TRW NOTE NO. 74-FMT-939 25990-H028-R0-00

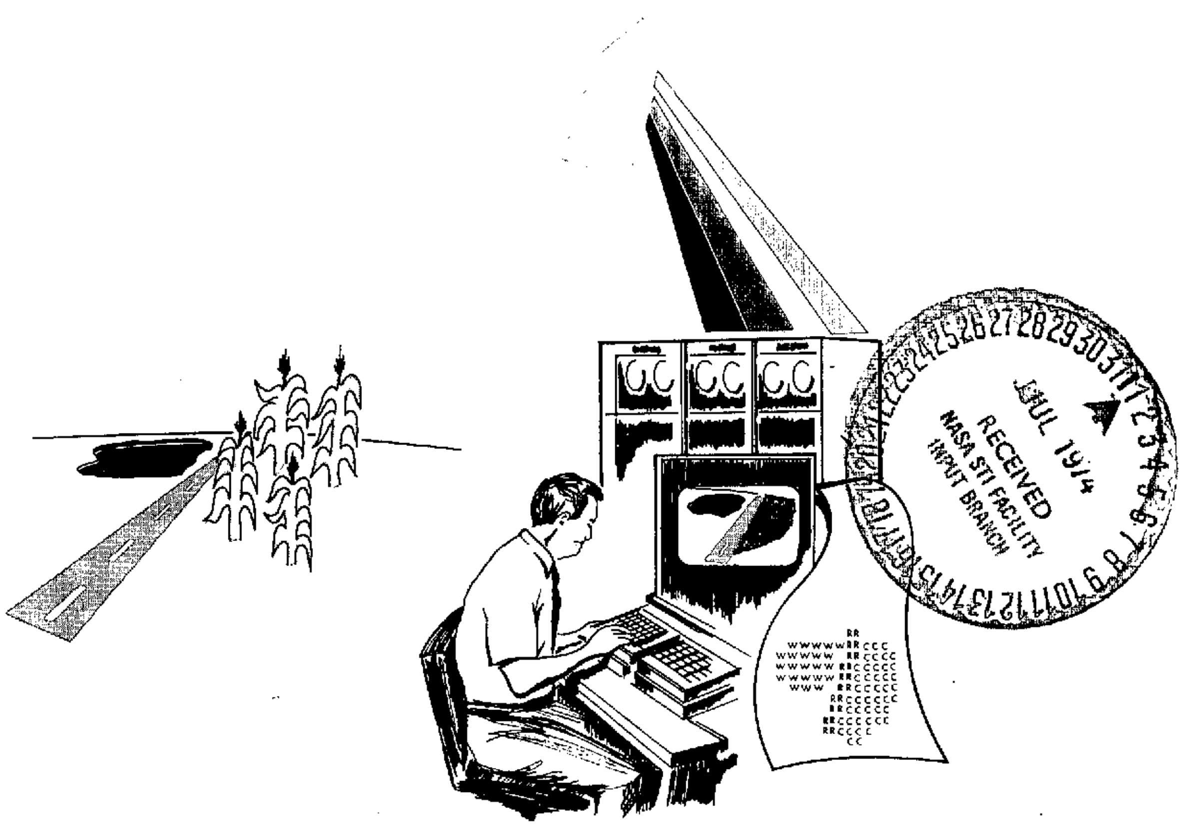
CR134303

PROJECT EARTH RESOURCES
TASK JSC/TRW 601

N74 26713

ASTEP User's Guide And Software Documentation

15 MAY 1974



RECEIVED
NASA STI FACILITY
INPUT BRANCH
JUL 1974

Prepared for
MISSION PLANNING AND ANALYSIS DIVISION
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
JOHNSON SPACE CENTER
HOUSTON, TEXAS
NAS9-13834

TRW
SYSTEMS GROUP

25900-H028-R0-00

TRW NOTE NO. 74-FMT-939

PROJECT EARTH RESOURCES
TASK JSC/TRW 601

ASTEP USER'S GUIDE AND SOFTWARE DOCUMENTATION

15 MAY 1974

Prepared for
MISSION PLANNING AND ANALYSIS DIVISION
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
JOHNSON SPACE CENTER
HOUSTON, TEXAS
NAS9-13834

Approved by: Pete Salvato Jr.
Pete Salvato, Jr.
Manager
JSC/TRW Task 601

Approved by: Robert H Kidd
R. H. Kidd
Assistant Project Manager,
Earth Resources
Mission Trajectory Control
Project

FLB
F.L.B

EDS
E.D.S.

ACKNOWLEDGEMENTS

This document was prepared by the following JSC/TRW Task 601 personnel listed alphabetically:

A. S. Gliniewicz
H. M. Lachowski
W. H. Pace, Jr.
P. Salvato, Jr.

We also wish to acknowledge Mr. A. D. Wylie , of the Mission Planning and Analysis Division, Johnson Space Center, for his technical direction on the development of ASTEP and his contribution to the preparation of this document.

PRECEDING PAGE BLANK NOT FILMED

ABSTRACT

This document constitutes the user's guide and software documentation for the Algorithm Simulation Test and Evaluation Program (ASTEP) as of 15 May 1974.

ASTEP is a modular computer program developed by TRW Systems for JSC for the purpose of testing and evaluating methods of processing remotely sensed multispectral scanner earth resources data. ASTEP is writtin in FORTRAND V on the UNIVAC 1110 under the EXEC 8 operating system and may be operated in either a batch or interactive mode. The program currently contains over one hundred subroutines consisting of data classification and display algorithms, statistical analysis algorithms, utility support routines, and feature selection capability.

The current program can accept data in LARSC1, LARSC2, ERTS, and Universal formats. The program can output processed image or data tapes in Universal format.

PRECEDING PAGE BLANK NOT FILMED

TABLE OF CONTENTS

PRECEDING PAGE BLANK NOT FILMED

1. INTRODUCTION

This report presents the user's guide and the software documentation for the Algorithm Simulation Test and Evaluation Program (ASTEP). The function of ASTEP is to serve as a tool to perform experiments with multispectral scanner data. The purpose of these experiments is to gain understanding of the problems associated with processing multispectral earth resources data and to test and evaluate processing algorithms. The major emphasis is to examine the statistical properties of the data and their impact upon classification algorithms. Examples of the experiments of this type include the following investigations:

a) Determining quantitatively the variation in spectral signatures for a given situation.

b) Determining if there are patterns in the signature variations - either spectrally or spatially.

c) Determining the statistical homogeneity of typical ground truth sites.

d) Determining if the statistical assumptions required for maximum likelihood processing of typical areas are satisfied.

e) Evaluating the performance of various clustering techniques.

f) Comparing the performance of clustering and maximum likelihood algorithms.

The ultimate purpose of experiments of this type is the development of new processing ideas and algorithms. The program is highly modular; therefore new processing routines may be added with minimal restructuring of the program.

A general overview of ASTEP is presented in Section 2. Section 3 gives the option descriptions. For each option and suboption, this section presents (a) its usage, (b) a test case, (c) the engineering description, and (d) the flow charts. Section 4 gives the subroutine purposes and dependencies. The JSC EXEC 8 system control cards required to execute ASTEP are listed in Section 5. Section 6 presents several complete illustrative ASTEP runs. Section 7 provides references for the various algorithms.

ASTEP is an evolving tool for the analysis of multispectral data. This user's guide describes the program as it currently exists. As new capability is added, it is the intent to conform to the general input rules and user cueing methods discussed in Section 2.2. The user's guide itself is modular and will be updated as required.

While ASTEP may be used in either a batch or an interactive mode, the inputs are designed to be convenient for the interactive mode , since the interactive mode is more appropriate for experimental analysis. When ASTEP is executed in the batch mode, inputs must be selected prior to execution and the user does have the benefit of the input prompters, supplied in the interactive mode, to aid in organizing his inputs.

2. ASTEP OVERVIEW

This section presents an overview of ASTEP. Its overall structure, options, layout, operational status, and general input rules are discussed.

2.1 OVERALL STRUCTURE

ASTEP consists of two basic parts: a driver and a set of applications modules. The program uses a set of data files that depend upon the particular applications modules selected by the user. The driver serves several functions. It is the holder of the common storage areas and transfers control to the appropriate applications module. Section 2.3 gives a brief description of each of the applications modules or options in the program.

Since the last program user's guide (User's Guide and Software Documentation for ASTEP, April 16, 1973), a number of new capabilities have been added to the program. A brief description of these new capabilities is given in Section 2.4.

ASTEP uses a number of data files for temporary and/or permanent storage. The assignment of the physical device (tape or disc) for the files is handled via the operating system control cards. Figure 2.1 gives a schematic representation of the use of these files. For convenience, all the files are shown by a tape symbol in Figure 2.1. Also, not all the possible interconnections and options are shown.

In general, the major communication between various application options in the program is via the unit which contains the reformatted data (DATUNT), the unit which contains - or will contain - the corresponding image array (IMGUNT), and the signature files (ISIGF1 and ISIGF2). All of the classification procedures act upon the reformatted data with the appropriate algorithm and generate the corresponding image array or map. The statistical modules act upon subsets of the data base defined by the appropriate characters in the image array and generate the requested statistics. The purpose of the signature files is to save, for later retrieval, spectral signature data.

2.2 GENERAL INPUT RULES

Upon execution of ASTEP, the user is required to select the desired print control for the run from ECHO or NOECHO. Selection of ECHO will cause the user's inputs to be printed out as they are read by the program. This print option will cause the output from batch runs to closely resemble the output from interactive runs. Selection of NOECHO will cause the user inputs to be suppressed and would normally be used in interactive (demand) runs where user inputs are typed in and do not need to be seen a second time.

The options and suboptions of ASTEP are selected by the appropriate Hollerith name, for example the commands DATDEF, ADPCLU, etc. defined in Section 2.3. Those options which require parameter values to be specified have namelist inputs. The general form of the namelist input is

NAMEXX - option name

$INNAME - namelist name consist of first four letters of option name plus the prefix IN

Upon completing the namelist reading the program will print out all of the numerical values in the namelist. At this point the user will have a choice whether to accept the values displayed or whether to change them. The program will require the user to input a YES if the values are to be accepted. Any other response will cause the program to re-read the namelist inputs and to repeat the cycle.

In addition, several options require file numbers and class descriptions to be entered. These are entered as Hollerith characters.

In the interactive or terminal mode when the program is waiting for an input, it will inform the user. A sufficient cue is given so that the user is made aware of the choices at that point or the parameters that need to be specified. These same cues are printed out in the batched mode, but in that case the user of course must anticipate the requests.

2.3 OPERATIONAL STATUS

Each of the operations or modules listed below exists within the current ASTEP. The program executes in both the batched and demand terminal modes. The program inputs and outputs for the batched and terminal executions are identical, with the possible exception of the ASTEP print options noted previously.

- DATA CLASSIFICATION AND DISPLAY ALGORITHMS

  Iterative Clustering (ITRCLU), An unsupervised classification procedure based upon an iterative clustering algorithm. Multiple passes through the data are required.

  Adaptive Clustering (ADPCLU), A fast clustering algorithm which may be used as a starter for iterative clustering or as a separate classification/data analysis procedure.

  Quantization (QUANTZ), Generates a grey scale type map via quantization of a single data channel. Intensity intervals are assigned unique characters.

  Maximum Likelihood (MAXLIK), Data classification via the maximum likelihood algorithm.

  Images (IMAGES), Displays character maps or images developed by all image generating options - for example iterative clustering, adaptive clustering, etc. A thresholding capability for display is available.

  Difference Images (DIFIMG), Differences two images - pixel by pixel - to produce a third image. The differencing rules or symbol equivalences are interactively defined by user.

  Training Field (TRNFLD), Classifies each field with a class number equal to the field number. The usual use of TRNFLD is to define training fields for signature computation to be used with maximum likelihood classification.

- DATA STATISTICAL ANALYSIS

  Factor Analysis (FACTOR), Computes statistics - mean vector and covariance matrix - of data subset, cluster, or unions of clusters. Determines eigenvalues and eigenvectors of the covariance matrix.

  Histogram (HSGRAM), Computes and displays a one, two, or three dimensional histogram of the data subset, cluster, or unions of clusters.

  Edit Signature (EDTSIG), Spectral signature - means and covariances - file utility routines. Allows user to save, retrieve, display, analyze, input, add, etc., the mean and covariance matrix data.

Compare (COMPAR), Compares regions in the data space occupied by various data subsets.

- FEATURE SELECTION

  Feature Selection (FEATSL), Allows user to analyze the feature selection problem. Determines best linear transformation to reduce the dimension of data to be processed - usually via the maximum likelihood option - or the best channel subset. The linear transformation may be applied in the TRNSFM option. Generates and displays the separability-to-be-gained map.

- UTILITY OPTIONS

  Data Definition (DATDEF), Allows user to define data subset from raw data tape to be processed as described subsequently by user. Multiple fields and channel subsets may be selected (LARSC1, LARSC2, ERTS, or Universal format data).

  Initialize Header (INTHDR), Allows user to restart with data subset defined by a previous use of data definition. Data subsets which can be processed by the ASTEP options may be saved and then Initialize Header is used to restart.

  Units (UNITS), Allows user to change any of the unit assignments used by the program.

  Copy Data (CPYDAT), Used to extract a subset of an entire packed data set for input to the DATDEF option. Copies a subset of the packed raw data tape onto a disk file on tape. This is a user convenience option which may be used to operate from a file rather than a tape.

  Tape Dump (DUMP), This option allows the user to print out contents of selected portions of packed data tapes in an integer form for interpretation by the user.

  Universal Write (UVWRIT), This option provides the capability to convert data (multispectral data or processed image data) from internal ASTEP format to Universal format and to write the data on tape or file.

Transform (TRNSFM), Enables the user to scale, translate, and transform multispectral data in ASTEP format. For example, it can be used in conjunction with the DATDEF and FEATSL options to reduce the original data channels to the "best" subset of channels or the "best" linear combinations of channels (best for maximum likelihood classification of the data).

o PROGRAM INFORMATION OPTIONS

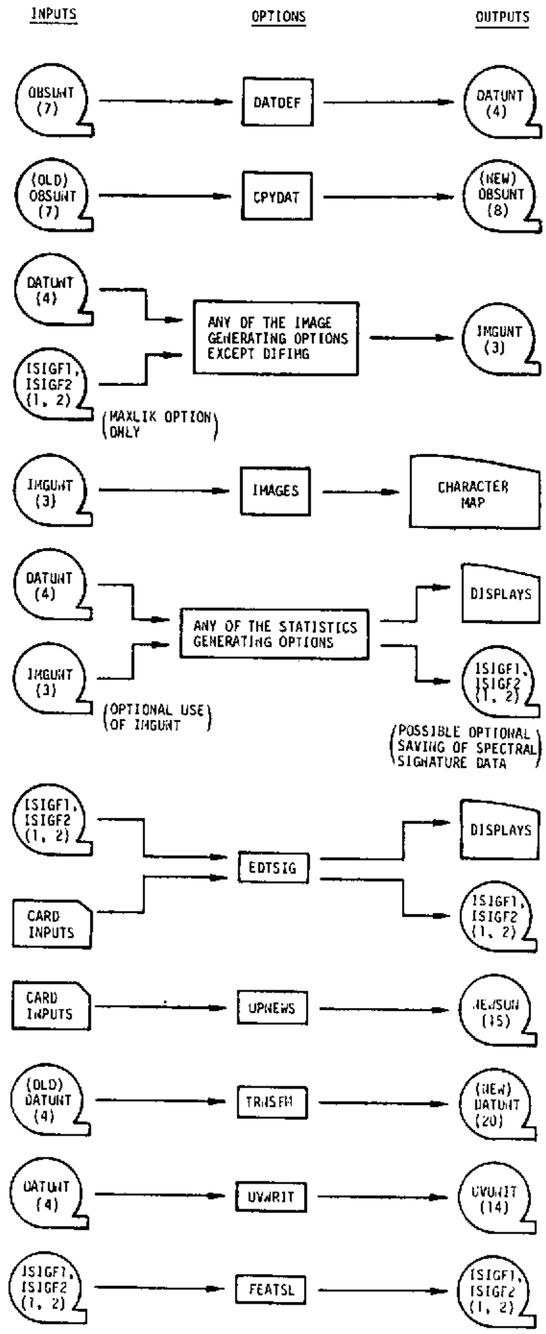
News (NEWS), Used to inform the user of any changes or modifications to the program.

Update News (UPNEWS), This option allows the user to create or update information in the News file.

2.4 NEW ASTEP CAPABILITIES ADDED SINCE PREVIOUS USER'S GUIDE

Since the last user's guide, a number of new capabilities have been added to ASTEP. Some changes were made to improve the overall efficiency of the program and will be transparent to the user. Other changes will be apparent to the user. These are changes in inputs to ASTEP options and new options.

Since the last user's guide, the options NEWS, UPNEWS, TRNSFM, TRNFLD, DUMP, and UVWRIT have been added. The DATDEF option has been modified to read Universal formatted tapes. The IMAGES option has been completely changed to include the suboptions STATUS, THRESH, SYMBOL, ALLCLS, ECHCLS, SUBSET, BORDER, and INSIDE. The ASTEP print options ECHO and NOECHO have been added. An error recovery capability has been added to prevent the program from terminating when errors are made in the namelist input.



