .AND. THETSG .GE. RLAT2)GO TO 15          POTEN
000038             10 CONTINUE                                                       POTEN
000039             15 LAT = I1                                                       POTEN
000040                LAT1= I2                                                       POTEN
000041                DLNMIN = AMIN1(DLONMN(I1),DLONMN (I2))                         POTEN
000042                DLNMAX = AMAX1(DLONMX(I1),DLONMX(I2))                          POTEN
000043                DO 20 I=1,NREV                                                 POTEN
000044                ISAV = I                                                       POTEN
000045                PHIDEL = PIMOD(PHINOD(I) - PHISEG)                             POTEN
000046                IF(PHIDEL .GE. DLNMIN .AND. PHIDEL .LE. DLNMAX)GO TO 25        POTEN
000047             20 CONTINUE                                                       POTEN
000048                IND = 1                                                        POTEN
000049             22 RETURN                                                         POTEN
000050             25 IREV = ISAV                                                    POTEN
000051                JDAY = KDAY                                                    POTEN
000052                PHINDE = PHINOD(IREV)                                          POTEN
000053                TMNODE = TIMNOD(IREV)                                          POTEN
000054                GO TO 27                                                       POTEN
000055                END                                                            POTEN
```

```
> ,LT RDCD,1,760331, 59942     , 1

000001                SUBROUTINE RDCD(IE)                                          RDCD
000002          C     THIS ROUTINE READS,CHECKS AND STORES CARD DATA INTO WORKING STORAGRDCD
000003                COMMON /CONS/                                                CONS
000004               1RADIUS,CIS,CNUV,ECCE,PI,PI2,PI32,RADIAN,IWOPI,NAME(2),INP,KO,NLINECONS
000005               2,LINMAX,NPAGE,ISLCE,ISWTB,ISWR,IREFF,IWTHF,IWTAPE,NOCROP,IBLKF,  CONS
000006               3IPCK,PPREC,NLPLC,ISWRTO,IVCT,NLATSW,IGRDN                        CONS
000007                REAL NAME                                                    CONS
000008                COMMON /FLAG/                                                FLAG
000009               1HEADER(18),ICASE,ICLSLH,ICSESG,ISTIME(3),ISDAY,NDAYS,IOPT,LVEH(2),FLAG
000010               2VEH,NODAY,IVTIM(3),IVPTIM(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARDFLAG
000011               3,ICDAY,ISWVST(2),ISLATS,INLATF,IRUNE,ISWATE,NLAT,NMAX,ISUSTR,IREPTFLAG
000012                LOGICAL IREPT                                                FLAG
000013                COMMON /WORK/      RAND,                                     WORK
000014               1IMODE,DLMAIN,DLNMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3),    WORK
000015               2DELTA,XTIME(3),XDLONG(3),XALT,SIGMA,DDLNGT,CHECKL,CHECKR,DECR,IS, WORK
000016               3CTA,YALT,FL,CHLONG,DDLNGR,ICLOCV(366),IZDAY,IPEV,IVEH,LAT,LATI,  WORK
000017               4IFLOD,IALNCT                                                WORK
000018                DOUBLE PRECISION RAND                                       WORK
000019                DOUBLE PRECISION PNTEM
000020                DIMENSION LVTEM(2)                                          RDCD
000021                IE = 0                                                      RDCD
000022                READ(INP,900)(HEADER(I),I=1,18)                             RDCD
000023                READ(INP,901)ICASE,IOPT,NDTEM,ICSESW,ICSESG,(ISTIME(I),I=1,3),(   RDCD
000024               1LVTEM(I),I=1,2),IREPT,DECR,RNTEM,NMTEM,ISWTEM,NLTTEM        RDCD
000025                IF(NDTEM .NE. 0)NDAYS = NDTEM                               RDCD
000026                IF(LVTEM(1) .EQ. 0)GO TO 5                                  RDCD
000027                LVEH(1) = LVTEM(1)                                          RDCD
000028                LVEH(2) = LVTEM(2)                                          RDCD
000029              5 IF(RNTEM .NE. 0)RAND = RNTEM                                RDCD
000030                IF(NMTEM .NE. 0)IGRDN= NMTEM                                RDCD
000031                IF(ISWTEM .NE. 0)ISWRTO = ISWTEM                            RDCD
000032                IF(NLTTEM .NE. 0)NLATSW = NLTTEM                            RDCD
000033                CALL EJECT(2)                                               RDCD
000034                CALL PAGER(5)                                                        *NEW
000035                WRITE(KO,905)(HEADER(I),I=1,18)                                      *NEW
000036                WRITE(KO,906)ICASE,IOPT,NDAYS,ICSESW,ICSESG, (ISTIME(I),I=1,3),      *NEW
000037               1(LVEH(I),I=1,2),IREPT,DECR,RAND,IGRDN,ISWRTO,NLATSW                  *NEW
000038                IERCT = 0                                                   RDCD
000039                IF(NDAYS .GE. 1 .AND. NDAYS .LE. 426)GO TO 10               RDCD
000040                IERCT = IERCT + 1                                           RDCD
000041                CALL PAGER(2)                                               RDCD
000042                WRITE(KO,902)                                               RDCD
000043             10 IVCT = 0                                                    RDCD
000044                DO 15 I=1,2                                                 RDCD
000045                IF(LVEH(I) .NE. 0)IVCT = IVCT + 1                           RDCD
000046                IF(LVEH(I) .GE. 0 .AND. LVEH(I) .LE. 2)GO TO 15             RDCD
000047                GO TO 20                                                    RDCD
000048             15 CONTINUE                                                    RDCD
000049                IF(IVCT .EQ. 0)GO TO 20                                     RDCD
000050                GO TO 25                                                    RDCD
000051             20 CALL PAGER(2)                                               RDCD
000052                WRITE(KO,903)                                               RDCD
000053                IERCT = IERCT + 1                                           RDCD
000054             25 IF(IGRDN.GE. 1 .AND. IGRDN.LE. 16000)GO TO 30               RDCD
000055                CALL PAGER(2)                                               RDCD
000056                WRITE(KO,904)                                               RDCD
000057                IERCT = IERCT + 1                                           RDCD
000058             30 IF(IERCT .NE. 0)IE = 1                                      RDCD
```

```
000059          RETURN                                                                  RDCD
000060      900 FORMAT(18A4)                                                            RDCD
000061      905 FORMAT(1H ,51X,6HHEADER/1H ,18X,18A4)                                           *NEW
000062      906 FORMAT(1H0,109HICASE  IOPT  NDAYS  ICSESW  ICSESG   ISTIME   LVEH               *NEW
000063         1 TRIPT    DECR            RAND          IGRON  ISWRIO  NLATSW/1H ,1X,F       *NEW
000064         2I4,3X,I2,4X,I3,4X,I4,4X,I4,3X,3(I2,1X),1X,I2,1X,I2,3X,L1,4X,F6.2,           *NEW
000065         3X,D19.12,2X,I5,3X,I4,5X,I3)                                                 *NEW
000066      901 FORMAT(I4,I1,I3,2I4,3I2,2I1,L1,F6.2,D12.0,I5,I4,I3)                 SFT1
000067      902 FORMAT(1H0,34H*** NDAYS IS NOT BETWEEN 1 AND 426)                   RDCD
000068      903 FORMAT(1H0,83H*** IVEH HAS AN ENTRY NOT BETWEEN 0 AND 2 OR DOES NOT         *NEW
000069         1T HAVE AT LEAST 1 NONZERO ENTRY)                                    RDCD    **-1
000070      904 FORMAT(1H0,51H*** IGRON DOES NOT HAVE A VALUE BETWEEN 1 AND 16000)SFT1
000071          END                                                                 RDCD
```

```
* :LT RDM1A,1,751001, 67867     , 1
000001                SUBROUTINE RDM1A(FL,U)                                    RDM1A
000002          C     RANDOM NUMBER GENERATOR FOR ALL COMPUTERS                 RDM1A
000003                DOUBLE PRECISION C1,C2,R1,R2,T,FL,TW035,ONE,ZERO,XMOD,YMOD RDM1A
000004                DATA C1,C2,ONE,ZERO/30517493?6.D1,84365.D0,1.D0,0.D0/,     RDM1A
000005               1     TW035/34359?36368.D0/                                RDM1A
000006                XMOD(T) = DMOD(T,TW035)                                   RDM1A
000007                YMOD(T) = DMOD(T,262144.D0)                               RDM1A
000008                T = FL                                                    RDM1A
000009                IF(T .LE. ZERO) T = ONE                                   RDM1A
000010                R2 = YMOD(T)                                              RDM1A
000011                R1 = T - R2                                               RDM1A
000012                T = XMOD(XMOD(C1*R2+C2*R1) + C2*R2)                       RDM1A
000013                U = T/TW035                                               RDM1A
000014                FL = T                                                    RDM1A
000015                RETURN                                                    RDM1A
000016                END                                                       RDM1A
```

```
* ELT RDWETR,1,751031, 57161    , 1

000001                SUBROUTINE RDWETR(IRC)                                        RDWETR
000002          C     THIS ROUTINE READS ONE PHYSICAL RECORD FROM NASA WEATHER TAPE  RDWETR
000003                COMMON /CONS/                                                 CONS
000004               1RADIUS,CIS,CNUV,ECCF,PI,PI2,PI32,RADIAN,IWUPI,NAME(2),INP,KO,NLINECONS
000005               2,LINMAX,NPAGE,ISEGF,ISWTB,ISWR,IREEF,IWTHF,IWTAPE,NOCROP,IBLKF,  CONS
000006               3IPCK,NPREC,NLREC,ISWRTO,IVCT,NLATSH,IGRDN                        CONS
000007                REAL NAME                                                     CONS
000008                COMMON /FLAG/                                                 FLAG
000009               1HEADER(18),ICASE,ICSESW,ICSESG,ISTIME(3),ISDAY,NDAYS,IOPT,LVEH(2),FLAG
000010               2AVER,NODAY,IV1ITR(3),IV2ITH(3),IGRID,ISWOST,JDAY,IDAY,IYEAR,IYEARPFLAG
000011               3,ILDAY,ISWYST(2),ISLATS,NLATE,IRUNF,ISWATE,NLAT,NMAX,ISUSTR,IREPTFLAG
000012                LOGICAL IREPT                                                 FLAG
000013                COMMON /WEATDT/                                               WEATDT
000014               1IA(500),IB(50),IGROPT,IPERWT(9,12)                            WEATDT
000015                DATA IC1/30/,IC2/50/                                          RDWETR
000016                IF(IRC.NE. 1)GO TO 10                                         RDWETR
000017                CALL ETPAK(ITAPE,2,500,IA,NSTAT,22)                                  *NEW
000018             5  IF(NSTAT+1)500,5,10                                                 *NEW
000019            10  I = (IRC-1) * IC2 + 1                                               **-?
000020                DO 20 IC=1,IC2
000021                IB(IC) = IA(I + IC - 1)
000022            20  CONTINUE
000023                DECODE(900,IB(1)) IGROPT
000024                DECODE(902,IB(7)) ((IPERWT(I,J),I=1,9),J=1,6)
000025                DECODE(902,IB(26)) ((IPERWT(I,J),I=1,9),J=7,12)
000026                IRC = IRC + 1                                                 RDWETR
000027                IF(IRC .GT. 10)IRC = 1                                        RDWETR
000028                RETURN                                                        RDWETR
000029            500 CALL PAGER(2)                                                 RDWETR
000030                WRITE(KO,901)                                                 RDWETR
000031            515 STOP                                                          RDWETR
000032            900 FORMAT(I5)                                                    RDWETR
000033            901 FORMAT(1H0,52H*** END OF FILE OCCURRED ON WEATHER TAPE PREMATURELYRDWETR
000034               1)                                                             RDWETR
000035            902 FORMAT(6(I3,8I2))                                             RDWETR
000036                END                                                           RDWETR
```

```
@ ELT READSW,1,760331, 59944   , 1
      SUBROUTINE READSW(IND)                                            READSW
C     THIS ROUTINE READS A SWATH TABLE RECORD GIVEN JDAY AND IVEH OR    READSW
C     READS THE SEARCH RECORD
      COMMON /CONS/                                                     CONS
     1RADIUS,CIS,CNU,,ECCE,PI,PI2,PI32,RADIAN,TWOPI,NAME(2),INP,KO,NLINECONS
     2,LINMAX,NPAGE,ISEGE,ISWTB,ISHR,IREFF,IWTMF,INTAPE,NOCRUP,IBLKF,   CONS
     3IPCK,NPREC,NLREC,ISWRTU,IVCT,NLATSW,IGRDN                         CONS
      REAL NAME                                                         CONS
      COMMON /DTRAC/                                                    DTRAC
     1IA(800),IP(230),IPC(46)                                           DTRAC
      COMMON /WORK/        RAND,                                        WORK
     1TMNODE,DLNMIN,DLNMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3),    WORK
     2DELTA,XTIME(3),XDLONG(3),XALT,SIGMA,DDLNGL,CHECKL,CHECKR,DECR,IS, WORK
     3CTA,YALT,EL,CHLFNG,DDLNGR,ICLDCV(366),IZDAY,IREV,IVEH,LAT,LAT1,   WORK
     4ICLOD,IACPCT                                                      WORK
      DOUBLE PRECISION RAND                                             WORK
      COMMON /FLAG/                                                     FLAG
     1HEADER(18),ICASE,ICSESW,ICSFSG,ISTIME(3),ISDAY,NDAYS,IUPT,LVEH(2),FLAG
     2NVEP,NODAY,IV1TIM(3),IV2TIM(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARBFLAG
     3,ILDAY,ISRVST(2),ISLATS,INLATF,IRUNF,ISRATE,NLAT,NMAX,ISUSTR,IREPTFLAG
      LOGICAL IREPT                                                     FLAG
      COMMON /SWATH/                                                    SWATH
     1TIMEI(100,3),DLONG1(100,3),DLONMN(100),DLONMX(100),TIMNOD(17),    SWATH
     2PHINOD(17),LATNO(100),IALT(100),KVEH,KDAY,NREV,INLAT(2),ISLAT(2)  SWATH
      DIMENSION XNAME(2),YNAME(2)                                       READSW
     1,A(800)                                                           READSW
      EQUIVALENCE (XNAME(1),IA(1)),(IA(1),A(1))                         READSW
      DATA YNAME/4HSWAT,4HH TB/                                         READSW
      NOWDS = 8*NLATSW                                                  READSW
      DO 5 I=1,800                                                      READSW
      A(I) = 0.0                                                        READSW
    5 CONTINUE                                                          READSW
      IF(IND  .EQ. 0)GO TO 10                                           READSW
      IF(IND  .EQ. 1)GO TO 100                                          READSW
      IF(IND  .EQ. 2)GO TO 200                                          READSW
    7 RETURN                                                            READSW
10    DEFINE FILE 8(1101,800,U,IDUM)
      READ(ISWTB'1,ERR=300) (IA(I),I=1,NOWDS)
      IF(XNAME(1) .EQ. YNAME(1) .AND. XNAME(2) .EQ. YNAME(2))GO TO 20   READSW
   15 CALL PAGER(2)                                                     READSW
      WRITE(KO,900)                                                     READSW
   17 STOP                                                              READSW
  900 FORMAT(1H0, 88H***ICSFSW DOES NOT MATCH THE CASE NO ON THE SWATH IREADSW
     1NPUT FILES OR SWATH FILES NOT MOUNTED)                            READSW
   20 IF(ICSFSW .NE. IA(3))GO TO 15                                     READSW
      IF(IVCT .LE. IA(4))GO TO 25                                       READSW
      CALL PAGER(2)                                                     READSW
      WRITE(KO,901)                                                     READSW
  901 FORMAT(1H0,74H*** IVEH LIST IS NOT COMPATIBLE WITH NVEH AS SPECIFI    *NEW
     1ED ON THE SWATH TABLES)                                           READSW  **-1
      GO TO 17                                                          READSW
   25 NODAY = IA(5)                                                     READSW
      NVEH  = IA(4)                                                     READSW
      NLAT = IA(6)                                                      READSW
      INLAT(1) = IA(7)                                                  READSW
      INLAT(2) = IA(8)                                                  READSW
      ISLAT(1) = IA(9)                                                  READSW
      ISLAT(2) = IA(10)                                                 READSW
```

```
000059          DO 30 I=1,3                                                   READASW
000060          IV1TIM(I) = IA(10+I)                                          READASW
000061          IV2TIM(I) = IA(13+I)                                          READASW
000062       30 CONTINUE                                                      READASW
000063          ISOSTR = IA(17)                                               READASW
000064          GO TO 7                                                       READASW
000065      100 IRDS = ISWATE - ISWVST(1) + 1.                                SET1
000066          IREC = (IRDS +1)*(IVEH-1) + JDAY + 1                          SET1
000067          READ(ISWTR'IREC,ERR=500) (IA(I),I=1,NOWDS)
000068          IP1 = NLAT                                                    READASW
000069          IP2 = 2*NLAT                                                  READASW
000070          IP3 = 3*NLAT                                                  READASW
000071          DO 110 I=1,NLAT                                               READASW
000072          LAIND(I) = IA(I)                                              READASW
000073          IALT(I) = IA(I+IP1)                                           READASW
000074      110 CONTINUE                                                      READASW
000075          DO 120 J=1,3                                                  READASW
000076          IP1 = (J-1)*NLAT                                              SET1
000077          DO 115 I=1,NLAT                                               READASW
000078          TIME1(I,J) = A(I+IP2+IPT)                                     SET1
000079          DLONG1(I,J) = A(I+IP3+IPT)                                    SET1
000080      115 CONTINUE                                                      READASW
000081      120 CONTINUE                                                      READASW
000082          GO TO 7                                                       READASW
000083      200 IRDS1 = ISWATE - ISWVST(1) + 1                                SET1
000084          IRDS = ISWATE - ISWVST(IVEH) + 1                              SET1
000085          IREC = (IRDS1   )*(IVEH-1) + IRDS + IVEH +1                   SET1
000086          READ(ISWTR'IREC,ERR=500) (IA(I),I=1,NOWDS)
000087          IP1 = NLAT                                                    READASW
000088          IP2 = 2*NLAT                                                  READASW
000089          DO 210 I=1,NLAT                                               READASW
000090          LAIND(I) = IA(I)                                              READASW
000091          DLONMN(I) = A(I+IP1)                                          READASW
000092          DLONMX(I) = A(I+IP2)                                          READASW
000093      210 CONTINUE                                                      READASW
000094          GO TO 7                                                       READASW
000095      500 WRITE(KO,902)
000096      902 FORMAT(1H0,85H *** AN IRRECOVERABLE I/O ERROR HAS OCCURRED ON READ
000097         1ING A RECORD FROM THE SWATH TABLE  /10X,31H THE JOB IS BEING ABAND
000098         2ONED ***)
000099          GO TO 17
000100          END                                                           READASW
```

```
* FLT SAGE,1,760401, 64552      , 1

000001          C      PROGRAM MAIN(ITEST=401,OTEST=401,SWATH=1500,SWATHR=500,SEGID=1300,MAIN
000002          C     1WEATAP=1004,WEATHR=400,SEGREF=2400,TAPE5=ITEST,TAPE6=OTEST,          SET1
000003          C     1TAPE8=SWATH,                                                         MAIN
000004          C     2     TAPE9=SWATHR,TAPE10=SEGREF,TAPE1=SEGID,TAPE33=WEATAP,TAPE11=WE MAIN
000005          C     3ATHR)                                                                MAIN
000006                COMMON /FLAG/                                                         FLAG
000007               1HEADER(18),ICASE,ICSESW,ICSESG,ISTIME(3),ISDAY,NDAYS,IOPT,LVEH(2),FLAG
000008               2NVEH,NODAY,IVITIM(3),IV2TIM(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARPFLAG
000009               3,ICDAY,ISWVST(2),ISLATS,INLATF,IRUNF,ISWATE,NLAT,NMAX,ISUSIR,IREPTFLAG
000010                LOGICAL IREPT                                                        FLAG
000011                COMMON /CONS/                                                        CONS
000012               1RADIUS,GIS,GNUV,ECCF,PI,PI2,PI32,RADIAN,TWOPI,NAME(2),INP,NO,NLINECONS
000013               2,LINMAX,NPAGE,ISEGF,ISWTU,ISWR,IREFF,IWTHF,IWTAPE,NOCROP,IBLKF,     CONS
000014               3IPCK,IPPLC,NLREC,ISWRTO,IVCT,NLATSW,IGRDN                           CONS
000015                REAL NAME                                                           CONS
000016                COMMON /SEGFIL/                                                     SEGFIL
000017               1COUNTR(2),THETSG,PHISEG,IREG,IZONF,ISTRAT,ISUBST,SEGMNT,ITRPRL(6)          *NEW
000018               2,ITRSEG,ISPRW                                                              *NEW
000019                INTEGER SEGMNT                                                     SEGFIL **-2
000020                REAL ICLOD                                                         SET1
000021                COMMON /ACQUIS/                                                    ACQUIS
000022               1CTAT(150),ELT(150),TST(150),YALT(150),IZDAY1(150),IREVT(150),      ACQUIS
000023               2ICLODT(150),IVEHT(150)                                            ACQUIS
000024                COMMON /WORK/      RAND,                                           WORK
000025               1IPNODE,ELNMIN,DLNMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3),    WORK
000026               2DELTA,XTIME(3),XDLONG(3),XALT,SIGMA,DDLNGL,CHECKL,CHECKP,DECR,TS,  WORK
000027               3CTA,YALT,EL,CHLONG,DDLNGK,ICLODV(366),IZDAY,IREV,IVEH,LAT,LAT1,    WORK
000028               4ICLOD,IACCT                                                       WORK
000029                DOUBLE PRECISION RAND                                             WORK
000030                DATA COUNT/4HZZZZ/                                                MAIN     **-11
000031                ISUCC = 0                                                         MAIN
000032                IRLECT = 0                                                        MAIN
000033                CALL RDCOM(IPE)                                                   MAIN
000034                IF(ITE .NE. 0)GO TO 500                                           MAIN
000035                IF(IOPT .NE. 0)GO TO 100                                          MAIN
000036              5 CALL RNIT(IPE)                                                    MAIN
000037                IF(IER .NE. 0)GO TO 500                                           MAIN
000038                IACCT = MSENVEH                                                           *NEW
000039              10 READ(ISEGF)COUNTR(1),IREG,IZONE,ISTRAT,ISUBST,SEGMNT,ITRSEG,             *NEW
000040                1(ITRPRL(I),I=1,6),THETSG,PHISEG,IGRID,ISPRW                                *NEW
000041                IF(COUNTR(1) .EQ. COUNT)GO TO 50                                  SET1     **-3
000042                ISUCC = ISUCC + 1                                                 MAIN
000043                CALL SEGPRO(ISEG)                                                 MAIN
000044                WRITE(IREFF)COUNTR(1),IREG,IZONE,ISTRAT,ISUBST,SEGMNT,ITRSEG,     SET1     **-3
000045               1ISPRW,(IZDAY(I),ELT(I),ICLODT(I),TST(I),I=1,IACCT)                         *NEW
000046                IRLECT = IRLECT + 1                                               MAIN     **-3
000047                IF(ISEG .NE. 0)GO TO 10                                           MAIN
000048                IF(IREPT)CALL SEGREP                                              MAIN
000049                GO TO 10                                                          MAIN
000050             50 CALL EJECT(11)                                                    MAIN
000051                WRITE(NO,900)ISTIME(2),ISTIME(3),ISTIME(1),ICSESW,IRLECT,ICSESG,  MAIN
000052               1ISUCC                                                            MAIN
000053            900 FORMAT(1H0,65H*** SAGE HAS SUCCESSFULLY WRITTEN THE SEGMENT REFERE MAIN
000054               1NCE DATA FILE//1H ,32H    SEGMENT REFERENCE DATA FILE*/1H ,20H     MAIN
000055               2  REFERENCE DTE ,I2,1H/,I2,1H/,I2/1H ,20H       SWATH FILES CASE USMAIN
000056               3ED ,I4 /1H ,28H       NO OF RECORDS WRITTEN ,I4/1H ,20H    SEGMENT MAIN
000057               4ID FILE*/1H ,32H       SEGMENT ID FILE CASE USED ,I4/1H ,25H       MAIN
000058               5O OF RECORDS READ ,I4)                                            MAIN
```

```
000059      CALL WEATF(2)                                                    MAIN
000060      CALL READSW(3)                                                   MAIN
000061      WRITE(IREFF)COUNT,IREG,IZONE,ISTRAT,ISUBST,SEGMNT,ITRSEG,ISPRW,          *NEW
000062     1(IZDAYT(I),FLT(I),ICLODT(I),TSI(I),I=1,IACCT)                    *NEW
000063      END FILE IREFF                                                   MAIN      **-3
000064      REWIND IREFF                                                     MAIN      **-1
000065      REWIND ISEGF                                                     MAIN
000066      REWIND ISWR                                                      MAIN
000067      GO TO 510                                                        MAIN
000068  100 CALL WEATH                                                       MAIN
000069      CALL EFFECT(6)                                                   MAIN
000070      WRITE(6,901)IGRDN,RAND                                           MAIN
000071  901 FORMAT('1H0,39H*** WEATHER DATA FILE UTILITY COMPLETED///1H ,22H  MAIN
     1WEATHER DATA FILE=,I4/1H ,24H        NO OF GRID POINTS =,I5/1H ,27H  MAIN
     2LAST RANDOM NO USED =,D15.8,I2)                                            *NEW
000072      IF(ITOP1 .EQ. 1)GO TO 510                                        MAIN      **-11
000073      GO TO 5                                                          MAIN
000074  500 CALL EFFECT(2)                                                   MAIN
000075      WRITE(6,902)                                                     MAIN
000076  902 FORMAT('1H0,61H*** DUE TO A FATAL ERROR SAGE DID NOT COMPLETE IT'S PMAIN
     1ROCESSING')                                                         MAIN
000077  510 STOP                                                             MAIN
000078      END                                                              MAIN
```

```
SEGPRO,1,760331, 59941    , 1

000001            SUBROUTINE SEGPRO(ISEG)                                  SEGPRO
000002      C     THIS ROUTINE CONTROLS THE PROCESSING FOR ONE SEGMENT     SEGPRO
000003            COMMON /FLAG/                                            FLAG
000004           1HEADER(18),ICASE,ICSESW,ICSESG,ISTIME(3),ISDAY,NDAYS,IOPT,LVEH(2),FLAG
000005           2NVEH,ICDAY,IV1TIM(3),IV2TIM(3),IGRID,ISWOSI,JDAY,IDAY,IYEAR,IYEARF,FLAG
000006           3,ICDAY,ISWYSI(2),ISLATS,INLAIL,IRUNE,ISWATE,NLAT,NMAX,ISUSIR,IRTPTH,FLAG
000007            LOGICAL IRLPT                                            FLAG
000008            COMMON /CONS/                                            CONS
000009           1RADIUS,CIS,CNNV,ECCF,PI,PI2,PI5?,RADIAN,IHOPI,NAMF(2),INP,IKO,NLINE,CONS
000010           2,1INMAX,NPAGE,ISEOF,ISWTB,ISKP,IREFF,IWTHF,IWTAPE,NOCRUP,IBLKF,  CONS
000011           3IPCK,IPREC,NLREC,ISWRTO,IVCT,NLAISW,IGRON                       CONS
000012            REAL NAME                                                CONS
000013            COMMON /WORK/       RAND,                                WORK
000014           1IMNOSE,DLNMIN,DLNMAX,DLONG(3,2),TIME(3,2),AII(2),PHINDE,SA(3),  WORK
000015           2DELTA,XTIPL(3),XDLONG(3),XALT,STGMA,CDLTGL,CHECKL,CHECKR,DECR,IS,  WORK
000016           3CTA,YALT,FL,CHLONG,DULNGR,ICLOCV(566),IZDAY,IREV,IVEH,LAI,LAI1,  WORK
000017           4ICLOD,IACOCT                                                WORK
000018            DOUBLE PRECISION RAND                                    WORK
000019            COMMON /SPATH/                                           SPATH
000020           1TIME1(100,3),DLONG1(100,3),DLONMN(100),DLONMX(100),TIMMOD(17),  SPATH
000021           2PHIMOD(17),LATOO(100),INLT(100),NVEH,KDAY,NREV,INLAT(2),ISLAT(2)  SPATH
000022            COMMON /SEGFIL/                                          SEGFIL
000023           1COUNTP(2),THETSG,PHISEG,IREG,IZONF,ISTRAT,ISUBST,SEGMNT,ITRPRL(6)   *NEW
000024           2,ITPSEG,ISPRX                                                       *NEW
000025            INTEGER SEGMNT                                           SEGFIL  **-2
000026            REAL TCRUP                                               SEGFIL
000027            COMMON /ACQUIS/                                          ACQUIS
000028           1CTAT(150),ELT(150),TST(150),YALTT(150),IZDAYT(150),IREVT(150),  ACQUIS
000029           2ICLODT(150),IVEHT(150)                                  ACQUIS
000030            DIMENSION IDUM(3)                                        SEGPRO
000031            ISEG = 0                                                 SEGPRO
000032            IACOCT = 0                                               SEGPRO
000033            IV = 1                                                   SEGPRO
000034            TEMP = INLAT(2)/RADIAN                                   SEGPRO
000035            TEMP1 =-ISLAT( 2  )/RADIAN                               SEGPRO
000036            IF(THETSG .GT. TEMP1)GO TO 500                           SEGPRO
000037            IF(THETSG .LT. TEMP1)GO TO 500                           SEGPRO
000038            TEMP2 = INLAT(1)/RADIAN                                  SEGPRO
000039            TEMP3 =-ISLAT(1)/RADIAN                                  SEGPRO
000040            IF(THETSG .GT. TEMP3 .AND.THETSG .LT.TEMP2)GO TO 500     SEGPRO
000041            IF(THETSG .GE. TEMP2 .AND. THETSG .LE. TEMP1)GO TO 5     SEGPRO
000042            LAISG = 1                                                SEGPRO
000043            GO TO 10                                                 SEGPRO
000044          5 LAISG = 0                                                SEGPRO
000045         10 CALL RELIEF(1)                                           SEGPRO
000046            REWIND ISWR                                              SEGPRO
000047            READ(ISWR)                                               SEGPRO
000048            CALL UNPKPK(2)                                           SEGPRO
000049            DO 12 I=1,150                                            SEGPRO
000050            IZDAYT(I) = 0                                            SEGPRO
000051            IREVT(I) = 0                                             SEGPRO
000052            CTAT(I) = 0.                                             SEGPRO
000053            ELT(I) = 0.                                              SEGPRO
000054            ICLODT(I) = 0                                            SEGPRO
000055            TST(I) = 0.                                              SEGPRO
000056            YALTT(I) = 0.                                            SEGPRO
000057            IVEHT(I) = 0                                             SEGPRO
000058         12 CONTINUE                                                 SEGPRO
```

```
000059          IF(NVEH .LT. 2)GO TO 15                                 SEGPRO
000060          IF(IVEH(1) .NE. 2)GO TO 15                              SEGPRO
000061          ITEM = ISWATE - ISWVST(1) + 1                           SEG1
000062          DO 14 I=1,ITEM                                          SEG1
000063          READ(1)SWK0                                             SEGPRO
000064       10 CONTINUE                                                SEGPRO
000065          ITE = 0                                                 SEG2
000066       15 IVEH = IVEH(IV)                                         SEGPRO
000067          ISEGNR = ISWVST(IVEH)                                   SEG2
000068          CALL REWDS(2)                                           SEGPRO
000069          CALL DAY(IST1ME,IDAY)                                   SEGPRO
000070          IYEAR = MOD(IST1ME(1),4)                                SEGPRO
000071          IYEARB = IST1ME(1) + 1900 - 1                           SEGPRO
000072          ICDAY = ISDAY                                           SEGPRO
000073          IF(ISDAY .GT. ISEGNR)GO TO 19                           SEG2
000074          GO TO 18                                                SEG2
000075       19 ITE = ISDAY - ISEGNR                                    SEG2
000076          DO 17 I=1,ITE                                           SEG2
000077          READ(1)SWK0                                             SEG2
000078       17 CONTINUE                                                SEG2
000079       18 DO 150 IO=1,NDAYS                                       SEG2
000080          IF(ICDAY .LT. ISEGPR)GO TO 150                          SEG1
000081          IF(ICDAY .GT. IRUNE)GO TO 160                           SEG1
000082          IF(ICDAY .GE. ISWDST .AND. ICDAY .LT. ISLATS)LATYP = 0  SEGPRO
000083          IF(ICDAY .GE. ISLATS .AND. ICDAY .LE. INLATE)LATYP = 2  SEGPRO
000084          IF(ICDAY .GT. INLATE .AND. ICDAY .GE. ISLATS)LATYP = 1  SEGPRO
000085          IF(LATYP .EQ. 2)GO TO 20                                SEGPRO
000086          IF(LATYP .EQ. LATSG)GO TO 20                            SEGPRO
000087          GO TO 140                                               SEG1
000088       20 CONTINUE                                                SEG2
000089       25 CALL POTEN(IPO)                                         SEGPRO
000090          IF(IPO .NE. 0)GO TO 130                                 SEGPRO
000091          CALL ACCVER(IAC)                                        SEGPRO
000092          IF(IAC .NE. 0)GO TO 130                                 SEGPRO
000093          ICLOD = ICLDCV(IDAY)                                    SEGPRO
000094          CALL IMAGE                                              SEGPRO
000095          IACOCT = IACOCT + 1                                     SEGPRO
000096          IF(IACOCT .GT. 150)GO TO 510                            SEGPRO
000097          I = IACOCT                                              SEGPRO
000098          IZDAYT(I) = IZDAY                                       SEGPRO
000099          IREVT(I) = IREV                                         SEGPRO
000100          CTAT(I) = CTA                                           SEGPRO
000101          ELT(I) = EL                                             SEGPRO
000102          ICLODT(I) = ICLOD                                       SEGPRO
000103          TST(I) = TS                                             SEGPRO
000104          YALTT(I) = YALT                                         SEGPRO
000105          IVEHT(I) = IVEH                                         SEGPRO
000106      130 IDAY = IDAY + 1                                         SEGPRO
000107          ICDAY = ICDAY + 1                                       SEGPRO
000108          IF(MOD(IYEARB+IYEAR,4) .EQ. 0)GO TO 50                  SEGPRO
000109          IF(IDAY .LT. 366)GO TO 60                               SEGPRO
000110          GO TO 55                                                SEGPRO
000111       50 CONTINUE                                                SEGPRO
000112          IF(IDAY .LT. 367)GO TO 60                               SEGPRO
000113          IYEAR = MOD(IYEAR+1,4)                                  SEGPRO
000114          IDAY = IDAY - 366                                       SEGPRO
000115          GO TO 60                                                SEGPRO
000116       55 CONTINUE                                                SEGPRO
000117          IYEAR = MOD(IYEAR+1,4)                                  SEGPRO
000118          IDAY = IDAY - 365                                       SEGPRO
```

```
   60 GO TO 150                                                          SEGPRO
  140 READ(ISWKD)                                                        SEGPRO
      GO TO 150                                                          SEGPRO
  150 CONTINUE                                                           SEGPRO
  160 IW = IW + 1                                                        SEGPRO
      IF(LWFH(IW) .EQ. 0 .OR. IW .GT. 2)GO TO 450                        SEGPRO
      ITEM1 = INDAY - (NDAYS*ITE)                                        SEGPRO
      IF(ITEM1 .LT. 0)GO TO 15                                           SEGPRO
      DO 200 I=1,ITEM1                                                   SEGPRO
      READ(ISWKD)                                                        SEGPRO
  200 CONTINUE                                                           SEGPRO
      GO TO 15                                                           SEGPRO
  450 IF(IACQCT .NE. 0)GO TO 520                                         SEGPRO
      CALL PAGER(2)                                                      SEGPRO
      WRITE(KO,900)SEGMNT                                                SEGPRO
  900 FORMAT(1H0,26H*** IN PROCESSING SEGMENT ,I4,32H NO VALID ACQUISITISEGPRO   #NEW
     1ON OCCURRED. )                                                     SEGPRO   **-1
      GO TO 515                                                          SEGPRO
  500 CALL PAGER(2)                                                      SEGPRO
      CALL DEGMID(THETSG,IDUM)                                           SEGPRO
      WRITE(KO,901)SEGMNT,(IDUM(I),I=1,3)                                SEGPRO
  901 FORMAT(1H0,12H*** SEGMENT ,I4,10H LATITUDE ,I3,2(1X,I2),53H IS NOTSEGPRO
     1 IN THE LATITUDE BAND SPECIFIED BY LATNO TABLE)                    SEGPRO
      GO TO 515                                                          SEGPRO
  510 CALL PAGER(2)                                                      SEGPRO
      WRITE(KO,902)SEGMNT                                                SEGPRO
  902 FORMAT(1H0, 26H*** IN PROCESSING SEGMENT ,I4,75H MORE THAN 150 ACQSEGPRO
     1UISTIONS OCCURRED. NO MORE ACQUISITIONS WILL BE PROCESSED)         SEGPRO
      GO TO 520                                                          SEGPRO
  515 ISEG = 1                                                           SEGPRO
  520 RETURN                                                             SEGPRO
      END                                                                SEGPRO
```

```
LT SEGREP,1,760331, 59939    , 1

000001                SUBROUTINE SEGREP                                          SEGREP
000002          C     THIS ROUTINE PRODUCES THE SEGMENT ACCESS DATA REPORT FOR 1 SEGMENTSEGREP
000003                COMMON /WORK/          RAND,                               WORK
000004               1ITYROU,IDLNMIN,IDLNMAX,IDLONG(5,2),ITIME(5,2),ALT(2),IPHINDE,SA(3),  WORK
000005               2UILTA,XTIME(3),XDLONG(3),XALT,STGMA,IDDLNGL,CHECKL,CHECKP,CECK,ITS,  WORK
000006               3CTA,YALT,FL,CHLONG,IDDLNGK,ICLIDCV(366),IZDAY,IPEV,IVEH,ILAT,ILATT,  WORK
000007               4ICLOD,IACCCT                                              WORK
000008                DOUBLE PRECISION RAND                                     WORK
000009                COMMON /ACQUIS/                                           ACQUIS
000010               1CTAT(150),FLT(150),ITST(150),YALTT(150),IZDAYT(150),IREVT(150),  ACQUIS
000011               2ICLODT(150),IVEHT(150)                                   ACQUIS
000012                COMMON /SEGFIL/                                           SEGFIL
000013               1COUNTR(2),IDETSG,PHISEG,IREG,IZONE,ISTRAT,ISUBST,SEGMNT,ITRPRL(60)      *NEW
000014               2,ITESEG,ISPRA                                            *NEW
000015                INTEGER SEGMNT                                           SEGFIL  **-?
000016                REAL ICROP                                               SEGFIL
000017                COMMON /CONS/                                            CONS
000018               1RADIUS,CIS,CNUV,ECCF,PI,PI2,PI5P,RADIAN,IWOPI,NAME(2),INP,IKO,NLINE,CONS
000019               2,INMAX,NPAGE,ISEGF,ISWTH,ISWR,IREFF,INTHF,ITAPE,NOCROP,IULNF,    CONS
000020               3IPCK,NPLEC,NLREC,ISWRTO,IVCT,NLATSW,IGRDN                        CONS
000021                REAL NAME                                                CONS
000022                DIMENSION IDATE(3),IDEG(3),IDEG1(3)                      SEGREP
000023                CALL IFDECT(7)                                           SEGREP
000024                WRITE(KO,900)COUNTR(1),SEGMNT                            SEGFIL
000025                WRITE(KO,901)                                            SEGREP
000026                DO 10 I=1,IACCCT                                         SEGREP
000027                CALL PAGER(1)                                            SEGREP
000028                CALL FZYMD(IZDAY(I),IDATE(I))                            SEGREP
000029                CALL DEGDDCEL(I(I),IDEG(I))                              SEGREP
000030                CALL DLGMOD(CTAT(I),IDEG1(I))                            SEGREP
000031                CLOUD = ICLOD(I)                                         SEGREP
000032                CLOUD = CLOUD/10.0                                       SEGREP
000033               WRITE(KO,903)I,IDATE(2),IDATE(3),IDATE(1),IREV(I),IVEH(I),       SEGFIL
000034               1(IDEG(L),L=1,3)                                          SEGFIL
000035               1,(IDEG(L),L=1,3),CLOUD     ,TST(I),YALT(I)              SEGREP
000036            10 CONTINUE                                                  SEGREP
000037                RETURN                                                   SEGREP
000038           900 FORMAT(1H ,35X,36HSEGMENT ACCESS DATA REPORT  COUNTRY ,A4/1H ,49X,SEGREP
000039              1 4HSEGMENT ,I4)                                           SEGREP
000040           901 FORMAT(1H0,18X,64HACCESS  ACQUISIT                         SEGREP
000041              1  CLOUD  ACQUISI,/1H ,18X,74HCOUNT   MO/DY/YR  REV  VEH      CTA  SEGFIL
000042              2      FL     COVER  TIME      ALTITUDE)                   SEGFIL
000043           903 FORMAT(1H ,19X,I3,4X,I2,1H/,I2,1H/,I2,2X,I2,4X,I1,3X,2(I3,1X,I2,1XSEGREP
000044              1  ,I2,2X),F5.1,3X,F6.0,4X,F5.0)
000045                END                                                      SEGREP
```

```
6=LT UNPKPK,1,751001, 67866    , 1

000001                SUBROUTINE UNPKPK(IND)                                    UNPKPK
000002          C     THIS ROUTINE PACKS OR UNPACKS CLOUD COVER DATA BETWEEN ICLDCV AND UNPKPK
000003          C     IPC                                                       UNPKPK
000004                COMMON /WORK/      RAND,                                  WORK
000005               1TMNODE,DLNMIN,DLNMAX,DLONG(3,2),TIME(3,7),ALT(2),PHINDE,SA(3), WORK
000006               2DELTA,XTIME(3),XDLONG(3),XALT,SIGMA,DDLNGL,CHECKL,CHECKR,DECR,TS, WORK
000007               3CTA,YALT,EL,CHLONG,DDLNGR,ICLDCV(366),17DAY,IREV,IVEH,LAT,LAT1, WORK
000008               4ICLOD,IACOCT                                             WORK
000009                DOUBLE PRECISION RAND                                    WORK
000010                COMMON /DIRAC/                                           DIRAC
000011               1IA(800),IP(230),IPC(46)                                  DIRAC
000012                COMMON /FLAG/                                            FLAG
000013               1HEADER(18),ICASE,ICSESW,ICSESG,ISTIME(3),ISDAY,NDAYS,IUPI,LVEH(2), FLAG
000014               2NVEH,NDAY,IV1TIM(3),IV2TIM(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARB FLAG
000015               3,ICDAY,ISWVST(2),ISLATS,INLATE,INUNE,ISWATE,NLAT,NMAX,ISOSTR,IREPT FLAG
000016                LOGICAL  IREPT                                           FLAG
000017                COMMON /CONS/                                            CONS
000018               1RADIUS,CIS,CNUV,ECCE,PI,PI2,PI32,RADIAN,TWOPI,NAME(2),INP,KO,NLIN CONS
000019               2,LINMAX,LPAGE,ISLGF,ISWTU,ISWF,IREFF,IATHF,INTAPE,NOCROP,IBLKF,  CONS
000020               3IPCK,NUREC,NLREC,ISWRTO,IVCT,NLATSW,IGRDN                CONS
000021                REAL MATE                                                CONS
000022                DIMENSION MULT(8)                                        UNPKPK
000023                DATA MULT/268435456,16777216,1048576,65536,4096,256,16,1/ UNPKPK
000024                IF(IND .NE. 0)GO TO 100                                  UNPKPK
000025                I = 1                                                    UNPKPK
000026                J = 1                                                    UNPKPK
000027                DO 5 I=1,NLREC                                           UNPKPK
000028                IPC(I) = 0                                               UNPKPK
000029              5 CONTINUE                                                 UNPKPK
000030             10 DO 20 K=1,IPCK                                           UNPKPK
000031                IPC(J) = IPC(J) + ICLDCV(I)*MULT(K)                      UNPKPK
000032                I = I + 1                                                UNPKPK
000033                IF(I .GT. NMAX)GO TO 200                                 UNPKPK
000034             20 CONTINUE                                                 UNPKPK
000035                J = J + 1                                                UNPKPK
000036                GO TO 10                                                 UNPKPK
000037            100 I = 1                                                    UNPKPK
000038                J = 1                                                    UNPKPK
000039            110 ITEMP1 = IPC(J)                                          UNPKPK
000040                DO 120 K=1,IPCK                                          UNPKPK
000041                ITEMP  = ITEMP1/MULT(K)                                  UNPKPK
000042                ITEMP1 = ITEMP1 - ITEMP*MULT(K)                          UNPKPK
000043                ICLDCV(I) = ITEMP*125                                    UNPKPK
000044                I = I + 1                                                UNPKPK
000045                IF(I .GT. NMAX)GO TO 200                                 UNPKPK
000046            120 CONTINUE                                                 UNPKPK
000047                J = J + 1                                                UNPKPK
000048                GO TO 110                                                UNPKPK
000049            200 RETURN                                                   UNPKPK
000050                END                                                      UNPKPK
```

• ELT WEATF,1,751007, 50499 , 1

```
      SUBROUTINE WEATF(IND)                                             WEATF
C     THIS ROUTINE READS A LOGICAL RECORD FROM THE WEATHER FILE        WEATF
      COMMON /CONS/                                                     CONS
     1RADIUS,CPS,CONV,DEGE,PI,PI2,PI32,RADIAN,THOUR,NAME(2),INP,KO,NLINECONS
     2,IPNMAX,NPAGE,IPAGE,ISHMB,ISWR,IHRHF,IHMHF,IHTAPE,NTCRUP,IBUNF,    CONS
     3IDUK,NPHIC,IUREC,ISHRMO,IWCT,NLMISH,IORDN                         CONS
      REAL IPAT                                                         CONS
      COMMON /FLAG/                                                     FLAG
     1HEADER(16),ICASE,ICRESH,ICSESG,ISTIME(3),ISDAY,NDAYS,IOPT,LWEH(2),FLAG
     2NREP,IDWY,IWTIME(3),IWPTTH(3),IGRID,ISNDSH,UDAY,IDWY,IME,IR,IMEARDFLAG
     3,ICDAY,ISAVST(2),ISLATS,INLATF,IRUNE,ISHATE,NLAT,NMAX,ISOSTR,IROPTFLAG
      LOGICAL IREPT                                                     FLAG
      COMMON /DIRRWC/                                                   DIRRWC
     1IA(2000),IB(250),IRC(46)                                          DIRRWC
      DIMENSION IBC(2)                                                  WEATF
      DATA IBC/ 4HLAT,4HHDR /                                           WEATF
      DO 5 I=1,NPREC                                                    WEATF
      IB(I) = 0                                                         WEATF
    5 CONTINUE                                                          WEATF
      DO 10 I=1,NLREC                                                   WEATF
      IPC(I) = 0                                                        WEATF
   10 CONTINUE                                                          WEATF
      IF(IND .EQ. 0)GO TO 20                                            WEATF
      IF(IND .EQ. 1)GO TO 100                                           WEATF
   15 RETURN                                                            WEATF
20    DEFINE FILE 11(3201,230,U,IDUM)
      READ(I,THE'1,ERR=200) (IB(I),I=1,NPREC)
      IF(IB(1) .EQ. IBC(1) .AND. IB(2) .EQ. IBC(2))GO TO 30             WEATF
      CALL PAGER(2)                                                     WEATF
      WRITE(KO,900)                                                     WEATF
  900 FORMAT(1H0,46H*** THE WEATHER DATA FILE HAS NOT BEEN MOUNTED)     WEATF
25    STOP
   30 NMAX = IB(3)                                                      WEATF
      GO TO 15                                                          WEATF
  100 IREC = IGRID/IBLKF + 1                                            WEATF
      INDEX = IGRID -(IGRID/IBLKF)*IBLKF                                WEATF
      IF(INDEX .NE. 0)IREC = IREC + 1                                   WEATF
      IF(INDEX .EQ. 0)INDEX = 5                                         SE11
      READ(IWTHF'IREC,ERR=200) (IB(I),I=1,NPREC)
      I = (INDEX-1)*NLREC                                               WEATF
      DO 110 J=1,NLREC                                                  WEATF
      IPC(J) = IB(I+J)                                                  WEATF
  110 CONTINUE                                                          WEATF
      GO TO 15                                                          WEATF
  200 WRITE(KO,901)
  901 FORMAT(1H0,86H *** AN IRRECOVERABLE I/O ERROR HAS OCCURRED ON READ
     1ING A RECORD FROM THE WEATHER FILE  /10X,31H THE JOB IS BEING ABAN
     2DONED ***)
      GO TO 25
      END                                                               WEATF
```

ORIGINAL PAGE IS OF POOR QUALITY

```
      * ALT WEATH,1,760505, 58232     , 1
000001                SUBROUTINE WEATH                                          WEATH
000002          C     THIS ROUTINE CONTROLS THE GENERATION OF THE WEATHER FILE  WEATH
000003                COMMON /WEATDT/                                           WEATDT
000004               1IA(500),IB(50),IGRDPT,IPERWT(9,12)                        WEATDT
000005                COMMON /CONS/                                             CONS
000006               1RADTUS,CIS,CNUV,ECCF,PI,PI2,PI32,RADIAN,TWOPI,NAME(2),INP,KO,NLINECONS
000007               2,LINMAX,NPAGE,ISEGF,ISWTB,ISWR,IREFF,IWTHF,IWTAPE,NOCROP,IBLKF,  CONS
000008               3IPLK,IPREC,NLREC,ISWRTO,IVCT,NLATSW,IGRDN                CONS
000009                REAL NAME                                                 CONS
000010                COMMON /FLAG/                                             FLAG
000011               1HEADER(18),ICASE,ICSESH,ICSESG,ISTIME(3),ISDAY,NDAYS,IOPT,LVLH(2),FLAG
000012               2NVLH,NODAY,IV1TIM(3),IV2TIM(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARRFLAG
000013               3,ICDAY,ISWVST(2),ISLATS,INLATF,IRUNF,ISRATE,NLAT,NMAX,ISOSTR,IREPTFLAG
000014                LOGICAL IREPT                                             FLAG
000015                COMMON /WORK/      RAND,                                  WORK
000016               1ITMODE,DLPNTH,DLNMAX,DLONG(3,2),TIME(3,2),ALT(2),PHINDE,SA(3),  WORK
000017               2DELTA,XTIME(3),XDLONG(3),XALT,SIGMA,DDLNGI,CHLCAL,CHECKR,DECR,IS,  WORK
000018               3CTA,YALT,FL,CHLONG,DDLNGR,ICLOCV(366),IZDAY,IREV,IVFH,LAT,LATI,  WORK
000019               4ICLOD,ICCOCT                                             WORK
000020                DOUBLE PRECISION RAND                                     WORK
000021                DIMENSION F(9),      IMTD(12),YCON(9,12),Y(10)            WEATH
000022                DOUBLE PRECISION FL                                       WEATH
000023                DATA IMTD/31,60,91,121,152,182,213,244,274,305,335,366/   WEATH
000024                DATA YCON/.069,.1628,.2178,.2791,.3491,.4324,.5389,.7050,1.0,  WEATH
000025               1.093),.1704,.2276,.2916,.3641,.4496,.5580,.7245,1.0,      WEATH
000026               2.0920,.1699,.2276,.2917,.3645,.4505,.5592,.7257,1.0,      WEATH
000027               3.0990,.1797,.2394,.3055,.3799,.4674,.5773,.7427,1.0,      WEATH
000028               4.1070,.1908,.2525,.3205,.3969,.4860,.5967,.7606,1.0,      WEATH
000029               5.1370,.2290,.2959,.3686,.4491,.5411,.6523,.8085,1.0,      WEATH
000030               6.1630,.2637,.3359,.4132,.4976,.5918,.7022,.8089,1.0,      WEATH
000031               7.1660,.2682,.3412,.4194,.5044,.5990,.7093,.8563,1.0,      WEATH
000032               8.1720,.2738,.3465,.4245,.5088,.6028,.7123,.8563,1.0,      WEATH
000033               9.1710,.2710,.3439,.4211,.5052,.5989,.7084,.8531,1.0,      WEATH
000034               A.1120,.1902,.2547,.3214,.3967,.4845,.5940,.7571,1.0,      WEATH
000035               B.0950,.1711,.2276,.2905,.3620,.4467,.5542,.7199,1.0/      WEATH
000036                FL = FAPD                                                 WEATH
000037                CALL NTRAN(IWTAPE,10,22)
000038                CALL WRITWT(0)                                            WEATH
000039                IREC = 1                                                  WEATH
000040                DO 150 I=1,IGRDN                                          WEATH
000041                CALL RDWETR(IREC)                                         WEATH
000042                IGRID = IGRDPT                                            WEATH
000043                DO 90 IMO=1,12                                            WEATH
000044                IF(IMO .EQ. 1)GO TO 5                                     WEATH
000045                IBG = IMTD(IMO - 1)
000046                GO TO 7                                                   WEATH
000047              5 IBG = 0                                                   SETL
000048              7 IEND = IMTD(IMO)                                          WEATH
000049                DO 15  K =1,9                                             WEATH
000050                 F(K) = 0                                                 WEATH
000051                DO 10 J=1,K                                               WEATH
000052                 F(K) =  F(K) + IPERWT(J,IMO)                             WEATH
000053             10 CONTINUE                                                  WEATH
000054             15 CONTINUE                                                  WEATH
000055                IF(F(9) .EQ. 0)GO TO 80                                   SETL
000056                IF(F(9) .GE. 90. .AND. F(9) .LE. 110.)GO TO 20            SETL
000057                CALL PAGER(2)                                             WEATH
000058                WRITE(KU,900) IGRID,IMO                                   WEATH
```

```
000059          900 FORMAT(1H0,84H*** THE CUMULATIVE FREQUENCY DISTRIBUTION -FSUBR IS WEATH
000060            1NOT BETWEEN 90 AND 110 FOR GRID ,I5,7H MONTH ,I2)                    SET1
000061              GO TO 40                                                            SET1
000062          20 Y(1) = 0.0                                                           SET1
000063              DO 30 K=1,9                                                         SET1
000064              Y(K+1) = F(K)/F(9)                                                  SET1
000065          30 CONTINUE                                                             WEATH
000066              GO TO 47                                                            WEATH
000067          40 Y(1) = 0.0                                                           WEATH
000068              DO 45 K=1,9                                                         WEATH
000069              Y(K+1)=YCON(K,IMO)                                                  WEATH
000070          45 CONTINUE                                                             WEATH
000071          47 L = 1                                                                WEATH
000072          50 CALL RDM1A(FL,U)                                                     WEATH
000073              DO 55 K=1,9                                                         WEATH
000074              KK = K                                                              WEATH
000075              IF(U .GE. Y(K) .AND. U .LE. Y(K+1))GO TO 60                         WEATH
000076          55 CONTINUE                                                             WEATH
000077          60 ICLDCV(IBG+L) = KK-1                                                 WEATH
000078              L = L + 1                                                           WEATH
000079              IF((IBG+L).LE. IEND)GO TO 50                                        WEATH
000080          90 CONTINUE                                                             WEATH
000081              CALL DUPKPK(0)                                                      WEATH
000082              CALL WRTIWT(1)                                                      WEATH
000083         150 CONTINUE                                                             WEATH
000084              RAND = FL                                                           SET1
000085              CALL WRTIWT(2)                                                      WEATH
000086              CALL CLOSE(IWTHF,0)                                                       *NEW
000087              CALL NTRAN(IWTAPE,11,22)                                                  **-1
000088              RETURN                                                              WEATH
000089              END                                                                 WEATH
```

```
. CLT WRITWT,1,751007, 50493    , 1

000001              SUBROUTINE WRITWT(IND)                                    WRITWT
000002          C   THIS ROUTINE WRITES A LOGICAL OR PHYSICAL RECORD ON THE WEATHER   WRITWT
000003          C   FILE IWTHF                                                 WRITWT
000004              COMMON /CONS/                                             CONS    **-1
000005             1RADIUS,GIS,GNUV,ECCE,PI,PI2,PI32,RADIAN,TWOPI,NAME(2),INP,KO,NLINECONS
000006             2,LINMAX,NPAGE,ISEGE,ISWTB,ISWR,IREFF,IWTHF,IWTAPE,NOCROP,IBLKF,  CONS
000007             3IPCK,NPREC,NLREC,ISWRTO,IVCT,NLATSW,IGRDN                CONS
000008              REAL NAME                                                 CONS
000009              COMMON /FLAG/                                             FLAG
000010             1HEADLP(10),ICASE,ICSESN,ICSESG,ISTIME(3),ISDAY,NDAYS,IUPI,LVLH(2),FLAG
000011             2NVLH,NODAY,IV1TIM(3),IV2TIM(3),IGRID,ISWDST,JDAY,IDAY,IYEAR,IYEARBFLAG
000012             3,ICDAY,ISWVST(2),ISLATS,INLATE,IRUNF,ISWATE,NLAT,NMAX,ISUSTR,IREPTFLAG
000013              LOGICAL IWEPT                                             FLAG
000014              COMMON /DIRAC/                                            DIRAC
000015             1IA(200),IB(250),IPC(46)                                   DIRAC
000016              DIMENSION IBC(2)                                          WRITWT
000017              DATA IBC/4HNLAT,4HNER /                                   WRITWT
000018              IF(IND .EQ. 0)GO TO 10                                    WRITWT
000019              IF(IND .EQ. 1)GO TO 100                                   WRITWT
000020              FIND(IWTHF'IREC)
000021              WRITE(IWTHF'IREC,ERR=200) (IB(I),I=1,NPREC)
000022            5 RETURN                                                    WRITWT
000023           10 DO 15 I=1,NPREC                                           WRITWT
000024              IB(I) = 0                                                 WRITWT
000025           15 CONTINUE                                                  WRITWT
000026              IB(1) = IPC(1)                                            WRITWT
000027              IB(2) = IBC(2)                                            WRITWT
000028              IB(3) = NMAX                                              WRITWT
000029              DEFINE FILE 11(3201,230,U,IDUM)                           *NEW
000030              FIND(IWTHF'1)                                             **-1
000031              WRITE(IWTHF'1,ERR=200) (IB(I),I=1,NPREC)
000032              IWRIT = 0                                                 WRITWT
000033              DO 20 I=1,NPREC                                           WRITWT
000034              IB(I) = 0                                                 WRITWT
000035           20 CONTINUE                                                  WRITWT
000036              GO TO 5                                                   WRITWT
000037          100 IF(IWRIT .NE. 0)GO TO 150                                 WRITWT
000038          105 INDEX = IGRID -(IGRID/IBLKF)*IBLKF                        WRITWT
000039              IF(INDEX .NE. 0)GO TO 110                                 WRITWT
000040              IREC = IGRID/IBLKF + 1                                    WRITWT
000041              IWRIT = 1                                                 WRITWT
000042              INDEX = IBLKF                                             WRITWT
000043          110 I = (INDEX-1)*NLREC                                       WRITWT
000044              DO 120 J=1,NLREC                                          WRITWT
000045              IB(I+J) = IPC(J)                                          WRITWT
000046          120 CONTINUE                                                  WRITWT
000047              GO TO 5                                                   WRITWT
000048          150 FIND(IWTHF'IREC)
000049              WRITE(IWTHF'IREC,ERR=200) (IB(I),I=1,NPREC)
000050              IWRIT = 0                                                 WRITWT
000051              DO 160 I=1,NPREC                                          WRITWT
000052              IB(I) = 0                                                 WRITWT
000053          160 CONTINUE                                                  WRITWT
000054              GO TO 105                                                 WRITWT
000055          200 WRITE(KO,900)                                             WRITWT
000056          900 FORMAT(1HO,31H *** AN IRRECOVERABLE I/O ERROR HAS OCCURRED ON WRIT
000057             1ING A RECORD ON THE WEATHER FILE  /10X,31H THE JOB IS BEING ABANDO
000058             2NED ***)
```

WRITWT

STOP
END

000059
000060

3. TOC

SACS BOOK VI

Table of Contents

		Page
Part I	Problem Description	108
Part II	Common Block Definitions	126
Part III	List of Subroutines and Subroutine Call Structure	134
Part IV	Subroutine Descriptions and Flowcharts	137
Part V	Subroutine Listings	165

PART I

PROBLEM DESCRIPTION

SACS PROBLEM DESCRIPTION

1.0 SCOPE

This document describes the requirements and processing logic for the Segment Acquisition Selector Program (SACS). This program is an integral part of the Large Area Crop Inventory Experiment (LACIE) System.

1.1 Program Capabilities. This program selects data from the Segment Reference File based upon data input manually and from a Crop Window File. It writes the extracted data to a Data Acquisition file and prints two summary reports describing the selected data. A processing flow diagram is shown in Figure 1-1.

1.2 Program Development and Organization. The program will be developed on the CDC 6600 Time Sharing System using ANSI Standard FORTRAN. The development procedures provide for adherence to the following guidelines:

a. Capabilities used in CDC FORTRAN will be compatible with UNIVAC 1108 compiler capabilities to provide for conversion with minimum amount of effort.

b. The program, its data and system subroutines will require no more than 24K words of core for the UNIVAC 1110.

c. Subroutines will be no longer than 100 FORTRAN statements and will clearly identify in a header comment block any PDP incompatibilities.

d. Variable naming conventions - Each variable will be started with one letter identifying its source or the common block in which it resides. It will be followed by 4 or 5 characters identifying the variable a prefix of I or X will be used to denote integer or real respectively, if required.

1.3 Operational Assumptions. The file restrictions are outlined in the file descriptions (see Appendix A Segment Reference Data File, Appendix B Crop Window File and Appendix C Data Acquisition File). Additional restrictions are outlined as follows:

a. All detected errors cause the program to abort and the output files and reports are terminated without completion.

b. The following program data maximums are checked.

1. Only one country may be selected for any one program run.
2. Only one case of input parameters will be processed against a single Segment Reference Data File and Crop Window File.
3. A maximum of 426 days is allowed for the acquisition period.
4. A maximum of 4 windows is allowed.

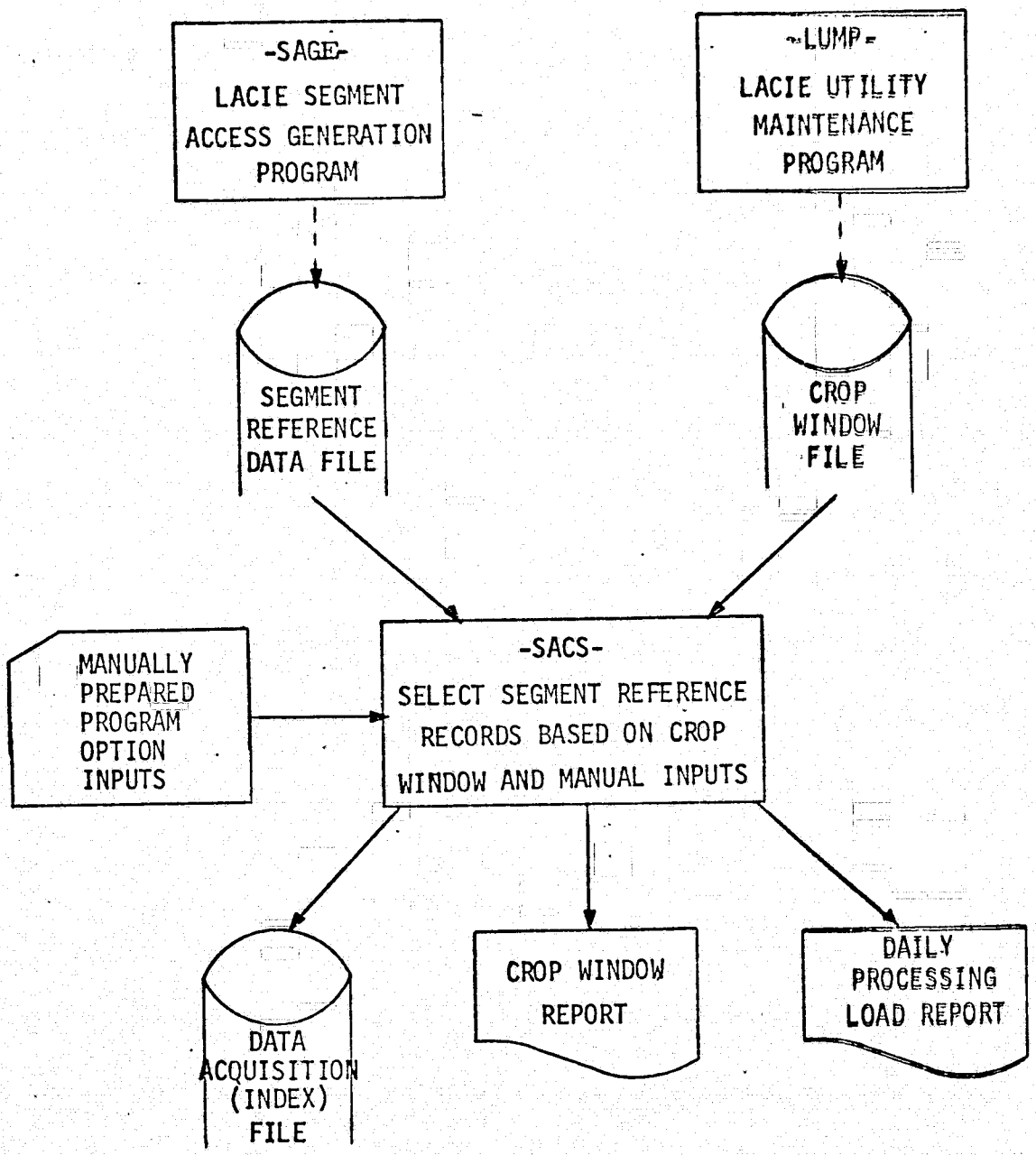


Figure 1-1 SACS Processing Flow Diagram

a. Name - SEG REFE

b. Access Method - Fixed length record sequential access method, standard FORTRAN access.

c. Status or retention cycle - not applicable.

d. Sort - A Header Record precedes the file and is followed by data records then a trailer record. The data records are in sort (from major to minor) as follows:

1. Country ID (ZZZZ for trailer record)
2. Region ID
3. Zone ID
4. Strata ID
5. Substrata ID
6. Segment ID

e. Media - Disk or Tape

2.2.2 Crop Window File. This input file contains a set of time windows for each geographical segment that specify growing periods for a crop. The file is generated by the LACIE Utility Maintenance Program (LUMP) and is used soley by this program (SACS) for the selection of data in the SEG REFER file. A detailed file description is found in Section 2.4 of the Users Manual. The file attributes are summarized as follows:

a. Name - CROPWIND

b. Access Method - Fixed length record sequential access method, Standard FORTRAN Access.

c. Status or Retention Cycle - Not applicable

d. Sort - A header record preceeds the file and is followed by data records then a trailer record. The data records are in sort (from major to minor) as follows:

1. Country ID (ZZZZ for trailer record)
2. Region ID
3. Zone ID
4. Strata ID
5. Substrata ID

2.0 INPUTS

Inputs to this program are supplied by the following:

a. Segment Reference Data File
b. Crop Window File
c. Manually generated parameter card input

2.1 Card Inputs

2.1.1 Card Input Data Quantities. Each parameter value entered via the input cards must be entered unless explicitly stated. All integers must be right justified, all hollerith fields must be left justified. Each parameter value is outlined as follows:

a. XITTL - Card 01, Columns 1 through 48. Contains the case title that is to be printed at the top of each output page.

b. IACASE - Card 02, Columns 1 through 4. Contains the integer case number that identifies the ACQUISI file and each printed output page.

c. ISCASE - Card 02, Columns 5 through 8, Contains the integer case number of the Segment Reference Data File (if 0 or blank no check on input case No. will be made).

d. IWCASE - Card 02, Columns 9 through 12. Contains the integer case number of the Crop window input file (if 0 or blank no check on input case No. will be made).

e. IPCC - Card 02, Columns 13 through 15. Contains an integer that specifies the maximum percent cloud cover to be used in selection of ACQUISI file segments. This number is expressed in tenths of a percent (e.g., 15.4% would be expressed as 154). All ACQUISI File cloud cover percents must be less than or equal to this value.

f. ISANG - Card 02, Column 16 through 19. Contains an integer that specifies the minimum degrees sun angle to be used in selection of ACQUISI file segments. This number is expressed in hundreds of degrees and ranges between 0 and 90 degrees (e.g., 12.50 degrees would be expressed as 1250). All ACQUISI file sun angles must be greater than or equal to the specified angle.

g. ICOUN - Card 02, columns 20 through 23. Contains 4 hollerith characters that identify the country to be selected.

h. IREG - Card 02, Columns 24 through 26. Contains a three digit integer that identifies the region to be selected.

i. IZONE - Card 02, Columns 27 through 29. Contains a three digit integer that identifies the zone to be selected. A zero entry allows all zones in a region to be selected.

j. Output options, Columns 30 through 31. Contain a logical flag that specifies whether the program reports are to be generated. A T specifies print the report; an F specifies don't print the report.

- IPRPT-30- Daily Processing Load Report
- ICRPT-31- Crop Window Report (segment acquisition)

2.1.2 Card Formats. The SACS program requires two input cards. Each card has fixed field inputs. The card formats are shown in Figure 2-1 and each field is described in Section 2.1.1.

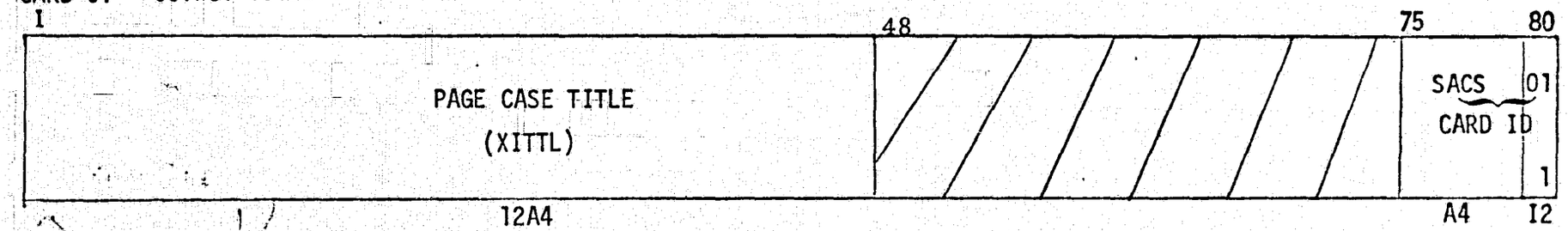
2.1.3 Deck Setup. Each of the two input cards must be provided and must be supplied in card number order (see columns 79 and 80).

2.1.4 Rules for Entering Data on Cards. (See 2.1.1)

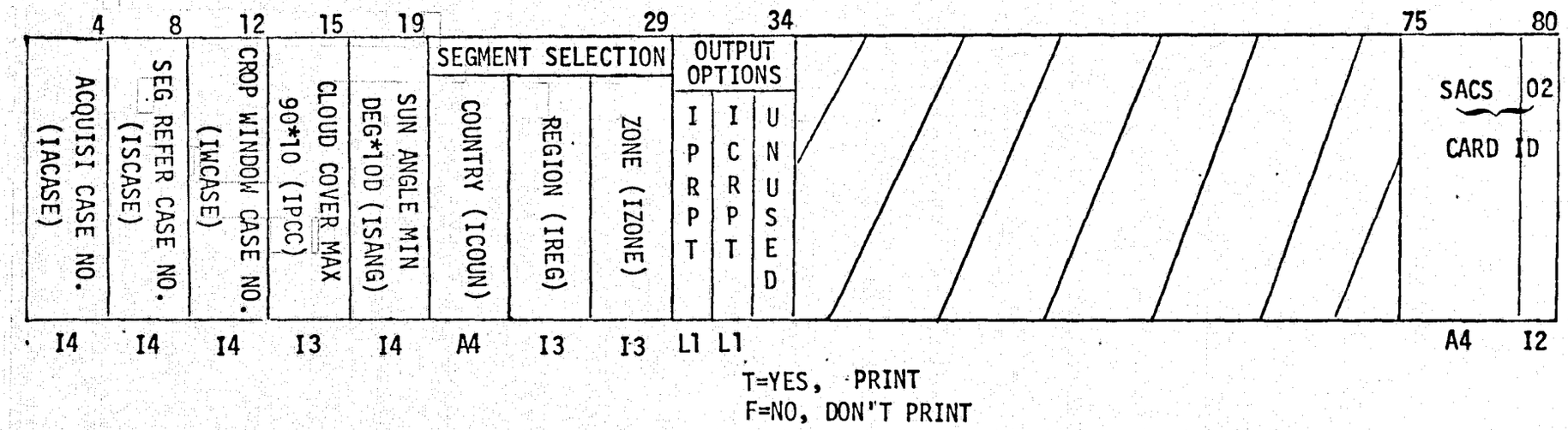
2.2 Input Files.

2.2.1 Segment Reference Data File. This input file is generated by the Segment Access Generator Program and contains a data base of crop status for selected geographical segments of the world. The file is described in Section 2.4 of the Users Manual. The file attributes are summarized as follows:

CARD 01 - OUTPUT TITLE CARD



CARD 02 - PROGRAM OPTIONS CARD



T=YES, PRINT
F=NO, DON'T PRINT

Figure 2-1 Input Card Formats

3.0 PROCESSING

3.1 Overview. The program control flow for the SACS program is shown in Figure 3-1. The routines that perform this processing are summarized as follows:

a. SACS - Main control routine that controls the program flow and sets up all vendor dependent program variables.

b. AINIL - Reads the manually generated card input parameters and performs all file label processing.

c. AREAD - Reads data records from both the Crop Window File and the Segment Reference Data File until a Segment Reference Data file record for a required substrata is located or an end of file or zone is encountered on either file.

d. APRPT - Prints the Daily Processing Load Report.

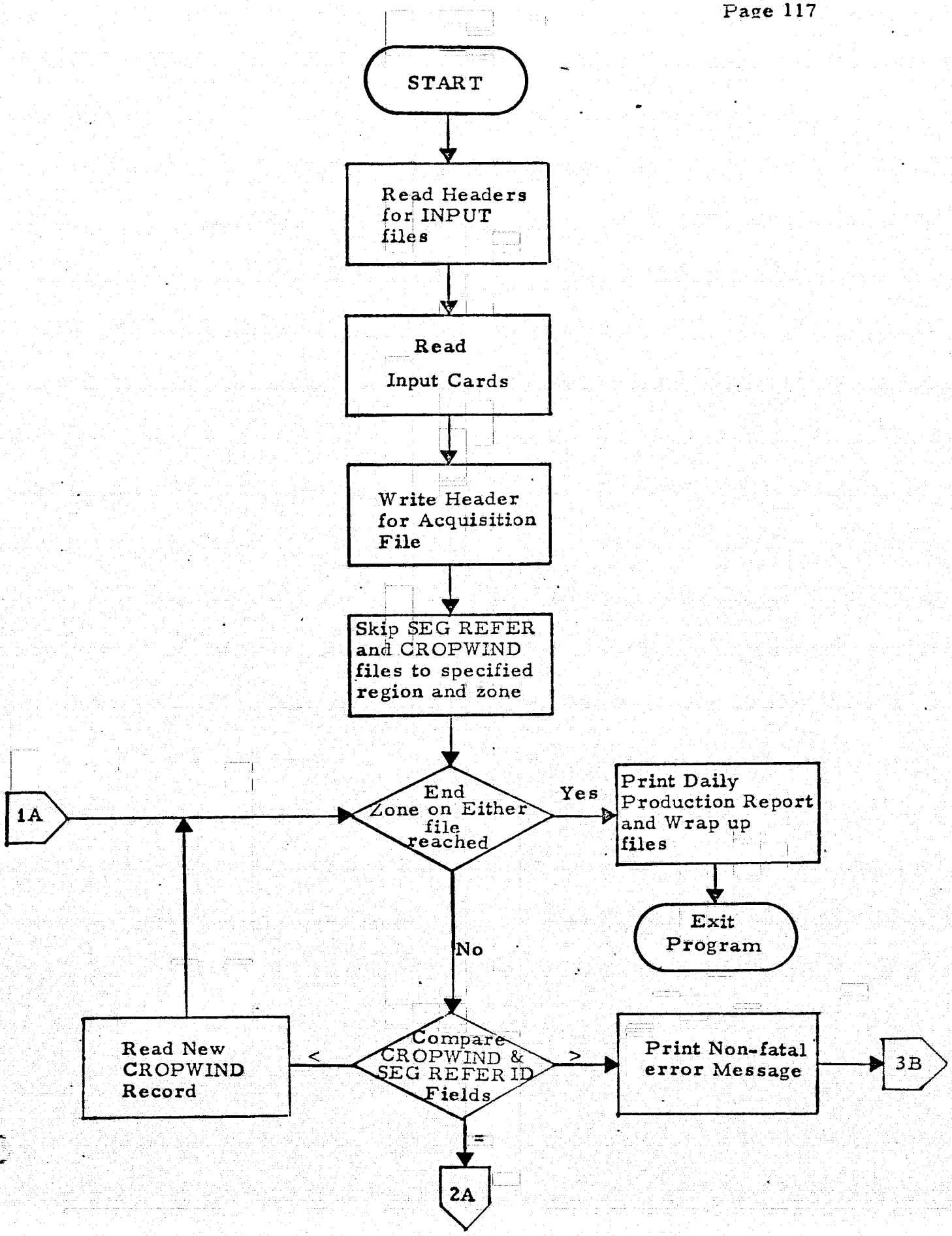
e. AFIND - Selects a Segment Reference Data file record based upon the crop time window and the cloud cover sun angle criteria.

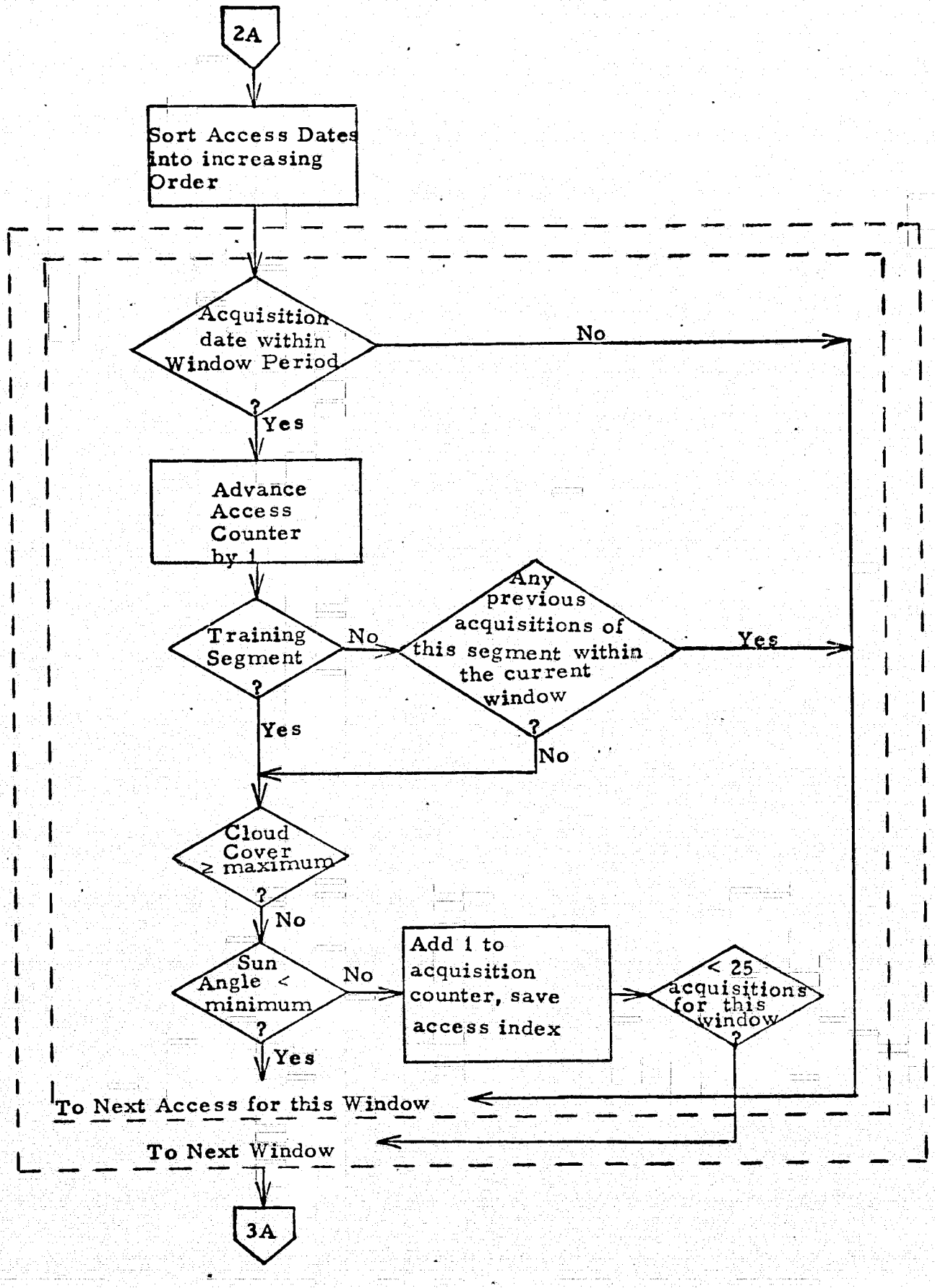
f. ACRPT - Writes the Segment Data Acquisition report.

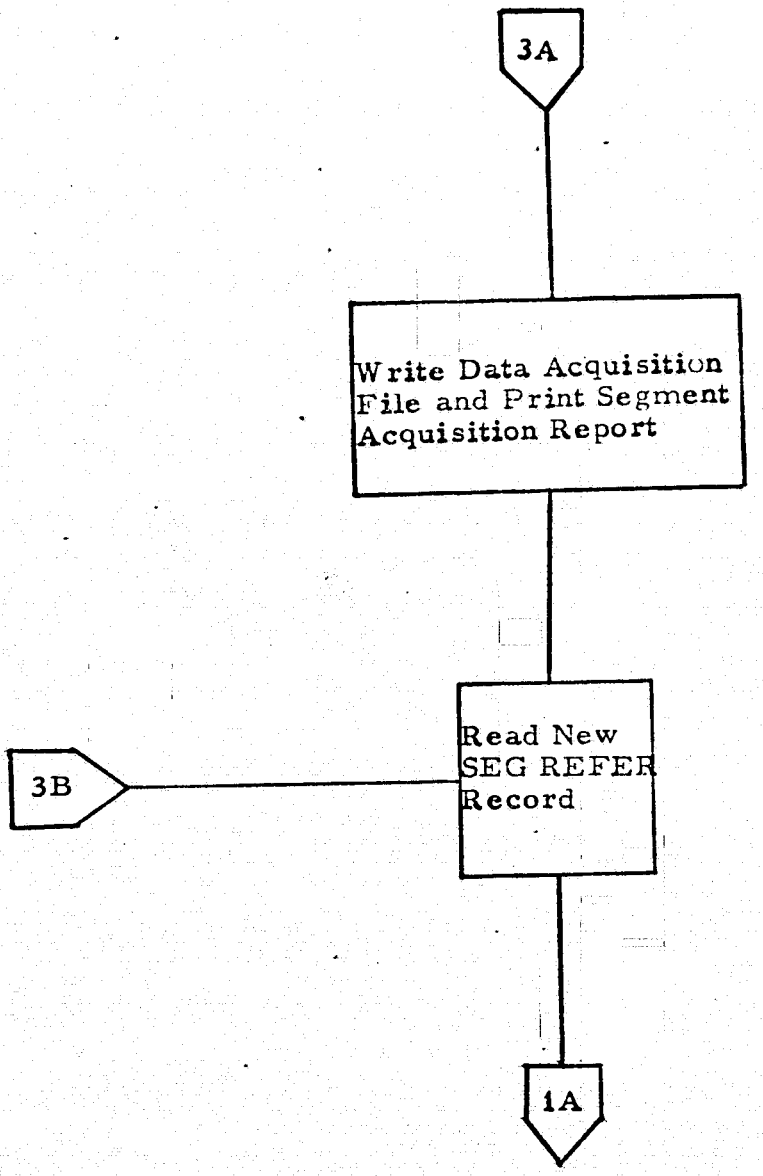
g. AWRTA - Writes the data and trailer records of the Data Acquisition file.

h. ABARF - This routine writes all error messages.

i. FZULU - Converts a date from zulu to MM DD YY


START
Read Headers for INPUT files
Read Input Cards
Write Header for Acquisition File
Skip SEG REFER and CROPWIND files to specified region and zone
1A
End Zone on Either file reached
Yes
Print Daily Production Report and Wrap up files
Exit Program
No
Read New CROPWIND Record
<
Compare CROPWIND & SEG REFER ID Fields
>
Print Non-fatal error Message
3B
=
2A


2A
Sort Access Dates into increasing Order
Acquisition date within Window Period ?
No
Yes
Advance Access Counter by 1
Training Segment ?
No
Any previous acquisitions of this segment within the current window ?
Yes
No
Yes
Cloud Cover ≥ maximum ?
No
Sun Angle < minimum ?
No
Add 1 to acquisition counter, save access index
< 25 acquisitions for this window ?
Yes
To Next Access for this Window
To Next Window
3A


3A
Write Data Acquisition File and Print Segment Acquisition Report
3B
Read New SEG REFER Record
1A

4.0 OUTPUT

Output from the SAC's program consists of the following:

a. Data Acquisition File (unit IAFILE)

b. Crop window report, under option control (see ICRPT, printed output unit IOFILE)

c. Daily Processing Load Report under option control (see IPRPT, printed output unit IOFILE)

4.1 Print Data.

4.1.1 Reports.

4.1.1.1 Control Card Echo Print

4.1.1.2 Crop Window Report. The format of the Crop Window Report is shown in Figure 4-1.

4.1.1.3 Daily Processing Report. The format of the Daily Processing Report is shown in Figure 4-2.

4.1.2 Card Outputs. Not Applicable.

4.1.3 Output Files. This program outputs the Data Acquisition File (see Section 2.4 of the Users Manual) the file indexes selected segment acquisitions in the Segment Reference Data File. The file attributes are summarized as follows:

a. Name - ACQUISI

b. Access Method - Fixed length record sequential access method standard FORTRAN access.

c. Status or Retention cycle - Not applicable.

d. Sort - A header record preceeds the data records and they are followed by a trailer. The data records are in sort (from major to minor) as follows:

The print format is:

HEADER

AAAA ——————————————→ AAA

ICASE	ICSESR	ICSECW	IPCC	ISANG	ICOUN	IREG	IZONE	IPRPT	ICRPT
NNNN	NNNN	NNNN	NNN	NNNN	NNNN	NNN	NNN	L	L

1. Country ID
2. Region ID
3. Zone ID
4. Strata ID
5. Substrata ID
6. Segment ID