CODE

OBSUNT - RAW OR PACKED OBSERVATION DATA TAPE

DATUNT - REFORMATED DATA TAPE WHICH CONTAINS ONLY THE USER DEFINED DATA OF INTEREST

IMGUNT - IMAGE AND CORRESPONDING THRESHOLD DATA AS GENERATED BY ONE OF THE CLASSIFICATION ALGORITHMS

ISIGF1 ISIGF2 - SPECTRAL SIGNATURE FILES

NEWSUN - FILE USED BY NEWS AND UPNEWS OPTIONS

UVUNIT - OUTPUT FILE FOR UNIVERSAL FORMAT MSS DATA OR IMAGE DATA

Figure 2.1 Schematic Representation of ASTEP's Use of Data Files

3. OPTION DESCRIPTION

This section contains an engineering description and a usage description of each of the options and suboptions of ASTEP. Each of the major options is presented separately and is identified by its Hollerith code. They are presented in the following order:

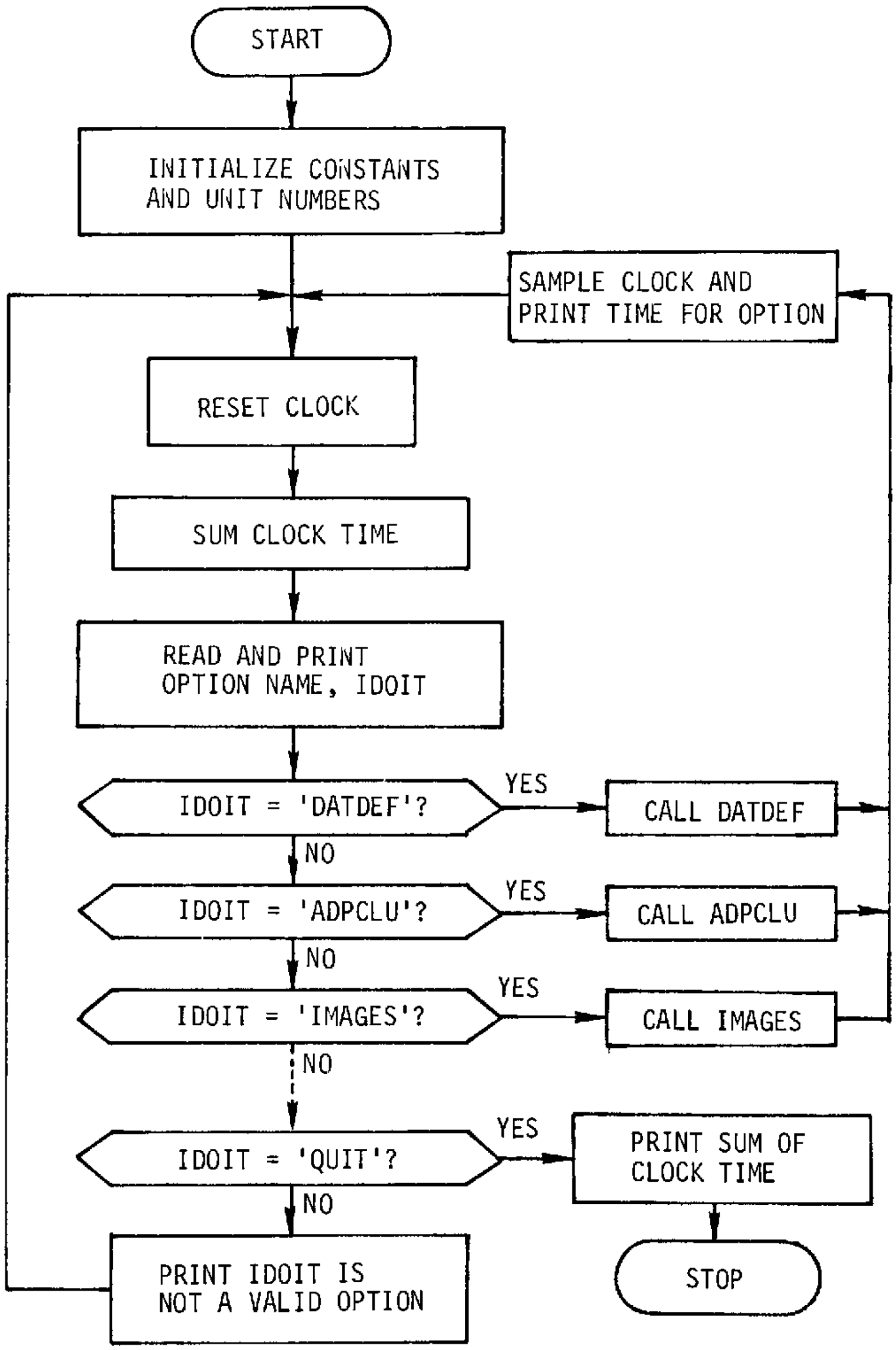
DATDEF - Data definition
ADPCLU - Adaptive clustering
COMPAR - Compare
CPYDAT - Copy data
DIFIMG - Difference images
DUMP - Dump tape
EDTSIG - Edit signature
FACTOR - Factor analysis
FEATSL - Feature selection
HSGRAM - Histogram
IMAGES - Images
INTHDR - Initialize header
ITRCLU - Iterative clustering
MAXLIK - Maximum likelihood
NEWS - News
QUANTZ - Quantize
TRNFLD - Training field
TRNSFM - Transform
UNITS - Units
UPNEWS - Update news
UVWRIT - Universal write
QUIT - Quit

Each subsection begins with a description of the use of the option and a sample case. The sample cases were run in the batch mode with the ECHO option. Thus, the user's namelist inputs are not shown in the samples, but the namelist values used by the program are printed. Following each sample is an engineering description and flow chart of the option and any suboptions.

Any flow charts for subroutines which are common to several options are included in Section 4.

Each option is called by ASTEP when its code name is input, as shown in the following flow chart.

ASTEP DRIVER FUNCTIONAL FLOW



ASTEP 1 of 1

Using the DATDEF Option

The DATDEF option performs the following five basic functions:

1) Reads data in one of several formats

2) Selects a subset of the data to be processed based on user inputs

3) Optionally scales and translates the data subset

4) Changes the data subset format to ASTEP internal format

5) Writes the data subset on a file for ASTEP processing.

The DATDEF option takes the raw packed data from tape or disk file, unpacks and processes selected portions of it, and writes these portions onto a tape or disk file for subsequent ASTEP options to read. DATDEF is usually the first option executed, but may be bypassed if the unpacked data tape is mounted on, or the disk file assigned to, the unit corresponding to DATUNT and the INTHDR option is executed. The user may determine up to 10 selected data portions by specifying the scan line and pixel field limits. The user also selects the channels from which data is to be included.

The data may be translated and scaled using:

$$Y_{NEW} = A + B\ Y_{OLD} \qquad (1)$$

where Y_{OLD} is the raw pixel value, Y_{NEW} the translated and scaled pixel value, and A and B are scalar constants. This feature is intended to facilitate the inversion of data whose magnitudes are inversely related to the magnitudes of the observation values.

The packed observation data may be in one of four formats indicated by the value of ITPFMT. Table 1 provides a schematic representation of the data portion of each record assuming M pixels per channel and N channels in each record.

Any packed data tape or file of one of the specified formats may be used as the source tape. It must be mounted on the physical unit corresponding to the value of OBSUNT. The particular characteristics of three specific tapes, i.e., record sizes, error record numbers and sizes, etc., are prestored and are available within the DATDEF option (see Table 2). The set of tape characteristics used is determined by the value of ITPNO input by the user. If ITPNO is input with a value less than or equal to zero, the user must input the values of all these characteristics. If ITPNO is input equal to 5, the program assumes that the packed data tape input was generated by the CPYDAT option. The data tape characteristics and size of its data subset are printed. The user should check the scan lines of his data fields to be sure that they lie within the data region on tape.

The input sequence to the DATDEF option is as follows. First the namelist $INDATD is input:

$INDATD

NFIELD - Number of data fields to be specified, a maximum of 10

ITPFMT - Input data tape format indicator (see Table 1)

ITPNO - Input data tape characteristic indicator,

≤ 0	implies that all characteristics must be input and it is required to input values for constants A and B
= 1,2, or 3	implies that one of the sets of prestored characteristics will be used, see Table 2. The input of values for constants A or B is optional.
= 4	not used
= 5	implies that the packed input data tape read was generated by the CPYDAT option. K(24), ITPFMT, A, and B are read from tape and user inputs are ignored.
> 5	illegal, program stops

K - Desired channel numbers, a maximum of 24 are permitted

A - Translational scalar offset constant (see Table 2 for default values and equation (1))

B - Scaling constant (see Table 2 for default values and equation (1))

IDEVCE - NTRAN device error suppression flag

= 0	implies no error suppression, default value
$\neq 0$	implies that all NTRAN device error codes encountered while reading OBSUNT will be ignored. This value should be used if the record length of the input tape is not an integral multiple of computer words.

At this point the user may respond with YES to indicate that the printed input values are what is desired or NO and input the $INDATD namelist again.

A value of 5 for ITPNO implies that the unpacked data set to be processed is to be read from a tape or file (designated DATUNT) which was previously generated by the CPYDAT option. The values of the variables ITPFMT, NWRN, NWR1, LEAD, MAXJ, MAXK, A, B, and the starting scan line number are read from the header record of this tape and override any user input values. The values of these quantities and the range of scan line numbers on this tape or file are printed to emphasize to the user that these variables now have new values and the size of the data region. If the user has input scan line designations that fall outside those on the input data region, the program sets the values of the scan lines to be extracted to the nearest scan line number that is in the data region, prints a warning message, and continues processing.

It should be noted that under the present system if records are read from a data tape which are not an even number of 36 bit computer words long, a frame count error is generated. This frame count error is reflected in a device error code -3 being returned from the system routine NTRAN. Without using a different input/output system there is no way to differentiate between a legitimate device error and a frame count error. So the user is allowed to suppress NTRAN device errors, by inputting IDEVCE ≠ 0, when it is known that the input data tape will cause frame count errors. However, all device errors on the input data tape will be ignored and the following message is printed to make sure the user is aware of this fact: THE USER HAS REQUESTED THAT ALL NTRAN DEVICE ERRORS ON UNIT nn BE SUPPRESSED, where nn is the physical unit number of the input data tape.

Next the field data is required. The entire sequence of fields is input and then the user is asked if the values are acceptable. If the user indicates that the values are not acceptable, he will be asked the number of the data field to be changed. After inputting all of the values for that field, he is asked the number of the next data field to be changed. After he has changed all of the data fields desired, he will enter the data field number zero. This terminates the data reading mode. A summary of all input data for all fields is printed, and the user is again asked if it is acceptable. When several data fields are input, the default values for each field are the parameter values input for the previous field. The data for each field is specified by the namelist $INFLDD whose definition follows:

$INFLDD

- ISTART - Starting scan line number
- IINC - Number of scan lines after ISTART to be considered, equal to the total number of scans considered minus 1
- ISKIP - Number of scan lines to be skipped between accepted scans. The ISTART scan is always accepted.
- JSTART - Number of the first pixel requested from each scan
- JINC - Number of pixels after JSTART to be considered, equal to the total number of pixels considered minus 1.
- JSKIP - Number of pixels to be skipped between accepted pixels. The JSTART pixel is always accepted.

At this point, the user is asked if the values in $INFLDD are acceptable. A response of NO allows the user to change any or all of the values in the namelist. A response of YES continues execution.

Next, the value of ITPNO is checked. If it is less than or equal to zero, the characteristics of the particular input data tape mounted on unit DATUNT must be supplied. They are input under namelist $SPTAPE.

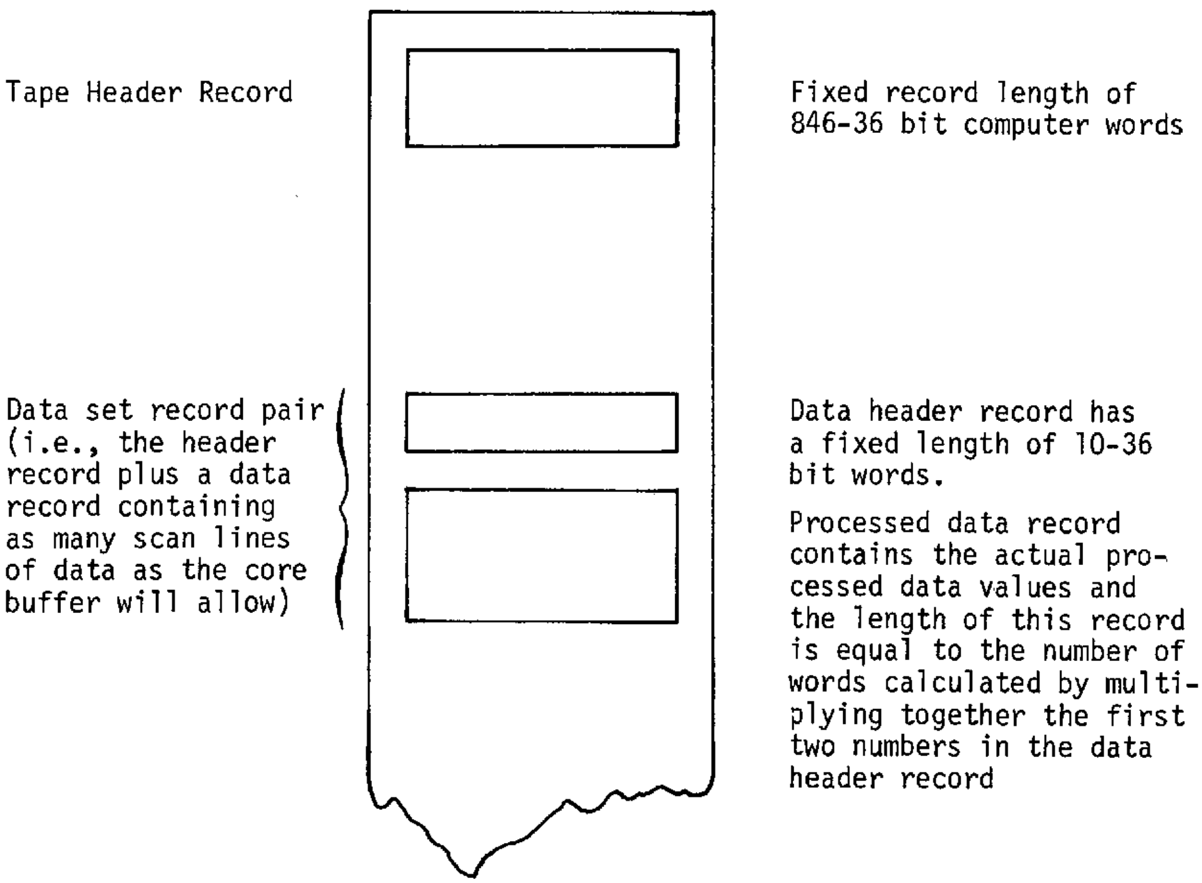
$SPTAPE

- NWRN - Number of words in the valid data record
- NWR1 - Number of words in the first record
- NWER1 - Number of words in the first error record (record will be read and ignored
- NERR1 - Number of first error record
- NWER2 - Number of words in the second error record
- NWERR2 - Number of second error record
- NWER3 - Number of words in the third error record
- NERR3 - Number of third error record
- LEAD - Number of bits to be ignored at the beginning of each data record
- MAXJ - Number of pixels per channel on each scan line
- MAXK - Number of channels per pixel on each scan line
- A - Translation coefficient (see equation (1))
- B - Scaling coefficient (see equation (1))

The input is summarized by the program and the user is asked if it is acceptable. A response of NO will allow the user to change any or all of the values in $SPTAPE. A response of YES continues execution.

This completes the required user inputs. The program then unpacks the MSS data and writes it in ASTEP internal format, unpacked and reformatted on the data unit (DATUNT).

This output data tape or file of unpacked and processed data is written using the system subroutine NTRAN and therefore under the constraints of the EXEC 8 operating system may not be read using any other method during the execution of this program. The data is processed in the sense that it may have been translated, scaled, or transformed from the raw form. This tape or file format is compatible with all other program options requiring it as input through the use of the RETDAT subroutine.



The tape header record has the following contents. The first 166 words of this record are identical with the contents of the variables in the /HEADER/common block:

NFIELD	-	Number of data fields
K(24)	-	Number of the data fields extracted
ITPFMT	-	Input tape format indicator
ISTARD(10)	-	Starting scan line numbers for each data field
ISKIPD(10)	-	The number of scan lines skipped between successive ones for each field
IINCD(10)	-	One less than the total number of scan lines considered for each data field
JSTARD(10)	-	The starting pixel number used for each field
JSKIPD(10)	-	The number of skipped pixels between successive pixels for each field
JINCD(10)	-	One less than the total number of pixels per scan line for each field
NRPF(10)	-	Number of data record pairs per data field
NPXPS(10)	-	Number of pixels per scan for each data field
IBUF1	-	
IBUF2	-	These are indices of the scratch array BUF of the common block BUFFER calculated in such a way that there is enough room for the temporary storage of unpacked, processed data between the start of the array and BUF(IBUF1). Also there must be enough room between BUF(IBUF1) and BUF(IBUF2), and between BUF(IBUF2) and BUF(NBUFSZ) to store at least one unpacked data record in each area.
NBUFSZ	-	The total number of cells of scratch storage available in array BUF for the DATDEF option.
ND	-	The total number of valid data channels.
NRT	-	The total number of output data record pairs for all data fields.
SPARE(25)	-	Extra storage for future expansion

The remaining words of the 846 word header are a copy of the UCCT header described in detail in Reference 1. If the packed data input is not in the UCCT format these cells will be zero.

The data header record is always ten words long. The value in the first word is the number of pixels in the record. The second word is the number of data channels. The other eight words are set to zero and left for future use.

The actual data values are stored in the second record of this pair. This record contains a number of processed pixel values, one per word, equal to the product of the first two numbers in the data header record. These pixel values are stored "pixelwise", that is, the first pixel value for each channel followed by the second pixel value for each channel, continuing until the total number of required pixels is reached.

Table 1. Data Portion of Record Format

ITPFMT	Type		
1	LARS1	Pixel	1,______,1 2,_____,2 3,_____,3 ... M,_____,M
		Channel	1,2,3,___,N 1,2,___,N 1,2,___,N ... 1,2,___,N
2	LARS2	Pixel	1,2,3,___,M 1,2,___,M 1,2,___,M ... 1,2,___,M
		Channel	1,______,1 2,_____,2 3,_____,3 ... N,_____,N
3*	ERTS	Pixel	1,2, 1,2, 1,2, 1,2, 3,4, 3,4,...., M-1, M, M-1, M
		Channel	1,1, 2,2, 3,3, 4,4, 1,1, 2,2,...., 3, 3, 4, 4
4**	UCCT	Pixel	1,2,3,___,M 1,2,___,M 1,2,___,M ... 1,2,___,M
		Channel	1,______,1 2,_____,2 3,_____,3 ... N,_____,N

* This format assumes four channels

** The detailed description of this format can be found in section 6, volume II of Reference 1. While Reference 1 does describe two formats, one, where all pixel values are given for one channel at a time, and a second, where the values of each pixel are given for all channels, one pixel at a time, only the former type is recommended. This is the format implemented in ASTEP.

Table 2. Input Data Tape Characteristics for ITPNO Values

Name	Definition	ITPNO Values	1	2	3
NWRN	Number of words in each valid data record		609	720	*
NWR1	Number of words in the first, ID., record		32	178	9
NWER1	Number of words in the first error record, the contents of the record are ignored		1	0	139[Γ]
NERR1	Number of the first error record		8	0	2
NWER2	Number of words in the second error record		0	0	0
NERR2	Number of the second error record		0	0	0
NWER3	Number of words in the third error record		0	0	0
NERR3	Number of the third error record		0	0	0
LEAD	Number of bits to be ignored at the beginning of each data record		32	32	0
MAXJ	Number of pixels per channel in a scan line		N.R	*	N.R.
MAXK	Number of channels of data in each scan line		12	N.R.	†
A	Translational offset constant, see equation (1)		0.	0.	0.
B	Scaling constant, see equation (1)		1.	1.	1.

* - Computed from data in the first record

† - Required to be 4

N.R. - Not required

Γ - This is the second record and is treated as an error record

Table 3. Warning and Error Messages

A NOT INPUT
A SET TO 0 AND B SET TO 1

'A' must be input for ITPNO $\leq$ 0. Processing continues.

B NOT INPUT
A SET TO 0 AND B SET TO 1

'B' must be input for ITPNO $\leq$ 0. Processing continues.

DUPLICATE CHANNEL SELECTED - ONE CHANNEL DELETED

The user has specified one channel twice in K. The duplicate channel is deleted and processing continues.

ERROR eee ON CPYDAT TAPE

The NTRAN error 'eee' occurred while trying to read the header record from the packed data input tape on unit DATUNT which was previously generated by the CPYDAT option. The program stops.

ERROR eee ON UNIT iii RECORD rrrr

NTRAN error, 'eee', has occurred while trying to write record, 'rrrr', on DATUNT whose physical unit number is 'iii'. Execution is stopped.

ERROR READING FIRST RECORD, NTRAN ERROR CODE = eeee

An NTRAN error, 'eeee', has occurred while trying to read the first record of the packed data tape, OBSUNT. Execution is stopped.

INPUT FIELD (iiii) IS GREATER THAN NFIELD (jjj)

In attempting to correct data field input the user has supplied a field number, 'iiii', larger than the total number of fields, 'jjj'. The user is requested to input the data field number he wishes to correct again.

INPUT VALUES FOR A AND B IGNORED FOR ITPNO = 5.

This message reminds the user that when reading a packed data input tape generated by the CPYDAT option (i.e., implied by ITPNO = 5) the constants A and B input are overridden by values stored in the tape header record. Processing continues.

Table 3. Warning and Error Messages (Continued)

ITPNO VALUE (iii) IS TOO LARGE

The ITPNO value, 'iii', has no meaning for values greater than 5. The program stops.

NBUFSZ = nnnnnn IS TOO SMALL TO READ THE UNIVERSAL TAPE HEADER RECORD - ABORT RUN

The buffer size supplied to the DATDEF option, 'nnnnnn', is too small. This variable is set in the main program ASTEPX. The program stops.

NO CHANNELS SELECTED

No values of K have been input, the user is again asked for namelist input under $INDATD.

NTRAN ERROR eeee ON UNIT iii PROCESSING SCAN LINE lllll

The NTRAN error,'eeee', occurred while reading OBSUNT physical unit, 'iii', scan line number 'lllll'.

NUMBER OF PIXELS FOR FIELD nnn TIMES THE NUMBER OF CHANNELS kkk IS GREATER THAN THE BUFFER SIZE ssssss

The buffer is divided into three parts. The basic increment is the maximum number of words in a normal record, NWRN, and the total buffer size divided by the number of channels desired plus two. This increment provides a buffer large enough to read a standard record. Two parts are used for the image and distance data or the buffered scan lines for each field of the raw data. The remaining part is used to store unpacked data by scan line. At least one scan line must be able to reside in the last part. If this is not possible, the above message is written with the size of the buffer for this data and execution is stopped.

THE USER HAS REQUESTED THAT ALL NTRAN DEVICE ERRORS ON UNIT iiii BE SUPPRESSED

When the user expects frame count errors on the packed data input tape, OBSUNT unit number, iiii, he can suppress these NTRAN errors, but not without suppressing the sensing of all device errors. The user is reminded of this fact by this message; processing continues.

Table 3. Warning and Error Messages (Continued)

WARNING, IINC FOR FIELD fff REDUCED FROM iiiii TO jjjjj TO CONFORM TO CPYDAT TAPE

When reading an input tape or file generated by the CPYDAT option the number, 'iiiii', of the last scan line to be considered in data field 'fff' was too large and was reduced to the number of the last scan line on the tape or file 'jjjjj'. Execution continues.

WARNING, ISTART FOR FIELD fff INCREASED FROM iiiii TO jjjjj TO CONFORM TO CPYDAT TAPE

While reading an input tape or file generated by the CPYDAT option the number, 'iiiii', of the first scan line to be considered in data field 'fff' was too small and was increased to the number of the first scan line on the tape or file 'jjjjj'. Execution continues.

WARNING, ISTART FOR FIELD fff REDUCED FROM iiiii TO jjjjj TO CONFORM TO CPYDAT TAPE

While reading an input tape or file generated by CPYDAT option the number, 'iiiii', of the first scan line plus the number of scan lines desired for data field 'fff' was larger than the last scan line, 'jjjjj', and the number of the last requested scan was reduced to the number of the last scan on the tape or file.

DATDEF OPTION

SAMPLES OF INPUT AND CORRESPONDING OUTPUT:

SAMPLE 1: Two fields of data are read from the standard input unit (DATUNT).

```
ENTER ASTEP OPTION OR TYPE A BLANK
>DATDEF


                    DATDEF  OPTION
                    ==============


$INDATD  NFIELD, ITPFMT, ITPNO, A, B, K, IDEVCE
 NFIELD      2 ITPFMT   1 ITPNO   1 A        .0 B       1.0 IDEVCE   0
 CHANNELS SELECTED,   9  11  12
TYPE YES IF INPUTS ARE OK
>YES
$INFLDD ISTART,ISKIP,IINC,JSTART,JSKIP,JINC
INPUT  1 FIELD DATA
INPUT  2 FIELD DATA
FIELD ISTART ISKIP IINC JSTART JSKIP JINC
  1      10      1     5    75     0     3
  2      30      1     5    75     0     3

TYPE YES IF INPUTS ARE OK
>YES
 THE OPTION DATDEF REQUIRED        .5242 SECONDS OF CPU TIME.

 ----------------------------------------------------------
```

SAMPLE 2. The observation data is read from a file written using the CPYDAT option; this is specified by setting ITPNO to 5. In addition, Channel 4 is specified twice in K. A warning statement is printed to indicate that a duplicate channel number has been deleted. The user makes an error on the first field in this example and therefore indicates that these data are not correct. The user then enters the number of the field to be changed and the correct data for that field. When all the field data is correct, the user responds zero and YES to indicate that the data is correct. It is noted that this error example is shown in the batch mode, where in reality it would actually occur only in the interactive mode. It is shown here to familiarize the user with the sequence of instructions under these circumstances.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>DATDEF


                     DATDEF  OPTION
                     ==============


$INDATD   NFIELD, ITPFMT, ITPNO, A, B, K, IDEVCE
DUPLICATE CHANNEL SELECTED-ONE CHANNEL DELETED.
 NFIELD     3 ITPFMT   1 ITPNO   5 A        .0 B        .0 IDEVCE   0

 CHANNELS SELECTED,   1   4   9
TYPE YES IF INPUTS ARE OK
>YES
TAPE GENERATED BY CPYDAT ON UNIT   5
ITPFMT   1 STARTING LINE   795 FINAL LINE   825
NWRN   609 NARI    20 LEAD    32 MAXJ   222 MAXK    12 A    255.0 B -1.0
$INFLDD ISTART,ISKIP,IINC,JSTART,JSKIP,JINC
INPUT   1 FIELD DATA
INPUT   2 FIELD DATA
INPUT   3 FIELD DATA
FIELD ISTART ISKIP IINC JSTART JSKIP JINC
  1     800    35    20   400     0    59
  2     800     0    20   461     0    59
  3     800     0    20   400     1   119

TYPE YES IF INPUTS ARE OK
>NO
TYPE NUMBER OF FIELD TO BE CHANGED. ZERO TERMINATES SCAN.
>  1
$INFLDD ISTART,ISKIP,IINC,JSTART,JSKIP,JINC
INPUT   1 FIELD DATA
TYPE NUMBER OF FIELD TO BE CHANGED. ZERO TERMINATES SCAN.
>  0
```

```
FIELD ISTART ISKIP IINC JSTART JSKIP JINC
  1      800     0    20   400     0    59
  2      800     0    20   461     0    59
  3      800     0    20   400     1   119

TYPE YES IF INPUTS ARE OK
>YES
 THE OPTION DATDEF REQUIRED       1.3292 SECONDS OF CPU TIME.

 ------------------------------------------------------------------
```

SAMPLE 3. This example shows how the user may input the specific characteristics of the observation data tape or file that has been mounted for this particular data extraction. ITPNO is set to -1 and the values are entered using namelist $SPTAPE.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>DATDEF


                    DATDEF  OPTION
                    ==============


$INDATD  NFIELD, ITPFMT, ITPNO, A, B, K, IDEVCE
A NOT INPUT.
A SET TO 0 AND B SET TO 1.
 NFIELD     5 ITPFMT  1 ITPNO -1 A        .0 B       1.0 IDEVCE  0
 CHANNELS SELECTED,  1  5 11 12
TYPE YES IF INPUTS ARE OK
>YES
$INFLDD ISTART,ISKIP,IINC,JSTART,JSKIP,JINC
INPUT  1 FIELD DATA
INPUT  2 FIELD DATA
INPUT  3 FIELD DATA
INPUT  4 FIELD DATA
INPUT  5 FIELD DATA
FIELD ISTART ISKIP IINC JSTART JSKIP JINC
  1       10     0     39     30      0     49
  2       10     1     39     30      2     49
  3       10     0     39     81      0     59
  4      200     1    299     30      0     59
  5      500     1    199     50      0     59

TYPE YES IF INPUTS ARE OK
>YES
$SPTAPE NARN,NWR1,NWER1,NERR1,NWER2,NERR2,NWER3,NERR3
        LEAD,MAXJ,MAXK,A,B

STANDARD DATA RECORD LENGTH    609
ERROR RECORD *** LENGTH
       1             32
       8            610
       0              0
       0              0
LEAD   32 MAXJ  222 MAXK   12 A      63.00 B      -1.00
TYPE YES IF INPUTS ARE OK
>YES
 THE OPTION DATDEF REQUIRED       8.4556 SECONDS OF CPU TIME.

 ----------------------------------------------------------------
```

SAMPLE 4. This is an example of data extraction from an observation tape written in ERTS format as indicated when ITPNO is set to 3.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>DATDEF

                      DATDEF  OPTION
                      ==============

$INDATD  NFIELD, ITPFMT, ITPNO, A, B, K, IDEVCE
 NFIELD     1 ITPFMT   3 ITPNO   3 A        .0 B        1.0 IDEVCE   0
 CHANNELS SELECTED,    1   2   3   4
TYPE YES IF INPUTS ARE OK
>YES
$INFLDD ISTART,ISKIP,IINC,JSTART,JSKIP,JINC
INPUT  1 FIELD DATA
FIELD ISTART ISKIP IINC JSTART JSKIP JINC
  1     1700      1    80    130      1   190

TYPE YES IF INPUTS ARE OK
>YES
 THE OPTION DATDEF REQUIRED     10.4956 SECONDS OF CPU TIME.

 ----------------------------------------------------------------
```

SAMPLE 5. An example of data extraction from an observation tape written in Universal Computer Compatible format as indicated by ITPNO being set to 4.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>DATDEF

                        DATDEF  OPTION
                        ==============

$INDATD  NFIELD, ITPFMT, ITPNO, A, B, K, IDEVCE
 NFIELD     2 ITPFMT   4 ITPNO   3 A         .0 B        1.0 IDEVCE   0
 CHANNELS SELECTED,   1   2   3   4
TYPE YES IF INPUTS ARE OK
>YES
$INFLDD ISTART,ISKIP,IINC,JSTART,JSKIP,JINC
INPUT  1 FIELD DATA
INPUT  2 FIELD DATA
FIELD ISTART ISKIP IINC JSTART JSKIP JINC
  1       5      0     9     1      0     5
  2       5      0     9   295      0     5

TYPE YES IF INPUTS ARE OK
>YES

 NUMBER OF SUBFRAME STATUS BITS BEING USED IS        0
                         UNIVERSAL HEADER RECORD
                         -----------------------


START  NO OF
BYTE   BYTES   DESCRIPTION

1       32     COMPUTING SYSTEM ID                        *
33      20     TAPE LIBRARY ID                            *
53      8      SENSOR ID                                  *S192
```