```
CASE XXXX        19                                44  |---- CASE TITLE ----| 91      97
                 SEGMENT ACQUISITION DATA                                             PAGE  XXXX
                 19               33               45          55          65
                 COUNTRY XXXX     REGION XXX       ZONE XXX    STRATA XXXX SUBSTRATA XXXX

 2   5           11                          34    35 ---------- 60   61 ---------- 85   86 ------ 13
S/W  SEGMENT     XXXXX WINDOW TITLE XXXXX             WINDOW 2             WINDOW3            WINDOW4
     ID-TOTAL    11              18
                 TOTAL           SELECTED
                 ACCESS          ACQUISITIONS
W    5           12              18
     XXXXX       XXX             XXX XXX XXX XXX
                                 XXX XXX
```

SPACING

1. SINGLE SPACE BETWEEN LINES OF SAME SEGMENT
2. DOUBLE SPACE BETWEEN SEGMENTS

PAGING

1. 39 LINES/PAGE CDC 6600
2. 58 LINES/PAGE UNIVAC, PDP

PRINTING

IF NO. ACQUISITIONS =0, SELECTED ACQUISITIONS =0 PRINT "NONE"

IF NO. ACQUISITIONS ≠0, AND NONE SELECTED PRINT "ALL ILLEGAL"

Figure 4-1 Format of Cropwindow Report

DATE MM DD YY DAILY PROCESSING LOAD REPORT + CASE TITLE ———— PAGE XX

START DATE XXXXXX COUNTRY XXXX REGION XXX ZONE XXX

2 RELATIVE DAY	NO. ACQUISITIONS	26 RELATIVE DAY	NO. ACQUISITIONS	51 RELATIVE DAY	NO ACQUISITIONS	76 RELATIVE DAY	NO ACQUISITIONS
01	XXXXXX	26	XXXXXX	51		76	
02							
03							
25		50	XXXXXX	75		100	

UP TO 6 PAGES EACH PAGE CONTAINING UP TO 100 DAYS

Figure 4-2 Format of Daily Processing Load Report

5.0 ERROR PROCESSING

All fatal errors detected by this program shall result in termination of program execution. The detected errors are listed in Table 5-I. The processing logic is described as follows:

a. Each time an error is detected, the routine ABARF is called with the error ID.

b. The routine prints the error and forces the program to terminate execution if the error is fatal or returns control if the error is non-fatal.

Non-fatal errors shall be printed and execution continued.

Table 5-I Error Messages

ERROR ID	MESSAGE
01	Cropwind file has invalid label name xxxxxxxx
02	Seg refer file has invalid label name xxxxxxxx
03	Cropwind input case xxx not equal to label case xxx
04	Seg refer input case xxx not equal to label case xxx
05	Invalid country input xxxx cropwind xxxx seg refer xxxx
06	No data selected for ACQUISI file
07	Seg refer acquisition day too large xxx, max-xxx, record=xxxxx
08	Input cards invalid or out of sequence xxxxxx (Column 75-80)
09	Input sun angle less than 0 or greater than 90.00 xxxxxx
10	Percent cloud cover less than 0 a greater than 100.0 xxxx
11	No. days in study exceeds 426
*12	Crop window is missing a zone xxx

*Non-fatal error message

PART II

COMMON BLOCK DEFINITIONS

COMMON STORAGE ALLOCATION

Name PBLK **Size** _____ **Page** _____ **of** _____

Function This Common Block Contains Potpourrie Variables

Name	Dimension	Format	Description	Symbol	Units
IPEOF	1	L	End of file flag F = more data, T = end file		
IPFTF	1	I1	First time Flag 0 = first time, 1 = subsequent times, -1 = last time		
IPTYPE	1	I1	Type wheat (o = unidentified, S = spring, 1 = winter) initially zero		
IPERR	1	I3	Acquisition day when it exceeds maximum		
IBCD	14	11A1	Numbers in Hollerith 0,1, 3...9, blank, -, ., *, Comma		
ITEN	5	I5	Powers of 10 1, 10, 100, 1000, 10000		
IPEFLG	13*1	12L	Flag to denote whether errors 1 through 12 are fatal (T = yes; F = no) TTTTTTTTTTFF		
IPWRL	150*1		Integer work area		
XPWRK	150*1		Real work area		

COMMON STORAGE ALLOCATION

Name IBLK **Size** **Page** 1 **of**

Function This Common Block Contains all the Manually Generated Input Parameters and the Vendor Dependent Variables

Name	Dimension	Format	Description	Symbol	Units
ICID	3*2	3A2	Columns 75-80 input card		
XITTL	12, 1	12A4	Report case oriented page title		
IACASE	1	A4	Case number that identifies the acquisition data file		
ISCASE	1	I4	Case number that identifies the segment reference data file (0, implies case number is not used)		
IWCASE	1	I4	Case number that identifies the crop window file (0, implies case number is not used)		
IPCC	1	I3	The maximum percent cloud cover to be used in selection of acquisition data file segments, the number is in tenths of a percent		10*90
ISANG	1	I4	The minimum degrees of sun angle to be used in selection of acquisition data file segments. Hundreds of degrees and ranges between 0 and 90 degrees sun angle in radians		Deg*100
XICOUN	1	A4	Country ID		
IREG	1	I3	Region ID		
IZONE	1	I3	Zone ID		
IPRPT	1	L	Processingload report flag, T=print		
ICRPT	1	L	Crop window report flag, T=print		

COMMON STORAGE ALLOCATION

Name IBLK Size Page 2 of

Function

Name	Dimension	Format	Description	Symbol	Units
IIFILE	1	I2	Fortran unit number for manually generated input		
IOFILE	1	I2	Fortran unit number for crop window and segment acquisition reports		
ISFILE	1	I2	Fortran unit number for segment reference data file		
ICFILE	1	I2	Fortran unit number for crop window file		
IAFILE	1	I2	Fortran unit number for data acquisition file		
INLINE	1	I2	Number of lines per page		
XTWTTL	5, 8	5A4	Crop window titles 4 spring windows followed by 4 winter		

COMMON STORAGE ALLOCATION

Name ABLK **Size** 1390 words **Page** ____ of ____

Function This common block contains the data elements required to generate the ACQUI file, the crop window report and the daily processing load report

Name	Dimension	Format	Description	Symbol	Units
ACRN	1	F6	Current record number		
AFNAM	2, 1	2A4	File name initialized by data statement "ACQUIS"		
AFTRL	1	1A4	TRAILERID "ZZZZ"		
IACNT	4, 1	I2	Total number of accesses for each window. IACNT(1)=window 1, ... IACNT(4)=WINDOW 4		
IATOT	1	I3	Total number accesses all windows		
IAPNT	25, 4	I3	Pointer to the segment reference file that identifies an acquisition entry that has been selected		
IAPAG	1	I3	Current page number (initially 0)		
IANLIN	1	I2	Number of lines remaining on page (initially 0)		
ANACQ	426	R1	Number of acquisitions per day. Subscript equals day relative to given start date (ISDATE)		

COMMON STORAGE ALLOCATION

Name CBLK Size Page of

Function This common block contains the data elements from the crop window file

Name	Dimension	Format	Description	Symbol	Units
ICCRN	1	FG	Current record number		
CFNAMI	2, 1	2A4	File name that must appear in header initially set to "CROPWIND"		
CFNAM2	2, 1	2A4	Actual name of file		
ICCASE	1	I3	Case number from file header		
ICWIND	1	I1	Number of crop windows (8)		
CCOUN	1	1A4	Country ID		
ICREG	1	I3	Region ID		
ICZONE	1	I3	Zone ID		
ICSTRA	1	I4	Strata ID		
ICSUBS	1	I4	Substrata ID		
ICLOW	8, 1	I5	Start date relative to Jan. 1, 1950		
ICHI	8, 1	I5	Stop date relative to Jan. 1, 1950		
			* Note first 4 represent winter wheat windows last 4 represent spring wheat windows		

COMMON STORAGE ALLOCATION

Name SBLK Size_____ Page_____ of_____

Function This common block contains the variables extracted from the segment reference data file (Seg Refr)

Name	Dimension	Format	Description	Symbol	Units
SCRN	1	FG	Current record number		
SFNAM1	2, 1	2A4	File name that must appear in header 'seg. refr'		
SFNAM2	2, 1	2A4	Actual name of file		
SFTRL	1	4A4	Trailer ID "ZZZZ"		
ISSCAS	1	I3	Case number from file header		
ISDATE	1	I5	Reference date, integer no. of days since 1 Jan. 1950		
ISNDAY	1	I3	Number of days in study (1 to 426)		
ISNAC	1	I3	No. of acquisitions in record		
SCOUN	1	1A4	Country ID		
ISREG	1	I3	Region ID		
ISZONE	1	I3	ZONE ID		
ISSTRA	1	I4	Strata ID		
ISUBS	1	I4	Substrata ID		
ISSEG	1	I5	Segment ID		
SPCSW	1	F6.2	Percent spring wheat		
SPCWW	1	F6.2	Percent winter wheat		
ISTRN	1	I1	Training segment indicator, 0=normal, 1=training,		

COMMON STORAGE ALLOCATION

Name SBLK **Size** **Page** 2 **of** 2

Function This common block contains the variables extracted from the segment reference data file (Seg. Refr.)

Name	Dimension	Format	Description	Symbol	Units
ISRDAY	150, 1	I3	Acquisition day for each of 150 acquisition entries		
SSANG	150, 1	I4	Sun angle radians		
ISPCC	150, 1	I4	Percent cloud cover * 10		90*90

PART III

LIST OF SUBROUTINES
AND
SUBROUTINE CALL STRUCTURE

LIST OF ROUTINES IN SACS

	Name	Function
1.	ABARF	Processes errors and prints messages
2.	ACRPT	Print acquisition report
3.	AFIND	Determines segment acquisitions from the segment reference file
4.	AINIL	Performs initialization functions
5.	APRPT	Prints the daily processing load report
6.	AREAD	Reads one record from the crop window and segment reference data files.
7.	AWRTA	Writes one record on the acquisition file.
8.	BINBCD	Converts an integer to display code and places it in a hollerith string for printing
9.	FZULU	Converts a zulu date to day, month and year
10.	MAIN	Main control
11.	SORTAG	Internal array sort routine

SACS (MAIN)
- AINIL
  - ABARF
  - AWRTA
- AREAD
  - ABARF
- AFIND
  - ABARF
- AWRTA
- ACRPT
  - BINBCD
- APRPT
  - FZULU

PART IV

SUBROUTINE DESCRIPTIONS AND FLOWCHARTS

SUPPLIED UTILITY ROUTINES

Routine Day

Call Day (IYMD, IDAY)

Given IYMD (3) where

IYMD (1)	IS Day No.
IYMD (2)	IS Month No.
IYMD (3)	IS Year No.

Compute year day no. in IDAY

Routine PIMOD

Call PIMOD (A)

Convert ±A in radians to an angle 0-2π

Routine SOL (Entry ALPHA)

Call ALPHA (IFLAG)

For emphemeris usage as called by hector

computes ALPHAM and ALPHAT and IFLAG = 1

Routine PAGER (Entry Eject)

Call PAGER (NLINES)

Updates line count in NLINE with NLINES

NPAGE = 0 causes page to be restored prior to print.

NPAGE - page no.

HEADER- 80 char. 20A5

ICASE- case no.

KO - 6 print unit

INMAX is max no. of lines allowed

Initially NLINE should be set > LINMAX and NPAGE = 0

SUPPLIED UTILITY ROUTINES
(CONTINUED)

Call EJECT (NLINES)

Causes page to be restored automatically and then prints headers.

Routine CLDAY

Call CLDAY

Given IDAY-DAY no. of the year compute in LMO-the month and in LDA the day no.

Need: IYEAR = 0 - Leap Year, ≠ 0 not Leap Year

Routine KEPLER

Call KEPLER (XM, XECC, XE, ERROR)

Given XM - Mean anomaly, XECC - eccentricity

Compute: E-eccentric anomaly, error = 0 means OK

Routine LFPA

Call LFPA [FLDA, LMO, LYR, ALFGM (can be dummy), DAYS]

Given: FLDA - day of month no., LMO - month no.,
LYR - year no. compute ALFGM - right ascension and
DAYS - Zulu day no.

Routine DEGMOD

Call DEGMOND (RAD, IDEG)

Given: angle rad in radians store the angle in deg., min., sec., in IDEG(1) - (3).

Routine FZULU

Call FZULU (IOATE, IOUT)

Given Zulu date in IDATE, compute year, month and day in IOUT(1) - IOUT(3).

Routine RDMIA

Call RDMIA(FL, U)

Given double precision random no. seed in FL, comoute random no. U (0-1) based on uniform distribution.

SUBROUTINE ABARF

Purpose:

This routine prints the error messages for the SACS program and returns control to the operating system.

Input:

The inputs to this routine are provided via the IBLK, CBLK, SBLK and ABLK common blocks. Table 6 shows the variables.

Output:

The outputs from this routine consists of a line of outputs printed on the output file (IOFILE). Table 5-I shows the error messages in the problem description.

Calling Sequence:

Call ABARF (id)

id, error number

Subroutines Used:

None.

Processing:

The routine shall use a "COMPUTED GO TO" to access a write statement that prints the proper error message.

Table 6. Routine ABARF Inputs

Variable Name	Set	Use	Description
CFNAM2		x	Cropwind file name
SFNAM2		x	SEG REFER file name
ICCASE		x	Cropwind case number from file
IWCASE		x	Input cropwind case number
ISCASE		x	Input SEG REFER case number
ISCASE		x	SEG REFER case number from file
ICOUN		x	Input card country ID
CCOUN		x	Cropwind country ID
SCOUN		x	SEG REFER country ID
SCRN		x	Current SEG REFER record number
ISNDAY		x	Maximum number of days in study
IPERR		x	Acquisition day when it exceeds maximum
PCRDID		x	Columns 75-78 of current input card
PCRDNO		x	Columns 79-80 of current input card
ISANG		x	Sun angle
IPCC			% cloud cover
ISNDAY		x	Number of days in study

Subroutine ACRPT

Purpose:

Prints the acquisition report - one record per entry. See the problem description for the report format.

SUBROUTINE AFIND

Purpose:

This subroutine selects acquisitions from a SEG REFER Data Record based upon the following:

a. The acquisition date is within a specified period
b. The cloud cover is less than a specified maximum
c. The sun angle is greater than a specified minimum

Input:

The inputs to this routine are provided via the IBLK; SBLK and CBLK. Table 4 identifies the variables.

Output:

The outputs from this routine are placed into the ABLK. Table 4 identifies the variables.

Calling Sequence:

Call AFIND

Subroutines Called:

ABARF - Error routine.

Processing:

A functional flow diagram is attached.

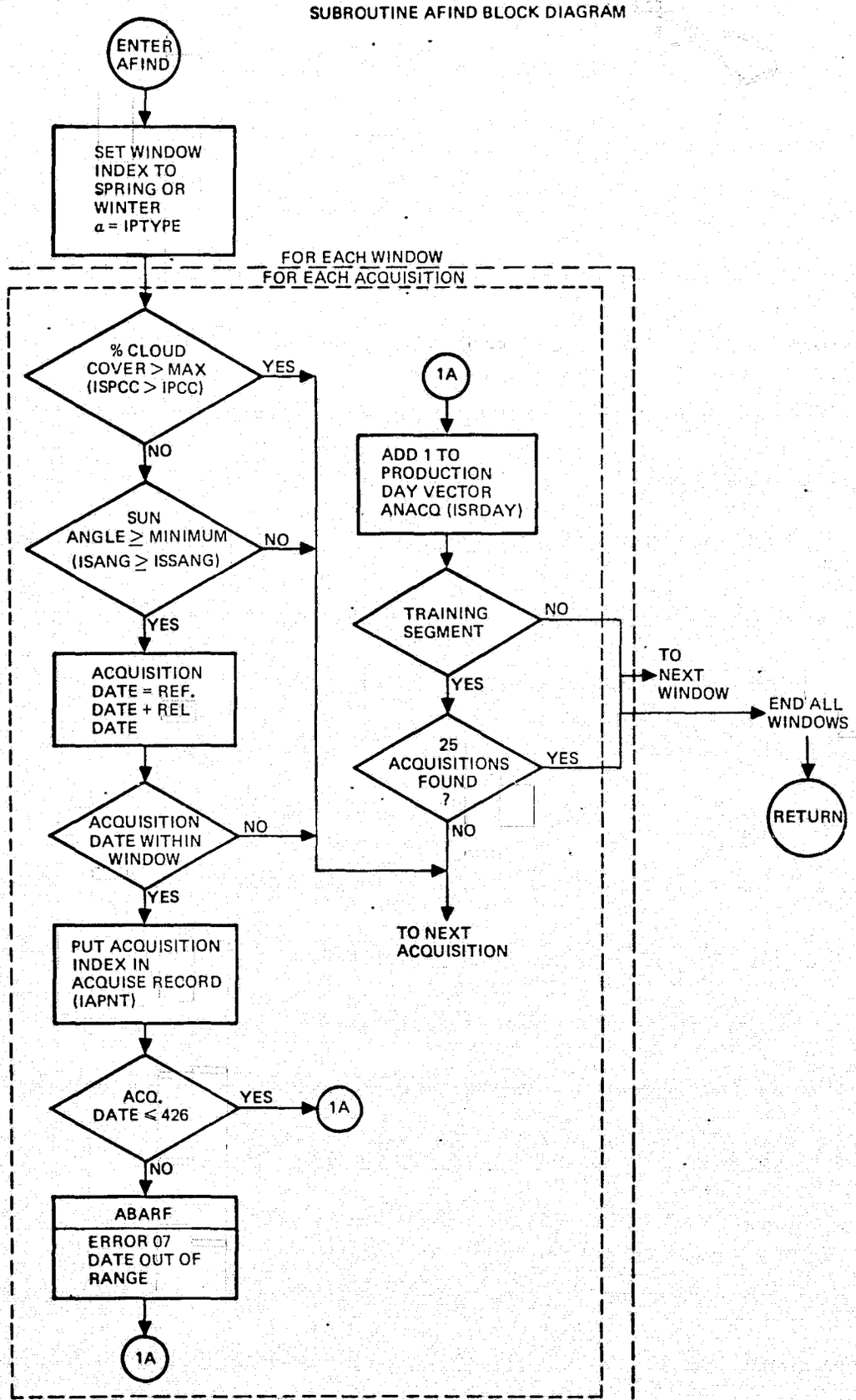
Error Exits

07-SEG REFER acquisition day too large xxx maximum = xxx, record = xxx

Table 4. Routine AFIND Inputs and Outputs

Variable Name	Set	Use	Description
IPTYPE		x	Type wheat (1=spring, 8=winter)
IPERR	x		Acquisition day for out of range date
IPNCW		x	No. crop windows
IPCC		x	Maximum percent cloud cover *10 to be used in acquisition selection
ISANG		x	Minimum degrees of sun angle *100 to be used in acquisition selection
IACNT	x		Total number of accesses for each window - IACNT (1) ⟶ IACNT (4)
IATOT	x		Total number of accesses
IAPNT	x		Pointer to SEGREFER acquisition entry
ANACQ	x		Number of acquisitions per day
ICLOW		x	Start date of window
ICHI		x	End date of window
ISDATE		x	SEG REFER FILE reference date
ISNDAY		x	Number of days in study
ISTRN		x	Training segment indicator (1 = training)
ISRDAY		x	Acquisition day (1 to 150)
ISSANG		x	Sun angle (1 to 150)
ISPCC		x	Percent cloud cover (1 to 150)

SUBROUTINE AFIND BLOCK DIAGRAM



SUBROUTINE AINIL

Purpose:

This subroutine performs all initialization functions for the SACS program. The functions performed are listed as follows:

a. Reads the manually generated input cards.
b. Reads and verifies the headers for SEG REFER and CROP WIND files.
c. Writes the header for the ACQUISI FILE
d. Initializes common variables.

Input:

The inputs to the routine consist of common blocks, and 3 input files. Table 1 identifies the common variables used. The files are outlined as follows.

a. SEG REFER - logical unit ISFILE, format shown in Appendix A.
b. CROPWIND - logical unit ICFILE, format shown in Appendix A.
c. CARD INPUT - logical unit IIFILE, format shown in Section 2.1.1.

Output:

The outputs from this routine consist of common variables and an output tape label. Table 1 identifies the variables. The ACQUISI file header is shown in Appendix C.

Calling Sequence:

Call AINIL

Subroutines Called:

ABARF and AWRTA

Processing:

A functional flow diagram is shown in figure 3.2.2-1.

Error Exits

01	CROPWIND file has invalid label name
02	SEGREFER file has invalid label name
03	CROPWIND input case xxx not equal label case xxx
04	SEGREFER input case xxx not equal label case xxx
11	No. of days in study exceeds 426
08	Input cards invalid or out of sequence
09	Input sun angle less than 0 or greater than 90.00
10	Percent cloud cover less than 0 or greater than 100.0

Table 1. Routine AINIL Inputs and Outputs

Variable Name	Set	Use	Description
ICID	x		Card ID columns 75-80
IPEOF	x		End of file flag (false)
IPTYPE	x		Type of wheat (unidentified, 0)
ICREG	x		Cropwind region ID (0)
ICZONE	x		Cropwind zone ID (0)
ICSTRA	x		Cropwind strata ID (0)
ICSUBS	x		Cropwind substrata ID (0)
CCOUN	x		Cropwind Country (XICOIN)
XITTL	x		Report case oriented page title
IACASE	x		Case number (IIFILE) for ACQUISI file
ISCASE	x		Case number (IIFILE) for Seg. Refer File
IWCASE	x		Case number (IIFILE) for crop wind file
IPCC	x		Percent cloud cover
ISANG	x		Minimum degrees sun angle (degrees)
XISANG	x		Minimum degrees sun angle (radians)
XICOUN	x		Country ID
IREG	x		Region ID
IZONE	x		Zone ID
IPRPT	x		Processing load report flag
ICRPT	x		Crop window report flag
IAPAG	x		Crop window report page number (0)
IANLIN	x		No. lines remaining on page (0)
ANACQ	x		No. acquisitions/day (0)

Table 1. Routine AINIL Inputs and Outputs
(Continued)

Variable Name	Set	Use	Description
CCRN	x		CROPWIND record number (1)
CFNAM1		x	Name of CROPWIND file
CFNAM2	x	x	Cropwind file name from label
ICCASE	x	x	Case number from CROPWIND label
ICWIND	x		No. of crop windows
SCRN	x		Seg. Refer record number (1)
SFNAM1		x	Seg. Refer file name
SFNAM2	x	x	Seg. Refer file name from label
ISCASE	x	x	Case number from header
ISDATE	x		Reference date
ISNDAY	x	x	No. of days in study (1-426)

SUBROUTINE APRPT

Purpose:

This subroutine prints the daily processing load report.

Input:

The inputs to this routine are provided via the IBLK and ABLK, Common Blocks. The variables are shown in Table 3.

Output:

The outputs from this routine consist of the printed Daily Processing Load report.

Calling Sequence:

Call APRPT

Subroutines Called:

FZULU convert Zulu date to MMDDYY

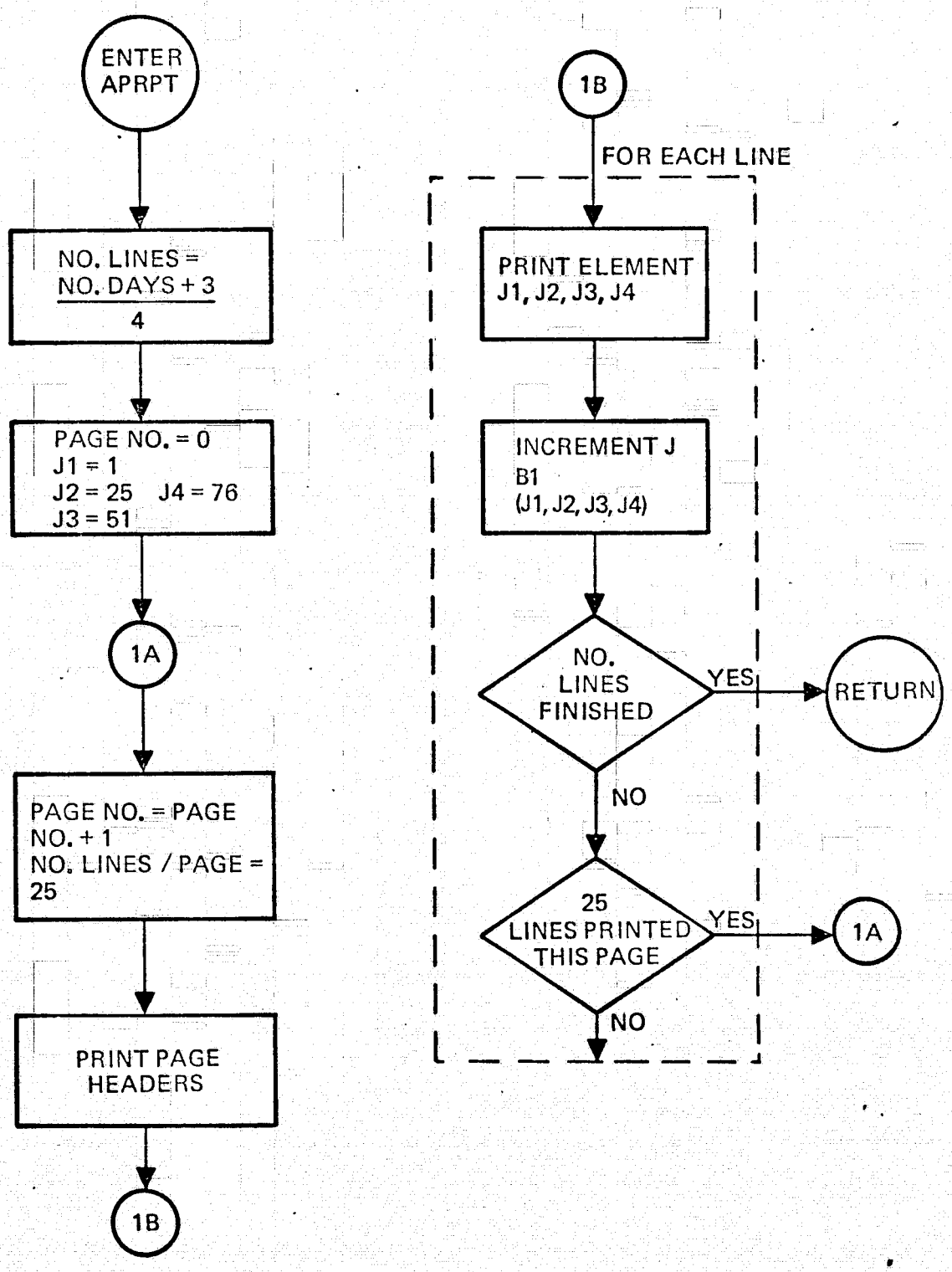
Processing:

A functional flow diagram is attached.

Table 3. Routine APRPT Inputs and OUTPUTS

Variable Name	Set	Use	Description
XITTL		x	Report case title
IOFFLE		x	FORTRAN unit number for report
INLINE		x	No. lines per page
ANACQ		x	No. acquisitions per day
ISDATE		x	Reference date Zulu
XICOUN		x	Country ID
IREG		x	Region ID
IZONE		x	Zone ID
IPTYPE		x	Type of wheat 0, unidentified 1=spring; 5=winter
ISNDAY		x	No. of days in study

SUBROUTINE APRPT BLOCK DIAGRAM



SUBROUTINE AREAD

Purpose:

This subroutine reads both the CROP WINDOW File and the Segment Reference Data File and returns a record of each or sets a flag indicating that an end of data has been encountered.

Input:

Inputs to this routine are provided by the IBLK, PBLK, CBLK and SBLK. Table 2 shows the variables.

Output:

Outputs from this routine are placed into the PBLK, CBLK and SBLK. Table 2 shows the variables.

Calling Sequence:

Call AREAD

Subroutines Called:

ABARF - Error routine.

Processing:

A functional flow diagram is attached.

Table 2. Routine AREAD Inputs and Outputs

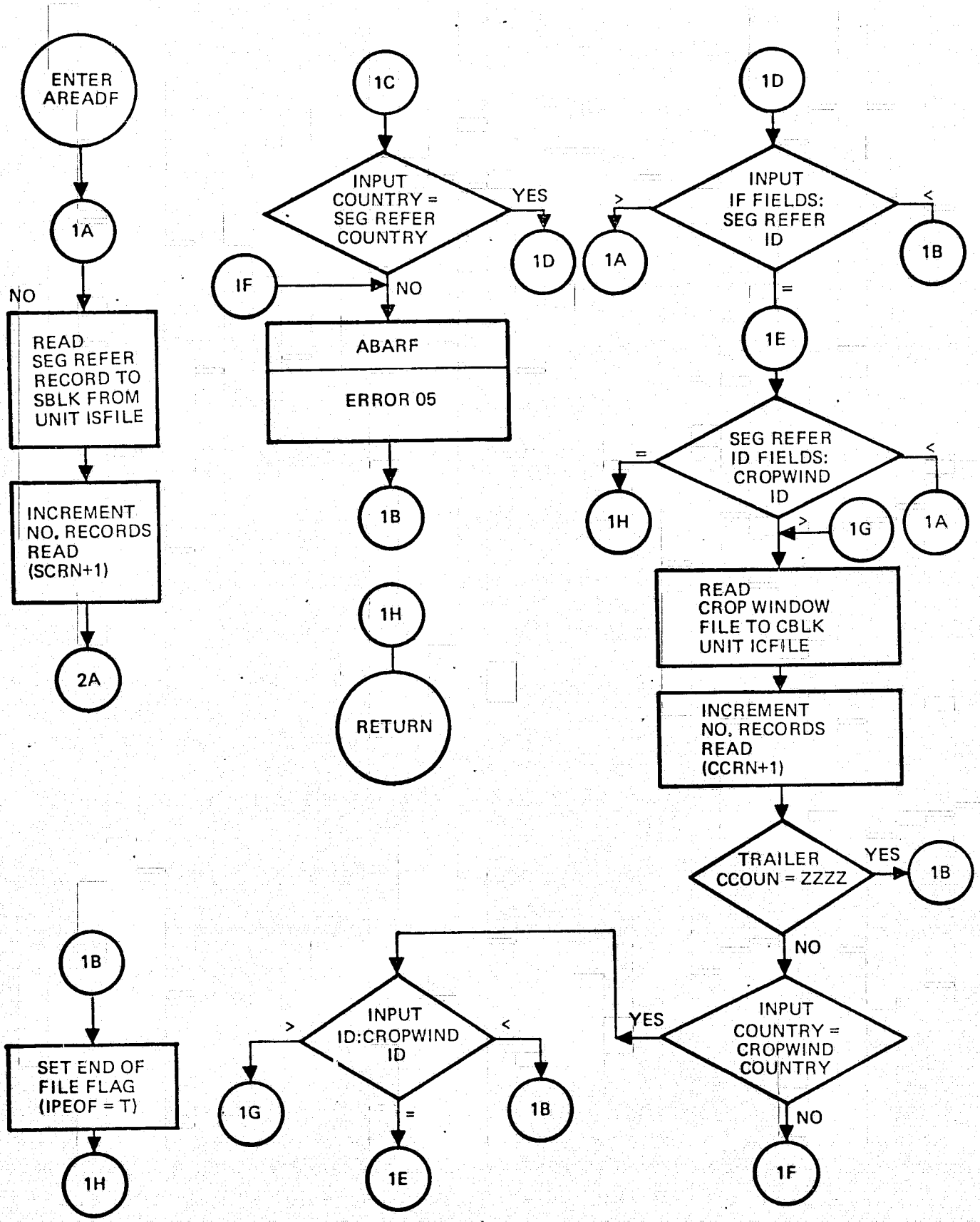
Variance Name	Set	Use	Description
IPEOF	x		End of file flag (T=EOF)
IPTYPE	x	x	Type wheat (0, unidentified; 1, spring; 5, winter)
XICOUN		x	Input country ID
IREG		x	Input region ID
IZONE		x	Input zone ID
ISFILE		x	Fortran unit number segment reference data file
ICFILE		x	Fortran unit number crop window file
CCRN	x		Crop window file current record number
CCOUN	x		Crop window Country ID
ICREG	x		Crop window Region ID
ICZONE	x		Crop window Zone ID
ICSTRA	x		Crop window Strata ID
ICSUBS	x		Crop window Substrata ID
ICLOW	x		Crop window, window start dates
ICHI	x		Crop window, window ending dates
SCRN	x		SEG REFER current record number
SCOUN	x		SEG REFER country ID
ISREG	x		SEG REFER region ID
ISZONE	x		SEG REFER zone ID
ISSTRA	x		SEG REFER strata ID
ISUBS	x		SEG REFER substrata ID
ISSEG	x		SEG REFER Segment ID
ISTRN	x		SEG REFER Training Segment Indicator
ISRDAY	x		SEG REFER relative acquisition day
ISSANG	x		SEG REFER sun angle
ISPCC	x		SEG REFER % cloud cover
ISPCSW	x	x	Per cent spring wheat *10
ISPCWW	x	x	Percent winter wheat *10
SFTRL		x	Trailer ID "ZZZZ"

ERROR EXITS

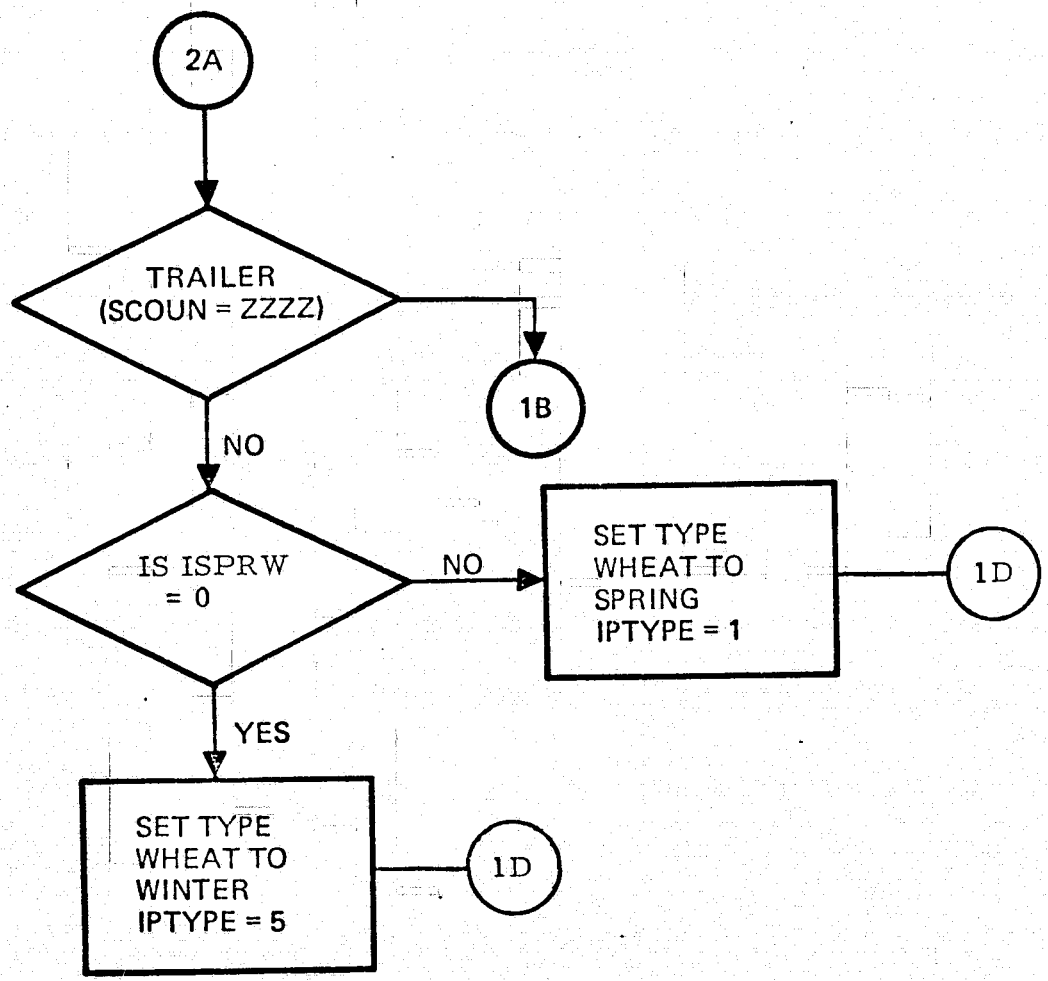
05 Invalid Country Input xxxx CROPWIND xxxx SEGREFEL xxxx

Country should be equal from all three sources

SUBROUTINE AREADF BLOCK DIAGRAM (PAGE 1 OF 2)



SUBROUTINE AREAD BLOCK DIAGRAM (PAGE 2 OF 2)



SUBROUTINE AWRTA

Purpose:

This subroutine writes the records of the Data Acquisition File. These records include the Header, data records and trailer.

Input:

The inputs to this routine come from the IBLK and the ABLK. The variables are shown in Table 5.

Output:

The outputs from this routine consist of records written to the "ACQUISI" file and variables updated in the ABLK. The variables are shown in Table 5 and the record formats are shown in Table 5.

Calling Sequence:

Call AWRTA.

Subroutines Called:

None.

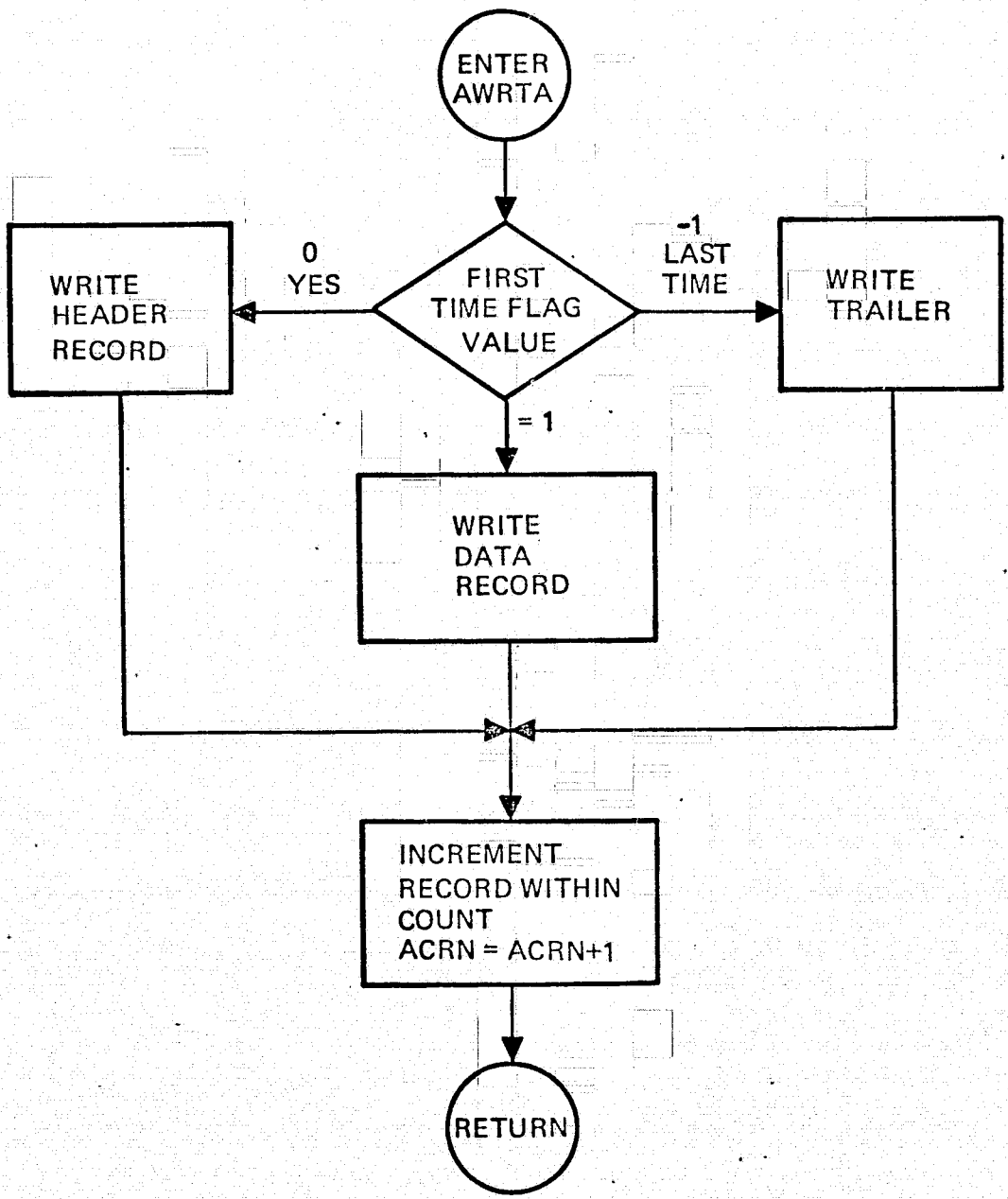
Processing:

A functional flow diagram is attached.

Table 5. Routine AWRTA Inputs and Outputs

Variable Name	Set	Use	Description
IPFTF		x	First time flag 0=first time; 1= subsequent times; 1=last time
IPNEW		x	Number of crop windows
XICOUN		x	Input country
IACASE		x	Case number of acquisition file
IAFILE		x	Fortran unit number of acquisition file
AFNAM		x	File name
ABLANK		x	Record padding
IACNT		x	Total number accesses per window
IAPNT		x	Pointer to SEG REFER acquisitions selected
ACRN	x		Current record number
ICCASE		x	Case number of cropwind file
ISNDAY		x	Number days in study
ISCASE		x	Case number from REFER file
KX	x		Contains list of acquisition dates for writing out on acquisition file

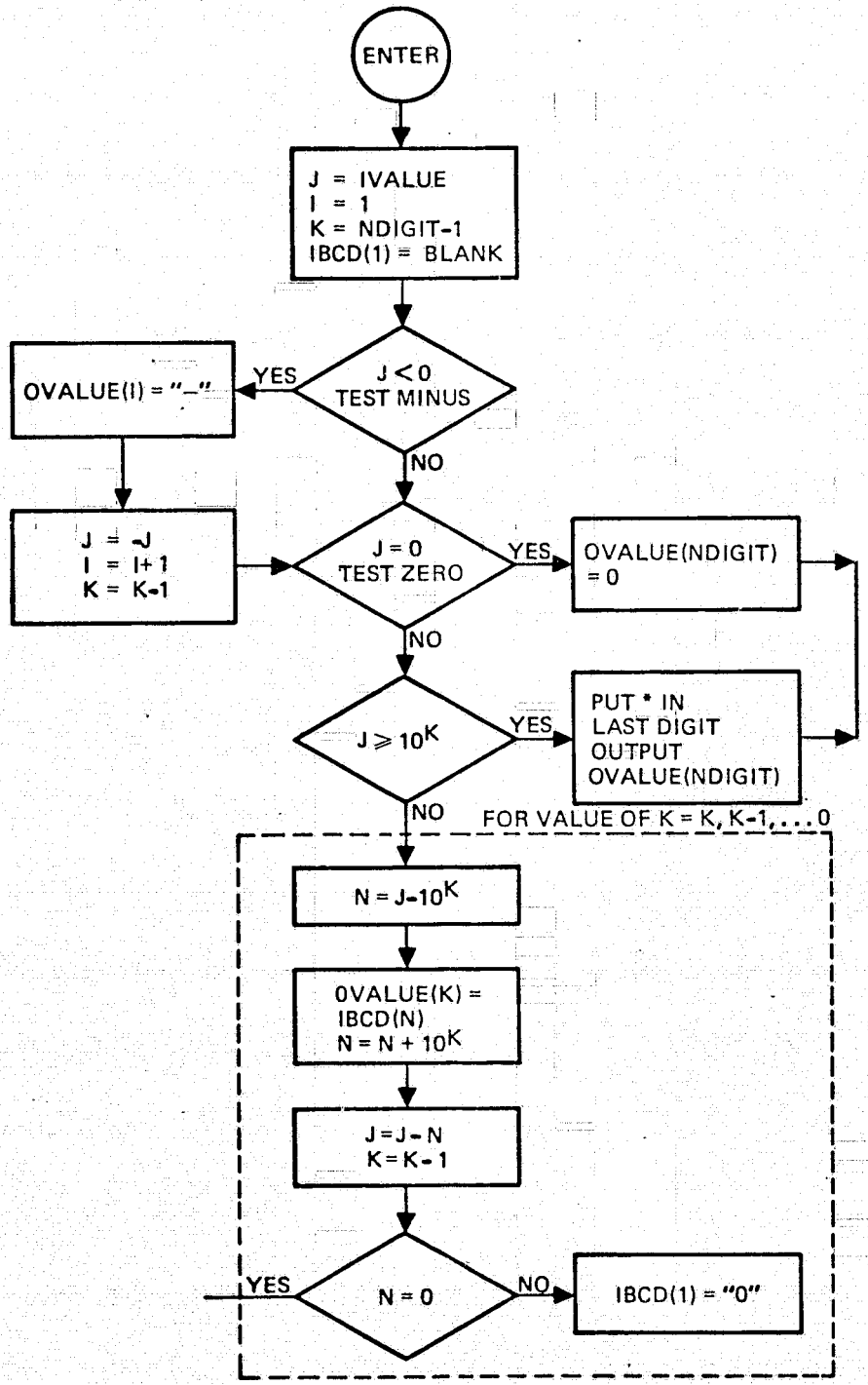
SUBROUTINE AWRTA BLOCK DIAGRAM



SUBROUTINE BINBCD BLOCK DIAGRAM

PURPOSE: TO CONVERT AN INTEGER TO DISPLAY CODE CALLING SEQUENCE:

CALL BINBCD (IVAL, NDIG, IOUT)
IVAL – BINARY INTEGER
NDIG – NO. OF DIGITS TO OUTPUT
IOUT – ORIGIN OF THE OUTPUT CHARACTER ARRAY



VARIABLES

IBCD ARAY	0
	1
	2
	⋮
	9
ITEN	1
	10
	100
	⋮
	100000

I = SUBSCRIPT INTO OUTPUT ARRAY

J = CURRENT VALUE OF INPUT INTEGER

K = NO. DIGITS REMAINING TO PROCESS–1

N = CURRENT DIGIT VALUE

SUBROUTINE MAIN

Purpose:

This subroutine is the main control routine for the SACS program. It is entered by the operating system each time the SACS program is executed.

Input:

The inputs to this routine are provided via common block supplied by the routines it calls. Since this routine is a control routine the inputs consist mainly of flags and counters.

Output:

The outputs from this routine consist of the setting of flags.

Calling Sequence:

Main Fortran routine.

Subroutines Collect

This routine calls the following routines.

a. AINI1 - initialization routine to read input cards and process file headers

b. AREAD - Reach data records from both crop windows and segment reference data file

c. APRPT - Prints the daily processing load report.

d. AFIND - Selects segment reference file acquisitions based upon crop windows and specified input parameters.

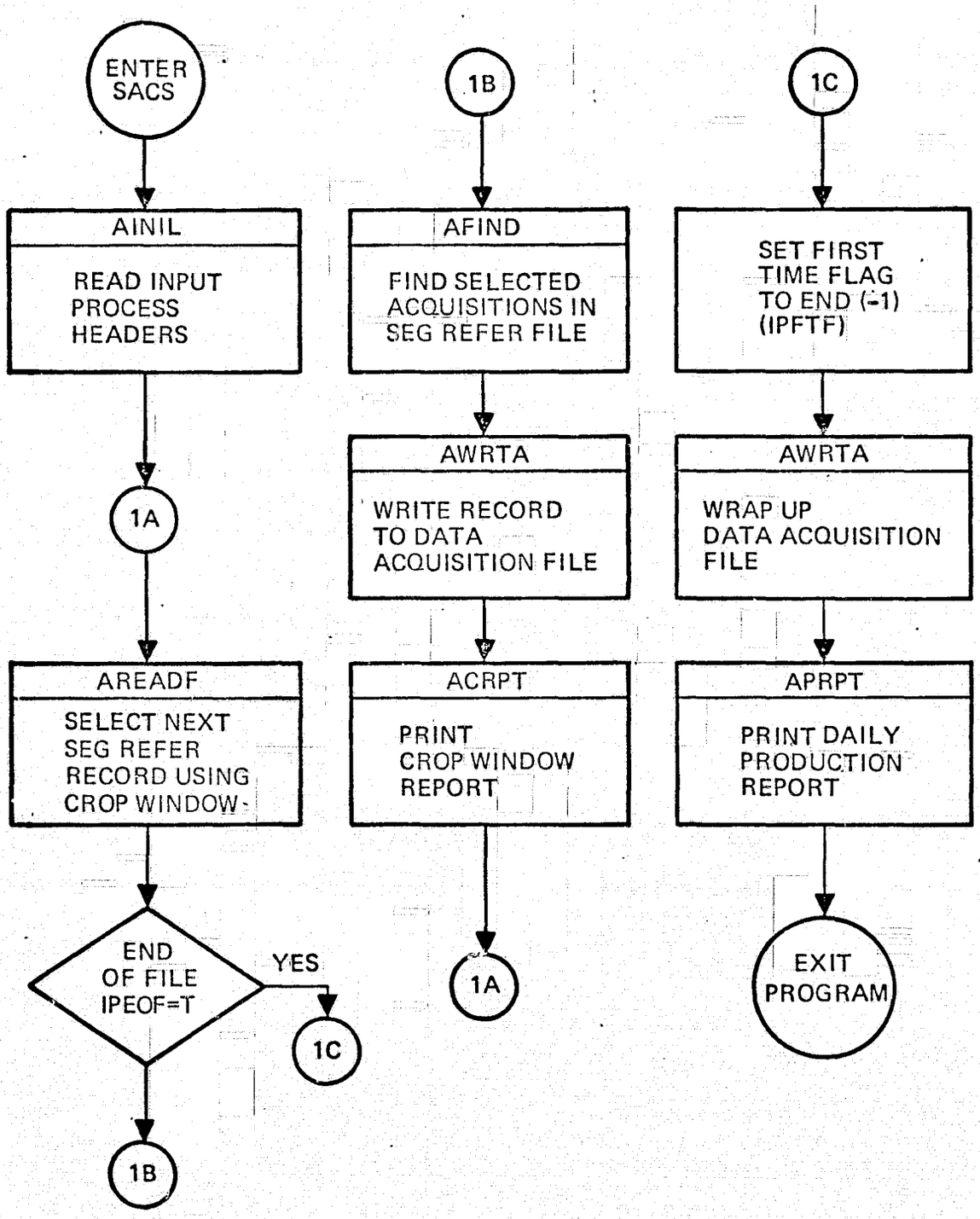
e. AWRTA - writes the data and trailer records of the data acquisition file.

f. ACRPT - writes the Segment Data Acquisition report (crop window)

Processing:

A functional flow diagram is attached.

SUBROUTINE MAIN BLOCK DIAGRAM



Subroutine SORTAG

Purpose:

Sorts Array A into increasing order from A(II) to A(JJ). The array tag is permuted the same as Array A. This routine processes arrays that can fit in core only. (A CDC library routine.)

Calling Sequence:

```
CALL SORTAG(A, II, JJ, TAG)
```

PART V

SUBROUTINE LISTINGS

```
000001            SUBROUTINE ABARF(IERR)                                        ABARF
000002      C*****************************************************************ABARF
000003      C                                                                   ABARF
000004      C ROUTINE      .ABARF                                               ABARF
000005      C                                                                   ABARF
000006      C PURPOSE      THIS ROUTINE PRINTS ERROR MESSAGES AND ABORTS THE    ABARF
000007      C              PROGRAM IF THE ERROR IS FATAL                        ABARF
000008      C                                                                   ABARF
000009      C COMMON       IBLK-MANUAL INPUT VARIABLES                          ABARF
000010      C              PBLK-PROGRAM FLAGS,COUNTERS,CONSTANTS                ABARF
000011      C              SBLK-SEGREFER FILE RECORD                            ABARF
000012      C              ABLK-ACQUISI FILE RECORD                             ABARF
000013      C              CBLK-CROPWIND FILE RECORD                            ABARF
000014      C                                                                   ABARF
000015      C LINKAGE      CALL ABARF(IERR)                                     ABARF
000016      C                   IERR= ERROR ID                                  ABARF
000017      C                                                                   ABARF
000018      C ROUTINES CALLED   NONE                                            ABARF
000019      C                                                                   ABARF
000020      C LOCAL VARIABLES                                                   ABARF
000021      C                                                                   ABARF
000022      C                                                                   ABARF
000023      C*****************************************************************ABARF
000024      C                                                                   ABARF
000025      C                                                                   IBLK
000026      C COMMON BLOCK IBLK-MANUAL INPUT PARAMETERS CONSTANTS               IBLK
000027      C                                                                   IBLK
000028            COMMON /IBLK/                                                 IBLK
000029           1XITTL(12),XISANG,XICOUN,IWTTL(16),                            JIM
000030           2ICID(3),IACASE,ISCASE,IWCASE,IPCC,ISANG,IREG,IZONE,IIFILE,IOFILE,  IBLK
000031           3ISFILE,ICFILE,IAFILE,INLINE,                                  IBLK
000032           4IPRPT,ICRPT                                                   IBLK
000033            LOGICAL IPRPT,ICRPT                                           IBLK
000034      C NAME     DESCRIPTION                                              IBLK
000035      C XITTL    REPORT CASE ORIENTED PAGE TITLE FROM INPUT(12A4)         IBLK
000036      C XISANG   MINIMUM SUN ANGLE IN RADIANS FROM INPUT                  IBLK
000037      C XICOUN   COUNTRY ID FROM INPUT (A4)                               IBLK
000038      C IWTTL    CROP WINDOW TITLES.  TITLE FOR EACH WINDOW OCCUPIES 8 WORDS  JIM
000039      C            AT 2 CHARACTERS PER WORD.                              JIM
000040      C ICID     COLUMNS 75-80 OF INPUT CARD 3A2                          IBLK
000041      C IACASE   ACQUISI FILE CASE NO. FROM INPUT                         IBLK
000042      C ISCASE   SEGREFER CASE NO. FROM INPUT                             IBLK
000043      C IWCASE   CROPWIND CASE NO. FROM INPUT                             IBLK
000044      C IPCC     MAXIMUM PERCENT CLOUD COVER * 10-TO BE USED IN SEGMENT SELECT  IBLK
000045      C ISANG    MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (I4)         IBLK
000046      C IREG     REGION ID FROM INPUT (I3)                                IBLK
000047      C IZONE    ZONE ID   FROM INPUT (I3)                                IBLK
000048      C IIFILE   LOGICAL UNIT NO. FOR CARD INPUT                          IBLK
000049      C IOFILE   LOGICAL UNIT NO. FOR REPORT OUTPUT                       IBLK
000050      C ISFILE   LOGICAL UNIT NO. FOR SEGREFER DATA FILE                  IBLK
000051      C ICFILE   LOGICAL UNIT NO. FOR CROPWINDOW FILE                     IBLK
000052      C IAFILE   LOGICAL UNIT NO. FOR ACQUISITION FILE                    IBLK
000053      C INLINE   NO. LINES PER PAGE                                       IBLK
000054      C IPRPT    PROCESSING LOAD REPORT FLAG (T=PRINT REPORT)             IBLK
000055      C ICRPT    ACQUISITION (CROP WINDOW) REPORT FLAG (T=PRINT REPORT)   IBLK
000056      C                                                                   PBLK
000057      C COMMON BLOCK PBLK-PROGRAM COUNTERS AND CONSTANTS                  PBLK
000058      C                                                                   PBLK
```

```
000059              COMMON /PBLK/                                          PBLK
000060             1XPWRK(150),IPFTF,IPTYPE,IPERR,IBCD(15),IPWRK(150),ITEN(5),  PBLK
000061             2IPEFLG(13),IPEOF                                      PBLK
000062              LOGICAL IPEOF
000063        C                                                            PBLK
000064        C NAME     DESCRIPTION                                       PBLK
000065        C XPWRK    FLOATING POINT WORK AREA-DIMENSION 150            PBLK
000066        C IPFTF    FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST)  PBLK
000067        C IPTYPE   TYPE WHEAT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)   JTM
000068        C IPERR    ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT)  PBLK
000069        C IBCD     HOLLERITH ARRAY/0/1/2/3/4/5/6/7/8/9/ /-/0/*/,/    PBLK
000070        C ITEN     POWERS OF TEN (1,10,100,1000,10000)               PBLK
000071        C IPWRK    INTEGER WORK AREA-DIMENSION 150                   PBLK
000072        C IPEOF    END OF FILE FLAG (TRUE=END FILE)                  PBLK
000073        C IPEFLG   FATAL ERROR FLAG FOR ERROR 1-13 (T=FATAL)         PBLK
000074        C                                                            ABLK
000075        C COMMON BLOCK ABLK-ACQUISI FILE AND REPORT VARIABLES        ABLK
000076        C                                                            ABLK
000077              COMMON/ABLK/                                           ABLK
000078             1ACRN,AFNAM(2),AFTRL,ANACQ(426),                        ABLK
000079             2IACNT(4),IATOT,IAPNT(25,4),IAPAG,IANLIN,IACNTL(4)      JTM
000080        C NAME     DESCRIPTION                                       ABLK
000081        C ACRN     CURRENT RECORD NO.                                ABLK
000082        C AFNAM    LABEL FILE NAME*ACQUISI *                         ABLK
000083        C AFTRL    TRAILER ID *ZZZZ*  A4                             ABLK
000084        C ANACQ    NO. OF ACQUISITIONS SELECTED FOR EACH OF 426 DAYS ABLK
000085        C IACNT    TOTAL NO. OF ACCESSES FOR EACH OF 4 WINDOWS       JTM
000086        C IATOT    TOTAL NO. OF ACQUISITIONS FOR ALL 4 WINDOWS       ABLK
000087        C   IAPNT     INDEX TO SEGREFER ACQUISITION (1-150) THAT IDENTIFIES  ABLK
000088        C            A SELECTED SEGMENT. IAPNT(I,J)= WINDOW 1 THROUGH 4.  ABLK
000089        C IAPAG    ACQUISITION REPORT CURRENT PAGE NUMBER            ABLK
000090        C IANLIN   ACQUISITION REPORT NO. LINES REMAINING CURRENT PAGE  ABLK
000091        C IACNTL   NO. OF LEGAL ACQUISITIONS SELECTED FOR EACH OF 4 WINDOWS  JTM
000092        C                                                            CBLK
000093        C COMMON BLOCK CBLK-CROPWIND HEADER AND DATA                 CBLK
000094        C                                                            CBLK
000095              COMMON /CBLK/                                          CBLK
000096             1CCRN,CFNAM1(2),CFNAM2(2),CCOUN,                        CBLK
000097             2ICCASE,ICWIND,ICREG,ICZONE,ICSTRA,ICSUBS,ICLOW(8),ICHI(8)  CBLK
000098        C NAME     DESCRIPTION                                       CBLK
000099        C CCRN     CURRENT CROPWIND RECORD NO.                       CBLK
000100        C CFNAM1   FILE NAME *CROPWIND*(2A4)                         CBLK
000101        C CFNAM2   LABEL FILE NAME-MUST MATCH CFNAM1                 CBLK
000102        C CCOUN    CROPWIND COUNTRY ID (A4)                          CBLK
000103        C ICCASE   CROPWIND CASE NO. FROM LABEL                      CBLK
000104        C ICWIND   NO. OF CROP WINDOWS (8)                           CBLK
000105        C ICREG    CROPWIND REGION ID                                CBLK
000106        C ICZONE   CROPWIND ZONE ID                                  CBLK
000107        C ICSTRA   CROPWIND STRATA ID                                CBLK
000108        C ICSUBS   CROPWIND SUBSTRATA ID                             CBLK
000109        C ICLOW    START DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950    CBLK
000110        C ICHI     STOP  DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950    CBLK
000111        C                                                            SBLK
000112        C COMMON BLOCK SBLK-SEGREFER FILE HEADER AND DATA RECORD VARIABLES  SBLK
000113        C                                                            SBLK
000114              COMMON/SBLK/                                           SBLK
000115             1SCRN,SFNAM1(2),SFNAM2(2),SFTRL,SCOUN,SPCSW,SPCWW,SSANG(150),  SBLK
000116             2ISSCAS,ISDATE,ISNDAY,ISNAC,ISREG,ISZONE,ISSTRA,        SBLK
000117             3ISUBS, ISSEC,ISTRN,ISRDAY(150),ISPCC(150),NVEH
000118             4,IORDER(150),IKDAY(150)                                      *NEW
```

```
000119          C NAME    DESCRIPTION                                                    SRLK    **-1
000120          C SCRN    SEGREFER CURRENT RECORD NO.                                    SRLK
000121          C SFNAM1  SEGREFER FILE NAME PROTOTYPE*SEGREFER*                         SRLK
000122          C SFNAM2  SEGREFER FILE NAME FROM LABEL                                  SRLK
000123          C SFTRL   TRAILER ID *ZZZZ*                                              SRLK
000124          C SCOUN   SEGREFER COUNTRY ID (F6.2)                                     SRLK
000125          C SPCSW   PERCENT SPRING WHEAT(F6.2)                                     SRLK
000126          C SPCWW   PERCENT WINTER WHEAT (F6.2)                                    JTH
000127          C SSANG   SUN ANGLE(IN RADIAND) FOR EACH OF 150 SEGMENTS                 SRLK
000128          C ISSCAS  SEGREFER CASE NO. FROM LABEL                                   SRLK
000129          C ISDATE  SEGREFER REFERENCE DATE-ZULU-                                  SRLK
000130          C ISNDAY  NO. DAYS IN STUDY (1-426)                                      SRLK
000131          C ISNAC   NO. ACQUISITIONS IN RECORD (0-150)                             SRLK
000132          C ISREG   SEGREFER REGION ID                                             SRLK
000133          C ISZONE  SEGREFER ZONE  ID                                              SRLK
000134          C ISSTRA  SEGREFER STRATA ID                                             SRLK
000135          C ISUBS   SEGREFER SUBSTRATA ID                                          SRLK
000136          C ISSEG   SEGREFER SEGMENT ID                                            SRLK
000137          C ISTRN   TRAINING SEGMENT INDICATOR(0=NORMAL,1=TRAINING)                SRLK
000138          C ISRDAY  ACQUISITION DAY FOR 150 ACQUISITIONS-ZULU-                     SRLK
000139          C         IF ZERO, END OF ALL ACQUISITIONS IN RECORD                     SRLK
000140          C ISPCC   PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS                  SRLK
000141          C NVEH    NO OF VEHICLES IN THE SWATH TABLE
000142          C IORDER  ORIGINAL ORDER OF ACCESS DATES AS READ FROM SEQ. REF. FILE     JTM
000143          C                                                                        JTM
000144               COMMON  /INPERR/  INPERR                                            JTM
000145          C INPERR  INPUT ERROR FLAG. SET = 1 IN ABARF IF THERE ARE ANY FATAL      JTM
000146          C              ERRORS DETECTED ON THE INPUT CARDS BY AINIL               JTM
000147          C                                                                        JTM
000148          C                                                                        JTM
000149          C                  IF THERE WERE ANY INPUT ERRORS, ABORT RUN.            JTM
000150               IF ( IERR .LT. 0 )  GO TO 210                                       JTM
000151          C                                                                        JTM
000152          C              GO TO WRITE STATEMENT FOR ERROR                           ABARF
000153               IF(IERR.GT.13) GOTO 210                                             ABARF
000154               GO TO (10,20,30,40,50,60,70,80,90,100,110,120,130),IERR             SFTI
000155          C                                                                        ABARF
000156           10  WRITE (IOFILE,11) CFNAM2                                           ABARF
000157               GOTO 200                                                            ABARF
000158           11  FORMAT (6H001   ,37HCROPWIND FILE HAS INVALID LABEL NAME ,2A4)      ABARF
000159          C                                                                        ABARF
000160           20  WRITE (IOFILE,21) SFNAM2                                           ABARF
000161               GOTO 200                                                            ABARF
000162           21  FORMAT (6H002   ,37HSEGREFER FILE HAS INVALID LABEL NAME  ,2A4)     ABARF
000163          C                                                                        ABARF
000164           30  WRITE (IOFILE,31)IWCASE,ICCASE                                     ABARF
000165               GO TO 190                                                           JTM
000166           31  FORMAT (6H003   ,20HCROPWIND INPUT CASE ,I3,25H NOT EQUAL TO LABEL ABARF
000167              C CASE ,I3)                                                          ABARF
000168          C                                                                        ABARF
000169           40  WRITE(IOFILE,41)ISCASE,ISSCAS                                      ABARF
000170               GO TO 190                                                           JTM
000171           41  FORMAT(6H004   ,20HSEGREFER INPUT CASE ,I3,25H NOT EQUAL TO LABEL  ABARF
000172              CCASE ,I3)                                                           ABARF
000173          C                                                                        ABARF
000174           50  WRITE(IOFILE,51)XICOUN,CCOUN,SCOUN                                 ABARF
000175               GOTO 200                                                            ABARF
000176           51  FORMAT (6H005   ,22HINVALID COUNTRY-INPUT=,A4,10H CROPWIND=,A4,10HABARF
000177              C SEGREFER=,A4)                                                      ABARF
000178          C                                                                        ABARF
```

```
000179          60   WRITE(IOFILE,61)                                                  ABARF
000180               GOTO 200                                                          ABARF
000181          61   FORMAT (6H0006   ,34HNO DATA SELECTED FOR ACQUISI FILE.)          ABARF
000182          C                                                                      ABARF
000183          70   WRITE(IOFILE,71)IPERR,ISNDAY,SCRN                                 ABARF
000184               GOTO 200                                                          ABARF
000185          71   FORMAT (6H0007   ,55HSEGREFER ACQUISITION DAY TOO LARGE ,I4,6H, MAXABARF
000186              C=,I3,09H, RECORD=,F5.0)                                           ABARF
000187          C                                                                      ABARF
000188          80   WRITE(IOFILE,81)ICID                                              ABARF
000189               GO TO 190                                                         JTM
000190          81   FORMAT (6H0008   ,39HINPUT CARDS INVALID OR OUT OF SEQUENCE ,3A2) ABARF
000191          C                                                                      ABARF
000192          90   WRITE(IOFILE,91)ISANG                                             ABARF
000193               GO TO 190                                                         JTM
000194          91   FORMAT (6H0009   ,52HINPUT SUN ANGLE LESS THAN ZERO OR GREATER THANJTM
000195              C 9000,,16)                                                        JTM
000196          C                                                                      ABARF
000197          100  WRITE(IOFILE,101)IPCC                                             ABARF
000198               GO TO 190                                                         JTM
000199          101  FORMAT (6H0010   ,55HPERCENT CLOUD COVER LESS THAN 0 OR GREATER THAJTM
000200              CN 10000, ,16)                                                     JTM
000201          C                                                                      ABARF
000202          110  WRITE(IOFILE,111)ISNDAY                                           ABARF
000203               GOTO 200                                                          ABARF
000204          111  FORMAT(6H0011    ,19HNO. DAYS IN STUDY =,I4,13H, EXCEEDS 426)     ABARF
000205          C                                                                      ABARF
000206           120 WRITE(IOFILE,121)ISZONE,ISSTRA,ISUBS,ISSEG                        SFII
000207               GOTO 200                                                          ABARF
000208          121  FORMAT (6H0012   ,34HCROPWINDOW IS MISSING RECORD ZONE=,I3,08H STRAABARF
000209              CTA=,I4,11H SUBSTRATA=,I4,09H SEGMENT=,I4)                         ABARF
000210           130 WRITE(IOFILE,131)SCRN,ISZONE,ISSTRA,ISUBS,ISSEG                   SFII
000211               IANLIN= IANLIN - 2                                                SFII
000212               GOTO 200                                                          ABARF
000213          131  FORMAT(6H0013    ,16HSEGREFER RECORD ,F6.0,41H SKIPPED BECAUSE PERCEABARF
000214              CNT NOT AT ZERO-ZONE=,I3,9H, STRATA=,I4,12H, SUBSTRATA=,I4,10H, SEGMEABARF
000215              CENT=,I4)                                                          ABARF
000216          C                                                                      ABARF
000217          190  INPERR= 1                                                         JTM
000218               RETURN                                                            JTM
000219          C              TEST FATAL OR NON-FATAL ERROR                          ABARF
000220          C                                                                      ABARF
000221          200  IF(IPFFLG(IERR) .NE. 0) GO TO 210
000222               RETURN                                                            ABARF
000223          C              FATAL ERROR  EXIT                                      ABARF
000224          210  WRITE (IOFILE,211)SCRN,CCRN,ACRN                                  ABARF
000225               STOP                                                              ABARF
000226          211  FORMAT (6H0099   ,33HPROGRAM ABORTED. SEGREFER RECORD=,F6.0,16H, CRABARF
000227              COPWIND RECORD=,F6.0,17H, ACQUISI RECORD=,F6.0)                    ABARF
000228               END                                                               ABARF
```

```
* FLT ACRPT,1,760426, 50345     , 1

000001            SUBROUTINE ACRPT                                                ACRPT
000002          C*********************************************************************ACRPT
000003          C                                                                   ACRPT
000004          C ROUTINE       ACRPT                                               ACRPT
000005          C                                                                   ACRPT
000006          C PURPOSE       PRINT ACQUISITION REPORT-ONE RECORD PER ENTRY       ACRPT
000007          C                                                                   ACRPT
000008          C LINKAGE       CALL ACRPT                                          ACRPT
000009          C                                                                   ACRPT
000010          C ROUTINES CALLED                                BINBCD-CONVERT BINARY TOACRPT
000011          C                                                                   ACRPT
000012          C LOCAL VARIABLES                                                   ACRPT
000013          C               NMAX, NO LINES TO PRINT                             ACRPT
000014          C               I,    WINDOW INDEX                                  ACRPT
000015          C               J,    ACQUISITION WITHIN LINE INDEX                 ACRPT
000016          C               K,    DO LOOP INDEX                                 ACRPT
000017          C               ISUB ,CURRENT ACQUISITION INDEX FOR WINDOW ONE      ACRPT
000018          C                                                                   ACRPT
000019          C*********************************************************************ACRPT
000020          C                                                                   ACRPT
000021          C                                                                   IRLK
000022          C COMMON BLOCK IRLK-MANUAL INPUT PARAMETERS CONSTANTS               IRLK
000023          C                                                                   IRLK
000024               COMMON /IRLK/                                                  IRLK
000025              1XTITL(12),XISANG,XICOUN,IWTTL(16),                             JIM
000026              2ICID(3),IACASE,ISCASE,IWCASE,IPCC,ISANG,IREG,IZONE,IIFILE,IOFILE, IRLK
000027              3ISFILE,ICFILE,IAFILE,INLINE,                                   IRLK
000028              4IPRPT,ICRPT                                                    IRLK
000029               LOGICAL IPRPT,ICRPT                                            IRLK
000030          C NAME     DESCRIPTION                                             IRLK
000031          C XTITL    REPORT CASE ORIENTED PAGE TITLE FROM INPUT(12A4)        IRLK
000032          C XISANG   MINIMUM SUN ANGLE IN RADIANS FROM INPUT                 IRLK
000033          C XICOUN   COUNTRY ID FROM INPUT (A4)                              IRLK
000034          C IWTTL    CROP WINDOW TITLES.  TITLE FOR EACH WINDOW OCCUPIES 8 WORDS JIM
000035          C            AT 2 CHARACTERS PER WORD.                             JIM
000036          C ICID     COLUMNS 75-80 OF INPUT CARD 3A3                         IRLK
000037          C IACASE   ACQUISI FILE CASE NO. FROM INPUT                        IRLK
000038          C ISCASE   SEGREFER CASE NO. FROM INPUT                            IRLK
000039          C IWCASE   CROPWIND CASE NO. FROM INPUT                            IRLK
000040          C IPCC     MAXIMUM PERCENT CLOUD COVER * 10-TO BE USED IN SEGMENT SELECT IRLK
000041          C ISANG    MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (I4)        IRLK
000042          C IREG     REGION ID FROM INPUT (I5)                               IRLK
000043          C IZONE    ZONE ID   FROM INPUT (I3)                               IRLK
000044          C IIFILE   LOGICAL UNIT NO. FOR CARD INPUT                         IRLK
000045          C IOFILE   LOGICAL UNIT NO. FOR REPORT OUTPUT                      IRLK
000046          C ISFILE   LOGICAL UNIT NO. FOR SEGREFER DATA FILE                 IRLK
000047          C ICFILE   LOGICAL UNIT NO. FOR CROPWINDOW FILE                    IRLK
000048          C IAFILE   LOGICAL UNIT NO. FOR ACQUISITION FILE                   IRLK
000049          C INLINE   NO. LINES PER PAGE                                      IRLK
000050          C IPRPT    PROCESSING LOAD REPORT FLAG (T=PRINT REPORT)            IRLK
000051          C ICRPT    ACQUISITION (CROP WINDOW) REPORT FLAG (T=PRINT REPORT)  IRLK
000052          C                                                                  PBLK
000053          C COMMON BLOCK PBLK-PROGRAM COUNTERS AND CONSTANTS                 PBLK
000054          C                                                                  PBLK
000055               COMMON /PBLK/                                                 PBLK
000056              1XPWRK(150),IPFTF,IPTYPE,IPERR,IBCD(15),IPWRK(150),ITEN(5),    PBLK
000057              2IPLFLG(13),IPTGF                                              PBLK
000058               LOGICAL IPTGF
```

```
000059          C                                                                       PBLK
000060          C NAME    DESCRIPTION                                                   PBLK
000061          C XPWRK   FLOATING POINT WORK AREA-DIMENSION 150                        PBLK
000062          C IPFTF   FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST)           PBLK
000063          C IPTYPE  TYPE WHEAT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)               JIM
000064          C IPERR   ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT)          PBLK
000065          C IHOL    HOLLERITH ARRAY/0/1/2/3/4/5/6/7/8/9/ /-/0/*/,/                PBLK
000066          C ITEN    POWERS OF TEN (1,10,100,1000,10000)                           PBLK
000067          C IPWRK   INTEGER WORK AREA-DIMENSION 150                               PBLK
000068          C IPEOF   END OF FILE FLAG (TRUE=END FILE)                              PBLK
000069          C IPEFLG  FATAL ERROR FLAG FOR ERROR 1-13 (T=FATAL)                     PBLK
000070          C                                                                       APLK
000071          C COMMON BLOCK ABLK-ACQUISI FILE AND REPORT VARIABLES                   APLK
000072          C                                                                       APLK
000073                COMMON/ABLK/                                                      APLK
000074               1ACRN,AFNAM(7),AFTRL,ANACQ(426),                                   APLK
000075               2IACNT(4),IATOT,IAPNT(25,4),IAPAG,IANLIN,IACNTL(4)                 JIM
000076          C NAME    DESCRIPTION                                                   ABLK
000077          C ACRN    CURRENT RECORD NO.                                            ABLK
000078          C AFNAM   LABEL FILE NAME*ACQUISI *                                     ABLK
000079          C AFTRL   TRAILER ID *ZZZZ*  A4                                         APLK
000080          C ANACQ   NO. OF ACQUISITIONS SELECTED FOR EACH OF 426 DAYS             APLK-
000081          C IACNT   TOTAL NO. OF ACCESSES FOR EACH OF 4 WINDOWS                   JIM
000082          C IATOT    TOTAL NO. OF ACQUISITIONS FOR ALL 4 WINDOWS                  APLK
000083          C   IAPNT    INDEX TO SEGREFER ACQUISITION (1-150) THAT IDENTIFIES      ABLK
000084          C            A SELECTED SEGMENT. IAPNT(I,J)= WINDOW 1 THROUGH 4.        APLK
000085          C IAPAG    ACQUISITION REPORT CURRENT PAGE NUMBER                       APLK
000086          C IANLIN   ACQUISITION REPORT NO. LINES REMAINING CURRENT PAGE          APLK
000087          C IACNTL  NO. OF LEGAL ACQUISITIONS SELECTED FOR EACH OF 4 WINDOWS      JIM
000088          C                                                                       SBLK
000089          C COMMON BLOCK SBLK-SEGREFER FILE HEADER AND DATA RECORD VARIABLES       SBLK
000090          C                                                                       SBLK
000091                COMMON/SBLK/                                                      SBLK
000092               1SCRN,SFNAM1(2),SFNAM2(2),SFTRL,SCOUN,SPCSW,SPCWW,SSANG(150),      SBLK
000093               2155CAS,ISDATE,ISNDAY,ISNAC,ISREG,ISZONE,ISSTRA,                   SBLK
000094               3ISUBS, ISSEG,ISTRN,ISRDAY(150),ISPCC(150),NVEH                    SBLK
000095               4,IOROER(150),IKDAY(150)                                           *NEW
000096          C NAME    DESCRIPTION                                                   SBLK  **-1
000097          C SCRN    SEGREFER CURRENT RECORD NO.                                   SBLK
000098          C SFNAM1  SEGREFER FILE NAME PROIOTYPE*SEGREFER*                        SBLK
000099          C SFNAM2  SEGREFER FILE NAME FROM LABEL                                 SBLK
000100          C SFTRL   TRAILER ID *ZZZZ*                                             SBLK
000101          C SCOUN   SEGREFER COUNTRY ID (F6.2)                                    SBLK
000102          C SPCSW   PERCENT SPRING WHEAT(F6.2)                                    SBLK
000103          C SPCWW   PERCENT WINTER WHEAT (F6.2)                                   JIM
000104          C SSANG   SUN ANGLE(IN RADIANS) FOR EACH OF 150 SEGMENTS                SBLK
000105          C ISSCAS  SEGREFER CASE NO. FROM LABEL                                  SBLK
000106          C ISDATE  SEGREFER REFERENCE DATE-ZULU-                                 SBLK
000107          C ISNDAY  NO. DAYS IN STUDY (1-426)                                     SBLK
000108          C ISNAC   NO. ACQUISITIONS IN RECORD (0-150)                            SBLK
000109          C ISREG   SEGREFER REGION ID                                            SBLK
000110          C ISZONE  SEGREFER ZONE   ID                                            SBLK
000111          C ISSTRA  SEGREFER STRATA ID                                            SBLK
000112          C ISUBS   SEGREFER SUBSTRATA ID                                         SBLK
000113          C ISSEG   SEGREFER SEGMENT ID                                           SBLK
000114          C ISTRN   TRAINING SEGMENT INDICATOR(0=NORMAL,1=TRAINING)               SBLK
000115          C ISRDAY  ACQUISITION DAY FOR 150 ACQUISITIONS-ZULU-                    SBLK
000116          C         IF ZERO, END OF ALL ACQUISITIONS IN RECORD                    SBLK
000117          C ISPCC   PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS                 SBLK
000118          C NVEH    NO OF VEHICLES IN THE SWATH TABLE
```