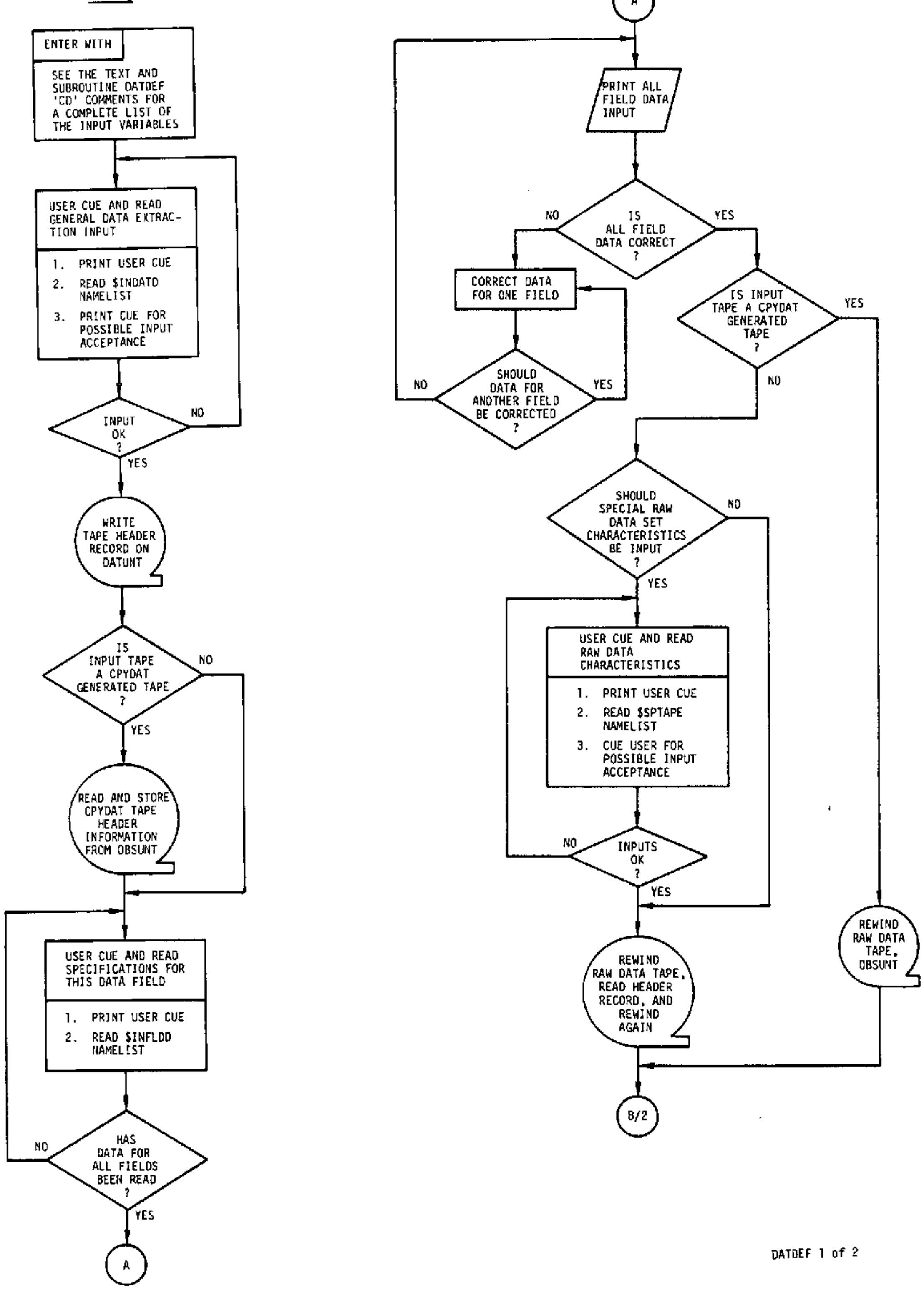
```
61       3      DATE OF TAPE GENERATION  DD MM YY *    4  10  73
64       1      TAPE SEQUENCE ID                  *       1
65       2      MISSION NUMBER                    *      96
67       2      SITE                              *   50407
69       1      LINE                              *       0
70       1      RUN                               *       0
71       2      ORBIT                             *       8
73       8      TIME OF FIRST SCAN  DD MM YY      * 245 211 242
                                    HH MM SS .MS  *   0   0   0   0
81       8      CHANNELS ACTIVE                   *

1111000000000000000000000000000000000000000000000000000000000000

89       1      PROCESSING FLAG                   *       1
90       1      NO.CHANNELS ON THIS TAPE          *       4
91       1      NO. OF BITS/PIXEL                 *       8
92       2      FIRST VIDEO PIXEL WITHIN SCAN     *       0
94       2      FIRST CAL. ELEMENT WITHIN SCAN    *       0
96       2      NO. PIXELS/SCAN/CHANNEL           *     300
98       2      NO. CALIB./SCAN/CHANNEL  (AREA 1) *       0
100      2      PHYSICAL RECORD SIZE IN BYTES     *    3060
102      1      NO. CHANNELS/PHYSICAL RECORD      *       0
103      1      PHYS. RECORDS/SCAN/CHANNEL        *       0
104      1      NO. RECORDS TO A DATA SET         *       1
105      2      ANCILLARY DATA LENGTH IN BYTES    *     316
107      1      DATA ORDER INDICATOR              *       0
108      2      START PIXEL NUMBER                *    2500
110      2      STOP PIXEL NUMBER                 *    2799
112      512    COEFFS,EXPS. A0-E0-A1-E1          *

         0           0           0           0
         0           0           0           0
         0           0           0           0
         0           0           0           0

624      64     COLOR CODE 0=NA,1=R,2=G,3=B       *
                                                          0
                                                          0
                                                          0
                                                          0

688      64     SCALE FACTOR                      *
                                                          0
                                                          0
                                                          0
                                                          0

752      1      OFFSET CONSTANT                   *       0
753      1      WORD SIZE OF GENERATING COMPUTER  *       8
1778     1      NO OF DATA SETS/PHYSICAL RECORD   *       1
1779     2      START OF SECOND CALIB. AREA       *       0
1781     2      NO. CAL. ELE. IN SECOND CAL. AREA *       0
1783     1      FIRST CAL. SOURCE INDICATOR       *       0
1783     1      SECOND CAL. SOURCE INDICATOR      *       0
```

```
1784     1       FILL ZEROS                                 *00000000
1785     2       NO. CHANNELS IN FIRST RECORD               *        4
1787     2       BYTES/SCAN/CHANNEL                         *      300
1789     2       PIXEL SKIP FACTOR                          *      257
1791     2       SCAN SKIP FACTOR                           *      264
2789     1       N THOUSAND SCAN LINES PER FRAME            *        0
2790     3       ALTITUDE IN METERS                         *        0
2793     2       GROUND SPEED IN METERS PER SECOND          *        0
2795     1       SCAN TYPE -LINEAR OR SMOOTHED              *        0
2796     1       ANGLE OF ARC IN DEGREES                    *        0
2797     1       CAMERA   0 = 70 MM,  1 = 5 INCH            *        0
2798     1       INPUT DEV. 0 = 9TRK,  1 = HDT              *        0
2799     1       TRUNCATION  0=LO, 1=HI, 2=NO               *        0
2800     8       CHANNELS REQUESTED                         *
00000000000000000000000000000000000000000000000000000000000000000000000000000000
2808     1       PROCESSING MODE  0=SER, 1=CONC.            *        0
2874     1       COLOR 0=NO COLOR,1=ASSIGN,2=FALSE          *        0
2875     1       IMAGE FORMAT                               *        0
2876     1       REPEAT OF PIXELS PER SCAN                  *        0
2877     1       REPEAT OF SCAN                             *        0
2878     4       PARTIAL SCAN -  FROM PIXEL NO.             *        0
                              -  TO  PIXEL  NO.             *        0
2882     2       SCAN RATE IN SCANS PER SECOND              *        0
2885     2       ANGLE OF DRIFT                             *     .000
2884     1       PIXEL  SIZE                                *        0


$NUMBER
KBIT     =              +8
LFPV     =              +1
NPV      =            +300
NC       =              +4
LADB     =            +316
NSPR     =              +1
NRPS     =              +1
NCFR     =              +4
NCPR     =              +0
NRPC     =              +0

$END
WRITE  240 WORDS ON UNIT  4
WRITE  240 WORDS ON UNIT  4
 THE OPTION DATDEF REQUIRED      3.2392 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

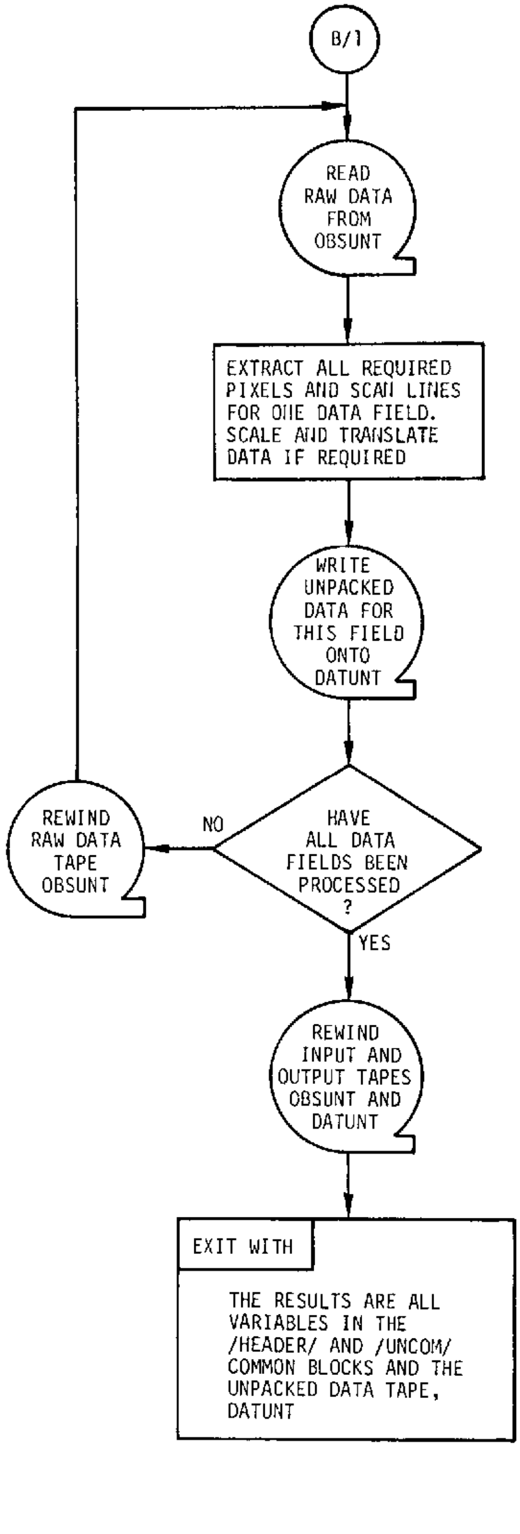
DATDEF ENGINEERING DESCRIPTION

DATDEF does not require an engineering description - see functional flow diagram.

DATDEF



DATDEF 1 of 2


B/1
READ RAW DATA FROM OBSUNT
EXTRACT ALL REQUIRED PIXELS AND SCAN LINES FOR ONE DATA FIELD. SCALE AND TRANSLATE DATA IF REQUIRED
WRITE UNPACKED DATA FOR THIS FIELD ONTO DATUNT
HAVE ALL DATA FIELDS BEEN PROCESSED ?
NO
REWIND RAW DATA TAPE OBSUNT
YES
REWIND INPUT AND OUTPUT TAPES OBSUNT AND DATUNT
EXIT WITH
THE RESULTS ARE ALL VARIABLES IN THE /HEADER/ AND /UNCOM/ COMMON BLOCKS AND THE UNPACKED DATA TAPE, DATUNT

DATDEF 2 of 2

Using the ADPCLU Option

Upon entering ADPCLU the user must define values for the parameters

C = threshold used in merging of clusters, units of the data

S = threshold used in grouping the data into strips, units of the data

RP = first threshold used in the priority search for assigning a strip to a cluster mean, units of the data

R1,R2 = second thresholds, 1st and 2nd data passes respectively, used in the priority search for assigning a strip to a cluster mean or to the unassigned category, units of the data

NVMMAX = maximum number of clusters to be allowed (≤ 20)

NPT = frequency for updating cluster priority list

NET = frequency for performing small cluster elimination tests

NMT = frequency for performing cluster merger tests

NMIN = population threshold for eliminating small clusters. The values of NET and NVMMAX should be considered in setting the value of NMIN.

IP = print control flag, if =0 no print during clustering process, if $\neq 0$ print merger and elimination messages.

The default values for these parameters are

C = 16.

S = 1.

RP = 8.

R1 = 18.

R2 = 24.

NVMMAX = 20

NPT = 100

NET = 500

NMT = 100

NMIN = 1

IP = 0

The user then must define the cluster mean and weights initialization procedure. The options are

ZERO - all values 0, this has the effect of forcing the first cluster mean to be 0 which causes the first data vector to start a cluster in slot number two.

OLD - use means and weights from last previous clustering (either ADPCLU or ITRCLU) results. For example, if one exits ADPCLU, calls IMAGES and then reenters ADPCLU, the previous means and weights remain available. One may continue to sequence through the clustering options and image display with the previous results available to restart via OLD.

NEW - allows user to input starting values or to change any of the current values. The parameters are

NVM = number of clusters

NVG = weight for each cluster

VM = cluster means, one-dimensional array of number of channels x NVM values representing a matrix of mean vectors input by columns.

Upon completion of the clustering a run summary id displayed. This output is a description of the clusters formed. It lists the cluster number, assigns a symbol to those points in the cluster, describes the size, gives the statistics (mean and sigma) of the points from the cluster center, and gives the L1 distances between the vector used as a center to form the cluster and the mean vector of the resulting cluster.

The user then must select one of the suboptions, MEANS, SIGMAS, ANGDIS, or QUIT. MEANS, SIGMAS, and ANGDIS are for output only and require no input parameters. QUIT returns control to ASTEP.

The MEANS suboption displays an m x n array where m is the number of data channels and n is the number of clusters. The columns are the mean vectors for the clusters formed during the 2nd pass.

The SIGMAS suboption displays an m x n array of the individual sigmas for each channel and cluster. The columns are the channel sigmas for the clusters formed. These are based upon the second pass assignments.

The output of the ANGDIS suboption is an n x n array. The diagonal of this array will be zero. Angles (in degrees) between a pair of mean vectors are given above the diagonal. The distances (given in channel units) between the vectors are given below the diagonal. An L1 distance measure is used.

The algorithm uses the first cluster (symbol A) in a special way. It always contains the unassigned cluster which consists of all points which could not be assigned to any other cluster. Depending upon the use of the OLD and NEW initialization procedures the first cluster may also represent a valid cluster, i.e., if OLD or NEW resulted in a non-zero mean vector for the first cluster. In this case the first cluster will contain both the unassigned points and the points resulting from the starting mean. In general, this situation should be avoided but it causes no real problems as i) the mean vector is not computed or updated for the unassigned points and ii) the thresholding in IMAGES will always delete the unassigned points.

ADPCLU OPTION

SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>ADPCLU

                     ADPCLU  OPTION
                     **************

$INADPC    C,S,RP,R1,R2,NVMMAX,NPT,NET,NMT,NMIN,IP
$INADPC
C         =       .18000000E+02
S         =       .20000000E+01
RP        =       .15000000E+02
R1        =       .30000000E+02
R2        =       .30000000E+02
NVMMAX    =                 +20
NPT       =                +100
NET       =                +500
NMT       =                +100
NMIN      =                  +5
IP        =                  +1

$END
TYPE YES IF INPUTS OK
>YES
CHOOSE VALUES FOR INITIALIZATION FROM
ZERO  OLD  NEW
>NEW
$ININIT   VM,NVG,NVM
NVM =  6 NVG =       1        1        1        1        1        1
                MEANS           4 BY  6
           1          2          3          4          5          6

  1     85.655     74.737     88.488     83.586     79.988     84.016
  2    126.034     82.826     94.491     85.908     98.166    112.111
  3    125.379     56.037     74.446     61.162     96.903    107.143
  4     90.138    110.251     80.348     81.395     74.785     85.921

TYPE YES IF INPUTS ARE CORRECT.
>YES
CLUSTER    5 WEIGHT     1 ELIMINATED, JPT =   500 NVM =    6
CLUSTER    5 WEIGHT     1 ELIMINATED, JPT =   500 NVM =    5
CLUSTER    5 WEIGHT     2 ELIMINATED, JPT =  1000 NVM =    8
CLUSTER    6 WEIGHT     3 ELIMINATED, JPT =  1000 NVM =    7
CLUSTER    6 WEIGHT     5 ELIMINATED, JPT =  1000 NVM =    6
CLUSTER    7 WEIGHT     1 ELIMINATED, JPT =  1500 NVM =    8
```

```
CLUSTER  SYMBOL  SIZE  R MEAN   R SIGMA    DIFF
   1        A       28   14.19      5.71    22.58
   2        B      449   10.18      5.38      .10
   3        C      426    8.02      4.45      .29
   4        D      464   10.72      5.46      .20
   5        E      153    9.24      4.33      .82
   6        F       79   13.26      4.56     1.29
   7        G       82    8.18      3.29      .39
CHOOSE OPTION FROM
  MEANS   SIGMAS  ANGDIS  QUIT
>MEANS
                  MEANS        4 BY  7
            1          2          3          4          5          6

  1    81.502     74.693     88.521     83.862     79.928     83.342
  2   119.759     82.811     94.519     85.976    100.222    110.772
  3   117.291     56.049     74.418     61.252     99.484    107.418
  4    86.101    109.757     80.427     81.138     77.471     84.848
            7

  1    79.817
  2    92.878
  3    90.037
  4    68.476

CHOOSE OPTION FROM
  MEANS   SIGMAS  ANGDIS  QUIT
>SIGMAS
                  SIGMAS       4 BY  7
            1          2          3          4          5          6

  1    20.238      1.984      2.598      3.365      1.763      2.976
  2    30.033      1.435      1.951      2.118      2.961      4.397
  3    29.463      1.907      2.756      2.644      3.667      5.624
  4    21.207      7.890      3.502      5.733      3.424      3.645
            7

  1     1.641
  2     2.431
  3     3.125
  4     3.190
```

```
CHOOSE OPTION FROM
  MEANS   SIGMAS  ANGDIS  QUIT
>ANGDIS
                    ANGDIS      7 BY  7
             1          2          3          4          5          6

  1        .000     20.186     11.016     14.333      3.336      2.447
  2     128.656       .000     13.308     10.415     18.457     18.362
  3      80.807     73.235       .000      3.805      8.306      8.728
  4      97.145     46.158     27.078       .000     11.767     12.075
  5      47.549     98.368     42.319     60.079       .000      1.189
  6      21.953    112.889     58.853     75.192     29.275       .000
  7      73.446     90.461     37.915     52.394     25.897     55.172
             7

  1       5.261
  2      18.896
  3       7.286
  4      11.008
  5       2.398
  6       3.377
  7        .000

CHOOSE OPTION FROM
  MEANS   SIGMAS  ANGDIS  QUIT
>QUIT
  THE OPTION ADPCLU REQUIRED       3.7904 SECONDS OF CPU TIME.
```

ADPCLU ENGINEERING DESCRIPTION

The adaptive clustering algorithm, ADPCLU, adaptively forms the cluster means with one pass through the data. A second pass through the data is used to generate the classification map. The adaptive mean computation employs local cluster or strip formulation, sequential search for strip assignment, periodic merging of similar clusters, and deletion of small clusters. The second pass through the data employs the strip formulation and sequential search through the clusters. Tests are made to determine when it is time to update the priority ranking of clusters, to merge clusters, and to delete clusters. The frequencies for performing these operations are input as system parameters. The tests for performing these operations are exercised only after the formation of a data strip has been completed. The details of these features are discussed below.

Strip Formulation.- If

$V_j(i)$ = the ith component of the jth vector to be assigned

S = strip refinement parameter

then, the local group or strip is defined by the vectors $V_{j+\ell}$, $\ell = 0, 1, \ldots, L$, where L is the last ℓ for which

$$|V_j(i) - V_{j+\ell}(i)| \leq S$$

is valid for all i. After generating the local subgroup, its mean and weight are computed.

Sequential Search.- If

$\left.\begin{matrix}RP\\R\end{matrix}\right\}$ Strip assignment threshold parameters

The sequential search computes the L1 distance between the mean of the local subgroup and each of the cluster means. The search terminates whenever this distance is less than RP. The order of searching the cluster means is in the order of their populations. Three outcomes are possible:

1. The subgroup is assigned to the first cluster for which the distance is less than RP.

2. The subgroup is assigned to the nearest cluster when the distance to the nearest cluster is greater than RP but less than R.

3. The subgroup is used to begin a new cluster; that is, the distance to the nearest cluster is greater than R.

The value of R=R1 for the first pass through the data and R=R2 for the second pass through the data.

After assignment of the strip - cases 1 and 2 - the mean and population count are updated.

An update to the ranking of the populations of the clusters occurs after NPT (system parameter) points have been clustered or after a merging or elimination of clusters has been performed. The counter NPC is then reset to zero and the clusters are not ranked again until NPC ≥ NPT or until after another cluster merging or elimination operation has been performed.

Cluster Merging.-

The cluster merging process operates by computing the L1 distance between the nearest pair of cluster means. If this distance is less than or equal to a threshold C, then the two means are averaged into one. The nearest distance between clusters is recomputed, and the merging process continues until all the clusters are separated by C or more. The merging operation is performed when the counter NMC of the number of clustered points since the last merger exceeds the threshold NMT (system parameter). After a merger, the counter NMC is reset to zero.

Deleting Clusters.-

The test for deleting clusters is made when the counter NEC exceeds the threshold NET (system parameter). NEC is the number of clustered points since the last deletion process. All clusters with less than NMIN points (system parameter) are deleted, and the counter NEC is reset to zero.

During the second pass for each pixel or strip of pixels, a distance for the image display is generated. This distance is the L1 distance from the mean of the strip to the cluster mean it is assigned to. All unassigned pixels are given a distance of $1. \times 10^{38}$. Later, in the images option, the distance values are compared to an input threshold and all pixels whose distance exceeds the threshold value are not displayed.

The ANGDIS option computes and displays the angles and L1 distances between all pairs of mean vectors resulting from clustering the data. Define