```
000119          C IORDER  ORIGINAL ORDER OF ACCESS DATES AS READ FROM SEQ. REF. FILE    JIM
000120          C                                                                       CBLK
000121          C COMMON BLOCK CBLK-CROPWIND HEADER AND DATA                            CBLK
000122          C                                                                       CBLK
000123                COMMON /CBLK/                                                     CBLK
000124               1CCRN,CFNAM1(2),CFNAM2(2),CCOUN,                                   CBLK
000125               2ICCASE,ICWIND,ICREG,ICZONE,ICSTRA,ICSUBS,ICLOW(8),ICHI(8)         CBLK
000126          C NAME    DESCRIPTION                                                   CBLK
000127          C CCRN    CURRENT CROPWIND RECORD NO.                                   CBLK
000128          C CFNAM1  FILE NAME.*CROPWIND*(2A4)                                     CBLK
000129          C CFNAM2  LABEL FILE NAME-MUST MATCH CFNAM1                             CBLK
000130          C CCOUN   CROPWIND COUNTRY ID (A4)                                      CBLK
000131          C ICCASE  CROPWIND CASE NO. FROM LABEL                                  CBLK
000132          C ICWIND  NO. OF CROP WINDOWS (8)                                       CBLK
000133          C ICREG   CROPWIND REGION ID                                            CBLK
000134          C ICZONE  CROPWIND ZONE ID                                              CBLK
000135          C ICSTRA  CROPWIND STRATA ID                                            CBLK
000136          C ICSUBS  CROPWIND SUBSTRATA ID                                         CBLK
000137          C ICLOW   START DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950                CBLK
000138          C ICHI    STOP  DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950                CBLK
000139                DIMENSION ILLEG(11),JIMAG(61)                                     ACRPT
000140                EQUIVALENCE (IPMRK,JIMAG)                                         ACRPT
000141                DATA ILLEG/1H1,1HL,1HE,1H ,1HI,1HL,1HL,1HE,1HG,1HA,1HL/           ACRPT
000142                DIMENSION  NONE(4)                                                JIM
000143                DATA  NONE  / 1HN,1HO,1HN,1HE /                                   JIM
000144          C                                                                       JIM
000145          C               TEST LAST TIME ENTERED,IF YES, PRINT STATISTICS         ACRPT
000146                IF(IPETF.LT.0) GOTO 50                                            ACRPT
000147          C               TEST REPORT WANTED                                      ACRPT
000148                IF(.NOT. ICRPT)GO TO 60                                           SE11
000149          C               COMPUTE LINE COUNT FROM MAX NO. SEGMENTS SELECTED       ACRPT
000150                NMAX=1                                                            ACRPT
000151                DO 5  I=1,4,1                                                     ACRPT
000152                J= (IACNTL(I)+3)/4                                                JIM
000153                IF (NMAX.GE.J) GOTO 5                                             ACRPT
000154                NMAX=J                                                            ACRPT
000155           5    CONTINUE                                                          ACRPT
000156          C               TEST PAGE HEADERS REQUIRED (NO. LINES ON PAGE TOO SMALL) ACRPT
000157                IANLIN=IANLIN-NMAX-1                                              ACRPT
000158                IF(IANLIN.GE.0) GOTO 10                                           ACRPT
000159          C               PRINT PAGE HEADERS,INCREMENT PAGE, RESET LINE COUNT     ACRPT
000160                IANLIN= INLINE - 8                                                JIM
000161                IAPAG=IAPAG+1                                                     ACRPT
000162                WRITE(IOFILE,200) IACASE,XTITL,IAPAG                              ACRPT
000163                WRITE (IOFILE,240)  ( I,I=1,4 )                                   JIM
000164                WRITE(IOFILE,250)                                                 ACRPT
000165                WRITE(IOFILE,260)                                                 ACRPT
000166          C                                                                       ACRPT
000167          C               INITIALIZE FIRST LINE OF SEGMENT AND PRINT              ACRPT
000168          C                                                                       ACRPT
000169           10   ISUM=0                                                           ACRPT
000170          C               INITIALIZE ACQUISITION  POINTERS TO BLANKS              ACRPT
000171                DO 20 J=1,60,1                                                    ACRPT
000172                JIMAG(J)=IBCD(11)                                                 ACRPT
000173           20   CONTINUE                                                         ACRPT
000174                IF(ISNAC.NE.0) GOTO 22                                            ACRPT
000175          C               NO ACQUISITIONS,PRINT NULL LINE AND EXIT                ACRPT
000176                WRITE(IOFILE,290)ISSEG                                            ACRPT
000177                GOTO 60                                                           ACRPT
000178          C                TEST IF ANY WINDOW REQUIRES  NONE OR ALL ILLEGAL       JIM
```

```
000179          C                PRINTED OUT UNDER SELECTED ACQUISITIONS.                JIM
000180             22 IFLG=0                                                             ACRPT
000181                DO 25 I=1,4,1                                                      ACRPT
000182                J=15*(I-1)+1                                                       ACRPT
000183                IF ( IACNT(I) .NE. 0 )  GO TO 23                                   JIM
000184          C                SET UP TO PRINT  NONE                                   JIM
000185                DO 21 K=1,4                                                        JIM
000186                JIMAG(J)= NONE(K)                                                  JIM
000187             21 J= J + 1                                                           JIM
000188                GO TO 25                                                           JIM
000189             23 IF ( IACNTL(I) .NE. 0 )  GO TO 25                                  JIM
000190          C                SET UP TO PRINT  ALL ILLEGAL                            JIM
000191                DO 24 K=1,11,1                                                     ACRPT
000192                JIMAG(J)=ILLEG(K)                                                  ACRPT
000193                J=J+1                                                              ACRPT
000194             24 CONTINUE                                                           ACRPT
000195             25 CONTINUE                                                           ACRPT
000196          C             SET UP NEXT LINE FOR EACH OF 4 WINDOWS                   ACRPT
000197             30 DO 40 I=1,4,1                                                      ACRPT
000198                L=ISUB+(25*(I-1))                                                  ACRPT
000199          C                EACH WINDOW MAY HAVE FROM 0 TO 4 ACQUISITIONS           ACRPT
000200                DO 35 K=1,4,1                                                      ACRPT
000201                J=(15*(I-1))+(4*(K-1))+1                                           ACRPT
000202                IF ( (ISUB+K) .GT. IACNTL(I) )  GO TO 40                           JIM
000203                L=L+1                                                              ACRPT
000204                CALL BINBCD (IAPNT(L+1),3,JIMAG(J))                                JIM
000205             35 CONTINUE                                                           ACRPT
000206             40 CONTINUE                                                           ACRPT
000207                ISUB=ISUB+4                                                        ACRPT
000208          C             TEST FIRST LINE OR SUBSEQUENT                            ACRPT
000209                IF(IFLG.NE.0) GOTO 45                                              ACRPT
000210                ISWFLG= 1HS                                                        JIM
000211                IF ( IPTYPE .EQ. 1 )  ISWFLG= 1HW                                  JIM
000212                WRITE (IOFILE,270) ISWFLG,ISSFG,ISNAC,IACNT(1),( JIMAG(L),L=1,15 )JIM
000213               1  , IACNT(2),( JIMAG(L),L=16,30 ), IACNT(3),( JIMAG(L),L=31,45 )  JIM
000214               2  , IACNT(4),( JIMAG(L),L=46,60 )                                JIM
000215                IFLG=1                                                             ACRPT
000216                GOTO 46                                                            ACRPT
000217          C             WRITE SUBSEQUENT LINES                                   ACRPT
000218             45 WRITE (IOFILE,280)  ( JIMAG(L),L=1,60 )                            JIM
000219             46 NMAX=NMAX-1                                                        ACRPT
000220                IF ( NMAX .LE. 0 )  GO TO 60                                       JIM
000221          C             BLANK OUT ACQUISTION POINTERS FOR NEXT PRINT LINE.       JIM
000222                DO 47 L=1,60                                                       JIM
000223                JIMAG(L)= IBCD(11)                                                 JIM
000224             47 CONTINUE                                                           JIM
000225                GO TO 30                                                           JIM
000226          C                                                                       ACRPT
000227          C             WRITE STATISTICS                                         ACRPT
000228          C                                                                       ACRPT
000229             50 WRITE(IOFILE,300)ACPN                                           ACRPT
000230             60 RETURN                                                          ACRPT
000231            200 FORMAT(6H1CASE ,I4,27H  SEGMENT ACQUISITION DATA ,12A4,16H LPP SIMF
                  1ULATION ,5HPAGE ,I4)
000233            210 FORMAT(1H0,17X,8HCOUNTRY ,A4,2X,7HREGION ,I3,2X,5HZONE ,I3,2X,7HSTACRPT
                  2RATA ,I4,2X,10HSUBSTRATA ,I4)                                       JIM
000235            240 FORMAT(//12H0S/W SEGMENT,2X,4(13H****** WINDOW,I2,7H ******,2X) ) JIM
000236            250 FORMAT(5X,8H0-TOTAL ,2X,4(5H0TOTAL,2X,8HSELECTED,9X))            JIM
000237            260 FORMAT(10X,4(7HACCESS ,12HACQUISITIONS,5X)/)                     JIM
000238            270 FORMAT(200 ,A1,2X,I5,10H-,I3,2X,4(I2,3X,15A1,4X))                JIM
```

```
000239          280  FORMAT(1H ,20X,4(15A1,9X))                          JIM
000240          290  FORMAT(1H0,19,4H-000,3X,4HNONE)                     JIM
000241          300  FORMAT(1H1,30HTOTAL ACQUISI RECORDS WRITTEN=,F7.0)   ACRPT
000242               END                                                 ACRPT
```

```
* FLT AFIND,1,760426, 50347    , 1

000001            SUBROUTINE AFIND                                                    AFIND
000002          C*************************************************************************AFIND
000003          C                                                                     AFIND
000004          C ROUTINE       AFIND                                               AFIND
000005          C                                                                     AFIND
000006          C PURPOSE       THIS ROUTINE SELECTS ACQUISITIONS FROM A SEGREFER DATA  AFIND
000007          C               RECORD BASED UPON                                     AFIND
000008          C                   .CLOUD COVER LT MAXIMUM                           AFIND
000009          C                   .SUN ANGLE GT MINIMUM                             AFIND
000010          C                   .ACQUISITION DATE WITHIN CROP WINDOW              AFIND
000011          C                                                                     AFIND
000012          C LINKAGE       CALL AFIND                                          AFIND
000013          C                                                                     AFIND
000014          C LOCAL VARIABLES                                                   AFIND
000015          C                                                                     AFIND
000016          C               I,    CURRENT WINDOW NUMBER                         AFIND
000017          C               J,    CURRENT ACQUISITION NUMBER                    AFIND
000018          C               K,    INDEX INTO WINDOW DATE-EITHER SPRING OR WINTER  AFIND
000019          C               IRDAY,RELATIVE ACQUISITION DAY FOR PRODUCTION VECTOR  AFIND
000020          C               NACQ, NO. ACQUISITIONS CURRENT WINDOW               AFIND
000021          C                                                                     AFIND
000022          C ROUTINES CALLED                                                   AFIND
000023          C               ABARF-ERROR ROUTINE                                 AFIND
000024          C                    07-REFERENCE DATE OUT OF RANGE                 AFIND
000025          C                                                                     AFIND
000026          C*************************************************************************AFIND
000027          C                                                                     IPLK
000028          C COMMON BLOCK IBLK-MANUAL INPUT PARAMETERS CONSTANTS                 IBLK
000029          C                                                                     IPLK
000030                COMMON /IPLK/                                                 IPLK
000031               1XITTL(12),XISANG,XICOUN,IWTTL(16),                            JTM
000032               2ICID(5),IACASE,ISCASE,IWCASE,IPCC,ISANG,IREG,IZONE,IIFILE,IOFILE, IBLK
000033               3ISFILE,ICFILE,IAFILE,INLINE,                                  IBLK
000034               4IPRPT,ICRPT                                                   IBLK
000035                LOGICAL IPRPT,ICRPT                                           IPLK
000036          C NAME     DESCRIPTION                                             IBLK
000037          C XITTL    REPORT CASE ORIENTED PAGE TITLE FROM INPUT(12A4)        IBLK
000038          C XISANG   MINIMUM SUN ANGLE IN RADIANS FROM INPUT                 IBLK
000039          C XICOUN   COUNTRY ID FROM INPUT (A4)                              IBLK
000040          C IWTTL    CROP WINDOW TITLES.  TITLE FOR EACH WINDOW-OCCUPIES 8 WORDS  JTM
000041          C            AT 2 CHARACTERS PER WORD.                            JTM
000042          C ICID     COLUMNS 75-80 OF INPUT CARD 5A2                         IBLK
000043          C IACASE   ACQUISI FILE CASE NO. FROM INPUT                        IBLK
000044          C ISCASE   SEGREFER CASE NO. FROM INPUT                            IBLK
000045          C IWCASE   CROPWIND CASE NO. FROM INPUT                            IBLK
000046          C IPCC     MAXIMUM PERCENT CLOUD COVER * 10-TO BE USED IN SEGMENT SELECT  IBLK
000047          C ISANG    MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (I4)        IBLK
000048          C IREG     REGION TO-FROM INPUT (I3)                               IBLK
000049          C IZONE    ZONE ID   FROM INPUT (I3)                               IBLK
000050          C IIFILE   LOGICAL UNIT NO. FOR CARD INPUT                         IBLK
000051          C IOFILE   LOGICAL UNIT NO. FOR REPORT OUTPUT                      IBLK
000052          C ISFILE   LOGICAL UNIT NO. FOR SEGREFER DATA FILE                 IBLK
000053          C ICFILE   LOGICAL UNIT NO. FOR CROPWINDOW FILE                    IBLK
000054          C IAFILE   LOGICAL UNIT NO. FOR ACQUISITION FILE                   IBLK
000055          C INLINE   NO. LINES PER PAGE                                      IBLK
000056          C IPRPT    PROCESSING LOAD REPORT FLAG (T=PRINT REPORT)            IBLK
000057          C ICRPT    ACQUISITION (CROP WINDOW) REPORT FLAG (T=PRINT REPORT)  IBLK
000058          C                                                                  IPLK
```

```
000059      C COMMON BLOCK PBLK-PROGRAM COUNTERS AND CONSTANTS                    PBLK
000060      C                                                                    PBLK
000061            COMMON /PBLK/                                                  PBLK
000062           1XPWRK(150),IPFIF,IPTYPE,IPERR,IBCD(15),IPWRK(150),ITEN(5),     PBLK
000063           2)PEFLG(13),IPEOF                                               PBLK
000064            LOGICAL IPEOF                                                  PBLK
000065      C                                                                    PBLK
000066      C NAME     DESCRIPTION                                               PBLK
000067      C XPWRK    FLOATING POINT WORK AREA-DIMENSION 150                    PBLK
000068      C IPFIF    FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST)       PBLK
000069      C IPTYPE   TYPE WHEAT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)           JTM
000070      C IPERR    ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT)      PBLK
000071      C IBCD     HOLLERITH ARRAY/0/1/2/3/4/5/6/7/8/9/ /-/0/*/,/            PBLK
000072      C ITEN     POWERS OF TEN (1,10,100,1000,10000)                       PBLK
000073      C IPWRK    INTEGER WORK AREA-DIMENSION 150                           PBLK
000074      C IPEOF    END OF FILE FLAG (TRUE=END FILE)                          PBLK
000075      C IPEFLG   FATAL ERROR FLAG FOR ERROR 1-15 (T=FATAL)                 PBLK
000076      C                                                                    SBLK
000077      C COMMON BLOCK SBLK-SEGREFER FILE HEADER AND DATA RECORD VARIABLES    SBLK
000078      C                                                                    SBLK
000079            COMMON/SBLK/                                                   SBLK
000080           1SCRN,SFNAM1(2),SFNAM2(2),SFTRL,SCOUN,SPCSW,SPCWW,SSANG(150),   SBLK
000081           2ISSCAS,ISDATE,ISNDAY,ISNAC,ISREG,ISZONE,ISSTRA,                SBLK
000082           3ISUBS, ISSEG,ISTRN,ISRDAY(150),ISPCC(150),NVEH
000083           4,IORDER(150),IKDAY(150)                                                *NEW
000084      C NAME     DESCRIPTION                                               SBLK    **-1
000085      C SCRN     SEGREFER CURRENT RECORD NO.                               SBLK
000086      C SFNAM1   SEGREFER FILE NAME PROTOTYPE*SEGREFER*                    SBLK
000087      C SFNAM2   SEGREFER FILE NAME FROM LABEL                             SBLK
000088      C SFTRL    TRAILER ID *ZZZZ*                                         SBLK
000089      C SCOUN    SEGREFER COUNTRY ID (F6.2)                                SBLK
000090      C SPCSW    PERCENT SPRING WHEAT(F6.2)                                SBLK
000091      C SPCWW    PERCENT WINTER WHEAT (F6.2)                               JTM
000092      C SSANG    SUN ANGLE(IN RADIANS) FOR EACH OF 150 SEGMENTS            SBLK
000093      C ISSCAS   SEGREFER CASE NO. FROM LABEL                              SBLK
000094      C ISDATE   SEGREFER REFERENCE DATE-ZULU-                             SBLK
000095      C ISNDAY   NO. DAYS IN STUDY (1-426)                                 SBLK
000096      C ISNAC    NO. ACQUISITIONS IN RECORD (0-150)                        SBLK
000097      C ISREG    SEGREFER REGION ID                                        SBLK
000098      C ISZONE   SEGREFER ZONE   ID                                        SBLK
000099      C ISSTRA   SEGREFER STRATA ID                                        SBLK
000100      C ISUBS    SEGREFER SUBSTRATA ID                                     SBLK
000101      C ISSEG    SEGREFER SEGMENT ID                                       SBLK
000102      C ISTRN    TRAINING SEGMENT INDICATOR(0=NORMAL,1=TRAINING)           SBLK
000103      C ISRDAY   ACQUISITION DAY FOR 150 ACQUISITIONS-ZULU-                SBLK
000104      C          IF ZERO, END OF ALL ACQUISITIONS IN RECORD                 SBLK
000105      C ISPCC    PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS             SBLK
000106      C NVEH     NO OF VEHICLES IN THE SWATH TABLE
000107      C IORDER   ORIGINAL ORDER OF ACCESS DATES AS READ FROM SEQ. REF. FILE JTM
000108      C                                                                    CBLK
000109      C COMMON BLOCK CBLK-CROPWIND HEADER AND DATA                         CBLK
000110      C                                                                    CBLK
000111            COMMON /CBLK/                                                  CBLK
000112           1CCRN,CFNAM1(2),CFNAM2(2),CCOUN,                                CBLK
000113           2ICCASE,ICWIND,ICREG,ICZONE,ICSTRA,ICSUBS,ICLOW(8),ICHI(8)      CBLK
000114      C NAME     DESCRIPTION                                               CBLK
000115      C CCRN     CURRENT CROPWIND RECORD NO.                               CBLK
000116      C CFNAM1   FILE NAME *CROPWIND*(2A4)                                 CBLK
000117      C CFNAM2   LABEL FILE NAME-MUST MATCH CFNAM1                         CBLK
000118      C CCOUN    CROPWIND COUNTRY ID (A4)                                  CBLK
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000119      C ICCASE  CROPWIND CASE NO. FROM LABEL                                  CBLK
000120      C ICWIND  NO. OF CROP WINDOWS (8)                                      CBLK
000121      C ICREG   CROPWIND REGION ID                                           CBLK
000122      C ICZONE  CROPWIND ZONE ID                                             CBLK
000123      C ICSTRA  CROPWIND STRATA ID                                           CBLK
000124      C ICSUBS  CROPWIND SUBSTRATA ID                                        CBLK
000125      C ICLOW   START DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950               CBLK
000126      C ICHI    STOP  DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950               CBLK
000127      C                                                                      ABLK
000128      C COMMON BLOCK ABLK-ACQUISI FILE AND REPORT VARIABLES                  ABLK
000129      C                                                                      ABLK
000130            COMMON/ABLK/                                                     ABLK
000131           1ACRN,AFNAM(2),AFTRL,ANACQ(426),                                  ABLK
000132           2IACNT(4),IATOT,IAPNT(25,4),IAPAG,IANLIN,IACNTL(4)                JTM
000133      C NAME    DESCRIPTION                                                  ABLK
000134      C ACRN    CURRENT RECORD NO.                                           ABLK
000135      C AFNAM   LABEL FILE NAME*ACQUISI *                                    ABLK
000136      C AFTRL   TRAILER ID *ZZZ*  A4                                         ABLK
000137      C ANACQ   NO. OF ACQUISITIONS SELECTED FOR EACH OF 426 DAYS            ABLK
000138      C IACNT   TOTAL NO. OF ACCESSES FOR EACH OF 4 WINDOWS                  JTM
000139      C IATOT    TOTAL NO. OF ACQUISITIONS FOR ALL 4 WINDOWS                 ABLK
000140      C   IAPNT    INDEX TO SEQREFER ACQUISITION (1-150) THAT IDENTIFIES     ABLK
000141      C            A SELECTED SEGMENT. IAPNT(I,J)= WINDOW I THROUGH 4.       ABLK
000142      C IAPAG    ACQUISITION REPORT CURRENT PAGE NUMBER                      ABLK
000143      C IANLIN   ACQUISITION REPORT NO. LINES REMAINING CURRENT PAGE         ABLK
000144      C IACNTL  NO. OF LEGAL ACQUISITIONS SELECTED FOR EACH OF 4 WINDOWS     JTM
000145            DIMENSION KAPNT(100)                                             AFIND
000146            EQUIVALENCE(KAPNT(1),IAPNT(1,1))                                 SETI
000147      C               FIND NO.ACQUISITIONS IN RECORD                         AFIND
000148            ISNAC=0                                                          AFIND
000149            DO 5  I=1,150                                                    AFIND
000150            IF(ISRDAY(I).EQ.0) GOTO 6                                        AFIND
000151            ISNAC=I                                                          AFIND
000152       5    CONTINUE                                                         AFIND
000153      C               ZERO  SEGMENT POINTER ARRAY AND TOTALS                 AFIND
000154       6    DO 10 I=1,100                                                    AFIND
000155            KAPNT(I)=0                                                       AFIND
000156       10   CONTINUE                                                         AFIND
000157            IATOT=0                                                          AFIND
000158      C             INITIALIZE CROPWIND WINDOW INDEX.  K=0 WINTER, K=4 SPRINGJTM
000159            K=IPTYPE-1                                                       AFIND
000160      C               FOR EACH WINDOW,I, TEST ALL SEGMENT ACQUISITIONS       AFIND
000161            DO 80 I=1,4,1                                                    AFIND
000162      C               INITIALIZE WINDOW LOOPVARIABLES K AND NACQ             AFIND
000163            K=K+1                                                            AFIND
000164            NACQ=0                                                           AFIND
000165            NACQI= 0                                                         JTM
000166      C               FOR EACH OF 150 SEGMENTS,SEE IF ACQUISITION IS WITHIN  AFIND
000167      C               WINDOW AND SATISFIES CLOUD AND SUN CRITERIA            AFIND
000168            DO 60 J=1,ISNAC                                                  SETI
000169      C               IS ACCESS DATE WITHIN WINDOW I                         JTM
000170            IF(ICHI(K).LT.ISRDAY(J)) GOTO 60                                 AFIND
000171            IF(ICLOW(K).GT.ISRDAY(J)) GOTO 60                                AFIND
000172      C               TEST DAY WITHIN SIZE  OF VECTOR                        AFIND
000173            IRDAY=ISRDAY(J)-ISDATE+1                                         AFIND
000174            IF(IRDAY.LE.ISRDAY) GOTO 40                                      AFIND
000175      C               ERROR 7,DAY OUT OF RANGE OF VECTOR                     AFIND
000176       30   IPERR=IRDAY                                                      AFIND
000177            CALL ABORF(7)                                                    AFIND
000178       40   IF(IRDAY.LT.1) GOTO 50                                           AFIND
```

```
000179          C                   ACCESS DATE IS WITHIN WINDOW I.  ADVANCE NACQI          JIM
000180          C                   (TOTAL ACCESS COUNT FOR WINDOW I)                       JIM
000181                NACQI= NACQT + 1                                                      JIM
000182          C                                                                           JIM
000183          C                   ANY PREVIOUS LEGAL ACQUISITIONS FOR CURRENT SEGMENT     JIM
000184          C                   WITHIN WINDOW I.                                        JIM
000185                IF ( NACQ .EQ. 0 )  GO TO 50                                          JIM
000186          C                   YES.  IF THIS IS NOT A TRAINING SEGMENT, SKIP CLOUD     JIM
000187          C                   COVER AND SUN ANGLE TEST.                               JIM
000188                IF(ISTRN .NE. 0)GO TO 60
000189          C                   NOW TEST FOR LEGAL ACQUISITION SATISFYING CLOUD COVER   JIM
000190          C                     .LT. MAXIMUM  AND SUN ANGLE .GE. MINIMUM.             JIM
000191           50   IF ( ISCC(J) .GE. IPCC )  GO TO 60                                    JIM
000192                IF ( SSANG(J) .LT. XISANG )  GO TO 60                                 JIM
000193          C                   SEGMENT SELECTED FOR WINDOW I.  UPDATE COUNT (NACQ) ANDJIM
000194          C                   INDEX ARRAY.                                            JIM
000195                NACQ= NACQ + 1                                                        JIM
000196                IAPNI(NACQ,I)= IORDER(J)                                              JIM
000197                ANACQ(IRDAY)=ANACQ(IRDAY)+1                                           AFIND
000198          C                   ACCEPT MAXIMUM OF 25 ACQUISITIONS FOR EACH WINDOW.      JIM
000199                IF(NACQ.GE.25) GOTO 70                                                AFIND
000200          C                   END OF SEGMENT                                          AFIND
000201           60   CONTINUE                                                             AFIND
000202          C                   END OF WINDOW, POST TOTAL ACQUISITION COUNT             AFIND
000203           70   IACNT(I)= NACQT                                                      JIM
000204                IACNII(I)= NACQ                                                       JIM
000205                IATOT= IATOT + NACQT                                                  JIM
000206           80   CONTINUE                                                             AFIND
000207                RETURN                                                               AFIND
000208                END                                                                  AFIND
```

```
* FLT AINIL,1,760426,-50349     , 1
000001          SUBROUTINE AINIL                                                AINIL
000002        C*****************************************************************AINIL
000003        C                                                                 AINIL
000004        C ROUTINE       AINIL                                             AINIL
000005        C                                                                 AINIL
000006        C PURPOSE       THIS ROUTINE INITIALIZESTHE SACS PROGRAM          AINIL
000007        C                    .READS SEGREFER LABEL                        AINIL
000008        C                    .READS CROPWIND LABEL                        AINIL
000009        C                    .READS INPUT PARAMETER CARDS                 AINIL
000010        C                                                                 AINIL
000011        C LINKAGE       CALL AINIL                                        AINIL
000012        C                                                                 AINIL
000013        C ROUTINES CALLED NONE                                            AINIL
000014        C                                                                 AINIL
000015        C LOCAL VARIABLES NONE                                            AINIL
000016        C                                                                 AINIL
000017        C ERRORS DETECTED                                                 AINIL
000018        C                    01,02,03,04,08,09,10,11                      AINIL
000019        C                                                                 AINIL
000020        C*****************************************************************AINIL
000021        C                                                                 AINIL
000022        C                                                                 IBLK
000023        C COMMON BLOCK IBLK-MANUAL INPUT PARAMETERS CONSTANTS             IBLK
000024        C                                                                 IBLK
000025             COMMON /IBLK/                                                IBLK
000026            1XITTL(12),XISANG,XICOUN,IWTTL(16),                           JIM
000027            2ICID(3),IACASE,ISCASE,IWCASE,IPCC,ISANG,IREG,IZONE,IIFILE,IOFILE, IBLK
000028            3ISFILE,ICFILE,IAFILE,INLINE,                                 IBLK
000029            4IPRPT,ICRPT                                                  IBLK
000030             LOGICAL IPRPT,ICRPT                                          IBLK
000031        C NAME    DESCRIPTION                                             IBLK
000032        C XITTL   REPORT CASE ORIENTED PAGE TITLE FROM INPUT(12A4)        IBLK
000033        C XISANG  MINIMUM SUN ANGLE IN RADIANS FROM INPUT                 IBLK
000034        C XICOUN  COUNTRY ID FROM INPUT (A4)                              IBLK
000035        C IWTTL   CROP WINDOW TITLES.  TITLE FOR EACH WINDOW OCCUPIES 8 WORDS JIM
000036        C           AT 2 CHARACTERS PER WORD.                             JIM
000037        C ICID    COLUMNS 75-80 OF INPUT CARD 5A2                         IBLK
000038        C IACASE  ACQUISI FILE CASE NO. FROM INPUT                        IBLK
000039        C ISCASE  SEGREFER CASE NO. FROM INPUT                            IBLK
000040        C IWCASE  CROPWIND CASE NO. FROM INPUT                            IBLK
000041        C IPCC    MAXIMUM PERCENT CLOUD COVER * 10-TO BE USED IN SEGMENT SELECT IBLK
000042        C ISANG   MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (I4)        IBLK
000043        C IREG    REGION ID FROM INPUT (I3)                               IBLK
000044        C IZONE   ZONE ID   FROM INPUT (I3)                               IBLK
000045        C IIFILE  LOGICAL UNIT NO. FOR CARD INPUT                         IBLK
000046        C IOFILE  LOGICAL UNIT NO. FOR REPORT OUTPUT                      IBLK
000047        C ISFILE  LOGICAL UNIT NO. FOR SEGREFER DATA FILE                 IBLK
000048        C ICFILE  LOGICAL UNIT NO. FOR CROPWINDOW FILE                    IBLK
000049        C IAFILE  LOGICAL UNIT NO. FOR ACQUISITION FILL                   IBLK
000050        C INLINE  NO. LINES PER PAGE                                      IBLK
000051        C IPRPT   PROCESSING LOAD REPORT FLAG (T=PRINT REPORT)            IBLK
000052        C ICRPT   ACQUISITION (CROP WINDOW) REPORT FLAG (T=PRINT REPORT)  IBLK
000053        C                                                                 PBLK
000054        C COMMON BLOCK PBLK-PROGRAM COUNTERS AND CONSTANTS                PBLK
000055        C                                                                 PBLK
000056             COMMON /PBLK/                                                PBLK
000057            1XPWRK(150),IPFIF,IPTYPE,IPERR,IBCD(15),IPWRK(150),ITEM(5),   PBLK
000058            2IPLFLG(13),IPEOF                                             PBLK
```

```
000059            LOGICAL IPEOF
000060      C                                                                  PBLK
000061      C NAME    DESCRIPTION                                              PBLK
000062      C XPWRK   FLOATING POINT WORK AREA-DIMENSION 150                   PBLK
000063      C IPFTF   FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST).     PBLK
000064      C IPITYPE TYPE WHEAT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)          JTM
000065      C IPERR   ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT)     PBLK
000066      C IHCD    HOLLERITH ARRAY/0/1/2/3/4/5/6//7/8/9/ /-/0/*/,/          PBLK
000067      C ITEN    POWERS OF TEN (1,10,100,1000,10000)                      PBLK
000068      C IPWRK   INTEGER WORK AREA-DIMENSION 150                          PBLK
000069      C IPEOF   END OF FILE FLAG (TRUE=END FILE)                         PBLK
000070      C IPEFLG  FATAL ERROR FLAG FOR ERROR 1-13 (T=FATAL)                PBLK
000071      C                                                                  SBLK
000072      C COMMON BLOCK SBLK-SEGREFER FILE HEADER AND DATA RECORD VARIABLES  SBLK
000073      C                                                                  SBLK
000074            COMMON/SBLK/                                                 SBLK
000075           1SCRN,SFNAM1(2),SFNAM2(2),SFTRL,SCOUN,SPCSW,SPCWW,SSANG(150),  SBLK
000076           2ISSCAS,ISDATE,ISNDAY,ISNAC,ISREG,ISZONE,ISSTRA,              SBLK
000077           3ISUBS, ISSEG,ISTRN,ISRDAY(150),ISPCC(150),NVEH               
000078           4,IORDER(150),IKDAY(150)                                      *NLW
000079      C NAME    DESCRIPTION                                              SBLK    **-1
000080      C SCRN    SEGREFER CURRENT RECORD NO.                              SBLK
000081      C SFNAM1  SEGREFER-FILE NAME PROTOTYPE*SEGREFER*                   SBLK
000082      C SFNAM2  SEGREFER FILE NAME FROM LABEL                            SBLK
000083      C SFTRL   TRAILER ID *ZZZZ*                                        SBLK
000084      C SCOUN   SEGREFER COUNTRY ID (F6.2)                               SBLK
000085      C SPCSW   PERCENT SPRING WHEAT(F6.2)                               SBLK
000086      C SPCWW   PERCENT WINTER WHEAT (F6.2)                              JTM
000087      C SSANG   SUN ANGLE(IN RADIANS) FOR EACH OF 150 SEGMENTS           SBLK
000088      C ISSCAS  SEGREFER CASE NO. FROM LABEL                             SBLK
000089      C ISDATE  SEGREFER REFERENCE DATE-ZULU-                            SBLK
000090      C ISNDAY  NO. DAYS IN STUDY (1-426)                                SBLK
000091      C ISNAC   NO. ACQUISITIONS IN RECORD (0-150)                       SBLK
000092      C ISREG   SEGREFER REGION ID                                       SBLK
000093      C ISZONE  SEGREFER ZONE   ID                                       SBLK
000094      C ISSTRA  SEGREFER STRATA ID                                       SBLK
000095      C ISUBS   SEGREFER SUBSTRATA ID                                    SBLK
000096      C ISSEG   SEGREFER SEGMENT ID                                      SBLK
000097      C ISTRN   TRAINING SEGMENT INDICATOR(0=NORMAL,1=TRAINING)          SBLK
000098      C ISRDAY  ACQUISITION DAY FOR 150 ACQUISITIONS-ZULU-               SBLK
000099      C         IF ZERO, END OF ALL ACQUISITIONS IN RECORD               SBLK
000100      C ISPCC   PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS            SBLK
000101      C NVEH    NO OF VEHICLES IN THE SWATH TABLE
000102      C IORDER  ORIGINAL ORDER OF ACCESS DATES AS READ FROM SEQ. REF. FILE JTM
000103      C                                                                  CBLK
000104      C COMMON BLOCK CBLK-CROPWIND HEADER AND DATA                       CBLK
000105      C                                                                  CBLK
000106            COMMON /CBLK/                                                CBLK
000107           1CCRN,CFNAM1(2),CFNAM2(2),CCOUN,                              CBLK
000108           2ICCASE,ICWIND,ICREG,ICZONE,ICSTRA,ICSUBS,ICLOW(8),ICHI(8)    CBLK
000109      C NAME    DESCRIPTION                                              CBLK
000110      C CCRN    CURRENT CROPWIND RECORD NO.                              CBLK
000111      C CFNAM1  FILE NAME *CROPWIND*(2A4)                                CBLK
000112      C CFNAM2  LABEL FILE NAME-MUST MATCH CFNAM1                        CBLK
000113      C CCOUN   CROPWIND COUNTRY ID (A4)                                 CBLK
000114      C ICCASE  CROPWIND CASE NO. FROM LABEL                             CBLK
000115      C ICWIND  NO. OF CROP WINDOWS (8)                                  CBLK
000116      C ICREG   CROPWIND REGION ID                                       CBLK
000117      C ICZONE  CROPWIND ZONE ID                                         CBLK
000118      C ICSTRA  CROPWIND STRATA ID                                       CBLK
```

```
000119      C ICSUBS  CROPWIND SUBSTRATA ID                                    CBLK
000120      C ICLOW   START DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950           CBLK
000121      C ICHI    STOP  DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950           CBLK
000122      C                                                                  APLK
000123      C COMMON BLOCK ABLK-ACQUISI FILE AND REPORT VARIABLES              APLK
000124      C                                                                  APLK
000125            COMMON/ABLK/                                                 APLK
000126           1ACRN,AFNAM(2),AFTRL,ANACQ(426),                              APLK
000127           2IACNT(4),IATOT,IAPNT(25,4),IAPAG,IANLIN,IACNTL(4)            JIM
000128      C NAME    DESCRIPTION                                              APLK
000129      C ACRN    CURRENT RECORD NO.                                       APLK
000130      C AFNAM   LABEL FILE NAME*ACQUISI *                                APLK
000131      C AFTRL   TRAILER ID *ZZZZ*  A4                                    APLK
000132      C ANACQ   NO. OF ACQUISITIONS SELECTED FOR EACH OF 426 DAYS        APLK
000133      C IACNT   TOTAL NO. OF ACCESSES FOR EACH OF 4 WINDOWS              JIM
000134      C IATOT    TOTAL NO. OF ACQUISITIONS FOR ALL 4 WINDOWS             APLK
000135      C   IAPNT     INDEX TO SEGRLFER ACQUISITION (1-150) THAT IDENTIFIES APLK
000136      C           A SELECTED SEGMENT. IAPNT(I,J)= WINDOW 1 THROUGH 4.    APLK
000137      C IAPAG    ACQUISITION REPORT CURRENT PAGE NUMBER                  APLK
000138      C IANLIN   ACQUISITION REPORT NO. LINES REMAINING CURRENT PAGE     APLK
000139      C IACNTL  NO. OF LEGAL ACQUISITIONS SELECTED FOR EACH OF 4 WINDOWS JIM
000140      C                                                                  JIM
000141            COMMON  /INPERR/  INPERR                                     JIM
000142      C INPERR  INPUT ERROR FLAG. SET = 1 IN ABARF IF THERE ARE ANY FATAL JIM
000143      C               ERRORS DETECTED ON THE INPUT CARDS BY AINIL        JIM
000144      C                                                                  JIM
000145            DIMENSION ICRD(2)                                            AINIL
000146            DATA ICRD /2HSA,2HCS/                                        AINIL
000147            DATA ICRD1,ICRD2/2H01,2H02/                                  AINIL
000148      C               READ SEGRLFER HEADER                               AINIL
000149            REWIND ISFILE                                                SFII
000150            REWIND ICFILE                                                SFII
000151            REWIND IAFILE                                                JIM
000152            READ(ISFILE)SFNAM2,ISSCAS,ISDATF,ISNDAY,I,I,I,NVEH
000153            SCRN=1                                                       AINIL
000154            IF (SFNAM1.EQ.SFNAM2) GOTO 4                                 AINIL
000155       3    CALL ABARF(2)                                                AINIL
000156       4    IF (SFNAM1(2).NE.SFNAM2(2))GOTO 3                            AINIL
000157       5    IF(ISNDAY.LE.426)GOTO 6                                      AINIL
000158            CALL ABARF(11)                                               AINIL
000159      C               READ CROPWIND FILE LABEL                           AINIL
000160          6 READ(ICFILE)CFNAM2,ICCASE,ICWIND                             SFII
000161            1  ,( IFILL,I=1,17 )
000162            CCRN=1                                                       AINIL
000163            IF(CFNAM1.EQ.CFNAM2) GOTO 8                                  AINIL
000164       7    CALL ABARF(1)                                                AINIL
000165       8    IF (CFNAM1(2).NE. CFNAM2(2))GO TO 7                          AINIL
000166      C               READ FIRST INPUT CARD                              AINIL
000167            READ(TIFILL,100)XTITL,ICID                                   AINIL
000168            IF(ICRD(1).EQ.ICID) GOTO 12                                  AINIL
000169       10   CALL ABARF(8)                                                AINIL
000170            GO TO 13                                                     JIM
000171       12   IF(ICRD(2).NE.ICID(2)) GOTO 10                               AINIL
000172            IF(ICID(3) .NE. 1)GO TO 10
000173       13   CONTINUE                                                     JIM
000174      C               READ SECOND INPUT CARD                             AINIL
000175            READ(TIFILE,110)IACASE,ISCASE,INCASE,IPCC,ISANG,XICOUN,IREG,IZONE,AINIL
000176           CIPRPI,TIRPI,ICID                                             AINIL
000177            WRITE(IOFILE,105)IACASE,XTITL,XTITL
000178            WRITE(IOFILE,106)IACASE,ISCASE,INCASE,IPCC,ISANG,XICOUN,IREG,IZONE
```

```
000179              1,IPRPT,ICRPT                                                    
000180                   IF ( ICRD(1) .NE. ICID )  GO TO 14                          JIM
000181                   IF(ICID(3) .EQ. 2)GO TO 15                                  
000182               14  CALL ABARF(8)                                               JIM
000183               15  CONTINUE                                                    JIM
000184              C                CHECK SEGREFER AND CROPWIND CASE NUMBERS        AINIL
000185                   IF(ISCASE.EQ.0) GOTO 16                                     AINIL
000186                   IF(ISCASE.EQ.ISSCAS) GOTO 16                                AINIL
000187                   CALL ABARF(4)                                               AINIL
000188               16  IF(IWCASE.EQ.0) GO TO 18                                   AINIL
000189                   IF(IWCASE.EQ.ICCASE) GOTO 18                                AINIL
000190                   CALL ABARF(3)                                               AINIL
000191              C                CHECK SUN AND CLOUD COVER RANGES                AINIL
000192                 18 IF (ISANG.GE.0) GOTO 22                                    AINIL
000193               20  CALL ABARF (9)                                              AINIL
000194                   GO TO 23                                                    JIM
000195               22  IF (ISANG.GT.9000) GOTO 20                                 AINIL
000196               23  CONTINUE                                                    JIM
000197                   IF (IPCC.GE.0)GOTO 26                                       AINIL
000198               24  CALL ABARF (10)                                             AINIL
000199                   GO TO 28                                                    JIM
000200               26  IF (IPCC.GT.1000) GOTO 24                                  AINIL
000201              C                IF THERE WERE ANY INPUT ERRORS, ABORT RUN.      JIM
000202               28  IF ( INPERR .NE. 0 )  CALL ABARF (-1)                      JIM
000203              C                                                                JIM
000204              C                SUN ANGLE TO RADIANS                            AINIL
000205                   XISANG=ISANG                                                AINIL
000206                   XISANG=XISANG/5729.578                                      AINIL
000207              C                INITIALIZE FLAGS COUNTERS                       AINIL
000208                   IPTYPE=0                                                    AINIL
000209                   IPEOF= .FALSE.                                              JIM
000210                   ICREG=0                                                     AINIL
000211                   ICZONE=0                                                    AINIL
000212                   ICSTRA=0                                                    AINIL
000213                   ICSUBS=0                                                    AINIL
000214                   CCOUN=XICOUN                                                AINIL
000215                   IAPAG=0                                                     AINIL
000216                   IANLIN=0                                                    AINIL
000217                   IPETE=0                                                     AINIL
000218                   DO 30 I=1,426,1                                             AINIL
000219                   ANADD(I)=0                                                  AINIL
000220               30  CONTINUE                                                    AINIL
000221              C                WRITE ACQUISITION LABEL                         JIM
000222                   CALL AWRTA                                                  JIM
000223                   RETURN                                                      AINIL
000224                100 FORMAT(12A4,26X,2A2,I2)
000225                105 FORMAT(1H1,5HCASE ,I4,8X,12A4/1H0,51X,6HHEADER/1H ,30X,12A4)
000226                108 FORMAT(1H0,19X,69HICASE  ICSESR  ICSECW  IPCC  ISANG  ICOUN  IREG
000227                   1  IZONE  IPRPT  ICRPT/1H ,20X,I4,3X,I4,4X,I4,4X,I3,3X,I4,3X,A4,
000228                  23X,I3,4X,I3,5X,L1,6X,I1)
000229                110 FORMAT(3I4,I3,I4,A4,I3,I3,2L1,43X,2A2,I2)
000230                   END                                                         AINIL
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
* FLT APRPT,1,760426, 50350     , 1

000001            SUBROUTINE APRPT                                                  APRPT
000002           C*****,*****APRPT
000003           C                                                                  APRPT
000004           C ROUTINE       APRPT                                              APRPT
000005           C                                                                  APRPT
000006           C PURPOSE       THIS ROUTINE PRINTS THE DAILY PROCESSING LOAD REPORT  APRPT
000007           C                                                                  APRPT
000008           C LINKAGE       CALL APRPT                                         APRPT
000009           C                                                                  APRPT
000010           C ROUTINES CALLED                                                  APRPT
000011           C               FZULU, CONVERT ZULU DATETO MM-DD-YY                APRPT
000012           C                                                                  APRPT
000013           C LOCAL VARIABLES                                                  APRPT
000014           C               IRDAT-YY,MM,DD REFERENCE  DATE                     APRPT
000015           C               NCPAG-NO. OF CURRENT PAGE                          APRPT
000016           C               NLINE-TOTAL NO. DETAIL LINES TO BE PRINTED         APRPT
000017           C               NLINER-NO. OF DETAIL LINES REMAINING TO BE PRINTED JTM
000018           C               NLINEP-NO. OF DETAIL LINES PER PAGE  (1-25)        JTM
000019           C               NACQM-ACQUISITION NO. FOR COLUMNS M M= 1 THROUGH 4 APRPT
000020           C                                                                  APRPT
000021           C*****,*****APRPT
000022           C                                                                  APRPT
000023           C                                                                  IPLK
000024           C COMMON BLOCK IBLK-MANUAL INPUT PARAMETERS CONSTANTS              IPLK
000025           C                                                                  IPLK
000026                 COMMON /IBLK/                                                IPLK
000027                1XITTL(12),XISANG,XICOUN,IWTTL(16),                           JTM
000028                2ICID(3),IACASE,ISCASE,IRCASE,IPCC,ISANG,IREG,IZONE,IIFILE,IOFILE, IPLK
000029                3ISFILE,ICFILE,IAFILE,INLINE,                                IPLK
000030                4IPRPT,ICRPT                                                 IPLK
000031                 LOGICAL IPRPT,ICRPT                                         IPLK
000032           C NAME     DESCRIPTION                                            IPLK
000033           C XITTL    REPORT CASE ORIENTED PAGE TITLE FROM INPUT(12A4)       IPLK
000034           C XISANG   MINIMUM SUN ANGLE IN RADIANS FROM INPUT                IPLK
000035           C XICOUN   COUNTRY ID FROM INPUT (A4)                             IPLK
000036           C IWTTL    CROP WINDOW TITLES.  TITLE FOR EACH WINDOW OCCUPIES 8 WORDS JTM
000037           C            AT 2 CHARACTERS PER WORD.                           JTM
000038           C ICID     COLUMNS 75-80 OF INPUT CARD 3A2                        IPLK
000039           C IACASE   ACQUISI FILE CASE NO. FROM INPUT                       IPLK
000040           C ISCASE   SEGREFER CASE NO. FROM INPUT                           IPLK
000041           C IRCASE   CROPWIND CASE NO. FROM INPUT                           IPLK
000042           C IPCC     MAXIMUM PERCENT CLOUD COVER * 10=TO BE USED IN SEGMENT SELECT IPLK
000043           C ISANG    MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (I4)       IPLK
000044           C IREG     REGION ID FROM INPUT (I3)                              IPLK
000045           C IZONE    ZONE ID   FROM INPUT (I3)                              IPLK
000046           C IIFILE   LOGICAL UNIT NO. FOR CARD INPUT                        IPLK
000047           C IOFILE   LOGICAL UNIT NO. FOR REPORT OUTPUT                     IPLK
000048           C ISFILE   LOGICAL UNIT NO. FOR SEGREFER DATA FILE                IPLK
000049           C ICFILE   LOGICAL UNIT NO. FOR CROPWINDOW FILE                   IPLK
000050           C IAFILE   LOGICAL UNIT NO. FOR ACQUISITION FILE                  IPLK
000051           C INLINE   NO. LINES PER PAGE                                     IPLK
000052           C IPRPT    PROCESSING LOAD REPORT FLAG (T=PRINT REPORT)           IPLK
000053           C ICRPT    ACQUISITION (CROP WINDOW) REPORT FLAG (T=PRINT REPORT) IPLK
000054           C                                                                  PPLK
000055           C COMMON BLOCK PBLK-PROGRAM COUNTERS AND CONSTANTS                 PPLK
000056           C                                                                  PPLK
000057                 COMMON /PBLK/                                                PPLK
000058                1XPARM(150),IPFIF,IPTYPE,IPERR,IBCD(15),IPARF(150),ITLN(5),  PPLK
```

```
000059          2IPEFLG(13),IPEOF                                                PBLK
000060           LOGICAL IPEOF
000061      C                                                                    PBLK
000062      C NAME    DESCRIPTION                                                PBLK
000063      C XPWRK   FLOATING POINT WORK AREA-DIMENSION 150                     PBLK
000064      C IPFTF   FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST)        PBLK
000065      C IPTYPE  TYPE WHEAT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)            JIM
000066      C IPERR   ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT)       PBLK
000067      C IPCD    HOLLERITH ARRAY/0/1/2/3/4/5/6/7/8/9/ /-/0/*/,/             PBLK
000068      C ITEN    POWERS OF TEN (1,10,100,1000,10000)                        PBLK
000069      C IPWRK   INTEGER WORK AREA-DIMENSION 150                            PBLK
000070      C IPEOF   END OF FILE FLAG (TRUE=END FILE)                           PBLK
000071      C IPEFLG  FATAL ERROR FLAG FOR ERROR 1-13 (T=FATAL)                  PBLK
000072      C                                                                    SBLK
000073      C COMMON BLOCK SBLK-SEGREFER FILE HEADER AND DATA RECORD VARIABLES   SBLK
000074      C                                                                    SBLK
000075           COMMON/SBLK/                                                    SBLK
000076          1SCRN,SFNAM1(2),SFNAM2(2),SFTRL,SCOUN,SPCSW,SPCWW,SSANG(150),    SBLK
000077          2ISSCAS,ISDATE,ISNDAY,ISNAC,ISREG,ISZONE,ISSTRA,                 SBLK
000078          3ISUBS, ISSEG,ISTRN,ISRDAY(150),ISPCC(150),NVEH
000079          4,IORDER(150),IKDAY(150)                                                *NEW
000080      C NAME    DESCRIPTION                                                SBLK-    **-1
000081      C SCRN    SEGREFER CURRENT RECORD NO.                                SBLK
000082      C SFNAM1  SEGREFER FILE NAME PROTOTYPE*SEGREFER*                     SBLK
000083      C SFNAM2  SEGREFER FILE NAME FROM LABEL                              SBLK
000084      C SFTRL   TRAILER ID *ZZZ*                                           SBLK
000085      C SCOUN   SEGREFER COUNTRY ID (F6.2)                                 SBLK
000086      C SPCSW   PERCENT SPRING WHEAT(F6.2)                                 SBLK
000087      C SPCWW   PERCENT WINTER WHEAT (F6.2)                                JIM
000088      C SSANG   SUN ANGLE(IN RADIANS) FOR EACH OF 150 SEGMENTS             SBLK
000089      C ISSCAS  SEGREFER CASE NO. FROM LABEL                               SBLK
000090      C ISDATE  SEGREFER REFERENCE DATE-ZULU-                              SBLK
000091      C ISNDAY  NO. DAYS IN STUDY (1-426)                                  SBLK
000092      C ISNAC   NO. ACQUISITIONS IN RECORD (0-150)                         SBLK
000093      C ISREG   SEGREFER REGION ID                                         SBLK
000094      C ISZONE  SEGREFER ZONE   ID                                         SBLK
000095      C ISSTRA  SEGREFER STRATA ID                                         SBLK
000096      C ISUBS   SEGREFER SUBSTRATA ID                                      SBLK
000097      C ISSEG   SEGREFER SEGMENT ID                                        SBLK
000098      C ISTRN   TRAINING SEGMENT INDICATOR(0=NORMAL,1=TRAINING)            SBLK
000099      C ISRDAY  ACQUISITION DAY FOR 150 ACQUISITIONS-ZULU-                 SBLK
000100      C         IF ZERO, END OF ALL ACQUISITIONS IN RECORD                 SBLK
000101      C ISPCC   PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS              SBLK
000102      C NVEH    NO OF VEHICLES IN THE SWATH TABLE
000103      C IORDER  ORIGINAL ORDER OF ACCESS DATES AS READ FROM SEQ. REF. FILE  JIM
000104      C                                                                    ABLK
000105      C COMMON BLOCK ABLK-ACQUISI FILE AND REPORT VARIABLES                ABLK
000106      C                                                                    ABLK
000107           COMMON/ABLK/                                                    ABLK
000108          1ACRN,AFNAM(2),AFTRL,ANACQ(426),                                 ABLK
000109          2IACNT(4),IATOT,IAPNT(25,4),IAPAG,IANLIN,IACNTL(4)               JIM
000110      C NAME    DESCRIPTION                                                ABLK
000111      C ACRN    CURRENT RECORD NO.                                         ABLK
000112      C AFNAM   LABEL FILE NAME*ACQUISI *                                  ABLK
000113      C AFTRL   TRAILER ID *ZZZZ*  A4                                      ABLK
000114      C ANACQ   NO. OF ACQUISITIONS SELECTED FOR EACH OF 426 DAYS          ABLK
000115      C IACNT   TOTAL NO. OF ACCESSES FOR EACH OF 4 WINDOWS                JIM
000116      C IATOT    TOTAL NO. OF ACQUISITIONS FOR ALL 4 WINDOWS               ABLK
000117      C   IAPNT    INDEX TO SEGREFER ACQUISITION (1-150) THAT IDENTIFIES   ABLK
000118      C            A SELECTED SEGMENT. IAPNT(I,J)= WINDOW I THROUGH 4.     ABLK
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000119      C IAPAG    ACQUISITION REPORT CURRENT PAGE NUMBER                                  ABLK
000120      C IANLIN   ACQUISITION REPORT NO. LINES REMAINING CURRENT PAGE                     ABLK
000121      C IACNTL   NO. OF LEGAL ACQUISITIONS SELECTED FOR EACH OF 4 WINDOWS                JIM
000122            DIMENSION IRDAT(3)                                                          APRPT
000123            IF(.NOT. IPRPT)GO TO 40                                                     SETI
000124      C                 INITIALIZE VARIABLES                                          APRPT
000125            CALL FZHLU(ISDATE,IRDAT)                                                    APRPT
000126            NCPAG=0                                                                     APRPT.
000127            NACQ1=0                                                                     APRPT
000128            NLINE=(ISNDAY+3)/4                                                          APRPT
000129            NLINER= NLINE                                                               JIM
000130      C                                                                                 APRPT
000131      C           INITIALIZE PAGE VARIABLES AND PRINT PAGE/COLUMN TITLES                APRPT
000132      C                                                                                 APRPT
000133       10   NCPAG=NCPAG+1                                                              APRPT
000134            NLINEP= MIN0(NLINER,25)                                                     JIM
000135            NACQ2= NACQ1 + NLINEP                                                       JIM
000136            NACQ3= NACQ2 + NLINEP                                                       JIM
000137            NACQ4= NACQ3 + NLINEP                                                       JIM
000138            WRITE(IOFILE,100)IACASE,XITTL,NCPAG                                         APRPT
000139            WRITE (IOFILE,110)  IRDAT(2),IRDAT(3),IRDAT(1),XICOUN,IREG,IZONE            JIM
000140            WRITE(IOFILE,140)                                                           APRPT
000141            WRITE(IOFILE,150)                                                           APRPT
000142            WRITE(IOFILE,130)                                                           APRPT
000143      C                                                                                 APRPT
000144      C           WRITE UP TO 25 DETAIL LINES PER PAGE                                  APRPT
000145      C                                                                                 APRPT
000146            DO 30 I=1,NLINEP                                                            JIM
000147            NACQ1=NACQ1+1                                                               APRPT
000148            NACQ2=NACQ2+1                                                               APRPT
000149            NACQ3=NACQ3+1                                                               APRPT
000150            NACQ4=NACQ4+1                                                               APRPT
000151            WRITE(IOFILE,160) NACQ1,ANACQ(NACQ1),NACQ2,ANACQ(NACQ2),NACQ3,ANACQ(NACQ3),NACQ4,ANACQ(NACQ4)  APRPT
000152      30    CONTINUE                                                                    JIM
000153            NLINER= NLINER - 25                                                         JIM
000154            NACQ1= NACQ4                                                                JIM
000155            IF ( NLINER .GT. 0 )  GO TO 10                                              JIM
000156      40    WRITE(IOFILE,170)                                                           APRPT
000157            RETURN                                                                      APRPT
000158       100  FORMAT(6H1CASE ,I4,30H DAILY PROCESSING LOAD REPORT ,12A4,16H LPP
000159           1STMULATION ,5HPAGE ,I1)
000160       110  FORMAT (1H0,9X,11HSTART DATE ,I2,1H-,I2,1H-,I2,2X,8HCOUNTRY ,A4,2X,  APRPT
000161           C,7HREGION ,I3,2X,5HZONE ,I3//)                                             JIM
000162       130  FORMAT (1H0)                                                               APRPT
000163       140  FORMAT (9H0RELATIVE,5X,3HNO.,8X,8HRELATIVE,5X,3HNO.,8X,8HRELATIVE,  APRPT
000164           C5X,3HNO.,8X,8HRELATIVE,5X,3HNO.)                                           APRPT
000165       150  FORMAT (6H   DAY,4X,12HACQUISITIONS,3X,5H  DAY,4X,12HACQUISITIONS,  APRPT
000166           C3X,5H  DAY,4X,12HACQUISITIONS,3X,                                          APRPT
000167           C5H  DAY,4X,12HACQUISITIONS)                                                APRPT
000168       160  FORMAT (1H ,4(I5,F12.0,7X))                                                JIM
000169       170  FORMAT (1H1)                                                               APRPT
000170            END                                                                         APRPT
000171
```

```
• ELT AREAD•1,760426• 50352    • 1

000001            SUBROUTINE AREAD                                          AREAD
000002          C*****************************************************************AREAD
000003          C                                                           AREAD
000004          C ROUTINE      AREAD                                        AREAD
000005          C                                                           AREAD
000006          C PURPOSE      THIS ROUTINE READS BOTH THE CROP WIND AND SEGREFER DATA  AREAD
000007          C              FILES AND RETURNS A RECORD OF EACH OR SETS A FLAG TO     AREAD
000008          C              INDICATE AN END OF DATA HAS BEEN ENCOUNTERED             AREAD
000009          C                                                           AREAD
000010          C LINKAGE      CALL AREAD                                   AREAD
000011          C                                                           AREAD
000012          C                                                           AREAD
000013          C LOCAL VARIABLES                                           AREAD
000014          C              I,    THROW AWAY INTEGER ARRAY DIMENSION 150 AREAD
000015          C              X,    THROW AWAY REAL ARRAY DIMENSION 150    AREAD
000016          C              IFLG• FLAG TO INDICATE ACQUISITIONS ARE TO BE COUNTED  AREAD
000017          C                    ONLY 0=ACCEPT AND COUNT, 1=COUNT ONLY  AREAD
000018          C                                                           AREAD
000019          C ROUTINES CALLED                                           AREAD
000020          C              ABARF,ERROR ROUTINE                          AREAD
000021          C                    05=COUNTRY NOT EQUAL FOR INPUT AND 2 FILES  AREAD
000022          C                    12=CROPWIND MISSING SEGREFER ZONE      AREAD
000023          C                    13=RECORD SKIPPED DUE TO 0 PERCENT WHEAT  AREAD
000024          C                                                           AREAD
000025          C*****************************************************************AREAD
000026          C                                                           AREAD
000027          C                                                           IBLK
000028          C COMMON BLOCK IBLK=MANUAL INPUT PARAMETERS CONSTANTS       IBLK
000029          C                                                           IBLK
000030               COMMON /IBLK/                                          IBLK
000031              1XTITL(12),XISANG,XICOUN,IWTTL(16),                     JIM
000032              2ICID(3),IACASE,ISCASE,IRCASE,IPCC,ISANG,IREG,IZONE,IIFILE,IOFILE,  IBLK
000033              3ISFILE,ICFILE,IAFILE,INLINE,                          IBLK
000034              4IPRPI,ICRPI                                           IBLK
000035               LOGICAL IPRPI,ICRPI                                   IBLK
000036          C NAME     DESCRIPTION                                     IBLK
000037          C XTITL    REPORT CASE ORIENTED PAGE TITLE FROM INPUT(12A4)  IBLK
000038          C XISANG   MINIMUM SUN ANGLE IN RADIANS FROM INPUT         IBLK
000039          C XICOUN   COUNTRY ID FROM INPUT (A4)                      IBLK
000040          C IWTTL    CROP WINDOW TITLES.  TITLE FOR EACH WINDOW OCCUPIES 8 WORDS  JIM
000041          C            AT 2 CHARACTERS PER WORD.                     JIM
000042          C ICID     COLUMNS 75-80 OF INPUT CARD 3A2                 IBLK
000043          C IACASE   ACQUISI FILE CASE NO. FROM INPUT                IBLK
000044          C ISCASE   SEGREFER CASE NO. FROM INPUT                    IBLK
000045          C IRCASE   CROPWIND CASE NO. FROM INPUT                    IBLK
000046          C IPCC     MAXIMUM PERCENT CLOUD COVER * 10-TO BE USED IN SEGMENT SELECT  IBLK
000047          C ISANG    MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (I4)  IBLK
000048          C IREG     REGION ID FROM INPUT (I3)                       IBLK
000049          C IZONE    ZONE ID   FROM INPUT (I3)                       IBLK
000050          C IIFILE   LOGICAL UNIT NO. FOR CARD INPUT                 IBLK
000051          C IOFILE   LOGICAL UNIT NO. FOR REPORT OUTPUT              IBLK
000052          C ISFILE   LOGICAL UNIT NO. FOR SEGREFER DATA FILE         IBLK
000053          C ICFILE   LOGICAL UNIT NO. FOR CROPWINDOW FILE            IBLK
000054          C IAFILE   LOGICAL UNIT NO. FOR ACQUISITION FILE           IBLK
000055          C INLINE   NO. LINES PER PAGE                              IBLK
000056          C IPRPI    PROCESSING LOAD REPORT FLAG (T=PRINT REPORT)    IBLK
000057          C ICRPI    ACQUISITION (CROP WINDOW) REPORT FLAG (T=PRINT REPORT)  IBLK
000058          C                                                          IBLK
```

```
000059      C COMMON BLOCK PBLK-PROGRAM COUNTERS AND CONSTANTS                    PBLK
000060      C                                                                    PBLK
000061            COMMON /PBLK/                                                  PBLK
000062           1XPWRK(150),IPFTF,IPTYPE,IPERR,IBCD(15),IPWRK(150),ITEN(5),     PBLK
000063           2IPEFLG(13),IPEOF                                               PBLK
000064            LOGICAL IPEOF                                                  PBLK
000065      C                                                                    PBLK
000066      C NAME     DESCRIPTION                                               PBLK
000067      C XPWRK    FLOATING POINT WORK AREA-DIMENSION 150                    PBLK
000068      C IPFTF    FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST)       PBLK
000069      C IPTYPE   TYPE WHEAT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)           JIM
000070      C IPERR    ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT)      PBLK
000071      C IBCD     HOLLERITH ARRAY/0,1,2,3,4,5,6,7,8,9/ /-/0/*/,/            PBLK
000072      C ITEN     POWERS OF TEN (1,10,100,1000,10000)                       PBLK
000073      C IPWRK    INTEGER WORK AREA-DIMENSION 150                           PBLK
000074      C IPEOF    END OF FILE FLAG (TRUE=END FILE)                          PBLK
000075      C IPEFLG   FATAL ERROR FLAG FOR ERROR 1-13 (T=FATAL)                 PBLK
000076      C                                                                    SBLK
000077      C COMMON BLOCK SBLK-SEGREFER FILE HEADER AND DATA RECORD VARIABLES    SBLK
000078      C                                                                    SBLK
000079            COMMON/SBLK/                                                   SBLK
000080           1SCRN,SFNAM1(2),SFNAM2(2),SFTRL,SCOUN,SPCSW,SPCWW,SSANG(150),   SBLK-
000081           2ISSCAS,ISDATE,ISNDAY,ISNAC,ISREG,ISZONE,ISSTRA,                SBLK
000082           3ISUBS, ISSEG,ISTRN,ISRDAY(150),ISPCC(150),NVEH
000083           4,IORDER(150),IKDAY(150)                                                    *NEW
000084      C NAME     DESCRIPTION                                               SBLK     **-1
000085      C SCRN     SEGREFER CURRENT RECORD NO.                               SBLK
000086      C SFNAM1   SEGREFER FILE NAME PROTOTYPE*SEGREFER*                    SBLK
000087      C SFNAM2   SEGREFER FILE NAME FROM LABEL                             SBLK
000088      C SFTRL    TRAILER ID *ZZZZ*                                         SBLK
000089      C SCOUN    SEGREFER COUNTRY ID (F6.2)                                SBLK
000090      C SPCSW    PERCENT SPRING WHEAT(F6.2)                                SBLK
000091      C SPCWW    PERCENT WINTER WHEAT (F6.2)                               JIM
000092      C SSANG    SUN ANGLE(IN RADIANS) FOR EACH OF 150 SEGMENTS            SBLK
000093      C ISSCAS   SEGREFER CASE NO. FROM LABEL                              SBLK
000094      C ISDATE   SEGREFER REFERENCE DATE-ZULU-                             SBLK
000095      C ISNDAY   NO. DAYS IN STUDY (1-426)                                 SBLK
000096      C ISNAC    NO. ACQUISITIONS IN RECORD (0-150)                        SBLK
000097      C ISREG    SEGREFER REGION ID                                        SBLK
000098      C ISZONE   SEGREFER ZONE   ID                                        SBLK
000099      C ISSTRA   SEGREFER STRATA ID                                        SBLK
000100      C ISUBS    SEGREFER SUBSTRATA ID                                     SBLK
000101      C ISSEG    SEGREFER SEGMENT ID                                       SBLK
000102      C ISTRN    TRAINING SEGMENT INDICATOR(0=NORMAL,1=TRAINING)           SBLK
000103      C ISRDAY   ACQUISITION DAY FOR 150 ACQUISITIONS-ZULU-                SBLK
000104      C          IF ZERO, END OF ALL ACQUISITIONS IN RECORD                 SBLK
000105      C ISPCC    PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS             SBLK
000106      C NVEH     NO OF VEHICLES IN THE SWATH TABLE
000107      C IORDER   ORIGINAL ORDER OF ACCESS DATES AS READ FROM SEQ. REF. FILE JIM
000108      C                                                                    CBLK
000109      C COMMON BLOCK CBLK-CROPKIND HEADER AND DATA                         CBLK
000110      C                                                                    CBLK
000111            COMMON /CBLK/                                                  CBLK
000112           1CCRN,CFNAM1(2),CFNAM2(2),CCOUN,                                CBLK
000113           2ICCASE,ICKIND,ICREG,ICZONE,ICSTRA,ICSUBS,ICLON(8),ICHI(8)      CBLK
000114      C NAME     DESCRIPTION                                               CBLK
000115      C CCRN     CURRENT CROPKIND RECORD NO.                               CBLK
000116      C CFNAM1   FILE NAME *CROPKIND*(2A4)                                 CBLK
000117      C CFNAM2   LABEL FILE NAME-MUST MATCH CFNAM1                         CBLK
000118      C CCOUN    CROPKIND COUNTRY ID (A4)                                  CBLK
```

```
000119      C ICCASE  CROPWIND CASE NO. FROM LABEL                                  CBLK
000120      C ICWIND  NO. OF CROP WINDOWS (8)                                       CBLK
000121      C ICREG   CROPWIND REGION ID                                            CBLK
000122      C ICZONE  CROPWIND ZONE ID                                              CBLK
000123      C ICSTRA  CROPWIND STRATA ID                                            CBLK
000124      C ICSUBS  CROPWIND SUBSTRATA ID                                         CBLK
000125      C ICLOW   START DATE OF 8 WINDOWS ZULU=RELATIVE JAN 1950                CBLK
000126      C ICHI    STOP  DATE OF 8 WINDOWS ZULU=RELATIVE JAN 1950                CBLK
000127            DIMENSION I(150),X(150),ITEMP1(8),ITEMP2(8)                       AREAD
000128            EQUIVALENCE (IPWRK,I)                                             AREAD
000129            EQUIVALENCE (XPWRK,X)                                             AREAD
000130            IACCT = 75*NVER                                                   AREAD
000131      C               READ A RECORD FROM SEGREFER FILE                        AREAD
000132       10   READ (ISFILE)(SCOUN,ISREG,ISZONE,                                 AREAD
000133           1ISSTRA,ISUBS,ISSLG,ISTRN,ISPRW,
000134           2(ISRDAY(J),SSANG(J),ISPCC(J),X(J),J=1,IACCT))
000135      C                                                                       JIM
000136      C               INCREMENT RECORDS READ  AND TEST TRAILER                AREAD
000137            SCRN=SCRN+1                                                       AREAD
000138            IF(SCOUN.EQ.SFTRL) GOTO 200                                       AREAD
000139      C               TEST INPUT COUTRY=SEGREFER COUNTRY. IF YES,CONTINUE     AREAD
000140            IF(XICOUN.EQ.SCOUN) GOTO 21                                       AREAD
000141      C               IF NO, ERROR-ALL COUNTRIES (SEGREFER,CROPWIND,INPUT)    AREAD
000142      C               MUST BE EQUAL-EXIT  TO  END FILE EXIT                   AREAD
000143       20   CALL ABARE(5)                                                     AREAD
000144            GOTO 200                                                          AREAD
000145      C               IDENTIFY TYPE OF WHEAT. 1 = WINTER, 5 = SPRING          JIM
000146      C               IF NEITHER, ERROR 15.                                   JIM
000147         21 CONTINUE                                                          SFT1
000148            IF(ISPRW .EQ. 0)GO TO 215
000149            IPTYPE = 5
000150            GO TO 50
000151         215 IPTYPE = 1
000152      C               COMPARE INPUT INPUT ID  FIELDS WITH SEGREFER ID FIELDS  AREAD
000153      C                       GT SKIP SEGREFER RECORD                         AREAD
000154      C                       EQ FIND CROPWIND RECORD                         AREAD
000155      C                       LT, FINISHED,TAKE END FILE EXIT                 AREAD
000156       50   IF(ISREG-IREG)10,55,200                                           AREAD
000157         55 IF(IZONE .EQ. 0)GO TO 70                                          AREAD
000158       60   IF(ISZONE-IZONE)10,70,200                                         AREAD
000159      C               PROPER SEGREF RECORD,COMPARE WITH CROP WIND RECORD       AREAD
000160      C                       GT,READ CROPWIND TO GET NEW RECORD.             AREAD
000161      C                       LT,ERROR-CROPWIND IS MISSING ZONE               AREAD
000162      C                       EQ,EXIT-JOB IS  COMPLETE                        AREAD
000163       70   IF(ISREG-ICREG)120,90,80                                          AREAD
000164      C               READ RECORD FROM CROP WIND FILE                         AREAD
000165       80   READ(ICFILE)(CCOUN,ICREG,ICZONE,ICSTRA,ICSUBS,                   AREAD
000166           2(ITEMP1(N),N=1,8),I(1),(I(J),J=2,6),(ITEMP2(N),N=1,8),I(7),
000167           3(I(J),J=8,12))
000168      C                                                                       JIM
000169            DO 82 L=1,4                                                       AREAD
000170            J = L-1                                                           AREAD
000171            ICLOW(L) = ITEMP1(L+J)                                            AREAD
000172            ICLOW(L+4) = ITEMP2(L+J)                                          AREAD
000173            ICHI(L) = ITEMP1(2+L)                                             AREAD
000174            ICHI(L+4) = ITEMP2(2+L)                                           AREAD
000175         82 CONTINUE                                                          AREAD
000176      C               INCREMENT NO. RECORDS READ AND CHECK FOR TRAILER        AREAD
000177            CCRN=CCRN+1                                                       AREAD
000178            IF(SFTRL.EQ.CCOUN) GOTO 200                                       AREAD
```

```
000179          C               TEST CROPWIND COUNTRY LEGAL,IF NOT ERROR                AREAD
000180          C                                                                       JIM
000181          C                       TEMPORARY BYPASS FOR CHECKOUT                   JIM
000182                IF ( XICOUN .EQ. 4HUSAI .AND. CCOUN .EQ. 4HUSA )  GO TO 90        JIM
000183                IF(CCOUN.NE.XICOUN) GOTO 20                                       AREAD
000184           90   IF(ISZONE-ICZONE)120,100,80                                       AREAD
000185          100   IF(ISSTRA-ICSTRA) 120,110,80                                      AREAD
000186          110   IF(ISUBS-ICSUBS) 120,210,80                                       AREAD.
000187          C               CROPWIND MISSING A ZONE FOR SEGREFER-PRINT ERROR        AREAD
000188          120   CALL ABARF(12)                                                    AREAD
000189                GOTO 10                                                           AREAD
000190          C               END OF FILE EXIT                                        AREAD
000191            200 IPEOF = .TRUE.                                                    SET1
000192                GO TO 900                                                         JIM
000193          C                                                                       JIM
000194          C               COUNT NO. OF ACCESS DATES AND ARRANGE THEM IN INCREASINGJIM
000195          C               ORDER IRRESPECTIVE OF VEHICLE NUMBER.                   JIM
000196           210  NDAYS= 0                                                         JIM
000197                DO 220 J=1,150                                                    JIM
000198                IF ( ISRDAY(J) .EQ. 0 )  GO TO 230                                JIM
000199                IORDER(J)= J                                                      JIM
000200           220  NDAYS= NDAYS + 1                                                 JIM
000201          C                                                                       JIM
000202            230 DO 225 IZ=1,150                                                                *NEW
000203                IKDAY(IZ) = ISRDAY(IZ)                                                         *NEW
000204           225  CONTINUE                                                                      *NEW
000205                CALL SORTAG(ISRDAY,1,NDAYS,IORDER)                                             *NEW
000206          C                                                                       JIM          **-1
000207          C               NOW PERMUTE SUN ANGLE AND PERCENT CLOUD COVER ARRAYS TO JIM
000208          C               AGREE WITH SORTED ORDER OF DATES.                       JIM
000209                DO 240 J=1,NDAYS                                                  JIM
000210                XPWRK(J)= SSANG(J)                                                JIM
000211           240  IPWRK(J)= ISPCC(J)                                               JIM
000212                DO 250 J=1,NDAYS                                                  JIM
000213                JJ= IORDER(J)                                                     JIM
000214                SSANG(J)= XPWRK(JJ)                                               JIM
000215           250  ISPCC(J)= IPWRK(JJ)                                              JIM
000216          C                                                                       JIM
000217           900  RETURN                                                           JIM
000218                END                                                               AREAD
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000001            SUBROUTINE AWRTA                                                AWRTA
000002      C*****************************************************************AWRTA
000003      C                                                                     AWRTA
000004      C ROUTINE      AWRTA                                                  AWRTA
000005      C                                                                     AWRTA
000006      C PURPOSE      THIS ROUTINE WRITES THE RECORDS OF THE DATA ACQUISITION AWRTA
000007      C              FILE. THE RECORDS INCLUDE THE HEADER,DATA AND TRAILER  AWRTA
000008      C              RECORDS                                                AWRTA
000009      C                                                                     AWRTA
000010      C LINKAGE      CALL AWRTA                                             AWRTA
000011      C                                                                     AWRTA
000012      C ROUTINES CALLED  NONE                                               AWRTA
000013      C                                                                     AWRTA
000014      C LOCAL VARIABLES NONE                                                AWRTA
000015      C                                                                     AWRTA
000016      C                                                                     AWRTA
000017      C*****************************************************************AWRTA
000018      C                                                                     AWRTA
000019      C                                                                     PBLK
000020      C COMMON BLOCK PBLK-PROGRAM COUNTERS AND CONSTANTS                    PBLK
000021      C                                                                     PBLK
000022            COMMON /PBLK/                                                   PBLK
000023           1XPWRK(150),IPFTF,IPTYPE,IPERR,IBCD(15),IPWRK(150),ITEN(5),      PBLK
000024           2IPLFLG(13),IPEOF                                                PBLK
000025            LOGICAL IPEOF                                                   PBLK
000026      C                                                                     PBLK
000027      C NAME     DESCRIPTION                                                PBLK
000028      C XPWRK    FLOATING POINT WORK AREA-DIMENSION 150                     PBLK
000029      C IPFTF    FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST)        PBLK
000030      C IPTYPE   TYPE WHEAT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)            JIM
000031      C IPERR    ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT).      PBLK
000032      C IBCD     HOLLERITH ARRAY/0/1/2/3/4/5/6///0/9/ /-/0/*/,/             PBLK
000033      C ITEN     POWERS OF TEN (1,10,100,1000,10000)                        PBLK
000034      C IPWRK    INTEGER WORK AREA-DIMENSION 150                            PBLK
000035      C IPEOF    END OF FILE FLAG (TRUE=END FILE)                           PBLK
000036      C IPEFLG   FATAL ERROR FLAG FOR ERROR 1-13 (T=FATAL)                  PBLK
000037      C                                                                     SBLK
000038      C COMMON BLOCK SBLK-SEGREFER FILE HEADER AND DATA RECORD VARIABLES     SBLK
000039      C                                                                     SBLK
000040            COMMON/SBLK/                                                    SBLK
000041           1SCRN,SFNAM1(2),SFNAM2(2),SFTRL,SCOUN,SPCSW,SPCWW,SSANG(150),    SBLK
000042           2ISSCAS,ISDATE,ISNDAY,ISNAC,ISREG,ISZONE,ISSTRA,                 SBLK
000043           3ISUBS, ISSEG,ISTEN,ISRDAY(150),ISPCC(150),NVEH                  *NEW
000044           4,IORDER(150),IKDAY(150)                                         **-1
000045      C NAME     DESCRIPTION                                                SBLK
000046      C SCRN     SEGREFER CURRENT RECORD NO.                                SBLK
000047      C SFNAM1   SEGREFER FILE NAME PROTOTYPE*SEGREFER*                     SBLK
000048      C SFNAM2   SEGREFER FILE NAME FROM LABEL                              SBLK
000049      C SFTRL    TRAILER ID *ZZZZ*                                          SBLK
000050      C SCOUN    SEGREFER COUNTRY ID (F6.2)                                 SBLK
000051      C SPCSW    PERCENT SPRING WHEAT(F6.2)                                 SBLK
000052      C SPCWW    PERCENT WINTER WHEAT (F6.2)                                JIM
000053      C SSANG    SUN ANGLE(IN RADIANS) FOR EACH OF 150 SEGMENTS             SBLK
000054      C ISSCAS   SEGREFER CASE NO. FROM LABEL                               SBLK
000055      C ISDATE   SEGREFER REFERENCE DATE-ZULU-                              SBLK
000056      C ISNDAY   NO. DAYS IN STUDY (1-426)                                  SBLK
000057      C ISNAC    NO. ACQUISITIONS IN RECORD (0-150)                         SBLK
000058      C ISREG    SEGREFER REGION ID                                         SBLK
```

```
000059      C ISZONE  SEGREFER ZONE   ID                                                  SBLK
000060      C ISSTRA  SEGREFER STRATA ID                                                  SBLK
000061      C ISUBS   SEGREFER SUBSTRATA ID                                               SBLK
000062      C ISSEG   SEGREFER SEGMENT ID                                                 SBLK
000063      C ISTRN   TRAINING SEGMENT INDICATOR(0=NORMAL,1=TRAINING)                     SBLK
000064      C ISRDAY  ACQUISITION DAY FOR 150 ACQUISITIONS-ZULU-                          SBLK
000065      C         IF ZERO, END OF ALL ACQUISITIONS IN RECORD                          SBLK
000066      C ISPCC   PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS                       SBLK
000067      C NVEH    NO OF VEHICLES IN THE SWATH TABLE
000068      C IORDER  ORIGINAL ORDER OF ACCESS DATES AS READ FROM SEQ. REF. FILE         JIM
000069      C                                                                           CBLK
000070      C COMMON BLOCK CBLK-CROPWIND HEADER AND DATA                                CBLK
000071      C                                                                           CBLK
000072           COMMON /CBLK/                                                          CBLK
000073          1CCRN,CFNAM1(2),CFNAM2(2),CCOUN,                                        CBLK
000074          2ICCASE,ICWIND,ICREG,ICZONE,ICSTRA,ICSUBS,ICLOW(8),ICHI(8)              CBLK
000075      C NAME    DESCRIPTION                                                       CBLK
000076      C CCRN    CURRENT CROPWIND RECORD NO.                                       CBLK
000077      C CFNAM1  FILE NAME *CROPWIND*(2A4)                                         CBLK
000078      C CFNAM2  LABEL FILE NAME-MUST MATCH CFNAM1                                 CBLK
000079      C CCOUN   CROPWIND COUNTRY ID (A4)                                          CBLK
000080      C ICCASE  CROPWIND CASE NO. FROM LABEL                                      CBLK
000081      C ICWIND  NO. OF CROP WINDOWS (8)                                           CBLK
000082      C ICREG   CROPWIND REGION ID                                                CBLK
000083      C ICZONE  CROPWIND ZONE ID                                                  CBLK
000084      C ICSTRA  CROPWIND STRATA ID                                                CBLK
000085      C ICSUBS  CROPWIND SUBSTRATA ID                                             CBLK
000086      C ICLOW   START DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950                    CBLK
000087      C ICHI    STOP  DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950                    CBLK
000088      C                                                                           ABLK
000089      C COMMON BLOCK ABLK-ACQUISI FILE AND REPORT VARIABLES                       ABLK
000090      C                                                                           ABLK
000091           COMMON/ABLK/                                                           ABLK
000092          1ACRN,AFNAM(2),AFTRL,ANACQ(426),                                        ABLK
000093          2IACNT(4),IATOT,IAPNT(25,4),IAPAG,IANLIN,IACNTL(4)                      JIM
000094      C NAME    DESCRIPTION                                                       ABLK
000095      C ACRN    CURRENT RECORD NO.                                                ABLK
000096      C AFNAM   LABEL FILE NAME*ACQUISI *                                         ABLK
000097      C AFTRL   TRAILER ID *ZZZZ*  A4                                             ABLK
000098      C ANACQ   NO. OF ACQUISITIONS SELECTED FOR EACH OF 426 DAYS                 APLK
000099      C IACNT   TOTAL NO. OF ACCESSES FOR EACH OF 4 WINDOWS                       JIM
000100      C IATOT    TOTAL NO. OF ACQUISITIONS FOR ALL 4 WINDOWS                      ABLK
000101      C   IAPNT    INDEX TO SEGREFER ACQUISITION (1-150) THAT IDENTIFIES          ABLK
000102      C            A SELECTED SEGMENT, IAPNT(I,J)= WINDOW 1 THROUGH 4.            ABLK
000103      C IAPAG    ACQUISITION REPORT CURRENT PAGE NUMBER                           ABLK
000104      C IANLIN   ACQUISITION REPORT NO. LINES REMAINING CURRENT PAGE              APLK
000105      C IACNTL  NO. OF LEGAL ACQUISTIIONS SELECTED FOR EACH OF 4 WINDOWS          JIM
000106      C                                                                           IBLK
000107      C COMMON BLOCK IBLK-MANUAL INPUT PARAMETERS CONSTANTS                       IBLK
000108      C                                                                           IBLK
000109           COMMON /IBLK/                                                          IBLK
000110          1XTITL(12),XTSANG,XICOUN,IWTIL(16),                                     JIM
000111          2IC1D(3),IACASE,ISCASE,IWCASE,IPCC,ISANG,IREG,IZONE,IIFILE,IOFILE,      IBLK
000112          3ISFILE,ICFILE,IAFILE,INFINE,                                           IBLK
000113          4IFRPT,ICRPT                                                            IBLK
000114           LOGICAL IPRPT,ICRPT                                                    IBLK
000115      C NAME    DESCRIPTION                                                       IBLK
000116      C XTITL   REPORT CASE ORIENTED PAGE TITLE FROM INPUT(12A4)                  IBLK
000117      C XTSANG  WINDOWN SUN ANGLE IN RADIANS FROM INPUT                           IBLK
000118      C XICOUN  COUNTRY ID FROM INPUT (A4)                                        IBLK
```