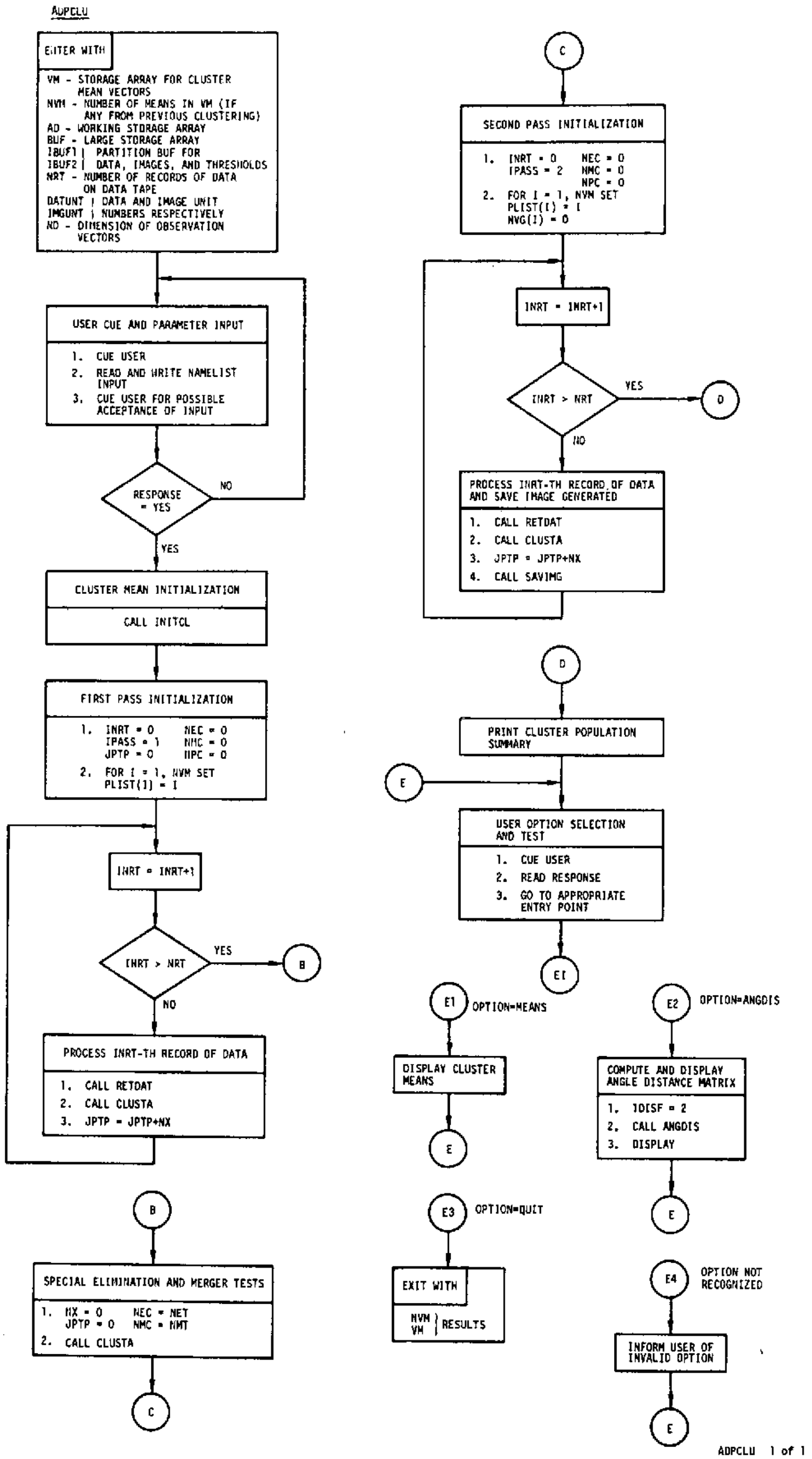
M_i = mean vector of i^{th} cluster

$m_i(k)$ = k-th component of the i^{th} mean vector

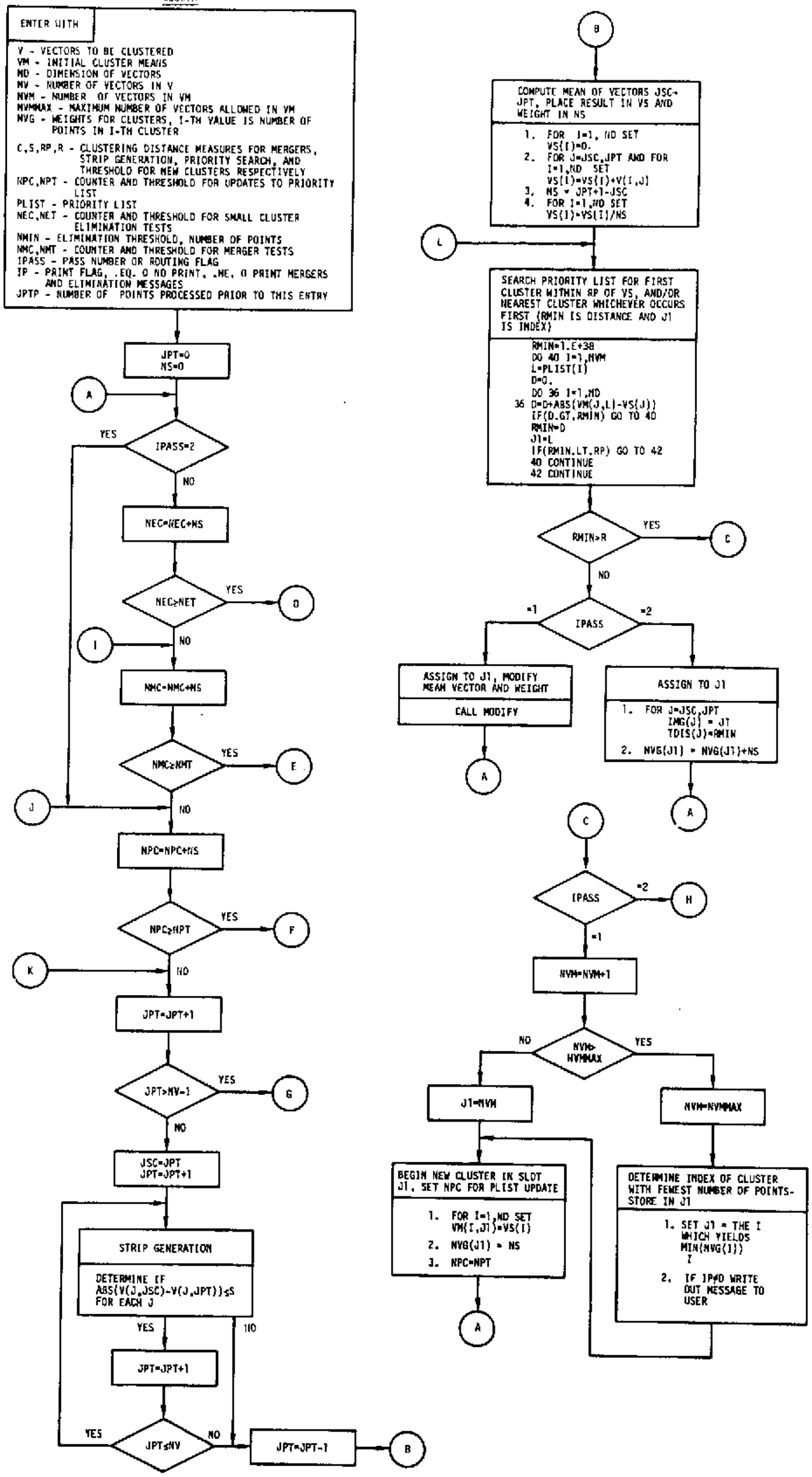
$$dij = \begin{cases} i > j & dij = \sum_k |m_i(k) - m_j(k)| \\ i = j & dij = 0 \\ i < j & dij = \frac{360}{2\pi} \cos^{-1} \frac{M_i \cdot M_j}{|M_i||M_j|} \end{cases}$$

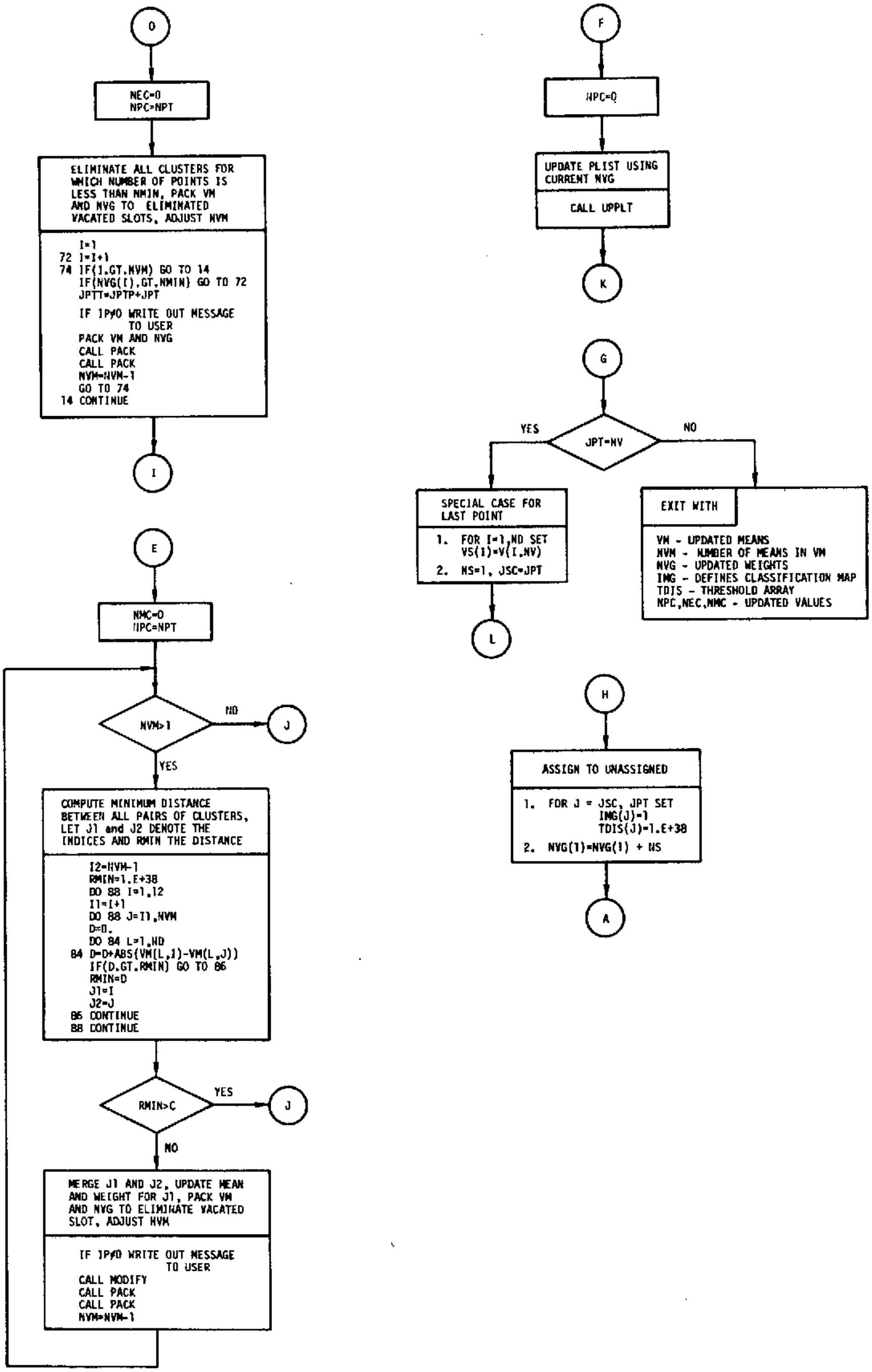
then the matrix D = [dij] is displayed.

The adaptive clustering algorithm is based upon the approach in Reference 2 and incorporates some of the ideas in Reference 3 to save computation time.


ADPCLU
ENTER WITH
VM - STORAGE ARRAY FOR CLUSTER MEAN VECTORS
NVM - NUMBER OF MEANS IN VM (IF ANY FROM PREVIOUS CLUSTERING)
AD - WORKING STORAGE ARRAY
BUF - LARGE STORAGE ARRAY
IBUF1 } PARTITION BUF FOR
IBUF2 } DATA, IMAGES, AND THRESHOLDS
NRT - NUMBER OF RECORDS OF DATA ON DATA TAPE
DATUNT } DATA AND IMAGE UNIT
IMGUNT } NUMBERS RESPECTIVELY
ND - DIMENSION OF OBSERVATION VECTORS
USER CUE AND PARAMETER INPUT
1. CUE USER
2. READ AND WRITE NAMELIST INPUT
3. CUE USER FOR POSSIBLE ACCEPTANCE OF INPUT
RESPONSE = YES
NO
YES
CLUSTER MEAN INITIALIZATION
CALL INITCL
FIRST PASS INITIALIZATION
1. INRT = 0 NEC = 0
IPASS = 1 NMC = 0
JPTP = 0 NPC = 0
2. FOR I = 1, NVM SET
PLIST(I) = I
INRT = INRT+1
INRT > NRT
YES
B
NO
PROCESS INRT-TH RECORD OF DATA
1. CALL RETDAT
2. CALL CLUSTA
3. JPTP = JPTP+NX
B
SPECIAL ELIMINATION AND MERGER TESTS
1. NX = 0 NEC = NET
JPTP = 0 NMC = NMT
2. CALL CLUSTA
C
C
SECOND PASS INITIALIZATION
1. INRT = 0 NEC = 0
IPASS = 2 NMC = 0
NPC = 0
2. FOR I = 1, NVM SET
PLIST(I) = I
NVG(I) = 0
INRT = INRT+1
INRT > NRT
YES
D
NO
PROCESS INRT-TH RECORD OF DATA AND SAVE IMAGE GENERATED
1. CALL RETDAT
2. CALL CLUSTA
3. JPTP = JPTP+NX
4. CALL SAVIMG
D
PRINT CLUSTER POPULATION SUMMARY
E
USER OPTION SELECTION AND TEST
1. CUE USER
2. READ RESPONSE
3. GO TO APPROPRIATE ENTRY POINT
EI
E1
OPTION=MEANS
DISPLAY CLUSTER MEANS
E
E2
OPTION=ANGDIS
COMPUTE AND DISPLAY ANGLE DISTANCE MATRIX
1. IDISF = 2
2. CALL ANGDIS
3. DISPLAY
E
E3
OPTION=QUIT
EXIT WITH
NVM } RESULTS
VM }
E4
OPTION NOT RECOGNIZED
INFORM USER OF INVALID OPTION
E
ADPCLU 1 of 1

CLUSTA


ENTER WITH
V - VECTORS TO BE CLUSTERED
VM - INITIAL CLUSTER MEANS
ND - DIMENSION OF VECTORS
NV - NUMBER OF VECTORS IN V
NVM - NUMBER OF VECTORS IN VM
NVMMAX - MAXIMUM NUMBER OF VECTORS ALLOWED IN VM
NVG - WEIGHTS FOR CLUSTERS, I-TH VALUE IS NUMBER OF POINTS IN I-TH CLUSTER
C,S,RP,R - CLUSTERING DISTANCE MEASURES FOR MERGERS, STRIP GENERATION, PRIORITY SEARCH, AND THRESHOLD FOR NEW CLUSTERS RESPECTIVELY
NPC,NPT - COUNTER AND THRESHOLD FOR UPDATES TO PRIORITY LIST
PLIST - PRIORITY LIST
NEC,NET - COUNTER AND THRESHOLD FOR SMALL CLUSTER ELIMINATION TESTS
NMIN - ELIMINATION THRESHOLD, NUMBER OF POINTS
NMC,NMT - COUNTER AND THRESHOLD FOR MERGER TESTS
IPASS - PASS NUMBER OR ROUTING FLAG
IP - PRINT FLAG, .EQ. 0 NO PRINT, .NE. 0 PRINT MERGERS AND ELIMINATION MESSAGES
JPTP - NUMBER OF POINTS PROCESSED PRIOR TO THIS ENTRY
JPT=0
NS=0
A
IPASS=2
YES
NO
NEC=NEC+NS
NEC≥NET
YES
D
I
NO
NMC=NMC+NS
NMC≥NMT
YES
E
J
NO
NPC=NPC+NS
NPC≥NPT
YES
F
K
NO
JPT=JPT+1
JPT>NV-1
YES
G
NO
JSC=JPT
JPT=JPT+1
STRIP GENERATION
DETERMINE IF ABS(V(J,JSC)-V(J,JPT))≤S FOR EACH J
YES
NO
JPT=JPT+1
JPT≤NV
YES
NO
JPT=JPT-1
B
B
COMPUTE MEAN OF VECTORS JSC-JPT, PLACE RESULT IN VS AND WEIGHT IN NS
1. FOR I=1, ND SET VS(I)=0.
2. FOR J=JSC,JPT AND FOR I=1,ND SET VS(I)=VS(I)+V(I,J)
3. NS = JPT+1-JSC
4. FOR I=1,ND SET VS(I)=VS(I)/NS
L
SEARCH PRIORITY LIST FOR FIRST CLUSTER WITHIN RP OF VS, AND/OR NEAREST CLUSTER WHICHEVER OCCURS FIRST (RMIN IS DISTANCE AND J1 IS INDEX)
RMIN=1.E+38
DO 40 I=1,NVM
L=PLIST(I)
D=0.
DO 36 J=1,ND
36 D=D+ABS(VM(J,L)-VS(J))
IF(D.GT.RMIN) GO TO 40
RMIN=D
J1=L
IF(RMIN.LT.RP) GO TO 42
40 CONTINUE
42 CONTINUE
RMIN>R
YES
C
NO
IPASS
=1
=2
ASSIGN TO J1, MODIFY MEAN VECTOR AND WEIGHT
CALL MODIFY
A
ASSIGN TO J1
1. FOR J=JSC,JPT IMG(J) = J1 TDIS(J)=RMIN
2. NVG(J1) = NVG(J1)+NS
A
C
IPASS
=2
H
=1
NVM=NVM+1
NVM>NVMMAX
NO
YES
J1=NVM
NVM=NVMMAX
BEGIN NEW CLUSTER IN SLOT J1, SET NPC FOR PLIST UPDATE
1. FOR I=1,ND SET VM(I,J1)=VS(I)
2. NVG(J1) = NS
3. NPC=NPT
A
DETERMINE INDEX OF CLUSTER WITH FEWEST NUMBER OF POINTS-STORE IN J1
1. SET J1 = THE I WHICH YIELDS MIN(NVG(I)) I
2. IF IP≠0 WRITE OUT MESSAGE TO USER


D
NEC=0
NPC=NPT
ELIMINATE ALL CLUSTERS FOR WHICH NUMBER OF POINTS IS LESS THAN NMIN, PACK VM AND NVG TO ELIMINATED VACATED SLOTS, ADJUST NVM
I=1
72 I=I+1
74 IF(I.GT.NVM) GO TO 14
IF(NVG(I).GT.NMIN) GO TO 72
JPTT=JPTP+JPT
IF IP≠0 WRITE OUT MESSAGE TO USER
PACK VM AND NVG
CALL PACK
CALL PACK
NVM=NVM-1
GO TO 74
14 CONTINUE
I
E
NMC=0
NPC=NPT
NVM>1
NO
J
YES
COMPUTE MINIMUM DISTANCE BETWEEN ALL PAIRS OF CLUSTERS, LET J1 and J2 DENOTE THE INDICES AND RMIN THE DISTANCE
I2=NVM-1
RMIN=1.E+38
DO 88 I=1,I2
I1=I+1
DO 88 J=I1,NVM
D=0.
DO 84 L=1,ND
84 D=D+ABS(VM(L,I)-VM(L,J))
IF(D.GT.RMIN) GO TO 86
RMIN=D
J1=I
J2=J
86 CONTINUE
88 CONTINUE
RMIN>C
YES
J
NO
MERGE J1 AND J2, UPDATE MEAN AND WEIGHT FOR J1, PACK VM AND NVG TO ELIMINATE VACATED SLOT, ADJUST NVM
IF IP≠0 WRITE OUT MESSAGE TO USER
CALL MODIFY
CALL PACK
CALL PACK
NVM=NVM-1
F
NPC=0
UPDATE PLIST USING CURRENT NVG
CALL UPPLT
K
G
JPT=NV
YES
NO
SPECIAL CASE FOR LAST POINT
1. FOR I=1,ND SET VS(I)=V(I,NV)
2. NS=1, JSC=JPT
L
EXIT WITH
VM - UPDATED MEANS
NVM - NUMBER OF MEANS IN VM
NVG - UPDATED WEIGHTS
IMG - DEFINES CLASSIFICATION MAP
TDIS - THRESHOLD ARRAY
NPC,NEC,NMC - UPDATED VALUES
H
ASSIGN TO UNASSIGNED
1. FOR J = JSC, JPT SET IMG(J)=1 TDIS(J)=1.E+38
2. NVG(1)=NVG(1) + NS
A

CLUSTA 2 OF 2

UPPLT

ENTER WITH

NVG - WEIGHTS, I-TH VALUE IS NUMBER OF POINTS IN I-TH CLUSTER
NVM - NUMBER OF ENTRIES IN NVG

↓

GENERATE PRIORITY LIST - PLIST-, J-TH VALUE IS INDEX OF J-TH LARGEST ENTRY IN NVG

```
      DO 10 I=1, 20
   10 TLIST(I)=0
      DO 30 L=1,NVM
      N=-1
      DO 20 I=1,NVM
      IF(TLIST(I).EQ.1) GO TO 20
      IF(NVG(I).LE.N) GO TO 20
      N=NVG(I)
      J=I
   20 CONTINUE
      TLIST(J)=1
   30 PLIST(L)=J
```

↓

EXIT WITH

PLIST

Using the COMPAR Option

In order to use the COMPAR option, the two signatures to be compared must be available in the currently assigned signature files.

The suboptions available are:

REFSIG - used to read the reference signature from a file and compute its eigenvectors. The signature name and file number are required inputs.