```
000119      C IWTTL   CROP WINDOW TITLES.  TITLE FOR EACH WINDOW OCCUPIES 8 WORDS   JIM
000120      C           AT 2 CHARACTERS PER WORD.                                   JIM
000121      C ICID    COLUMNS 75-80 OF INPUT CARD 3A2                               IBLK
000122      C IACASE  ACQUISI FILE CASE NO. FROM INPUT                              IBLK
000123      C ISCASE  SEGREFER CASE NO. FROM INPUT                                  IBLK
000124      C IWCASE  CROPWIND CASE NO. FROM INPUT                                  IBLK
000125      C IPCC    MAXIMUM PERCENT CLOUD COVER * 10=TO BE USED IN SEGMENT SELECT IBLK
000126      C ISANG   MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (I4)              IBLK
000127      C IREG    REGION ID FROM INPUT (I3)                                     IBLK
000128      C IZONE   ZONE ID   FROM INPUT (I3)                                     IBLK
000129      C IIFILE  LOGICAL UNIT NO. FOR CARD INPUT                               IBLK
000130      C IOFILE  LOGICAL UNIT NO. FOR REPORT OUTPUT                            IBLK
000131      C ISFILE  LOGICAL UNIT NO. FOR SEGREFER DATA FILE                       IBLK
000132      C ICFILE  LOGICAL UNIT NO. FOR CROPWINDOW FILE                          IBLK
000133      C IAFILE  LOGICAL UNIT NO. FOR ACQUISITION FILE                         IBLK
000134      C INLINE  NO. LINES PER PAGE                                            IBLK
000135      C IPRPT   PROCESSING LOAD REPORT FLAG (T=PRINT REPORT)                  IBLK
000136      C ICRPT   ACQUISITION (CROP WINDOW) REPORT FLAG (T=PRINT REPORT)        IBLK
000137            DIMENSION KAPNT(100),ITEMP1(25),ITEMP2(25)                        AWRTA
000138            DIMENSION KX(100)
000139            EQUIVALENCE(KAPNT(1),IAPNT(1,1))                                  SET1
000140      C                 TEST FIRST TIME FLAG                                  AWRTA
000141      C                       LT 0, WRITE TRAILER                             AWRTA
000142      C                       EQ 0, WRITE HEADER                              AWRTA
000143      C                       GT 0, WRITE ACQUISITION RECORD                  AWRTA
000144            IF (IPFTF)20,10,30                                                AWRTA
000145      C                 WRITE HEADER RECORD                                   AWRTA
000146       10   DO 11 I=1,68                                                      JIM
000147            IPWRK(I)=0                                                        AWRTA
000148       11   CONTINUE                                                          AWRTA
000149            IPNCH= 4                                                          JIM
000150            WRITE (IAFILE)  AFNAM,IACASE,ISNDAY,IPNCH,(IWTTL(L),L=1,16)       JIM
000151           2 ,ICCASE,ISCASE, (IPWRK(L),L=1,84)                                JIM
000152            IPFTF= 1                                                          JIM
000153            ACRN=0                                                            AWRTA
000154            GOTO 40                                                           AWRTA
000155      C                 WRITE TRAILER- SET UP FAKE DETAIL RECORD              AWRTA
000156       20   DO 21 I=1,100                                                     AWRTA
000157            KX(I) = 0
000158       21   CONTINUE                                                          AWRTA
000159            SCOUN=AFTRL                                                       AWRTA
000160            ISREG=999                                                         AWRTA
000161            ISZONE=999                                                        AWRTA
000162            GO TO 65
000163      C                                                                       JIM
000164       30   CONTINUE                                                          JIM
000165            DO 60 I=1,100
000166            IX = KAPNT(I)
000167            IF(IX .NE. 0)GO TO 45
000168            KX(I) = 0
000169            GO TO 60
000170         45 KX(I) = IKDAY(IX)                                                 *NEW
000171         60 CONTINUE                                                         **-1
000172         65 WRITE(IAFILE)(SCOUN,ISREG,ISZONE,ISSTRA,ISUBS,ISSEG,
000173           1(KX(I),I=1,100),IATOT)
000174            IF(IPFTF .LT. 0) END FILE IAFILE
000175      C                 INCREMENT RECORD COUNT                                AWRTA
000176       40   ACRN=ACRN+1                                                       AWRTA
000177            RETURN                                                            AWRTA
000178            END                                                               AWRTA
```

```
• FLT BINBCD,1,751008, 68224   , 1

000001            SUBROUTINE BINBCD(IVAL,NDIG,IOUT)                                    BINBCD
000002            DIMENSION IOUT(6)                                                    BINBCD
000003          C*****************************************************************BINBCD
000004          C                                                                      BINBCD
000005          C ROUTINE       BINBCD                                                 BINBCD
000006          C                                                                      BINBCD
000007          C PURPOSE       TO CONVERT AND INTEGER TO DISPLAY CODE AND PLACE CHAR. BINBCD
000008          C               STRING OUTPUT IN AN ARRAY  NAI (N IS INPUT)            BINBCD
000009          C                                                                      BINBCD
000010          C LINKAGE       CALL BINBCD (IVAL,NDIG,IOUT)                           BINBCD
000011          C                    IVAL, BINARY INTEGER                              BINBCD
000012          C                    NDIG, NO. OF DIGITS TO OUTPUT                     BINBCD
000013          C                    IOUT, ORIGIN OF THE OUTPUT CHARACTER ARRAY        BINBCD
000014          C                                                                      BINBCD
000015          C ROUTINES CALLED   NONE                                               BINBCD
000016          C                                                                      BINBCD
000017          C LOCAL VARIABLES                                                      BINBCD
000018          C                    I      SUBSCRIPT INTO OUTPUT ARRAY                BINBCD
000019          C                    J      CURRENT VALUE OF LEAST SIG. DIGITS         BINBCD
000020          C                    K      NO. DIGITS REMAINING TO PROCESS            BINBCD
000021          C                    N      VALUE OF CURRENT MOST SIGNIFICANT DIGIT    BINBCD
000022          C                                                                      BINBCD
000023          C                                                                      BINBCD
000024          C                                                                      BINBCD
000025          C                                                                      BINBCD
000026          C COMMENTS      ALGORITHM IS DESCRIBED AS FOLLOWS                      BINBCD
000027          C                    0      PRODUCES RIGHT JUSTIFIED 0                 BINBCD
000028          C                    -      PRODUCES LEFT JUSTIFIED -                  BINBCD
000029          C                    *      PRODUCED FOR OUT OF RANGE                  BINBCD
000030          C                    B      BLANK  PRODUCED FOR LEADING ZEROS          BINBCD
000031          C                    NO. GENERATED FROM LEFT TO RIGHT BY REDUCING     BINBCD
000032          C                         NO. BY DECREASING POWERS OF 10               BINBCD
000033          C                                                                      BINBCD
000034          C*****************************************************************BINBCD
000035          C                                                                      PBLK
000036          C COMMON BLOCK PBLK-PROGRAM COUNTERS AND CONSTANTS                     PBLK
000037          C                                                                      PBLK
000038                COMMON /PBLK/                                                    PBLK
000039               1XPWRK(150),IPETF,IPTYPE,IPERR,IBCD(15),IPWRK(150),ITEN(5),      PBLK
000040               2IPEFLG(15),IPEOF                                                PBLK
000041                LOGICAL IPEOF                                                   *NEW
000042          C                                                                      PBLK    **-1
000043          C NAME     DESCRIPTION                                                 PBLK
000044          C XPWRK    FLOATING POINT WORK AREA-DIMENSION 150                      PBLK
000045          C IPETF    FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST)         PBLK
000046          C IPTYPE   TYPE WHEAT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)             JIM
000047          C IPERR    ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT)        PBLK
000048          C IBCD     HOLLERITH ARRAY/0/1/2/3/4/5/6/7/8/9/ /-/0/*/,/              PBLK
000049          C ITEN     POWERS OF TEN (1,10,100,1000,10000)                         PBLK
000050          C IPWRK    INTEGER WORK AREA-DIMENSION 150                             PBLK
000051          C IPEOF    END OF FILE FLAG (TRUE=END FILE)                            PBLK
000052          C IPEFLG   FATAL ERROR FLAG FOR ERROR 1-15 (T=FATAL)                   PBLK
000053          C                INITIALIZE LOCAL VARIABLES                            BINBCD
000054                J=IVAL                                                           BINBCD
000055                I=1                                                              BINBCD
000056                K= NDIG                                                          JIM
000057                IBCD(1)=IBCD(11)                                                 BINBCD
000058          C                TEST INPUT VALUE 0,NEGETIVE, OR OUT OF RANGE          BINBCD
```

```
000059          IF(J)10,20,30                                          BINBCD
000060      C                   IVALUE IS NEGETIVE                     BINBCD
000061       10 IOUT(1)=IBCD(12)                                       BINBCD
000062          K=K-1                                                  BINBCD
000063          I=I+1                                                  BINBCD
000064          J=-J                                                   BINBCD
000065          GOTO 30                                                BINBCD
000066      C                   IVALUE IS ZERO                         BINBCD
000067       20 IOUT(NDIG)=IBCD(13)                                    BINBCD
000068          GOTO 100                                               BINBCD
000069      C                   TEST INPUT VALUE IN RANGE              BINBCD
000070       30 N= J - ITEN(K+1)                                       JIM
000071          IF(N.LT.0) GOTO 50                                     BINBCD
000072      C                   INPUT VALUE OUT OF RANGE               BINBCD
000073       40 IOUT(NDIG)=IBCD(14)                                    BINBCD
000074          GOTO 100                                               BINBCD
000075      C                   CONVERT EACH DIGIT FROM LEFT TO RIGHT  BINBCD
000076          50 II=1                                                BINBCD
000077          DO 70 I=II,NDIG,1                                      BINBCD
000078          N=J/ITEN(K)                                            BINBCD
000079          IOUT(I)=IBCD(N+1)                                      BINBCD
000080          N=N*ITEN(K)                                            BINBCD
000081          J=J-N                                                  BINBCD
000082          K=K-1                                                  BINBCD
000083          IF(N.LE.0)GOTO 70                                      BINBCD
000084          IBCD(1)= IBCD(13)                                      JIM
000085       70 CONTINUE                                               BINBCD
000086       100 RETURN                                                BINBCD
000087          END                                                    BINBCD
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
* ELT BLKDAT,1,760426, 50355   , 1
000001            BLOCK DATA                                                    BLKDTA
000002      C         BLOCK DATA FOR PBLK,IBLK,ABLK,CBLK,SBLK                   BLKDTA
000003      C                                                                   BLKDTA
000004      C                                                                   PBLK
000005      C COMMON BLOCK PBLK-PROGRAM COUNTERS AND CONSTANTS                  PBLK
000006      C                                                                   PBLK
000007            COMMON /PBLK/                                                 PBLK
000008           1XPWRK(150),IPFTF,IPTYPE,IPERR,IBCD(15),IPWRK(150),ITEN(5),    PBLK
000009           2IPEFLG(13),IPEOF                                              PBLK
000010            LOGICAL IPEOI                                                 PBLK
000011      C                                                                   PBLK
000012      C NAME     DESCRIPTION                                              PBLK
000013      C XPWRK    FLOATING POINT WORK AREA-DIMENSION 150                   PBLK
000014      C IPFTF    FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST)      PBLK
000015      C IPTYPE   TYPE REPORT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)         JTM
000016      C IPERR    ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT)     PBLK
000017      C IBCD     HOLLERITH ARRAY/0/1/2/3/4/5/6///8/9/ /-/0/*/,/           PBLK
000018      C ITEN     POWERS OF TEN (1,10,100,1000,10000)                      PBLK
000019      C IPWRK    INTEGER WORK AREA-DIMENSION 150                          PBLK
000020      C IPEOF    END OF FILE FLAG (TRUE=END FILE)                         PBLK-
000021      C IPEFLG   FATAL ERROR FLAG FOR ERROR 1-13 (T=FATAL)                PBLK
000022            DATA IBCD /1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H ,1H-,1H0,1H*BLKDTA
000023           1,1H,/,ITEN/1,10,100,1000,10000/,IPEFLG/1,1,1,1,1,1,1,1,1,1,0,0/
000024      C                                                                   IBLK
000025      C COMMON BLOCK IBLK-MANUAL INPUT PARAMETERS CONSTANTS               IBLK
000026      C                                                                   IBLK
000027            COMMON /IBLK/                                                 IBLK
000028           1XTITL(12),XISANG,XICOUN,IWITL(16),                            JTM
000029           2ICID(3),IACASE,ISCASE,IWCASE,IPCC,ISANG,IREG,IZONE,IIFILE,IOFILE, IBLK
000030           3ISFILE,ICFILE,IAFILE,INLINE,                                  IBLK
000031           4IPRPT,ICRPT                                                   IBLK
000032            LOGICAL IPRPT,ICRPT                                           IBLK
000033      C NAME     DESCRIPTION                                              IBLK
000034      C XTITL    REPORT CASE ORIENTED PAGE TITLE FROM INPUT(12A4)         IBLK
000035      C XISANG   MINIMUM SUN ANGLE IN RADIANS FROM INPUT                  IBLK
000036      C XICOUN   COUNTRY ID FROM INPUT (A4)                               IBLK
000037      C IWITL    CROP WINDOW TITLES.  TITLE FOR EACH WINDOW OCCUPIES 8 WORDS JTM
000038      C            AT 2 CHARACTERS PER WORD.                              JTM
000039      C ICID     COLUMNS 75-80 OF INPUT CARD 3A2                          IBLK
000040      C IACASE   ACQUISI FILE CASE NO. FROM INPUT                         IBLK
000041      C ISCASE   SEGREGER CASE NO. FROM INPUT                             IBLK
000042      C IWCASE   CROPWIND CASE NO. FROM INPUT                             IBLK
000043      C IPCC     MAXIMUM PERCENT CLOUD COVER * 10-TO BE USED IN SEGMENT SELECT IBLK
000044      C ISANG    MINIMUM DEGREES SUN ANGLE *100 (IN DEGREES) (I4)         IBLK
000045      C IREG     REGION ID FROM INPUT (I3)                                IBLK
000046      C IZONE    ZONE ID   FROM INPUT (I3)                                IBLK
000047      C IIFILE   LOGICAL UNIT NO. FOR CARD INPUT                          IBLK
000048      C IOFILE   LOGICAL UNIT NO. FOR REPORT OUTPUT                       IBLK
000049      C ISFILE   LOGICAL UNIT NO. FOR SEGREGER DATA FILE                  IBLK
000050      C ICFILE   LOGICAL UNIT NO. FOR CROPWINDOW FILE                     IBLK
000051      C IAFILE   LOGICAL UNIT NO. FOR ACQUISITION FILE                    IBLK
000052      C INLINE   NO. LINES PER PAGE                                       IBLK
000053      C IPRPT    PROCESSING LOAD REPORT FLAG (T=PRINT REPORT)             IBLK
000054      C ICRPT    ACQUISITION (CROP WINDOW) REPORT FLAG (T=PRINT REPORT)   IBLK
000055            DATA IIFILE/5/,IOFILE/6/,ISFILE/10/,ICFILE/2/,IAFILE/12/,      BLKDTA
000056           1INLINE/39/,                                                   BLKDTA
000057           1  IWITL/4H****,4HWIND,4HOW 1,4H****,                          JTM
000058           2        4H****,4HWIND,4HOW 2,4H****,                          JTM
```

```
000059            3          4H****,4HWIND,4HOW 3,4H****,                    JTM
000060            4          4H****,4HWIND,4HOW 4,4H****/                    JTM
000061      C                                                                 ABLK
000062      C COMMON BLOCK ABLK-ACQUISI FILE AND REPORT VARIABLES             ABLK
000063      C                                                                 ABLK
000064           COMMON/ABLK/                                                 ABLK
000065          1ACRN,AFNAM(2),AFTRL,ANACQ(426),                              ABLK
000066          2IACNT(4),IATOT,IAPNT(25,4),IAPAG,IANLIN,IACNTL(4)            JTM
000067      C NAME     DESCRIPTION                                            ABLK
000068      C ACRN     CURRENT RECORD NO.                                     ABLK
000069      C AFNAM    LABEL FILE NAME*ACQUISI *                              ABLK
000070      C AFTRL    TRAILER ID *ZZZZ*  A4                                  ABLK
000071      C ANACQ    NO. OF ACQUISITIONS SELECTED FOR EACH OF 426 DAYS      ABLK
000072      C IACNT    TOTAL NO. OF ACCESSES FOR EACH OF 4 WINDOWS            JTM
000073      C IATOT     TOTAL NO. OF ACQUISITIONS FOR ALL 4 WINDOWS           ABLK
000074      C   IAPNT     INDEX TO SEGREFER ACQUISITION (1-150) THAT IDENTIFIES ABLK
000075      C             A SELECTED SEGMENT. IAPNT(I,J)= WINDOW I THROUGH 4. ABLK
000076      C IAPAG    ACQUISITION REPORT CURRENT PAGE NUMBER                 ABLK
000077      C IANLIN   ACQUISITION REPORT NO. LINES REMAINING CURRENT PAGE    ABLK
000078      C IACNTL  NO. OF LEGAL ACQUISITIONS SELECTED FOR EACH OF 4 WINDOWS JTM
000079           DATA AFNAM/4HACQU,4HISI /,AFTRL/4HZZZZ/                      BLKDTA
000080      C                                                                 CBLK
000081      C COMMON BLOCK CBLK-CROPWIND HEADER AND DATA                      CBLK
000082      C                                                                 CBLK
000083           COMMON /CBLK/                                                CBLK
000084          1CCRN,CFNAM1(2),CFNAM2(2),CCOUN,                              CBLK
000085          2ICCASE,ICWIND,ICREG,ICZONE,ICSTRA,ICSUBS,ICLOW(8),ICHI(8)    CBLK
000086      C NAME     DESCRIPTION                                            CBLK
000087      C CCRN     CURRENT CROPWIND RECORD NO.                            CBLK
000088      C CFNAM1   FILE NAME *CROPWIND*(2A4)                              CBLK
000089      C CFNAM2   LABEL FILE NAME-MUST MATCH CFNAM1                      CBLK
000090      C CCOUN    CROPWIND COUNTRY ID (A4)                               CBLK
000091      C ICCASE   CROPWIND CASE NO. FROM LABEL                           CBLK
000092      C ICWIND   NO. OF CROP WINDOWS (8)                                CBLK
000093      C ICREG    CROPWIND REGION ID                                     CBLK
000094      C ICZONE   CROPWIND ZONE ID                                       CBLK
000095      C ICSTRA   CROPWIND STRATA ID                                     CBLK
000096      C ICSUBS   CROPWIND SUBSTRATA ID                                  CBLK
000097      C ICLOW    START DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950         CBLK
000098      C ICHI     STOP  DATE OF 8 WINDOWS ZULU-RELATIVE JAN 1950         CBLK
000099           DATA CFNAM1/4HCROP,4HWIND/                                   BLKDTA
000100      C                                                                 SBLK
000101      C COMMON BLOCK SBLK-SEGREFER FILE HEADER AND DATA RECORD VARIABLES SBLK
000102      C                                                                 SBLK
000103           COMMON/SBLK/                                                 SBLK
000104          1SCRN,SFNAM1(2),SFNAM2(2),SFTRL,SCOUN,SPCSW,SPCWW,SSANG(150), SBLK
000105          2ISSCAS,ISDATE,ISRDAY,ISNAC,ISREG,ISZONE,ISSTRA,              SBLK
000106          3ISUBS,ISSEG,ISTRK,ISRDAY(150),ISPCL(150)                     SBLK
000107          4,IORDER(150),IKDAY(150)                                      *NEW
000108      C NAME     DESCRIPTION                                            SBLK  **-1
000109      C SCRN     SEGREFER CURRENT RECORD NO.                            SBLK
000110      C SFNAM1   SEGREFER FILE NAME PROTOTYPE*SEGREFER*                 SBLK
000111      C SFNAM2   SEGREFER FILE NAME FROM LABEL                          SBLK
000112      C SFTRL    TRAILER ID *ZZZZ*                                      SBLK
000113      C SCOUN    SEGREFER COUNTRY ID (F6.2)                             SBLK
000114      C SPCSW    PERCENT SPRING WHEAT(F6.2)                             SBLK
000115      C SPCWW    PERCENT WINTER WHEAT (F6.2)                            JTM
000116      C SSANG    SUN ANGLE(IN RADIANS) FOR EACH OF 150 SEGMENTS         SBLK
000117      C ISSCAS   SEGREFER CASE NO. FROM LABEL                           SBLK
000118      C ISDATE   SEGREFER REFERENCE DATE-ZULU-                          SBLK
```

```
000119          C ISNDAY  NO. DAYS IN STUDY (1-426)                                  SBLK
000120          C ISNAC   NO. ACQUISITIONS IN RECORD (0-150)                        SBLK
000121          C ISREG   SEGREFER REGION ID                                        SBLK
000122          C ISZONE  SEGREFER ZONE   ID                                        SBLK
000123          C ISSTRA  SEGREFER STRATA ID                                        SBLK
000124          C ISUBS   SEGREFER SUBSTRATA ID                                     SBLK
000125          C ISSEG   SEGREFER SEGMENT ID                                       SBLK
000126          C ISTRN   TRAINING SEGMENT INDICATOR(0=NORMAL,1=TRAINING)           SBLK
000127          C ISRDAY  ACQUISITION DAY FOR 150 ACQUISITIONS-ZULU-                SBLK
000128          C         IF ZERO, END OF ALL ACQUISITIONS IN RECORD                SBLK
000129          C ISPCC   PERCENT CLOUD COVER * 10 FOR 150 ACQUISITIONS             SBLK
000130          C IORDER  ORIGINAL ORDER OF ACCESS DATES AS READ FROM SEQ. REF. FILE JIM
000131                DATA SENAM1/4HSEG ,4HREFE/,SETRL/4HZZZZ/                      SET1
000132          C                                                                   JIM
000133                COMMON  /INPERR/  INPERR                                      JIM
000134          C INPERR  INPUT ERROR FLAG. SET = 1 IN ABARF IF THERE ARE ANY FATAL JIM
000135          C                 ERRORS DETECTED ON THE INPUT CARDS BY AINIL       JIM
000136          C                                                                   JIM
000137                DATA  INPERR  / 0 /                                           JIM
000138                END                                                           BLKDTA
```

```
• ELT FZULU,1,751003, 60350    , 1

000001            SUBROUTINE FZULU(IDATE,IOUT)                                    FZULU
000002          C ************************************************************** FZULU
000003          C                                                                 FZULU
000004          C ROUTINE      FZULU                                              FZULU
000005          C                                                                 FZULU
000006          C PURPOSE      TO CONVERT ZULU DATE TO YEAR MONTH AND DAY         FZULU
000007          C                                                                 FZULU
000008          C LINKAGE      CALL FZULU (IDATE,IOUT)                            FZULU
000009          C                   IDATE,  BINARY INTEGER ZULU DATE              FZULU
000010          C                   IOUT,   CALENDER DATE,DIMENSION 3  FOR INTEGER FZULU
000011          C                           YEAR, MONTH AND DAY RESPECTIVELY      FZULU
000012          C                                                                 FZULU
000013          C ROUTINES CALLED   NONE                                          FZULU
000014          C                                                                 FZULU
000015          C LOCAL VARIABLES                                                 FZULU
000016          C                                                                 FZULU
000017          C                   IDAYS   VECTOR CONTAINING NO. DAYS OF YEAR ON FZULU
000018          C                           LAST DAY OF MONTHS 0 THROUGH 12 FOR NORMAL FZULU
000019          C                           YEAR FOLLOWED BY 13 MONTHS OF LEAP YEAR FZULU
000020          C                                                                 FZULU
000021          C                   NLEAP,  NO. OF LEAP YEARS SINCE 1900 TO SPECIFIED FZULU
000022          C                           DATE                                  FZULU
000023          C                   NFLG,   SUBSCRIPT INTO IDAYS VECTOR           FZULU
000024          C                           START=1, NOT LEAP YEAR                FZULU
000025          C                           START=14, LEAP YEAR                   FZULU
000026          C                   JDAY,   JULIAN DAY OF YEAR                    FZULU
000027          C                   I,      LOCAL USE                             FZULU
000028          C COMMENTS          NONE                                          FZULU
000029          C************************************************************** FZULU
000030                DIMENSION IOUT (3)                                          FZULU
000031                DIMENSION IDAYS(26)
000032                DATA IDAYS/0,31,59,90,120,151,181,212,243,273,304,334,365,  FZULU
000033               1           0,31,60,91,121,152,182,213,244,274,305,335,366/  FZULU
000034                IDATE = IDATE + 18263                                       FZULU
000035                DO 5 I=1,3                                                  FZULU
000036                IOUT(I) = 0                                                 FZULU
000037              5 CONTINUE                                                    FZULU
000038          C                   FIND NO. LEAP YEARS SINCE 1900 AND GET CURRENT YEAR FZULU
000039                NLEAP=(1095+IDATE)/1461                                     FZULU
000040                IOUT(1) =(IDATE-NLEAP)/365
000041          C                   SEE IF CURRENT YEAR ISLEAP YEAR-SET NFLG=1 OR 14 FZULU
000042                I=IOUT(1)/4
000043                I=I*4                                                       FZULU
000044                NFLG=1                                                      FZULU
000045                IF (IOUT(1)-I) 10,10,20
000046           10   NFLG=14                                                     FZULU
000047          C                   JULIAN DAY = ZULU DAY-NO YEARS*365-NO.LEAP YEARS FZULU
000048           20 JDAY=IDATE-(IOUT(1)*365)-NLEAP + 1
000049          C                   GET DAY MONTH FROM TABLE SEARCH               FZULU
000050          C                                                                 FZULU
000051           30   IF(JDAY-IDAYS(NFLG))50,50,40                               FZULU
000052           40   IOUT(2)=IOUT(2)+1                                          FZULU
000053                NFLG=NFLG+1                                                 FZULU
000054                GOTO 30                                                     FZULU
000055           50   NFLG=NFLG-1                                                FZULU
000056                IOUT(3)=JDAY-IDAYS(NFLG)                                    FZULU
000057                RETURN                                                      FZULU
000058                END                                                         FZULU
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000059          C                                                       MAIN
000060           20   IPFTF=-1                                          MAIN
000061          C                    WRITE TRAILER ACQUISITION FILE     MAIN
000062                CALL AWRTA                                        MAIN
000063          C                    PRINT STATISTICS CROP WIND REPORT  MAIN
000064                CALL ACRPT                                        MAIN
000065          C                    PRINT DAILY PRODUCTION REPORT      MAIN
000066                CALL APRPT                                        MAIN
000067                STOP                                              MAIN
000068                END                                               MAIN
```

```
* FLT SACS,1,751008, 68213     , 1
000001          C     PROGRAM MAIN(ITEST=401,OTEST=401,TAPE5=ITEST,TAPE6=OTEST,          MAIN
000002          C    1SEGREF=2400,CROPW=401,ACQUIS=2000,TAPE10=SEGREF,TAPE2=CROPW,       MAIN
000003          C    2TAPE12=ACQUIS)                                                    MAIN
000004          C*******************************************************************MAIN
000005          C                                                                       MAIN
000006          C ROUTINE       SACS                                                    MAIN
000007          C                                                                       MAIN
000008          C PURPOSE       MAIN CONTROL ROUTINE FOR SACS PROGRAM                   MAIN
000009          C                                                                       MAIN
000010          C LINKAGE       ENTER FROM OPERATING SYSTEM                             MAIN
000011          C                                                                       MAIN
000012          C ROUTINES CALLED                                                       MAIN
000013          C               AINIL,  READ INPUT CARDS/PROCESS FILE HEADERS           MAIN
000014          C               AREAD,  SELECT RECORDS FROM SEGREFER/CROPWIND FILES     MAIN
000015          C               AFIND,  FIND ALL SEGMENTS IN SPECIFIED WINDOWS          MAIN
000016          C               AWRIA,  WRITE ACQUISITION FILE RECORD                   MAIN
000017          C               ACRPT,  PRINT ONE SEGMENT ON CROP WINDOW REPORT         MAIN
000018          C               APRPT,  PRINT DAILY PRODUCTION REPORT                   MAIN
000019          C                                                                       MAIN
000020          C LOCAL VARIABLES NONE                                                  MAIN
000021          C                                                                       MAIN
000022          C COMMENTS      NONE                                                    MAIN
000023          C                                                                       MAIN
000024          C*******************************************************************MAIN
000025          C                                                                       PRLK
000026          C COMMON BLOCK PRLK-PROGRAM COUNTERS AND CONSTANTS                      PRLK
000027          C                                                                       PRLK
000028               COMMON /PRLK/                                                      PRLK
000029              1XPWRK(150),IPEIF,IPTYPE,IPERR,IHCD(15),IPWRK(150),ITEN(5),        PRLK
000030              2IPEFLG(15),IPEOF                                                  PRLK
000031               LOGICAL IPEOF                                                     *NEW
000032          C                                                                       PRLK    **-1
000033          C NAME     DESCRIPTION                                                  PRLK
000034          C XPWRK    FLOATING POINT WORK AREA-DIMENSION 150                       PRLK
000035          C IPEIF    FIRST TIME FLAG(0=FIRST TIME, 1=SUBSEQUENT,-1=LAST)          PRLK
000036          C IPTYPE   TYPE WHEAT (0=UNIDENTIFIED, 1=WINTER, 5=SPRING)              JIM
000037          C IPERR    ACQUISITION DAY WHEN IT EXCEEDS MAXIMUM(ERROR PRINT)         PRLK
000038          C IHCD     HOLLERITH ARRAY/0/1/2/3/4/5/6/7/8/9/ /-/0/*/,/               PRLK
000039          C ITEN     POWERS OF TE : (1,10,100,1000,10000)                         PRLK
000040          C IPWRK    INTEGER WORK AREA-DIMENSION 150                              PRLK
000041          C IPEOF    END OF FILE FLAG (TRUE=END FILE)                             PRLK
000042          C IPEFLG   FATAL ERROR FLAG FOR ERROR 1-15 (T=FATAL)                    PRLK
000043          C               CALL INITIALIZATION ROUTINE                             MAIN
000044               CALL AINIL                                                         MAIN
000045          C               READ SEGREFER AND CROPWIND RECORDS.                     MAIN
000046           10  CALL  AREAD                                                       MAIN
000047          C               TEST END OF FILE                                        MAIN
000048               IF(IPEOF)GO TO 20                                                  SET1
000049          C               FIND ACQUISITIONS IN SEG REFER FILE                     MAIN
000050               CALL AFIND                                                         MAIN
000051          C               WRITE ACQUISITION FILE RECORD                           MAIN
000052               CALL AWRIA                                                         MAIN
000053          C               WRITE ACQUISITION REPORT LINE(S)                        MAIN
000054               CALL ACRPT                                                         MAIN
000055          C               LOOP FOR NEXT SEGMENT                                   MAIN
000056               GO TO 10                                                           MAIN
000057          C                                                                       MAIN
000058          C               END OF INPUT FILE                                       MAIN
```

```
* FLT SORTAG,1,751008, 68226   , 1

000001            SUBROUTINE SORTAG(A,II,JJ,TAG)
000002          C SORTS ARRAY A INTO INCREASING ORDER, FROM A(II) TO A(JJ)
000003          C ARRAY TAG IS PERMUTED THE SAME AS ARRAY A
000004          C ORDERING IS BY INTEGER SUBTRACTION. THUS FLOATING POINT
000005          C   NUMBERS MUST BE IN NORMALIZED FORM.
000006          C ARRAYS IU(K) AND IL(K) PERMIT SORTING UP TO 2**(K+1)-1 ELEMENTS
000007          C CDC 6400 TIME IS 3.26 SEC. FOR 10000 RANDOM ITEMS,
000008          C   AND OTHERWISE PROPORTIONAL TO N*LOG(N)
000009          C     R. SINGLETON, SEPTEMBER 1968
000010            DIMENSION A(1),IU(16),IL(16),TAG(1)
000011            INTEGER A,T,TT
000012            M=1
000013            I=II
000014            J=JJ
000015          5 IF(I .GE. J) GO TO 70
000016         10 K=I
000017            IJ=(J+I)/2
000018            T=A(IJ)
000019            IF(A(I) .LE. T) GO TO 20
000020            A(IJ)=A(I)
000021            A(I)=T
000022            T=A(IJ)
000023            TG=TAG(IJ)
000024            TAG(IJ)=TAG(I)
000025            TAG(I)=TG
000026         20 L=J
000027            IF(A(J) .GE. T) GO TO 40
000028            A(IJ)=A(J)
000029            A(J)=T
000030            T=A(IJ)
000031            TG=TAG(IJ)
000032            TAG(IJ)=TAG(J)
000033            TAG(J)=TG
000034            IF(A(I) .LE. T) GO TO 40
000035            A(IJ)=A(I)
000036            A(I)=T
000037            T=A(IJ)
000038            TG=TAG(IJ)
000039            TAG(IJ)=TAG(I)
000040            TAG(I)=TG
000041            GO TO 40
000042         30 A(L)=A(K)
000043            A(K)=TT
000044            TG=TAG(L)
000045            TAG(L)=TAG(K)
000046            TAG(K)=TG
000047         40 L=L-1
000048            IF(A(L) .GT. T) GO TO 40
000049            TT=A(L)
000050         50 K=K+1
000051            IF(A(K) .LT. T) GO TO 50
000052            IF(K .LE. L) GO TO 30
000053            IF(L-I .LE. J-K) GO TO 60
000054            IL(M)=I
000055            IU(M)=L
000056            I=K
000057            M=M+1
000058            GO TO 80
```

```
000059          60 II(M)=K
000060             III(M)=J
000061             J=L
000062             M=M+1
000063             GO TO 80
000064          70 M=M-1
000065             IF(M .EQ. 0) RETURN
000066             I=IL(M)
000067             J=III(M)
000068          80 IF(J-I .GE. 11) GO TO 10
000069             IF(I .EQ. II) GO TO 5
000070             I=I-1
000071          90 I=I+1
000072             IF(I .EQ. J) GO TO 70
000073             I=A(I+1)
000074             IF(A(I) .LE. T) GO TO 90
000075             IG=TAG(I+1)
000076             K=I
000077         100 A(K+1)=A(K)
000078             TAG(K+1)=TAG(K)
000079             K=K-1
000080             IF(T .LT. A(K)) GO TO 100
000081             A(K+1)=T
000082             TAG(K+1)=IG
000083             GO TO 90
000084             END
```

7. TOT

ORIGINAL PAGE IS
OF POOR QUALITY

POUT BOOK VII

Table of Contents

		Page
Part I	Problem Description	203
Part II	Common Block Definitions	236
Part III	List of Subroutines and Subroutine Call Structure	263
Part IV	Subroutine Descriptions and Flowcharts	268
Part V	Subroutine Listings	313

PART I

PROBLEM DESCRIPTION FOR

THE POST PROCESSOR

Table of Contents

		Page
1.0	Scope	1
1.1	Program Capabilities	1
1.2	Method of Program Development	2
1.3	Operational Assumptions	2
2.0	Input	3
2.1	Cards	3
2.2	Files	4
3.0	Processing	11
3.1	Overview	11
3.2	Program Flow	12
3.3	Procedures and Equations	17
3.4	POUT Symbol Table	23
4.0	Output	26
4.1	Printed Data	26
5.0	Error Processing	30
5.1	Input Errors	30
5.2	Processing Errors	31

Problem Description for the Post Processor

1.0 SCOPE

1.1 PROGRAM CAPABILITIES

Depending on the user specified option flags, the POUT program will read from associated LACIE file(s) and produce a printed report.

The four major types of reports that can be produced are:

1. Substrata Reference Data Report - the data is processed from the Substrata Historical File (SUBHST).
2. Population Mean, Standard Deviation and Histogram Reports. There are five parameter types in this group. Each parameter produces its own report.
   a. Population Sampling Error (Segment Truth File - SEGTRU)
   b. Population CAMS Error (CAMS Output File - CAMSF)
   c. Population Yield Error (YES Output File - YESOUT)
   d. Population Area Error (CAS Cum Output File - CASF)
   e. Population Production Error (CAS Cum Output File - CASF)
3. Histograms of Monte Carlo Statistics Reports. There are four parameter types in this group. Each parameter produces its own report except for Confidence Level which produces six reports. All data is processed from the CAS Distribution Output File (CASDIS).
   a. Monte Carlo Area Error
   b. Monte Carlo Production Error
   c. Monte Carlo Yield Error
   d. Confidence Level
      - Area Confidence Level Est/Est
      - Area Confidence Level True/Est

- Area Confidence Level True/WC
- Production Confidence Level Est/Est
- Production Confidence Level True/Est
- Production Confidence Level True/WC

4. Frequency of Sample Segment Acquisitions Report - the data is processed from the Data Acquisition File (ACQUIS).

1.2 METHOD OF PROGRAM DEVELOPMENT

The program will be developed in FORTRAN. The direct access file processing is the only known conversion problem and will be handled in one routine. Modular programming techniques will be used throughout to make the program development, modification and debugging easier.

1.3 OPERATIONAL ASSUMPTIONS

- Only one major type of printed report will be produced per run.
- Population reports will be produced by option for either zone, region or country level.
- For Population or Monte Carlo reports any one, a set, or all of the parameters can be selected in one run.
- The Substrata Historical File will also be required to produce the first three Population reports.
- Printed report control will be by card input.
- All control card input data will be echo printed.
- All control card input data will be checked for errors before any error will cause the processing to terminate in the middle of a case.
- External print units will be in English, an optional override is available for metric units via the control card input.
- The Header Card and the four Data Cards are required input.
- All data to be processed will be obtained from disk (or tape) files.
- All input data files will be checked for correct case numbers.

2.0 INPUT

Program input will be in two forms.

1. Card input for identification and option selection.
2. File input for requested option data.

2.1 CARDS

2.1.1 List of Data Quantities

See Table 2-2 for Input Data Description.

2.1.2 Card Formats

"POUT" is punched in card columns 75-78 of all cards. A sequence number is punched in card columns 79-80.

See Figure 2-1 for Data Card Formats.

2.1.3 Deck Set Up

1. Header Card - sequence 01
2. Data Card 1 - sequence 02
3. Data Card 2 - sequence 03
4. Data Card 3 - sequence 04
5. Data Card 4 - sequence 05

2.1.4 Rules for Entering Data on Cards

2.1.4.1 General

1. Integers must be entered right-justified.
2. Alphanumeric names must be entered left-justified.
3. F format numbers must have the decimal point present, i.e., F6.1 ±XXX.X
4. The card sequence numbers in CC. 79-80 must be present on all cards.

2.1.4.2 Specific Fields

The correspondence between the RPTYPE, PARMTR and ICASIN input values, and the required input files versus the produced reports is shown in Table 2-1 below.

Histogram ranges, PARMTR, BIOWD, WPRTY, IPRD input values are required only if RPTYPE=02 and =03. LEVEL is required only if RPTYPE=02.

Table 2-1

RPTYPE	PARMTR(I) where I =	ICASIN(I) where I =	Req. Input Files	Report
01	-	1	SUBHST	Substrata Reference Data
02	1*	1	SEGTRU	Population Sampling Error
	2*	2	CAMSF	Population CAMS Error
	3*	3	YESOUT	Population Yield Error
	4	4	CASF	Population Area Error
	5	5	CASF	Population Production Error
03	1	1	CASDIS	Monte Carlo Area Error
	2	2	CASDIS	Monte Carlo Production Error
	3	3	CASDIS	Monte Carlo Yield Error
	4	4	CASDIS	Confidence Level
04	-	1	ACQUIS	Frequency of Sample Segment Acquisition

*SUBHST file also required for each report.

2.2 FILES

Any one of the following files, depending on the report option selected, will be input. The Substrata Historical File will always be input when any of the first three Population reports are selected.

SUBHST - Substrata Historical File
SEGTRU - Segment Truth File
CAMSF - CAMS Output File
CASF - CAS Cum Output File
CASDIS - CAS Distribution Output File
ACQUIS - Data Acquisition File
YESOUT - YES Output File

Complete descriptions of all files used by the POUT program can be found in the Reference File Description Document.

Table 2-2. Input Data Description

Name	Symbol	Dimension	Nominal Value	Range	Units	Description
IHEADR	-	12	Blanks	-	-	72 character case header which prints out at the top of every page
RPTYPE	-	1	0	1-4	-	Major type of report identification 01 - Substrata Reference Data Report 02 - Population Mean, Standard Deviation and Histogram Report 03 - Histograms of Monte Carlo Statistics Report 04 - Frequency of Sample Segment Acquisitions Reports
AUNITS	-	1	0	0, 1	-	External print units flag 0 - English units Wheat area value in 10,000 acres Production value in 100,000 bushels Yield value in bushels/acre ≠0 - Metric units Wheat area value in 1000 hectares Production value in 1000 metric tons Yield value in quintals/hectare
START	-	1	-500.0	-999.9 to 999.9	-	Histogram interval start value
INTVL1	-	1	100.0	0.1 to 100.0	-	Histogram interval value in percent
BREAK1	-	1	-100.0	-999.9 to 999.9	-	Histogram breakpoint to change interval value
INTVL2	-	1	5.0	0.1-100.0	-	Histogram interval value in percent
BREAK2	-	1	100.0	-999.9 to 999.9	-	Histogram breakpoint to change interval value

Table 2-2. Input Data Description (cont'd)

Name	Symbol	Dimension	Nominal Value	Range	Units	Description
INTVL3	-	1	100.0	0.1-100.0	-	Histogram interval value in percent
STOP	-	1	500.0	-999.9 to 999.9	-	Histogram interval terminal value A maximum of 51 range intervals is allowed although less may be used. Intervals may not overlap.
PARMTR	-	5	0	0-3	-	Report parameter type option flags for RPTYPE = 02 or 03. =0 - no report; ≠0 - produce report. PARMTR(1)≠0 RPTYPE = 02 Produce Population Sampling Error Report ≠0 RPTYPE = 03 Produce Monte Carlo Area Error Report PARMTR(2)≠0 RPTYPE = 02 Produce Population CAMS Error Report ≠0 RPTYPE = 03 Produce Monte Carlo Production Error Report PARMTR(3)≠0 RPTYPE = 02 Produce Population Yield Error Report ≠0 RPTYPE = 03 Produce Monte Carlo Yield Error Report PARMTR(4)≠0 RPTYPE = 02 Produce Population Area Error Report ≠0 RPTYPE = 03 Produce Confidence Level Report PARMTR(5)≠0 RPTYPE = 02 Produce Population Production Error Report If RPTYPE = 02 or 03 and all PARMTR values are zero, then all reports of that RPTYPE will be produced.

Table 2-2. Input Data Description (cont'd)

Name	Symbol	Dimension	Nominal Value	Range	Units	Description
ICASIN	-	5	0	0-9999	-	Case number identification associated with the required input file. A case number must be input for every report produced. Relationship between RPTYPE, PARMTR and ICASIN follows. ICASIN(1) = case no. RPTYPE = 01 = 02 PARMTR(1)≠0 = 03 PARMTR(1)≠0 = 04 ICASIN(2) = case no. RPTYPE = 02 PARMTR(2)≠0 = 03 PARMTR(2)≠0 ICASIN(3) = case no. RPTYPE = 02 PARMTR(3)≠0 = 03 PARMTR(3)≠0 ICASIN(4) = case no. RPTYPE = 02 PARMTR(4)≠0 = 03 PARMTR(4)≠0 ICASIN(5) = case no. RPTYPE = 02 PARMTR(5)≠0 If all PARMTR values are zero for a RPTYPE = 02 or 03, then all the ICASIN values (case number) must be entered for that RPTYPE value.
LEVEL	-	1	1	0-3	-	Parameter report level indicator for RPTYPE = 2. =1 - reports produced at zone level; =2 - at region level; =3 - at country level.
ICSESH	-	1	0	0-9999	-	Case number identification for the SUBHST input file. Required only when RPTYPE = 2 and PARMTR = 1, 2 or 3.

Table 2-2. Input Data Description (cont'd)

Name	Symbol	Dimension	Nominal Value	Range	Units	Description
BIOWD	-	4	0	0, 1	-	Prediction bio-window flags: BIOWD(n) =1 to process bio-window n =0 otherwise for RPTYPE = 02 and PARMTR(2)≠0 Selects by bio-window in record. PARMTR(3)≠0 Uses last yield date in record. PARMTR(4)≠0 or PARMTR(5)≠0 Selects by bio-window in record. for RPTYPE = 03 Selects by bio-window in record.
WPRTY	-	4	0	0-4	-	Bio-window priorities: List of bio-windows in decreasing order of priority, e.g., 4, 1, 3, 2 or 3, 1, 0, 0. Used only by RPTYPE = 02 and PARMTR(2)≠0 in conjunction with the prediction dates below.
IPRD	-	3, 14	0	year: >64 month: 01-12 day: 01-31	-	Prediction dates (up to 14 dates) for the selection of data for RPTYPE = 02 or 03. IPRD(1, n) = year - 1900 IPRD(2, n) = month IPRD(3, n) = day The dates must be in ascending order. The first zero date terminates the list. For RPTYPE = 02 and PARMTR(2)≠0 Prediction date selected as a function of priority above and latest date less than or equal to the input prediction date. PARMTR(3)≠0 Selects the latest yield date less than or equal to the input prediction date. PARMTR(4) or (5)≠0 and all RPTYPE = 03 the input prediction dates must match with file dates; otherwise a message is printed and that date skipped.

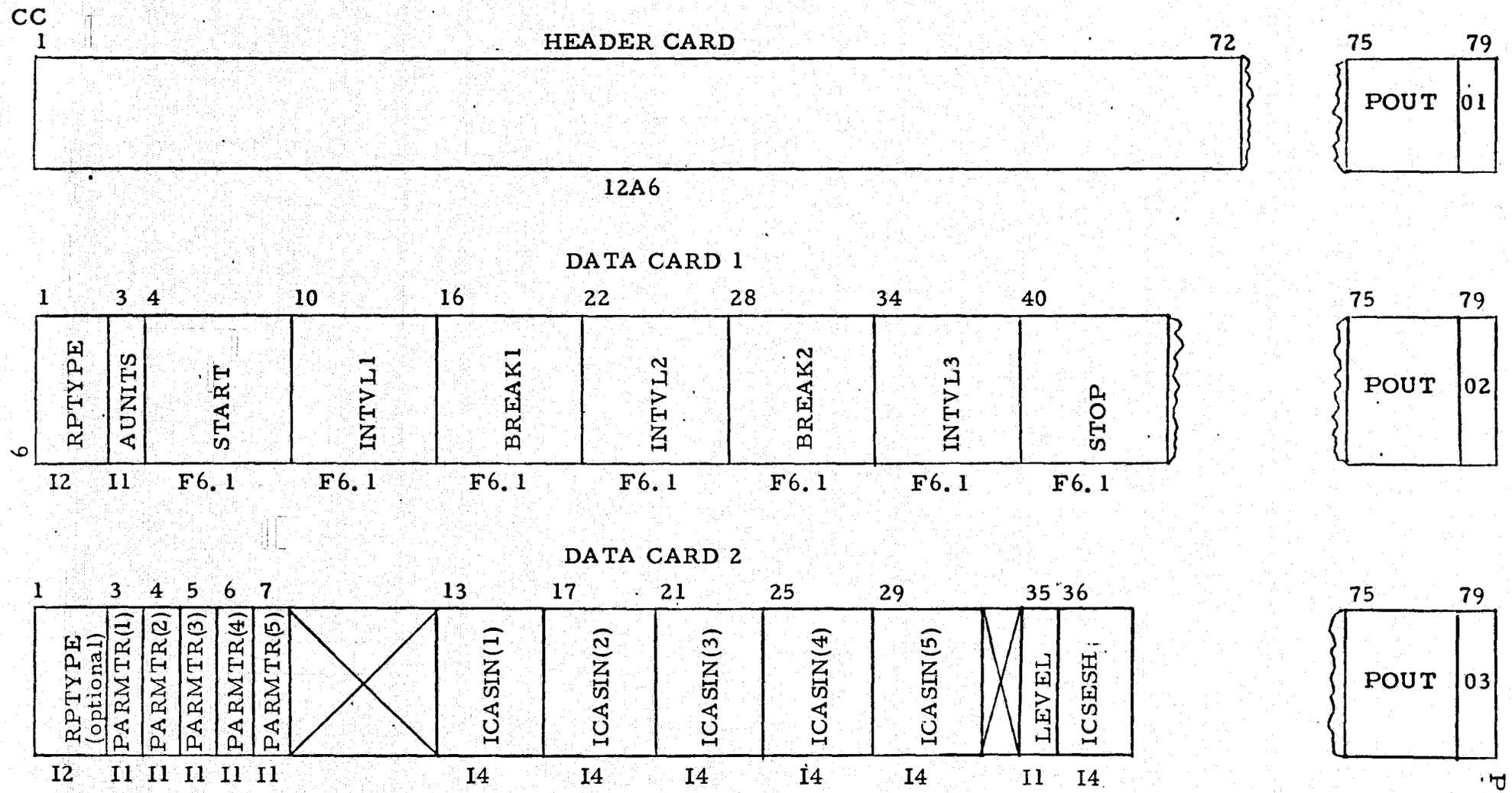
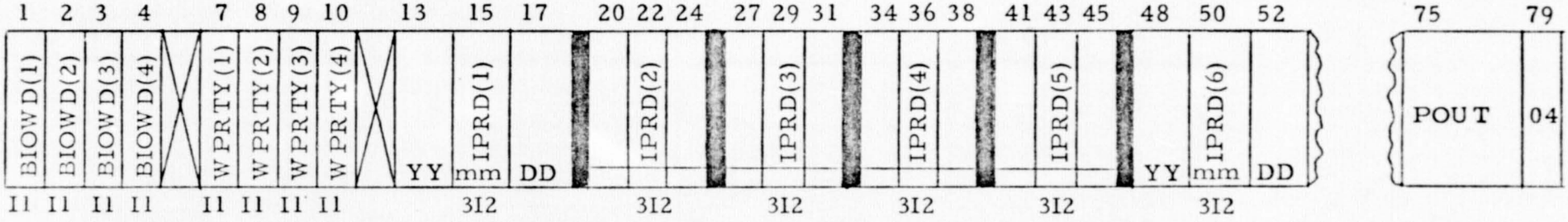


Figure 2-1. Data Card Formats

DATA CARD 3



DATA CARD 4

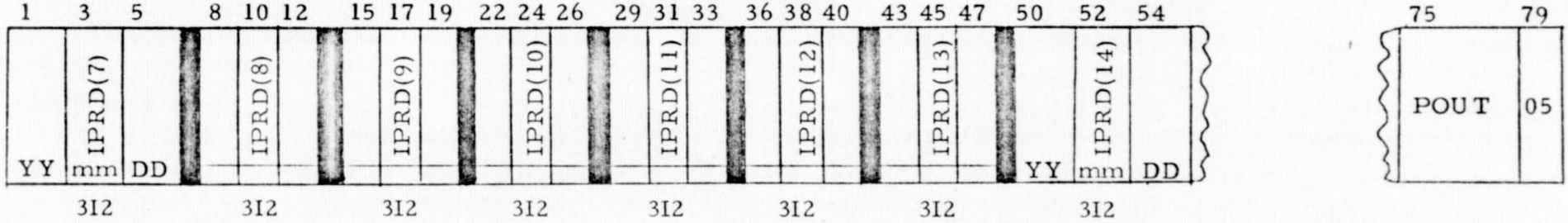
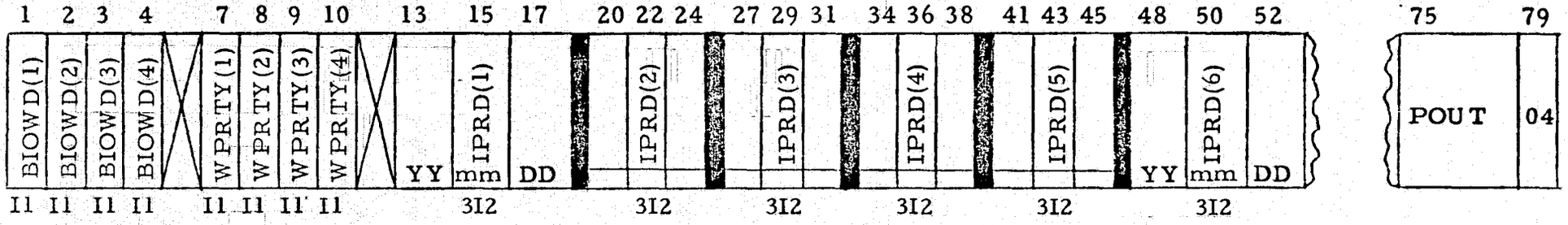


Figure 2-1. Data Card Formats (cont'd)

DATA CARD 3



DATA CARD 4

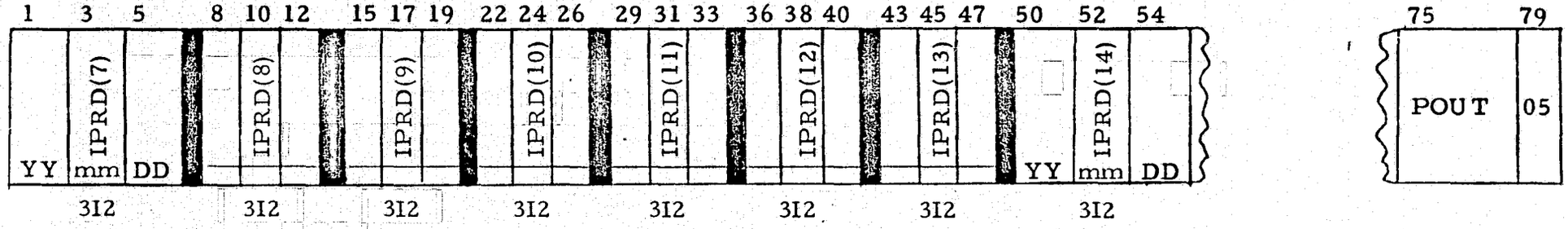


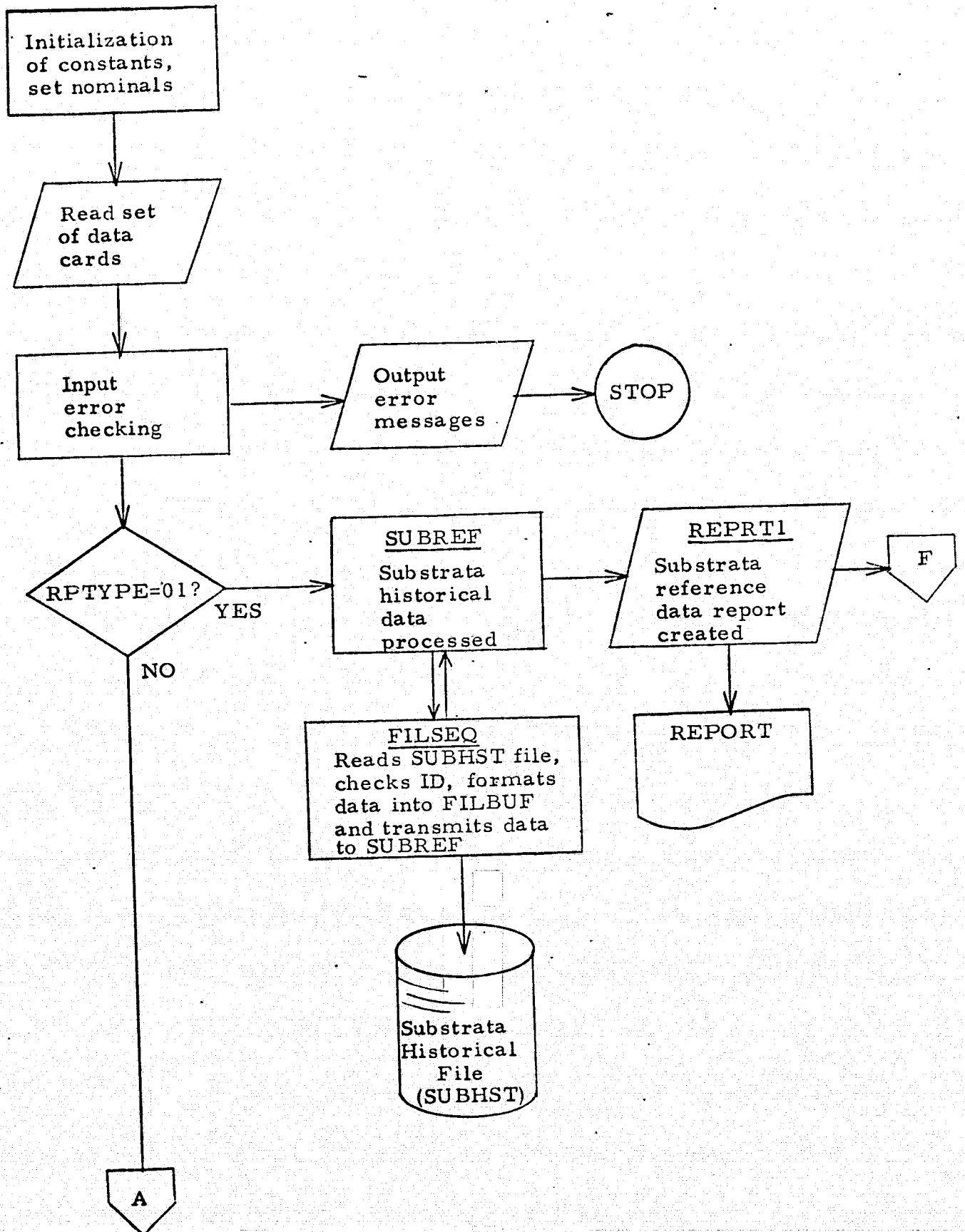
Figure 2-1. Data Card Formats (cont'd)

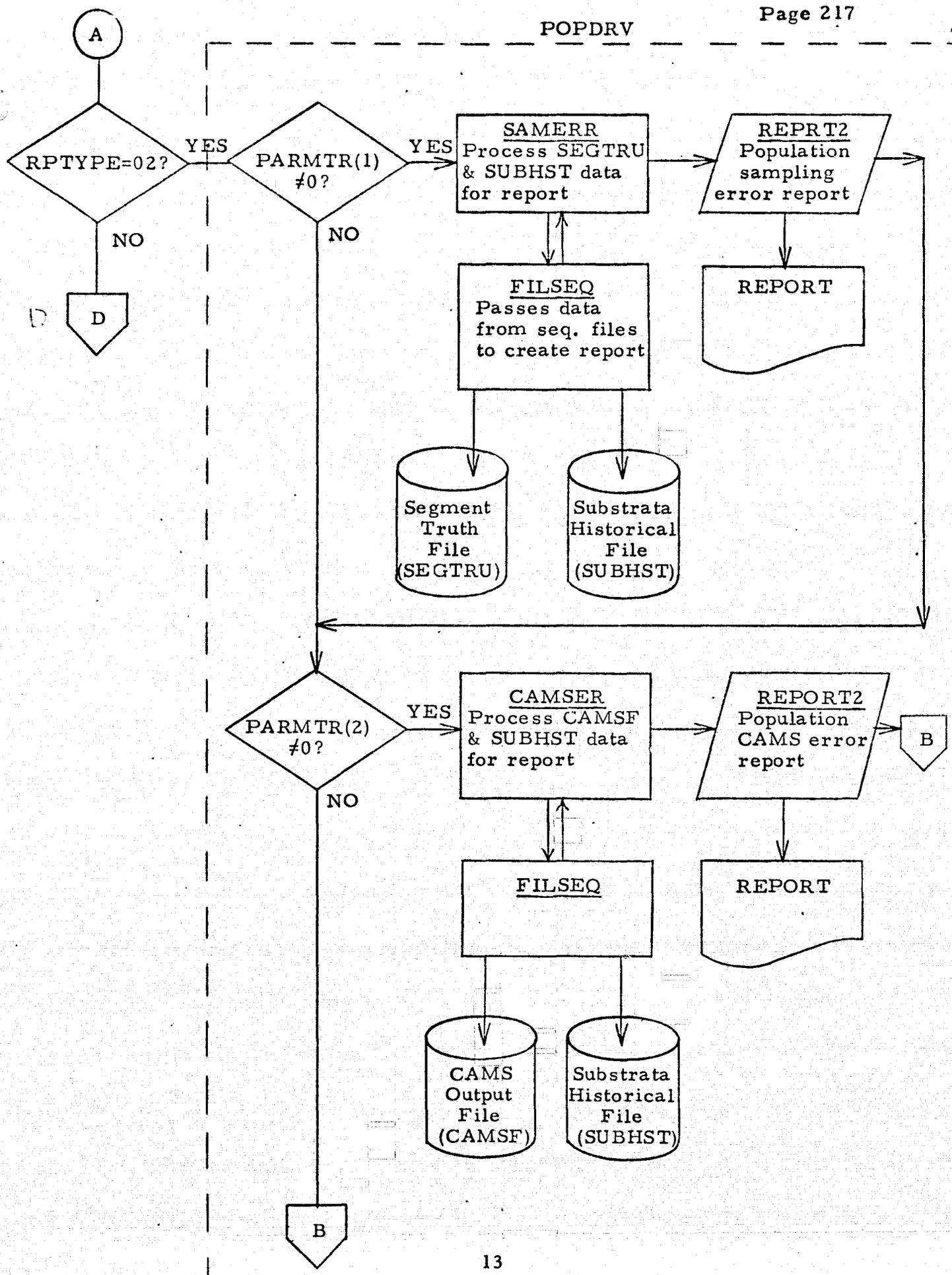
3.0 PROCESSING

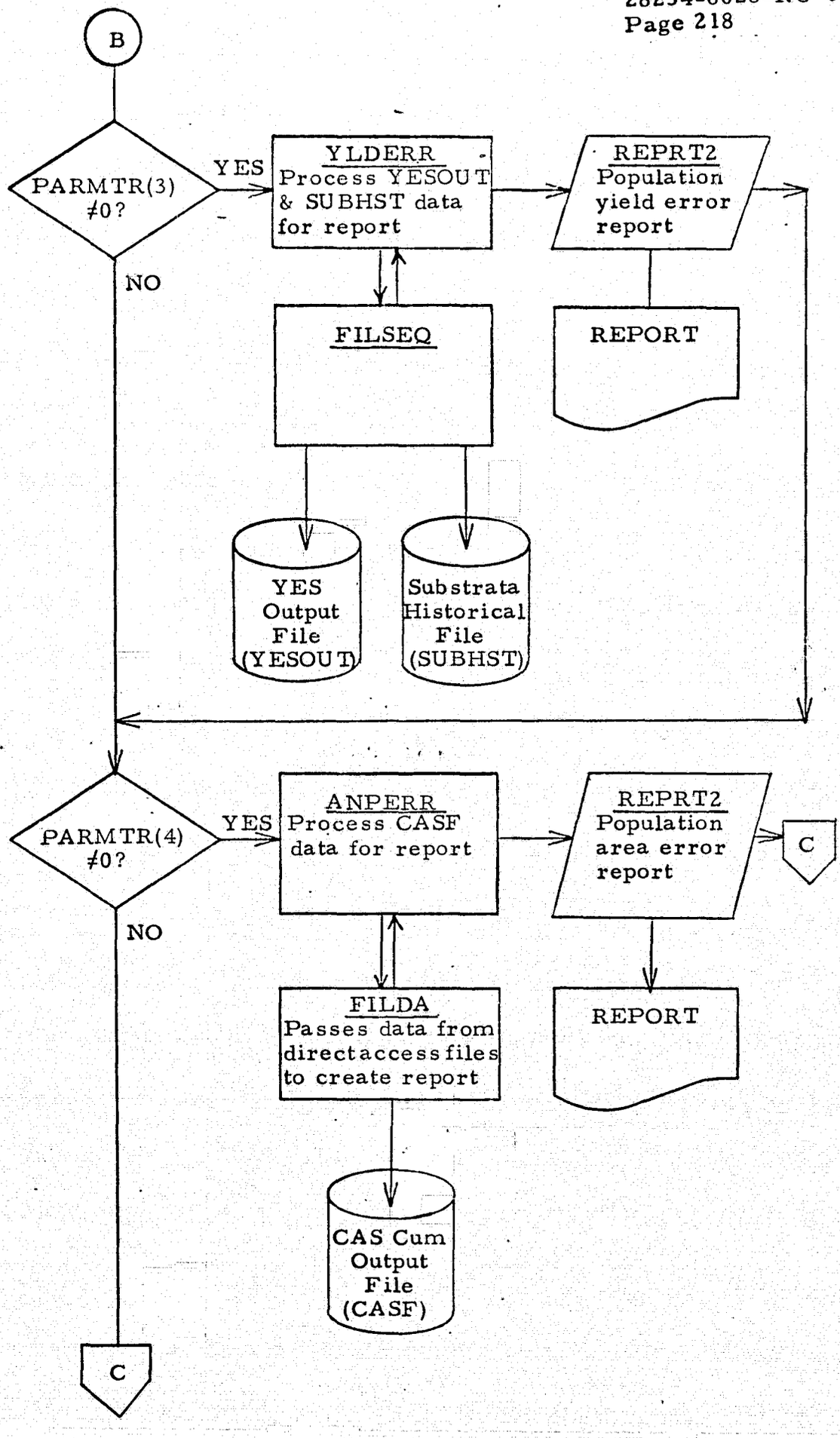
3.1 OVERVIEW

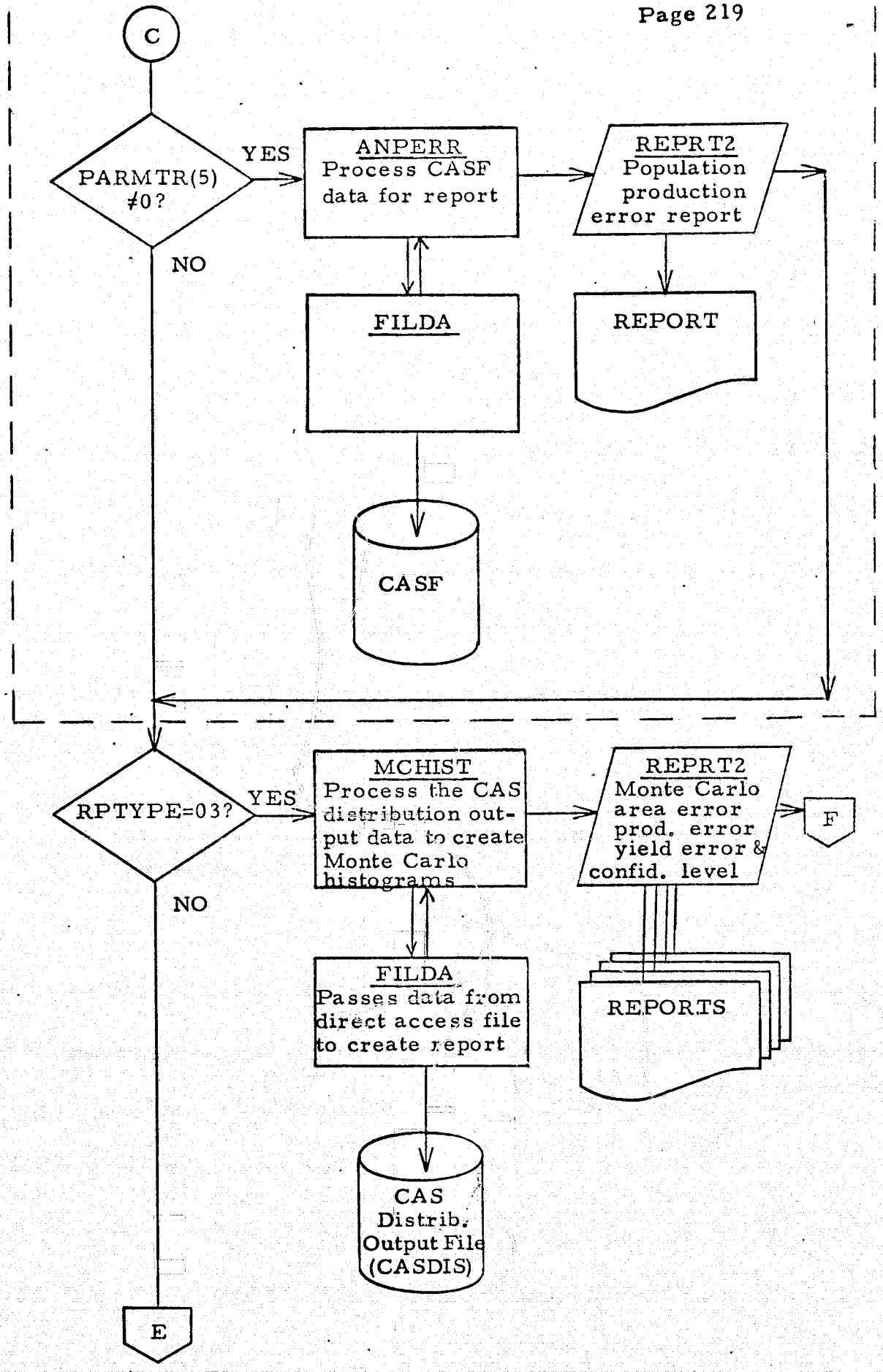
The POUT program is essentially subdivided into four areas of processing. The area executed depends on the major type option selected on the card input. The main program POUT will read the card input and then after error-checking will transfer to the selected option. The required file(s) will be processed and the printed report(s) will be produced.

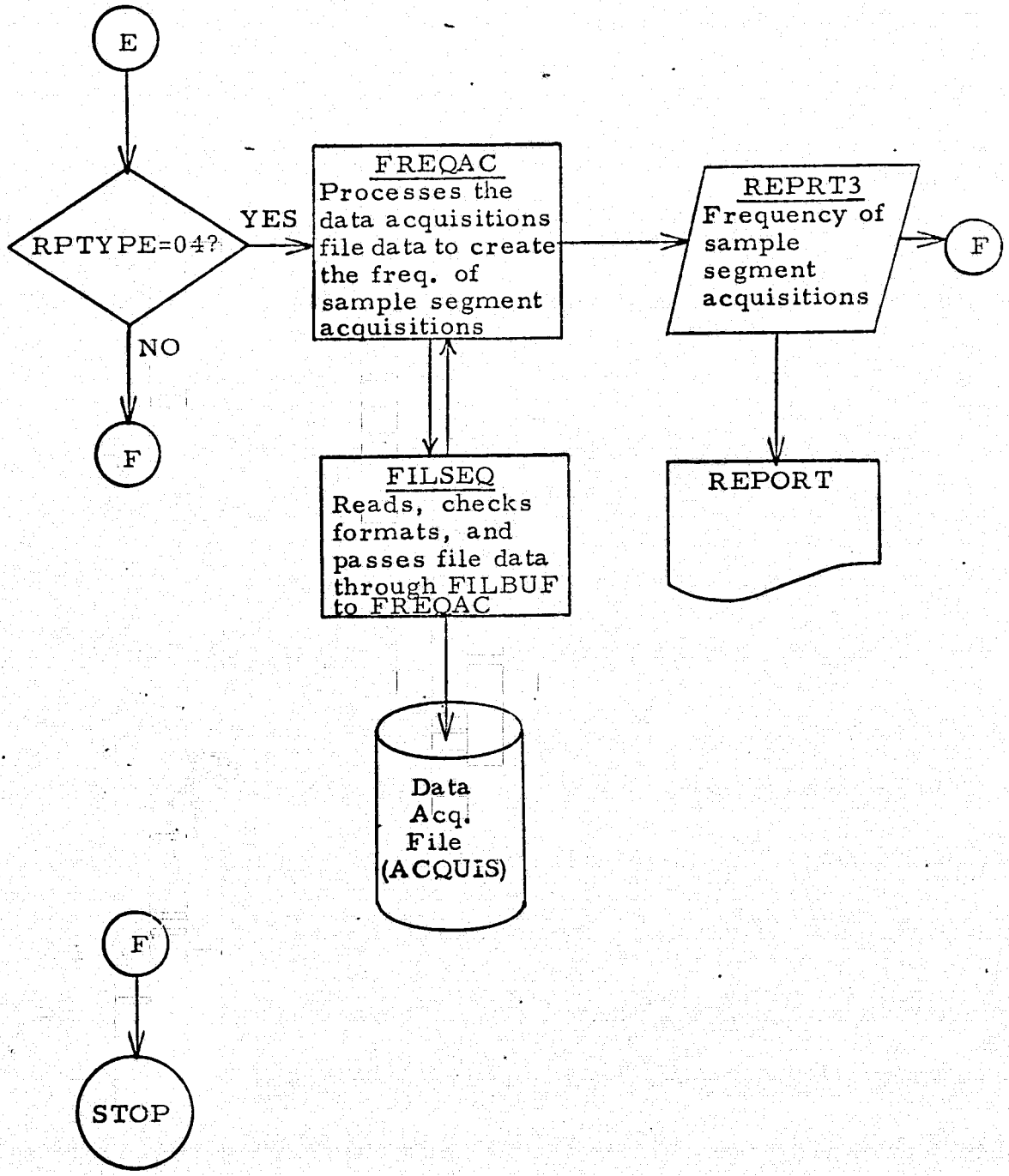
3.2 PROGRAM FLOW




POPDRV
A
RPTYPE=02?
YES
NO
D
PARMTR(1) ≠0?
YES
NO
SAMERR
Process SEGTRU & SUBHST data for report
REPRT2
Population sampling error report
FILSEQ
Passes data from seq. files to create report
REPORT
Segment Truth File (SEGTRU)
Substrata Historical File (SUBHST)
PARMTR(2) ≠0?
YES
NO
CAMSER
Process CAMSF & SUBHST data for report
REPORT2
Population CAMS error report
B
FILSEQ
REPORT
CAMS Output File (CAMSF)
Substrata Historical File (SUBHST)
B


B
PARMTR(3) ≠0?
YES
NO
YLDERR
Process YESOUT & SUBHST data for report
REPRT2
Population yield error report
FILSEQ
REPORT
YES Output File (YESOUT)
Substrata Historical File (SUBHST)
PARMTR(4) ≠0?
YES
NO
ANPERR
Process CASF data for report
REPRT2
Population area error report
C
FILDA
Passes data from direct access files to create report
REPORT
CAS Cum Output File (CASF)
C


C
PARMTR(5) ≠0?
YES
NO
ANPERR
Process CASF data for report
REPRT2
Population production error report
FILDA
REPORT
CASF
RPTYPE=03?
YES
NO
MCHIST
Process the CAS distribution output data to create Monte Carlo histograms
REPRT2
Monte Carlo area error prod. error yield error & confid. level
F
FILDA
Passes data from direct access file to create report
REPORTS
CAS Distrib. Output File (CASDIS)
E


E
RPTYPE=04?
YES
NO
F
FREQAC
Processes the data acquisitions file data to create the freq. of sample segment acquisitions
REPRT3
Frequency of sample segment acquisitions
F
FILSEQ
Reads, checks formats, and passes file data through FILBUF to FREQAC
REPORT
Data Acq. File (ACQUIS)
F
STOP

3.3 PROCEDURES AND EQUATIONS

3.3.1 Initialization

The general initialization tasks are performed by the main program POUT.

1. Initialize flags and constants; set nominals.
2. Read data cards.
3. Process data cards for input errors.
4. Determine report type to be produced.

3.3.2 Substrata Reference Data Report

The Substrata Reference Data will be created and produced by subroutine SUBREF from the Substrata Historical File.

1. The Substrata Historical File will be accessed sequentially through the file interface subroutine FILSEQ with RPTYPE = 01 and ICASIN(1) = valid case number.

   FILSEQ is a subroutine that reads a requested sequential file, checks for a match on CASE ID and returns one detail record at a time to the requesting subroutine through the FILBUF common block.

2. Process and sum the data by strata, zone, region and country for the following equations:

   (1) $XR(K) = \frac{AREA(K)}{XNA * AREAPS}$ where K is any substratum

   (2) $HWA = \sum_{I=K}^{J} XN(I) * XR(I) * 10289.712 * \frac{XHPW(I)}{100}$

   (3) $TWA = \sum_{I=K}^{J} XN(I) * XR(I) * 10289.712 * \frac{XTPW(I)}{100}$

   (4) $XINBS = \sum_{I=K}^{J} \frac{XBTPW(I) * \frac{XHPW(I)}{100} * XN(I) * XR(I) * 10289.712}{HWA}$

(5) $$XSIMBS = \frac{TWA - HWA}{HWA}$$

(6) $$XINCV = \left[\sum_{I=K}^{J} (XN(I)*XR(I)*10289.712*\frac{XTPW(I)}{100}*XCV2(I))^2\right]^{\frac{1}{2}} * \frac{1}{TWA}$$

(7) $$XSIMCV = \left[\frac{J}{J-1}*\left[\sum_{I=K}^{J} (TWA(I)-HWA(I))^2 - \frac{1}{J}*\left[\sum_{I=K}^{J} (TWA(I)-HWA(I))\right]^2\right]\right]^{\frac{1}{2}} * \frac{1}{TWA}$$

3. Output of the printed report for each strata, zone, region and country as processed is produced by calls to subroutine REPRT1.

3.3.3 Population Mean, Standard Deviation and Histogram Reports

Population reports will be selected (by input option), created and produced by control subroutine POPDRV.

1. For PARMTR(1)≠0 the Segment Truth File along with the Substrata Historical File (SUBHST) will be accessed sequentially through subroutine FILSEQ with RPTYPE = 02, PARMTR(1)≠0 and ICASIN(1) = valid case number to produce the Population Sampling Error Report.

2. For PARMTR(2)≠0 the CAMS Output File along with the SUBHST File will be accessed sequentially through subroutine FILSEQ with RPTYPE = 02, PARMTR(2)≠0 and ICASIN(2) = valid case number to produce the Population CAMS Error Report.

3. For PARMTR(3)≠0 the YES Output File along with the SUBHST File will be accessed sequentially through subroutine FILSEQ with RPTYPE = 02, PARMTR(3)≠0 and ICASIN(3) = valid case number to produce the Population Yield Error Report.

4. For PARMTR(4)≠0 or PARMTR(5)≠0 the CAS Cum Data File will be accessed through the direct access subroutine FILDA. The required input data for either of these accesses

is PARMTR(4)≠0, RPTYPE = 02 and ICASIN(4) = valid case number or PARMTR(5)≠0, RPTYPE = 02, and ICASIN(5) = valid case number.

5. Each of the five parameter reports computes and sums the same equations except for the computation of the specific error and reference values which appear below the basic equations.

(8) $$XMVAL = \frac{100}{J*REF(J)} * \sum_{I=K}^{J} DELERR(I)$$

(9) $$STDDEV = \frac{100}{REF(J)} * \left[\frac{1}{J-1} \left\{ \sum_{I=K}^{J} (DELERR(I))^2 - \frac{1}{J} * \left(\sum_{I=K}^{J} DELERR(I) \right)^2 \right\} \right]^{\frac{1}{2}}$$

(10) $$XMPSUM = J * XMVAL$$

(11) $$SDPSUM = J^{\frac{1}{2}} * STDDEV$$

(12) $$PCDERR = \frac{100}{REF(J)} * DELERR$$

where each parameter type (below) for any K substratum or S stratum

a. Population Sampling Error

(13) $$DELERR(K) = \left(\frac{1}{XM(K)} \sum_{L=I}^{XM(K)} XTPW(L) - XTPW(K) \right) * XN(K) * XR(K) * 10289.712$$

(14) $$REF(J) = \sum_{I=K}^{J} XTPW(I) * XN(I) * XR(I) * 10289.712$$

b. Population CAMS Error

(15) $$DELERR(K) = \left[\frac{1}{XM(K)} \sum_{L=I}^{XM(K)} (XEPW(L) - XTPW(L)) \right] * XN(K) * XR(K) * 10289.712$$

(16) $$REF(J) = \sum_{I=K}^{J} XTPW(I) * XN(I) * XR(I) * 10289.712$$

c. Population Yield Error

$$(17)\quad DELERR(S) = (XEYLD(S) - XTYLD(S) * \sum_{I=K}^{S} XTPW(I) * XN(I) * XR(I) * 10289.712$$

$$(18)\quad REF(J) = \sum_{S}^{J} * \left[\sum_{I=K}^{S} XTPW(I) * XN(I) * XR(I) * 10289.712 * XTYLD(S) \right]$$

d. Population Area Error

$$(19)\quad DELERR(S) = \frac{AERRS}{XNT}$$

$$(20)\quad REF(J) = \sum_{I=S}^{J} \frac{TWAS(I)}{XNT}$$

e. Population Production Error

$$(21)\quad DELERR(S) = \frac{PRERRS}{XNT}$$

$$(22)\quad REF(J) = \sum_{I=S}^{J} \frac{TPRODS(I)}{XNT}$$

6. Output of the printed reports produced by subroutine REPRT2.

3.3.4 Histograms of Monte Carlo Statistics

Monte Carlo Histograms will be selected (by input option), created and produced by subroutine MCHIST.

1. The CAS Distribution Output File will be accessed through the direct access subroutine FILDA with RPTYPE = 03, PARMTR and ICASIN values for each parameter of this type required.

2. Each of the nine reports computes and sums the equations (8), (9) and (12) of Section 3.3.3 with J = XNT. Each parameter type is shown below where L = zone (Z), region (R) or country (C) level.

a. Monte Carlo Area Error

(23) DELERR(L) = AERRL

(24) $REF = \frac{AEREFL}{XNT}$

b. Monte Carlo Production Error

(25) DELERR(L) = PRERRL

(26) $REF = \frac{PEREFL}{XNT}$

c. Monte Carlo Yield Error

(27) DELERR(L) = YERRL

(28) REF = YEREFL = 100

d. Confidence Level (Country Level Only)

(29) DELERR(C) = XCL

where XCL = CLEWA,
CLATEC,
CLATWC,
CLEPRD,
CLPTEC,
or CLPTWC

(30) REF = CLALL = 100

3. Output of the printed reports is produced by subroutine REPRT2.

3.3.5 Frequency of Sample Segment Acquisitions

The Frequency of Sample Segment Acquisitions Report will be created and produced by subroutine FREQAC.

1. The Data Acquisition File will be accessed sequentially through subroutine FILSEQ with RPTYPE = 04 and ICASIN(1) = valid case number.

2. The percentage of the number of segments will be tallied by zone, region and country for the bio-window combinations 0 1 2 3 4 1&2 1&3 1&4 2&3 2&4 3&4 1, 2&3 1, 2&4 1, 3&4 2, 3&4 1, 2, 3&4

3. Output of the printed report is produced by subroutine REPRT3.

3.3.6 Method of Parameter Range and Histogram Computations for 3.3.3 and 3.3.4 Above

If a set of range values is not input, then the nominal values will be used for all parameters; see Table 2-2 for these values.

As each parameter value is read from a file, the value is tallied according to the range interval it fits into. When all values have been read, the tally entries for each range are normalized by the total number of values read. The Histogram table is generated from these normalized values in the following manner.

$f_1, f_2, \cdots, f_{51}$ are number values in range 1, 2, $\cdots$, 51

Maximum of 51 range intervals are allowed.

Value	Sum of Values
f_1	f_1
f_2	$f_1 + f_2$
$\vdots$	$\vdots$
f_{51}	$f_1 + f_2 + \cdots + f_{51}$

POUT Symbol Table

Name	Symbol	Description	Source	Used	Range	Units
J	j	All substrata in a strata, zone, region or country		General	1-3200	
K		K^{th} substratum		General	1-3200	
AREA		Substrata land area	SUBHST	1		Hectares
XNA	N_A	Number of allocated segments in a substrata	SUBHST	1		
AREAPS		Area per segment (built-in value = 10289.712)				Hectares
XR	R	Ratio of the true substratum land area to the gross-pseudo substratum (before exclusion) area	1	2, 3, 4, 6, 13 14, 15, 16, 17, 18		
XN	N	Number of agricultural segments in substrata	SUBHST	2, 3, 4, 6, 13, 14, 15, 16, 17, 18		
XHPW	$\tilde{PW}$	Historical proportion of wheat	SUBHST	2, 4		
HWA		Historical wheat area (sum)	2	4, 5		Hectares
HWA(K)		Historical wheat area, K^{th} substratum	2	7		Hectares
XTPW	PW	True proportion of wheat	SUBHST	3, 6, 13, 14, 15, 16, 17, 18		
TWA		True wheat area (sum)	3	5, 6, 7		Hectares
TWA(K)		True wheat area, K^{th} substratum	3	7		Hectares
XBTPW	δPW	Bias of true proportion of wheat	SUBHST	4		
XINBS		Input bias	4	REPRT1		% Historical
XSIMBS		Simulation bias	5	REPRT1		% Historical
XCV2	CV_2	Coefficient of variation for within county variation of PW	SUBHST	6		

POUT Symbol Table

Name	Symbol	Description	Source	Used	Range	Units
XINCV		Input coefficient of variation (CV)	6	REPRT1		% True
XSIMCV		Simulation coefficient of variation (CV)	7	REPRT1		% True
DELERR	ΔE	Reading	13, 15, 17, 19, 21, 23, 25, 27, 29	8, 9, 12		
REF		Reference value	14, 16, 18, 20, 22, 24, 26, 28, 30	8, 9, 12		
XMVAL		Mean value	8	10		
STDDEV		Standard deviation	9	11		
XMPSUM		Mean of population sum	10	REPRT2		
SDPSUM		Standard deviation of population sum	11	REPRT2		
PCDERR	% ΔE	Percent reading	12	REPRT2		
XM 24	M	Number of sample segments in the K^{th} sub-stratum	SUBHST	13		
XEPW	$\hat{P}$	Estimated proportion of wheat	CAMSF	15		
I		I^{th} sample segment		General		
XEYLD	$\hat{Y}$	Estimated yield	YESOUT	17	0-99.99	Quintals/ Hectares
XTYLD	Y	True yield	YESOUT	17, 18		Quintals/ Hectares
S		S^{th} stratum		General		
XNT	N_T	Number of Monte Carlo iterations	CASF CASDIS	General		
AERRS	EA_S	Area error	CASF	19		

3.4 continued)

POUT Symbol Table

Name	Symbol	Description	Source	Used	Range	Units
TWAS	WA_S	True wheat area for the S^{th} stratum	CASF	20		Hectares
PRERRS	EP_S	Production error	CASF	21		
TPRODS	PRD_S	True production	CASF	22		Quntials
AERRL		Area error	CASDIS	23		
AEREFL		Reference value of area error	CASDIS	24		Hectares
PRERRL		Production error	CASDIS	25		
PEREFL		Reference value of production error	CASDIS	26		
YERRL		Yield error	CASDIS	27		
YEREFL		Reference value of yield error (= 100)		28		
CLEWA		Area confidence level Est/Est	CASDIS	29		
CLATEC		Area confidence level True/Est	CASDIS	29		
CLATWC		Area confidence level True/WC	CASDIS	29		
CLEPRD		Production confidence level Est/Est	CASDIS	29		
CLPTEC		Production confidence level True/Est	CASDIS	29		
CLPTWC		Production confidence level True/WC	CASDIS	29		
CLALL		Reference value for all confidence levels area and production (= 100)		30		

4.0 OUTPUT

The major product of this program is its printed reports.

4.1 PRINTED DATA

4.1.1 Reports

The type of report is determined by option on the input cards. See Tables 4.1-4.3 for format and report contents.

4.1.2 Echo Print Input Card Images

The control card inputs to POUT are printed out in the following format after all nominal values have been stored.