TSTSIG - used to read the test signature from a file and compute its eigenvectors. The signature name and file number are required inputs.

PROJEC - used to compute and print the projection data. The namelist input INPROJ is required and consists of the following variables:

- NRV = the dimension of the subspace to be formed from the reference eigenvectors.
- IRV = the index numbers of the reference eigenvectors to be used to form the subspace. The eigenvectors are ordered so that the first corresponds to the largest eigenvalue and the others follow in descending order.

QUIT - used to return control to the driver so that another ASTEP option may be selected.

COMPAR OPTION

SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>COMPAR

                  COMPAR  OPTION
                  ******.********

 SELECT FROM REFSIG, TSTSIG, PROJEC, QUIT.
>REFSIG
REFSIG HAS BEEN SELECTED.

INPUT NAME TO SEARCH ON
>SIGAB

CHOOSE FILE NUMBER FROM  1  2
>1

SIGAB  ND =  4 K =   1  6  9 12
NUM(1) =    467


 SELECT FROM REFSIG, TSTSIG, PROJEC, QUIT.
>TSTSIG
TSTSIG HAS BEEN SELECTED.

INPUT NAME TO SEARCH ON
>SIGA

CHOOSE FILE NUMBER FROM  1  2
>1

SIGA   ND =  4 K =   1  6  9 12
NUM(1) =    29
```

```
 SELECT FROM REFSIG, TSTSIG, PROJEC, QUIT.
>PROJEC
PROJEC HAS BEEN SELECTED.
$INPROJ NRV IRV
$INPROJ
NRV    =                +1,
IRV    =                +1,          +0,          +0,          +0.
                        +0,          +0,          +0,          +0.
                        +0,          +0,          +0,          +0.
                        +0,          +0,          +0,          +0.
                        +0,          +0,          +0,          +0.
                        +0,          +0,          +0,          +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES
REFERENCE SIGAB  AND TEST SIGA
                 COMPON      1 BY  5
          1          2          3          4          5

 1      .868      -.400       .207      -.208      1.000

                 SIZES       2 BY  5
          1          2          3          4          5

 1      .868       .400       .207       .208      1.000
 2      .496       .917       .978       .978       .027

                 ANGLES      1 BY  5
          1          2          3          4          5

 1     29.730     66.443     78.033     78.004     1.572



 SELECT FROM REFSIG, TSTSIG, PROJEC, QUIT.
>QUIT
QUIT   HAS BEEN SELECTED.
 THE OPTION COMPAR REQUIRED        .2446 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

If the statistical variations between fields of the same crop are to be explained by the mixture theory, then data from all of the fields must lie in the same subspace. In general, the eigenvectors corresponding to the significant eigenvalues for all fields of the same type should lie in the same subspace.

The eigenvectors from two covariance matrices can be compared by writing the second set as linear combinations of the first set. Precisely,

$$f_i = \sum_{j=1}^{m} b_{ij} e_j$$

where

f_i = the $i\underline{th}$ eigenvector from the second covariance matrix

e_j = the $j\underline{th}$ eigenvector from the first covariance matrix

b_{ij}= the component of f_i along e_j and is given by $f_i^T e_j$

The eigenvectors are assumed to be ordered according to the sizes of the corresponding eigenvalues, with the largest first and the smallest last.

If the first q eigenvectors from the first covariance matrix are chosen to form a subspace, then

$$f_i = \sum_{j=1}^{q} b_{ij} e_j + \sum_{j=q+1}^{m} b_{ij} e_j$$

where the first term is the part of f_i that lies inside the subspace and the second term is the part of f_i that lies outside the subspace. If exactly q of the f_i vectors lie totally inside the subspace, then they also span that subspace.

In actual practice, none of the f_i vectors will lie exactly in the subspace. A measure of how well each vector lies in the subspace can be obtained by computing the magnitudes of the components inside and outside the subspace. For each f_i the magnitude of the component inside is

$$(f_i)_{in} = \left[\sum_{j=1}^{q} b_{ij}^2 \right]^{1/2}$$

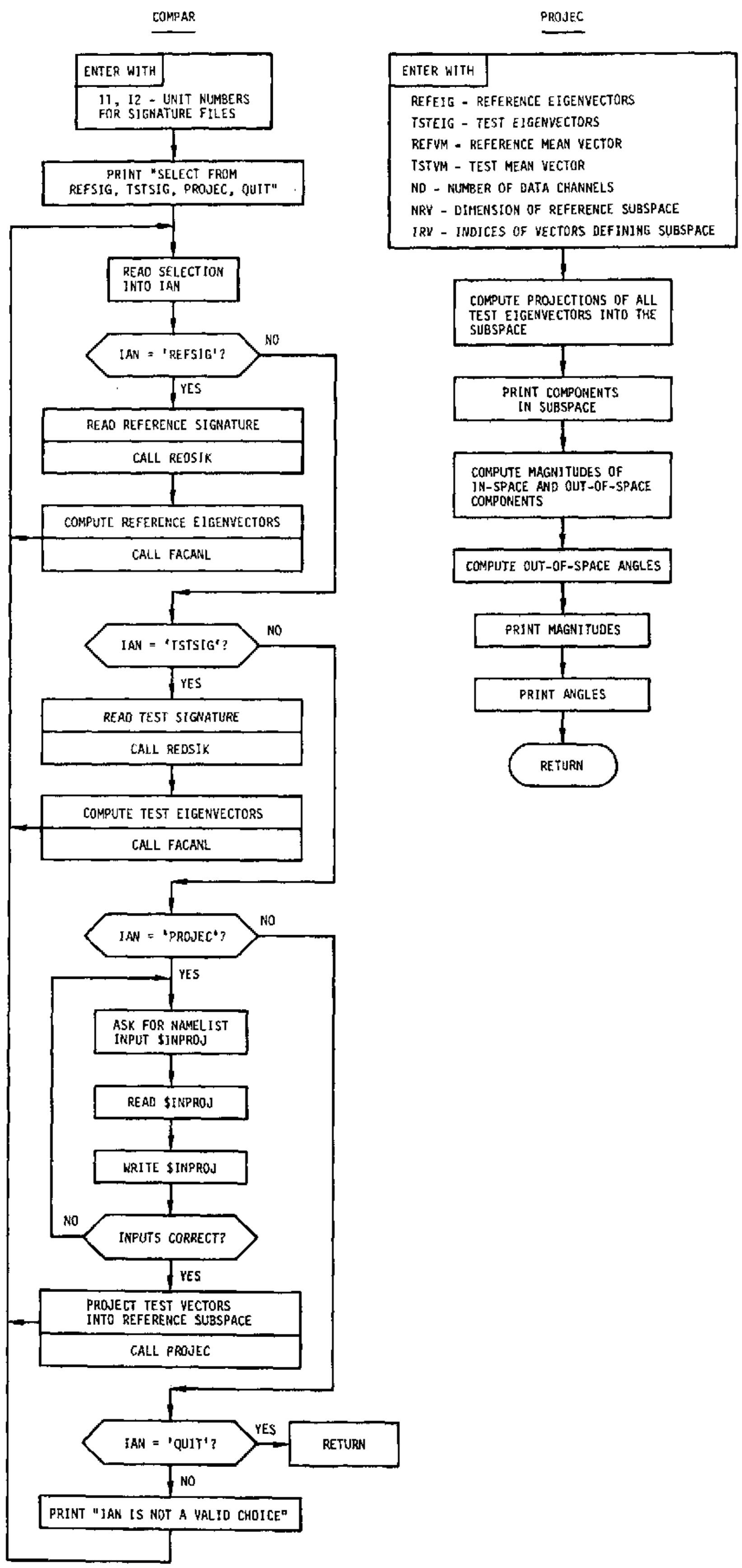
and the magnitude of the component outside is

$$(f_i)_{out} = \left[\sum_{j=q+1}^{m} b_{ij}^2 \right]^{1/2} = [1 - (f_i)_{in}^2]^{1/2}$$

In order to visualize the closeness of a vector to a subspace the out-of-space angle α_i can be computed from

$$\alpha_i = \tan^{-1} \frac{(f_i)_{out}}{(f_i)_{in}}$$

This angle will be zero for a vector totally in the subspace and 90 degrees for a vector totally out of the subspace.



COMPAR
ENTER WITH
I1, I2 - UNIT NUMBERS FOR SIGNATURE FILES
PRINT "SELECT FROM REFSIG, TSTSIG, PROJEC, QUIT"
READ SELECTION INTO IAN
IAN = 'REFSIG'?
NO
YES
READ REFERENCE SIGNATURE
CALL REDSIK
COMPUTE REFERENCE EIGENVECTORS
CALL FACANL
IAN = 'TSTSIG'?
NO
YES
READ TEST SIGNATURE
CALL REDSIK
COMPUTE TEST EIGENVECTORS
CALL FACANL
IAN = 'PROJEC'?
NO
YES
ASK FOR NAMELIST INPUT $INPROJ
READ $INPROJ
WRITE $INPROJ
NO
INPUTS CORRECT?
YES
PROJECT TEST VECTORS INTO REFERENCE SUBSPACE
CALL PROJEC
IAN = 'QUIT'?
YES
RETURN
NO
PRINT "IAN IS NOT A VALID CHOICE"
PROJEC
ENTER WITH
REFEIG - REFERENCE EIGENVECTORS
TSTEIG - TEST EIGENVECTORS
REFVM - REFERENCE MEAN VECTOR
TSTVM - TEST MEAN VECTOR
ND - NUMBER OF DATA CHANNELS
NRV - DIMENSION OF REFERENCE SUBSPACE
IRV - INDICES OF VECTORS DEFINING SUBSPACE
COMPUTE PROJECTIONS OF ALL TEST EIGENVECTORS INTO THE SUBSPACE
PRINT COMPONENTS IN SUBSPACE
COMPUTE MAGNITUDES OF IN-SPACE AND OUT-OF-SPACE COMPONENTS
COMPUTE OUT-OF-SPACE ANGLES
PRINT MAGNITUDES
PRINT ANGLES
RETURN

SHEET 1 OF 1

Using the CPYDAT Option

The CPYDAT option is used to copy a portion of the packed observation data tape (or file) onto another tape (or file). Subsequent execution of DATDEF using the new tape or file may take significantly less computer time because the tape (or file) is much shorter. Also, if access to tape drives is limited, CPYDAT may be used to copy a portion of the data tape onto a file which may be easier to access. The original data is stored on OBSUNT and the copied file is on OBS1. Since this option uses the same system of subprograms as does the DATDEF option, duplicate definitions and explanations will be avoided by referencing parts of the description of that option where appropriate.

After selecting CPYDAT, the user is asked to input ITPFMT, ITPNO, ISTART, and IINC under the namelist $INCPYD. ITPFMT and ITPNO are explained completely under DATDEF option. Briefly, ITPFMT indicates the pixel-channel format of the data tape while ITPNO specifies particulars of the specific tape (error records, record size, etc.). If ITPNO is five, the data tape was generated under a previous execution of CPYDAT. If ITPNO is less than zero, the particulars of the tape may be input under namelist name $SPTAPE; see DATDEF for more information.

ISTART is the first scan line to be copied and IINC+1 is the number of scan lines to be copied.

The user must use the UNITS option to change the OBSUNT to the OBS1 unit before the new CPYDAT tape may be used by DATDEF on the same run. The user always references the same scan line numbers and the original tape for a CPYDAT tape.

The CPYDAT option is intended to be used with packed data in the LARSI, LARSII, or ERTS format. The same function can be accomplished with more flexibility for data in the UCCT format by using the UVWRIT option.

Most of the messages which can be generated during the execution of the CPYDAT option can also be generated during an execution of the DATDEF option, but some messages are exclusive to the CPYDAT option. These messages and their meanings follow:

BUFFER NOT LARGE ENOUGH FOR CPYDAT, NO TAPE WRITTEN

The size of the internal scratch array BUF, NBUFSZ, is not large enough to accomodate one record from the input tape in each half of the array. There is probably an error in the value of NBUFSZ, an incorrect unit assignment has been made for OBSUNT, the wrong tape has been mounted, or a hardware I/O or positioning error has occurred. The program stops.

THE CPYDAT OPTION CAN NOT READ DATA IN UNIVERSAL FORMAT - RETURN TO MAIN PROGRAM

The capability to read a UCCT with the CPYDAT option has not yet been implemented. Execution is returned to the main program where the DATDEF option, which has this capability, may be executed.

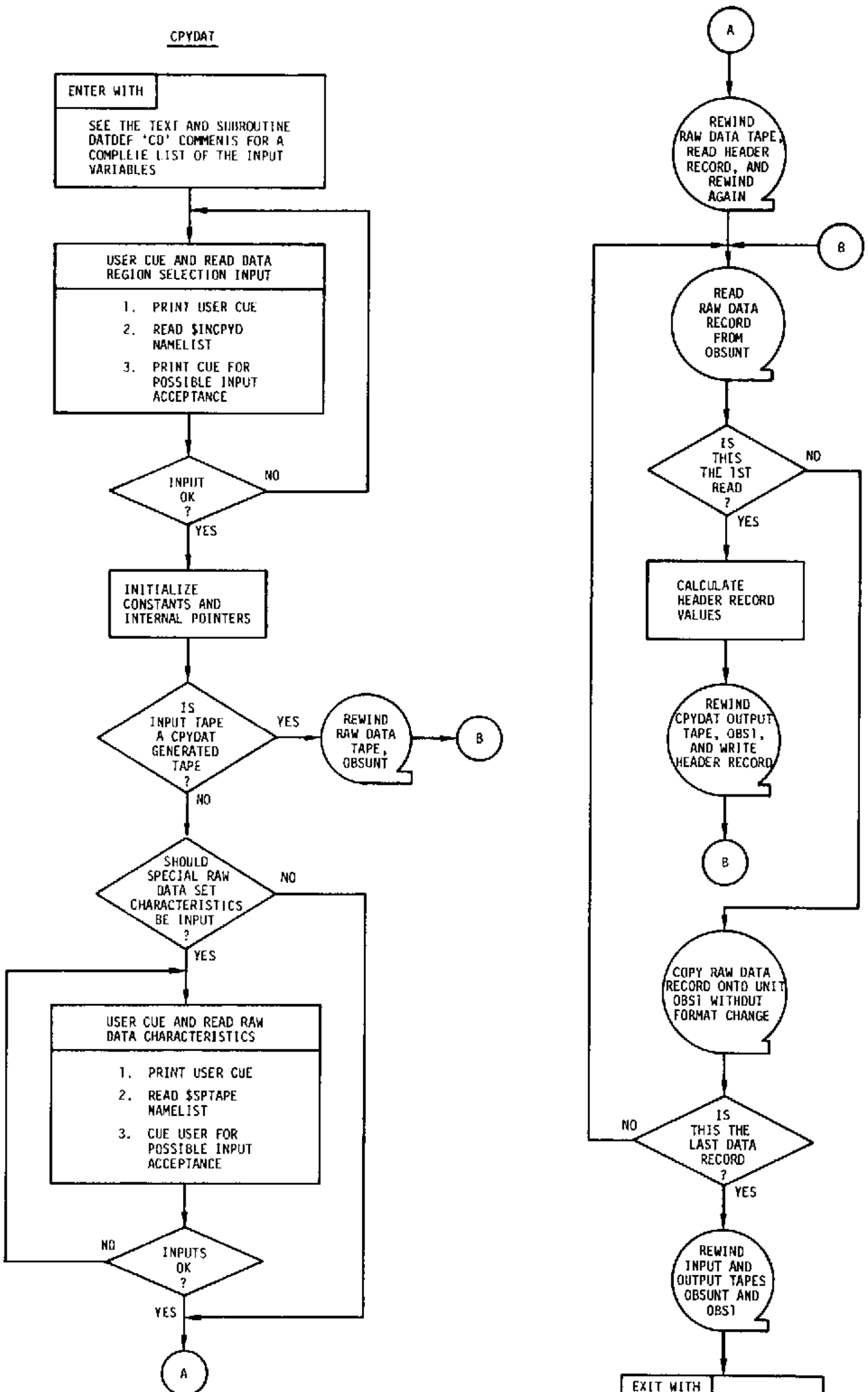
CPYDAT OPTION

SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>CPYDAT

                     CPYDAT  OPTION
                     ==============

$INCPYD A, B, ITPFMT, ITPNO, ISTART, IINC
     A      B ITPFMT  ITPNO ISTART   IINC
 255.0  -1.0       1      1    795     30
TYPE YES IF INPUTS ARE OK
>YES
 THE OPTION CPYDAT REQUIRED      7.2772 SECONDS OF CPU TIME.