```
                    HEADER
AAA————————————————————————AAA     LPP SIMULATION PAGE 1

   RPTYPE  PARMTR   ICASIN                                  LEVEL  ICSESH
     NN    N N N N N  NNNN  NNNN  NNNN  NNNN  NNNN          N     NNNN

INTERVAL  START    INTVL1   BREAK1   INTVL2   BREAK2   INTVL3    STOP
          ±XXX.X   XXX.X    ±XXX.X   XXX.X    ±XXX.X   XXX.X     ±XXX.X

          BIOWD  WPRTY
          NNNN   NNNN

          IPRD
          YY1 MM1 DD1  YY2 MM2 DD2 ————————————————————————>
          YYn MMn DDn ————————————————————————————————————>
```

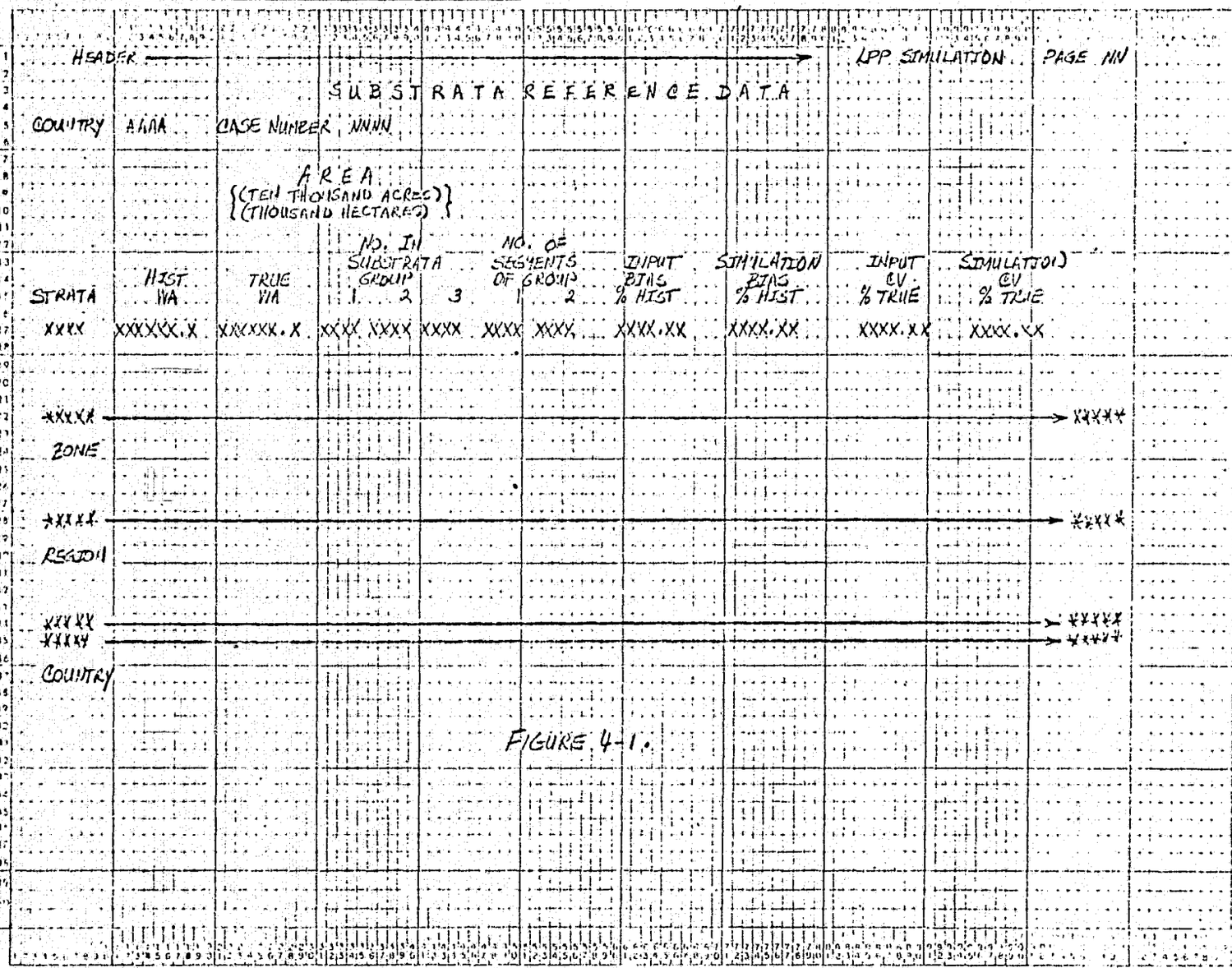


FIGURE 4-1.

PROG ID. _____ PAGE _____

DATE _____

PROGRAMMER OR DOCUMENTALIST _____

PROGRAM TITLE _____

STANDARD FORMAT FOR HISTOGRAM OUTPUT

Fold back at dotted line

```
HEADER ------------------------------------------------------>  LPP SIMULATION  PAGE NN
(Parameter Title for each type of Report - Population or Monte Carlo)

CASE NUMBER NNNN

FILE NAME  AAAA/AAA

(PARAMETER) XXX.XX   1 XXX.XXX TO XXX.XXX  18 XXX.XXX TO XXX.XXX  35 XXX.XX TO XXX.XXX
                     2                     19                     36
                     3                     20                     37

                    17                     34                     51

DATA TYPE AAAAAAAAAA  MEAN XXX.XXX  STD. DEVIATION XXX.XXX  REF. VALUE XXX.XXX

*MEAN OF POPULATION SUM XXX.XXX  *STD. DEVIATION OF POPULATION SUM XXX.XXX

MINIMUM READING XXX.XXX  MAXIMUM READING XXX.XXX  NUMBER OF READINGS XXXX

COUNTY AAAA  REGION AAAA  ZONE AAAA  PREDICTION DATE XX/XX/XX

HISTOGRAM DATA  1. XXX.XXX  XXX.XXX  18. XXX.XXX  XXX.XXX  35. XXX.XXX  XXX.XXX
                2.                    19.                    36.
                3.                    20.                    37.

                17.                   34.                    51.

DATA TYPE AAAAAAAAAA  MEAN XXX.XXX  STD. DEVIATION XXX.XXX  REF. VALUE XXX.XXX

*MEAN OF POPULATION SUM XXX.XXX  *STD. DEVIATION OF POPULATION SUM XXX.XXX

MINIMUM READING XXX.XXX  MAXIMUM READING XXX.XXX  NUMBER OF READINGS XXXX

COUNTY AAAA  REGION AAAA  ZONE AAAA  PREDICTION DATE XX/XX/XX

* Not valid for Monte Carlo Statistics Reports
```

FIGURE 4-2.

FIGURE 4-2

Fold back at dotted line

PRINT CHART PROG ID. _____ PAGE _____

DATE _____

TITLE _____

USER OR DOCUMENTALIST

FREQUENCY OF SAMPLE SEGMENT ACQUISITIONS

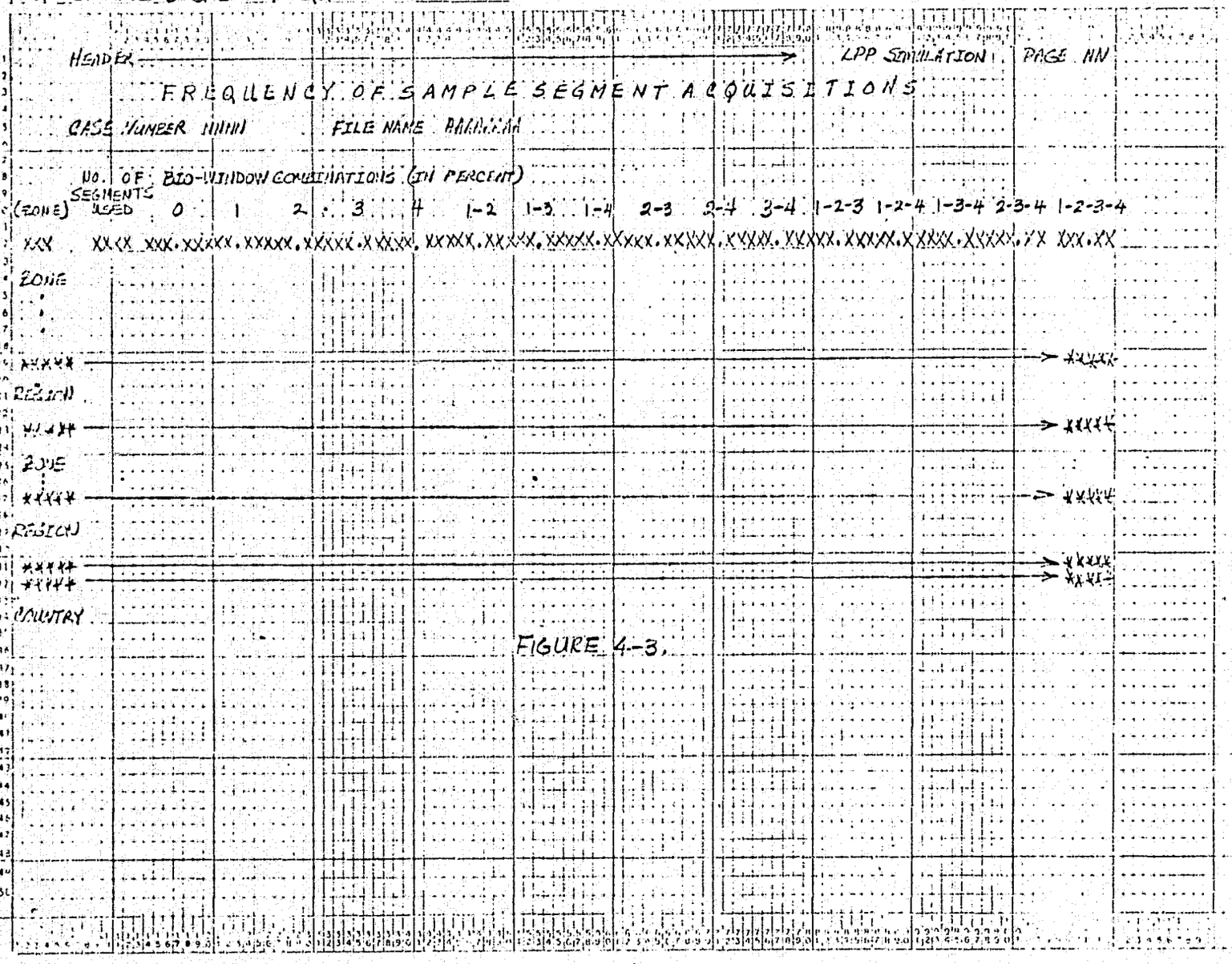


FIGURE 4-3.

5.0 ERROR PROCESSING

5.1 INPUT ERRORS

All input errors in this program will be fatal.

1. Label and sequence checking on control cards. Message

   *** IMPROPER LABEL AND SEQUENCE NUMBER ON
   POUT CONTROL CARD NO. ____. LABEL AND SEQ. NO. = ____.

2. Range testing on RPTYPE value. Message

   *** RPTYPE VALUE OUT OF RANGE. RPTYPE = _____.

3. Range testing on Histogram interval values. Messages

   *** START VALUE GREATER THAN BREAK1 VALUE.
   START = _____ BREAK1 = _____.

   *** BREAK1 VALUE GREATER THAN BREAK2 VALUE.
   BREAK1 = _____ BREAK2 = _____.

   *** BREAK2 VALUE GREATER THAN STOP VALUE.
   BREAK2 = _____ STOP = _____.

   *** THE SEGMENT BETWEEN START AND BREAK1 WILL
   NOT DIVIDE INTO EQUAL INTVL1 INTERVALS.
   BREAK1 - START = _____. INTVL1 = _____.

   *** THE SEGMENT BETWEEN BREAK1 AND BREAK2 WILL
   NOT DIVIDE INTO EQUAL INTVL2 INTERVALS.
   BREAK2 - BREAK1 = _____. INTVL2 = _____.

   *** THE SEGMENT BETWEEN BREAK2 AND STOP WILL
   NOT DIVIDE INTO EQUAL INTVL3 INTERVALS.
   STOP - BREAK2 = _____. INTVL3 = _____.

   *** REQUESTED NO. OF HISTOGRAM INTERVALS EXCEEDS
   MAXIMUM OF 51.
   REQUESTED NO. OF INTERVALS THAT WOULD BE
   GENERATED = _____.

4. Range testing on PARMTR values. Message

   *** PARMTR(i) VALUE OUT OF RANGE. PARMTR(i) = _____.

5.2 PROCESSING ERRORS

1. *** INPUT PREDICTION DATE DOES NOT MATCH DATE ON FILE.
   RPTYPE = ____. PARMTR(i) = ____. INPUT DATE = ____.

   Message is non-fatal. The input date is skipped.

2. Case number on control card does not match case number on file. Message

   *** ICASIN(i) DOES NOT MATCH CASE NUMBER ON FILE.
   ICASIN(i) = ____ FILE CASE NO. = ____ FILE NAME = ____.

PART II

COMMON BLOCK DEFINITIONS

COMMON STORAGE ALLOCATION

Name BWCMBN Size 17 Page 1 of 1

Function Holds the bio-window combinations data for printing

Name	Dimension	Format	Description	Symbol	Units
NSEGS		I	Number of segments used in the bio-window combinations		
BIOCMB	16		Bio-window combinations		percent

COMMON STORAGE ALLOCATION

Name CARDIN Size 15 Page 1 of 1

Function Holds the second and third input data cards except for histogram range input

Name	Dimension	Format	Description	Symbol	Units
RPTYPE		I	Report type selection flag		
			01 - Substrata Reference Data 02 - Population Histogram		
			03 - Monte Carlo Histogram 04 - Freq. of Sample Seg. Acquisition		
AUNITS		I	External print units flag 0 - English, 1 - Metric		
PARMTR	5	I	Parameter type option flags for RPTYPE = 02 or 03		
ICASIN	5	I	Case number identification associated with the required input file		
LEVEL		I	Parameter report level indicators 1 = zone, 2 = region, 3 = country		
JPARM		I	Which PARMTR (index) being processed		
IERR		I	Error flag for multiple report types (no match on case number, etc.)		

COMMON STORAGE ALLOCATION

Name CONVRT Size 6 Page 1 of 1

Function Contains the print units conversion factors

Name	Dimension	Format	Description	Symbol	Units
CTAMER	3	R	Conversion factors to convert internal metric units to external English units. CTAMER(1) = Wheat area = .0002471044 CTAMER(2) = Production = 3.6743544 CTAMER(3) = Yield = 1.4869664		
CTMTRC	3	R	Conversion factors to convert internal metric units to external metric units CTMTRC(1) = Wheat area = .001 CTMTRC(2) = Production = .1 CTMTRC(3) = Yield = 1		

COMMON STORAGE ALLOCATION

Name FACQUS Size 107 Page 1 of 1

Function Holds 1 detail record from Data Acquisition File, input file (ACQUIS)

Name	Dimension	Format	Description	Symbol	Units
CIYB6		R	Country ID		
REGN7		I	Region ID		
ZONE7		I	Zone ID		
STRTA7		I	Strata ID		
SBSTA7		I	Substrata ID		
SEGMT7		I	Segment ID		
IWINDO	25,4	I	2nd dimension = 4 crop windows 1st dimension = up to 25 zulu dates/window in ascending order		
ITOTL		I	Total number of accesses		

COMMON STORAGE ALLOCATION

Name FBLKBX Size (28x18) Page 1 of 1

Function Holds 1 header/detail record from CAS cum. output file - input file (CASF)

Name	Dimension	Format	Description	Symbol	Units
FBLKBX	28x18	R	Header record, or detail record sectioned into 18 groups		

COMMON STORAGE ALLOCATION

Name FCAMSF Size 19 Page 1 of 1

Function Holds 1 detail record from CAMS Output File - Input file (CAMSF)

Name	Dimension	Format	Description	Symbol	Units
COUN3		A4	Country ID		
REGN3		I	Region ID		
ZONE3		I	Zone ID		
STRTA3		I	Strata ID		
SBSTA3		I	Substrata ID		
SEGMT3		I	Segment ID		
TPW3		R	True proportion of wheat for segment		
EWINDO	3, 4	R	Bio-window values where zulu date of acquisition (integer). Est. proportion of wheat. Error in proportion of wheat estimate — for each of the 4 windows		

FCAMSF

EWINDO ≡ IWIN

ZULU(I) ≡ IWIN(1,I)
EPW(I) ≡ EWINDO(2,I)

I = 1 to 4

COMMON STORAGE ALLOCATION

Name FCASCM Size 28 Page 1 of 2

Function Holds 1 header/detail group of a record from common block FBLKBX

Name	Dimension	Format	Description	Symbol	Units
NAME5		A6	File name 'CASCUM'		
ICASE5		I	Case number		
COUN5		A6	Country ID		
NT5		I	Current Monte Carlo iteration number		
NREGS5		I	Number of regions		
NZTOT5		I	Total number of zones		
NSTRT5		I	Total number of strata		
NBW5		I	Number of bio-windows		
NPD5		I	Number of prediction dates		
BWNBR5	4	I	Bio-window numbers		
PDNBR5	14	I	Prediction dates in zulu		
filler					

COMMON STORAGE ALLOCATION

Name FCASCM Size 28 Page 2 of 2

Function Holds 1 header/detail group (data set 14) from CAS cum. Output File - input file (CASF)

Name	Dimension	Format	Description	Symbol	Units
REGN5		I	Region ID		
ZONE5		I	Zone ID		
STRTA5		I	Strata ID		
filler					
HWA5		R	Historical wheat area (WA)	$\tilde{WA}$	
TWA5		R	True WA	WA	
EWA5		R	Estimated $\tilde{WA}$	$\tilde{WA}$	
AERR5		R	Error in WA		
AVAR5		R	Variance in WA		
TPROD5		R	True production		
EPROD5		R	Estimated production		
PRERR5		R	Production error		
			(Remainder of record is not used)		

COMMON STORAGE ALLOCATION

Name FCASDS Size 303 Page 1 of 3

Function Holds 1 header/detail record from CAS Dist. Output File - input file (CASDIS)

Name	Dimension	Format	Description	Symbol	Units
NAME6		A6	File name 'CASDIS'		
ICASE6		I	Case number		
COUN6		A6	Country ID		
NT6		I	Current Monte Carlo iteration number		
NREGS6		I	Number of regions		
NZTOT6		I	Total number of zones		
filler					
NBW6		I	Number of bio-windows		
NPD6		I	Number of prediction dates		
BWNBR6	4	I	Bio-window numbers		
PDNBR6	14	I	Prediction dates inzulu		
filler			(276 words of filler)		
⋮					

COMMON STORAGE ALLOCATION

Name FCASDS Size 303 Page 2 of 3

Function Holds 2 header/detail record (country records) from CAS Dist. Output File - input file (CASDIS)

Name	Dimension	Format	Description	Symbol	Units
			1st Country Record		
AEREF		R	Reference value for area error		
PEREF		R	Reference value for production error		
YEREFC		R	Reference value for yield error		
AERR	100	R	Word N+3 specifies the area error for the nth Monte Carlo iteration		
PRERR	100	R	Word N + 103 specifies the production error for the nth Monte Carlo iteration		
YERR	100	R	Word N + 203 specifies the yield error for the nth Monte Carlo iterations		
			2nd Country Record		
CLEARF		R	Ref. value of area confidence level Est/Est		
CLEPRF		R	Ref. value of prod. confidence level Est/Est		
CLTARF		R	Ref. value of area confidence level True/Est.		
CLEW6	100	R	Area confidence level Est/Est		
CLEPR6	100	R	Production confidence level Est/Est		
CLATC6	100	R	Area confidence level True/Est		

COMMON STORAGE ALLOCATION

Name FCASDS Size 303 Page 3 of 3

Function Holds 2 header/detail record (country and region or zone) from CAS Dist. Output File - input file (CASDIS)

Name	Dimension	Format	Description	Symbol	Units
			3rd Country Record		
CLTPRF		R	Ref. value of prod. confidence level True/Est		
CLAWCR		R	Ref. value of area confidence level True/WC		
CLPWCR		R	Ref. value of prod. confidence level True/WC		
CLPTC6	100	R	Prod. confidence level True/Est		
CLATW6	100	R	Area confidence level True/WC		
CLPTW6	100	R	Prod. confidence level True/WC		
			Region or Zone Record		
AEREF		R	Ref. value for area error		
PEREF		R	Ref. value for prod. error		
YEREF		R	Ref. value for yield error		
AERR	100	R	Area error, N-Monte Carlo iterations		
PRERR	100	R	Prod. error, N-Monte Carlo iterations		
YERR	100	R	Yield error, N-Monte Carlo iterations		

COMMON STORAGE ALLOCATION

Name FHEADR Size 4 Page 1 of 1

Function Holds the first 4 words of each sequential file

Header Record

Name	Dimension	Format	Description	Symbol	Units
NAME	2	(I) A4	File name 8 characters; 4 char/word		
ICASEF		I	File case number		
IMXSEG		I	Maximum number of segments in any substrata (SUBHST)		

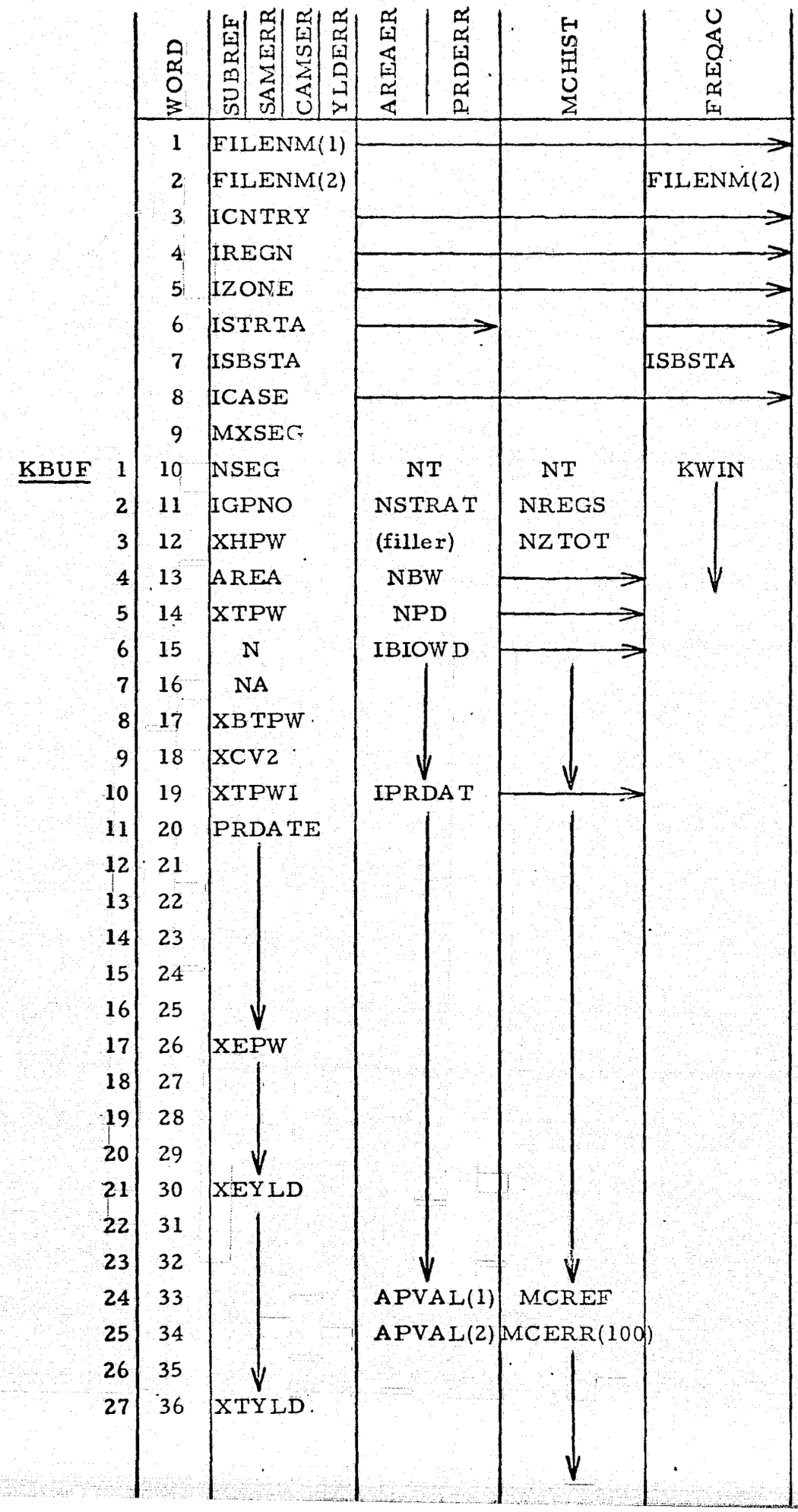
COMMON STORAGE ALLOCATION

Name FILBUF Size 134 Page 1 of 1

Function Holds file data returned by file access routines to the calling routine

Name	Dimension	Format	Description	Symbol	Units
FILENM	2	A6	File name		
ICNTRY		A6	Country ID		
IREGN		I	Region ID		
IZONE		I	Zone ID		
ISTRTA		I	Strata ID		
ISBSTA		I	Substrata ID		
ICASE		I	Case number		
MXSEG		I	Maximum number of segments in any substrata (SUBHST only)		
KBUF	125		Bucket area that the detail info is equivalenced into		

FILBUF
EQUIVALENCES



COMMON STORAGE ALLOCATION

Name FSBHST Size 168 Page 1 of 1

Function Holds one detail record from Substrata Historical File, input file (SUBHST)

Name	Dimension	Format	Description	Symbol	Units
COUN1		A4	Country ID		
REGN1		I	Region ID		
ZONE1		I	Zone ID		
STRTA1		I	Strata ID		
SBSTA1		I	Substrata ID		
INEG		I	List of sample segments		
NGRP		I	Group no.		
HPW		R	Hist. proportion of wheat		
AREA1		R	Land area		kilometers^2
TPW1		R	True proportion of wheat for substrata		
NAGR		I	No. of agricultural segments in substrata		
NA1		I	No. of allocated segments in substrata		
BTPW		R	Bias of true proportion of wheat		
RTMP		R	Ratio of true mixed pixels		
CV1		R			
CV2		R	Coef. of variation for within county variation of PW		
CV3		R			
CV4		R			

COMMON STORAGE ALLOCATION

Name FSGTRU Size 16 Page 1 of 1

Function Holds one detail record from Segment Truth File, input file (SEGTRU)

Name	Dimension	Format	Description	Symbol	Units
COUN2		A4	Country ID		
REGN2		I	Region ID		
ZONE2		I	Zone ID		
STRTA2		I	Strata ID		
SBSTA2		I	Substrata ID		
SEGMT2		I	Segment ID		
ITS			DUME2 Dim 8		
ITSPRI	6		DUME2 Dim 8		
SWFLG		I	DUME2 Dim 8		
TPW2		R	True proportion of wheat for segment		
TPM2		R			

COMMON STORAGE ALLOCATION

Name FYESOT Size 23 Page 1 of 1

Function Holds one detail record from YES Output File - input file (YESOUT)

Name	Dimension	Format	Description	Symbol	Units
COUN4		A4	Country ID		
REGN4		I	Region ID		
ZONE4		I	Zone ID		
STRTA4		I	Stratum ID		
YSTR		R	True yield for stratum		Quintals/ Hectare
IZPRDD	6	I	Zulu prediction date		
YSCI	6	R	Est. yield for prediction date		Quintals/ Hectare
YSYCI	6	R	Std. deviation of yield error		Quintals/ Hectare
			YSPNT ≡ IYSPNT IZPRDD(I) = IYSPNT(1,I) I = 1 to 4		
			YSCI(I) = YSPNT(2,I)		

COMMON STORAGE ALLOCATION

Name HSTOGM Size 114 Page 1 of 1

Function Holds the Histogram data values for printing

Name	Dimension	Format	Description	Symbol	Units
XMVAL		R	Mean value of the error		percent
STDDEV		R	Standard deviation of the error	σ	
XMPSUM		R	Mean of population sum for RPTYPE = 02 only		
SDPSUM		R	Std. deviation of population sum for RPTYPE = 02 only		
NRANGE		I	Number of range intervals		
IRANGE	51	I	Number of readings per each range interval		
NTOTL		I	Total number of readings		
RANGES	52	R	Range interval points		percent
MINVAL		R	Minimum reading value		percent
MAXVAL		R	Maximum reading value		percent
DATPRD	3	I	Prediction date if RPTYPE = 02 or 03 and if 02 PARMTR(n), n = 2, 3, 4 & 5		
NZRNG		I	Index of the zero range		
REF			Reference value		

COMMON STORAGE ALLOCATION

Name IXRCD Size 1136 Page 1 of 1

Function An array containing the index record for the direct access routine RANACF

Name	Dimension	Format	Description	Symbol	Units
IXRCD	1136		Subroutine RANACF index record array		

COMMON STORAGE ALLOCATION

Name PRNTID Size 15 Page 1 of 1

Function Holds identification information for report printing

Name	Dimension	Format	Description	Symbol	Units
CNTRY		A4	Country ID		
REGION		I	Region ID		
ZONE		I	Zone ID		
STRATA		I	Strata ID		
ICASE		I	Case number		
FILNAM	2	A6	For RPTYPE = 02 or 03 file name		
ITOTYP		I	Type of information to be printed 1=strata, 2=zone, 3=region, 4=country		
NAMFMT		I	File name format flag (REPRT2) 0 = 2A4 format, 1 = A6 format		

COMMON STORAGE ALLOCATION

Name PRTCTL Size 4 Page 1 of 1

Function Print control flags

Name	Dimension	Format	Description	Symbol	Units
NPAGE		I	Current page		
NLINES		I	Line counter		
MAXLIN		I	Maximum lines per page		
KOUT		I	Print output unit number		
HEADER	16	I	Run title output (print) area		

COMMON STORAGE ALLOCATION

Name RANGE Size 7 Page 1 of 1

Function Holds the histogram range input values

Name	Dimension	Format	Description	Symbol	Units
START		R	Histogram start value		percent
INTVL1		R	Histogram 1st interval value		percent
BREAK1		R	Histogram breakpoint to change interval		percent
INTVL2		R	Histogram 2nd interval value		percent
BREAK2		R	Histogram breakpoint to change next interval		percent
INTVL3		R	Histogram 3rd interval value		percent
STOP		R	Histogram terminal value		

COMMON STORAGE ALLOCATION

Name READNG Size 5600 Page 1 of 1

Function Intermediate storage area for the "error readings" of the Population and Monte Carlo Reports. Intermediate Buffer for Confidence Levels.

Name	Dimension	Format	Description	Symbol	Units
READNG	5000	R	"Error readings" at substrata, strata, zone, region or country level		
CLEWA	100	R	Area confidence level Est/Est		
CLEPRD	100	R	Production confidence level Est/Est		
CLATEC	100	R	Area confidence level True/Est		
CLPTEC	100	R	Production confidence level True/Est		
CLATWC	100	R	Area confidence level True/WC		
CLPTWC	100	R	Production confidence level True/WC		

COMMON STORAGE ALLOCATION

Name SELCTN Size 50 Page 1 of 1

Function Holds the input bio-window and prediction date selection criteria for RPTYPE = 02 or 03

Name	Dimension	Format	Description	Symbol	Units
BIOWD	4	I	Bio-windows to be processed flags		
WPRTY	4	I	Bio-window order of priority		
IPRD	3, 14	I	Prediction dates to be processed		year, month, day

COMMON STORAGE ALLOCATION

Name SUBVAL Size 11 Page 1 of 1

Function Holds the Substrata Historical Data values for printing

Name	Dimension	Format	Description	Symbol	Units
HWA		R	Historical wheat area	$\tilde{WA}$	Hectares
TWA		R	True wheat area	WA	Hectares
NSUBGP	3	I	Number in each substrata group type		
NSEGGP	2	I	Number of segments in groups 1 and 2		
XINBS		R	Input bias		% Hist.
XSIMBS		R	Simulation bias		% Hist.
XINCV		R	Input coefficient of variation		% True
XSIMCV		R	Simulation coefficient of variation		% True

PART III

LIST OF SUBROUTINES AND SUBROUTINE CALL STRUCTURE

Name	Function
POUT	Main control subroutine for POUT program.
SUBREF	This subroutine processes the substrata reference data report - RPTYPE = 1.
POPDRV	This subroutine controls the generation of the population reports - RPTYPE = 2.
SAMERR	This subroutine produces the population segment sampling error report - RPTYPE = 2, PARMTR = 1.
CAMSER	This subroutine produces the population CAMS error report - RPTYPE = 2, PARMTR = 2.
FREQAC	This subroutine produces the frequency of sample segment acquisitions report - RPTYPE = 4.
YLDERR	This subroutine produces the population yield error report - RPTYPE = 2, PARMTR = 3.
REPRT1	This subroutine prints the substrata reference data report.
REPRT2	This subroutine prints the histogram data reports.
REPRT3	This subroutine prints the frequency of sample segments acquisitions.
PAGHDR	This routine controls count on printed output and provides a line of printed heading on top of each output page.
GENRNG	This subroutine generates the histogram ranges from the interval input.
FILSEG	This subroutine reads the required sequential files determined by the RPTYPE flag and returns the required data in common block FILBUF.
CLMOVE	This routine (used by MCHIST) moves the confidence levels, one at a time, from the reading buffer to FILBUF.
PUNITS	This routine sets up the proper print and conversion units for the population and Monte Carlo reports.
MCHIST	This subroutine produces the Monte Carlo reports - RPTYPE = 3.
ANPERR	This subroutine produces the population area error report or the population production error report - RPTYPE = 2, PARMTR = 4, 5.

Name	Function
FILDA	This subroutine controls the reading of the CASF file or CASDIS file.
RANACF	This routine is a generalized random access I/O routine (utilizing define file on the Univac).

POUT Subroutine Call Structure

```
POUT
    PAGHDR
    SUBREF
        FILSEQ
            PAGHDR
        REPRT1
            PAGHDR
    POPDRV
        GENRNG
        SAMERR
            FILSEQ
            REPRT2
                PUNITS
                PAGHDR
        CAMSER
            FILSEQ
            LFPA
            FZULU
            REPRT2
            PAGHDR
        YLDERR
            FILSEQ
            LFPA
            PAGHDR
            FZULU
            REPRT2
        ANPERR
            FILDA
                PAGHDR
                RANACF
            LFPA
            REPRT2
```

MCHIST
- FILDA
- PAGHDR
- GENRNG
- LFPA
- CLMOVE
- FZULU
- REPRT2

FREQAC
- FILSEQ
- REPRT3
  - PAGHDR

PART IV

SUBROUTINE DESCRIPTIONS AND FLOWCHARTS

Program Main Routine POUT

Purpose:

This is the driver for the Post Processor (POUT) program. The driver reads in the control card input and error check all values except the file case numbers (ICASIN(i)). All card input is checked before any error will cause processing to terminate. All card input is Echo printed and all errors are noted with an appropriate error message. The routine then calls the requested RPTYPE report subroutine.

Input:

/CARDIN/	IHEADR
	RPTYPE
	AUNITS
	PARMTR (5)
	ICASIN (5)
	LEVEL
/RANGE/	START
	INTVL1
	BREAK1
	INTVL2
	BREAK2
	INTVL3
	STOP
/SELCTN/	BIOWD (4)
	WPRTY (4)
	IPRD (3, 14)

Output:

/CARDIN/	
/RANGE/	
/SELCTN/	
/PRTCTL/	NPAGE
	NLINES
	MAXLIN
	KOUT
	HEADER (16)
/CONVRT/	CTAMER (3)
	CTMTRC (3)

Linkage:

Main program.

Subroutine Used:

CALL SUBREF	process and produce Substrata Reference Data Report if RPTYPE = 01.
CALL POPDRV	process and produce Population Reports if RPTYPE = 02.
CALL MCHIST	process and produce Monte Carlo Reports if RPTYPE = 03.
CALL FREQAC	process and produce Frequency of Sample Segments Acquisitions Report.
CALL PAGHDR	controls count on printed output and provides a line of printed heading on the top of each output page with page number.

Local Variables:

NOMRNG = 0 = 1	Histogram range values input Nominal ranges to be used
RNGNML =	Histogram nominal range values
AMCVT =	American conversion factors
XMCVT =	Metric conversion factors
IN = 5	Standard read (input) unit
IOUT = 6	Standard write (output) unit
LNMAX = 45	Maximum no. of print lines per page
LABEL = "POUT"	Card label identifier
NCARD = "01, 02, etc."	Card label numbers
MRPTYP = 4	Maximum RPTYPE value
XMXRGE = 51	Maximum no. of histogram ranges
INERRS	Logical flag indicating that an input error has occurred. At the completion of input this flag is tested for run termination due error(s).

Processing:

1. Zero out arrays and preset constants.

2. Read Header Card
   Save header
   Check card label and sequence no.
   Print error message and set INERRS = true

3. Read Data Card 1
   Check card label and sequence no.
   Check RPTYPE in range
   Check RANGE values for nominal usage
   Check input RANGE values for interval or size errors

4. Read Data Card 2
   Check card label and sequence
   Check PARMTR values and if set on ICASIN values
   Check LEVEL value if RPTYPE = 02; use nominal of 1 if = 0

5. Read Data Cards 4 and 5
   Check card labels and sequences
   Check only if RPTYPE = 02 or 03

6. Echo print all input values with all nominals set where necessary.

7. Terminate run if INERRS = true.

8. Run case according to RPTYPE value.

9. Stop.

Subroutine SUBREF

Purpose:

This routine produces the Substrata Reference Data Report. It accesses the Substrata Historical File (SUBHST) through the sequential file read routine FILSEQ. It computes the required quantities and passes them to routine REPRT1 for printing.

Input:

/FILBUF/	ICNTRY	
	IREGN	
	IZONE	
	ISTRTA	
	ICASE	
	NSEG	Number of sample segments in this substrata.
	IGPNO	Group number of the sample segments.
	XHPW	Historical proportion of wheat.
	AREA	Land area of the substrata.
	XTPW	True proportion of wheat of substrata.
	N	No. of agricultural segments in substrata.
	NA	No. of allocated segments in substrata.
	XBTPW	Bias of true proportion of wheat.
	XCV2	Coefficient of varia;tion from within county variation PW.

Output:

/PRNTID/	CNTRY
	REGION
	ZONE
	STRATA
	ICASE
	ITOTYP
/SUBVAL/	HWA
	TWA
	NSUBGP (3)
	NSEGGP (2)
	XINBS
	XSIMBS
	XINCV
	XSIMCV

Linkage:

CALL SUBREF called by POUT.

Subroutines Used:

CALL FILSEQ (IH)	reads data file SUBHST sequentially. IH = 1 read header record. IH = 0 read detail record. Return of IH < 0 end of file on file being read.
CALL REPRT1	prints the Substrata Reference Data Report.

Local Variable Description:

AREAPS	= 10289.712, area per segment in Hectares.
XNA	Flt. pt., no. of allocated segments.
XR	Ratio of the true substratum land area to the gross-pseudo substratum area.
XN	Flt. pt., no. of agricultural segments in substrata.
HWAK	Historical wheat area, substrata level.
HWAS	Historical wheat area, strata level.
HWAZ	Historical wheat area, zone level.
HWAR	Historical wheat area, region level.
HWAC	Historical wheat area, country level.
TWAK	True wheat area, substrata level.
TWAS	True wheat area, strata level.
TWAZ	True wheat area, zone level.
TWAR	True wheat area, region level.
TWAC	True wheat area, country level.
XNR	XN * XR * 10289.72, substrata level.
J	No. of substrata in a stratum.
XINBSS	Input bias, strata level.
XINBSZ	Input bias, zone level.
XINBSR	Input bias, region level.
XINBSC	Input bias, country level.
XSMBSZ	Simulation bias, zone level.
XSMBSR	Simulation bias, region level.
XSMBSC	Simulation bias, country level.
XINCVS	Input coefficient of variation, strata level.
XINCVZ	Input coefficient of varia;ion, zone level.
XINCVR	Input coefficient of variation, region level.
XINCVC	Input coefficient of variation, country level.
XSMCVS(2)	Simulation coefficient of variation, strata level.
XSMCVZ	Simulation coefficient of variation, zone level.
XSMCVR	Simulation coefficient of variation, region level.
XSMCVC	Simulation coefficient of variation, country level.
IH	Flag to subroutine FILSEQ where IH = 1 read header record or IH = 0 read detail record.
KSBSTA	Substrata counter.

Processing:

1. IH = 1, call FILSEQ to read header record.

2. Verification done by FILSEQ.

3. SAVE ICASE in common PRNTID.

4. SET KSBSTA = 0 and J = 0 and zero out accumulative variables. Call FILSEQ to read detail record.

   KSBSTA = KSBSTA + 1
   IF (KSBSTA. EQ 1) SAVE COUNTRY ID
   REGION ID
   ZONE ID
   STRATA ID
   in PRNTID

5. Test for a change in identification area
   STRATA
   ZONE
   REGION

6. If same, perform calculations

   XNA = NA
   XN = N
   XR = AREA/(XNA * AREAPS)
   XNR = XN * XR * AREAPS
   HWAK = XNR * XHPW/100
   HWAS = HWAS + HWAK
   TWAK = XNR * XTPW/100
   TWAS = TWAS + TWAK
   NSUBGP (IGPNO) = NSUBGP (IGPNO) + 1
   IF (IGPNO. EQ. 1 or 2)
   NSEGGP (IGPNO) = NSEGGP (IGPNO) + NSEG
   XINBSS = XBTPW * HWAK + XINBSS
   XINCVS = (XCV2 * TWAK) **2 + XINCNS
   XSMCV = TWAK - HWAK
   XSMCVS(1) = XSMCVS(1) + XSMCV **2
   XSMCVS(2) = XSMCV(2) + XSMCV
   J = J + 1
   Read another detail record and go to 5.

7. Change in strata ID

   HWA = HWAS
   TWA = TWAS
   XINBS = XINBSS/HWA
   XSIMBS = (TWA - HWA)/HWA
   $XINCV = (XINCVS)^{1/2}/TWA$
   $XSIMCV = (J/(J - 1) * (XSMCVS(1) - 1/J * XSMCVS(2)^2))^{1/2}$

NSUBGP(i) = NSBGPS(i)
NSEGGP(j) = NSGGPS(j) where i = 1, 2 & 3; j = 1 & 2
Call REPRT1 with ITOTYP = 1
HWAZ = HWAZ + HWA
TWAZ = TWAZ + TWA
XINBSZ = XINBSZ + XINBS
XSMBSZ = XSMBSZ + XSIMBS
XINCVZ = XINCVZ + XINCV
XSMCVZ = XSMCVZ + XSIMCV
NSBGPZ(i) = NSBGPZ(i) + NSUBGP(i)
NSGGPZ(i) = NSGGPZ(j) + NSEGGP(j)
HWAS = 0
TWAS = 0
XINBSS = 0
XINCVS = 0
XSMCVS(1) = 0
XSMCVS(2) = 0
J = 0
NSBGPS(i)
NSGGPS(j)
If no change in zone ID, read another detail record and go to 5.

8. Change in zone ID

HWA = HWAZ
TWA = TWAZ
XINBS = XINBSZ
XSIMBS = XSMBSZ
XINCV = XINCVZ
XSIMCV = XSMCVZ
NSUBGP(i) = NSBGPZ(i)
NSEGGP(j) = NSEGGZ(j)
Call REPRT1 with ITOTYP = 2
HWAR = HWAR + HWA
TWAR = TWAR + TWA
XINBSR = XINBSR + XINBS
XSMBSR = XSMBSR + XSIMBS
XINCVR = XINCVR + XINCV
XSMCVR = XSMCVR + XSIMCV
NSBGPR(i) = NSBGPR(i) + NSUBGP(i)
NSGGPR(j) = NSGGPR(j) + NSEGGP(j)
HWAZ = 0
TWAZ = 0
XINBSZ = 0
XSMBSZ = 0
XINCVZ = 0
XSMCVZ = 0
NSBGPZ(i) = 0
NSEGGZ(j) = 0
If no change in region ID, read another detail record and go to 5.

9. Change in region ID
   Move region values to common variables as in 8 above.
   CALL REPRT1 with ITOTYP = 3.
   Sum region values into country values.
   Zero out region values.
   If more data to be read, read another detail record and go to 5.

10. All records have been processed.
    Move country values to common variables as in 8 above.
    CALL REPRT1 with ITOTYP = 4

11. Return to POUT.

Subroutine POPDRV

Purpose:

This routine is the driver routine for the generation of the Population Reports (RPTYPE = 02).

Input:

/CARDIN/	RPTYPE
	PARMTR(5)
	IERR

Output:

/CARDIN/	JPARM	
	IERR	Reset to zero each subroutine call

Linkage:

CALL POPDRV — called by POUT

Subroutines Used:

CALL SAMERR	Generates the Population Sampling Error Report (PARMTR(1) ≠ 0)
CALL CAMSER	Generates the Population CAMS Error Report (PARMTR(2) ≠ 0)
CALL YLDERR	Generates the Population Yield Error Report (PARMTR(3) ≠ 0)
CALL ANPERR	
CALL GENRNG	Generates ranges for histogram

Local Variable:

JERR

Processing:

1. CALL GENRNG — generates the range intervals for the histogram data.
2. IF (PARMTR(1) ≠ 0 — set JPARM = 1
   CALL SAMERR

3. IF PARMTR(2) ≠ 0 set JPARM = 2
CALL CAMSER

4. IF PARMTR(3) ≠ 0 set JPARM = 3
CALL YLDERR

5. IF PARMTR(4) ≠ 0 set JPARM = 4
CALL ANPERR

6. IF PARMTR(5) ≠ 0 set JPARM = 5
CALL ANPERR

7. Return.

Subroutine SAMERR

Purpose:

This routine produces the Population Sampling Error Report. It accesses the SUBHST and Segment Truth (SEGTRU) Files through the sequential file read routine FILSEQ. It computes the required quantities and passes them to routine REPRT2 for printing.

Input:

/CARDIN/	LEVEL	
/HSTOGM/	NRANGE	
	RANGE (52)	
	NZRNG	
/FILBUF/	FILENM (2)	
	ICNTRY	
	IREGN	
	IZONE	
	ISTRTA	
	ISBSTA	
	ICASE	
	NSEG	Number of sample segments in substrata
	AREA	Land area of substrata
	XTPW	True proportion of wheat, Kth substratum
	N	No. of agricultural segments in substrata
	NA	No. of allocated segments in substrata
	XTPWI	True proportion of wheat ith segment

Output:

/PRNTID/	CNTRY
	REGION
	ZONE
	STRATA
	ICASEN
	FILNAM (2)
	ITOTYP
	NAMFMT
/READNG/	READNG (3200)

/HSTOGM/	XMVAL
	STDDEV
	XMPSUM
	SDPSUM
	NRANGE
	IRANGE (51)
	NTOTL
	RANGES (52)
	MINVAL
	MAXVAL
	REF
	NZRNG

Linkage:

CALL SAMERR called by POPDRV

Subroutines Used:

CALL FILSEQ (IH)	reads data files SUBHST and SEGTRU. IH - header/detail record flag = 1 read header records = 0 read detail records
CALL REPRT2	prints the Population Sampling Error Report.

Local Variable Description:

IH	Flag passed to routine FILSEQ whether IH = 1, read header records; IH = 0, read detail records.
AREAPS	
XNA	
XR	
XN	
XM	
J	Substrata counter at zone, region or country level
DELSB	Sum of proportion of wheat at substrata level
DELERR	
DELSQ	Square of DELERR
KXM	No. of segments in substrata

Processing:

1. Initialize arrays, constants, and work areas.

2. Get header record.
   Check if IERR = 0 - no match on case ID.
   Save case number and file name.

3. Get detail records.
   Check for EOF - go to 8.

4. First time through initialize AREA ID.

5. If not first time through check for change in substrata ID and go sum segments (8.).

6. Accumulate true proportion wheat values.

7. Read another detail record.

8. Compute the sum for the substrata.

9. Save the reading in READNG.

10. Find minimum and maximum reading.

11. Add reading to sums.

12. Test level complete - otherwise go to 3.

13. If all readings processed for level compute the mean, standard deviation, etc. Tally the percentage of reading against histogram ranges.

14. Call REPRT2.

15. Return.

Subroutine CAMSER

Purpose:

This routine produces the Population CAMS Error Report. It accesses the SUBHST and CAMS Output (CAMSF) Files through the sequential file read routine FILSEQ. It computes the required quantities and passes them to routine REPRT2 for printing.

Input:

/CARDIN/	LEVEL	
/HSTOGM/	NRANGE	
	RANGES	
	NZRNG	
/FILBUF/	FILENM (2)	
	ICNTRY	
	IREGN	
	IZONE	
	ISTRTA	
	ISBSTA	
	ICASE	
	NSEG	No. of sample seg. in substrata
	AREA	Land area of substrata
	XTPW	True proportion of wheat, Kth substratum
	N	No. of agricultural seg. in substrata
	NA	No. of allocated seg. in substrata
	XTPWI	True proportion of wheat, Ith segment
	PRDATE (4)	Prediction dates of the four bio-windows
	XEPW (4)	Est. proportion of wheat of the four bio-windows
/SELCTN/	BIOWD (4)	
	WPRTY (4)	
	IPRD (3, 14)	
/PRTCTL/	KOUT	

Output:

/PRNTID/	CNTRY
	REGION
	ZONE
	STRATA
	ICASEN
	FILNAM (2)

/HSTOGM/ XMVAL
STDDEV
XMPSUM
SDPSUM
NRANGE
IRANGE (51)
NTOTL
RANGES (52)
MINVAL
MAXVAL
PRDATE (3)
NZRNG
REF

/READNG/

Linkage:

CALL CAMSER called by POPDRV

Subroutines Used:

CALL FILSEQ (IH) reads data files SUBHST and CAMSF.
IH = 1 read header records
= 0 read detail records

CALL FZULU (IDATE, IOUT) converts Zulu date to calendar date.
IDATE = Zulu date, integer
IOUT(1) = year, integer
IOUT(2) = month, integer
IOUT(3) = day, integer

CALL LFPA (FLDA, LMO, LYR, ALFGM, DAYS) converts calendar date to Zulu.
FLDA = day, flt. pt.
LMO = month, integer
LYR = year, integer
ALFGM = dummy
DAYS = Zulu date, flt. pt.

CALL REPRT2 prints the Population CAMS Error Report.

CALL PAGHDR

Local Variable Description:

IH
AREAPS
XNA
XR

XN
XNR
XM
J
DELSB
DELERR
DELSQ

Processing:

1. Initialize arrays, constants and work areas.

2. Get bio-window number or prediction date.

3. Get header record.
Verify if case number match file.

4. Get detail records.
Check for EOF, go to 12.

5. Set ID area if first time through.

6. If not first time through check for change in substrata ID; if so, go to 12.

7. Accumulate est. proportion wheat for bio-window.

8. Get another detail record for bio-window.

9. Get matching prediction date values if prediction date convert to Zulu. Check against file dates - if no match, skip to next prediction point, go to 2.

10. Accumulate information for prediction date.

11. Read another detail for prediction date.

12. Compute sum of substrata.

13. Save the reading.

14. Find minimum and maximum reading.

15. Add reading to sums.

16. Test level complete - otherwise go to 4.

17. If complete, process all values for level. Compute mean, standard deviation, etc. Tally readings against histogram ranges.

18. Call REPRT2.

19. Go get another prediction point.

20. If done, return.

Subroutine FREQAC

Purpose:

This routine produces the Frequency of Sample Segment Acquisition Report by scanning the Data Acquisition File (ACQUIS) and tallying the combination of accesses for each segment.

Input:

/FILBUF/	FILENM (2)
	ICNTRY
	IREGN
	IZONE
	ISTRTA
	ICASE
	KWIN (4)

Output:

/PRNTID/	CNTRY
	REGION
	ZONE
	STRATA
	ICASEN
	FILNAM (2)
	ITOTYP
/BWCMBN/	NSEGS
	BIOCMB (16)

Linkage:

CALL FREQAC called by POUT

Subroutines Used:

CALL FILSEQ
REPRT3

Local Variable Description:

KNTBIO (16, 3)	(I, J)	I - bio-window count J - zone, region, country level counts of segment acquisitions
KNT (3)		Total no. of segments (readings) at each level (zone, etc.)
KBIOZ (16)		≡ KNTBIO(I, 1)
KBIOR (16)		≡ KNTBIO(I, 2)
KBIOC (16)		≡ KNTBIO(I, 3)

IFIRST	First time flag	= - 0 - first time ≠ - not first time
IH		
KNTZ	≡ KNT(1)	
KNTR	≡ KNT(2)	
KNTC	≡ KNT(3)	

Processing:

1. Initialize storage arrays.

2. Get header record and save ID information.

3. Get detail record.
   If EOF, go to 8.

4. If first time through save ID (area) information.

5. If not first time through, check for change in ID area and go to 8.

6. Test bio-windows for accesses and tally the combinations.

7. Go to 3 to get another detail record.

8. Compute percentage of combinations at zone level and print that zone.

9. Add zone values to region. If change in zero, compute percentage and print.

10. Add region to country values. If EOF, compute percentages and print.

11. When completed, return.

Subroutine YLDERR

Purpose:

This routine produces the Population Yield Error Report. It accesses the SUBHST and YES Output (YESOUT) Files through the sequential file read routine FILSEQ. It computes the required quantities and passes them to routine REPRT2 for printing.

Input:

/CARDIN/	LEVEL
/HSTOGM/	NRANGE RANGES NZRNG
/FILBUF/	FILENM (2) ICNTRY IREGN IZONE ICASE AREA XTPW N NA PRDATE (6) XEYLD (6) XTYLD
/SELCTN/	BIOWD (4) IPRD (3, 14)
/PRTCTL/	KOUT

Output:

/PRNTID/	CNTRY REGION ZONE ICASEN FILNAM (2) NAMFMT

c.4

/HSTOGM/	XMVAL
	STDDEV
	XMPSUM
	SDPSUM
	NTOTL
	MINVAL
	MAXVAL
	PRDATE (3)
	REF
/READNG/	READNG (500)

Linkage:

CALL YLDERR called by POPDRV

Subroutines Used:

CALL FILSEQ reads data files SUBHST and YESOUT

CALL FZULU (IDATE, IOUT)

CALL LFPA (FLDA, LMO, LYR, ALFGM, DAYS)

CALL REPRT2 prints yield error

CALL PAGHDR

Local Variable Description:

IH
AREAPS
XNA
XN
XNR
J
KXS
DELSTA
REFSTA
DELERR
DELSQ

Processing:

1. Initialize arrays, constants, etc.

2. Get bio-window number or prediction date.

3. Get header record.
   Verify match on case number.

4. Get detail records.
   Check EOF, go to 10.

5. Set ID area if first time through.

6. If not first time through check for change in level ID.

7. Determine prediction point processing.

8. Accumulate information.

9. Read another detail record.

10. Compute sum of strata.

11. Save reading.

12. Find minimum and maximum readings.

13. Add reading to sums.

14. Test level complete, otherwise go to 4.

15. If complete, process all values for level. Compute mean, standard deviation, etc. Tally readings against ranges.

16. Call REPRT2.

17. Get another prediction point.

18. Otherewise, return.

Subroutine REPRT1

Purpose:

Control and print the Substrata Reference Data Report.

Input:

/CARDIN/	AUNITS
/PRTCTL/	NPAGE NLINES MAXLIN KOUT HEADER
/PRNTID/	CNTRY REGION ZONE STRATA ICASEN
/SUBVAL/	HWA TWA NSUBGP (3) NSEGGP (2) XINBS XSIMBS XINCV XSIMCV ITOTYP
/CONVRT/	CTAMER (3) CTMTRC (3)

Output:

The printed report.

Linkage:

CALL REPRT1 called from SUBREF

Subroutines Used:

None.

Local Variable Description:

Subroutine REPRT2

Purpose:

Subroutine REPRT2 produces the printed reports for the Population and Monte Carlo Reports.

The first time through for any run, the histogram ranges are printed for all parameters. If the confidence levels are requested as part of a larger case, they have their own ranges. The ranges are printed only once just before the confidence level reports.

For each report, the histogram values are computed from the number of values in each range. The percentages are summed at each range point. These percentages are then printed as the histogram data followed by its mean, standard deviation, reference value, minimum and maximum readings, and number of readings. For each population report the mean and standard deviation of the population sum are also printed.

Finally, the identification line containing country, region, zone and prediction point is printed.

Subroutine REPRT3

Purpose:

Subroutine REPRT3 produces the printed report of the Frequency of Sample Segments Acquisitions.

Input:

/PRNTID/	CNTRY
	REGION
	ZONE
	ICASEN
	FILNAM (2)
	ITOTYP
/BWCMBN/	NSEGS
	BIOCMB (16)
/PRTCTL/	NPAGE

Output:

The printed report.

Linkage:

CALL REPRT3

Subroutines Used:

CALL PAGHDR (N) controls the line count on the printed output.

Local Variable Descriptors:

MPAGE	checked against NPAGE to determine if the page number has been incremented and if so, prints the report headings.
JFIRST	first time flag.

Processing:

1. Print page headings if JFIRST = 0.
2. Go to 3. if ITOTYP = 1
   5. if ITOTYP = 2
   7. if ITOTYP = 3

3. Print report headings if NPAGE ≠ MPAGE.
4. Print the zone level values for the 16 combinations. Go to 9.
5. Print report headings if NPAGE ≠ MPAGE.
6. Print the region level values for the 16 combinations. Go to 9.
7. Print report headings if NPAGE ≠ MPAGE.
8. Print the country level values for the 16 combinations.
9. Return.

Subroutine PAGHDR (N)

Purpose:

Subroutine PAGHDR controls the line count on printed output and provides a line of printed heading at the top of each output page including the page number.

Input:

/PRTCTL/ NPAGE
NLINES
MAXLIN
KOUT
HEADER (16)

Linkage:

CALL PAGHDR (N)

where N = count of lines to be printed
-N = skip to a new page before printing N lines

Local Variable Descriptions:

LPRINT = a flag which indicates to the routine what the previous command was
= 1 increment the line count
= 0 skip to the bottom of page
= -1 the first time the routine called, always skip to new page

NLINES = the current starting line number of a page

Processing:

1. Test LPRINT value for previous command.
2. LPRINT less than zero; set constants N1 and N2 and reset LPRINT = 1. Go to 11.
3. LPRINT equals zero; set LPRINT = 1. Go to 11.
4. Test N for less than or equal zero. If yes, go to 11.
5. LPRINT greater than zero. Compute J = NLINES+N.
6. Test J if greater than N1 constant and if so, go to 8.

7. Otherwise set NLINES = J; go to 13.

8. If N not equal zero, go to 13.

9. Move line printer to bottom of page by printing an appropriate number of blank lines.

10. Set LPRINT = 0. Go to

11. Increment page number and print header at the top of the new page.

12. Reset NLINES value.

13. Return to calling routine.

Subroutine GENRNG (ICL)

Purpose:

Subroutine GENRNG generates the histogram ranges from the input interval or nominal interval values except for the confidence levels of RPTYPE = 03 which have a special set of intervals.

Input:

/RANGE/	START
	INTVL1
	BREAK1
	INTVL2
	BREAK2
	INTVL3
	STOP

Output:

/HSTOGM/	NRANGE
	RANGES (52)
	NZRNG

Linkage:

CALL GENRNG (ICL) where

ICL = 0	generate standard ranges for histogram data
ICL ≠ 0	generate special ranges for the confidence levels; that is, no RNGMIN or RNGMAX entries.

Local Variable Descriptions:

K	is count of the number of range values during generation.
N1	is the number of intervals between any two breakpoint values.
VALU	is the actual range value computed from the initial breakpoint value plus the interval increment.
RNGMIN, RNGMAX	very large minimum and maximum extreme points

Processing:

1. Zero all range values.
2. Set NRANGE and NZRNG to zero.
3. Set K = 1.
4. If ICL = 0 then RANGES(K) = RNGMIN.
5. If ICL ≠ 0 then K = 0.
6. Compute N1.
7. Generate the N1 RANGES values continually searching for the special case of zero.
8. Add N1 to NRANGE value.
9. If INTVL2 value ≠ 0, go to 11.
10. Set RANGES(K+1) = BREAK1. Go to 18.
11. Compute N1 for next set of ranges.
12. Repeat the process for generation of the RANGES for the next set.
13. Add N1 to NRANGE value.
14. If INTVL3 value ≠ 0, go to 16.
15. Set RANGES(K+1) = BREAK2. Go to 18.
16. Compute N1 for last set of ranges.
17. Repeat the generation process of the RANGES for the final set.
18. If ICL ≠ 0, go to 21.
19. Set RANGES(K+2) = RNGMAX.
20. Add 2 to NRANGE.
21. Return.

Subroutine FILSEQ

Purpose:

Reads the required sequential file determined by RPTYPE and returns the required data in FILBUF common block.

Input:

/CARDIN/	RPTYPE	
	PARMTR (5)	
	ICASIN (5)	
/FSBHST/		
/FSGTRU/		
/FCAMSF/		
/FYESOT/		
/FACQUS/		
/PRTCTL/		

Output:

/FILBUF/	FILNAM (2)	
	ICNTRY	
	IREGN	
	IZONE	
	ISTRTA	
	ISBSTA	
	ICASE	
	KBUF (100)	see individual routines for which quantities used

Linkage:

CALL FILSEQ (IH)

Inputs:

IH = 1 Reads header record
IH = 0 Reads detail record(s)

Output:

IH = -1 EOF indicator

Local Variable Description:

Subroutine CLMOVE (NCL)

Purpose:

Subroutine CLMOVE moves one set of confidence levels for RPTYPE = 03 and JPARM = 4 from the /READNG/ buffer to /FILBUF/.

Input:

/READNG/	CLEWA (100)	
	CLEPRD (100)	
	CLATEC (100)	
	CLPTEC (100)	
	CLATWC (100)	
	CLPTWC (100)	

Output:

/FILBUF/	MCREF	confidence level reference value = 100.
	MCERR	confidence level values.

Linkage:

CALL CLMOVE (NCL) where

NCL = 1	Area confidence level est/est
= 2	Production confidence level est/est
= 3	Area confidence level true/est
= 4	Production confidence level true/est
= 5	Area confidence level true/WC
= 6	Production confidence level true/WC

Local Variable Descriptions:

M = 4900 + 100 * NCL location of the requested confidence level in /READNG/ buffer.

Processing:

1. Compute the location of the confidence level requested in /READNG/ buffer.
2. Move the 100 values from /READNG/ to array MCERR.
3. Return.

Subroutine PUNITS (FACTOR, IUNITS)

Purpose:

Subroutine PUNITS sets up the proper print and conversion units for the Population and Monte Carlo Reports.

Input:

/CARDIN/	RPTYPE
	JPARM
	AUNITS
/CONVRT/	CTAMER (3)
	CTMTRC (3)

Linkage:

CALL PUNITS (FACTOR, IUNITS) where

FACTOR	=	the conversion factor
IUNITS	=	the descriptor of the units

Local Variable Descriptions:

JFAC	=	an array of indices for the CTAMTER or CTMTRC array
IFAC	=	index into the JFAC array
NFAC	=	index into the CTAMER or CTMTRC array
IAMERU	=	an array of English units descriptors that correspond to the units in the CTAMER array.
IMTRCU	=	an array of metric units descriptors that correspond to the units in the CTMTRC array.

Processing:

1. Set IFAC = 0.
2. If RPTYPE = 03, set IFAC = IFAC + 5.
3. IFAC = IFAC + JPARM.
4. NFAC = JFAC (IFAC).
5. If AUNITS $\neq$ 0, go to 12.
6. Process English units.

7. FACTOR = 1.0. If RPTYPE = 03 and NFAC = 3, go to 9.
8. FACTOR = CTAMER (NFAC).
9. MFAC = (NFAC - 1) * 3 (the index into IAMERU array).
10. Move descriptor from IAMERU to IUNITS array.
11. Go to 17.
12. Process metric units.
13. FACTOR = 1.0. If RPTYPE = 03 and NFAC = 3, go to 15.
14. FACTOR = CTMTRC (NFAC).
15. MFAC = (NFAC - 1) * 3 (the index into IMTRCU array).
16. Move descriptor from IMTRCU to IUNITS array.
17. Return.

Subroutine MCHIST

Purpose:

This routine produces the Monte Carlo Reports. It accesses the CAS Distribution Output File (CASDIS) through the direct access file read routine FILDA. It computes the required quantities and passes them through common to routine REPRT2 for printing.

Input:

/CARDIN/	LEVEL
	JPARM
	PARMTR
	ICASIN
/HSTOGM/	NRANGE
	RANGES (52)
	NZRNG
/FILBUF/	FILENM (1)
	ICNTRY
	IREGN
	IZONE
	ICASE
	NT
	NREGS
	NZTOT
	NBW
	NPD
	IBIOWD (4)
	IPRDAT (14)
	MCREF
	MCERR (100)
/SELCTN/	BIOWD (4)
	IPRD (3, 14)
/PRTCTL/	KOUT
/RANGE/	START
	INTVL1
	BREAK1
	INTVL2
	BREAK2
	INTVL3
	STOP

Output:

/PRNTID/	CNTRY REGION ZONE ICASEN FILNAM (1) NAMFMT.
/HSTOGM/	XMVAL STDDEV REF MINVAL MAXVAL NTOTL DATPRD (3)
/READNG/	READNG (5600)

Linkage:

CALL MCHIST called by POUT

Subroutines Used:

CALL FILDA (IH, IREC)

CALL FZULU (IDATE, IOUT)

CALL LFPA (FLDA, LMO, LYR, ALFGM, DAYS)

CALL REPRT2

CALL PAGHDR

Local Variable Description:

IH
IREC
J
DELERR
DELSQ

Processing:

1. Initialize arrays, constants, etc.

2. Get header record; verify case numbers on PARMTR values requested.
   DO I = 1, 4
   If PARMTR(I) ≠ 0, then verify ICASIN(I) = ICASE6.
   If not equal, write message "ICASIN(I) does not match case number on file."
   Set PARMTR(I) = -PARMTR(I)
   Save in /PRNTID/ case number
   file name
   country ID
   Set NAMFMT = 1 (file name format type).

3. Loop on PARMTR(I), I = 1 to 4. If PARMTR(I) ≤ 0 skip, otherwise JPARM = I.

4. Loop on LEVEL (set from 1 to 3) if PARMTR(4) ≠ 0 LEVEL = 3 only.

5. Set IPNT: Get bio-window or prediction date.
   For BIOWD(I), IREC = I, I = 1 to 4.
   For IPRD(3, I), IREC = 4 + I, I = 1 to 14.

6. If LEVEL = 1, then get all NZTOT records.
   If LEVEL = 2, then get all NREGS records.

Direct Access File Requirements for
RPTYPE = 03 the
CAS Distribution Output File

IH = 1 Read header record and save in /FILBUF/

FILENM (1)	=	NAME6
ICNTRY	=	COUN6
ICASE	=	ICASE6
NT	=	NT6
NREGS	=	NREG6
NZTOT	=	NZTOT6
NBW	=	NBW6
NPD	=	NPD6
IBIOWD (4)	=	BWNBR6
IPRDAT (14)	=	PDNBR6

IH = 0, JPARM = 1, 2, or 3, NREC = IREC = N and LEVEL = 1 or 2

Read detail record.

Subroutine ANPERR

Purpose:

This routine produces if PARMTR(4) ≠ 0 the Population Area Error Report or if PARMT4(5) ≠ 0 the Population Production Error Report. It accesses the CAS Cum Output File (CASF) through the direct access file read routine FILDA. It computes the required quantities and passes them through common to routine REPRT2 for printing.

Input:

/CARDIN/ LEVEL
JPARM

/HSTOGM/ NRANGE
RANGES (52)
NZRNG

/FILBUF/ FILENM (1)
ICNTRY
IREGN
IZONE
ICASE
NT
NSTRAT
NBW
NPD
IBIOWD (4)
IPRDAT (14)
APVAL (2)
Note: APVAL(1) ≡ AERRS ≡ PRERRS
APVAL(2) ≡ TWAS ≡ TPRODS

/SELCTN/ BIOWD (4)
IPRD (3, 14)

/PRTCTL/ KOUT

Output:

/PRNTID/ CNTRY
REGION
ZONE
ICASEN
FILNAM (1)
NAMFMT

/HSTOGM/	XMVAL
	STDDEV
	XMPSUM
	SDPSUM
	NTOTL
	MINVAL
	MAXVAL
	DATPRD (3)
	REF
/READNG/	READNG (500)

Linkage:

CALL ANPERR called by POPDRV

Subroutines Used:

CALL FILDA reads data file CASF

CALL FZULU (IDATE, IOUT)

CALL LFPA (FLDA, LMO, LYR, ALFGM, DAYS)

CALL REPRT2 prints report

CALL PAGHDR

Local Variable Description:

IH
J
DELSTA
REFSTA
DELERR = Σ (READNG)
DELSQ = Σ (READNG)

Processing:

1. Initialize arrays, constants, etc.

2. Get bio-window number or prediction date.

3. Get header record.
   Verify match on case number.
   Move case number, file name, and country ID to /PRNTID/.
   Set NAMFMT flag.

4. Get detail record (strata records only).
   If all NSTRAT done, go to 10.

5. Set ID area if first time through REGION and ZONE in /PRNTID/.

6. If not first time through, check for change in LEVEL ID.

7. Determine if prediction point valid for record.

8. Accumulate values (Eqns. 19 and 20, or 21 and 22).
   READNG(J) = DELERRS(S)
   MINVAL, MAXVAL
   DELERR = DELERR + READNG(J)

9. Read another detail record (go to 4).

10. Compute sums (Eqns. 8, 9, 10, 11, and 12).

11. Tally readings against ranges.

12. Call REPRT2.

13. Get another prediction point at 2.

14. Otherwise, return.

Subroutine FILDA (IH, IPNT, NREC)

Purpose:

This routine calls the direct access routines to read the direct access files determined by RPTYPE = 02 or 03 and JPARM value, and returns the required data in common blocks FILBUF and READNG.

Input:

/CARDIN/	RPTYPE JPARM LEVEL
/FBLKBX/	FBLKBX (504)
/FCASCM/	FCASCM (28)
/FCASDS/	FCASDS (303)

Output:

/FILBUF/	FILENM (1) ICNTRY IREGN IZONE ICASE NT NSTRAT NREGS NZTOT NBW NPD IBIOWD IPRDAT APVAL (2) MCREF MCERR (100)
/IXRCD/	IXRCD (1135)
/READNG/	CLEWA (100) CLEPRD (100) CLATEC (100) CLPTEC (100) CLATWC (100) CLPTWC (100)

Linkage:

CALL FILDA (IH, IPNT, NREC)

Inputs:

IH = 1 Read header record
IH = 0 Read detail record

IPNT = for RPTYPE = 02 the number of the prediction point (or group no. within each detail record)

NREC = for all RPTYPE the number of the record to be read

Subroutines Used:

CALL RANACF (IFILE, IREC, BUF, N, IX, L, IOPT)

Local Variable Description:

IFILE = 14 for RPTYPE = 02
4 for RPTYPE = 03

IREC = 1 for header record
for RPTYPE = 02
= 63 through 387
for RPTYPE = 03
= NREC

BUF = for RPTYPE = 02 FBLKBX
for RPTYPE = 03 FCASDS

N = NWRDS = 504 for RPTYPE = 02
303 for RPTYPE = 03

IX = IXRCD an array dimensioned 1135

L = 387 for RPTYPE = 02
1135 for RPTYPE = 03

IOPT = Entry point option
= 0 to open the file
= 1 to read the file
= -1 to close the file

IH = 1 Read the header record
= 0 Read a detail record

IPNT = Prediction point number
= 1-4 for bio-windows 1-4
= 5-18 for prediction dates 1-14

NREC = No. of the detail record to read

Processing:

1. If IH = 1 Open file and read header record
   a. file CASF if RPTYPE = 02
   b. file CASDIS if RPTYPE = 03

2. For RPTYPE = 02
   If IH = 0 Read detail record no. NREC into FBLKBX.
   Move IPNT group to FCASCM.
   See note 1.

3. For RPTYPE = 03
   IH = 0 Read detail record no. NREC into FCASDS.
   See note 2.

Note 1:

For CAS Cum Output File (RPTYPE = 02)

Read a detail record into FBLKBX.

Move group IREC into FCASCM.

For all JPARM values
  Move REGN5 to IREGN
  ZONE5 to IZONE

For JPARM = 4 move TWAS to APVAL(1)
  AERRS to APVAL(2)

For JPARM = 5 move TPRODS to APVAL(1)
  PRERRS to APVAL(2)

Note 2:

For CAS Distribution File (RPTYPE = 03)

If LEVEL = 3 (country)

Read first country record to FCASDS.

For JPARM = 1 move AEREF to MCREF
  AERR(I) to MCERR(I), I = 1, 100

For JPARM = 2 move PEREF to MCREF
  PRERR(I) to MCERR(I), I = 1, 100

For JPARM = 3 move 100 to MCREF
  YERR(I) to MCERR(I), I = 1, 100

For JPARM = 4 go to get second country record.

JPARM < 4 Return

Read third country record to FCASDS.

Move CLPTC6(I) to CLPTEC(I)
CLATW6(I) to CLATWC(I)
CLPTW6(I) to CLPTWC(I), I = 1, 100

Return.

If LEVEL = 1 or 2 (zone or region)

Read zone or region record into FCASDS.

For JPARM = 1 move AEREF to MCREF
AERR(I) to MCERR(I), I = 1, 100

For JPARM = 2 move PEREF to MCREF
PRERR(I) to MCERR(I), I = 1, 100

For JPARM = 3 move 100 to MCREF
YERR(I) to MCERR(I), I = 1, 100

PART V

SUBROUTINE LISTINGS

```
• EL, ANPERR,1,760511, 59150   , 1

000001            SUBROUTINE ANPERR                                                    ANPERR
000002      C        GENERATES THE POPULATION AREA ERROR REPORT OR THE POPULATION      ANPERR
000003      C        PRODUCTION ERROR REPORT                                           ANPERR
000004      C                                                                          ANPERR
000005      C        COMMON BLOCKS                                                     ANPERR
000006            COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                               CARDIN
000007           *,              ICASIN(5),LEVEL,JPARM,IERR                            CARDIN
000008            INTEGER RPTYPE,AUNITS,PARMTR                                         CARDIN
000009            COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA                  FILBUF
000010           *,              ISBSTA,ICASE,MXSEG                                    MODF
000011           *,              KBUF(125)                                             MODF
000012      C                                                                          FILBUF
000013            DIMENSION KWIN(4),IBIOMD(4),IPRDAT(14),PRDATE(6),XEPW(4),XEYLD(6)    FILBUF
000014           *,          APVAL(2)                                                  FILBUF
000015           *,          MCERR(100)                                                MODF
000016      C                                                                          FILBUF
000017            REAL    MCREF,MCERR                                                  MODF
000018            INTEGER FILENM,PRDATE                                                FILBUF
000019      C                                                                          FILBUF
000020            EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                                FILBUF
000021           *,  (KBUF(2),IGPRD,NSTRAT,NREGS)                                      MODF
000022           *,  (KBUF(3),XMPW,NZTOT)                                              MODF
000023           *,  (KBUF(4),AREA,NBR)                                                MODF
000024           *,  (KBUF(5),XTPW,NPD)                                                MODF
000025           *,  (KBUF(6),N,IBIOMD(1))                                             MODF
000026           *,  (KBUF(7),LA)                                                      MODF
000027           *,  (KBUF(8),XMTPW)                                                   MODF
000028           *,  (KBUF(9),XCV2)                                                    MODF
000029           *,  (KBUF(10),XTPWI,IPRDAT(1))                                        MODF
000030           *,  (KBUF(11),PRDATE(1))                                              MODF
000031           *,  (KBUF(17),XEPW(1))                                                MODF
000032           *,  (KBUF(21),XEYLD(1))                                               MODF
000033           *,  (KBUF(24),APVAL(1),MCREF)                                         MODF
000034           *,  (KBUF(25),MCERR(1))                                               MODF
000035           *,  (KBUF(27),XTYLD)                                                  MODF
000036            COMMON /HSTOGM/ XMVAL,STDDEV,XMPSUM,SDPSUM                           HSTOGM
000037           *,              NRANGE,IRANGE(51),NTOTL,RANGES(52)                    HSTOGM
000038           *,              MINVAL,MAXVAL,DATPRD(3),NZRNG,REF                     HSTOGM
000039      C                                                                          HSTOGM
000040            REAL    MINVAL,MAXVAL                                                HSTOGM
000041            INTEGER DATPRD                                                       HSTOGM
000042            EQUIVALENCE  ( LYR,DATPRD(1) ), ( LMO,DATPRD(2) )                    ANPERR
000043           1  , ( LDA,DATPRD(3) )                                               ANPERR
000044      C                                                                          ANPERR
000045            COMMON /PRNTID/ CNTRY,REGION,ZONE,STRATA,ICASEN                      PRNTID
000046           *,              FILENO,FILNAM(2)                                      PRNTID
000047           *,              ITUTYP,NAMFMT                                         PRNTID
000048      C                                                                          PRNTID
000049            INTEGER CNTRY,REGION,ZONE,STRATA,FILENO,FILNAM                       MODF
000050            COMMON /PRTCTL/ NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)                  PRTCTL
000051            INTEGER HEADER                                                       PRTCTL
000052            COMMON /READNG/ READNG(5600)                                         MODF
000053      C                                                                          MODF
000054            DIMENSION CLENA(100),CLEPRD(100),CLATEC(100),CLPTEC(100)             MODF
000055           *,          CLATMC(100),CLPTMC(100)                                   MODF
000056      C                                                                          MODF
000057            EQUIVALENCE                          (READNG(5001),CLENA(1))         MODF
000058           *,  (READNG(5101),CLEPRD(1)),         (READNG(5201),CLATEC(1))        MODF
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000059            *,  (READNG(5301),CLPTEC(1)),      (READNG(5402),CLATWC(1))      MODF
000060            *,  (READNG(5501),CLPTWC(1))                                     MODF
000061            COMMON /SELCTN/ BIOWD(4),WPRTY(4),IPRD(3,14)                     SELCTN
000062          C                                                                  SELCTN
000063            INTEGER BIOWD,WPRTY                                              SELCTN
000064          C                                                                  ANPERR
000065          C                                                                  ANPERR
000066          C       INITIALIZE COUNTERS                                        ANPERR
000067            IPNT= 0                                                          ANPERR
000068            JPRD= 0                                                          ANPERR
000069            KPRD= 0                                                          ANPERR
000070            JBW = 0                                                          ANPERR
000071            DATPRD(1)= 0                                                     ANPERR
000072            DATPRD(2)= 0                                                     ANPERR
000073            DATPRD(3)= 0                                                     ANPERR
000074            IH= 1                                                            ANPERR
000075          C                                                                  ANPERR
000076          C       READ HEADER FROM CAS CUM FILE                              ANPERR
000077            CALL FILDA (IH,IH,IH)                                            ANPERR
000078          C                                                                  ANPERR
000079          C       CHECK CASE NUMBER                                          ANPERR
000080            IF ( ICASE .EQ. ICASIN(4) )  GO TO 150                           ANPERR
000081            CALL PAGHDR(-2)                                                  ANPERR
000082            WRITE (IOUT,1)  ICASE,ICASIN(4)                                  ANPERR
000083          1 FORMAT (59H ***** ERROR IN SUBROUTINE ANPERR *****/              ANPERR
000084           1  55H   CASE NUMBER FROM CAS CUM FILE = ,I5,                    ANPERR
000085           2  58H  DOES NOT AGREE WITH ICASIN(4) = ,I5)                     ANPERR
000086            STOP                                                             ANPERR
000087          C                                                                  ANPERR
000088          150 IH= 0                                                          ANPERR
000089            CNTRY= ICNTRY                                                    ANPERR
000090            ICASEN= ICASE                                                    ANPERR
000091            FILNAM(1)= FILENM(1)                                             ANPERR
000092            FILNAM(2)= FILENM(2)                                             ANPERR
000093            NAMENT= 1                                                        ANPERR
000094            XMT= MT                                                          ANPERR
000095          C                                                                  ANPERR
000096          C       PREDICTION POINT LOOP                                      ANPERR
000097          200 JBW= JBW + 1                                                   ANPERR
000098            IF ( JBW .GT. 4 )  GO TO 220                                     ANPERR
000099          C       JBW .LE. 4.  (BIOWINDOW)                                   ANPERR
000100            IBW= IBIOWD(JBW)                                                 ANPERR
000101            IF ( IBW .EQ. 0 )  GO TO 220                                     ANPERR
000102            DATPRD(1)= IBW                                                   ANPERR
000103            IF ( BIOWD(IBW) .NE. 0 )  GO TO 240                              ANPERR
000104          C       IBW = IBIOWD(JBW) .NE. 0, BUT BIOWD(IBW) = 0.              ANPERR
000105          C       (BIOWINDOW IBW PROCESSED BY CAS, BUT REPORT NOT DESIRED)   ANPERR
000106            IPNT= IPNT + 1                                                   ANPERR
000107            GO TO 200                                                        ANPERR
000108          C                                                                  ANPERR
000109          C       JBW .GT. 4.  (PREDICTION DATE)                             ANPERR
000110          220 JPRD= JPRD + 1                                                 ANPERR
000111            IF ( JPRD .GT. NPD )  GO TO 900                                  ANPERR
000112          225 KPRD= KPRD + 1                                                 ANPERR
000113            LYR= IPRD(1,KPRD)                                                ANPERR
000114            IF ( LYR .EQ. 0 )  GO TO 900                                     ANPERR
000115            LMO= IPRD(2,KPRD)                                                ANPERR
000116            LDA= IPRD(3,KPRD)                                                ANPERR
000117            FLDA= LDA                                                        ANPERR
000118            CALL LTPA (FLDA,LMO,LYR,ALPGM,DAYS)                              ANPERR
```

```
000119                IDAYS= DAYS                                                    ANPERR
000120                IF ( IDAYS .EQ. IPRDAT(JPRD) )  GO TO 240                      ANPERR
000121                CALL PAGHDR (-2)                                               ANPERR
000122                WRITE (KOUT,2)  KPRD,LMO,LDA,LYR,IDAYS,IPRDAT(JPRD)           ANPERR
000123              2 FORMAT (39H *****  ERROR IN SUBROUTINE ANPERR *****/           ANPERR
000124             1  7H    THE ,I3,20H-TH PREDICTION DATE ,I2,1H/,I2,1H/,I2 ,4H  = IS ANPERR
000125             2  ,13H  (ZULU DATE)/42H  DOES NOT AGREE WITH THE PREDICTION DATE ANPERR
000126             3  ,I5,22H FROM THE CAS CUM FILE )                               ANPERR
000127        C                                                                     ANPERR
000128                IF ( IDAYS .GT. IPRDAT(JPRD) )  GO TO 230                      ANPERR
000129        C                                                                     ANPERR
000130        C          IDAYS .LT. IPRDAT(JPRD)                                    ANPERR
000131        C          INPUT PREDICTION DATE .LT. PREDICTION DATE FROM FILE.      ANPERR
000132        C          SKIP INPUT DATE.                                           ANPERR
000133                IF ( KPRD .LT. 14 )  GO TO 225                                 ANPERR
000134                GO TO 900                                                      ANPERR
000135        C                                                                     ANPERR
000136        C          IDAYS .GT. IPRDAT(JPRD)                                    ANPERR
000137        C          SKIP DATE FROM CAS CUM FILE.                               ANPERR
000138              230 KPRD= KPRD - 1                                               ANPERR
000139                IPNT= IPNT + 1                                                 ANPERR
000140                GO TO 220                                                      ANPERR
000141        C                                                                     ANPERR
000142        C                                                                     ANPERR
000143              240 IPNT= IPNT + 1                                               ANPERR
000144                ISTRAT= 0                                                      ANPERR
000145                NREC= 62                                                       ANPERR
000146        C                                                                     ANPERR
000147        C          ZONE, REGION, OR COUNTRY LOOP                              ANPERR
000148        C                                                                     ANPERR
000149              300 DO 310 I=1,500                                               ANPERR
000150                READDG(I)= 0.0                                                 ANPERR
000151              310 CONTINUE                                                     ANPERR
000152                DO 320 I=1,51                                                  ANPERR
000153                IRANGE(I)= 0                                                   ANPERR
000154              320 CONTINUE                                                     ANPERR
000155        C                                                                     ANPERR
000156                MINVAL = 1.E10                                                 ANPERR
000157                MAXVAL = -1.E10                                                ANPERR
000158                REGION= 0                                                      ANPERR
000159                ZONE  = 0                                                      ANPERR
000160                DELERR= 0.0                                                    ANPERR
000161                DELSQ = 0.0                                                    ANPERR
000162                REF   = 0.0                                                    ANPERR
000163                J     = 0                                                      ANPERR
000164                NTOTL = 0                                                      ANPERR
000165                IF ( ISTRAT .GT. 0 )  GO TO 415                                ANPERR
000166        C                                                                     ANPERR
000167        C          STRATA LOOP                                                ANPERR
000168              400 ISTRAT= ISTRAT + 1                                           ANPERR
000169                NREC  = NREC + 1                                               ANPERR
000170        C          READ STRATA RECORD FROM CAS CUM FILE                       ANPERR
000171                CALL FILDA (IH,IPNT,NREC)                                      ANPERR
000172                IF ( ISTRTA .GT. 0 .AND. ISTRTA .LT. 10000 )  GO TO 410        ANPERR
000173                CALL PAGHDR(-2)                                                ANPERR
000174                WRITE (KOUT,3)  ISTRTA,ISTRAT,NREC                             ANPERR
000175              3 FORMAT (39H *****  ERROR IN SUBROUTINE ANPERR *****/           ANPERR
000176             1  21H    ILLEGAL STRATA ID ,I10,6H  FOR ,I4,                    ANPERR
000177             2  20H-TH STRATA  (RECORD ,I4,17H OF CAS CUM FILE) )             ANPERR
000178                STOP                                                           ANPERR
```

```
000179      C                                                                 ANPERR
000180       410  IF ( ZONE .NE. 0 )  GO TO 420                               ANPERR
000181      C        FIRST STRATA OF THIS ZONE OR REGION                       ANPERR
000182      C        STORE CURRENT REGION AND ZONE                             ANPERR
000183       415  REGION= IREGN                                                ANPERR
000184            ZONE  = IZONE                                                ANPERR
000185      C                                                                 ANPERR
000186       420  IF ( LEVEL - 2 )  430,440,460                               ANPERR
000187      C                                                                 ANPERR
000188      C        LEVEL = 1  (ZONE)                                        ANPERR
000189       430  IF ( IZONE .NE. ZONE )  GO TO 500                           ANPERR
000190            GO TO 460                                                   ANPERR
000191      C        LEVEL = 2  (REGION)                                      ANPERR
000192       440  IF ( REGION .NE. IREGN )  GO TO 500                         ANPERR
000193      C                                                                 ANPERR
000194      C        STORE POPULATION ERROR IN READNG ARRAY AND UPDATE         ANPERR
000195      C        MIN., MAX. VALUES AND SUMS.                              ANPERR
000196       460  J= J + 1                                                    ANPERR
000197            ERR= APVAL(1)/XNT                                           ANPERR
000198            READNG(J)= ERR                                              ANPERR
000199            MINVAL= AMIN1(MINVAL,ERR)                                   ANPERR
000200            MAXVAL= AMAX1(MAXVAL,ERR)                                   ANPERR
000201            DELERR= DELERR + ERR                                        ANPERR
000202            DELSQ = DELSQ  + ERR*ERR                                    ANPERR
000203            REF= REF + APVAL(2)                                         ANPERR
000204      C        TEST FOR END OF COUNTRY                                  ANPERR
000205            IF (ISTRAT .LT. NSTRAT )  GO TO 400                         ANPERR
000206      C                                                                 ANPERR
000207      C        END OF ZONE, REGION, OR COUNTRY                          ANPERR
000208      C        COMPUTE MEAN VALUE, STANDARD DEVIATION, ETC.  (EQS. 8-12) ANPERR
000209       500  XJ= J                                                       ANPERR
000210            NTOTL= J                                                    ANPERR
000211            REF= REF/XNT                                                ANPERR
000212            XMVAL= 100.0*DELERR/(XJ*REF)                                ANPERR
000213            STDDEV= 0.0                                                 ANPERR
000214            IF ( J .GT. 1 ) STDDEV= 100.0/REF * SQRT( (DELSQ - DELERR**2/XJ)  ANPERR
000215           1                                          /(XJ-1.0) )     ANPERR
000216            XMPSUM= XJ*XMVAL                                            ANPERR
000217            SDPSUM= STDDEV*SQRT(XJ)                                     ANPERR
000218      C                                                                 ANPERR
000219      C        TALLY READINGS IN RANGES                                 ANPERR
000220            DO 550 J=1,NTOTL                                            ANPERR
000221            PCDERR= 0.0                                                 ANPERR
000222            IF ( REF .NE. 0.0 )  PCDERR= 100.0*READNG(J)/REF            ANPERR
000223            IF ( PCDERR .NE. 0.0 )  GO TO 520                           ANPERR
000224      C        PCDERR= 0.                                               ANPERR
000225            IRANGE(NZRNG)= IRANGE(NZRNG) + 1                            ANPERR
000226            GO TO 550                                                   ANPERR
000227      C        PCDERR .NE. 0.                                           ANPERR
000228       520  DO 530 K=1,NRANGE                                           ANPERR
000229            IF ( PCDERR .GE. RANGES(K+1) )  GO TO 530                   ANPERR
000230            IRANGE(K)= IRANGE(K) + 1                                    ANPERR
000231            GO TO 550                                                   ANPERR
000232       530  CONTINUE                                                    ANPERR
000233       550  CONTINUE                                                    ANPERR
000234      C                                                                 ANPERR
000235            CALL REPRT2                                                 ANPERR
000236      C                                                                 ANPERR
000237            IF ( ISTRAT .LT. NSTRAT )  GO TO 300                        ANPERR
000238      C        END OF COUNTRY, GO BACK FOR NEXT PREDICTION POINT.       ANPERR
```

```
ANPERR
ANPERR
ANPERR
ANPERR

      GO TO 200
C
  900 RETURN
      END
```



```
000239
000240
000241
000242
```

```
* L.. CAMSER,1,760513, 61008   , 1
000001                SUBROUTINE CAMSER                                          CAMSER
000002          C                                                                CAMSER
000003          C *** ROUTINE PRODUCES THE POPULATION CAMS ERROR REPORT(S)        CAMSER
000004          C                                                                CAMSER
000005          C                                                                CAMSER
000006          C *** COMMON STORAGE                                             CAMSER
000007          C                                                                CAMSER
000008                COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                     CARDIN
000009               *,              ICASIN(5),LEVEL,JPARM,IERR                  CARDIN
000010                INTEGER RPTYPE,AUNITS,PARMTR                               CARDIN
000011          C                                                                CAMSER
000012                COMMON /RANGE / START,INTVL1,BREAK1,INTVL2,BREAK2           RANGE
000013               *,               INTVL3,STOP                                RANGE
000014          C                                                                RANGE
000015                REAL    INTVL1,INTVL2,INTVL3                               RANGE
000016          C                                                                CAMSER
000017                COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA        FILBUF
000018               *,               IBOSTA,ICASE,MXSEG                         MODF
000019               *,               KBUF(125)                                  MODF
000020          C                                                                FILBUF
000021                DIMENSION KWIN(4),INIOWD(4),IPRDAT(14),PRDATE(6),XEPW(4),XEYLD(6) FILBUF
000022               *,         APVAL(2)                                         FILBUF
000023               *,         MCERR(100)                                       MODF
000024          C                                                                FILBUF
000025                REAL    MCREF,MCERR                                        MODF
000026                INTEGER FILENM,PRDATE                                      FILBUF
000027          C                                                                FILBUF
000028                EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                      FILBUF
000029               *,  (KBUF(2),IGPNO,NSTRAT,NREGS)                            MODF
000030               *,  (KBUF(3),XIPW,NZTOT)                                    MODF
000031               *,  (KBUF(4),APEA,NBW)                                      MODF
000032               *,  (KBUF(5),XTPW,NPD)                                      MODF
000033               *,  (KBUF(6),NI,IOIOWD(1))                                  MODF
000034               *,  (KBUF(7),N1)                                            MODF
000035               *,  (KBUF(8),XBIPW)                                         MODF
000036               *,  (KBUF(9),XCV2)                                          MODF
000037               *,  (KBUF(10),XTPWI,IPRDAT(1))                              MODF
000038               *,  (KBUF(11),PRDATE(1))                                    MODF
000039               *,  (KBUF(17),XEPW(1))                                      MODF
000040               *,  (KBUF(21),XEYLD(1))                                     MODF
000041               *,  (KBUF(24),APVAL(1),MCREF)                               MODF
000042               *,  (KBUF(25),MCERR(1))                                     MODF
000043               *,  (KBUF(27),XIYLD)                                        MODF
000044          C                                                                CAMSER
000045                COMMON /PRNTID/ CNTRY,REGION,ZONE,STRATA,ICASEN            PRNTID
000046               *,               FILENO,FILNAM(2)                           PRNTID
000047               *,               ITOTYP,NAMFMT                              PRNTID
000048          C                                                                PRNTID
000049                INTEGER CNTRY,REGION,ZONE,STRATA,FILENO,FILNAM             MODF
000050          C                                                                CAMSER
000051                COMMON /HSTOGM/ XMVAL,STODEV,XMPSUM,SOPSUM                 HSTOGM
000052               *,               NRANGE,IRANGE(51),NTOTL,RANGES(52),        HSTOGM
000053               *,               MINVAL,MAXVAL,DATPRD(3),NZRNG,REF          HSTOGM
000054          C                                                                HSTOGM
000055                REAL    MINVAL,MAXVAL                                      HSTOGM
000056                INTEGER DATPRD                                             HSTOGM
000057          C                                                                CAMSER
000058                COMMON /SELCTN/ BIOWD(4),WPRTY(4),IPRD(3,14)               SELCTN
```

```
000059      C                                                                       SELCTN
000060            INTEGER BIOWD,WPATY                                               SELCTN
000061      C                                                                       CAMSER
000062            COMMON /PRTCTL/ NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)               PRTCTL
000063            INTEGER HEADER                                                    PRTCTL
000064            COMMON /MATCH1/ MATCH                                                    *NEW
000065      C                                                                       CAMSER
000066            COMMON /READNG/ READNG(5600)                                      MODF
000067      C                                                                       MODF
000068            DIMENSION CLEWA(100),CLEPRO(100),CLATEC(100),CLPTEC(100)          MODF
000069           *,         CLATWC(100),CLPTWC(100)                                MODF
000070      C                                                                       MODF
000071            EQUIVALENCE                          (READNG(5001),CLEWA(1))    MODF
000072           *,  (READNG(5101),CLEPRO(1)),         (READNG(5201),CLATEC(1))   MODF
000073           *,  (READNG(5301),CLPTEC(1)),         (READNG(5401),CLATWC(1))   MODF
000074           *,  (READNG(5501),CLPTWC(1))                                       MODF
000075      C                                                                       CAMSER
000076      C *** LOCAL STORAGE                                                     CAMSER
000077      C                                                                       CAMSER
000078            INTEGER SBSTRA                                                    MODF
000079            DATA AREAPS /10289,712/                                           CAMSER
000080            JWIN = 0                                                          CAMSER
000081            JPRO = 0                                                          CAMSER
000082      C                                                                       CAMSER
000083      C *** PROCESS FOR BIO-WINDOWS FIRST                                     CAMSER
000084      C                                                                       CAMSER
000085         10 IH = 1                                                          MODF1
000086            IF (JWIN.LT.0) GO TO 30                                         MODF1
000087      C                                                                       CAMSER
000088         15 JWIN = JWIN+1                                                   CAMSER
000089      C                                                                       CAMSER
000090            IF (JWIN.GT.4) GO TO 20                                         CAMSER
000091            IF (BIOWD(JWIN).NE.0) GO TO 50                                  CAMSER
000092            GO TO 15                                                        CAMSER
000093      C                                                                       CAMSER
000094         20 JWIN = -1                                                       CAMSER
000095         30 JPRO = JPRO+1                                                   CAMSER
000096      C                                                                       CAMSER
000097            IF (JPRO.GT.14) GO TO 500                                       CAMSER
000098            IF (IPRO(1,JPRO)).EQ.0) GO TO 500                               CAMSER
000099      C                                                                       CAMSER
000100      C                                                                       CAMSER
000101         50 DO 60 I = 1,3200                                                CAMSER
000102         60 READNG(I) = 0.0                                                 CAMSER
000103      C                                                                       CAMSER
000104            J = 0                                                           CAMSER
000105            KXM = 0                                                         CAMSER
000106            NTOTL = 0                                                       CAMSER
000107      C                                                                       CAMSER
000108            DO 70 I = 1,51                                                  CAMSER
000109         70 IRANGE(I) = 0                                                   CAMSER
000110      C                                                                       CAMSER
000111            DELSO  = 0.0                                                    MODF1
000112            DELERR = 0.0                                                    MODF1
000113            REF    = 0.0                                                    MODF1
000114            MINVAL = 1.E10                                                  MODF1
000115            MAXVAL = -1.E10                                                 MODF1
000116            DELSI  = 0.0                                                    MODF1
000117            IF (IH.EQ.0) GO TO 80                                           MODF1
000118      C                                                                       CAMSER
```

```
000119          C *** GET HEADER RECORD                                                 CAMSER
000120          C                                                                       CAMSER
000121                CALL FILSEQ(IH)                                                   CAMSER
000122          C                                                                       CAMSER
000123                IF (IERR.NE.0) GO TO 500                                          CAMSER
000124          C                                                                       CAMSER
000125                ICASEN = ICASE                                                    CAMSER
000126                FILNAM(1) = FILENM(1)                                             CAMSER
000127                FILNAM(2) = FILENM(2)                                             CAMSER
000128                NALEMT = 0                                                        CAMSER
000129          C                                                                       CAMSER
000130                IH = 0                                                            CAMSER
000131          C                                                                       CAMSER
000132          C *** GET DETAIL RECORDS                                                CAMSER
000133          C                                                                       CAMSER
000134           80 CALL FILSEQ(IH)                                                     CAMSER
000135          C                                                                       CAMSER
000136                IF (IH.LT.0) GO TO 200                                            MODF1
000137          C                                                                       CAMSER
000138                IF (KXN.GT.0) GO TO 100                                           MODF
000139          C                                                                       CAMSER
000140                IF(MATCH .EQ. -2)GO TO 210                                              *NEW
000141                CNTRY = ICNTRY                                                    CAMSER
000142                REGION = IREGN                                                    CAMSER
000143                ZONE   = IZONE                                                    CAMSER
000144                STRATA = ISTRTA                                                   CAMSER
000145                SBSTRA = ISBSTA                                                   CAMSER
000146          C                                                                       CAMSER
000147                GO TO 120                                                         CAMSER
000148          C                                                                       CAMSER
000149          C TEST FOR CHANGE IN ID AREA                                            CAMSER
000150          C                                                                       CAMSER
000151           100 IF (ISBSTA.NE.SBSTRA.OR.ISTRTA.NE.STRATA) GO TO 200                MODF1
000152               GO TO (105,110,120), LEVEL                                         CAMSER
000153           105 IF (IZONE.NE.ZONE) GO TO 200                                       MODF2
000154           110 IF (IREGN.NE.REGION) GO TO 200                                     MODF2
000155          C                                                                       CAMSER
000156           120 IF (JWIN.LT.0) GO TO 150                                           CAMSER
000157          C                                                                       CAMSER
000158                IF (PRDATE(JWIN).EQ.0) GO TO 80                                   MODF1
000159                IDATE = PRDATE(JWIN)                                              CAMSER
000160                DELSB = DELSB+(XEPW(JWIN)-XTPWI)                                  CAMSER
000161                KXN = KXN+1                                                       CAMSER
000162          C                                                                       CAMSER
000163                GO TO 80                                                          CAMSER
000164          C                                                                       CAMSER
000165          C *** USE PREDICTION DATE INPUT                                         CAMSER
000166          C                                                                       CAMSER
000167           150 FLDA = IPRD(3,JPRD)                                                CAMSER
000168               LMO  = IPRD(2,JPRD)                                                CAMSER
000169               LYR  = IPRD(1,JPRD)                                                CAMSER
000170          C                                                                       CAMSER
000171                CALL LEPA(FLDA,LMO,LYR,ALFGM,DAYS)                                CAMSER
000172          C                                                                       CAMSER
000173                IDAYS = DAYS                                                      CAMSER
000174                DO 160 I = 1,4                                                    CAMSER
000175                IX = IPRIY(I)                                                     CAMSER
000176                IF (IX.EQ.0) GO TO 160                                            MODF1
000177                IF(PRDATE(IX) .EQ. 0)GO TO 160                                          *NEW
000178                IF(IDAYS .GE. PRDATE(IX))GO TO 180
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000179          160 CONTINUE                                                    CAMSER
000180              GO TO 80                                                    MODF1
000181          C                                                               CAMSER
000182          180 IDATL = IDAYS                                               CAMSER
000183              DELSB = DELSB+(XEP*(IX)-XTPWI)                              CAMSER
000184              KXM = KXM+1                                                 CAMSER
000185          C                                                               CAMSER
000186              GO TO 80                                                    CAMSER
000187          C                                                               CAMSER
000188          C *** CHANGE IN I.D. AREA                                       CAMSER
000189          C                                                               CAMSER
000190          200 CONTINUE                                                    MODF
000191              IF (KXM.EQ.0) GO TO 210                                     MODF
000192              XNA = NA                                                    MODF
000193              XN  = N                                                     CAMSER
000194              XR  = AREA/(XNA*AREAPS)                                     CAMSER
000195              XNR = XN*XR*AREAPS                                          CAMSER
000196              J = J+1                                                     CAMSER
000197              XM = KXM                                                    CAMSER
000198          C                                                               CAMSER
000199              READNG(J)  = DELSB/XM*XNR                                   CAMSER
000200          C                                                               CAMSER
000201              MINVAL = AMIN1(MINVAL,READNG(J))                            CAMSER
000202              MAXVAL = AMAX1(MAXVAL,READNG(J))                            CAMSER
000203          C                                                               CAMSER
000204              DELERR = DELERR+READNG(J)                                   CAMSER
000205              REF    = REF+XTPW*XNR                                       CAMSER
000206              DELSQ  = DELSQ+READNG(J)**2                                 CAMSER
000207          C                                                               CAMSER
000208          210 CONTINUE                                                    MODF
000209              DELSB = 0                                                   CAMSER
000210              KXM   = 0                                                   CAMSER
000211          C                                                               CAMSER
000212              IF (IH.LT.0) GO TO 250                                      MODF1
000213              GO TO (220,230,240), LEVEL                                  CAMSER
000214          C                                                               CAMSER
000215          220 IF (IZONE.NE.ZONE) GO TO 250                                MODF2
000216          C                                                               CAMSER
000217          230 IF (IREGN.EQ.REGION) GO TO 80                               CAMSER
000218              GO TO 250                                                   CAMSER
000219          C                                                               CAMSER
000220          240 IF (IH.EQ.0) GO TO 80                                       CAMSER
000221          C                                                               CAMSER
000222          250 IF (J.EQ.0) GO TO 400                                       MODF1
000223          C                                                               MODF1
000224              NTOTL = J                                                   MODF1
000225          C                                                               CAMSER
000226              DO 300 I = 1,NTOTL                                          CAMSER
000227              PCDERR = 0.0                                                MODF1
000228              IF (REF.NE.0.0) PCDERR = 100.*READNG(I)/REF                 MODF1
000229              DO 280  K = 1,NRANGE                                        CAMSER
000230              IF (PCDERR.NE.0.) GO TO 260                                 CAMSER
000231          C                                                               CAMSER
000232              IRANGE(NZRNG) = IRANGE(NZRNG)+1                             CAMSER
000233              GO TO 300                                                   CAMSER
000234          C                                                               CAMSER
000235          260 IF (PCDERR.GE.RANGES(K+1)) GO TO 280                        CAMSER
000236          C                                                               CAMSER
000237              IRANGE(K) = IRANGE(K)+1                                     CAMSER
000238              GO TO 300                                                   CAMSER
```

```
000239          C                                                               CAMSER
000240            280 CONTINUE                                                  CAMSER
000241            300 CONTINUE                                                  CAMSER
000242          C                                                               CAMSER
000243              DATPRD(1) = JWIN                                            MODF2
000244              IF (JWIN.GT.0) GO TO 310                                    MODF2
000245              CALL FZULU(IDATE,DATPRD)                                    CAMSER
000246            310 CONTINUE                                                  MODF2
000247              TJ = NTOTL                                                  CAMSER
000248              XMVAL = 0.0                                                 MODF1
000249              IF (REF.NE.0.0) XMVAL = 100./(TJ*REF)*DELERR                MODF1
000250              STDDEV = 0.0                                                MODF
000251              IF(DELERR**2/TJ .GT. DELSQ)GO TO 320
000252              IF (J.LT.2.OR.REF.EQ.0.0) GO TO 320                         MODF1
000253              STDDEV = 100./REF*SQRT(1./(TJ-1.)*(DELSQ-1./TJ*DELERR**2))  CAMSER
000254            320 CONTINUE                                                  MODF
000255              XMPSUM = TJ*XMVAL                                           CAMSER
000256              SDPSUM = SQRT(TJ)*STDDEV                                    CAMSER
000257          C                                                               CAMSER
000258              CALL REPRT2                                                 CAMSER
000259          C                                                               CAMSER
000260              IF (IH.EQ.0) GO TO 50                                       MODF1
000261              GO TO 10                                                    CAMSER
000262          C                                                               CAMSER
000263            400 CONTINUE                                                  MODF1
000264              IF (JWIN.LT.0) GO TO 420                                    MODF1
000265              CALL PAGHDR(-2)                                             MODF1
000266              WRITE (KOUT,6310) RPTYPE,JPARM,PARMTR(JPARM),JWIN           MODF1
000267              IF (IH.EQ.0) GO TO 50                                       MODF1
000268              GO TO 10                                                    MODF1
000269          C                                                               MODF1
000270            420 CALL PAGHDR(-2)                                           MODF1
000271              WRITE (KOUT,6300) RPTYPE,JPARM,PARMTR(JPARM),(IPRD(I,JPRD),I=1,3)MODF1
000272              IF (IH.EQ.0) GO TO 50                                       MODF1
000273              GO TO 10                                                    MODF1
000274            500 RETURN                                                    CAMSER
000275          C                                                               CAMSER
000276          C                                                               CAMSER
000277          C *** ERROR MESSAGE                                             CAMSER
000278          C                                                               CAMSER
000279           6300 FORMAT (54H *** INPUT PREDICTION DATE DOES NOT MATCH DATE ON FILE/CAMSER
000280          *         12H     RPTYPE=,I2,8H PARMTR(,I1,2H)=,I1,12H INPUT DATE=, CAMSER
000281          *           3I2)                                                CAMSER
000282           6310 FORMAT (43H *** NO INFORMATION FOR BIO-WINDOW SELECTED/    MODF1
000283          *         12H     RPTYPE=,I2,8H PARMTR(,I1,2H)=,I1,12H BIO-WINDOW=, MODF1
000284          *           I1)                                                 MODF1
000285              END                                                         CAMSER
```

```
* ELT CLMOVE,1,760511, 59143    , 1
000001            SUBROUTINE CLMOVE(NCL)                                        CLMOVE
000002      C                                                                   CLMOVE
000003      C *** THIS ROUTINE CALLED BY MCHIST MOVES THE CONFIDENCE LEVELS,    CLMOVE
000004      C *** ONE AT A TIME, FROM THE READNG BUFFER TO /FILBUF/             CLMOVE
000005      C                                                                   CLMOVE
000006      C                                                                   CLMOVE
000007      C *** COMMON STORAGE                                                CLMOVE
000008      C                                                                   CLMOVE
000009            COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA           FILBUF
000010           *,               ISBSTA,ICASE,MXSEG                           MODF
000011           *,               KBUF(125)                                    MODF
000012      C                                                                   FILBUF
000013            DIMENSION KWIN(4),IBIORD(4),IPRDAT(14),PRDATE(6),XEPW(4),XEYLD(6) FILBUF
000014           *,          APVAL(2)                                           FILBUF
000015           *,          MCERR(100)                                         MODF
000016      C                                                                   FILBUF
000017            REAL    MCREF,MCERR                                           MODF
000018            INTEGER FILENM,PRDATE                                         FILBUF
000019      C                                                                   FILBUF
000020            EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                        FILBUF
000021           *,  (KBUF(2),IGPNO,LSTRAT,NREGS)                               MODF
000022           *,  (KBUF(3),XHPW,WZTOT)                                       MODF
000023           *,  (KBUF(4),AREA,IBW)                                         MODF
000024           *,  (KBUF(5),XTPW,NPD)                                         MODF
000025           *,  (KBUF(6),N,IBIORD(1))                                      MODF
000026           *,  (KBUF(7),NA)                                               MODF
000027           *,  (KBUF(8),XHTPW)                                            MODF
000028           *,  (KBUF(9),XCV2)                                             MODF
000029           *,  (KBUF(10),XTPWI,IPRDAT(1))                                 MODF
000030           *,  (KBUF(11),PRDATE(1))                                       MODF
000031           *,  (KBUF(17),XLPW(1))                                         MODF
000032           *,  (KBUF(21),XEYLD(1))                                        MODF
000033           *,  (KBUF(24),APVAL(1),MCREF)                                  MODF
000034           *,  (KBUF(25),MCERR(1))                                        MODF
000035           *,  (KBUF(27),XTYLD)                                           MODF
000036      C                                                                   CLMOVE
000037            COMMON /READNG/ READNG(5600)                                  MODF
000038      C                                                                   MODF
000039            DIMENSION CLENA(100),CLEPRD(100),CLATEC(100),CLPTEC(100)      MODF
000040           *,         CLATWC(100),CLPTWC(100)                             MODF
000041      C                                                                   MODF
000042            EQUIVALENCE                          (READNG(5001),CLENA(1))  MODF
000043           *,  (READNG(5101),CLEPRD(1)),         (READNG(5201),CLATEC(1)) MODF
000044           *,  (READNG(5301),CLPTEC(1)),         (READNG(5401),CLATWC(1)) MODF
000045           *,  (READNG(5501),CLPTWC(1))                                   MODF
000046      C                                                                   CLMOVE
000047      C                                                                   CLMOVE
000048      C *** COMPUTE THE STARTING INDEX IN BUFFER                          CLMOVE
000049      C                                                                   CLMOVE
000050            M = 4900+100*NCL                                              CLMOVE
000051      C                                                                   CLMOVE
000052      C *** SET REFERENCE VALUE                                           CLMOVE
000053      C                                                                   CLMOVE
000054            MCREF = 100.                                                  CLMOVE
000055      C                                                                   CLMOVE
000056      C *** MOVE M-C READING VALUES                                       CLMOVE
000057      C                                                                   CLMOVE
000058            DO 10 IMX = 1,100                                             CLMOVE
```

```
000059              ICL = M+INX                                     CLMOVE
000060            10 MCERR(INX) = READNG(ICL)                       CLMOVE
000061         C                                                    CLMOVE
000062              RETURN                                          CLMOVE
000063         C                                                    CLMOVE
000064              END                                             CLMOVE
```

```
* ELT FILDA,1,760511, 59152     , 1
000001                SUBROUTINE FILDA (IH,IPNT,NREC)                                  FILDA
000002          C       CALL THE RANDOM ACCESS ROUTINE RANACF TO READ THE CAS CUMULATIVEFILDA
000003          C       FILE (CASE) OR THE CAS DISTRIBUTION FILE (CASDIS)                FILDA
000004          C                                                                        FILDA
000005          C         COMMON BLOCKS                                                  FILDA
000006                COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                           CARDIN
000007               *,              ICASIN(5),LEVEL,JPARM,IERR                        CARDIN
000008                INTEGER RPTYPE,AUNITS,PARMTR                                     CARDIN
000009                COMMON /FBLKBX/ FBLKBX(28,18)                                    FBLKBX
000010                COMMON /FCASDS/ FCASDS(303)                                      FCASDS
000011          C                                                                      FCASDS
000012                DIMENSION BRNBR6(4),PDNBR6(14),AERR(100),PRERR(100),YERR(100)    FCASDS
000013               *,         CLEN6(100),CLEPR6(100),CLATC6(100),CLPTC6(100)       FCASDS
000014               *,         CLAT26(100),CLPT26(100)                              FCASDS
000015          C                                                                      FCASDS
000016                INTEGER BRNBR6,PDNBR6,COUN6                                      FCASDS
000017          C                                                                      FCASDS
000018                EQUIVALENCE                                                      FCASDS
000019               *  (FCASDS(1),NAME6,AEREF),          (FCASDS(2),ICASE6,PEREF)    MODF1
000020               *, (FCASDS(3),COUN6)                                             FCASDS
000021               *, (FCASDS(4),NT6,AERR(1),CLEN6(1),CLPTC6(1))                    FCASDS
000022               *, (FCASDS(5),NREGS6),            (FCASDS(6),NZTOT6)             FCASDS
000023               *, (FCASDS(8),NBA6),              (FCASDS(9),NPD6)               FCASDS
000024               *, (FCASDS(10),BRNBR6(1)),        (FCASDS(14),PDNBR6(1))         FCASDS
000025               *, (FCASDS(104),PRERR(1),CLEPR6(1),CLAT26(1))                    FCASDS
000026               *, (FCASDS(204),YERR(1),CLATC6(1),CLPT26(1))                     FCASDS
000027                COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA             FILBUF
000028               *,               ISUBTA,ICASE,MAXSEG                            MODF
000029               *,               KBUF(125)                                      MODF
000030          C                                                                      FILBUF
000031                DIMENSION KWIN(4),IBIOWD(4),IPRDAT(14),PRDATE(6),XEPW(4),XEYLD(6) FILBUF
000032               *,          AFVAL(2)                                            FILBUF
000033               *,          MCERR(100)                                          MODF
000034          C                                                                      FILBUF
000035                REAL    MCREF,MCERR                                            MODF
000036                INTEGER FILENM,PRDATE                                          FILBUF
000037          C                                                                      FILBUF
000038                EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                          FILBUF
000039               *,  (KBUF(2),IGPNO,NSTRAT,NREGS)                                MODF
000040               *,  (KBUF(3),XPPW,NZTOT)                                        MODF
000041               *,  (KBUF(4),AREA,NBA)                                          MODF
000042               *,  (KBUF(5),XTPW,NPD)                                          MODF
000043               *,  (KBUF(6),N,IBIOWD(1))                                       MODF
000044               *,  (KBUF(7),NA)                                                MODF
000045               *,  (KBUF(8),XBTPW)                                             MODF
000046               *,  (KBUF(9),XCV2)                                              MODF
000047               *,  (KBUF(10),XTPAI,IPRDAT(1))                                  MODF
000048               *,  (KBUF(11),PRDATE(1))                                        MODF
000049               *,  (KBUF(17),XEPW(1))                                          MODF
000050               *,  (KBUF(21),XEYLD(1))                                         MODF
000051               *,  (KBUF(24),AFVAL(1),MCREF)                                   MODF
000052               *,  (KBUF(25),MCERR(1))                                         MODF
000053               *,  (KBUF(27),XTYLD)                                            MODF
000054                COMMON  /IXRCD/ IXRCD(500)                                     FILDA
000055                COMMON /PRTCTL/ IPAGE,NLINES,MAXLIN,NOUT,HEADER(16)            PRTCTL
000056                INTEGER HEADER                                                 PRTCTL
000057                COMMON /LARGE/ LARGE(5000)                                     MODF
000058          C                                                                      MODF
```

```
      DIMENSION CLEWA(100),CLEPRD(100),CLATEC(100),CLPTEC(100)          MODF
     *,         CLATWC(100),CLPTWC(100)                                 MODF
C                                                                       MODF
      EQUIVALENCE                          (READNG(5001),CLEWA(1))      MODF
     *,  (READNG(5101),CLEPRD(1)),         (READNG(5201),CLATEC(1))     MODF
     *,  (READNG(5301),CLPTEC(1)),         (READNG(5401),CLATWC(1))     MODF
     *,  (READNG(5501),CLPTWC(1))                                       MODF
C                                                                       FILDA
      DIMENSION  IBLKBX(504)                                            FILDA
      EQUIVALENCE  ( IBLKBX,FBLKBX )                                    FILDA
C                                                                       FILDA
C        INPUT ...                                                      FILDA
C          LEVEL  = 1 FOR ZONE LEVEL                                    FILDA
C                 = 2 FOR REGION LEVEL,                                 FILDA
C                 = 3 FOR COUNTRY LEVEL.                                FILDA
C                                                                       FILDA
C          RPTYPE = REPORT TYPE                                         FILDA
C                   (=2 FOR POPULATION ERRORS)                          FILDA
C                   (=3 FOR MONTE CARLO ERRORS OR CONFIDENCE LEVELS)    FILDA
C          JPARM  = INDEX WHICH DETERMINES PARAMETER TO BE REPORTED...  FILDA
C            IF RPTYPE = 2 AND                                          FILDA
C              IF JPARM = 4, THEN POPULATION AREA ERROR IS REPORTED.    FILDA
C              IF JPARM = 5, THEN POPUL. PRODUCTION ERROR IS REPORTED.  FILDA
C                                                                       FILDA
C            IF RPTYPE = 3 AND                                          FILDA
C              IF JPARM = 1, THEN AREA DATA IS EXTRACTED FROM CASDIS.   FILDA
C              IF JPARM = 2,THEN PRODUCTION DATA IS EXTRACTED FROM CASDISFILDA
C              IF JPARM = 3, THEN YIELD DATA IS EXRACTED FROM CASDIS    FILDA
C              IF JPARM = 4, THEN CONFIDENCE LEVELS ARE EXTRACTED FROM  FILDA
C                            CASDIS                                     FILDA
C                                                                       FILDA
C                                                                       FILDA
C        LINKAGE ...                                                    FILDA
C          CALL FILDA (IH,IPNT,NREC)                                    FILDA
C            WHERE                                                      FILDA
C              IH   = 1 TO READ HEADER RECORD,                          FILDA
C                   = 0 TO READ DATA (DETAIL) RECORD.                   FILDA
C              IPNT = PREDICTION POINT NUMBER                           FILDA
C              NREC = RECORD NUMBER TO READ                             FILDA
C                                                                       FILDA
C        LOCAL VARIABLES ...                                            FILDA
C          IFILE = FILE NUMBER                                          FILDA
C          NWRDS = NUMBER OF WORDS TO READ                              FILDA
C          LIX   = LENGTH OF INDEX RECORD (MAXIMUM RECORD NUMBER)       FILDA
C                                                                       FILDA
      INTEGER  CASF, CASDIS                                             FILDA
C                                                                       FILDA
      DATA  CASF  , LCASF , CASDIS, LCASD , LIXCAS, LIXDIS              FILDA
     1    /  14   ,  504  ,    4  ,  303  ,   588 ,  506  /            FILDA
C       NOTE ... LIXDIS = 506 ALLOWS UP TO 8 PREDICTION POINTS          FILDA
C       (INCLUDING BIG-INDEXS)                                          FILDA
C       ( 1463 DATA RECORDS + 1 HEADER RECORD + 1 INDEX RECORD )        FILDA
C                                                                       FILDA
C                                                                       FILDA
      IREC = NREC                                                       FILDA
      IPP  = IPNT                                                       FILDA
C                                                                       FILDA
C       TEST REPORT TYPE AND SET FILE NUMBER, FILE LENGTH, AND FILE     FILDA
C       NAME TO READ CAS CUM FILE IF RPTYPE = 2 OR TO READ CAS DISTR.   FILDA
C       FILE IF RPTYPE = 3                                              FILDA
```

```
000119          C                                                               FILDA
000120                IF ( RPTYPE .EQ. 2 )  GO TO 120                           FILDA
000121                IF ( RPTYPE .EQ. 3 )  GO TO 130                           FILDA
000122          C        ANY VALUE OF RPTYPE EXCEPT 2 OR 3 IS AN ERROR            FILDA
000123           110  CALL PAGHDR (-2)                                          FILDA
000124                WRITE (KOUT,1)  RPTYPE,JPARM,LEVEL                        FILDA
000125           1    FORMAT (//38H0***** ERROR IN SUBROUTINE FILDA *****/      FILDA
000126               1  28H   ILLEGAL VALUE OF RPTYPE =,I4,12H  OR JPARM =,I4,  FILDA
000127               2  10H.  LEVEL =I4)                                        FILDA
000128                STOP                                                      FILDA
000129          C                                                               FILDA
000130          C        RPTYPE = 2                                             FILDA
000131           120  IF ( JPARM .LT. 4 )  GO TO 110                           FILDA
000132                IFILE = CASF                                              FILDA
000133                NWRDS = LCASF                                             FILDA
000134                FILENM(1)= 6HCASCUM                                       FILDA
000135                LIX   = LIXCAS                                            FILDA
000136                IF(IH .NE. 0)DEFINE FILE 14(388,504,U,IDUM)
000137                GO TO 150                                                 FILDA
000138          C                                                               FILDA
000139          C        RPTYPE = 3                                             FILDA
000140           130  IFILE = CASDIS                                           FILDA
000141                NWRDS = LCASD                                             FILDA
000142                FILENM(1)= 6HCASDIS                                       FILDA
000143                LIX   = LIXDIS                                            FILDA
000144                IF(IH .NE. 0)DEFINE FILE 4(506,303,U,IDUM)
000145          C                                                               FILDA
000146           150  FILENM(2)= 1H                                            FILDA
000147          C                                                               FILDA
000148                IF ( IH .EQ. 0 )  GO TO 200                               FILDA
000149          C                                                               FILDA
000150          C        IH = 1.  OPEN RANDOM ACCESS FILE AND READ HEADER RECORD FILDA
000151                CONTINUE
000152                CALL RANACF (IFILE,1,FBLKBX,NWRDS,IXRCD,LIX,1)            FILDA
000153          C        CHECK FILE NAME                                        FILDA
000154                IF ( IBLKBX(1) .EQ. FILENM(1) )  GO TO 170                FILDA
000155          C                                                               FILDA
000156          C        ERROR. FILE NAME DOES NOT MATCH                        FILDA
000157                CALL PAGHDR (-2)                                          FILDA
000158                WRITE (KOUT,2)  FILENM(1),FBLKBX(1,1)                     FILDA
000159           2    FORMAT (38H0***** ERROR IN SUBROUTINE FILDA *****/        FILDA
000160               1  16H   FILE NAME OF ,A6,10H  FILE  = ,A6,31H--DOES NOT MATCH EXPEFILDA
000161               2CTED VALUE )                                              FILDA
000162                STOP                                                      FILDA
000163          C                                                               FILDA
000164          C        EXTRACT INFORMATION FROM HEADER RECORD                 FILDA
000165           170  ICNTRY= IBLKBX(3)                                        FILDA
000166                ICASE = IBLKBX(2)                                         FILDA
000167                NT    = IBLKBX(4)                                         FILDA
000168                NREGS = IBLKBX(5)                                         FILDA
000169                NZIOT = IBLKBX(6)                                         FILDA
000170                IF(RPTYPE .NE. 3)NSTRAT = IBLKBX(7)                       MOD3
000171                NBK   = IBLKBX(8)                                         FILDA
000172                NPD   = IBLKBX(9)                                         FILDA
000173          C                                                               FILDA
000174                DO 180 I=1,4                                              FILDA
000175                IBIOWD(I)= IBLKBX(I+9)                                    FILDA
000176           180  CONTINUE                                                 FILDA
000177                IF ( NPD .EQ. 0 )  GO TO 200                              FILDA
000178                DO 190 I=1,NPD                                            FILDA
```

```
000179            IPRDAT(I)= IBLKBX(I+13)                                    FILDA
000180     190    CONTINUE                                                   FILDA
000181          C                                                            FILDA
000182            GO TO 900                                                  FILDA
000183          C                                                            FILDA
000184          C     READ DATA (DETAIL) RECORD AND EXTRACT DESIRED DATA     FILDA
000185          C                                                            FILDA
000186     200  IF ( RPTYPE .EQ. 3 )  GO TO 300                             FILDA
000187          C                                                            FILDA
000188          C     RPTYPE = 2.. READ DETAIL RECORD FROM CAS CUM FILE      FILDA
000189            CALL RANACF (IFILE,IREC,FBLKBX,NWRDS,IXRCD,LIX,1)          FILDA
000190          C                                                            FILDA
000191            IREGN= IBLKBX(1)                                           FILDA
000192            IZONE= IBLKBX(2)                                           FILDA
000193            ISTRTA= IBLKBX(3)                                          FILDA
000194            IF ( JPARM .EQ. 5 )  GO TO 240                             FILDA
000195          C                                                            FILDA
000196          C     JPARM = 4. STORE TRUE WA AND AREA ERROR                FILDA
000197            APVAL(1)= FBLKBX(8,IPP)                                    FILDA
000198            APVAL(2)= FBLKBX(6,IPP)                                    FILDA
000199            GO TO 900                                                  FILDA
000200          C                                                            FILDA
000201          C     STORE TRUE PRODUCTION AND PRODUCTION ERROR             FILDA
000202     240  APVAL(1)= FBLKBX(12,IPP)                                    FILDA
000203            APVAL(2)= FBLKBX(10,IPP)                                   FILDA
000204            GO TO 900                                                  FILDA
000205          C                                                            FILDA
000206          C                                                            FILDA
000207          C     RPTYPE = 3.  READ DETAIL RECORD FROM CAS DISTR. FILE   FILDA
000208     300  IF ( JPARM .EQ. 4 )  IREC= IREC + 1                         FILDA
000209            CALL RANACF (IFILE,IREC,FCASDS,NWRDS,IXRCD,LIX,1)          FILDA
000210          C                                                            FILDA
000211            GO TO (310,320,330,400), JPARM                             FILDA
000212          C                                                            FILDA
000213          C     JPARM = 1                                              FILDA
000214     310  II= 4                                                       FILDA
000215            GO TO 340                                                  FILDA
000216          C     JPARM = 2                                              FILDA
000217     320  II= 104                                                     FILDA
000218            GO TO 340                                                  FILDA
000219          C     JPARM = 3                                              FILDA
000220     330  II=204                                                      FILDA
000221          C                                                            FILDA
000222     340  MCREF = FCASDS(JPARM)                                       FILDA
000223            DO 350 I=1,100                                             FILDA
000224            MCERR(I)= FCASDS(II)                                       FILDA
000225     350  II= II + 1                                                  FILDA
000226            GO TO 900                                                  FILDA
000227          C                                                            FILDA
000228          C     JPARM = 4                                              FILDA
000229     400  DO 410 I=1,100                                             FILDA
000230            CLERA(I)= FCASDS(I+3)                                      FILDA
000231            CLERRD(I)= FCASDS(I+103)                                   FILDA
000232     410  CLATEC(I)= FCASDS(I+203)                                    FILDA
000233          C                                                            FILDA
000234          C     READ NEXT GROUP OF CONFIDENCE LEVELS                   FILDA
000235            IREC= IREC + 1                                             FILDA
000236            CALL RANACF (IFILE,IREC,FCASDS,NWRDS,IXRCD,LIX,1)          FILDA
000237            DO 450 I=1,100                                             FILDA
000238            CLPTEC(I)= FCASDS(I+3)                                     FILDA
```

```
000239                                                    FILDA
000240                                                    FILDA
000241                                                    FILDA
000242                                                    FILDA
000243                                                    FILDA
                  CLATWC(I)= FCASDS(I+103)
                  CLPTWC(I)= FCASDS(I+203)
              430
              C
              900 RETURN
                  END
```

```
* ELT FILSEQ,1,760513, 61014    , 1

000001            SUBROUTINE FILSEQ(IH)                                          FILSEQ
000002          C                                                                FILSEQ
000003          C THIS ROUTINE READS THE REQUIRED SEQUENTIAL FILE(S) DETERMINED BY THE  FILSEQ
000004          C     RPTYPE FLAG AND RETURNS THE REQUIRED DATA IN COMMON BLOCK FILBUF  FILSEQ
000005          C                                                                FILSEQ
000006          C     INPUT -  IH = 1  READS HEADER RECORD                       FILSEQ
000007          C              = 0  READS DETAIL RECORDS                         FILSEQ
000008          C                                                                FILSEQ
000009          C     OUTPUT-  IH = -1 END-OF-FILE ENCOUNTERED ON FILE(S)        FILSEQ
000010          C                                                                FILSEQ
000011          C *** COMMON STORAGE                                             FILSEQ
000012          C                                                                FILSEQ
000013            COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                         CARDIN
000014           *,              ICASIN(5),LEVEL,JPARM,IERR                      CARDIN
000015            INTEGER RPTYPE,AUNITS,PARMTR                                   CARDIN
000016          C                                                                FILSEQ
000017            COMMON /PRTCTL/ NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)            PRTCTL
000018            INTEGER HEADER                                                 PRTCTL
000019          C                                                                FILSEQ
000020            COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA            FILBUF
000021           *,              ISBSTA,ICASE,MXSEG                              MODF
000022           *,              KBUF(125)                                       MODF
000023          C                                                                FILBUF
000024            DIMENSION KWIN(4),IRIOWD(4),IPRDAT(14),PRDATE(6),XEPW(4),XEYLD(6) FILBUF
000025           *,           APVAL(2)                                          FILBUF
000026           *,           MCERR(100)                                        MODF
000027          C                                                                FILBUF
000028            REAL    MCREF,MCERR                                            MODF
000029            INTEGER FILENM,PRDATE                                          FILBUF
000030          C                                                                FILBUF
000031            EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                          FILBUF
000032           *,  (KBUF(2),ISPAD,ISTRAT,NREGS)                                MODF
000033           *,  (KBUF(3),XPPW,XZTOT)                                        MODF
000034           *,  (KBUF(4),AREA,NBW)                                          MODF
000035           *,  (KBUF(5),XTPW,NPD)                                          MODF
000036           *,  (KBUF(6),W,IRIOWD(1))                                       MODF
000037           *,  (KBUF(7),HA)                                                MODF
000038           *,  (KBUF(8),XBTPW)                                             MODF
000039           *,  (KBUF(9),XCV2)                                              MODF
000040           *,  (KBUF(10),XTPAT,IPRDAT(1))                                  MODF
000041           *,  (KBUF(11),PRDATE(1))                                        MODF
000042           *,  (KBUF(17),XEPW(1))                                          MODF
000043           *,  (KBUF(21),XEYLD(1))                                         MODF
000044           *,  (KBUF(24),APVAL(1),MCREF)                                   MODF
000045           *,  (KBUF(25),MCERR(1))                                         MODF
000046           *,  (KBUF(27),XTYLD)                                            MODF
000047          C                                                                FILSEQ
000048            COMMON /FHEADR/ NAME(2),ICASEF,IMXSEG                          FHEADR
000049          C                                                                FILSEQ
000050            COMMON /FSBHST/ COUN1,REGN1,ZONE1,STRTA1,SBSTA1,INEG,NDSEG(162) FSBHST
000051            DIMENSION XSEG(162)                                            FSBHST
000052            INTEGER COUN1,REGN1,ZONE1,STRTA1,SBSTA1                        FSBHST
000053            EQUIVALENCE (XSEG(1),NDSEG(1))                                 FSBHST
000054          C                                                                FILSEQ
000055            COMMON /FSGTRU/ COUN2,REGN2,ZONE2,STRTA2,SBSTA2,SEGMT2         FSGTRU
000056           *,              DUMF2(8),TPW2,TPN2                              FSGTRU
000057            COMMON /MATCH1/ MATCH                                                *NEW
000058          C                                                                FSGTRU
```

```
000059          INTEGER COUN2,REGN2,ZONE2,STRTA2,SBSTA2,SEGMT2                  FSGTRU
000060      C                                                                   FILSEQ
000061          COMMON /FCAMSF/ COUN3,REGN3,ZONE3,STRTA3,SBSTA3,SEGMT3          FCAMSF
000062         *,               TPW3,EWINDO(3,4)                                FCAMSF
000063      C                                                                   FCAMSF
000064          DIMENSION IWIN(3,4)                                             FCAMSF
000065      C                                                                   FCAMSF
000066          INTEGER COUN3,REGN3,ZONE3,STRTA3,SBSTA3,SEGMT3                  MODF
000067      C                                                                   FCAMSF
000068          EQUIVALENCE (IWIN(1,1),EWINDO(1,1))                             FCAMSF
000069      C                                                                   FILSEQ
000070          COMMON /FYESOT/ COUN4,REGN4,ZONE4,STRTA4,YSTR,YSPNT(3,6)        FYESOT
000071      C                                                                   FYESOT
000072          DIMENSION IYSPNT(3,6)                                           FYESOT
000073      C                                                                   FYESOT
000074          INTEGER COUN4,REGN4,ZONE4,STRTA4                                FYESOT
000075      C                                                                   FYESOT
000076          EQUIVALENCE (IYSPNT(1,1),YSPNT(1,1))                            FYESOT
000077      C                                                                   FILSEQ
000078          COMMON /FACQUS/ COUN7,REGN7,ZONE7,STRTA7,SEGMT7,IWINDO(25,4),ITOTLFACQUS
000079      C                                                                   FACQUS
000080          INTEGER COUN7,REGN7,ZONE7,STRTA7,SEGMT7                         FACQUS
000081      C                                                                   FILSEQ
000082          COMMON /SUBHS/ICSESH                                            MOD3
000083      C                                                                   FILSEQ
000084      C *** LOCAL STORAGE                                                 FILSEQ
000085      C                                                                   FILSEQ
000086          INTEGER FSUB,FSEG,FCAM,FYES,FACQ                                FILSEQ
000087          DATA FSUB,FSEG,FCAM,FYES,FACQ /3,13,7,10,12/                    FILSEQ
000088          DATA IZZZZ /4HZZZZ/                                             FILSEQ
000089      C                                                                   FILSEQ
000090      C *** BRANCH TO READ HEADER RECORD(S) OR DETAIL RECORD(S)           FILSEQ
000091      C                                                                   FILSEQ
000092          IF (IH.EW.0) GO TO 300                                          FILSEQ
000093      C                                                                   FILSEQ
000094      C *** BRANCH BY RPTYPE TO PROCESS HEADER RECORD(S)                  FILSEQ
000095      C                                                                   FILSEQ
000096          GO TO (100,200,600,120), RPTYPE                                 FILSEQ
000097      C                                                                   FILSEQ
000098      C *** RPTYPE = 1, SUBSTRATA HISTORICAL FILE (SUBHST)                FILSEQ
000099      C                                                                   FILSEQ
000100      C     REWIND SUBHST AND READ HEADER RECORD                          FILSEQ
000101      C                                                                   FILSEQ
000102        100 REWIND FSUB                                                   FILSEQ
000103      C                                                                   FILSEQ
000104          READ (FSUB) NAME(1),NAME(2),ICASEF,IMXSEG                       FILSEQ
000105      C                                                                   FILSEQ
000106      C     SAVE HEADER INFO                                              FILSEQ
000107      C                                                                   FILSEQ
000108          FILENM(1) = NAME(1)                                             FILSEQ
000109          FILENM(2) = NAME(2)                                             FILSEQ
000110          ICASE = ICASEF                                                  FILSEQ
000111          MXSEG = IMXSEG                                                  FILSEQ
000112          I = 1                                                           FILSEQ
000113          NWRDS = MXSEG+12                                                MODF
000114          GO TO 150                                                       FILSEQ
000115      C                                                                   FILSEQ
000116      C *** RPTYPE = 4, DATA ACQUISITION FILE (ACQUIS)                    FILSEQ
000117      C                                                                   FILSEQ
000118        120 REWIND FACQ                                                   FILSEQ
```

```
000119          C                                                                FILSEQ
000120               READ (FACQ) NAME(1),NAME(2),ICASEF                          FILSEQ
000121          C                                                                FILSEQ
000122          C    SAVE HEADER INFO                                            FILSEQ
000123          C                                                                FILSEQ
000124               FILENM(1) = NAME(1)                                         FILSEQ
000125               FILENM(2) = NAME(2)                                         FILSEQ
000126               ICASE = ICASEF                                              FILSEQ
000127               I = 1                                                       FILSEQ
000128          C                                                                FILSEQ
000129          C *** TEST FOR MATCH ON REQUESTED FILE CASE NUMBER                FILSEQ
000130          C                                                                FILSEQ
000131            150 IF (ICASIN(I).EQ.ICASEF) GO TO 500                         FILSEQ
000132          C                                                                FILSEQ
000133          C *** ERROR - CASE NUMBERS DO NOT MATCH                          FILSEQ
000134          C                                                                FILSEQ
000135               CALL PAGHDR(-4)                                             FILSEQ
000136               WRITE (KOUT,6200) I,ICASIN(I),ICASEF,NAME(1),NAME(2)        FILSEQ
000137          C *** STOP                                                       FILSEQ
000138               GO TO 700                                                   FILSEQ
000139          C                                                                FILSEQ
000140          C *** RPTYPE = 2                                                 FILSEQ
000141          C    PARMTR(1) = 1, SEGMENT TRUTH FILE (SEGTRU) + SUBHST (JPARM = 1)  FILSEQ
000142          C    PARMTR(2) = 1, CAMS OUTPUT FILE (CAMSF) + SUBHST    (JPARM = 2)  FILSEQ
000143          C    PARMTR(3) = 1, YES OUTPUT FILE (YESOUT) + SUBHST    (JPARM = 3)  FILSEQ
000144          C                                                                FILSEQ
000145            200 GO TO (210,220,230),JPARM                                  FILSEQ
000146          C                                                                FILSEQ
000147          C    JPARM = 1, SEGTRU + SUBHST                                  FILSEQ
000148          C                                                                FILSEQ
000149            210 REWIND FSEG                                                FILSEQ
000150          C                                                                FILSEQ
000151               READ (FSEG) NAME(1),NAME(2),ICASEF                          FILSEQ
000152          C                                                                FILSEQ
000153               GO TO 250                                                   FILSEQ
000154          C                                                                FILSEQ
000155          C    JPARM = 2, CAMSF + SUBHST                                   FILSEQ
000156          C                                                                FILSEQ
000157            220 REWIND FCAM                                                FILSEQ
000158          C                                                                FILSEQ
000159               READ (FCAM) NAME(1),NAME(2),ICASEF                          FILSEQ
000160          C                                                                FILSEQ
000161               GO TO 250                                                   FILSEQ
000162          C                                                                FILSEQ
000163          C    JPARM = 3, YESOUT + SUBHST                                  FILSEQ
000164          C                                                                FILSEQ
000165            230 REWIND FYES                                                FILSEQ
000166          C                                                                FILSEQ
000167               READ (FYES) NAME(1),NAME(2),ICASEF                          FILSEQ
000168          C                                                                FILSEQ
000169          C    SAVE HEADER INFO                                            FILSEQ
000170          C                                                                FILSEQ
000171            250 FILENM(1) = NAME(1)                                        FILSEQ
000172               FILENM(2) = NAME(2)                                         FILSEQ
000173               ICASE = ICASEF                                              FILSEQ
000174          C                                                                FILSEQ
000175          C    READ SUBHST HEADER RECORD                                   FILSEQ
000176          C                                                                FILSEQ
000177               REWIND FSUB                                                 FILSEQ
000178          C                                                                FILSEQ
```

```
000179            READ (FSUB) NAME(1),NAME(2),ICASEF,IMXSEG                    FILSEQ
000180          C                                                              FILSEQ
000181            MXSEG = IMXSEG                                               FILSEQ
000182            NWRDS = MXSEG+12                                             MODF
000183          C *** TEST FOR MATCH ON REQUESTED FILE CASE NUMBERS             FILSEQ
000184          C                                                              FILSEQ
000185            IF (ICASIN(JPARM).EQ.ICASE) GO TO 270                        FILSEQ
000186          C                                                              FILSEQ
000187          C *** ERROR - NO MATCH                                         FILSEQ
000188          C                                                              FILSEQ
000189            IERR = JPARM                                                 FILSEQ
000190          C                                                              FILSEQ
000191            CALL PAGHDR(-4)                                              FILSEQ
000192            WRITE (KOUT,6200) JPARM,ICASIN(JPARM),ICASE,FILENM(1),FILENM(2)  FILSEQ
000193          C                                                              FILSEQ
000194          C *** TEST FOR MATCH ON SUBHST FILE                            FILSEQ
000195            270 IF(ICSESH .NE. ICASEF)GO TO 280                          MOD3
000196            MATCH = 0                                                    FILSEQ
000197            GO TO 500                                                    FILSEQ
000198          C *** ERROR - NO MATCH                                         FILSEQ
000199          C                                                              FILSEQ
000200            280 IF (IERR.EQ.0) CALL PAGHDR(0)                            FILSEQ
000201          C                                                              FILSEQ
000202            IERR = JPARM                                                 FILSEQ
000203            CALL PAGHDR(4)                                               FILSEQ
000204            WRITE(KOUT,6200)JPARM,ICSESH,ICASEF,NAME(1),NAME(2)          MOD3
000205          C                                                              FILSEQ
000206          C *** RETURN                                                   FILSEQ
000207          C                                                              FILSEQ
000208            GO TO 500                                                    FILSEQ
000209          C                                                              FILSEQ
000210          C *** READ DETAIL RECORD(S)                                    FILSEQ
000211          C                                                              FILSEQ
000212            300 GO TO (310,360,600,330), RPTYPE                          FILSEQ
000213          C                                                              FILSEQ
000214          C *** RPTYPE = 1, SUBHST FILE                                  FILSEQ
000215          C                                                              FILSEQ
000216            310 READ (FSUB) COUN1,REGN1,ZONE1,STRTA1,SBSTA1,INEG         FILSEQ
000217              *,            (NUSEG(I),I=1,NWRDS)                         FILSEQ
000218          C                                                              FILSEQ
000219          C     TEST FOR END-OF-FILE MARKER                              FILSEQ
000220          C                                                              FILSEQ
000221            IF (COUN1.NE.IZZZZ) GO TO 320                                FILSEQ
000222          C                                                              FILSEQ
000223          C *** END-OF-FILE                                              FILSEQ
000224          C                                                              FILSEQ
000225            IP = -1                                                      FILSEQ
000226            GO TO 500                                                    FILSEQ
000227          C                                                              FILSEQ
000228          C *** MOVE DETAIL VALUES TO FILBUF                             FILSEQ
000229          C                                                              FILSEQ
000230            320 ICNTRY = COUN1                                           FILSEQ
000231            IREGN = REGN1                                                FILSEQ
000232            IZONE = ZONE1                                                FILSEQ
000233            ISTRTA = STRTA1                                              FILSEQ
000234          C                                                              FILSEQ
000235            NSEG = INEG                                                  FILSEQ
000236            ISPSU = NUSEG(MXSEG+1)                                       FILSEQ
000237          C       CONVERT MISI PI FROM PERCENT TO FRACTION               MODF2
000238            XMPU = XSEG(MXSEG+2)*0.01                                    MODF2
```

```
C        CONVERT SUBSTRATA LAND AREA FROM KM**2 TO HECTARES        MODF2
      AREA = XSEG(MXSEG+3)*100.0                                   MODF2
C        CONVERT TRUE PW FROM PERCENT TO FRACTION                  MODF2
      XTPW = XSEG(MXSEG+4)*0.01                                    MODF2
      N = NDSEG(MXSEG+5)                                           FILSEQ
      NA = NDSEG(MXSEG+6)                                          FILSEQ
      XHTPW = XSEG(MXSEG+7)                                        FILSEQ
      XCV2 = XSEG(MXSEG+10)                                        FILSEQ
C                                                                  FILSEQ
      GO TO 500                                                    FILSEQ
C                                                                  FILSEQ
C *** RPTYPE = 4, ACQUIS FILE                                      FILSEQ
C                                                                  FILSEQ
  330 READ (FACQ) COUN7,REGN7,ZONE7,STRTA7,SBSTA7,SEGMT7           FILSEQ
     *,          ((IWINDO(J,I),J=1,25),I=1,4),ITOTL                FILSEQ
C                                                                  FILSEQ
C     TEST FOR END-OF-FILE MARKER                                  FILSEQ
C                                                                  FILSEQ
      IF (COUN7.NE.IZZZZ) GO TO 340                                FILSEQ
C                                                                  FILSEQ
C *** END-OF-FILE                                                  FILSEQ
C                                                                  FILSEQ
      IH = -1                                                      FILSEQ
      GO TO 500                                                    FILSEQ
C                                                                  FILSEQ
C *** MOVE DETAIL VALUES TO FILBUF                                 FILSEQ
C                                                                  FILSEQ
  340 ICNTRY = COUN7                                               FILSEQ
      IREGN  = REGN7                                               FILSEQ
      IZONE  = ZONE7                                               FILSEQ
      ISTRTA = STRTA7                                              FILSEQ
      ISBSTA = SBSTA7                                              FILSEQ
C                                                                  FILSEQ
      DO 350 I = 1,4                                               FILSEQ
  350 KBUF(I) = IWINDO(1,I)                                        FILSEQ
C                                                                  FILSEQ
      GO TO 500                                                    FILSEQ
C                                                                  FILSEQ
C *** RPTYPE = 2                                                   FILSEQ
C                                                                  FILSEQ
  360 IF (JPARM.EQ.3) GO TO 440                                    FILSEQ
C                                                                  FILSEQ
C                                                                  FILSEQ
      IF (MATCH.EQ.1) GO TO 390                                    FILSEQ
C                                                                  FILSEQ
      READ (FSUB) COUN1,REGN1,ZONE1,STRTA1,SBSTA1,INEG             FILSEQ
     *,           (NDSEG(I),I=1,NWRDS)                             FILSEQ
C                                                                  FILSEQ
C *** TEST FOR END-OF-FILE MARKER                                  FILSEQ
C                                                                  FILSEQ
      IF (COUN1.NE.IZZZZ) GO TO 370                                FILSEQ
C                                                                  FILSEQ
      IH = -1                                                      FILSEQ
      GO TO 500                                                    FILSEQ
C                                                                  FILSEQ
  370 IF (MATCH.EQ.0) GO TO 380                                    FILSEQ
C                                                                  FILSEQ
      MATCH = 0                                                    FILSEQ
C                                                                  FILSEQ
      IF(IREGN - REGN1)380,800,860                                 MOD2
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000299          800 IF(IZONE - ZONE1)380,810,860                                    MOD2
000300          810 IF(ISTRTA - STRTA1)380,820,860                                  MOD2
000301          820 IF(ISBSTA - SBSTA1)380,830,860                                  MOD2
000302          830 MATCH = 1                                                       MOD2
000303              GO TO 380                                                       MOD2
000304          860 MATCH = -1                                                      MOD2
000305              GO TO 360                                                       MOD2
000306        C                                                                     FILSEQ
000307        C *** MOVE SUBHST VALUES TO FILBUF                                    FILSEQ
000308        C                                                                     FILSEQ
000309          380 ICNTRY = COUN1                                                  FILSEQ
000310              IREGN  = REGN1                                                  FILSEQ
000311              IZONE  = ZONE1                                                  FILSEQ
000312              ISTRTA = STRTA1                                                 FILSEQ
000313              ISBSTA = SBSTA1                                                 FILSEQ
000314        C                                                                     FILSEQ
000315              NSEG   = IWEG                                                   FILSEQ
000316        C        CONVERT SUBSTRATA LAND AREA FROM KM**2 TO HECTARES           MODF2
000317              AREA = XSEG(NXSEG+3)*100.0                                      MODF2
000318        C        CONVERT TRUE PW FROM PERCENT TO FRACTION                     MODF2
000319              XTPW = XSEG(NXSEG+4)*0.01                                       MODF2
000320              N    = NDSLG(NXSEG+5)                                           FILSEQ
000321              NA   = NDSEG(NXSEG+6)                                           FILSEQ
000322        C                                                                     FILSEQ
000323              IF (MATCH.EQ.1) GO TO 500                                       FILSEQ
000324        C                                                                     FILSEQ
000325          390 IF (JPARM.EQ.2) GO TO 420                                       FILSEQ
000326        C                                                                     FILSEQ
000327        C *** JPARM =1, SEGMENT TRUTH FILE (SEGTRU) + SUBHST                  FILSEQ
000328        C                                                                     FILSEQ
000329          400 READ (FSEG) COUN2,REGN2,ZONE2,STRTA2,SBSTA2,SEGMT2              FILSEQ
000330             *,           (DUNE2(I),I=1,8),TPW2,TPM2                        FILSEQ
000331        C                                                                     FILSEQ
000332        C *** TEST FOR EOF                                                    FILSEQ
000333        C                                                                     FILSEQ
000334              IF (COUN2.NE.IZZZZ) GO TO 405                                   FILSEQ
000335        C                                                                     FILSEQ
000336              IH = -1                                                         FILSEQ
000337              GO TO 500                                                       FILSEQ
000338        C                                                                     FILSEQ
000339          405 IF ( IREGN - REGN2 )  409,404,400                               MODF2
000340          404 IF ( IZONE - ZONE2 )  409,406,400                               MODF2
000341          406 IF (ISTRTA - STRTA2)  409,407,400                               MODF2
000342          407 IF (ISBSTA - SBSTA2)  409,408,400                               MODF2
000343          408 MATCH = 1                                                       MODF2
000344              GO TO 410                                                       MODF2
000345          409 CONTINUE                                                        MODF2
000346        C                                                                     FILSEQ
000347              IF(MATCH .EQ. 0)MATCH =-1                                       MOD2
000348              IF(MATCH .EQ. 1)MATCH = -2                                      MOD2
000349        C                                                                     FILSEQ
000350              IREGN = REGN2                                                   FILSEQ
000351              IZONE = ZONE2                                                   FILSEQ
000352              ISTRTA = STRTA2                                                 FILSEQ
000353              ISBSTA = SBSTA2                                                 FILSEQ
000354        C                                                                     FILSEQ
000355        C        CONVERT TRUE PW FROM PERCENT TO FRACTION                     MODF2
000356          410 XTPW = TPW2*0.01                                                MODF2
000357              IF(MATCH .EQ. -1)GO TO 360                                      MOD2
000358        C                                                                     FILSEQ
```

```
000359          GO TO 500                                                       FILSEQ
000360      C                                                                   FILSEQ
000361      C *** JPARM = 2, CAMS OUTPUT FILE (CAMSF) + SUBHST                  FILSEQ
000362      C                                                                   FILSEQ
000363        420 READ (FCAM) COUN3,REGN3,ZONE3,STRTA3,SBSTA3,SEGMT3            FILSEQ
000364           *,           TPW3,((IWIN(J,I),J=1,3),I=1,4)                    FILSEQ
000365      C                                                                   FILSEQ
000366          IF (COUN3.NE.IZZZZ) GO TO 425                                   FILSEQ
000367      C                                                                   FILSEQ
000368          IH = -1                                                         FILSEQ
000369          GO TO 500                                                       FILSEQ
000370      C                                                                   FILSEQ
000371        425 IF ( IREGN - REGN3 )  429,424,420                             MODF2
000372        424 IF ( IZONE - ZONE3 )  429,426,420                             MODF2
000373        426 IF (ISTRTA - STRTA3)  429,427,420                             MODF2
000374        427 IF (ISBSTA - SBSTA3)  429,428,420                             MODF2
000375        428 MATCH = 1                                                     MODF2
000376            GO TO 430                                                     MODF2
000377        429 CONTINUE                                                      MODF2
000378      C                                                                   FILSEQ
000379      C                                                                   FILSEQ
000380          IF (MATCH.EQ.0) MATCH = -1                                      MODF
000381          IF (MATCH.EQ.1) MATCH = -2                                      MODF
000382      C                                                                   FILSEQ
000383          IREGN = REGN3                                                   FILSEQ
000384          IZONE = ZONE3                                                   FILSEQ
000385          ISTRTA = STRTA3                                                 FILSEQ
000386          ISBSTA = SBSTA3                                                 FILSEQ
000387      C                                                                   FILSEQ
000388      C       CONVERT TRUE PW FROM PERCENT TO FRACTION                    MODF2
000389        430 XTPWI = TPW3*0.01                                             MODF2
000390      C                                                                   FILSEQ
000391          DO 435 I = 1,4                                                  FILSEQ
000392          PROBIF(I) = IWIN(1,I)                                           FILSEQ
000393      C       CONVERT ESTIMATED PW FROM PERCENT TO FRACTION                MODF2
000394        435 XEPW(I)   = EWINDO(2,I)*0.01                                  MODF2
000395      C                                                                   FILSEQ
000396          IF (MATCH.EQ.-1) GO TO 360                                      MODF
000397          GO TO 500                                                       FILSEQ
000398      C                                                                   FILSEQ
000399      C *** JPARM = 3, YES OUTPUT FILE (YESOUT) + SUBHST                  FILSEQ
000400      C                                                                   FILSLQ
000401        440 IF (MATCH.EQ.1) GO TO 470                                     FILSEQ
000402      C                                                                   FILSEQ
000403          READ (FYES) COUN4,REGN4,ZONE4,STRTA4,YSTR                       FILSEQ
000404           *,           ((IYSPNT(J,I),J=1,3),I=1,6)                       FILSEQ
000405      C                                                                   FILSEQ
000406          IF (COUN4.NE.IZZZZ) GO TO 450                                   FILSEQ
000407      C                                                                   FILSEQ
000408          IH = -1                                                         FILSEQ
000409          GO TO 500                                                       FILSEQ
000410      C                                                                   FILSEQ
000411        450 IF (MATCH.EQ.0) GO TO 460                                     FILSEQ
000412      C                                                                   FILSEQ
000413          MATCH = 0                                                       FILSEQ
000414      C                                                                   FILSEQ
000415          IF(IREGN - REGN4)460,900,940                                    MOD2
000416        900 IF ( IZONE - ZONE4 )  460,910,940                             MODF2
000417        910 IF(ISTRTA - STRTA4)460,920,940                                MOD2
000418        920 MATCH = 1                                                     MOD2
```

```
000419          GO TO 460                                                       MOD2
000420        940 MATCH = -1                                                    MOD2
000421          GO TO 440                                                       MOD2
000422      C                                                                   FILSEQ
000423      C *** MOVE YESOUT VALUES TO FILBUF                                  FILSEQ
000424      C                                                                   FILSEQ
000425        460 ICNTRY = COUN4                                                FILSEQ
000426          IREGN  = REGN4                                                  FILSEQ
000427          IZONE  = ZONE4                                                  FILSEQ
000428          ISTRTA = STRTA4                                                 FILSEQ
000429      C                                                                   FILSEQ
000430          XTYLD = YSTR                                                    FILSEQ
000431          DO 465 I = 1,6                                                  FILSEQ
000432          PRDATE(I) = IYSPNT(1,I)                                         FILSEQ
000433        465 XEYLD(I)  = YSPNT(2,I)                                        FILSEQ
000434      C                                                                   FILSEQ
000435          IF (MATCH.EQ.1) GO TO 500                                       FILSEQ
000436      C                                                                   FILSEQ
000437        470 READ (FSUB) COUN1,REGN1,ZONE1,STRTA1,SBSTA1,INEG              FILSEQ
000438           *,           (NDSEG(I),I=1,NWRDS)                            FILSEQ
000439      C                                                                   FILSEQ
000440          IF (COUN1.NE.IZZZZ) GO TO 480                                   FILSEQ
000441      C                                                                   FILSEQ
000442          IH = -1                                                         FILSEQ
000443          GO TO 500                                                       FILSEQ
000444      C                                                                   FILSEQ
000445        480 IF ( IREGN - REGN1 )  488,482,470                             MODF2
000446        482 IF ( IZONE - ZONE1 )  488,484,470                             MODF2
000447        484 IF (ISTRTA - STRTA1)  488,486,470                             MODF2
000448        486 MATCH = 1                                                     MODF2
000449          GO TO 490                                                       MODF2
000450        488 CONTINUE                                                      MODF2
000451      C                                                                   FILSEQ
000452          IF(MATCH .EQ. 0)MATCH =-1                                       MOD2
000453          IF(MATCH .EQ. 1)MATCH =-2                                       MOD2
000454      C                                                                   FILSEQ
000455          IREGN = REGN1                                                   FILSEQ
000456          IZONE = ZONE1                                                   FILSEQ
000457          ISTRTA = STRTA1                                                 FILSEQ
000458      C                                                                   FILSEQ
000459      C        CONVERT SUBSTRATA LAND AREA FROM KM**2 TO HECTARES         MODF2
000460        490  AREA = XSEG(MXSEG+3)*100.0                                  MODF2
000461      C        CONVERT TRUE PW FROM PERCENT TO FRACTION                   MODF2
000462          XTPW = XSEG(MXSEG+4)*0.01                                       MODF2
000463          N    = NDSEG(MXSEG+5)                                           FILSEQ
000464          NA   = NDSEG(MXSEG+6)                                           FILSEQ
000465      C                                                                   FILSEQ
000466          IF(MATCH .EQ. -1)GO TO 440                                      MOD2
000467      C                                                                   FILSEQ
000468      C *** RETURN                                                        FILSEQ
000469      C                                                                   FILSEQ
000470        500 RETURN                                                        FILSEQ
000471      C                                                                   FILSEQ
000472      C                                                                   FILSEQ
000473      C                                                                   FILSEQ
000474        600 CONTINUE                                                      FILSEQ
000475      C                                                                   FILSEQ
000476      C                                                                   FILSEQ
000477      C                                                                   FILSEQ
000478          GO TO 500                                                       FILSEQ
```

```
000479          C                                                                FILSEQ
000480          C *** NON-MATCH ON CASE NUMBER FOR FSUB (RPTYPE=1) OR FACQ (RPTYPE=4)  FILSEQ
000481          C                                                                FILSEQ
000482            700 STOP                                                       FILSEQ
000483          C                                                                FILSEQ
000484          C *** ERROR MESSAGES                                             FILSEQ
000485          C                                                                FILSEQ
000486           6200 FORMAT (50H *** ICASIN(I) DOES NOT MATCH CASE NUMBER ON FILE./  FILSEQ
000487               *         12H      ICASIN(,I1,2H)=,I4,14HFILE CASE NO.=,I4,  FILSEQ
000488               *         11H FILE NAME,2A4/1H /1X,16(6H******))             FILSEQ
000489                END                                                        FILSEQ
```

```
* L=  FREQAC,1,760511, 59124   , 1
000001            SUBROUTINE FREQAC                                                    FREQAC
000002      C                                                                          FREQAC
000003      C THIS ROUTINE PRODUCES THE FREQUENCY OF SAMPLE SEGMENT ACQUISITIONS      FREQAC
000004      C     REPORT                                                               FREQAC
000005      C                                                                          FREQAC
000006      C                                                                          FREQAC
000007      C *** COMMON STORAGE                                                       FREQAC
000008      C                                                                          FREQAC
000009            COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA                  FILBUF
000010           *,               ISUBSTA,ICASE,MXSEG                                 MODF
000011           *,               KBUF(125)                                           MODF
000012      C                                                                          FILBUF
000013            DIMENSION KWIN(4),IBIOWD(4),IPRDAT(14),PRDATE(6),XEPW(4),XEYLD(6)    FILBUF
000014           *,         APVAL(2)                                                  FILBUF
000015           *,         MCERR(100)                                                MODF
000016      C                                                                          FILBUF
000017            REAL    MCREF,MCERR                                                 MODF
000018            INTEGER FILENM,PRDATE                                              FILBUF
000019      C                                                                          FILBUF
000020            EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                              FILBUF
000021           *,  (KBUF(2),IGPRD,NSTRAT,NREGS)                                    MODF
000022           *,  (KBUF(3),XHPW,NZTOT)                                            MODF
000023           *,  (KBUF(4),AREA,NBK)                                              MODF
000024           *,  (KBUF(5),XTPW,NPD)                                              MODF
000025           *,  (KBUF(6),NP,IPIOWD(1))                                          MODF
000026           *,  (KBUF(7),NA)                                                    MODF
000027           *,  (KBUF(8),XBIPW)                                                 MODF
000028           *,  (KBUF(9),XCV2)                                                  MODF
000029           *,  (KBUF(10),XTPWI,IPRDAT(1))                                      MODF
000030           *,  (KBUF(11),PRDATE(1))                                            MODF
000031           *,  (KBUF(17),XEPW(1))                                              MODF
000032           *,  (KBUF(21),XEYLD(1))                                             MODF
000033           *,  (KBUF(25),APVAL(1),MCREF)                                       MODF
000034           *,  (KBUF(25),MCERR(1))                                             MODF
000035           *,  (KBUF(27),XTYLD)                                                MODF
000036      C                                                                          FREQAC
000037            COMMON /PRNTID/ CNTRY,REGION,ZONE,STRATA,ICASEN                      PRNTID
000038           *,               FILENO,FILNAM(2)                                    PRNTID
000039           *,               ITUTYP,NAMFMT                                       PRNTID
000040      C                                                                          PRNTID
000041            INTEGER CNTRY,REGION,ZONE,STRATA,FILENO,FILNAM                       MODF
000042      C                                                                          FREQAC
000043            COMMON /BWCMMN/ NSEGS,BIUCMR(16)                                     BWCMUN
000044      C                                                                          FREQAC
000045      C *** LOCAL STORAGE                                                        FREQAC
000046      C                                                                          FREQAC
000047            DIMENSION KNT(3),KNTBIO(16,3),KBIOZ(16),KBIOR(16), KBIOC(16)       FREQAC
000048      C                                                                          FREQAC
000049      C     SUBSCRIPT = 1  ZONE LEVEL                                            FREQAC
000050      C     SUBSCRIPT = 2  REGION LEVEL                                          FREQAC
000051      C     SUBSCRIPT = 3  COUNTRY LEVEL                                         FREQAC
000052      C                                                                          FREQAC
000053            EQUIVALENCE                                                        FREQAC
000054           *   (KNTBIO(1,1),KBIOZ(1)), (KNT(1),KNTZ)                          FREQAC
000055           *,  (KNTBIO(1,2),KBIOR(1)), (KNT(2),KNTR)                          FREQAC
000056           *,  (KNTBIO(1,3),KBIOC(1)), (KNT(3),KNTC)                          FREQAC
000057      C                                                                          FREQAC
000058      C                                                                          FREQAC
```