 ----------------------------------------------------------
```

CPYDAT ENGINEERING DESCRIPTION

CPYDAT does not require an engineering description - see functional flow diagram.



CPYDAT
ENTER WITH
SEE THE TEXT AND SUBROUTINE DATDEF 'CD' COMMENTS FOR A COMPLETE LIST OF THE INPUT VARIABLES
USER CUE AND READ DATA REGION SELECTION INPUT
1. PRINT USER CUE
2. READ $INCPYD NAMELIST
3. PRINT CUE FOR POSSIBLE INPUT ACCEPTANCE
INPUT OK ?
NO
YES
INITIALIZE CONSTANTS AND INTERNAL POINTERS
IS INPUT TAPE A CPYDAT GENERATED TAPE ?
YES
REWIND RAW DATA TAPE, OBSUNT
B
NO
SHOULD SPECIAL RAW DATA SET CHARACTERISTICS BE INPUT ?
NO
YES
USER CUE AND READ RAW DATA CHARACTERISTICS
1. PRINT USER CUE
2. READ $SPTAPE NAMELIST
3. CUE USER FOR POSSIBLE INPUT ACCEPTANCE
NO
INPUTS OK ?
YES
A
A
REWIND RAW DATA TAPE, READ HEADER RECORD, AND REWIND AGAIN
B
READ RAW DATA RECORD FROM OBSUNT
IS THIS THE 1ST READ ?
NO
YES
CALCULATE HEADER RECORD VALUES
REWIND CPYDAT OUTPUT TAPE, OBS1, AND WRITE HEADER RECORD
B
COPY RAW DATA RECORD ONTO UNIT OBS1 WITHOUT FORMAT CHANGE
NO
IS THIS THE LAST DATA RECORD ?
YES
REWIND INPUT AND OUTPUT TAPES OBSUNT AND OBS1
EXIT WITH
ALL RESULTS ARE THE HEADER RECORD AND DATA RECORDS WRITTEN ONTO UNIT OBS1

CPYDAT 1 of 1

Format of the Packed Data Tape, or File Generated by the CPYDAT Option

When this tape or file is generated by the CPYDAT option using the system subroutine NTRAN each data record is copied exactly without any change in format; however a special header record is written which contains information required to process the data. The header record contains 20 words. The variables stored in each word are given below (please refer to previously given definitions of the variables involved):

Word Number		Variable Name or Value
1	=	ITPFMT
2	=	ISTART
3	=	IINC
4	=	NWRN
5	=	20
6	=	LEAD
7	=	MAXJ
8	=	MAXK
9	=	A
10	=	B

11-20 are presently unused.

This header record is the first record in the file. The original header record on the data tape is not copied into the file by CPYDAT.

Using the DIFIMG Option

The DIFIMG option allows the user to compute an image of the differences between the images on units IMG1 and IMG2 and store this image on unit IMGUNT. Upon selection of the DIFIMG option, the user is required to input the equivalence list of class symbols. In the equivalence list the user specifies which class symbols are to be defined as equal or equivalent in the two input images on units IMG1 and IMG2.

The first entry on the equivalence list is a class symbol on input image 1 (i.e., IMG1). Following entries are the class symbols on input image 2 (i.e., IMG2) which the user believes to be equivalent to the class symbol on input image 1.

A typical entry in the equivalence list is:

AB - specifies that the B's on input image 2 are equivalent to the A's on input image 1

or

AAB - specifies that the A's and B's on input image 2 are equivalent to the A's on input image 1.

The group of characters corresponding to image 2 in the list (B or AB in the above examples) may contain up to 5 characters. Up to 10 such entries in the equivalence list may be input. A blank card terminates the scan for entries in the list.

Images 1 and 2 are the first and second images that were generated with any of the classification options. Prior to using the DIFIMG option, it is recommended that the user read the description of the UNITS option. The UNITS option is intended to be used in conjunction with the DIFIMG option to conveniently manipulate the image unit numbers such that each image output may be placed on a unique unit by the user.

The output from the DIFIMG option is a table that contains three classes, A, B, and C and the size of each class. The classes are defined as follows:

Class A - those pixels in image 1 that were not in the equivalence list specified by the user

Class B - those pixels in image 1 different from image 2

Class C - those pixels in images 1 and 2 that are the same.

DIFIMG OPTION

SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>DIFIMG

                 DIFIMG  OPTION
                 **************

INPUT EQUIVALENCE LISTS
>AA
>BB
>CC
>DEH
>EE
>
CLASS  SYMBOL  SIZE
  1       A      63
  2       B     580
  3       C    1038
CLASS A ARE THOSE PIXELS OF IMAGE 1 THAT ARE NOT IN THE INPUT EQUIVALENCE LIST
CLASS B ARE THOSE PIXELS OF IMAGE 1 DIFFERENT FROM IMAGE 2
CLASS C ARE THOSE PIXELS THE SAME IN IMAGES 1 AND 2

 THE OPTION DIFIMG REQUIRED       .4742 SECONDS OF CPU TIME.
 --------------------------------------------------------
```

DIFIMG ENGINEERING DESCRIPTION

DIFIMG differences two input alphabetic image maps to produce a third alphabetic image map. It requires the use of an equivalence table to define the logical rules for differencing two alphabetic characters. The equivalence table is input by the user and has the form

$$\begin{array}{ll} \text{entry 1} & A_1 B_{11} B_{12} \cdots B_{1m} \\ \text{entry 2} & A_2 B_{21} B_{22} \cdots B_{2n} \\ \vdots & \\ \text{entry n} & A_n B_{n1} B_{n2} \cdots B_{ng} \end{array}$$

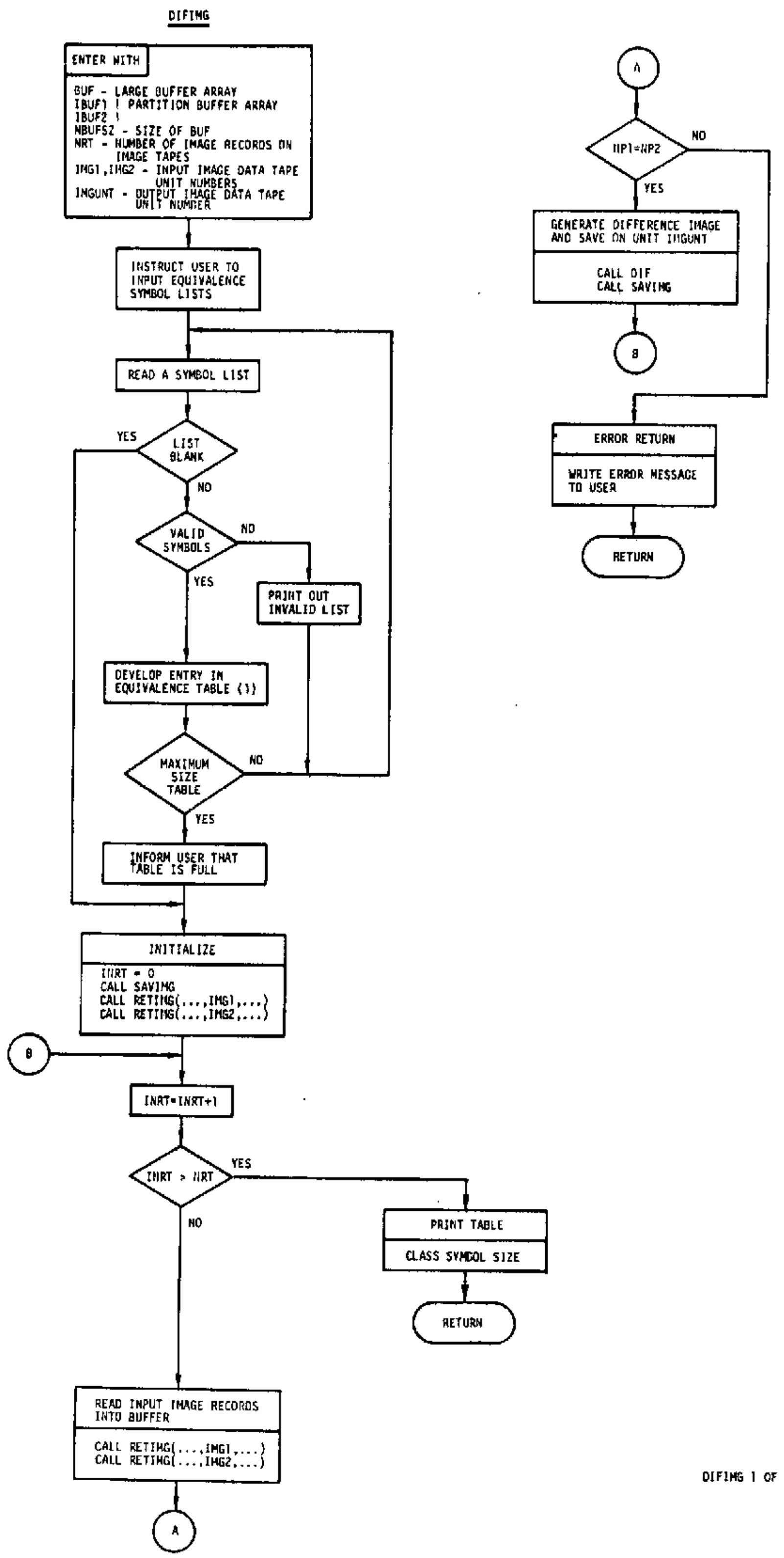
where each A_i and B_{j1} is an alphabetic character.

Let C_1, C_2 and C_0 be corresponding characters on the input images 1 and 2 and output images respectively. Then the differencing rule is

$$C_o \begin{cases} = C \text{ if for some } j \\ \quad C_1 = A_j \text{ and } C_2 = B_{ji} \text{ for some } i \\ = B \text{ if for some } j \; C_1 = A_j \text{ and } C_2 \neq B_{ji} \text{ for any } i \\ = A \text{ if } C_1 \neq A_j \text{ for any } j \end{cases}$$

The corresponding threshold data for the image display for the output image is set to zero.

The limits on the number of entries in the equivalence table, n, and the symbol list for the second input image, m, are $n \leq 10$ and $m \leq 5$.


DIFIMG
ENTER WITH
BUF - LARGE BUFFER ARRAY
IBUF1 } PARTITION BUFFER ARRAY
IBUF2 }
NBUFSZ - SIZE OF BUF
NRT - NUMBER OF IMAGE RECORDS ON IMAGE TAPES
IMG1,IMG2 - INPUT IMAGE DATA TAPE UNIT NUMBERS
INGUNT - OUTPUT IMAGE DATA TAPE UNIT NUMBER
INSTRUCT USER TO INPUT EQUIVALENCE SYMBOL LISTS
READ A SYMBOL LIST
LIST BLANK
YES
NO
VALID SYMBOLS
NO
YES
PRINT OUT INVALID LIST
DEVELOP ENTRY IN EQUIVALENCE TABLE (1)
MAXIMUM SIZE TABLE
NO
YES
INFORM USER THAT TABLE IS FULL
INITIALIZE
INRT = 0
CALL SAVING
CALL RETING(...,IMG1,...)
CALL RETING(...,IMG2,...)
B
INRT=INRT+1
INRT > NRT
YES
NO
PRINT TABLE
CLASS SYMBOL SIZE
RETURN
READ INPUT IMAGE RECORDS INTO BUFFER
CALL RETING(...,IMG1,...)
CALL RETING(...,IMG2,...)
A
A
NP1=NP2
NO
YES
GENERATE DIFFERENCE IMAGE AND SAVE ON UNIT INGUNT
CALL DIF
CALL SAVING
B
ERROR RETURN
WRITE ERROR MESSAGE TO USER
RETURN
DIFIMG 1 OF 1

ENTER WITH

IMG1, IMG2 - TWO DIFFERENT ARRAYS OF INTEGERS INDICATING THE GROUPS THAT VECTORS ARE ASSIGNED

NV - NUMBER OF POINTS IN IMG1 AND/OR IMG2

EQT - EQUIVALENCE LIST DEFINING DIFFERENCE. ALL CHARACTERS ARE REPRESENTED BY INTEGERS. THE K-TH ENTRY IN EQT IS AS FOLLOWS - - -

EQT(K,2)=THE CHARACTER ON IMG1 TO BE TESTED FOR MATCHING ONE OR MORE POSSIBLE CHARACTERS ON IMG2

EQT(K,J)=THE POSSIBLE CHARACTERS ON IMG2,J=3,...., 3+EQT(K,1)

EQT(K,1)=NUMBER OF POSSIBLE CHARACTERS ON IMG2

IEQT - NUMBER OF ENTRIES IN EQT

↓

DIFFERENCE IMAGES USING EQUIVALENCE TABLE FOR EACH PIXEL I

FOR I=1,NV

1. TEST EACH A ENTRY IN EQUIVALENCE TABLE AGAINST IMAGE 1, IF MATCH SET FLAG=1 AND DO STEP 2, OTHERWISE DO STEP 3.
2. TEST EACH B ENTRY CORRESPONDING TO THE MATCHED A ENTRY AGAINST IMAGE 2, IF MATCH SET FLAG=2, MATCH OR NO MATCH DO STEP 3.
3. SET POINT ON IMAGE 2 BY

$$\text{FLAG} = \begin{cases} 0 - A \\ 1 - B \\ 2 - C \end{cases}$$

↓

EXIT WITH

IMG2 - IMAGE ARRAY RESULTING FROM DIFFERENCING PROCESS

NVG(I) - NUMBER OF PIXELS IN EACH CLASS

NVM - NUMBER OF CLASSES (=3)

DIF 1 OF 1

Using the DUMP Option

DUMP option is a convenient way for a user to visually inspect contents of a tape file by translating it into readable form and printing it. The information from the input tape is broken into 8-bit bytes and converted to integers from 0 to 255 for printing.

Upon entering DUMP option the user inputs the following parameters using the $INDUMP namelist:

- NREC - number of physical records to be printed (default value is 5)
- IFIRST - index number of first record to be printed (default value is 1)
- NWORDS - maximum number of words to be translated for one record (default value is 5000)
- UNITNO - unit number for input tape (default value is 7)

The first record is always translated and printed, independent of the value of IFIRST parameter, since it is usually a header record. Data requested is printed as shown in the DUMP Option Sample Input and Corresponding Output section.

It should be noted that DUMP cannot be used to dump a Fastrand data file unless all of the records on the file are of the same length and NWORDS is set to that length. The reason is that there are no gaps between records on the file to tell NTRAN where to stop reading.

DUMP OPTION
SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>DUMP

                  DUMP    OPTION
                  **************

*** WARNING ***
THIS OPTION MAY PRODUCE A LARGE VOLUME OF OUTPUT.
INTERACTIVE USER SHOULD ADJUST NUMBER  OF RECORDS (NREC),
AND NUMBER OF WORDS TO BE OUTPUT PER RECORD (NWORDS), ACCORDINGLY.