```
000059                IH = 1                                                FREQAC
000060                IFIRST = 0                                            FREQAC
000061          C                                                           FREQAC
000062                DO 20 I = 1,3                                         FREQAC
000063                KNT(I) = 0                                            FREQAC
000064                DO 20 J = 1,16                                        FREQAC
000065             20 KNTBIO(J,I) = 0                                       FREQAC
000066          C                                                           FREQAC
000067          C *** GET HEADER RECORD                                     FREQAC
000068          C                                                           FREQAC
000069                CALL FILSEQ(IH)                                       FREQAC
000070          C                                                           FREQAC
000071                ICASEN = ICASE                                        FREQAC
000072                FILNAM(1) = FILENM(1)                                 FREQAC
000073                FILNAM(2) = FILENM(2)                                 FREQAC
000074          C                                                           FREQAC
000075          C *** GET DETAIL RECORD                                     FREQAC
000076          C                                                           FREQAC
000077                IH = 0                                                FREQAC
000078          C                                                           FREQAC
000079             50 CALL FILSEQ(IH)                                       FREQAC
000080                IF (IH.LT.0) GO TO 200                                FREQAC
000081          C                                                           FREQAC
000082                IF (IFIRST.GT.0) GO TO 100                            FREQAC
000083          C                                                           FREQAC
000084                CNTRY = ICNTRY                                        FREQAC
000085                REGION = IREGN                                        FREQAC
000086                ZONE   = IZONE                                        FREQAC
000087                IFIRST = 1                                            MOD1
000088          C                                                           FREQAC
000089                GO TO 120                                             FREQAC
000090          C                                                           FREQAC
000091          C *** TEST FOR CHANGE IN I.D. AREA                          FREQAC
000092          C                                                           FREQAC
000093            100 IF (IZONE.GT.ZONE) GO TO 200                          FREQAC
000094                IF (IREGN.GT.REGION) GO TO 200                        FREQAC
000095          C                                                           FREQAC
000096            120 K = 0                                                 FREQAC
000097                KI = 0                                                FREQAC
000098                KNTZ = KNTZ+1                                         FREQAC
000099                DO 130 I = 1,4                                        FREQAC
000100                IF (KBIN(I).EQ.0) GO TO 130                           FREQAC
000101                K = K+1                                               FREQAC
000102                KI = KI+I                                             FREQAC
000103            130 CONTINUE                                              FREQAC
000104          C                                                           FREQAC
000105                IF (K.NE.0) GO TO 140                                 FREQAC
000106          C                                                           FREQAC
000107          C *** ZERO ACQUISITIONS                                     FREQAC
000108          C                                                           FREQAC
000109                KBIOZ(1) = KBIOZ(1)+1                                 FREQAC
000110                GO TO 50                                              FREQAC
000111          C                                                           FREQAC
000112            140 GO TO (150,160,180,180),K                             FREQAC
000113          C                                                           FREQAC
000114            150 KBIOZ(KI+1) = KBIOZ(KI+1)+1                           FREQAC
000115                GO TO 50                                              FREQAC
000116          C                                                           FREQAC
000117            160 IF (KBIN(1).EQ.0) GO TO 170                           FREQAC
000118          C                                                           FREQAC
```

```
000119            KBIOZ(KI+3) = KBIOZ(KI+3)+1                                   FREQAC
000120            GO TO 50                                                      FREQAC
000121          C                                                               FREQAC
000122        170 KBIOZ(KI+4) = KBIOZ(KI+4)+1                                   FREQAC
000123            GO TO 50                                                      FREQAC
000124          C                                                               FREQAC
000125        180 KBIOZ(KI+6) = KBIOZ(KI+6)+1                                   FREQAC
000126            GO TO 50                                                      FREQAC
000127          C                                                               FREQAC
000128          C *** CHANGE IN I.D. AREA                                       FREQAC
000129          C                                                               FREQAC
000130        200 DO 210 I = 1,16                                               FREQAC
000131            BIOZ = KBIOZ(I)                                               MOD1
000132            TZ =KNTZ                                                      MOD1
000133            BIOCHR(I) = (BIOZ/TZ)*100.                                    MOD1
000134        210 CONTINUE                                                      FREQAC
000135          C                                                               FREQAC
000136            NSEGS = KNTZ                                                  FREQAC
000137          C                                                               FREQAC
000138          C *** CALL REPORT WITH ITOTYP=1 FOR ZONE PRINT                  FREQAC
000139          C                                                               FREQAC
000140            ITOTYP = 1                                                    FREQAC
000141            CALL REPRT3                                                   FREQAC
000142          C                                                               FREQAC
000143          C *** SAVE ZONE VALUES IN REGION ACCUMULATORS                   FREQAC
000144          C                                                               FREQAC
000145            DO 220 I = 1,16                                               FREQAC
000146            KBIOR(I) = KBIOR(I)+KBIOZ(I)                                  FREQAC
000147        220 CONTINUE                                                      FREQAC
000148          C                                                               FREQAC
000149            KNTR = KNTR+KNTZ                                              FREQAC
000150          C                                                               FREQAC
000151          C *** ZERO OUT ZONE VALUES                                      FREQAC
000152          C                                                               FREQAC
000153            KNTZ = 0                                                      FREQAC
000154            DO 230 I = 1,16                                               FREQAC
000155        230 KBIOZ(I) = 0                                                  FREQAC
000156          C                                                               FREQAC
000157            IF (IH.LT.0) GO TO 240                                        FREQAC
000158          C                                                               FREQAC
000159            ZONE = IZONE                                                  FREQAC
000160            IF (IREGN.EQ.REGION) GO TO 120                                FREQAC
000161          C                                                               FREQAC
000162        240 DO 250 I = 1,16                                               FREQAC
000163            BIOR = KBIOR(I)                                               MOD1
000164            TR = KNTR                                                     MOD1
000165            BIOCHR(I) = (BIOR/TR)*100.                                    MOD1
000166        250 CONTINUE                                                      FREQAC
000167          C                                                               FREQAC
000168            NSEGS = KNTR                                                  FREQAC
000169          C                                                               FREQAC
000170          C *** CALL REPORT WITH ITOTYP=2 FOR REGION PRINT                FREQAC
000171          C                                                               FREQAC
000172            ITOTYP = 2                                                    FREQAC
000173            CALL REPRT3                                                   FREQAC
000174          C                                                               FREQAC
000175          C *** SAVE REGION VALUES IN COUNTRY ACCUMULATORS                FREQAC
000176          C                                                               FREQAC
000177            DO 260 I = 1,16                                               FREQAC
000178        260 KBIOC(I) = KBIOC(I)+KBIOR(I)                                  FREQAC
```

```
000179          C
000180                KNTC = KNTC+KNTR                                        FREQAC
000181          C                                                             FREQAC
000182          C *** ZERO OUT REGION VALUES                                  FREQAC
000183          C                                                             FREQAC
000184                KNTR = 0                                                FREQAC
000185                DO 270 I = 1,16                                         FREQAC
000186            270 KRIOK(I) = 0                                            FREQAC
000187          C                                                             FREQAC
000188                IF (IP.LT.0) GO TO 300                                  FREQAC
000189          C                                                             FREQAC
000190                REGION = IREGN                                          FREQAC
000191                GO TO 120                                               FREQAC
000192          C                                                             FREQAC
000193          C *** MOVE COUNTRY VALUES TO PRINT BUFFER                     FREQAC
000194          C                                                             FREQAC
000195            300 DO 320 I = 1,16                                         FREQAC
000196                TC = KNTC                                               MOD1
000197                BIOC =KRIOC(I)                                          MOD1
000198                BIOCMB(I) = (BIOC/TC)*100.                              MOD1
000199            320 CONTINUE                                                FREQAC
000200          C                                                             FREQAC
000201                NSEGS = KNTC                                            FREQAC
000202          C                                                             FREQAC
000203          C *** CALL REPORT WITH ITOTYP=3 FOR COUNTRY                   FREQAC
000204          C                                                             FREQAC
000205                ITOTYP = 3                                              FREQAC
000206                CALL REPRT3                                             FREQAC
000207          C                                                             FREQAC
000208          C *** RETURN TO DRIVER PROGRAM                                FREQAC
000209          C                                                             FREQAC
000210                RETURN                                                  FREQAC
000211          C                                                             FREQAC
000212                END                                                     FREQAC
```

```
* EOF FZULU,1,760511,-59142    , 1
000001          SUBROUTINE FZULU(IDAT,IOUT)                                    MOD3
000002        C ***************************************************************FZULU
000003        C                                                                FZULU
000004        C ROUTINE       FZULU                                            FZULU
000005        C                                                                FZULU
000006        C PURPOSE       TO CONVERT ZULU DATE TO YEAR MONTH AND DAY       FZULU
000007        C                                                                FZULU
000008        C LINKAGE       CALL FZULU (IDATE,IOUT)                          FZULU
000009        C                    IDATE,  BINARY INTEGER ZULU DATE            FZULU
000010        C                    IOUT,   CALENDER DATE,DIMENSION 3  FOR INTEGERFZULU
000011        C                            YEAR, MONTH AND DAY RESPECTIVELY    FZULU
000012        C                                                                FZULU
000013        C ROUTINES CALLED   NONE                                         FZULU
000014        C                                                                FZULU
000015        C LOCAL VARIABLES:                                               FZULU
000016        C                                                                FZULU
000017        C                    IDAYS      VECTOR CONTAINING NO. DAYS OF YEAR ON FZULU
000018        C                               LAST DAY OF MONTHS 0 THROUGH 12 FOR NORMALFZULU
000019        C                               YEAR FOLLOWED BY 13 MONTHS OF LEAP YEAR FZULU
000020        C                                                                FZULU
000021        C                    NLEAP,     NO. OF LEAP YEARS SINCE 1900 TO SPECIFIED FZULU
000022        C                               DATE                             FZULU
000023        C                    NFLG,      SUBSCRIPT INTO IDAYS VECTOR      FZULU
000024        C                                       START=1, NOT LEAP YEAR   FZULU
000025        C                                       START=14, LEAP YEAR      FZULU
000026        C                    JDAY,      JULIAN DAY OF YEAR               FZULU
000027        C                    I,         LOCAL USE                        FZULU
000028        C COMMENTS           NONE                                        FZULU
000029        C***************************************************************FZULU
000030              DIMENSION IOUT (3)                                         FZULU
000031              DIMENSION IDAYS(26)
000032              DATA IDAYS/0,31,59,90,120,151,181,212,243,273,304,334,365,  FZULU
000033             1            0,31,60,91,121,152,182,213,244,274,305,335,366/ FZULU
000034              IDATE = IDAT + 18263                                       MOD3
000035              DO 5 I=1,3                                                 FZULU
000036              IOUT(I) = 0                                                FZULU
000037            5 CONTINUE                                                   FZULU
000038        C                   FIND NO. LEAP YEARS SINCE 1900 AND GET CURRENT YEAR FZULU
000039              NLEAP=(1095+IDATE)/1461                                    FZULU
000040              IOUT(1) = (IDATE -NLEAP)/365
000041        C                   SEE IF CURRENT YEAR ISLEAP YEAR-SET NFLG=1 OR 14 FZULU
000042              I = IOUT(1)/4
000043              I=I*4                                                      FZULU
000044              NFLG=1                                                     FZULU
000045              IF(IOUT(1) -I)10,10,20
000046        10    NFLG=14                                                    FZULU
000047        C                   JULIAN DAY = ZULU DAY-NO YEARS*365-NO.LEAP YEARS FZULU
000048           20 JDAY=IDATE-(IOUT*365)-NLEAP + 1                            FZULU
000049        C                   GET DAY MONTH FROM TABLE SEARCH              FZULU
000050        C                                                                FZULU
000051        30    IF(JDAY-IDAYS(NFLG))50,50,40                               FZULU
000052        40    IOUT(2)=IOUT(2)+1                                          FZULU
000053              NFLG=NFLG+1                                                FZULU
000054              GOTO 30                                                    FZULU
000055        50    NFLG=NFLG-1                                                FZULU
000056              IOUT(3)=JDAY-IDAYS(NFLG)                                   FZULU
000057              RETURN                                                     FZULU
000058              END                                                        FZULU
```

```
@ ELT GENRNG,1,760511, 59136   , 1
000001            SUBROUTINE GENRNG(ICL)                                    MODF2
000002      C                                                               GENRNG
000003      C THIS ROUTINE GENERATES THE HISTOGRAM RANGES FROM THE INTERVAL INPUT  GENRNG
000004      C                                                               GENRNG
000005      C                                                               GENRNG
000006      C *** COMMON STORAGE                                            GENRNG
000007      C                                                               GENRNG
000008            COMMON /RANGE / START,INTVL1,BREAK1,INTVL2,BREAK2         RANGE
000009           *,               INTVL3,STOP                               RANGE
000010      C                                                               RANGE
000011            REAL    INTVL1,INTVL2,INTVL3                              RANGE
000012      C                                                               GENRNG
000013            COMMON /HSTOGM/ XMVAL,STDDEV,XMPSUM,SDPSUM                HSTOGM
000014           *,               NRANGE,IRANGE(51),NTOTL,RANGES(52)        HSTOGM
000015           *,               MINVAL,MAXVAL,DATPRD(3),NZRNG,REF         HSTOGM
000016      C                                                               HSTOGM
000017            REAL    MINVAL,MAXVAL                                     HSTOGM
000018            INTEGER DATPRD                                            HSTOGM
000019      C                                                               GENRNG
000020            DATA RNGMIN /-1.0E6/, RNGMAX /1.0E6/                      GENRNG
000021            DO 10 I = 1,52                                            GENRNG
000022         10 RANGES(I) = 0.0                                           GENRNG
000023      C                                                               GENRNG
000024            NRANGE = 0                                                MODF
000025            NZRNG = 0                                                 GENRNG
000026      C                                                               GENRNG
000027      C *** GENERATE RANGE VALUES                                     GENRNG
000028      C                                                               GENRNG
000029            K = 1                                                     GENRNG
000030      C                                                               GENRNG
000031            IF (ICL.EQ.0) RANGES(K) = RNGMIN                          MODF2
000032            IF (ICL.NE.0) K = 0                                       MODF2
000033      C                                                               GENRNG
000034            N1 = (BREAK1-START)/INTVL1                                GENRNG
000035      C                                                               GENRNG
000036            DO 20 I = 1,N1                                            GENRNG
000037            K = K+1                                                   GENRNG
000038            VALU = START+INTVL1*(I-1)                                 GENRNG
000039            IF (NZRNG.NE.0) GO TO 20                                  GENRNG
000040            IF (VALU.LT.0.) GO TO 20                                  GENRNG
000041            RANGES(K) = 0.0                                           GENRNG
000042            NZRNG = K                                                 GENRNG
000043            K = K+1                                                   GENRNG
000044            NRANGE = NRANGE+1                                         GENRNG
000045         20 RANGES(K) = VALU                                          GENRNG
000046            NRANGE = NRANGE+N1                                        GENRNG
000047            IF (INTVL2.NE.0.) GO TO 30                                GENRNG
000048      C                                                               GENRNG
000049            RANGES(K+1) = BREAK1                                      GENRNG
000050            GO TO 100                                                 GENRNG
000051      C                                                               GENRNG
000052         30 N1 = (BREAK2-BREAK1)/INTVL2                               GENRNG
000053      C                                                               GENRNG
000054            DO 40 I = 1,N1                                            GENRNG
000055            K = K+1                                                   GENRNG
000056            VALU = BREAK1+INTVL2*(I-1)                                GENRNG
000057            IF (NZRNG.NE.0) GO TO 40                                  GENRNG
000058            IF (VALU.LT.0.) GO TO 40                                  GENRNG
```

```
000059                RANGES(K) = 0.0                                    GENRNG
000060                NZRNG = K                                          GENRNG
000061                K = K+1                                            GENRNG
000062                NRANGE = NRANGE+1                                  GENRNG
000063             40 RANGES(K) = VALU                                   GENRNG
000064                NRANGE = NRANGE+N1                                 GENRNG
000065                IF (INTVL3.NE.0.) GO TO 50                         GENRNG
000066             C                                                     GENRNG
000067                RANGES(K+1) = BREAK2                               GENRNG
000068                GO TO 100                                          GENRNG
000069             C                                                     GENRNG
000070             50 N1 = (STOP-BREAK2)/INTVL3                          GENRNG
000071             C                                                     GENRNG
000072                DO 60 I = 1,N1                                     GENRNG
000073                K = K+1                                            GENRNG
000074                VALU = BREAK2+INTVL3*(I-1)                         GENRNG
000075                IF (NZRNG.NE.0) GO TO 60                           GENRNG
000076                IF (VALU.LT.0.) GO TO 60                           GENRNG
000077                RANGES(K) = 0.0                                    GENRNG
000078                NZRNG = K                                          GENRNG
000079                K = K+1                                            GENRNG
000080                NRANGE = NRANGE+1                                  GENRNG
000081             60 RANGES(K) = VALU                                   GENRNG
000082                NRANGE = NRANGE+N1                                 GENRNG
000083                RANGES(K+1) = STOP                                 GENRNG
000084            100 IF (ICL.NE.0) GO TO 120                           MODF2
000085                RANGES(K+2) = BIGMAX                               MODF2
000086                NRANGE = NRANGE+2                                  GENRNG
000087                GO TO 150                                          MODF2
000088            120 CONTINUE                                          MOD3
000089             C                                                     MODF2
000090            150 CONTINUE                                          MODF2
000091             C                                                     GENRNG
000092                RETURN                                             GENRNG
000093             C                                                     GENRNG
000094                END                                                GENRNG
```

```
* ELF LFPA.1,760511, 59141      , 1

000001            SUBROUTINE LFPA(FLDA,LMO,LYR,ALFGM,DAYS)                    LFPA
000002      C                                                                 LFPA
000003      C MODULE - P1LFPA                                                 LFPA
000004      C OCTOBER 1, 1973                                                 LFPA
000005      C                                                                 LFPA
000006      C                                                                 LFPA
000007      C  JUNE 26,1973                                                   LFPA
000008      C                                                                 LFPA
000009      C  MODULE PPLFPA - POINT TARGET PERFORMANCE PREDICTOR              LFPA
000010      C                                                                 LFPA
000011      C  GIVEN  DAY,MONTH,YEAR - SUBR. LFPA RETURNS THE RIGHT ASCENSION OF  LFPA
000012      C  GREENWICH AT MIDNIGHT OF A GIVEN DAY-                          LFPA
000013      C                                                                 LFPA
000014      C     REFERENCE EPOCH IS 0 HOUR 1JAN1950                          LFPA
000015            DATA RADIAN /57.29578/                                      LFPA
000016      C                                                                 LFPA
000017      C     COMPUTE DAYS IN FULL YEARS FROM EPOCH TO LYR                LFPA
000018      C     DAYS TO 1JAN1963 IS 4748.                                   LFPA
000019      C                                                                 LFPA
000020            DAYS=4748.                                                  LFPA
000021      C     NOTE--LYR MUST BE GREATER THAN OR EQUAL TO (19)64           LFPA
000022            LASTYR=LYR-1                                                LFPA
000023            DO 20 I=63,LASTYR                                           LFPA
000024            KREMD=MOD((I-60),4)                                         LFPA
000025            IF(KREMD.GT.0) GO TO 10                                     LFPA
000026            DAYS=DAYS+366                                               LFPA
000027            GO TO 20                                                    LFPA
000028         10 DAYS=DAYS+365                                               LFPA
000029         20 CONTINUE                                                    LFPA
000030            IF(LMO-1)30,40,3)                                           LFPA
000031         30 IF(LMO-2)60,50,60                                           LFPA
000032         40 DAYS=DAYS+FLDA-1.                                           LFPA
000033            GO TO 270                                                   LFPA
000034         50 DAYS=DAYS+FLDA+30.                                          LFPA
000035            GO TO 270                                                   LFPA
000036         60 KDEL=MOD((LYR-60),4)                                        LFPA
000037            IF(KDEL.GT.0) GO TO 70                                      LFPA
000038            DAYS=DAYS+59.                                               LFPA
000039            GO TO 80                                                    LFPA
000040         70 DAYS=DAYS+58.                                               LFPA
000041         80 IF(LMO-3) 270,170,90                                        LFPA
000042         90 IF(LMO-5) 180,190,100                                       LFPA
000043        100 IF(LMO-7) 200,210,110                                       LFPA
000044        110 IF(LMO-9) 220,230,120                                       LFPA
000045        120 IF(LMO-11)240,250,260                                       LFPA
000046        170 DAYS=DAYS+FLDA                                              LFPA
000047            GO TO 270                                                   LFPA
000048        180 DAYS=DAYS+FLDA+31.                                          LFPA
000049            GO TO 270                                                   LFPA
000050        190 DAYS=DAYS+FLDA+61.                                          LFPA
000051            GO TO 270                                                   LFPA
000052        200 DAYS=DAYS+FLDA+92.                                          LFPA
000053            GO TO 270                                                   LFPA
000054        210 DAYS=DAYS+FLDA+122.                                         LFPA
000055            GO TO 270                                                   LFPA
000056        220 DAYS=DAYS+FLDA+153.                                         LFPA
000057            GO TO 270                                                   LFPA
000058        230 DAYS=DAYS+FLDA+184.                                         LFPA
```

```
000059          GO TO 270                                                       LFPA
000060      240 DAYS=DAYS+FLOA+214.                                             LFPA
000061          GO TO 270                                                       LFPA
000062      250 DAYS=DAYS+FLOA+245.                                             LFPA
000063          GO TO 270                                                       LFPA
000064      260 DAYS=DAYS+FLOA+275.                                             LFPA
000065      270 CONTINUE                                                        LFPA
000066          ALFGM=AMOD(SNGL((.10007554203+(.985647346D0)*DAYS+              LFPA
000067         1(2.9015L-13)*DAYS**2))*360.)                                    LFPA
000068          ALFGM=ALFGM/RADIAN                                              LFPA
000069          RETURN                                                          LFPA
000070          END                                                             LFPA
```

```
* ELT MCHIST,1,760511, 59147   , 1
000001            SUBROUTINE MCHIST                                                    MCHIST
000002      C                                                                          MCHIST
000003      C *** THIS ROUTINE PRODUCES THE MONTE CARLO REPORTS                        MCHIST
000004      C                                                                          MCHIST
000005      C                                                                          MCHIST
000006      C *** COMMON STORAGE                                                       MCHIST
000007      C                                                                          MCHIST
000008            COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                               CARDIN
000009           *,              ICASIN(5),LEVEL,JPARM,IERR                            CARDIN
000010            INTEGER RPTYPE,AUNITS,PARMTR                                        CARDIN
000011      C                                                                          MCHIST
000012            COMMON /RANGE / START,INTVL1,BREAK1,INTVL2,BREAK2                    RANGE
000013           *,               INTVL3,STOP                                         RANGE
000014      C                                                                          RANGE
000015            REAL    INTVL1,INTVL2,INTVL3                                       RANGE
000016      C                                                                          MCHIST
000017            COMMON /PRTCTL/ NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)                PRTCTL
000018            INTEGER HEADER                                                     PRTCTL
000019      C                                                                          MCHIST
000020            COMMON /SELCTN/ BIOND(4),WPRTY(4),IPRD(3,14)                       SELCTN
000021      C                                                                          SELCTN
000022            INTEGER BIOND,WPRTY                                                SELCTN
000023      C                                                                          MCHIST
000024            COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA                FILBUF
000025           *,               ISBSTA,ICASE,MXSEG                                 MODF
000026           *,               KBUF(125)                                          MODF
000027      C                                                                          FILBUF
000028            DIMENSION KWIN(4),IBIOPD(4),IPRDAT(14),PRDATE(6),XEPM(4),XEYLD(6)  FILBUF
000029           *,         APVAL(2)                                                 FILBUF
000030           *,         MCERR(100)                                               MODF
000031      C                                                                          FILBUF
000032            REAL    MCREF,MCERR                                                MODF
000033            INTEGER FILENM,PRDATE                                              FILBUF
000034      C                                                                          FILBUF
000035            EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                              FILBUF
000036           *,  (KBUF(2),IGPNO,NSTRAT,NREGS)                                    MODF
000037           *,  (KBUF(3),XMPSM,NZTOT)                                           MODF
000038           *,  (KBUF(4),AREA,XBM)                                              MODF
000039           *,  (KBUF(5),XTP(1),IPPD)                                           MODF
000040           *,  (KBUF(6),N,IBIOPD(1))                                           MODF
000041           *,  (KBUF(7),NA)                                                    MODF
000042           *,  (KBUF(8),XRTPM)                                                 MODF
000043           *,  (KBUF(9),XCV2)                                                  MODF
000044           *,  (KBUF(10),XTP(1),IPRDAT(1))                                     MODF
000045           *,  (KBUF(11),PRDATE(1))                                            MODF
000046           *,  (KBUF(17),XEP(1))                                               MODF
000047           *,  (KBUF(21),XEYLD(1))                                             MODF
000048           *,  (KBUF(24),APVAL(1),MCREF)                                       MODF
000049           *,  (KBUF(25),MCERR(1))                                             MODF
000050           *,  (KBUF(27),XTYLD)                                                MODF
000051      C                                                                          MCHIST
000052            COMMON /HSTOGM/ AMVAL,STDDEV,XMPSUM,SQPSUM                         HSTOGM
000053           *,               NRANGE,IRANGE(51),NTOTL,RANGES(52)                 HSTOGM
000054           *,               MINVAL,MAXVAL,DATPRD(3),NZRNG,REF                  HSTOGM
000055      C                                                                          HSTOGM
000056            REAL    MINVAL,MAXVAL                                              HSTOGM
000057            INTEGER DATPRD                                                     HSTOGM
000058      C                                                                          MCHIST
```

```
000059            COMMON /PRNTID/ CNTRY,REGION,ZONE,STRATA,ICASEN                  PRNTID
000060           *,               FILENO,FILNAM(2)                                 PRNTID
000061           *,               ITUTYP,NAMFMT                                    PRNTID
000062      C                                                                      PRNTID
000063            INTEGER CNTRY,REGION,ZONE,STRATA,FILENO,FILNAM                   MODF
000064      C                                                                      MCHIST
000065            COMMON /READNG/ READNG(5600)                                     MODF
000066      C                                                                      MODF
000067            DIMENSION CLEWA(100),CLEPRO(100),CLATEC(100),CLPTEC(100)         MODF
000068           *,         CLATWC(100),CLPTWC(100)                               MODF
000069      C                                                                      MODF
000070            EQUIVALENCE                          (READNG(5001),CLEWA(1))   MODF
000071           *, (READNG(5101),CLEPRO(1)),          (READNG(5201),CLATEC(1))  MODF
000072           *, (READNG(5301),CLPTEC(1)),          (READNG(5401),CLATWC(1))  MODF
000073           *, (READNG(5501),CLPTWC(1))                                      MODF
000074      C                                                                      MCHIST
000075      C                                                                      MCHIST
000076      C *** LOCAL STORAGE                                                    MCHIST
000077      C                                                                      MCHIST
000078            DATA CLSTRT,CLIVL1,CLBRK1,CLIVL2,CLBRK2,CLIVL3,CLSTOP            MCHIST
000079           *     /0.0,2.0,100.0,4*0.0/                                      MCHIST
000080            IPNT = 0                                                        MCHIST
000081            NREC = 1                                                        MCHIST
000082            IH   = 1                                                        MCHIST
000083            ICL = 0                                                         MODF2
000084      C                                                                     MCHIST
000085      C *** GET HEADER RECORD                                               MCHIST
000086      C                                                                     MCHIST
000087            CALL FILDA(IH,IPNT,NREC)                                        MCHIST
000088      C                                                                     MCHIST
000089            ICASEN = ICASE                                                  MCHIST
000090            FILNAM(1) = FILENM(1)                                           MCHIST
000091            NAMFMT = 1                                                      MCHIST
000092            CNTRY  = ICNTRY                                                 MCHIST
000093      C                                                                     MCHIST
000094      C *** CHECK CASE NUMBERS                                              MCHIST
000095            DO 50 I = 1,4                                                   MCHIST
000096            IF (PARMTR(I).EQ.0) GO TO 50                                    MCHIST
000097            IF (ICASIN(I).EQ.ICASE) GO TO 50                                MCHIST
000098            CALL PAGHDR(-4)                                                 MCHIST
000099            WRITE (KOUT,6500) I,ICASIN(I),ICASE,FILNAM(1)                   MCHIST
000100            PARMTR(I) = -PARMTR(I)                                          MCHIST
000101         50 CONTINUE                                                        MCHIST
000102      C                                                                     MCHIST
000103            MAXPRP = NBR*NPD                                                MCHIST
000104      C                                                                     MCHIST
000105      C *** GENERATE RANGE VALUES                                           MCHIST
000106      C                                                                     MCHIST
000107            CALL GENRNG(ICL)                                                MODF2
000108      C *** LOOP ON PARMTR VALUES                                           MCHIST
000109            DO 400 I = 1,4                                                  MCHIST
000110      C                                                                     MCHIST
000111            IPRT = 0                                                        MCHIST
000112            JMIN = 0                                                        MCHIST
000113            JPRD = 0                                                        MCHIST
000114      C                                                                     MCHIST
000115            IF (PARMTR(I).LE.0) GO TO 400                                   MCHIST
000116      C                                                                     MCHIST
000117            JPARM = I                                                       MCHIST
000118            LEVEL = 1                                                       MCHIST
```

```
000119          NRPT  = 1                                                    MCHIST
000120        C ***************************************************************MODF2
000121        C                                                                MODF2
000122        C *** ALL MONTE CARLO REPORTS WILL BE PROCESSED AT COUNTRY LEVEL ONLY **MODF2
000123        C                                                                MODF2
000124          LEVEL = 3                                                      MODF2
000125        C                                                                MODF2
000126        C ***************************************************************MODF2
000127        C                                                                MODF2
000128        C                                                                MCHIST
000129        C *** SET VALUES FOR SPECIAL CASE (PARMTR(4)=N)                  MCHIST
000130        C                                                                MCHIST
000131          IF (JPARM.LT.4) GO TO 100                                      MCHIST
000132        C                                                                MCHIST
000133          LEVEL = 3                                                      MCHIST
000134          NRPT  = 6                                                      MCHIST
000135        C                                                                MCHIST
000136          START  = CLSTRT                                                MCHIST
000137          INTVL1 = CLIVL1                                                MCHIST
000138          BREAK1 = CLBRK1                                                MCHIST
000139          INTVL2 = CLIVL2                                                MCHIST
000140          BREAK2 = CLBRK2                                                MCHIST
000141          INTVL3 = CLIVL3                                                MCHIST
000142          STOP   = CLSTOP                                                MCHIST
000143        C                                                                MCHIST
000144          ICL = 1                                                        MODF2
000145          CALL GETRNG(ICL)                                               MODF2
000146        C                                                                MCHIST
000147        C *** LOOP ON PREDICTION POINTS                                  MCHIST
000148        C                                                                MCHIST
000149        100 CONTINUE                                                     MCHIST
000150        C                                                                MCHIST
000151          IF (JWIN.LT.0) GO TO 130                                       MCHIST
000152        C *** PROCESS BIO-WINDOWS                                        MCHIST
000153          JWIN = JWIN+1                                                  MCHIST
000154        C                                                                MCHIST
000155          IF (JWIN.GT.4) GO TO 120                                       MCHIST
000156        C                                                                MCHIST
000157          IG = IBIOWD(IPNT+1)                                            MOD3
000158        C                                                                MCHIST
000159          IF(IG .EQ. JWIN)GO TO 110                                      MOD3
000160          IF (BIOWD(JWIN).EQ.0) GO TO 100                                MCHIST
000161        C                                                                MCHIST
000162          CALL PATHDR(-4)                                                MCHIST
000163          WRITE (IOUT,6510) JWIN,ICASE,FILNAM(1)                         MCHIST
000164          GO TO 100                                                      MCHIST
000165        C                                                                MCHIST
000166        110 IPNT = IPNT+1                                                MCHIST
000167          IF(BIOWD(IG) .NE. 0)GO TO 150                                  MOD3
000168          GO TO 100                                                      MCHIST
000169        C                                                                MCHIST
000170        120 JWIN = -1                                                    MCHIST
000171          JPNT = IPNT                                                    MCHIST
000172          JMX  = 1                                                       MCHIST
000173        C                                                                MCHIST
000174        130 JPRD = JPRD+1                                                MCHIST
000175          IF(IPRD(1,JPRD) .EQ. 0)GO TO 400                               MOD3
000176        C                                                                MCHIST
000177          FLDA = IPRD(3,JPRD)                                            MCHIST
000178          LMD  = IPRD(2,JPRD)                                            MCHIST
```

```
000179                LYR  = IPRD(1,JPRD)                                    MCHIST
000180          C                                                            MCHIST
000181                CALL LFPA(FLDA,LMD,LYR,ALFGM,DAYS)                     MCHIST
000182                IDATE = DAYS                                           MCHIST
000183          C                                                            MCHIST
000184                DO 135 INX = JNX,NPD                                   MCHIST
000185                IF (IDATE.EQ.IPRDAT(INX)) GO TO 140                    MCHIST
000186            135 CONTINUE                                               MCHIST
000187          C                                                            MCHIST
000188                CALL PAGHDR(-4)                                        MCHIST
000189                WRITE (KOUT,6520) RPTYPE,JPARM,PARMTR(JPARM),(IPRD(L,JPRD),L=1,3)
000190                GO TO 130                                              MCHIST
000191          C                                                            MCHIST
000192            140 IPNT = JPNT+INX                                        MCHIST
000193                JNX  = INX+1                                           MCHIST
000194          C                                                            MCHIST
000195                DO 145 K = 1,3                                         MCHIST
000196            145 DATPRD(1) = IPRD(1,JPRD)                               MCHIST
000197          C                                                            MCHIST
000198          C *** LOOP ON LEVEL VALUE                                    MCHIST
000199          C                                                            MCHIST
000200            150 CONTINUE                                               MCHIST
000201          C                                                            MCHIST
000202                GO TO (155,160,165),LEVEL                              MCHIST
000203          C                                                            MCHIST
000204          C *** LEVEL = 1                                              MCHIST
000205          C                                                            MCHIST
000206            155 IREC = 14+(IPNT-1)*63                                  MODF2
000207                NMAX = NZTOT                                           MCHIST
000208                GO TO 170                                              MCHIST
000209          C *** LEVEL = 2                                              MCHIST
000210          C                                                            MCHIST
000211            160 IREC =  4+(IPNT-1)*63                                  MODF2
000212                NMAX = NREGS                                           MCHIST
000213                GO TO 170                                              MCHIST
000214          C *** LEVEL = 3                                              MCHIST
000215          C                                                            MCHIST
000216            165 IREC =  1+(IPNT-1)*63                                  MODF2
000217                NMAX = 1                                               MCHIST
000218          C                                                            MCHIST
000219            170 CONTINUE                                               MCHIST
000220          C                                                            MCHIST
000221          C *** LOOP ON NUMBER OF REPORTS & NMAX VALUE                 MCHIST
000222          C                                                            MCHIST
000223                DO 300 IRPT = 1,NRPT                                   MCHIST
000224          C                                                            MCHIST
000225          C     INITIALIZE ACCUMULATOR VALUES                          MCHIST
000226          C                                                            MCHIST
000227                J      = 0                                             MCHIST
000228                IH     = 0                                             MCHIST
000229                DELHR  = 0.0                                           MCHIST
000230                REF    = 0.0                                           MCHIST
000231                DELSQ  = 0.0                                           MCHIST
000232                MINVAL = 1.E10
000233                MAXVAL = -1.E10
000234                NREC = IREC                                            MODF2
000235          C                                                            MCHIST
000236                DO 175 IX = 1,51                                       MCHIST
000237            175 IRANGE(IX) = 0                                         MCHIST
000238          C                                                            MCHIST
```

```
000239          C *** IF CONFIDENCE LEVELS, GET ONE SET OF VALUES PER REPORT          MCHIST
000240          C                                                                     MCHIST
000241          C                                                                     MCHIST
000242               DO 200 IDX = 1,NMAX                                              MCHIST
000243          C                                                                     MCHIST
000244               NREC = NREC+1                                                    MCHIST
000245          C                                                                     MCHIST
000246               CALL FILDA(IN,IPNT,NREC)                                         MCHIST
000247          C                                                                     MCHIST
000248               IF (JPARM.EQ.4) CALL CLMOVE(IRPT)                                MCDF2
000249               REF = REF+MCREF                                                  MCHIST
000250          C                                                                     MCHIST
000251               DO 190 IRNG = 1,NT                                               MCHIST
000252          C                                                                     MCHIST
000253               J = J+1                                                          MCHIST
000254          C                                                                     MCHIST
000255               READNG(J) = MCERR(IRNG)                                          MCHIST
000256          C                                                                     MCHIST
000257               MINVAL = AMIN1(MINVAL,READNG(J))                                 MCHIST
000258               MAXVAL = AMAX1(MAXVAL,READNG(J))                                 MCHIST
000259          C                                                                     MCHIST
000260               DELERR = DELERR+READNG(J)                                        MCHIST
000261               DELSQ  = DELSQ+READNG(J)**2                                      MCHIST
000262          C                                                                     MCHIST
000263            190 CONTINUE                                                        MCHIST
000264            200 CONTINUE                                                        MCHIST
000265          C                                                                     MCHIST
000266          C *** TALLY PERCENT RANGES                                            MCHIST
000267          C                                                                     MCHIST
000268               NTOTL = J                                                        MCHIST
000269          C                                                                     MCHIST
000270               DO 250 N1 = 1,NTOTL                                              MCHIST
000271               PCDERR = 0.0                                                     MCHIST
000272               IF (REF.NE.0.0) PCDERR = 100.*READNG(N1)/REF                     MCHIST
000273          C                                                                     MCHIST
000274               DO 220 K = 1,NRANGE                                              MCHIST
000275               IF (PCDERR.NE.0) GO TO 210                                       MCHIST
000276          C                                                                     MCHIST
000277               IRANGE(NZRNG) = IRANGE(NZRNG)+1                                  MCHIST
000278               GO TO 250                                                        MCHIST
000279          C                                                                     MCHIST
000280            210 IF (PCDERR.GE.RANGES(K+1)) GO TO 220                            MCHIST
000281          C                                                                     MCHIST
000282               IRANGE(K) = IRANGE(K)+1                                          MCHIST
000283               GO TO 250                                                        MCHIST
000284          C                                                                     MCHIST
000285            220 CONTINUE                                                        MCHIST
000286            250 CONTINUE                                                        MCHIST
000287          C                                                                     MCHIST
000288               DATPRD(1) = JWIN                                                 MCDF2
000289               IF (JWIN.GT.0) GO TO 280                                         MCHIST
000290          C                                                                     MCHIST
000291               CALL FZULU(IDATE,DATPRD)                                         MCHIST
000292          C                                                                     MCHIST
000293            280 TJ = NTOTL                                                      MCHIST
000294          C                                                                     MCHIST
000295               XMVAL = 0.0                                                      MCHIST
000296               IF (REF.NE.0.0) XMVAL = 100./(TJ*REF)*DELERR                     MCHIST
000297               STDDEV = 0.0                                                     MCHIST
000298               IF((DELERR**2/TJ - DELSQ) .GT. 0.)GO TO 290
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000299            IF (J.GT.1.AND.REF.NE.0.0) STDDEV =100./REF                    MODF2
000300           *                  *SQRT(1./(TJ-1.)*(DELSQ-1./TJ*DELERR**2))    MCHIST
000301         C                                                                  MCHIST
000302           290 CALL REPRT2
000303         C                                                                  MCHIST
000304           300 CONTINUE                                                     MCHIST
000305         C                                                                  MCHIST
000306            IF (LEVEL.EQ.3) GO TO 350                                       MCHIST
000307         C                                                                  MCHIST
000308            LEVEL = LEVEL+1                                                 MCHIST
000309            GO TO 150                                                       MCHIST
000310         C                                                                  MCHIST
000311           350 IF (IPNT.LT.MAXPRP) GO TO 100                                MCHIST
000312         C                                                                  MCHIST
000313           400 CONTINUE                                                     MCHIST
000314         C                                                                  MCHIST
000315           500 RETURN                                                       MCHIST
000316         C                                                                  MCHIST
000317         C *** ERROR MESSAGES                                               MCHIST
000318         C                                                                  MCHIST
000319            6500 FORMAT (50H *** ICASIN(I) DOES NOT MATCH CASE NUMBER ON FILE./ MCHIST
000320           *        12H     ICASIN(,I1,2H)=,I4,15H FILE CASE NO.=,I4,      MODF2
000321           *        11H FILE NAME ,A6/1H /1X,18(6H******))                  MODF2
000322            6510 FORMAT (38H *** REQUESTED BIO-WINDOW NOT ON FILE./         MCHIST
000323           *        20H     BIO-WINDOW NO.=,I1,15H FILE CASE NO.=,I4,      MODF2
000324           *        11H FILE NAME ,A6/1H /1X,18(6H******))                  MODF2
000325            6520 FORMAT (54H *** INPUT PREDICTION DATE DOES NOT MATCH DATE ON FILE/MCHIST
000326           *        12H     RPTYPE=,I2,8H PARMTR(,I1,2H)=,I1,12H INPUT DATE=, MCHIST
000327           *        3I2/1H /1X,18(6H******))                                MCHIST
000328.           END                                                              MCHIST
```

```
@ ED/ PAGHDR,1,760511, 59134    , 1
000001                SUBROUTINE PAGHDR(N)                                  PAGHDR
000002          C                                                           PAGHDR
000003          C     ROUTINE CONTROLS COUNT ON PRINTED OUTPUT AND PROVIDES A LINE  PAGHDR
000004          C       OF PRINTED HEADING ON THE TOP OF EACH OUTPUT PAGE.  PAGHDR
000005          C                                                           PAGHDR
000006          C *** COMMON STORAGE                                        PAGHDR
000007          C                                                           PAGHDR
000008                COMMON /PRTCTL/ NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)   PRTCTL
000009                INTEGER HEADER                                        PRTCTL
000010          C                                                           PAGHDR
000011          C *** LOCAL STORAGE                                         PAGHDR
000012          C                                                           PAGHDR
000013                DATA LPRINT /-1/                                      PAGHDR
000014          C                                                           PAGHDR
000015          C                                                           PAGHDR
000016                IF (LPRINT) 3,4,5                                     PAGHDR
000017          C                                                           PAGHDR
000018              3 N1 = MAXLIN-3                                         PAGHDR
000019                N2 = MAXLIN-1                                         MODF
000020                LPRINT = 1                                            PAGHDR
000021                GO TO 30                                              PAGHDR
000022          C                                                           PAGHDR
000023              4 LPRINT = 1                                            PAGHDR
000024                GO TO 30                                              PAGHDR
000025          C                                                           PAGHDR
000026              5 IF (N.LE.0) GO TO 10                                  PAGHDR
000027          C                                                           PAGHDR
000028                J = NLINES+N                                          PAGHDR
000029                IF (J.GT.N1) GO TO 10                                 PAGHDR
000030          C                                                           PAGHDR
000031                NLINES = J                                            PAGHDR
000032                GO TO 50                                              PAGHDR
000033          C                                                           PAGHDR
000034             10 CONTINUE                                              PAGHDR
000035                IF (N.NE.0) GO TO 30                                  MODF
000036          C                                                           MODF
000037                DO 20 J = NLINES,N2                                   MODF
000038             20 WRITE (KOUT,100)                                      PAGHDR
000039          C                                                           PAGHDR
000040                LPRINT = 0                                            PAGHDR
000041                GO TO 50                                              PAGHDR
000042          C                                                           PAGHDR
000043             30 NPAGE = NPAGE+1                                       PAGHDR
000044                WRITE (KOUT,120) (HEADER(I),I=1,16),NPAGE             PAGHDR
000045          C                                                           PAGHDR
000046                NLINES = 3+IABS(N)                                    PAGHDR
000047          C                                                           PAGHDR
000048             50 RETURN                                                PAGHDR
000049          C                                                           PAGHDR
000050            100 FORMAT (1H )                                          PAGHDR
000051            120 FORMAT (1H1,6X,16A6,I4/1H0)                           PAGHDR
000052          C                                                           PAGHDR
000053                END                                                   PAGHDR
```

```
* E.  POPDRV,1,760511, 59117   , 1
000001            SUBROUTINE POPDRV                                          POPDRV
000002          C                                                            POPDRV
000003          C THIS ROUTINE IS THE DRIVER FOR THE POPULATION REPORTS      POPDRV
000004          C                                                            POPDRV
000005          C                                                            POPDRV
000006          C *** COMMON STORAGE                                         POPDRV
000007          C                                                            POPDRV
000008            COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                     CARDIN
000009           *,              ICASIN(5),LEVEL,JPARM,IERR                  CARDIN
000010            INTEGER RPTYPE,AUNITS,PARMTR                               CARDIN
000011          C                                                            POPDRV
000012          C *** GENERATE RANGES FOR HISTOGRAMS                         POPDRV
000013          C                                                            POPDRV
000014            ICL = 0                                                    MODE2
000015            CALL GENRNG(ICL)                                           MODE2
000016          C                                                            POPDRV
000017          C *** PARMTR(1)                                              POPDRV
000018          C                                                            POPDRV
000019            IF (PARMTR(1).EQ.0) GO TO 20                               POPDRV
000020          C                                                            POPDRV
000021            JPARM = 1                                                  POPDRV
000022          C                                                            POPDRV
000023            CALL SAMERR                                                POPDRV
000024          C                                                            POPDRV
000025            IERR = 0                                                   POPDRV
000026          C                                                            POPDRV
000027          C *** PARMTR(2)                                              POPDRV
000028          C                                                            POPDRV
000029           20 IF (PARMTR(2).EQ.0) GO TO 40                             POPDRV
000030          C                                                            POPDRV
000031            JPARM = 2                                                  POPDRV
000032          C                                                            POPDRV
000033            CALL CAMSER                                                POPDRV
000034          C                                                            POPDRV
000035            IERR = 0                                                   POPDRV
000036          C                                                            POPDRV
000037          C *** PARMTR(3)                                              POPDRV
000038          C                                                            POPDRV
000039           40 IF (PARMTR(3).EQ.0) GO TO 60                             POPDRV
000040          C                                                            POPDRV
000041            JPARM = 3                                                  POPDRV
000042          C                                                            POPDRV
000043            CALL YLDERR                                                POPDRV
000044          C                                                            POPDRV
000045            IERR = 0                                                   POPDRV
000046          C                                                            POPDRV
000047          C *** PARMTR(4)                                              POPDRV
000048          C                                                            POPDRV
000049           60 IF (PARMTR(4).EQ.0) GO TO 80                             POPDRV
000050          C                                                            POPDRV
000051            JPARM = 4                                                  POPDRV
000052          C                                                            POPDRV
000053            CALL ANPERR                                                POPDRV
000054          C                                                            POPDRV
000055            IERR = 0                                                   POPDRV
000056          C                                                            POPDRV
000057          C *** PARMTR(5)                                              POPDRV
000058          C                                                            POPDRV
```

```
000059          80 IF (PARMTR(5).EQ.0) GO TO 100                        POPDRV
000060        C                                                         POPDRV
000061            JPARM = 5                                             POPDRV
000062        C                                                         POPDRV
000063            CALL AMPERR                                           POPDRV
000064        C                                                         POPDRV
000065        C                                                         POPDRV
000066          100 RETURN                                              POPDRV
000067        C                                                         POPDRV
000068            END                                                   POPDRV
```

```
* ...T POUT,1,760511, 59113     , 1
000001          C
000002          C     THE POST PROCESSOR PROGRAM                                  POUT
000003          C                                                                 POUT
000004          C *** COMMON STORAGE                                              POUT
000005          C                                                                 POUT
000006                COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                      CARDIN
000007               *,              ICASIN(5),LEVEL,JPARM,IERR                   CARDIN
000008                INTEGER RPTYPE,AUNITS,PARMTR                                CARDIN
000009          C                                                                 POUT
000010                COMMON /RANGE / START,INTVL1,BREAK1,INTVL2,BREAK2            RANGE
000011               *,               INTVL3,STOP                                 RANGE
000012          C                                                                 RANGE
000013                REAL    INTVL1,INTVL2,INTVL3                                RANGE
000014          C                                                                 POUT
000015                COMMON /SELCTN/ BIOWD(4),WPRTY(4),IPRD(3,14)                SELCTN
000016          C                                                                 SELCTN
000017                INTEGER BIOWD,WPRTY                                         SELCTN
000018          C                                                                 POUT
000019                COMMON /PRTCTL/ NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)         PRTCTL
000020                INTEGER HEADER                                              PRTCTL
000021          C                                                                 POUT
000022                COMMON /CONVRT/ CTAMER(3),CTMTRC(3)                         CONVRT
000023          C                                                                 POUT
000024                COMMON /SUBHS/ICSESH                                        MOD3
000025          C                                                                 POUT
000026          C *** LOCAL STORAGE                                               POUT
000027          C                                                                 POUT
000028                DIMENSION IHEADR(12)                                        POUT
000029                DIMENSION RANGE(7),RNGNML(7),INPVL1(15),INPVL2(50)          POUT
000030               *,         AMCVT(3),XMCVT(3),NCARD(5),MHDR(4)                POUT
000031          C                                                                 POUT
000032                LOGICAL INERRS                                              POUT
000033                EQUIVALENCE (RANGE(1),START), (INPVL1(1),RPTYPE)            POUT
000034               *,           (INPVL2(1),BIOWD(1))                          POUT
000035                DATA IN,IOUT /5,6/, NOMRNG /1/, IBLNK /1H /, LNMAX /45/     POUT
000036                DATA (RNGNML(I),I=1,7) /-500.,100.,-100.,5.,100.,100.,500./ POUT
000037                DATA NHDR,NINP1,NINP2,NRGE /12,15,50,7/, INERRS /.FALSE./   POUT
000038                DATA (AMCVT(I),I=1,3) /2.471044E-04,3.6743544E-05,1.4869664/ MODE2
000039                DATA (XMCVT(I),I=1,3) /.001,.0001,1./                       MODE2
000040                DATA LABEL,(NCARD(I),I=1,5) /4HPOUT,2H01,2H02,2H03,2H04,2H05/ POUT
000041                DATA (MHDR(I),I=1,4) /6H    LP,6HP SIMU,6HLATION,6H  PAGE/  POUT
000042                DATA MRPTYP /4/, XMXRGE /51./, JLEVEL/1/                    POUT
000043          C                                                                 POUT
000044          C INITIALIZE INPUT QUANTITIES                                     POUT
000045          C                                                                 POUT
000046                DO 10 I = 1,NHDR                                            POUT
000047             10 IHEADR(I) = IBLNK                                           POUT
000048          C                                                                 POUT
000049                DO 20 I = 1,NINP1                                           POUT
000050             20 INPVL1(I) = 0                                               POUT
000051          C                                                                 POUT
000052                DO 30 I = 1,NRGE                                            POUT
000053             30 RANGE(I)  = 0.0                                             POUT
000054          C                                                                 POUT
000055                DO 40 I = 1,NINP2                                           POUT
000056             40 INPVL2(I) = 0                                               POUT
000057          C                                                                 POUT
000058                NPAGE = 0                                                   POUT
```

```
000059            NLINES = 0                                                    POUT
000060            MAXLIN = LNMAX                                                POUT
000061            KOUT = IOUT                                                   POUT
000062          C                                                               POUT
000063            DO 50 I = 1,3                                                 POUT
000064            CTAMER(I) = AMCVT(I)                                          POUT
000065         50 CTMTRC(I) = XMCVT(I)                                          POUT
000066          C                                                               POUT
000067            DO 60 I = 1,4                                                 POUT
000068            J = I+12                                                      POUT
000069         60 HEADER(J) = RHDR(I)                                           POUT
000070          C                                                               POUT
000071          C                                                               POUT
000072          C *** READ INPUT CONTROL CARDS                                  POUT
000073          C                                                               POUT
000074          C CARD 1 - HEADER CARD                                          POUT
000075          C                                                               POUT
000076            READ (IN,5000) (IHEADR(I),I=1,12),INLBL,INCRD                 POUT
000077          C                                                               POUT
000078            DO 90 I = 1,12                                                POUT
000079         90 HEADER(I) = IHEADR(I)                                         POUT
000080          C                                                               POUT
000081            IF (INLBL.EQ.LABEL.AND.INCRD.EQ.NCARD(1)) GO TO 100           POUT
000082          C                                                               POUT
000083          C     LABEL ERROR ON HEADER CARD                                POUT
000084          C                                                               POUT
000085            INERRS = .TRUE.                                               POUT
000086          C                                                               POUT
000087            CALL PAGHDR(3)                                                POUT
000088            WRITE (KOUT,6000) NCARD(1),INLBL,INCRD                        POUT
000089          C                                                               POUT
000090          C CARD 2 - DATA CARD 1                                          POUT
000091          C                                                               POUT
000092        100 READ (IN,5010) RPTYPE,AUNITS,(RANGE(I),I=1,7),INLBL,INCRD     POUT
000093          C                                                               POUT
000094            IF (INLBL.EQ.LABEL.AND.INCRD.EQ.NCARD(2)) GO TO 110           POUT
000095          C     LABEL ERROR                                               POUT
000096          C                                                               POUT
000097            INERRS = .TRUE.                                               POUT
000098          C                                                               POUT
000099            CALL PAGHDR(3)                                                POUT
000100            WRITE (KOUT,6000) NCARD(2),INLBL,INCRD                        POUT
000101          C                                                               POUT
000102        110 IF (RPTYPE.GT.0.AND.RPTYPE.LT.MRPTYP+1) GO TO 115             POUT
000103          C                                                               POUT
000104          C     RPTYPE VALUE OUT OF RANGE                                 POUT
000105          C                                                               POUT
000106            INERRS = .TRUE.                                               POUT
000107            CALL PAGHDR(2)                                                POUT
000108            WRITE (KOUT,6010) RPTYPE                                      POUT
000109          C                                                               POUT
000110        115 IF (RPTYPE.EQ.1.OR.RPTYPE.EQ.4) GO TO 200                     POUT
000111          C                                                               POUT
000112            DO 120 I = 1,NRGE                                             POUT
000113            IF (RANGE(I).NE.0.0) NOMRNG = 0                               POUT
000114        120 CONTINUE                                                      POUT
000115          C                                                               POUT
000116            IF (NOMRNG.EQ.0) GO TO 125                                    POUT
000117          C                                                               POUT
000118            DO 122 I = 1,NRGE                                             POUT
```

```
000119          122 RANGE(I) = RNGNML(I)                                        POUT
000120              GO TO 200                                                   POUT
000121        C                                                                 POUT
000122        C     CHECK FOR INTERVAL RANGE ERRORS                             POUT
000123        C                                                                 POUT
000124          125 CONTINUE                                                    POUT
000125              A = 0.                                                      POUT
000126              B = 0.                                                      POUT
000127              C = 0.                                                      POUT
000128              D = 0.                                                      POUT
000129              IF (START.LT.BREAK1) GO TO 130                              POUT
000130        C                                                                 POUT
000131              INERRS = .TRUE.                                             POUT
000132              CALL PAGHDR(3)                                              POUT
000133              WRITE (KOUT,6020) START,BREAK1                              POUT
000134        C                                                                 POUT
000135          130 A = BREAK1-START                                            POUT
000136              D = D +A/INTVL1                                             POUT
000137              IF (AMOD(A,INTVL1).EQ.0) GO TO 140                          POUT
000138        C                                                                 POUT
000139              INERRS = .TRUE.                                             POUT
000140              CALL PAGHDR(3)                                              POUT
000141              WRITE (KOUT,6050) A,INTVL1                                  POUT
000142        C                                                                 POUT
000143          140 IF (INTVL2.EQ.0..AND.BREAK2.EQ.0.) GO TO 180                POUT
000144        C                                                                 POUT
000145              IF (BREAK1.LT.BREAK2) GO TO 150                             POUT
000146        C                                                                 POUT
000147              INERRS = .TRUE.                                             POUT
000148              CALL PAGHDR(3)                                              POUT
000149              WRITE (KOUT,6030) BREAK1,BREAK2                             POUT
000150        C                                                                 POUT
000151          150 B = BREAK2-BREAK1                                           POUT
000152              D = D+B/INTVL2                                              POUT
000153              IF (AMOD(B,INTVL2).EQ.0) GO TO 160                          POUT
000154        C                                                                 POUT
000155              INERRS = .TRUE.                                             POUT
000156              CALL PAGHDR(3)                                              POUT
000157              WRITE (KOUT,6060) B,INTVL2                                  POUT
000158        C                                                                 POUT
000159          160 IF (INTVL3.EQ.0..AND.STOP.EQ.0) GO TO 180                   POUT
000160        C                                                                 POUT
000161              IF (BREAK2.LT.STOP) GO TO 170                               POUT
000162        C                                                                 POUT
000163              INERRS = .TRUE.                                             POUT
000164              CALL PAGHDR(3)                                              POUT
000165              WRITE (KOUT,6040) BREAK2,STOP                               POUT
000166        C                                                                 POUT
000167          170 C = STOP-BREAK2                                             POUT
000168              D = D+C/INTVL3                                              POUT
000169              IF (AMOD(C,INTVL3).EQ.0) GO TO 180                          POUT
000170        C                                                                 POUT
000171              INERRS = .TRUE.                                             POUT
000172              CALL PAGHDR(3)                                              POUT
000173              WRITE (KOUT,6070) C,INTVL3                                  POUT
000174        C                                                                 POUT
000175          180 CONTINUE                                                    POUT
000176              D = D+3                                                     POUT
000177              IF (D.LE.XNRNGE) GO TO 200                                  POUT
000178        C                                                                 POUT
```

```
000179          INERRS = .TRUE.                                                 POUT
000180          CALL PAGHDR(3)                                                  POUT
000181          WRITE (KOUT,6080) XMXRGE,D                                      POUT
000182    C                                                                     POUT
000183    C CARD 3 - DATA CARD 2                                                POUT
000184    C                                                                     POUT
000185      200 READ (IN,5020) (PARMTR(I),I=1,5),(ICASIN(I),I=1,5),             POUT
000186         1LEVEL,ICSESS,INLBL,INCRD                                        MOD3
000187    C                                                                     POUT
000188          IF (INLBL.EQ.LABEL.AND.INCRD.EQ.NCARD(3)) GO TO 205             POUT
000189    C                                                                     POUT
000190          INERRS = .TRUE.                                                 POUT
000191          CALL PAGHDR(3)                                                  POUT
000192          WRITE (KOUT,6000) NCARD(3),INLBL,INCRD                          POUT
000193    C                                                                     POUT
000194      205 IF (RPTYPE.EQ.1.OR.RPTYPE.EQ.4) GO TO 220                       POUT
000195    C                                                                     POUT
000196          SETALL = 0                                                      POUT
000197          DO 210 I =1,5                                                   POUT
000198          IF (PARMTR(I).NE.0) SETALL=1                                    POUT
000199      210 CONTINUE                                                        POUT
000200    C                                                                     POUT
000201          IF (SETALL.EQ.1.) GO TO 218                                     POUT
000202    C                                                                     POUT
000203          DO 215 I = 1,5                                                  POUT
000204          IF (I.EQ.5.AND.RPTYPE.EQ.3) GO TO 218                           POUT
000205      215 PARMTR(I) = 1                                                   POUT
000206    C                                                                     POUT
000207      218 IF (RPTYPE.EQ.3) GO TO 220                                      POUT
000208    C                                                                     POUT
000209          IF (LEVEL.EQ.0) LEVEL = JLEVEL                                  POUT
000210    C                                                                     POUT
000211      220 DO 240 I = 1,5                                                  POUT
000212    C                                                                     POUT
000213          IF (I.EQ.5.AND.RPTYPE.EQ.3) GO TO 250                           POUT
000214    C                                                                     POUT
000215          IF (RPTYPE.GT.1.OR.RPTYPE.LT.4) GO TO 230                       POUT
000216    C                                                                     POUT
000217          IF (ICASIN(I).NE.0) GO TO 250                                   POUT
000218    C                                                                     POUT
000219          INERRS = .TRUE.                                                 POUT
000220          CALL PAGHDR(3)                                                  POUT
000221          WRITE (KOUT,6090) I,I,ICASIN(I),RPTYPE                          POUT
000222    C                                                                     POUT
000223          GO TO 250                                                       POUT
000224    C                                                                     POUT
000225      230 IF (PARMTR(I).EQ.0) GO TO 240                                   POUT
000226    C                                                                     POUT
000227          IF (ICASIN(I).NE.0) GO TO 240                                   POUT
000228    C                                                                     POUT
000229          INERRS = .TRUE.                                                 POUT
000230          CALL PAGHDR(3)                                                  POUT
000231          WRITE (KOUT,6100) I,I,ICASIN(I),RPTYPE,I,PARMTR(I)              POUT
000232    C                                                                     POUT
000233      240 CONTINUE                                                        POUT
000234    C                                                                     POUT
000235    C CARDS 4 & 5 - DATA CARDS 3 & 4                                      POUT
000236    C                                                                     POUT
000237      250 READ (IN,5030) (HIO*D(I),I=1,4),(SPRTY(I),I=1,4)                POUT
000238         *,                  ((IPRO(J,I),J=1,3),I=1,6),INLBL,INCRD      POUT
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000239      C                                                                 POUT
000240            IF (INLBL.EQ.LABEL.AND.INCRD.EQ.NCARD(4)) GO TO 260         POUT
000241      C                                                                 POUT
000242            INERRS = .TRUE.                                             POUT
000243            CALL PAGHDR(3)                                              POUT
000244            WRITE (KOUT,6000) NCARD(4),INLBL,INCRD                      POUT
000245      C                                                                 POUT
000246        260 READ (IN,5040) ((IPRD(J,I),J=1,3),I=7,14),INLBL,INCRD       POUT
000247      C                                                                 POUT
000248            IF (INLBL.EQ.LABEL.AND.INCRD.EQ.NCARD(5)) GO TO 270         POUT
000249      C                                                                 POUT
000250            INERRS = .TRUE.                                             POUT
000251            CALL PAGHDR(3)                                              POUT
000252            WRITE (KOUT,6000) NCARD(5),INLBL,INCRD                      POUT
000253      C                                                                 POUT
000254      C *** ECHO PRINT CARD IMAGES                                      POUT
000255      C                                                                 MODF1
000256        270 CONTINUE                                                    POUT
000257      C                                                                 MODF1
000258            CALL PAGHDR(-15)                                            POUT
000259      C                                                                 POUT
000260            WRITE (KOUT,5500) RPTYPE,(PARMTR(I),I=1,5),(ICASIN(I),I=1,5),LEVELPOUT
000261           1,ICSESN                                                     MOD3
000262            WRITE (KOUT,5510) (RANGE(I),I=1,7)                          POUT
000263            WRITE (KOUT,5520) (BIOWD(I),I=1,4),(WPRTY(I),I=1,4)         POUT
000264            WRITE (KOUT,5530) ((IPWD(J,I),J=1,3),I=1,14)                POUT
000265      C                                                                 POUT
000266      C *** TERMINATE RUN IF ERRORS FOUND IN INPUT DATA                 POUT
000267      C                                                                 POUT
000268            IF (.NOT.INERRS) GO TO 280                                  POUT
000269      C                                                                 POUT
000270            CALL PAGHDR(2)                                              POUT
000271            WRITE (KOUT,6110)                                           POUT
000272            GO TO 500                                                   POUT
000273      C                                                                 POUT
000274      C *** RUN CASE                                                    POUT
000275      C                                                                 POUT
000276        280 GO TO (300,350,400,450,500),RPTYPE,                         POUT
000277      C                                                                 POUT
000278      C *** SUBSTRATA REFERENCE DATA REPORT                             POUT
000279      C                                                                 POUT
000280        300 CALL SUBREF                                                 POUT
000281      C                                                                 POUT
000282            GO TO 500                                                   POUT
000283      C                                                                 POUT
000284      C *** POPULATION REPORTS                                          POUT
000285      C                                                                 POUT
000286        350 CALL POPDRV                                                 POUT
000287      C                                                                 POUT
000288            GO TO 500                                                   POUT
000289      C                                                                 POUT
000290      C *** MONTE CARLO REPORTS                                         POUT
000291      C                                                                 POUT
000292        400 CALL MCHIST                                                 POUT
000293      C                                                                 POUT
000294            GO TO 500                                                   POUT
000295      C                                                                 POUT
000296      C *** CALL FREQAC                                                 POUT
000297      C                                                                 POUT
000298        450 CALL FREQAC                                                 POUT
```

```
000299      C                                                                 POUT
000300            GO TO 500                                                   POUT
000301      C                                                                 POUT
000302      C *** FINISHED - STOP RUN                                         POUT
000303      C                                                                 POUT
000304        500 STOP                                                        POUT
000305      C *** INPUT FORMATS                                               POUT
000306       5000 FORMAT (12A6,2X,A4,A2)                                      POUT
000307       5010 FORMAT (I2,I1,7F6.1,29X,A4,A2)                              POUT
000308       5020 FORMAT(2A,5I1,5X,9I4,2X,I1,I4,35X,A4,A2)                    MOD3
000309       5030 FORMAT (2(4I1,2X),6(3I2,1X),20X,A4,A2)                      MODF2
000310       5040 FORMAT (8(3I2,1X),18X,A4,A2)                                MODF2
000311       5500 FORMAT(7X,14HPPTYPE   PARMT,5X,6HICASIN,22X,5HLEVEL,2X,6HICSESH/  MOD3
000312          19X,I2,4X,5I2,3A,9I5,4X,I5/1H0)                               MOD3
000313       5510 FORMAT (49H INTERVAL  START  INTVL1  BREAK1  INTVL2  BREAK2 ,  POUT
000314           *        15H INTVL3    STOP/10X,7(F6.1,2X)/1H0)               MODF1
000315       5520 FORMAT (7X,14HBICND    WPRTY/6X,4I2,2X,4I2/1H0)               MODF
000316       5530 FORMAT (/X,8HIPRO//(7X,7(3I2,3X)))                          POUT
000317      C *** ERROR MESSAGE FORMATS                                       POUT
000318       6000 FORMAT (47H *** IMPROPER LABEL AND SEQUENCE NUMBER ON POUT,  POUT
000319           *        18H CONTROL CARD NO. ,A2,1H./20H LABEL AND SEQ. NO.=,  POUT
000320           *        A4,A2/1H )                                           POUT
000321       6010 FORMAT (39H *** RPTYPE VALUE OUT OF RANGE. PPTYPE=,I2/1H )  POUT
000322       6020 FORMAT (43H *** START VALUE GREATER THAN BREAK1 VALUE./      POUT
000323           *        7H START=,F6.1,8H BREAK1=,F6.1/1H )                  POUT
000324       6030 FORMAT (44H *** BREAK1 VALUE GREATER THAN BREAK2 VALUE./     POUT
000325           *        8H BREAK1=,F6.1,8H BREAK2=,F6.1/1H )                 POUT
000326       6040 FORMAT (42H *** BREAK2 VALUE GREATER THAN STOP VALUE./       POUT
000327           *        8H BREAK2=,F6.1,6H STOP=,F6.1/1H )                   POUT
000328       6050 FORMAT (50H *** THE SEGMENT BETWEEN START AND BREAK1 WILL NOT,  POUT
000329           *        36H DIVIDE INTO EQUAL INTVL1 INTERVALS./             POUT
000330           *        14H BREAK1-START=,F6.1,8H INTVL1=,F6.1/1H )          POUT
000331       6060 FORMAT (51H *** THE SEGMENT BETWEEN BREAK1 AND BREAK2 WILL NOT,  POUT
000332           *        36H DIVIDE INTO EQUAL INTVL2 INTERVALS./             POUT
000333           *        15H BREAK2-BREAK1=,F6.1,8H INTVL2=,F6.1/1H )         POUT
000334       6070 FORMAT (49H *** THE SEGMENT BETWEEN BREAK2 AND STOP WILL NOT,  POUT
000335           *        36H DIVIDE INTO EQUAL INTVL3 INTERVALS./             POUT
000336           *        13H STOP-BREAK2=,F6.1,8H INTVL3=,F6.1/1H )           POUT
000337       6080 FORMAT (53H *** REQUESTED NO. OF HISTOGRAM INTERVALS EXCEEDS THE,  POUT
000338           *        11H MAXIMUM OF,F3.0/32H REQUESTED NO. OF INTERVALS THAT,  POUT
000339           *        29H WOULD BE GENERATED=,F11.0/1H )                   MODF1
000340       6090 FORMAT (12H *** ICASIN(,I1,32H) VALUE MISSING FOR REPORT TYPE./  POUT
000341           *        8H ICASIN(,I1,2H)=,I4,8H RPTYPE=,I2/1H )             POUT
000342       6100 FORMAT (12H *** ICASIN(,I1,32H) VALUE MISSING FOR REPORT TYPE./  POUT
000343           *        8H ICASIN(,I1,2H)=,I4,8H RPTYPE=,I2,8H PARMTR(,I1,2H)=,  POUT
000344           *        I1/1H )                                              POUT
000345       6110 FORMAT (/55H *** RUN TERMINATED BECAUSE OF INPUT DATA ERRORS. ***)  POUT
000346      C                                                                 POUT
000347      C                                                                 POUT
000348            END                                                         POUT
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
* EL, PUNITS,1,760511, 59144    , 1
000001                SUBROUTINE PUNITS (FACTOR,IUNITS)                                  PUNITS
000002          C                                                                        PUNITS
000003          C *** THIS ROUTINE SETS-UP THE PROPER PRINT AND CONVERSION UNITS         PUNITS
000004          C *** FOR THE POPULATION AND MONTE CARLO REPORTS                         PUNITS
000005          C                                                                        PUNITS
000006          C *** COMMON STORAGE                                                     PUNITS
000007          C                                                                        PUNITS
000008                COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                             CARDIN
000009               *,              ICASIP(5),LEVEL,JPARM,IERR                        CARDIN
000010                INTEGER RPTYPE,AUNITS,PARMTR                                      CARDIN
000011          C                                                                        PUNITS
000012                COMMON /CONVRT/ CTAMER(3),CTMTRC(3)                                CONVRT
000013          C                                                                        PUNITS
000014          C                                                                        PUNITS
000015          C *** LOCAL STORAGE                                                      PUNITS
000016          C                                                                        PUNITS
000017                DIMENSION IUNITS(1)                                                PUNITS
000018                DIMENSION JFAC(10),IAMERU(9),IMTRCU(9)                             PUNITS
000019          C                                                                        PUNITS
000020                DATA (JFAC(I),I=1,10) /1,1,2,1,2,1,2,3,3,0/                        PUNITS
000021                DATA (IAMERU(I),I=1,9) /6H10**4 ,6HACRES ,6H      ,                PUNITS
000022               *                       6H10**6 ,6HBUSHEL,6HS     ,               PUNITS
000023               *                       6HPER CE,6HNT    ,6H      /               PUNITS
000024                DATA (IMTRCU(I),I=1,9) /6H10**3 ,6HHECTAR,6HES    ,               PUNITS
000025               *                       6H10**3 ,6HMETRIC,6H TONS ,               PUNITS
000026               *                       6HPER CE,6HNT    ,6H      /               PUNITS
000027          C                                                                        PUNITS
000028                IFAC = 0                                                           PUNITS
000029                IF (RPTYPE.EQ.3) IFAC = IFAC+5                                     PUNITS
000030                IFAC = IFAC+JPARM                                                  PUNITS
000031          C                                                                        PUNITS
000032                NFAC = JFAC(IFAC)                                                  PUNITS
000033          C                                                                        PUNITS
000034                IF (AUNITS.NE.0) GO TO 50                                          PUNITS
000035          C                                                                        PUNITS
000036          C *** AMERICAN UNITS                                                     PUNITS
000037          C                                                                        PUNITS
000038                FACTOR = 1.0                                                       PUNITS
000039                IF (RPTYPE.EQ.3.AND.NFAC.EQ.3) GO TO 20                            PUNITS
000040                FACTOR = CTAMER(NFAC)                                              PUNITS
000041          C                                                                        PUNITS
000042             20 NFAC = (NFAC-1)*3                                                  PUNITS
000043          C                                                                        PUNITS
000044                DO 30 I = 1,3                                                      PUNITS
000045                J = NFAC+I                                                         PUNITS
000046             30 IUNITS(I) = IAMERU(J)                                              PUNITS
000047          C                                                                        PUNITS
000048                GO TO 100                                                          PUNITS
000049          C                                                                        PUNITS
000050          C *** METRIC UNITS                                                       PUNITS
000051          C                                                                        PUNITS
000052             50 FACTOR = 1.0                                                       PUNITS
000053                IF (RPTYPE.EQ.3.AND.NFAC.EQ.3) GO TO 70                            PUNITS
000054                FACTOR = CTMTRC(NFAC)                                              PUNITS
000055          C                                                                        PUNITS
000056             70 NFAC = (NFAC-1)*3                                                  PUNITS
000057          C                                                                        PUNITS
000058                DO 80 I = 1,3                                                      PUNITS
```

```
000059                  J = MFAC+I                                      PUNITS
000060               80 IUNITS(I) = IMTRU(J)                            PUNITS
000061                  C                                               PUNITS
000062                  C                                               PUNITS
000063                  C  100 RETURN                                   MODF
000064                  C                                               PUNITS
000065                     END                                          PUNITS
```

```
* ECF RANACF,1,760513, 61018   , 1

000001                SUBROUTINE RANACF (IFILE,IREC,BUF,N,IX,L,IOPT)                    RANACF
000002          C        STANDARDIZED RANDOM ACCESS I/O ROUTINE                         RANACF
000003          C        OPENS, CLOSES, READS, AND WRITES A RANDOM ACCESS FILE USING    RANACF
000004          C        INSTALLATION DEPENDENT RANDOM ACCESS ROUTINES                  RANACF
000005          C        ON THE CDC 6600 COMPUTING SYSTEM AT TRW THE OPENMS,CLOSEMS,    RANACF
000006          C        READMS, AND WRITMS ENTRY POINTS ARE USED TO ACTUALLY PERFORM   RANACF
000007          C        THE RANDOM ACCESS I/O OPERATIONS                               RANACF
000008          C                                                                       RANACF
000009          C        INPUT PARAMETERS ...                                           RANACF
000010          C        IFILE = LOGICAL UNIT NUMBER OF THE RANDOM ACCESS FILE,         RANACF
000011          C        IREC  = RECORD NUMBER TO BE READ OR WRITTEN,                   RANACF
000012          C        BUF   = ARRAY OF LENGTH N TO CONTAIN THE RECORD,               RANACF
000013          C     (BUF IS INPUT IF IOPT = 2 AND IS OUTPUT IF IOPT = 1)              RANACF
000014          C        N     = LENGTH OF RECORD TO BE READ OR WRITTEN,                RANACF
000015          C        IX    = ARRAY OF DIMENSION L CONTAINING THE INDEX RECORD,      RANACF
000016          C        L     = LENGTH OF INDEX,                                       RANACF
000017          C        IOPT  = ENTRY POINT OPTION ..                                  RANACF
000018          C                = 0 TO OPEN THE FILE,                                  RANACF
000019          C                = 1 TO READ THE FILE,                                  RANACF
000020          C                = 2 TO WRITE THE FILE,                                 RANACF
000021          C                = -1 TO CLOSE THE FILE.                                RANACF
000022          C                                                                       RANACF
000023                DIMENSION  BUF(N),IX(L)                                           RANACF
000024          C                                                                       RANACF
000025          C        LINKAGE ... CALL RANACF (IFILE,IREC,BUF,N,IX,L,IOPT)           RANACF
000026          C        RANACF IS CALLED FROM INPCHK, CAS, WRAPUP                      RANACF
000027          C                                                                       RANACF
000028          C        SUBROUTINES USED ... OPENMS,CLOSEMS, READMS, WRITMS            RANACF
000029          C                                                                       RANACF
000030          C                                                                       RANACF
000031          C ***************************************************************** RANACF
000032          C                                                                       RANACF
000033                IF ( IOPT .EQ. -1 )  GO TO 400                                    RANACF
000034                IF ( IOPT - 1 )  100,200,300                                      RANACF
000035          C        IOPT = 0. OPEN FILE                                            RANACF
000036            100 DEFINE FILE IFILE(L,N,U,IDUM)
000037                GO TO 900                                                         RANACF
000038          C        IOPT = 1. READ RECORD IREC                                     RANACF
000039            200 READ(IFILE'IREC,ERR=600)(BUF(I),I=1,N)                                    *NEW
000040                GO TO 900                                                         RANACF  **-1
000041          C        IOPT = 2. WRITE RECORD IREC                                    RANACF
000042            300 WRITE(IFILE'IREC,ERR=600)(BUF(I),I=1,N)                                   *NEW
000043                GO TO 900                                                         RANACF  **-1
000044          C        IOPT = -1. CLOSE FILE                                          RANACF
000045            400 CALL CLOSE(IFILE,0)                                                       *NEW
000046                GO TO 900                                                                 **-2
000047            600 WRITE(6,901)IFILE,IREC
000048                STOP
000049            901 FORMAT(1H0,91H *** AN IRRECOVERABLE I/O ERROR HAS OCCURRED ON READ
000050               1IG A RECORD FROM A DIRECT ACCESS FILE /
000051               210X,8H FILE=,I5,8H RECORD=,I5,31H THE JOB IS BEING ABANDONED ***)
000052             900  RETURN                                                        RANACF
000053                END                                                            RANACF
```

```
C PEPRT1,1,760511, 59129   , 1
000001                SUBROUTINE PEPRT1                                          REPRT1
000002          C                                                                REPRT1
000003          C THIS ROUTINE CONTROLS AND PRINTS THE SUBSTRATA REFERENCE DATA REPORT  REPRT1
000004          C                                                                REPRT1
000005          C *** COMMON STORAGE                                             REPRT1
000006          C                                                                REPRT1
000007                COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                     CARDIN
000008               *,              ICASIN(5),LEVEL,JPARM,IERR                  CARDIN
000009                INTEGER RPTYPE,AUNITS,PARMTR                               CARDIN
000010          C                                                                REPRT1
000011                COMMON /PRTCTL/ MPAGE,NLINES,MAXLIN,KOUT,HEADER(16)        PRTCTL
000012                INTEGER HEADER                                             PRTCTL
000013          C                                                                REPRT1
000014                COMMON /CONVRT/ CTAMER(3),CTMTRC(3)                        CONVRT
000015          C                                                                REPRT1
000016                COMMON /PRNTID/ CNTRY,REGION,ZONE,STRATA,ICASEN            PRNTID
000017               *,               FILENO,FILNAM(2)                           PRNTID
000018               *,               ITOTYP,NAMFMT                              PRNTID
000019          C                                                                PRNTID
000020                INTEGER CNTRY,REGION,ZONE,STRATA,FILENO,FILNAM             MODF
000021          C                                                                REPRT1
000022                COMMON /SUBVAL/ AWA,TWA,NSUBGP(3),NSEGGP(2),XINBS          SUBVAL
000023               *,               XSIMBS,XINCV,XSIMCV                        SUBVAL
000024          C                                                                REPRT1
000025          C                                                                REPRT1
000026          C *** LOCAL STORAGE                                              REPRT1
000027          C                                                                REPRT1
000028                DIMENSION FACTOR(3),IAMERU(4),IMTRCU(4),IUNITS(4)          REPRT1
000029          C                                                                REPRT1
000030               *,         UNITS(3)                                         MODF
000031                DATA JFIRST /0/, MPAGE /0/                                 REPRT1
000032                DATA (IMTRCU(I),I=1,4) /6H(THOUS,6HAND HE,6HCTARES,6H)     /  REPRT1
000033                DATA (IAMERU(I),I=1,4) /6H(TEN T,6HHOUSAN,6HD ACRE,6HS)    /  REPRT1
000034          C     CHECK FOR FIRST TIME THROUGH                               REPRT1
000035          C                                                                REPRT1
000036                IF (JFIRST.NE.0) GO TO 100                                 REPRT1
000037          C                                                                REPRT1
000038          C SAVE FIRST PAGE NUMBER                                         REPRT1
000039          C                                                                REPRT1
000040                MPAGE = MPAGE+1                                            REPRT1
000041                JFIRST = MPAGE                                             REPRT1
000042          C                                                                REPRT1
000043                IF (AUNITS.NE.0) GO TO 50                                  REPRT1
000044          C                                                                REPRT1
000045          C AMERICAN UNITS                                                 REPRT1
000046          C                                                                REPRT1
000047                DO 30 I = 1,4                                              REPRT1
000048                IF (I.GT.3) GO TO 30                                       MODF
000049                UNITS(I) = CTAMER(I)                                       MODF
000050             30 IUNITS(I) = IAMERU(I)                                      REPRT1
000051                GO TO 80                                                   REPRT1
000052          C                                                                REPRT1
000053          C METRIC UNITS                                                   REPRT1
000054          C                                                                REPRT1
000055             50 DO 60 I = 1,4                                              REPRT1
000056                IF (I.GT.3) GO TO 60                                       MODF
000057                UNITS(I) = CTMTRC(I)                                       MODF
000058             60 IUNITS(I) = IMTRCU(I)                                      REPRT1
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000059          C                                                               REPRT1
000060          C PRINT HEADINGS                                                REPRT1
000061          C                                                               REPRT1
000062            80 CALL PAGHDR(-11)                                           REPRT1
000063          C                                                               REPRT1
000064               WRITE (KOUT,1000) CNTRY,ICASEN,(IUNITS(I),I=1,4)           REPRT1
000065            100 GO TO (150,200,250,300),ITUTYP                            REPRT1
000066          C                                                               REPRT1
000067          C PRINT STRATA LINE                                             REPRT1
000068          C                                                               REPRT1
000069            150 CALL PAGHDR(1)                                            REPRT1
000070          C                                                               REPRT1
000071          C CHECK FOR CHANGE IN PAGE NUMBER                               REPRT1
000072          C                                                               REPRT1
000073               IF (NPAGE.EQ.MPAGE) GO TO 180                              REPRT1
000074               MPAGE = NPAGE                                              REPRT1
000075               CALL PAGHDR(11)                                            REPRT1
000076               WRITE (KOUT,1000) CNTRY,ICASEN,(IUNITS(I),I=1,4)           REPRT1
000077          C                                                               REPRT1
000078            180 HMAX = HMA*UNITS(1)                                       MODF
000079               TMAX = TMA*UNITS(1)                                        MODF
000080               WRITE (KOUT,1010) STRATA,HMAX,TMAX,(NSUBGP(I),I=1,3)       MODF
000081              *,                 (NSEGGP(I),I=1,2),XINBS,XSIMBS,XINCV,XSIMCV  REPRT1
000082          C                                                               REPRT1
000083               GO TO 500                                                  REPRT1
000084          C                                                               REPRT1
000085          C PRINT ZONE LINE                                               REPRT1
000086          C                                                               REPRT1
000087            200 CALL PAGHDR(5)                                            REPRT1
000088          C                                                               REPRT1
000089               IF (NPAGE.EQ.MPAGE) GO TO 220                              REPRT1
000090          C                                                               REPRT1
000091               MPAGE = NPAGE                                              REPRT1
000092               CALL PAGHDR(11)                                            REPRT1
000093               WRITE (KOUT,1000) CNTRY,ICASEN,(IUNITS(I),I=1,4)           REPRT1
000094          C                                                               REPRT1
000095            220 HMAX = HMA*UNITS(1)                                       MODF
000096               TMAX = TMA*UNITS(1)                                        MODF
000097               WRITE(KOUT,1030)                                           MOD1
000098               WRITE(KOUT,2000)                                           MOD1
000099               WRITE (KOUT,1010) ZONE,HMAX,TMAX,(NSUBGP(I),I=1,3)         MODF
000100              *,                 (NSEGGP(I),I=1,2),XINBS,XSIMBS,XINCV,XSIMCV  REPRT1
000101               WRITE (KOUT,1030)                                          MODF
000102               WRITE (KOUT,1040)                                          REPRT1
000103          C                                                               REPRT1
000104               GO TO 500                                                  REPRT1
000105          C                                                               REPRT1
000106          C PRINT REGION LINE                                             REPRT1
000107          C                                                               REPRT1
000108            250 CALL PAGHDR(5)                                            REPRT1
000109          C                                                               REPRT1
000110               IF (NPAGE.EQ.MPAGE) GO TO 280                              REPRT1
000111          C                                                               REPRT1
000112               MPAGE = NPAGE                                              REPRT1
000113               CALL PAGHDR(11)                                            REPRT1
000114               WRITE (KOUT,1000) CNTRY,ICASEN,(IUNITS(I),I=1,4)           REPRT1
000115          C                                                               REPRT1
000116            280 HMAX = HMA*UNITS(1)                                       MODF
000117               TMAX = TMA*UNITS(1)                                        MODF
000118               WRITE (KOUT,1040)                                          REPRT1
```

```
000119            WRITE(KOUT,2001)                                              MOD1
000120            WRITE (KOUT,1010) REGION,HWAX,TWAX,(NSUBGP(I),I=1,3)          MODF
000121           *,                 (NSEGGP(I),I=1,2),XINBS,XSIMBS,XINCV,XSIMCV REPRT1
000122            WRITE (KOUT,1030)                                             MODF
000123            WRITE (KOUT,1040)                                             REPRT1
000124      C                                                                   REPRT1
000125            GO TO 500                                                     REPRT1
000126      C                                                                   REPRT1
000127      C PRINT COUNTRY LINE                                                REPRT1
000128      C                                                                   REPRT1
000129        300 CALL PAGHDR(5)                                                REPRT1
000130      C                                                                   REPRT1
000131            IF (NPAGE.EQ.IPAGE) GO TO 320                                 REPRT1
000132      C                                                                   REPRT1
000133            CALL PAGHDR(11)                                               REPRT1
000134            WRITE (KOUT,1000) CNTRY,ICASEN,(IUNITS(I),I=1,4)              REPRT1
000135      C                                                                   REPRT1
000136        320 HWAX = HWA*UNITS(1)                                           MODF
000137            TWAX = TWA*UNITS(1)                                           MODF
000138            WRITE (KOUT,1040)                                             REPRT1
000139            WRITE (KOUT,1030)                                             REPRT1
000140            WRITE (KOUT,1040)                                             REPRT1
000141            WRITE(KOUT,2002)                                              MOD1
000142            WRITE (KOUT,1020) CNTRY,HWAX,TWAX,(NSUBGP(I),I=1,3)           MODF
000143           *,                 (NSEGGP(I),I=1,2),XINBS,XSIMBS,XINCV,XSIMCV REPRT1
000144      C                                                                   REPRT1
000145      C                                                                   REPRT1
000146        500 RETURN                                                        REPRT1
000147      C                                                                   REPRT1
000148      C *** PRINT FORMATS                                                 REPRT1
000149       1000 FORMAT (33X,45HS U B S T R A T A  R E F E R E N C E  D A T A/1H / REPRT1
000150           *        12H   COUNTRY  ,A4,5X,11HCASE NUMBER,I6/1H /          REPRT1
000151           *        30X,7HA R E A/23X,4A6/1H /35X,6HNO. IN,8X,6HNO. OF/   REPRT1
000152           *        30X, 9HSUBSTRATA,5X,8HSEGMENTS,5X,19HINPUT    SIMULATION, REPRT1
000153           *         5X,19HINPUT    SIMULATION/14X,4HFIRST,6X,4HTRUE,     REPRT1
000154           *      7X, 5HGROUP,6X,8HOF GROUP,5X,4HBIAS,7X,4HBIAS,10X,3HCV2,
000155           *      8X,3HCV1/9H   STRATA,6X,2HWA,8X,2HWA,7X,11H1    2    3,
000156           *      5X,16H1    2  FCT HIST,4X,8HFCT HIST,4X,8HFCT TRUE,
000157           *      4X,8HFCT TRUE)
000158       1010 FORMAT (I8,F11.1,F10.1,I6,2I5,I6,I5,2F11.4,2X,2F11.4)          REPRT1
000159       1020 FORMAT (4X,A4,F11.1,F10.1,I6,2I5,I6,I5,2F11.4,2X,2F11.4)       REPRT1
000160       1030 FORMAT (4X,17(6H******),2H**)                                  REPRT1
000161       1040 FORMAT (1H )                                                   REPRT1
000162       2000 FORMAT(8H    ZONE)                                             MOD1
000163       2001 FORMAT(8H     REG)                                           MOD1
000164       2002 FORMAT(8H    COUN)                                           MOD1
000165      C                                                                   REPRT1
000166            END                                                           REPRT1
```

```
* ...T REPRT2,1,760513, 61017   , 1
000001            SUBROUTINE REPRT2                                                    REPRT2
000002      C                                                                          REPRT2
000003      C *** ROUTINE CONTROLS AND PRINTS THE HISTOGRAM DATA REPORTS               REPRT2
000004      C                                                                          REPRT2
000005      C                                                                          REPRT2
000006      C *** COMMON STORAGE                                                       REPRT2
000007      C                                                                          REPRT2
000008            COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                               CARDIN
000009           *,              ICASIN(5),LEVEL,JPARM,IERR                            CARDIN
000010            INTEGER RPTYPE,AUNITS,PARMTR                                         CARDIN
000011            COMMON /PRTCTL/ NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)                  PRTCTL
000012            INTEGER HEADER                                                       PRTCTL
000013      C                                                                          REPRT2
000014            COMMON /PRNTID/ CNTRY,REGION,ZONE,STRATA,ICASEN                      PRNTID
000015           *,               FILENO,FILNAM(2)                                    PRNTID
000016           *,               ITOTYP,NAMFMT                                       PRNTID
000017      C                                                                          PRNTID
000018            INTEGER CNTRY,REGION,ZONE,STRATA,FILENO,FILNAM                       MODF
000019      C                                                                          REPRT2
000020            COMMON /HSTOGM/ AMVAL,STDDEV,XMPSUM,SQFSUM                           HSTOGM
000021           *,               NRANGE,IRANGE(51),NTOTL,RANGES(52)                  HSTOGM
000022           *,               MINVAL,MAXVAL,DATPRD(3),NZRNG,REF                   HSTOGM
000023      C                                                                          HSTOGM
000024            REAL    MINVAL,MAXVAL                                              HSTOGM
000025            INTEGER DATPRD                                                     HSTOGM
000026      C                                                                          REPRT2
000027      C *** LOCAL STORAGE                                                        REPRT2
000028      C                                                                          REPRT2
000029            DIMENSION IUNITS(3)                                                  MODF
000030            DIMENSION FVAL(51),SUMFVL(51)                                      REPRT2
000031            DIMENSION IDTYPE(2,2),IPTYPE(5,9),ND(4),NP(45)                     REPRT2
000032            EQUIVALENCE (ND(1),IDTYPE(1,1)),(NP(1),IPTYPE(1,1))                REPRT2
000033      C                                                                          REPRT2
000034            DATA (ND(I),I=1,4) /6HPOPULA,6HTION  ,6HMONTE ,6HCARLO /           REPRT2
000035            DATA (NP(I),I=1,45)/6H  POPU,6HLATION,6H SAMPL,6HING ER,6HROR    , REPRT2
000036           *                    6H  POPU,6HLATION,6H CAMS ,6HERROR ,6H      , REPRT2
000037           *                    6H  POPU,6HLATION,6H YIELD,6H ERROR,6H      , REPRT2
000038           *                    6H  POPU,6HLATION,6H AREA ,6HERROR ,6H      , REPRT2
000039           *                    6H  POPU,6HLATION,6H PRODU,6HCTION ,6HERROR , REPRT2
000040           *                    6H MONTE,6H CARLO,6H AREA ,6HERROR ,6H      , MODF2
000041           *                    6H MONTE,6H CARLO,6H PRODU,6HCTION ,6HERROR , MODF2
000042           *                    6H MONTE,6H CARLO,6H YIELD,6H ERROR,6H      , MODF2
000043           *                    6H      ,6H      ,6H CONFI,6HDENCE ,6HLEVEL / MODF2
000044            DATA JFIRST /0/, MFIRST /0/                                        MODF2
000045            IF (RPTYPE.EQ.2) IX = JPARM                                        REPRT2
000046            IF (RPTYPE.EQ.3) IX = RPTYPE*2+(JPARM-1)                           REPRT2
000047      C                                                                          MODF
000048            CALL PUNITS(FACTOR,IUNITS)                                           MODF
000049      C                                                                          MODF
000050            MINVAL = MINVAL/REF*100.                                             MODF
000051            MAXVAL = MAXVAL/REF*100.                                             MODF
000052            REF    = REF*FACTOR                                                  MODF
000053      C                                                                          REPRT2
000054      C *** PRINT HEADINGS                                                       REPRT2
000055      C                                                                          REPRT2
000056            CALL PAGHDR(-6)                                                    REPRT2
000057            IF (JFIRST.GT.0) GO TO 100                                         REPRT2
000058            WRITE (KOUT,1000) (IPTYPE(I,IX),I=1,5),ICASEN,FILNAM(1),FILNAM(2)  REPRT2
```

```
000059      C                                                                 REPRT2
000060          GO TO 120                                                     REPRT2
000061        100 WRITE (KOUT,1010) (IPTYPE(I,IX),I=1,5),ICASEN,FILNAM(1)     REPRT2
000062      C                                                                 REPRT2
000063        120 LINES = NRANGE/3                                            REPRT2
000064          IF (MOD(NRANGE,3).NE.0) LINES = LINES+1                       REPRT2
000065          NL = LINES+7                                                  REPRT2
000066          IF (RPTYPE.EQ.2) NL = NL+2                                    REPRT2
000067          IF (JFIRST.EQ.0) GO TO 130                                    MODF2
000068          IF (RPTYPE.NE.3) GO TO 205                                    MODF2
000069          IF (JPARM .NE.4) GO TO 205                                    MODF2
000070          IF (MFIRST.NE.0) GO TO 205                                    MODF2
000071      C                                                                 MODF2
000072        130 JFIRST = JPARM                                              MODF2
000073          IF (RPTYPE.EQ.3.AND.JPARM.EQ.4) MFIRST = JPARM                MODF2
000074      C                                                                 REPRT2
000075          CALL PAGHDR(NL)                                               REPRT2
000076      C                                                                 REPRT2
000077          LCOL1 = 1                                                     REPRT2
000078          LCOL2 = LCOL1+LINES                                           REPRT2
000079          LCOL3 = LCOL2+LINES                                           REPRT2
000080          JX = 3                                                        REPRT2
000081          JX2 = JX+2                                                    REPRT2
000082          IF(IX .EQ. 9)GO TO 140                                        MOD3
000083          WRITE (KOUT,1020) (IPTYPE(I,IX),I=JX,JX2),LCOL1,RANGES(LCOL1+1),  MODF
000084         *                  LCOL2,RANGES(LCOL2),RANGES(LCOL2+1),LCOL3,  REPRT2
000085         *                  RANGES(LCOL3),RANGES(LCOL3+1)               REPRT2
000086          GO TO 145                                                     MOD3
000087        140 L1 = LCOL1                                                  MOD3
000088          L2 = LCOL2                                                    MOD3
000089          L3 = LCOL3                                                    MOD3
000090          WRITE(KOUT,1025)(IPTYPE(I,IX),I=JX,JX2),L1,RANGES(L1),RANGES(L1+1)MOD3
000091         1,L2,RANGES(L2),RANGES(L2+1),L3,RANGES(L3),RANGES(L3+1)        MOD3
000092        145 CONTINUE                                                    MOD3
000093      C                                                                 REPRT2
000094          IMD = MOD(NRANGE,3)                                           MODF2
000095          LINES1 = LINES-1                                              MODF2
000096          IF (IMD.NE.0) LINES1 = LINES-2                                MODF2
000097          IF(IX .EQ. 9)LINES1 = LINES                                   MOD3
000098          DO 150 I = 2,LINES1                                           REPRT2
000099          L1 = LCOL1+(I-1)                                              REPRT2
000100          L2 = LCOL2+(I-1)                                              REPRT2
000101          L3 = LCOL3+(I-1)                                              REPRT2
000102      C                                                                 REPRT2
000103          WRITE (KOUT,1030) L1,RANGES(L1),RANGES(L1+1),L2,RANGES(L2),   REPRT2
000104         *                  RANGES(L2+1),L3,RANGES(L3),RANGES(L3+1)     REPRT2
000105      C                                                                 REPRT2
000106        150 CONTINUE                                                    REPRT2
000107          IF(IX .EQ. 9)GO TO 180                                        MOD3
000108      C                                                                 REPRT2
000109          L1 = L1+1                                                     MODF2
000110          L2 = L2+1                                                     MODF2
000111          L3 = L3+1                                                     MODF2
000112      C                                                                 REPRT2
000113          WRITE (KOUT,1040) L1,RANGES(L1),RANGES(L1+1),L2,RANGES(L2),   REPRT2
000114         *                  RANGES(L2+1),L3,RANGES(L3)                  REPRT2
000115          IF (IMD.EQ.0) GO TO 180                                       MODF2
000116      C                                                                 MODF2
000117          L1 = L1+1                                                     MODF2
000118          L2 = L2+1                                                     MODF2
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000119          C                                                                MODF2
000120              IF (IMD.EQ.1) GO TO 170                                      MODF2
000121            160 WRITE (KOUT,1050) L1,RANGES(L1),RANGES(L1+1),L2,RANGES(L2) REPRT2
000122               *,                 RANGES(L2+1)                             MODF2
000123              GO TO 180                                                    REPRT2
000124            170 WRITE (KOUT,1060) L1,RANGES(L1),RANGES(L1+1)               MODF2
000125          C                                                                REPRT2
000126            180 WRITE (KOUT,1200)                                          MODF2
000127              JX = RPTYPE-1                                                MODF2
000128              WRITE (KOUT,1070) (IDTYPE(I,JX),I=1,2),XMVAL,STDDEV,REF      REPRT2
000129             *,                 (IUNITS(I),I=1,3)                         MODF
000130              IF (RPTYPE.EQ.3) GO TO 190                                   REPRT2
000131              WRITE (KOUT,1080) XMPSUM,SDPSUM                              REPRT2
000132            190 WRITE (KOUT,1090) MINVAL,MAXVAL,NTOTL                      REPRT2
000133              WRITE (KOUT,1100) CNTRY                                      REPRT2
000134              IF (RPTYPE.EQ.3) GO TO 194                                   MODF2
000135              IF (LEVEL.LT.3)   WRITE (KOUT,1110) REGION                   REPRT2
000136              IF (LEVEL.EQ.1)   WRITE (KOUT,1120) ZONE                     REPRT2
000137              GO TO 196                                                    MODF2
000138          C                                                                MODF2
000139            194 IF (LEVEL.EQ.1) WRITE (KOUT,1180)                          MODF2
000140              IF (LEVEL.EQ.2) WRITE (KOUT,1185)                            MODF2
000141              IF (LEVEL.EQ.3) WRITE (KOUT,1190)                            MODF2
000142            196 CONTINUE                                                   MODF2
000143              IF (IX.EQ.1.OR.DATPRD(1).EQ.0) GO TO 200                     MODF1
000144              IF (DATPRD(1).LT.5) WRITE (KOUT,1135) DATPRD(1)              MODF2
000145              IF (DATPRD(1).GT.59)                                         MODF2
000146             *   WRITE (KOUT,1130) DATPRD(2),DATPRD(3),DATPRD(1)           MODF2
000147            200 WRITE (KOUT,1200)                                          REPRT2
000148          C                                                                REPRT2
000149            205 CONTINUE                                                   MODF2
000150          C                                                                REPRT2
000151              DO 210 I = 1,51                                              REPRT2
000152              FVAL(I)   = 0.0                                              REPRT2
000153            210 SUMFVL(I) = 0.0                                            REPRT2
000154          C                                                                REPRT2
000155              DO 230 I = 1,NRANGE                                          REPRT2
000156              XY = IRANGE(I)                                                       *NEW
000157              XNTOTL = NTOTL                                                       *NEW
000158              FVAL(I) = (XY/XNTOTL)*100.                                           *NEW
000159              DO 220 L = 1,I                                               REPRT2  **-1
000160            220 SUMFVL(I) = SUMFVL(I)+FVAL(L)                              REPRT2
000161            230 CONTINUE                                                   REPRT2
000162          C                                                                REPRT2
000163              NL = NL-1                                                    REPRT2
000164              CALL PAGHDR(NL)                                              REPRT2
000165          C                                                                REPRT2
000166              LCOL1 = 1                                                    REPRT2
000167              LCOL2 = LCOL1+LINES                                          REPRT2
000168              LCOL3 = LCOL2+LINES                                          REPRT2
000169          C                                                                REPRT2
000170              WRITE (KOUT,1140) LCOL1,FVAL(LCOL1),SUMFVL(LCOL1),           REPRT2
000171             *                  LCOL2,FVAL(LCOL2),SUMFVL(LCOL2),           REPRT2
000172             *                  LCOL3,FVAL(LCOL3),SUMFVL(LCOL3)            REPRT2
000173          C                                                                REPRT2
000174              LINES1 = LINES-1                                             MODF2
000175              IF(IX .EQ. 9)LINES1 = LINES                                  MOD3
000176              DO 250 I = 2,LINES1                                          REPRT2
000177          C                                                                REPRT2
000178              L1 = LCOL1+(I-1)                                             REPRT2
```

```
000179          L2 = LCOL2+(I-1)                                                REPRT2
000180          L3 = LCOL3+(I-1)                                                REPRT2
000181    C                                                                     REPRT2
000182          WRITE (KOUT,1150) L1,FVAL(L1),SUMFVL(L1),L2,FVAL(L2),SUMFVL(L2),  REPRT2
000183         *                  L3,FVAL(L3),SUMFVL(L3)                        REPRT2
000184    C                                                                     REPRT2
000185      250 CONTINUE                                                        REPRT2
000186          IF(IX .EQ. 9)GO TO 290                                          MOD3
000187    C                                                                     REPRT2
000188          L1 = L1+1                                                       MODF2
000189          L2 = L2+1                                                       MODF2
000190          L3 = L3+1                                                       MODF2
000191    C                                                                     MODF2
000192          IF (IMD.EQ.1) GO TO 270                                         MODF2
000193          IF (IMD.EQ.2) GO TO 280                                         MODF2
000194    C                                                                     REPRT2
000195          WRITE (KOUT,1150) L1,FVAL(L1),SUMFVL(L1),L2,FVAL(L2),SUMFVL(L2),  REPRT2
000196         *                  L3,FVAL(L3),SUMFVL(L3)                        REPRT2
000197          GO TO 290                                                       REPRT2
000198      270 WRITE (KOUT,1161) L1,FVAL(L1),SUMFVL(L1)                        REPRT2
000199          GO TO 290                                                       REPRT2
000200      280 WRITE (KOUT,1171) L1,FVAL(L1),SUMFVL(L1),L2,FVAL(L2),SUMFVL(L2)  REPRT2
000201      290 WRITE (KOUT,1200)                                               REPRT2
000202          JX = RPTYPE-1                                                   REPRT2
000203          WRITE (KOUT,1070) (IDTYPE(1,JX),I=1,2),XMVAL,STDDEV,REF         REPRT2
000204         *,                 (IUNITS(I),I=1,3)                            MODF
000205          IF (RPTYPE.EQ.3) GO TO 300                                      REPRT2
000206          WRITE (KOUT,1080) XMPSUM,SDPSUM                                 REPRT2
000207      300 WRITE (KOUT,1090) MINVAL,MAXVAL,NTOTL                           REPRT2
000208          WRITE (KOUT,1100) CMARY                                         REPRT2
000209          IF (RPTYPE.EQ.3) GO TO 310                                      MODF2
000210          IF (LEVEL.LT.3) WRITE (KOUT,1110) REGION                        REPRT2
000211          IF (LEVEL.EQ.1) WRITE (KOUT,1120) ZONE                          REPRT2
000212          GO TO 320                                                       MODF2
000213    C                                                                     MODF2
000214      310 IF (LEVEL.EQ.1) WRITE (KOUT,1180)                               MODF2
000215          IF (LEVEL.EQ.2) WRITE (KOUT,1185)                               MODF2
000216          IF (LEVEL.EQ.3) WRITE (KOUT,1190)                               MODF2
000217      320 CONTINUE                                                        MODF2
000218          IF (IX.EQ.1.OR.DATPRD(1).EQ.0) GO TO 500                        MODF1
000219          IF (DATPRD(1).LT.5) WRITE (KOUT,1135) DATPRD(1)                 MODF2
000220          IF (DATPRD(1).GT.99)                                            MODF2
000221         *    WRITE (KOUT,1130) DATPRD(2),DATPRD(3),DATPRD(1)             MODF2
000222    C                                                                     REPRT2
000223      500 RETURN                                                          REPRT2
000224    C                                                                     REPRT2
000225    C *** FORMATS                                                         REPRT2
000226    C                                                                     REPRT2
000227     1000 FORMAT (19X,5A6/1H /12H CASE NUMBER,I6/1H /10H FILE NAME,4X,2A4/  REPRT2
000228         *        1H )                                                    REPRT2
000229     1010 FORMAT (19X,5A6/1H /12H CASE NUMBER,I6/1H /10H FILE NAME,4X,A6/  REPRT2
000230         *        1H )                                                    REPRT2
000231     1020 FORMAT (10X,8HRANGE OF,3A6,1H&,I3,10H LESS THAN,F9.3,            MODF
000232         *        2(I4,F8.3,3H TO,F8.3))                                  REPRT2
000233     1025 FORMAT(10X,8HRANGE OF,3A6,1H&,I3,F8.3,3H TO,F8.3,                MOD3
000234        12(I4,F8.3,3H TO,F8.3))                                          MOD3
000235     1030 FORMAT (36X,3(I4,F8.3,3H TO,F8.3))                             REPRT2
000236     1040 FORMAT (36X,2(I4,F8.3,3H TO,F8.3),I4,9H GR. THAN,F10.3)        MODF2
000237     1050 FORMAT (36X,2(I4,F8.3,3H TO,F8.3))                             MODF2
000238     1060 FORMAT (36X,I4,F8.3,3H TO,F8.3)                                MODF2
```

```
000239          1070 FORMAT (11H DATA TYPE ,2A6,5H MEAN,F8.3,22H (PCT)  STD. D: IATION,MODF1
000240             *         F8.3,18H (PCT)  REF. VALUE,F11.3,1X,3A6/1H )                MODF1
000241          1080 FORMAT (24X,22HMEAN OF POPULATION SUM,F8.3,                          MODF1
000242             *         22H (PCT)  STD. DEVIATION,                                   MODF1
000243             *         18H OF POPULATION SUM,F8.3,6H (PCT)/1H )                     MODF1
000244          1090 FORMAT (24X,15HMINIMUM READING,F8.3,23H (PCT)  MAXIMUM READING,     MODF1
000245             *         F8.3,20H (PCT)  NUMBER OF READINGS,I6/1H )                   MODF1
000246          1100 FORMAT (23X,12H COUNTRY     ,A4)                                    MODF1
000247          1110 FORMAT (1H+,43X,6HREGION,I8)                                        MODF1
000248          1120 FORMAT (1H+,63X,4HZONE,I8)                                          MODF1
000249          1130 FORMAT (1H+,83X,15HPREDICTION DATE,I4,1H/,I2,1H/,I2)                MODF1
000250          1135 FORMAT (1H+,65X,17HBIC-WINDOW NUMBER,I2)                            MODF2
000251          1140 FORMAT (19X,16HHISTOGRAM DATA  ,5(I4,1H-,F9.3,F10.3))               REPRT2
000252          1150 FORMAT (35X,5(I4,1H-,F9.3,F10.3))                                   REPRT2
000253          1160 FORMAT (35X,I4,1H-,F9.3,F10.3)
000254          1170 FORMAT (35X,2(I4,1H-,F9.3,F10.3))
000255          1180 FORMAT (1H+,48X,17H*** ZONE DATA ***)                               MODF2
000256          1185 FORMAT (1H+,48X,18H*** REGION DATA ***)                             MODF2
000257          1190 FORMAT (1H+,48X,20H*** COUNTRY DATA ***)                            MODF2
000258          1200 FORMAT (1H )                                                        REPRT2
000259          C                                                                        REPRT2
000260               END                                                                 REPRT2
```

```
EWF REPRT3 [illegible] STD. DEVIATION,MODF1
          [illegible] VALUE [illegible]                                    MODF1
      [illegible] SUBROUTINE REPRT3 [illegible]                            REPRT3
      C         [illegible] STD. DEVIATION [illegible]                     REPRT3
      C THIS ROUTINE PRINTS THE FREQUENCY OF SAMPLE SEGMENTS ACQUISITIONS  REPRT3
      C [illegible] (PCT) MAXIMUM READING,                                 REPRT3
      C         [illegible] (PCT) [illegible]                              REPRT3
      C *** COMMON STORAGE [illegible]                                     REPRT3
      C [illegible]                                                        REPRT3
      [illegible] COMMON /PRTCTL/ NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)      PRTCTL
      [illegible] INTEGER HEADER [illegible]                               PRTCTL
      C [illegible]                                                        REPRT3
      [illegible] COMMON /PRNTID/ ENTRY,REGION,ZONE,STRATA,ICASEN [illegible] PRNTID
      [illegible] FILNAM(2))                                               PRNTID
      [illegible] ITOTYP,VAREXT [illegible]                                PRNTID
      C [illegible]                                                        PRNTID
      [illegible] INTEGER ENTRY,REGION,ZONE,STRATA,FILENO,FILNAM           MODF1
      [illegible] COMMON /ABCMBN/ NSEGS,BIOCMB(16) [illegible]             BPCMBN
      C [illegible]                                                        REPRT3
      C [illegible]                                                        REPRT3
      C *** LOCAL STORAGE                                                  REPRT3
      C                                                                    REPRT3
            DATA JFIRST /0/, MPAGE /0/                                     REPRT3
      C                                                                    REPRT3
      C *** CHECK FOR FIRST TIME THROUGH                                   REPRT3
      C                                                                    REPRT3
            IF (JFIRST.NE.0) GO TO 100                                     REPRT3
      C                                                                    REPRT3
      C *** SAVE FIRST PAGE NUMBER                                         REPRT3
      C                                                                    REPRT3
            NPAGE = NPAGE+1                                                REPRT3
            JFIRST = NPAGE                                                 REPRT3
      C                                                                    REPRT3
      C *** PRINT PAGE HEADINGS                                            REPRT3
      C                                                                    REPRT3
            CALL PAGHDR(-8)                                                REPRT3
            WRITE (KOUT,1000) ICASEN,FILNAM(1),FILNAM(2)                   REPRT3
      C                                                                    REPRT3
      C *** PRINT TOTAL LINE                                               REPRT3
      C                                                                    REPRT3
        100 GO TO (150,200,300),ITOTYP                                     REPRT3
      C                                                                    REPRT3
      C *** PRINT ZONE LINE                                                REPRT3
      C                                                                    REPRT3
        150 CALL PAGHDR(2)                                                 MOD1
      C                                                                    REPRT3
            IF (NPAGE.EQ.MPAGE) GO TO 180                                  REPRT3
      C                                                                    REPRT3
            MPAGE = NPAGE                                                  REPRT3
            CALL PAGHDR(2)                                                 MOD1
            WRITE (KOUT,1000) ICASEN,FILNAM(1),FILNAM(2)                   REPRT3
      C                                                                    REPRT3
        180 WRITE(KOUT,2000)                                               MOD1
            WRITE(KOUT,1010)ZONE,NSEGS,(BIOCMB(I),I=1,16)                  MOD1
      C                                                                    REPRT3
            GO TO 500                                                      REPRT3
      C                                                                    REPRT3
      C *** PRINT REGION LINE                                              REPRT3
      C                                                                    REPRT3
        200 CALL PAGHDR(5)                                                 REPRT3
```

```
000059          C                                                               REPRT3
000060                IF (NPAGE.EQ.MPAGE) GO TO 220                             REPRT3
000061          C                                                               REPRT3
000062                MPAGE = NPAGE                                             REPRT3
000063                CALL PAGHDR(2)                                            MOD1
000064                WRITE (KOUT,1000) ICASEN,FILNAM(1),FILNAM(2)              REPRT3
000065          C                                                               REPRT3
000066            220 WRITE (KOUT,1040)                                         REPRT3
000067                WRITE (KOUT,1030)                                         REPRT3
000068                WRITE(KOUT,2001)                                          MOD1
000069                WRITE (KOUT,1010) REGION,NSEGS,(BIOCMB(I),I=1,16)         REPRT3
000070                WRITE (KOUT,1040)                                         REPRT3
000071          C                                                               REPRT3
000072                GO TO 500                                                 REPRT3
000073          C                                                               REPRT3
000074          C *** PRINT COUNTRY LINE                                        REPRT3
000075          C                                                               REPRT3
000076            300 CALL PAGHDR(5)                                            REPRT3
000077          C                                                               REPRT3
000078                IF (NPAGE.EQ.MPAGE) GO TO 320                             REPRT3
000079          C                                                               REPRT3
000080                CALL PAGHDR(2)                                            MOD1
000081                WRITE (KOUT,1000) ICASEN,FILNAM(1),FILNAM(2)              REPRT3
000082          C                                                               REPRT3
000083            320 WRITE (KOUT,1040)                                         REPRT3
000084                WRITE (KOUT,1030)                                         REPRT3
000085                WRITE (KOUT,1030)                                         REPRT3
000086                WRITE(KOUT,2002)                                          MOD1
000087                WRITE (KOUT,1020) CNTRY,NSEGS,(BIOCMB(I),I=1,16)          REPRT3
000088          C                                                               REPRT3
000089          C                                                               REPRT3
000090            500 RETURN                                                    REPRT3
000091          C                                                               REPRT3
000092          C *** PRINT FORMATS                                             REPRT3
000093          C                                                               REPRT3
000094           1000 FORMAT (16X,50HF R E Q U E N C Y  O F  S A M P L E  S E G M E N T,REPRT3
000095          *        25H  A C Q U I S I T I O N S/1H /7X,11HCASE NUMBER,I6,  REPRT3
000096          *         9X,11HFILE NAME  ,2A4/1H /6X,18HNO. OF  BIO-WINDOW,   REPRT3
000097          *        26H COMBINATIONS (IN PERCENT)/7X,8HSEGMENTS/            REPRT3
000098          *         9X, 9HUSED     0,5X,1H1,5X,1H2,5X,1H3,5X,1H4,         REPRT3
000099          *         4X,46H1-2   1-3   1-4   2-3   2-4   3-4  1-2-3 1-2-4, REPRT3
000100          *        20H 1-3-4 2-3-4 1-2-3-4)                               MOD1
000101           1010 FORMAT (I5,I8,1X,15F6.2,F7.2)                           REPRT3
000102           1020 FORMAT (1X,A4,I8,1X,15F6.2,F7.2)                        REPRT3
000103           1030 FORMAT (1X,18(6H******))                                REPRT3
000104           1040 FORMAT (1H )                                            REPRT3
000105           2000 FORMAT(5H ZONE)                                         MOD1
000106           2001 FORMAT(5H  REG)                                         MOD1
000107           2002 FORMAT(5H COUN)                                         MOD1
000108          C                                                             REPRT3
000109                END                                                     REPRT3
```

```
@ EDIT SAMERR,1,760513, 61005   , 1

000001            SUBROUTINE SAMERR                                          SAMERR
000002          C                                                            SAMERR
000003          C *** ROUTINE PRODUCES THE POPULATION SAMPLING ERROR REPORT   SAMERR
000004          C                                                            SAMERR
000005          C                                                            SAMERR
000006          C *** COMMON STORAGE                                         SAMERR
000007          C                                                            SAMERR
000008            COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                     CARDIN
000009           *,              ICASIN(5),LEVEL,JPARM,IERR                  CARDIN
000010            INTEGER RPTYPE,AUNITS,PARMTR                               CARDIN
000011          C                                                            SAMERR
000012            COMMON /RANGE / START,INTVL1,BREAK1,INTVL2,BREAK2           RANGE
000013           *,               INTVL3,STOP                                RANGE
000014          C                                                            RANGE
000015            REAL    INTVL1,INTVL2,INTVL3                               RANGE
000016          C                                                            SAMERR
000017            COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA        FILBUF
000018           *,               ISESTA,ICASE,MXSEG                         MODF
000019           *,               KBUF(125)                                  MODF
000020          C                                                            FILBUF
000021            DIMENSION KWIN(4),IBIOND(4),IPRDAT(14),PRDATE(6),XEPW(4),XEYLD(6) FILBUF
000022           *,         APVAL(2)                                         FILBUF
000023           *,         MCERR(100)                                       MODF
000024          C                                                            FILBUF
000025            REAL    MCREF,MCERR                                        MODF
000026            INTEGER FILENM,PRDATE                                      FILBUF
000027          C                                                            FILBUF
000028            EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                      FILBUF
000029           *,  (KBUF(2),IGPSU,ISTRAT,NREGS)                            MODF
000030           *,  (KBUF(3),XMPA,NZTOT)                                    MODF
000031           *,  (KBUF(4),AREA,NBR)                                      MODF
000032           *,  (KBUF(5),XTPN,NPD)                                      MODF
000033           *,  (KBUF(6),IBIOND(1))                                     MODF
000034           *,  (KBUF(7),A)                                             MODF
000035           *,  (KBUF(8),XBTPN)                                         MODF
000036           *,  (KBUF(9),XCV2)                                          MODF
000037           *,  (KBUF(10),XTPAI,IPRDAT(1))                              MODF
000038           *,  (KBUF(11),PRDATE(1))                                    MODF
000039           *,  (KBUF(17),XEPW(1))                                      MODF
000040           *,  (KBUF(21),XEYLD(1))                                     MODF
000041           *,  (KBUF(24),APVAL(1),MCREF)                               MODF
000042           *,  (KBUF(25),MCERR(1))                                     MODF
000043           *,  (KBUF(27),XTYLD)                                        MODF
000044          C                                                            SAMERR
000045            COMMON /PRNTID/ CNTRY,REGION,ZONE,STRATA,ICASEN            PRNTID
000046           *,               FILENO,FILNAM(2)                           PRNTID
000047           *,               ITUTYP,NAMEMT                              PRNTID
000048          C                                                            PRNTID
000049            INTEGER CNTRY,REGION,ZONE,STRATA,FILENO,FILNAM             MODF
000050          C                                                            SAMERR
000051            COMMON /HSTOGM/ XMVAL,STUDEV,XMPSUM,SDFSUM                 HSTOGM
000052           *,               NRANGE,IRANGE(51),NTOTL,RANGES(52)         HSTOGM
000053           *,               MINVAL,MAXVAL,DATPRD(3),NZRNG,REF          HSTOGM
000054          C                                                            HSTOGM
000055            REAL    MINVAL,MAXVAL                                      HSTOGM
000056            INTEGER DATPRD                                             HSTOGM
000057          C                                                            SAMERR
000058            COMMON /MATCH1/ MATCH                                              *NEW
```

```
000059            COMMON /READNG/ READNG(5600)                                  MODF
000060      C                                                                   MODF
000061            DIMENSION CLEWA(100),CLEPRD(100),CLATEC(100),CLPTEC(100)      MODF
000062           *,         CLATWC(100),CLPTWC(100)                             MODF
000063      C                                                                   MODF
000064            EQUIVALENCE                          (READNG(5001),CLEWA(1))  MODF
000065           *,  (READNG(5101),CLEPRD(1)),         (READNG(5201),CLATEC(1))  MODF
000066           *,  (READNG(5301),CLPTEC(1)),         (READNG(5401),CLATWC(1))  MODF
000067           *,  (READNG(5501),CLPTWC(1))                                    MODF
000068      C                                                                   SAMERR
000069      C                                                                   SAMERR
000070      C *** LOCAL STORAGE                                                 SAMERR
000071            INTEGER SBSTRA                                                SAMERR
000072      C                                                                   SAMERR
000073            DATA AREAPS /1021.712/                                        SAMERR
000074            IH = 1                                                        MOD2
000075          5 CONTINUE                                                      MOD2
000076            DO 10 I = 1,3200                                              SAMERR
000077         10 READNG(I) = 0.0                                               SAMERR
000078      C                                                                   SAMERR
000079            J = 0                                                         SAMERR
000080            KXM = 0                                                       SAMERR
000081            NTOTL = 0                                                     SAMERR
000082      C                                                                   SAMERR
000083            DO 20 I = 1,51                                                SAMERR
000084         20 IRANGE(I) = 0                                                 SAMERR
000085      C                                                                   SAMERR
000086            DELSB = 0.0                                                   MOD2
000087            DELERR = 0.0                                                  MOD2
000088            REF = 0.0                                                     MOD2
000089            MINVAL = 1.E10                                                MOD2
000090            MAXVAL = -1.E10                                               MOD2
000091            DELSG = 0.0                                                   MOD2
000092            IF(IH .EQ. 0)GO TO 50                                         MOD2
000093      C                                                                   SAMERR
000094      C *** GET HEADER RECORD                                             SAMERR
000095      C                                                                   SAMERR
000096            CALL FILSEQ(IH)                                               SAMERR
000097      C                                                                   SAMERR
000098            IF (IERR.NE.0) GO TO 500                                      SAMERR
000099      C                                                                   SAMERR
000100            ICASEN = ICASE                                                SAMERR
000101            FILNAM(1) = FILENM(1)                                         SAMERR
000102            FILNAM(2) = FILENM(2)                                         SAMERR
000103            NAMENT = 0                                                    SAMERR
000104      C                                                                   SAMERR
000105            IH = 0                                                        SAMERR
000106      C                                                                   SAMERR
000107      C *** GET DETAIL RECORDS                                            SAMERR
000108      C                                                                   SAMERR
000109         50 CALL FILSEQ(IH)                                               SAMERR
000110      C                                                                   SAMERR
000111            IF (IH.LT.0) GO TO 200                                        SAMERR
000112      C                                                                   SAMERR
000113            IF(KXM .GT. 0)GO TO 100                                       MOD2
000114            IF(MATCH .EQ. -2)GO TO 210                                    MOD2
000115      C                                                                   *NEW
000116            CNTRY = ICNTRY                                                SAMERR
000117            REGION = IREGN                                                SAMERR
000118            ZONE = IZONE                                                  SAMERR
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000119          STRATA = ISTRTA                                                 SAMERR
000120          SBSTRA = ISBSTA                                                 SAMERR
000121    C                                                                     SAMERR
000122          GO TO 120                                                       SAMERR
000123    C                                                                     SAMERR
000124    C TEST FOR CHANGE IN I.D. AREA                                        SAMERR
000125    C                                                                     SAMERR
000126      100 IF(ISBSTA .NE. SBSTRA .OR. ISTRTA .NE. STRATA)GO TO 200         MOD2
000127          GO TO (105,110,120), LEVEL                                      SAMERR
000128      105 IF(IZONE .NE. ZONE)GO TO 200                                    MOD2
000129          IF(IREGN .NE. REGION)GO TO 200                                  MOD2
000130          GO TO 120                                                       SAMERR
000131      110 IF(IREGN .NE. REGION)GO TO 200                                  MOD2
000132      120 DELSQ = DELSQ+(XTP*I-XTPW)                                      SAMERR
000133          KXN = KXN+1                                                     SAMERR
000134    C                                                                     SAMERR
000135          GO TO 50                                                        SAMERR
000136    C                                                                     SAMERR
000137    C *** CHANGE IN I.D.AREA                                              SAMERR
000138    C                                                                     SAMERR
000139      200 CONTINUE                                                        MOD2
000140          IF(KXN .EQ. 0)GO TO 210                                         MOD2
000141          XNA = NA                                                        MOD2
000142          XN  = N                                                         SAMERR
000143          XR  = AREA/(XNA*AREAPS)                                         SAMERR
000144          XNR = XN*XR*AREAPS                                              SAMERR
000145    C                                                                     SAMERR
000146          J = J+1                                                         SAMERR
000147          XM = KXN                                                        SAMERR
000148          READNG(J)  = DELSQ/XM*XNR                                       SAMERR
000149    C                                                                     SAMERR
000150          MINVAL = AMIN1(MINVAL,READNG(J))                                SAMERR
000151          MAXVAL = AMAX1(MAXVAL,READNG(J))                                SAMERR
000152    C                                                                     SAMERR
000153          DELERR = DELERR+READNG(J)                                       SAMERR
000154          REF    = REF+XTP*XNR                                            SAMERR
000155          DELSQ  = DELSQ+READNG(J)**2                                     SAMERR
000156    C                                                                     SAMERR
000157      210 CONTINUE                                                        MOD2
000158          DELSQ = 0                                                       SAMERR
000159          KXN   = 0                                                       SAMERR
000160    C                                                                     SAMERR
000161          IF(TH .LT. 0)GO TO 250                                          MOD2
000162          GO TO (220,230,240), LEVEL                                      SAMERR
000163    C                                                                     SAMERR
000164      220 IF ( IZONE .NE. ZONE )  GO TO 250                               MODE2
000165    C                                                                     SAMERR
000166      230 IF (IREGN.EQ.REGION) GO TO 50                                   SAMERR
000167          GO TO 250                                                       SAMERR
000168    C                                                                     SAMERR
000169      240 IF (IN.EQ.0) GO TO 50                                           SAMERR
000170    C                                                                     SAMERR
000171      250 IF(J .EQ. 0)GO TO 400                                           MOD2
000172          NTOTL  = J                                                      MOD2
000173    C                                                                     SAMERR
000174          DO 300 I = 1,NTOTL                                              SAMERR
000175          PCDERR = 0.0                                                    MOD2
000176          IF(REF .NE. 0.0)PCDERR = 100.*READNG(I)/REF                     MOD2
000177          DO 260  K = 1,NRANGE                                            SAMERR
000178          IF (PCDERR.GE.0.) GO TO 260                                     SAMERR
```

```
000179      C                                                                 SAMERR
000180            IRANGE(NZRNG) = IRANGE(NZRNG)+1                             SAMERR
000181            GO TO 300                                                   SAMERR
000182      C                                                                 SAMERR
000183        260 IF (PCDERR.GE.RANGES(K+1)) GO TO 280                        SAMERR
000184      C                                                                 SAMERR
000185            IRANGE(K) = IRANGE(K)+1                                     SAMERR
000186            GO TO 300                                                   SAMERR
000187      C                                                                 SAMERR
000188        280 CONTINUE                                                    SAMERR
000189        300 CONTINUE                                                    SAMERR
000190      C                                                                 SAMERR
000191            TJ = NTOTL                                                  SAMERR
000192            XMVAL = 0.0                                                 MOD2
000193            IF(REF .NE. 0.0)XMVAL = 100./(TJ*REF)*DELERR                MOD2
000194            STDDEV = 0.0                                                MOD2
000195            IF(DELERR**P/TJ .GT. DELSQ)GO TO 390
000196            IF(J .GT. 1 .AND. REF .NE. 0.0)STDDEV = 100./REF*SQRT(1./(TJ-1.)  MOD2
000197           1   *(DELSQ-1./TJ*DELERR**2))                                MOD2
000198        390 XMPSUM = TJ*XMVAL
000199            SDPSUM = SQRT(TJ)*STDDEV                                    SAMERR
000200      C                                                                 SAMERR
000201            CALL REPRT2                                                 SAMERR
000202      C                                                                 SAMERR
000203        400 IF(IN .EQ. 0)GO TO 5                                        MOD2
000204        500 RETURN                                                      SAMERR
000205      C                                                                 SAMERR
000206            END                                                         SAMERR
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
o EXT SUBREF,1,760511, 59116   , 1
000001                SUBROUTINE SUBREF                                          SUBREF
000002          C                                                                SUBREF
000003          C     THIS ROUTINE PRODUCES THE SUBSTRATA REFERENCE DATA REPORT   SUBREF
000004          C                                                                SUBREF
000005          C *** COMMON STORAGE                                             SUBREF
000006          C                                                                SUBREF
000007                COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA        FILBUF
000008               *,               ISBSTA,ICASE,MXSEG                         MODF
000009               *,               KBUF(125)                                  MODF
000010          C                                                                FILBUF
000011                DIMENSION KWIN(4),IBIOWD(4),IPRDAT(14),PRDATE(6),XEPW(4),XEYLD(6) FILBUF
000012               *,           APVAL(2)                                      FILBUF
000013               *,           MCERR(100)                                    MODF
000014          C                                                                FILBUF
000015                REAL    MCREF,MCERR                                        MODF
000016                INTEGER FILENM,PRDATE                                      FILBUF
000017          C                                                                FILBUF
000018                EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                      FILBUF
000019               *,  (KBUF(2),IBIOWD,NSTRAT,NREGS)                           MODF
000020               *,  (KBUF(3),XMIN,NZTOT)                                    MODF
000021               *,  (KBUF(4),AREA,NB4)                                      MODF
000022               *,  (KBUF(5),XTEN,NPD)                                      MODF
000023               *,  (KBUF(6),N,IBIOWD(1))                                   MODF
000024               *,  (KBUF(7),DA)                                            MODF
000025               *,  (KBUF(8),XSIRD)                                         MODF
000026               *,  (KBUF(9),XCYR)                                          MODF
000027               *,  (KBUF(10),XTRNI,IPRDAT(1))                              MODF
000028               *,  (KBUF(11),PRDATE(1))                                    MODF
000029               *,  (KBUF(17),XEPW(1))                                      MODF
000030               *,  (KBUF(21),XEYLD(1))                                     MODF
000031               *,  (KBUF(24),APVAL(1),MCREF)                               MODF
000032               *,  (KBUF(25),MCERR(1))                                     MODF
000033               *,  (KBUF(27),XTYLD)                                        MODF
000034          C                                                                SUBREF
000035                COMMON /PRNTID/ CNTRY,REGION,ZONE,STRATA,ICASEN            PRNTID
000036               *,               FILENO,FILNAM(2)                           PRNTID
000037               *,               ITOTYP,NAMFMT                              PRNTID
000038          C                                                                PRNTID
000039                INTEGER CNTRY,REGION,ZONE,STRATA,FILENO,FILNAM             MODF
000040          C                                                                SUBREF
000041                COMMON /SUBVAL/ AWA,TWA,NSUBGP(3),NSEGGP(2),XINBS          SUBVAL
000042               *,               XSINBS,XINCV,XSINCV                        SUBVAL
000043          C                                                                SUBREF
000044          C                                                                SUBREF
000045          C *** LOCAL STORAGE                                              SUBREF
000046          C                                                                SUBREF
000047                DIMENSION SVAL(11),ZVAL(11),RVAL(11),CVAL(11)              SUBREF
000048               *,         XSMCVS(2),NSBGPS(3),NSGGPS(2),NSBGPZ(3),NSGGPZ(2) SUBREF
000049               *,         NSBGPR(3),NSGGPR(2),NSBGPC(3),NSGGPC(2)         SUBREF
000050                EQUIVALENCE                                                SUBREF
000051               *  (SVAL(1),AWAS),  (SVAL(2),TWAS),  (SVAL(3),XINBSS)      SUBREF
000052               *, (SVAL(4),XINCVS),(SVAL(5),XSMCVS(1))                    SUBREF
000053               *, (SVAL(7),NSBGPS(1)),  (SVAL(10),NSGGPS(1))              SUBREF
000054          C                                                                SUBREF
000055               *, (ZVAL(1),AWAZ),  (ZVAL(2),TWAZ),  (ZVAL(3),XINBSZ)      SUBREF
000056               *, (ZVAL(4),XSMBSZ),  (ZVAL(5),XINCVZ),  (ZVAL(6),XSMCVZ)  SUBREF
000057               *, (ZVAL(7),NSBGPZ(1)),  (ZVAL(10),NSGGPZ(1))              SUBREF
000058          C                                                                SUBREF
```

```
000059          *, (RVAL(1),HWAK),  (RVAL(2),TWAR),  (RVAL(3),XINBSR)                SUBREF
000060          *, (RVAL(4),XSMBSR),  (RVAL(5),XINCVR),  (RVAL(6),XSMCVR)            SUBREF
000061          *, (RVAL(7),NSBGPR(1)),  (RVAL(10),NSGGPR(1))                        SUBREF
000062      C                                                                        SUBREF
000063          *, (CVAL(1),HWAC),  (CVAL(2),TWAC),  (CVAL(3),XINBSC)                SUBREF
000064          *, (CVAL(4),XSMBSC),  (CVAL(5),XINCVC),  (CVAL(6),XSMCVC)            SUBREF
000065          *, (CVAL(7),NSBGPC(1)),  (CVAL(10),NSGGPC(1))                        SUBREF
000066           DATA AREAPS /10289.712/                                             SUBREF
000067           IH =1                                                               SUBREF
000068           KSBSTA = 0                                                          SUBREF
000069           J = 0                                                               SUBREF
000070      C                                                                        SUBREF
000071           DO 20 I = 1,11                                                      SUBREF
000072           SVAL(I) = 0                                                         SUBREF
000073           ZVAL(I) = 0                                                         SUBREF
000074           RVAL(I) = 0                                                         SUBREF
000075        20 CVAL(I) = 0                                                         SUBREF
000076      C                                                                        SUBREF
000077      C GET HEADER RECORD INFO.                                                SUBREF
000078      C                                                                        SUBREF
000079           CALL FILSEQ(IH)                                                     SUBREF
000080      C                                                                        SUBREF
000081           ICASEP = ICASE                                                      SUBREF
000082           IH = 0                                                              SUBREF
000083      C GET DETAIL RECORD                                                      SUBREF
000084      C                                                                        SUBREF
000085        50 CALL FILSEQ(IH)                                                     SUBREF
000086           IF (IH.LT.0) GO TO 200                                              SUBREF
000087           KSBSTA = KSBSTA+1                                                   SUBREF
000088      C                                                                        SUBREF
000089           IF (KSBSTA.GT.1) GO TO 100                                          SUBREF
000090      C                                                                        SUBREF
000091           CNTRY = ICNTRY                                                      SUBREF
000092           REGION = IREGN                                                      SUBREF
000093           ZONE = IZONE                                                        SUBREF
000094           STRATA = ISTRTA                                                     SUBREF
000095      C                                                                        SUBREF
000096           GO TO 120                                                           SUBREF
000097      C                                                                        SUBREF
000098      C TEST FOR CHANGE IN IDENTIFICATION AREA                                 SUBREF
000099      C                                                                        SUBREF
000100        100 IF (ISTRTA.GT.STRATA) GO TO 200                                    SUBREF
000101           IF (IZONE.GT.ZONE) GO TO 200                                        SUBREF
000102           IF (IREGN.GT.REGION) GO TO 200                                      SUBREF
000103      C                                                                        SUBREF
000104        120 XNA = NA                                                           SUBREF
000105           XN = N                                                              SUBREF
000106           XP = AREA/(XNA*AREAPS)                                              SUBREF
000107           XNR = XN*XR*AREAIS                                                  SUBREF
000108      C                                                                        SUBREF
000109           HWAK = XNR*XHP*                                                     MOD3
000110           HWAS = HWAS+HWAK                                                    SUBREF
000111      C                                                                        SUBREF
000112           TWAK = XNR*XTP*                                                     MOD3
000113           TWAS = TWAS+TWAK                                                    SUBREF
000114      C                                                                        SUBREF
000115           NSBGPS(IGPNO) = NSBGPS(IGPNO)+1                                     SUBREF
000116           IF (IGPNO.LT.3) NSGGPS(IGPNO) = NSGGPS(IGPNO)+NSEG                  SUBREF
000117      C                                                                        SUBREF
000118           XINBSS = XBTPW*HWAK+XINBSS                                          SUBREF
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
000119                XINCVS = (XCV2*TWAK)**2+XINCVS                          SUBREF
000120                XSMCV  = TWAK-HWAK                                      SUBREF
000121                XSMCVS(1) = XSMCVS(1)+XSMCV**2                          SUBREF
000122                XSMCVS(2) = XSMCVS(2)+XSMCV                             SUBREF
000123           C                                                            SUBREF
000124                J = J+1                                                 SUBREF
000125           C                                                            SUBREF
000126                GO TO 50                                                SUBREF
000127           C                                                            SUBREF
000128           C CHANGE IN I.D. AREA                                        SUBREF
000129           C                                                            SUBREF
000130             200 HWA = HWAS                                             SUBREF
000131                 TWA = TWAS                                             SUBREF
000132                 XINBS = XINBSS/HWA                                     SUBREF
000133                 XSIMBS = (TWA-HWA)/HWA                                 SUBREF
000134                 XINCV = SQRT(XINCVS)/TWA                               SUBREF
000135                 XSIMCV = 0.0                                           MODF
000136                 IF((XSMCVS(2)**2/J - XSMCVS(1)) .GT. 0.)GO TO 205
000137                 IF (J.LT.2) GO TO 205                                  MODF
000138                 C = J
000139                 XSIMCV = SQRT(C/(C-1)*(XSMCVS(1)-1./C*XSMCVS(2)**2))/TWA
000140             205 CONTINUE                                               MODF
000141           C                                                            SUBREF
000142                 DO 210 I = 1,3                                         SUBREF
000143                 IF (I.EQ.3) GO TO 210                                  SUBREF
000144                 NSEGGP(I) = NSGGPS(I)                                  SUBREF
000145             210 NSURGP(I) = NSBGPS(I)                                  SUBREF
000146           C                                                            SUBREF
000147           C CALL REPRT1 WITH ITOTYP=1 FOR STRATA PRINT                 SUBREF
000148           C                                                            SUBREF
000149                 ITOTYP = 1                                             SUBREF
000150                 CALL REPRT1                                            SUBREF
000151                 XINBS = XINBS*HWA                                      MOD3
000152                 XSIMBS = XSIMBS*HWA                                    MOD3
000153                 XINCV = XINCV*TWA                                      MOD3
000154                 XSIMCV = XSIMCV*TWA                                    MOD3
000155           C                                                            SUBREF
000156           C SAVE STRATA VALUES IN ZONE ACCUMULATORS                    SUBREF
000157           C                                                            SUBREF
000158                 HWAZ = HWAZ+HWA                                        SUBREF
000159                 TWAZ = TWAZ+TWA                                        SUBREF
000160                 XINBSZ = XINBSZ+XINBS                                  SUBREF
000161                 XSIMBSZ = XSIMBSZ+XSIMBS                               SUBREF
000162                 XINCVZ = XINCVZ+XINCV                                  SUBREF
000163                 XSMCVZ = XSMCVZ+XSIMCV                                 SUBREF
000164           C                                                            SUBREF
000165                 DO 220 I = 1,3                                         SUBREF
000166                 IF (I.EQ.3) GO TO 220                                  SUBREF
000167                 NSGGPZ(I) = NSGGPZ(I)+NSEGGP(I)                        SUBREF
000168             220 NSBGPZ(I) = NSBGPZ(I)+NSURGP(I)                        SUBREF
000169           C                                                            SUBREF
000170           C ZERO OUT STRATA ACCUMULATOR VALUES                         SUBREF
000171           C                                                            SUBREF
000172                 DO 230 I = 1,11                                        SUBREF
000173             230 SVAL(I) = 0                                            SUBREF
000174                 J = 0                                                  SUBREF
000175                 IF (IH.EQ.0) STRATA = ISTRTA                           SUBREF
000176           C                                                            SUBREF
000177           C RESET STRATA I.D.                                          SUBREF
000178           C                                                            SUBREF
```

```
000179          C                                                               SUBREF
000180          C TEST FOR CHANGE IN ZONE I.D. OR END-OF-FILE                   SUBREF
000181          C                                                               SUBREF
000182               IF (IH.LT.0) GO TO 240                                     SUBREF
000183          C                                                               SUBREF
000184               IF (IZONE.GT.ZONE) GO TO 240                               SUBREF
000185               IF (IREGN.EQ.REGION) GO TO 120                             SUBREF
000186          C                                                               SUBREF
000187          240 HWA = HWAZ                                                  SUBREF
000188              TWA = TWAZ                                                  SUBREF
000189              XINBS = XINBSZ/HWA                                          MOD3
000190              XSIMBS = XSMBSZ/HWA                                         MOD3
000191              XINCV = XINCVZ/TWA                                          MOD3
000192              XSIMCV = XSMCVZ/TWA                                         MOD3
000193          C                                                               SUBREF
000194               DO 250 I = 1,3                                             SUBREF
000195               IF (I.EQ.3) GO TO 250                                      SUBREF
000196               NSEGGP(I) = NSGGPZ(I)                                      SUBREF
000197          250 NSUBGP(I) = NSBGPZ(I)                                       SUBREF
000198          C                                                               SUBREF
000199          C CALL REPRT1 WITH ITOTYP=2 FOR ZONE PRINT                      SUBREF
000200          C                                                               SUBREF
000201               ITOTYP = 2                                                 SUBREF
000202               CALL REPRT1                                                SUBREF
000203          C                                                               SUBREF
000204          C SAVE ZONE VALUES IN REGION ACCUMULATORS                       SUBREF
000205          C                                                               SUBREF
000206               HWAR = HWAR+HWA                                            SUBREF
000207               TWAR = TWAR+TWA                                            SUBREF
000208               XINBSR = XINBSR + XINBSZ                                   MOD3
000209               XSIBSR = XSMBSR + XSMBSZ                                   MOD3
000210               XINCVR = XINCVR + XINCVZ                                   MOD3
000211               XSICVR = XSMCVR + XSMCVZ                                   MOD3
000212          C                                                               SUBREF
000213               DO 260 I =1,3                                              SUBREF
000214               IF (I.EQ.3) GO TO 260                                      SUBREF
000215               NSGGPR(I) = NSGGPR(I)+NSEGGP(I)                            SUBREF
000216          260 NSBGPR(I) = NSBGPR(I)+NSUBGP(I)                             SUBREF
000217          C                                                               SUBREF
000218          C ZERO OUT ZONE ACCUMULATOR VALUES                              SUBREF
000219          C                                                               SUBREF
000220               DO 270 I = 1,11                                            SUBREF
000221          270 ZVAL(I) = 0                                                 SUBREF
000222          C                                                               SUBREF
000223          C RESET ZONE I.D.                                               SUBREF
000224          C                                                               SUBREF
000225               IF (IH.EQ.0) ZONE = IZONE                                  SUBREF
000226          C                                                               SUBREF
000227          C TEST FOR CHANGE IN REGION ID OR END-OF-FILE                   SUBREF
000228          C                                                               SUBREF
000229               IF (IH.LT.0) GO TO 280                                     SUBREF
000230          C                                                               SUBREF
000231               IF (IREGN.EQ.REGION) GO TO 120                             SUBREF
000232          C                                                               SUBREF
000233          280 HWA = HWAR                                                  SUBREF
000234              TWA = TWAR                                                  SUBREF
000235              XINBS = XINBSR/HWA                                          MOD3
000236              XSIMBS = XSMBSR/HWA                                         MOD3
000237              XINCV = XINCVR/TWA                                          MOD3
000238              XSIMCV = XSMCVR/TWA                                         MOD3
```

```
000239          C                                                               SUBREF
000240                DO 290 I = 1,3                                            SUBREF
000241                IF (I.EQ.3) GO TO 290                                     SUBREF
000242                NSEGGP(I) = NSGGPR(I)                                     SUBREF
000243            290 NSUBGP(I) = NSBGPR(I)                                     SUBREF
000244          C                                                               SUBREF
000245          C CALL REPRT1 WITH ITOTYP=3 FOR REGION PRINT                    SUBREF
000246          C                                                               SUBREF
000247                ITOTYP = 3                                                SUBREF
000248                CALL REPRT1                                               SUBREF
000249          C                                                               SUBREF
000250          C SAVE REGION VALUES IN COUNTRY ACCUMULATORS                    SUBREF
000251          C                                                               SUBREF
000252                HWAC = HWAC+HWA                                           SUBREF
000253                TWAC = TWAC+TWA                                           SUBREF
000254                XINBSC = XINBSC + XINBSR                                  MOD3
000255                XSMBSC = XSMBSC + XSMBSR                                  MOD3
000256                XINCVC = XINCVC + XINCVR                                  MOD3
000257                XSMCVC = XSMCVC + XSMCVR                                  MOD3
000258          C                                                               SUBREF
000259                DO 300 I = 1,3                                            SUBREF
000260                IF (I.EQ.3) GO TO 300                                     SUBREF
000261                NSGGPC(I) = NSGGPC(I)+NSEGGP(I)                           SUBREF
000262            300 NSBGPC(I) = NSBGPC(I)+NSUBGP(I)                           SUBREF
000263          C                                                               SUBREF
000264          C ZERO OUT REGION ACCUMULATOR VALUES                            SUBREF
000265          C                                                               SUBREF
000266                DO 310 I = 1,11                                           SUBREF
000267            310 HVAL(I) = 0                                               SUBREF
000268          C                                                               SUBREF
000269          C TEST FOR END-OF-FILE OR RESET REGION I.D.                     SUBREF
000270          C                                                               SUBREF
000271                IF (IH.LT.0) GO TO 320                                    SUBREF
000272          C                                                               SUBREF
000273                REGION = IREGN                                            SUBREF
000274                GO TO 120                                                 SUBREF
000275          C                                                               SUBREF
000276          C MOVE COUNTRY VALUES TO PRINT AREA                             SUBREF
000277          C                                                               SUBREF
000278            320 HWA = HWAC                                                SUBREF
000279                TWA = TWAC                                                SUBREF
000280                XINBS = XINBSC/HWA                                        MOD3
000281                XSMBS = XSMBSC/HWA                                        MOD3
000282                XINCV = XINCVC/TWA                                        MOD3
000283                XSMCV = XSMCVC/TWA                                        MOD3
000284          C                                                               SUBREF
000285                DO 330 I = 1,3                                            SUBREF
000286                IF (I.EQ.3) GO TO 330                                     SUBREF
000287                NSEGGP(I) = NSGGPC(I)                                     SUBREF
000288            330 NSUBGP(I) = NSBGPC(I)                                     SUBREF
000289          C                                                               SUBREF
000290          C CALL REPRT1 WITH ITOTYP=4 FOR COUNTRY PRINT                   SUBREF
000291          C                                                               SUBREF
000292                ITOTYP = 4                                                SUBREF
000293                CALL REPRT1                                               SUBREF
000294          C                                                               SUBREF
000295                RETURN                                                    SUBREF
000296          C                                                               SUBREF
000297                END                                                       SUBREF
```

```
* EL, YLDERR,1,760513, 61010    , 1
000001            SUBROUTINE YLDERR                                                    YLDERR
000002          C                                                                      YLDERR
000003          C *** ROUTINE PRODUCES THE POPULATION YIELD ERROR REPORT(S)           YLDERR
000004          C                                                                      YLDERR
000005          C                                                                      YLDERR
000006          C *** COMMON STORAGE                                                   YLDERR
000007          C                                                                      YLDERR
000008            COMMON /CARDIN/RPTYPE,AUNITS,PARMTR(5)                               CARDIN
000009           *,              ICASIN(5),LEVEL,JPARM,IERR                          CARDIN
000010            INTEGER RPTYPE,AUNITS,PARMTR                                       CARDIN
000011          C                                                                      YLDERR
000012            COMMON /RANGE / START,INTVL1,BREAK1,INTVL2,BREAK2                    RANGE
000013           *,               INTVL3,STOP                                         RANGE
000014          C                                                                      RANGE
000015            REAL     INTVL1,INTVL2,INTVL3                                      RANGE
000016          C                                                                      YLDERR
000017            COMMON /FILBUF/ FILENM(2),ICNTRY,IREGN,IZONE,ISTRTA                  FILBUF
000018           *,               ISBSTA,ICASE,MXSEG                                  MODF
000019           *,               KBUF(125)                                           MODF
000020          C                                                                      FILBUF
000021            DIMENSION KWIN(4),IRIOWD(4),IPRDAT(14),PRDATE(6),XEPW(4),XEYLD(6)  FILBUF
000022           *,          APVAL(2)                                                FILBUF
000023           *,          MCERR(100)                                              MODF
000024          C                                                                      FILBUF
000025            REAL     MCREF,MCERR                                               MODF
000026            INTEGER FILENM,PRDATE                                              FILBUF
000027          C                                                                      FILBUF
000028            EQUIVALENCE (KBUF(1),NSEG,NT,KWIN(1))                              FILBUF
000029           *,  (KBUF(2),IGPNO,NSTRAT,NREGS)                                    MODF
000030           *,  (KBUF(3),XHPW,NZTOT)                                            MODF
000031           *,  (KBUF(4),AREA,NBK)                                              MODF
000032           *,  (KBUF(5),XTPW,NPD)                                              MODF
000033           *,  (KBUF(6),N,IRIOWD(1))                                           MODF
000034           *,  (KBUF(7),RA)                                                    MODF
000035           *,  (KBUF(8),XBTPW)                                                 MODF
000036           *,  (KBUF(9),XCV2)                                                  MODF
000037           *,  (KBUF(10),XTPWT,IPRDAT(1))                                      MODF
000038           *,  (KBUF(11),PRDATE(1))                                            MODF
000039           *,  (KBUF(17),XEPW(1))                                              MODF
000040           *,  (KBUF(21),XEYLD(1))                                             MODF
000041           *,  (KBUF(24),APVAL(1),MCREF)                                       MODF
000042           *,  (KBUF(25),MCERR(1))                                             MODF
000043           *,  (KBUF(27),XTYLD)                                                MODF
000044          C                                                                      YLDERR
000045            COMMON /PRNTID/ CNTRY,REGION,ZONE,STRATA,ICASEN                      PRNTID
000046           *,               FILENO,FILNAM(2)                                    PRNTID
000047           *,               ITOTYP,NAMFMT                                       PRNTID
000048          C                                                                      PRNTID
000049            INTEGER CNTRY,REGION,ZONE,STRATA,FILENO,FILNAM                     MODF
000050          C                                                                      YLDERR
000051            COMMON /HSTOGM/ XMVAL,STDDEV,XNPSUM,SDPSUM                         HSTOGM
000052           *,               NRANGE,IRANGE(51),NTOTL,RANGES(52)                  HSTOGM
000053           *,               MINVAL,MAXVAL,DATPRD(3),NZRNG,REF                   HSTOGM
000054          C                                                                      HSTOGM
000055            REAL     MINVAL,MAXVAL                                             HSTOGM
000056            INTEGER DATPRD                                                     HSTOGM
000057          C                                                                      YLDERR
000058            COMMON /PRTCTL/ NPAGE,NLINES,MAXLIN,KOUT,HEADER(16)                PRTCTL
```

```
000059              INTEGER HEADER                                                  PRTCTL
000060        C                                                                     YLDERR
000061              COMMON /SELCTN/ BIOWD(4),WPRTY(4),IPRD(3,14)                    SELCTN
000062        C                                                                     SELCTN
000063              INTEGER BIOWD,WPRTY                                             SELCTN
000064              COMMON /MATCH1/ MATCH                                                   *NEW
000065        C                                                                     YLDERR
000066              COMMON /READNG/ READNG(5600)                                    MODF
000067        C                                                                     MODF
000068              DIMENSION CLEWA(100),CLEPRD(100),CLATEC(100),CLPTEC(100)        MODF
000069             *,          CLATWC(100),CLPTWC(100)                            MODF
000070        C                                                                     MODF
000071              EQUIVALENCE                          (READNG(5001),CLEWA(1))  MODF
000072             *,  (READNG(5101),CLEPRD(1)),         (READNG(5201),CLATEC(1)) MODF
000073             *,  (READNG(5301),CLPTEC(1)),         (READNG(5401),CLATWC(1)) MODF
000074             *,  (READNG(5501),CLPTWC(1))                                    MODF
000075        C                                                                     YLDERR
000076        C *** LOCAL STORAGE                                                   YLDERR
000077        C                                                                     YLDERR
000078              DATA AREAPS /10289.712/                                         YLDERR
000079        C                                                                     YLDERR
000080              JWIN = 0                                                        YLDERR
000081              JPRD = 0                                                        YLDERR
000082        C                                                                     YLDERR
000083        C *** PROCESS FOR BIO-WINDOWS FIRST                                   YLDERR
000084        C                                                                     YLDERR
000085           10 IH = 1                                                        MODF2
000086              IF ( JWIN .LT. 0 )  GO TO 30                                    MODF2
000087        C                                                                     YLDERR
000088           15 JWIN = JWIN+1                                                 YLDERR
000089              IF (JWIN.GT.4) GO TO 20                                         YLDERR
000090              IF (BIOWD(JWIN).NE.0) GO TO 50                                  YLDERR
000091              GO TO 15                                                        YLDERR
000092        C                                                                     YLDERR
000093           20 JWIN = -1                                                     YLDERR
000094           30 JPRD = JPRD+1                                                 YLDERR
000095        C                                                                     YLDERR
000096              IF (JPRD.GT.14) GO TO 500                                       YLDERR
000097              IF (IPRD(1,JPRD).EQ.0) GO TO 500                                YLDERR
000098        C                                                                     YLDERR
000099        C                                                                     YLDERR
000100           50 DO 60 I = 1,500                                               YLDERR
000101           60 READNG(I) = 0.0                                               YLDERR
000102        C                                                                     YLDERR
000103              J = 0                                                           YLDERR
000104              KXS = 0                                                         YLDERR
000105              NTOTL = 0                                                       YLDERR
000106        C                                                                     YLDERR
000107              DO 70 I = 1,51                                                  YLDERR
000108           70 IRANGE(I) = 0                                                 YLDERR
000109              REFSIA = 0.0                                                    MODF2
000110              DELSIA = 0.0                                                    MODF2
000111              DELERR = 0.0                                                    MODF2
000112              DELSQ  = 0.0                                                    MODF2
000113              REF    = 0.0                                                    MODF2
000114              MINVAL = 1.E10
000115              MAXVAL = -1.E10
000116              IF ( IH .EQ. 0 )  GO TO 80                                      MODF2
000117        C                                                                     YLDERR
000118        C *** GET HEADER RECORD                                               YLDERR
```

```
000119          C                                                               YLDERR
000120                CALL FILSEG(IH)                                           MODF
000121          C                                                               YLDERR
000122                IF (IERR.NE.0) GO TO 500                                  YLDERR
000123          C                                                               YLDERR
000124                ICASEN = ICASE                                            YLDERR
000125                FILNAM(1) = FILENM(1)                                     YLDERR
000126                FILNAM(2) = FILENM(2)                                     YLDERR
000127                NAMENT = 0                                                YLDERR
000128          C                                                               YLDERR
000129                IH = 0                                                    YLDERR
000130          C                                                               YLDERR
000131          C *** GET DETAIL RECORDS                                        YLDERR
000132          C                                                               YLDERR
000133             80 CALL FILSEG(IH)                                           YLDERR
000134          C                                                               YLDERR
000135                IF (IH.LT.0) GO TO 200                                    YLDERR
000136          C                                                               YLDERR
000137                IF ( KXS .GT. 0 )  GO TO 100                              MODF2
000138                IF(MATCH .EQ. -2)GO TO 210                                      *NEW
000139          C                                                               YLDERR
000140                CNTRY = ICNTRY                                            YLDERR
000141                REGION = IREGN                                            YLDERR
000142                ZONE   = IZONE                                            YLDERR
000143                STRATA = ISTRTA                                           YLDERR
000144          C                                                               YLDERR
000145                GO TO 120                                                 YLDERR
000146          C                                                               YLDERR
000147          C *** TEST FOR CHANGE IN ID AREA                                YLDERR
000148          C                                                               YLDERR
000149            100 IF ( ISTRTA .NE. STRATA )  GO TO 200                      MODF2
000150                GO TO (105,110,120), LEVEL                                YLDERR
000151            105 IF ( IZONE .NE. ZONE .OR. IREGN .NE. REGION )  GO TO 200  MODF2
000152                GO TO 120                                                 YLDERR
000153            110 IF ( IREGN .NE. REGION )  GO TO 200                       MODF2
000154          C                                                               YLDERR
000155            120 XNA = NA                                                  YLDERR
000156                XN  = N                                                   YLDERR
000157                XR  = AREA/(XNA*AREAPS)                                   YLDERR
000158                XNR = XN*XR*AREAPS                                        YLDERR
000159                IF (KXS.GT.0) GO TO 180                                   YLDERR
000160                IF (JPIN.LT.0) GO TO 150                                  YLDERR
000161          C                                                               YLDERR
000162                DO 130 I = 1,6                                            YLDERR
000163                IX = 6-(I-1)                                              YLDERR
000164                IF (PRDATE(IX).EQ.0) GO TO 130                            YLDERR
000165                IDATE = PRDATE(IX)                                        YLDERR
000166                GO TO 180                                                 YLDERR
000167            130 CONTINUE                                                  YLDERR
000168          C                                                               YLDERR
000169          C                                                               YLDERR
000170          C *** PREDICTION DATE INPUT                                     YLDERR
000171          C                                                               YLDERR
000172            150 FLDA = IPRD(3,JPRD)                                       YLDERR
000173                LMO  = IPRD(2,JPRD)                                       YLDERR
000174                LYR  = IPRD(1,JPRD)                                       YLDERR
000175          C                                                               YLDERR
000176                CALL LEPA(FLDA,LMO,LYR,ALFGM,DAYS)                        YLDERR
000177          C                                                               YLDERR
000178                IDATE = DAYS                                              YLDERR
```

```
000179      C                                                                 YLDERR
000180           DO 160 I = 1,6                                               YLDERR
000181           IX = 6-(I-1)                                                 YLDERR
000182           IF(PRDATE(IX) .EQ. 0)GO TO 160                                        *NEW
000183           IF (PRDATE(IX).LE.IDATE) GO TO 180                           YLDERR
000184         160 CONTINUE                                                   YLDERR
000185      C                                                                 YLDERR
000186      C *** NO DATE MATCH                                               YLDERR
000187      C                                                                 YLDERR
000188           CALL PAGHDR(-2)                                              YLDERR
000189           WRITE (KOUT,6300) KPTYPE,JPARM,PARMTR(JPARM),(IPRD(I,JPRD),I=1,3) YLDERR
000190      C                                                                 YLDERR
000191           GO TO 30                                                     YLDERR
000192      C                                                                 YLDERR
000193         180 DELSTA = DELSTA+XTPW*XNR                                   YLDERR
000194           REFSTA = REFSTA+XTPW*XNR*XTYLD                               YLDERR
000195           KXS = KXS+1                                                  YLDERR
000196      C                                                                 YLDERR
000197           GO TO 80                                                     YLDERR
000198      C                                                                 YLDERR
000199      C *** CHANGE IN ID AREA                                           YLDERR
000200      C                                                                 YLDERR
000201         200 IF ( KXS .EQ. 0 )  GO TO 210                               MODF2
000202           J= J+1                                                       MODF2
000203           READNG(J) = (XEYLD(IX)-XTYLD)*DELSTA                         YLDERR
000204      C                                                                 YLDERR
000205           MINVAL = AMIN1(MINVAL,READNG(J))                             YLDERR
000206           MAXVAL = AMAX1(MAXVAL,READNG(J))                             YLDERR
000207      C                                                                 YLDERR
000208           DELERR = DELERR+READNG(J)                                    YLDERR
000209           REF    = REF+REFSTA                                          YLDERR
000210           DELSQ  = DELSQ+READNG(J)**2                                  YLDERR
000211      C                                                                 YLDERR
000212         210 CONTINUE                                                   MODF2
000213           DELSTA = 0                                                   YLDERR
000214           REFSTA = 0                                                   YLDERR
000215           KXS    = 0                                                   YLDERR
000216           IF ( IH .LT. 0 )  GO TO 250                                  MODF2
000217      C                                                                 YLDERR
000218           GO TO (220,230,240), LEVEL                                   YLDERR
000219      C                                                                 YLDERR
000220         220 IF ( IZONE .NE. ZONE )  GO TO 250                          MODF2
000221      C                                                                 YLDERR
000222         230 IF (IREGN.EQ.REGION) GO TO 80                              YLDERR
000223           GO TO 250                                                    YLDERR
000224      C                                                                 YLDERR
000225         240 IF (IH.EQ.0) GO TO 80                                      YLDERR
000226      C                                                                 YLDERR
000227         250 IF ( J .EQ. 0 )  GO TO 400                                 MODF2
000228           NTOTL = J                                                    MODF2
000229      C                                                                 YLDERR
000230           DO 390 I = 1,NTOTL                                           YLDERR
000231           PCDERR = 0.0                                                 MODF2
000232           IF ( REF .NE. 0.0 )  PCDERR = 100.0*READNG(I)/REF            MODF2
000233      C                                                                 YLDERR
000234           DO 280 K = 1,NRANGE                                          YLDERR
000235           IF (PCDERR.LE.0) GO TO 260                                   YLDERR
000236      C                                                                 YLDERR
000237           IRANGE(LZRNG) = IRANGE(NZRNG)+1                              YLDERR
000238           GO TO 300                                                    YLDERR
```

```
000239          C                                                                       YLDERR
000240            260 IF (PCDERR.GE.RANGES(K+1)) GO TO 280                              YLDERR
000241          C                                                                       YLDERR
000242                IRANGE(K) = IRANGE(K)+1                                           YLDERR
000243                GO TO 300                                                         YLDERR
000244          C                                                                       YLDERR
000245            280 CONTINUE                                                          YLDERR
000246            300 CONTINUE                                                          YLDERR
000247          C                                                                       YLDERR
000248                DATPRD(1) = JWIN                                                  MODF2
000249                IF (JWIN.GT.0) GO TO 310                                          MODF2
000250                CALL FZULU(IDATE,DATPRD)                                          YLDERR
000251            310 CONTINUE                                                          MODF2
000252          C                                                                       YLDERR
000253                TJ = NTOTL                                                        YLDERR
000254                XMVAL = 0.0                                                       MODF2
000255                IF ( REF .NE. 0.0 )  XMVAL = 100.0/(TJ*REF)*DELERR                MODF2
000256                STDDEV = 0.0                                                      MODF2
000257                IF(DELERR**2/TJ .GT. DELSQ)GO TO 390
000258                IF ( J .GT. 1 .AND. REF .NE. 0.0 )  STDDEV = 100.0/REF*           MODF2
000259               *  SQRT ( 1.0/(TJ-1.0)*(DELSQ-1.0/TJ*DELERR**2) )                  MODF2
000260            390 XMPSUM = TJ*XMVAL                                                 YLDERR
000261                SMPSUM = SQRT(TJ)*STDDEV                                          YLDERR
000262          C                                                                       YLDERR
000263                CALL REPRT2                                                       YLDERR
000264          C                                                                       YLDERR
000265                IF ( IH .EQ. 0 )  GO TO 50                                        MODF2
000266                GO TO 10                                                          YLDERR
000267             400  CONTINUE                                                       MODF2
000268                IF ( JWIN .LT. 0 )  GO TO 420                                     MODF2
000269                CALL PAGHDR (-2)                                                  MODF2
000270                WRITE (KOUT,6310)  RPTYPE,JPARM,PARMTR(JPARM),JWIN                MODF2
000271                IF ( IH .EQ. 0 )  GO TO 50                                        MODF2
000272                GO TO 10                                                          MODF2
000273          C                                                                       MODF2
000274            420 CALL PAGHDR (-2)                                                  MODF2
000275                WRITE (KOUT,6300)  RPTYPE,JPARM,PARMTR(JPARM),JWIN                MODF2
000276                IF ( IH .EQ. 0 )  GO TO 50                                        MODF2
000277                GO TO 10                                                          MODF2
000278          C                                                                       YLDERR
000279            500 RETURN                                                            YLDERR
000280          C                                                                       YLDERR
000281          C *** ERROR MESSAGE                                                     YLDERR
000282          C                                                                       YLDERR
000283             6300 FORMAT (54H *** INPUT PREDICTION DATE DOES NOT MATCH DATE ON FILE/YLDERR
000284                 *        12H     RPTYPE=,I2,8H PARMTR(,I1,2H)=,I1,12H INPUT DATE=, YLDERR
000285                 *        3I2)                                                    YLDERR
000286             6310 FORMAT (43H *** NO INFORMATION FOR BIO-WINDOW SELECTED/         MODF2
000287                 *        12H     RPTYPE=,I2,8H PARMTR(,I1,2H)=,I1,12H BIO-WINDOW=, MODF2
000288                 *        I1)                                                     MODF2
000289                   END                                                           YLDERR

      END CUR
```