BINDUMP    NREC,IFIRST,NWORDS,UNITNO
  NREC IFIRST NWORDS UNITNO
    3      1    300     19
TYPE YES IF INPUTS OK
>YES


 LNTRAN =   300 FOR RECORD     1

 RECORD      1 WITH  300 WORDS  (   1350 BYTES )
```

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	132	144	0	0	0	0	0	0	1	4
90	8	0	1	0	0	0	91	0	0	11	64	0	0	1	0	95	0	0	10	0	90	0	0	0	0	0	0	0	0	0
120	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
150	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
180	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
210	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
240	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
270	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
330	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
360	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
390	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
420	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
450	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
480	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
510	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
540	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
570	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
600	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
630	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
660	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
690	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
720	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
750	0	0	36	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
780	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
810	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
840	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
870	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
900	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
930	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
960	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
990	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1020	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1140	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1170	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1200	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1230	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1260	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1290	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

LNTRAN = 300 FOR RECORD 2

RECORD 2 WITH 300 WORDS (1350 BYTES)

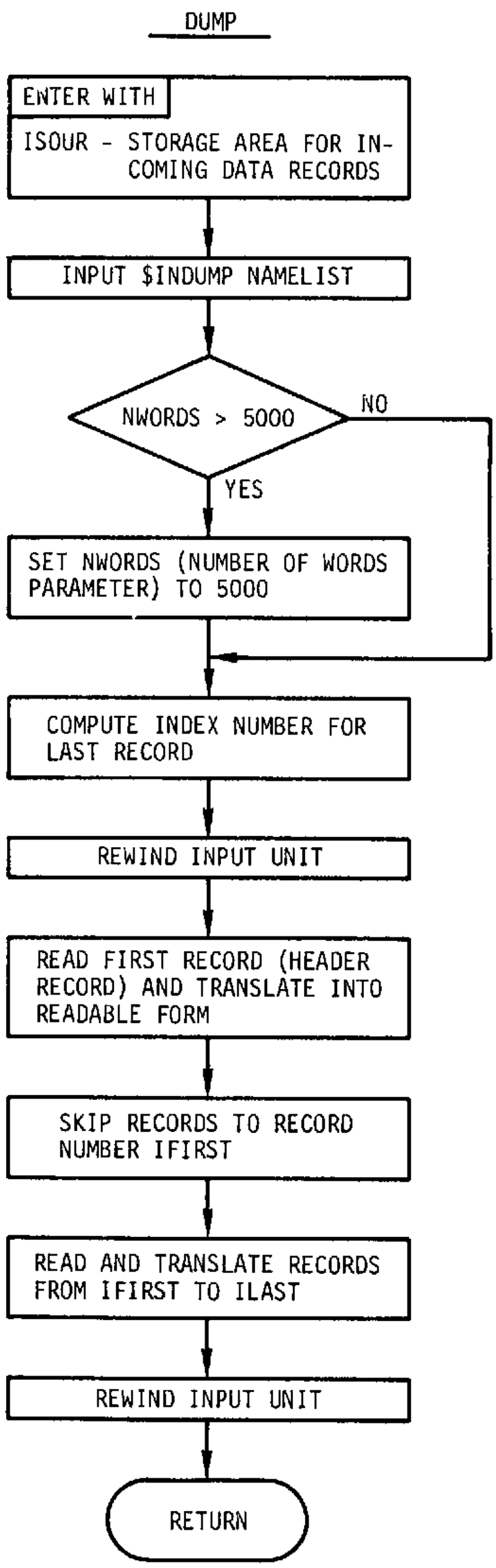
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	81	81	82	88	83	84	82	83	86	85	86	83	85	84	82	82	82	80	93	80	76	76	77
120	75	78	75	74	75	76	76	74	74	74	75	73	74	74	72	72	75	74	76	82	80	81	81	81	82	83	87	84	84	84
150	84	84	82	86	84	85	96	84	81	81	82	81	83	82	81	82	84	82	83	88	82	83	81	82	82	83	86	82	83	57
180	59	57	57	60	57	58	62	62	60	64	61	61	61	59	59	62	62	74	56	57	57	54	56	55	54	55	58	58	55	58
210	59	58	58	58	55	55	56	57	57	57	81	74	77	72	71	75	74	77	76	79	78	80	79	85	85	79	84	86	77	107
240	110	106	108	113	117	112	122	116	109	111	106	109	106	108	110	115	123	122	123	121	122	0	0	0	0	0	0	0	0	0
270	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
330	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	82	83	83	83
360	85	85	85	85	87	82	85	84	82	85	84	83	83	82	89	75	75	74	74	75	73	76	74	74	76	73	74	74	73	74
390	74	73	71	71	73	72	71	82	85	84	86	88	87	86	86	87	85	88	88	88	88	86	88	87	93	87	82	83	80	82
420	82	82	82	84	82	86	83	82	86	83	82	81	82	81	82	81	83	82	57	60	58	59	60	58	61	61	62	60	61	63
450	60	60	60	64	60	71	63	54	56	55	54	56	55	55	57	55	57	57	56	64	57	56	58	55	55	55	54	53	54	84
480	76	79	80	78	80	73	75	77	81	82	83	90	88	83	83	82	89	90	106	111	118	116	116	117	122	122	121	113	122	119
510	97	108	116	116	111	122	122	118	120	119	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
540	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
570	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0
600	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	81	85	85	87	88	88	86	88	86	89	89	88	87	87	86
630	83	87	82	89	80	76	78	77	75	76	76	74	75	75	76	76	74	75	74	74	73	74	72	73	73	75	82	84	84	89
660	85	84	84	84	85	86	88	88	87	88	89	84	89	91	84	82	83	83	82	83	83	82	83	84	84	84	84	84	82	82
690	84	83	83	84	82	82	82	59	60	60	62	61	59	60	61	60	64	63	62	63	59	63	60	60	71	58	57	57	54	56
720	56	56	57	54	56	54	58	57	58	58	58	56	57	56	55	56	57	54	85	84	89	84	79	81	79	76	75	72	79	86
750	87	88	85	86	81	86	108	117	114	113	124	114	116	119	122	122	120	123	119	118	111	115	123	118	111	117	110	120	118	0
780	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
810	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
840	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
870	0	0	0	0	79	81	83	84	84	87	85	86	87	87	85	86	87	84	86	82	86	82	88	77	76	76	75	76	73	74
900	74	75	73	73	74	74	75	74	74	72	74	76	77	74	73	82	82	86	86	88	87	87	87	87	85	87	89	90	87	84
930	86	86	88	87	82	83	83	83	83	83	85	84	83	84	85	86	84	85	84	84	84	83	88	87	84	82	59	57	63	59
960	62	60	60	62	65	63	62	62	62	62	61	62	60	67	60	58	58	57	56	55	56	55	54	56	57	55	56	55	56	56
990	56	54	56	61	54	56	56	82	82	80	85	83	80	83	79	74	70	78	84	88	87	78	76	75	74	91	109	109	106	109
1020	119	115	122	123	120	119	119	125	124	123	122	125	122	119	120	111	121	117	0	0	0	0	0	0	0	0	0	0	0	0
1050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0
1110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	82	82	85	86	85	84	85
1140	88	90	87	90	82	85	85	84	83	84	87	82	78	77	76	77	74	74	76	75	75	74	76	73	75	73	73	76	73	74
1170	75	75	75	73	83	82	87	86	86	87	87	88	87	87	84	85	84	85	85	85	84	93	84	84	84	83	82	82	83	83
1200	82	84	81	84	84	83	83	82	83	83	81	84	83	81	83	60	58	61	61	62	61	62	64	65	65	61	61	59	61	59
1230	63	61	69	58	58	57	55	57	56	56	55	55	55	55	54	56	56	54	56	58	55	56	56	55	55	60	83	79	83	83
1260	80	77	76	74	69	68	75	75	77	79	75	72	81	76	107	109	111	107	104	108	110	121	116	122	118	119	116	122	114	112
1290	107	114	114	119	112	114	110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

LNTRAN = 300 FOR RECORD 3

RECORD 3 WITH 300 WORDS (1350 BYTES)

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	83	85	87	86	93	90	90	86	88	85	85	88	84	87	86	82	86	85	87	78	76	76	77
120	75	76	74	77	73	75	73	74	74	75	75	73	75	73	74	76	71	75	82	84	87	87	91	91	87	89	89	90	89	88
150	89	89	86	85	89	86	88	80	81	81	81	81	82	83	83	82	84	84	83	84	84	84	82	85	83	83	82	82	83	60
180	60	65	67	69	68	61	65	63	63	62	64	64	64	59	62	62	63	66	53	56	55	57	56	57	54	57	54	55	54	56
210	56	55	54	54	54	55	54	58	55	57	78	81	78	76	74	73	75	82	83	85	86	82	90	82	84	82	87	90	75	101
240	108	105	110	111	119	120	119	118	118	111	118	117	110	112	116	107	109	113	103	106	95	0	0	0	0	0	0	0	0	0
270	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
330	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	79	81	81	83
360	88	86	90	83	83	84	83	83	83	79	82	81	80	81	85	82	74	75	76	74	74	74	74	75	74	73	74	73	73	74
390	76	73	75	74	73	73	72	79	81	83	86	91	87	89	88	87	88	87	85	89	86	86	85	87	87	93	81	82	81	80
420	80	81	85	81	81	84	83	83	82	85	82	84	85	84	82	84	83	84	57	59	58	62	67	66	64	62	60	60	58	58
450	61	60	58	60	60	59	72	59	56	55	54	56	56	54	57	54	55	55	54	56	56	58	56	62	56	54	55	55	53	77
480	77	80	78	77	72	74	78	85	86	90	84	90	88	85	86	84	91	82	97	108	109	112	110	122	119	118	109	121	108	119
510	113	109	106	96	105	98	106	105	105	113	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
540	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
570	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0
600	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	81	84	83	85	86	88	87	87	83	86	87	82	85	83	83
630	84	80	82	88	81	75	76	75	74	75	74	74	73	75	73	76	74	73	75	72	75	75	73	72	74	72	83	81	85	86
660	88	88	85	88	88	89	87	85	87	86	85	85	89	85	97	85	83	83	83	82	82	83	82	84	80	83	82	81	82	83
690	82	84	81	82	81	83	85	59	59	58	58	63	64	60	58	65	61	59	59	60	60	60	58	64	58	78	58	55	57	54
720	55	53	56	53	55	54	53	55	54	55	53	55	53	55	56	53	54	56	80	80	78	83	75	74	79	83	86	87	90	87
750	86	90	83	90	89	92	82	107	116	117	121	116	122	114	122	115	114	115	107	104	107	102	107	117	109	113	107	110	111	0
780	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
810	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
840	0	0	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
870	0	0	0	0	85	85	84	88	86	89	86	85	85	85	86	84	86	85	85	85	84	82	90	80	75	76	75	76	72	74
900	75	78	76	72	75	74	75	75	74	74	73	73	72	73	79	85	85	84	85	89	87	86	86	88	87	87	86	86	88	86
930	87	90	87	94	84	83	82	83	82	84	82	89	87	82	82	82	82	82	83	82	83	81	81	84	81	92	59	59	59	59
960	60	62	60	60	61	60	61	60	59	62	61	60	67	62	73	58	55	57	56	56	56	57	64	59	54	56	54	54	55	54
990	53	56	55	53	60	54	70	87	81	86	81	80	74	77	83	85	86	80	88	86	87	84	85	94	93	83	107	113	114	120
1020	117	118	116	106	107	109	106	101	102	98	98	99	99	98	97	106	106	98	0	0	0	0	0	0	0	0	0	0	0	0
1050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0
1110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	84	84	85	86	83	86	83
1140	85	86	84	85	84	83	85	87	83	84	84	90	82	77	78	75	78	76	74	75	77	73	76	74	75	75	73	73	74	73
1170	73	77	75	79	84	85	83	85	85	85	85	87	88	88	86	85	84	87	87	87	89	85	94	85	81	83	83	84	83	83
1200	85	82	83	84	84	82	80	80	82	80	81	85	83	82	86	59	58	58	56	58	59	58	59	59	59	60	60	59	62	59
1230	62	63	60	76	60	57	58	56	57	56	55	57	57	55	59	56	55	55	51	54	53	57	56	59	60	67	82	83	75	84
1260	83	76	77	80	81	87	83	81	84	83	85	85	90	91	81	107	115	119	116	119	115	112	111	119	106	107	106	103	95	94
1290	84	90	94	94	93	90	90	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

THE OPTION DUMP REQUIRED 2.1622 SECONDS OF CPU TIME.


DUMP
ENTER WITH
ISOUR - STORAGE AREA FOR INCOMING DATA RECORDS
INPUT $INDUMP NAMELIST
NWORDS > 5000
NO
YES
SET NWORDS (NUMBER OF WORDS PARAMETER) TO 5000
COMPUTE INDEX NUMBER FOR LAST RECORD
REWIND INPUT UNIT
READ FIRST RECORD (HEADER RECORD) AND TRANSLATE INTO READABLE FORM
SKIP RECORDS TO RECORD NUMBER IFIRST
READ AND TRANSLATE RECORDS FROM IFIRST TO ILAST
REWIND INPUT UNIT
RETURN

DUMP 1 of 1

Using the EDTSIG Option

The EDTSIG option is used to manipulate signatures and signature files. A signature consists of the mean vector, covariance matrix, and number of pixels for a set of data. A signature file consists of a set of signatures along with their associated six-character names and the number of data channels and the channel numbers for each one. A signature file is stored on an external medium such as tape or drum storage.

The suboptions within EDTSIG are:

BEGFIL - used to begin a signature file. The file may be on unit 1 or unit 2, depending on the user's choice.

WRTSIG - used to read a signature into the core storage area in EDTSIG from cards. The values are input through a namelist called INWRTS which consists of:

- ND - number of data channels
- K - list of channel numbers
- NUM - number of pixels
- VM - mean vector
- COV - covariance matrix (input by columns)

SAVSIG - used to save a signature by copying it from the core storage area into a signature file. The unit number and the signature name are specified by the user. The signature file must be prepared by BEGFIL before the first signature is saved. Up to one hundred signatures can be saved in one file.

LISFIL - used to obtain information about the contents of a signature file on the unit selected by the user. The signature name, number of channels, channel numbers, and number of pixels are printed for each signature in the file.

REDSIG - used to read a signature from a file into the core storage area. The unit number and the name of the signature are specified by the user.

PRTSIG - used to print the signature currently residing in core storage.

EIGSIG - used to compute and print the eigenvalues, eigenvectors, and factor analysis data for the signature in core storage.

ADDSIG - used to combine the information from two signatures into a third signature which corresponds to the union of the two sets of data. The first signature must be in core storage and the second must be read from a signature file. The unit number and name for the second signature are specified by the user. The result replaces the first signature in core storage.

QUIT - used to return the program control to the driver so that another ASTEP option can be selected.

Signatures may also be saved in the FACTOR option of ASTEP, but the file must have been initialized with BEGFIL in EDTSIG.

EDTSIG OPTION

SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>EDTSIG

                    EDTSIG  OPTION
                    **************


CHOOSE EDTSIG OPTION FROM
  BEGFIL  SAVSIG  REDSIG  WRTSIG
  LISFIL  ADDSIG  EIGSIG  PRTSIG
  QUIT
>LISFIL
LISFIL OPTION HAS BEEN SELECTED.

CHOOSE FILE NUMBER FROM  1  2
>1
SIGALL  NO =  4 K =   1  6  9 12
NUM =   1681    0     0     0     0     0     0     0     0     0     0     0
SIGA    NO =  4 K =   1  6  9 12
NUM =     29    0     0     0     0     0     0     0     0     0     0     0
SIGB    NO =  4 K =   1  6  9 12
NUM =    438    0     0     0     0     0     0     0     0     0     0     0
SIGAB   NO =  4 K =   1  6  9 12
NUM =    467    0     0     0     0     0     0     0     0     0     0     0
LISTING OF FILE  1 COMPLETED

CHOOSE EDTSIG OPTION FROM
  BEGFIL  SAVSIG  REDSIG  WRTSIG
  LISFIL  ADDSIG  EIGSIG  PRTSIG
  QUIT
>REDSIG
REDSIG OPTION HAS BEEN SELECTED.

INPUT NAME TO SEARCH ON
>SIGA

CHOOSE FILE NUMBER FROM  1  2
>1

SIGA    NO =  4 K =   1  6  9 12
NUM(1) =    29

CHOOSE EDTSIG OPTION FROM
  BEGFIL  SAVSIG  REDSIG  WRTSIG
  LISFIL  ADDSIG  EIGSIG  PRTSIG
  QUIT
>PRTSIG
PRTSIG OPTION HAS BEEN SELECTED.
                MEAN          1 BY  4
        1          2          3          4

 1    85.655    126.034    125.379    90.138
```

```
                        COVMAT      4 BY  4
              1          2          3          4

    1     15.591     22.512     18.064      8.192
    2     22.512     69.106     60.736     20.317
    3     18.064     60.736     74.958     15.553
    4      8.192     20.317     15.553     13.623

  CHOOSE EDTSIG OPTION FROM
    BEGFIL   SAVSIG   REDSIG   WRTSIG
    LISFIL   ADDSIG   EIGSIG   PRTSIG
    QUIT
  >EIGSIG
  EIGSIG OPTION HAS BEEN SELECTED.
                        ARRAY       4 BY  4
              1          2          3          4

    1    144.810       .836       .836     17.697
    2     16.325       .094       .930     77.437
    3      6.466       .037       .967     87.561
    4      5.676       .033      1.000     77.993

                        EIGENV      4 BY  4
              1          2          3          4

    1       .224       .473       .826       .208
    2       .667       .370      -.242      -.600
    3       .682      -.652       .109       .313
    4       .198       .463      -.497       .707

  CHOOSE EDTSIG OPTION FROM
    BEGFIL   SAVSIG   REDSIG   WRTSIG
    LISFIL   ADDSIG   EIGSIG   PRTSIG
    QUIT
  >ADDSIG
  ADDSIG OPTION HAS BEEN SELECTED.

  INPUT NAME TO SEARCH ON
  >SIGB

  CHOOSE FILE NUMBER FROM  1  2
  >1

  SIGB   ND =   4 K =    1   6   9  12
  NUM(1) =    438

  CHOOSE EDTSIG OPTION FROM
    BEGFIL   SAVSIG   REDSIG   WRTSIG
    LISFIL   ADDSIG   EIGSIG   PRTSIG
    QUIT
  >PRTSIG
  PRTSIG OPTION HAS BEEN SELECTED.
```

```
                    MEAN          1 BY   4
            1          2          3          4

 1     75.415     85.510     60.343     109.002

                    COVMAT        4 BY   4
            1          2          3          4

 1     11.990     29.490     46.613     -15.044
 2     29.490    115.062    179.542     -48.557
 3     46.613    179.542    288.565     -84.153
 4    -15.044    -48.557    -84.153      75.165

CHOOSE EDTSIG OPTION FROM
  BEGFIL  SAVSIG  REDSIG  WRTSIG
  LISFIL  ADDSIG  EIGSIG  PRTSIG
  QUIT
>QUIT
QUIT   OPTION HAS BEEN SELECTED.
 THE OPTION EDTSIG REQUIRED         .4334 SECONDS OF CPU TIME.

 -------------------------------------------------------------
```

EDTSIG ENGINEERING DESCRIPTION

EDTSIG is the subdriver for the spectral signature file maintenance options. Signature files may be read or written using tape unit 1 or tape unit 2. Each entry in the file has the form

Record 1	NAME	- six-character Hollerith name
	ND	- dimension of mean vector and covariance matrix
	K	- the channel numbers used to compute the signature data
	NUM	- the number of pixels
Record 2	VM	- mean vector
	COV	- covariance matrix

The end of the signature file is marked with a record of the form of 1 above with the name NOMORE.

ADDSIG Suboption

ADDSIG combines two signatures into a new signature. A detailed description of the procedure is included in this section.

BEGFIL Suboption

BEGFIL is used to begin a new signature file. It writes the special NOMORE record on the unit selected.

EIGSIG Suboption

EIGSIG computes and prints the factor analysis data for the signature that is currently in core storage. The method used is the same as in the ASTEP option FACTOR.

LISFIL Suboption

LISFIL lists all of the heading data, (i.e. NAME, dimension of signature data, channels used, and number of pixels from the signature file selected.

PRTSIG Suboption

PRTSIG prints the signature information currently in the EDTSIG core storage area.

REDSIG Suboption

REDSIG is used to retrieve a signature file. It searches the unit selected for the user specified NAME. It retrieves one signature and returns control to EDTSIG.

SAVSIG Suboption

SAVSIG is used to save signatures on a signature file. It adds the signature data under the name specified to the file unit specified.

WRTSIG Suboption

WRTSIG allows signatures to be input by the user to be saved in the signature files. Namelist is used to read in the signature data. SAVSIG is then used to add the data to the file.

QUIT Suboption

QUIT returns control to the ASTEP driver.

ADDSIG Engineering Description

The ADDSIG subroutine computes the number of samples, mean vector, and covariance matrix for the union of two sets of data vectors. This is done by combining the statistics from the two data sets without actually processing the data vectors again.

The total number of sample vectors, n, is simply

$$n = n_1 + n_2$$

where

n_1 = number of samples in data set 1

n_2 = number of samples in data set 2

The mean vector, M, is given by

$$M = \frac{n_1}{n} M_1 + \frac{n_2}{n} M_2$$

where

M_1 = mean of data set 1

M_2 = mean of data set 2

The covariance matrix, C, is

$$C = \frac{n_1 - 1}{n-1} C_1 + \frac{n_1}{n-1} (M_1 - M)(M_1 - M)^T + \frac{n_2 - 1}{n-1} C_2 + \frac{n_2}{n-1} (M_2 - M)(M_2 - M)^T$$

where

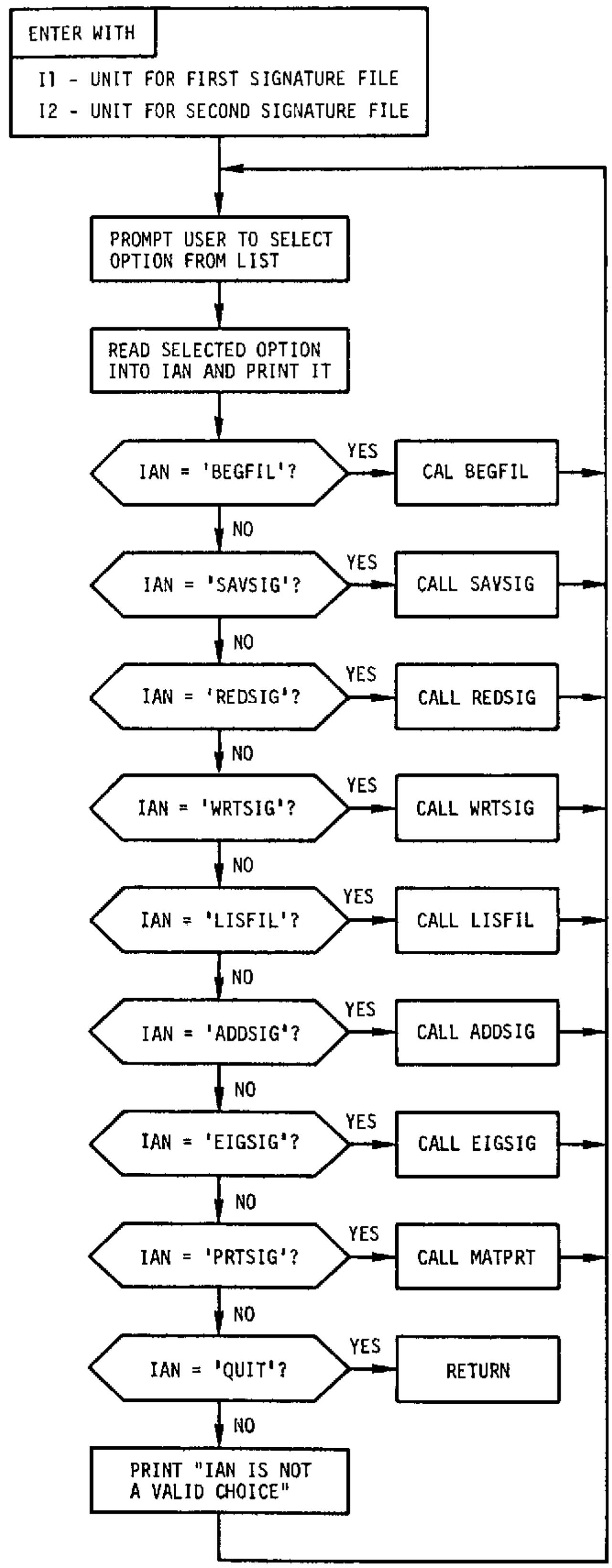
C_1 = covariance matrix of data set 1

C_2 = covariance matrix of data set 2

EIGSIG Engineering Description

The EIGSIG subroutine computes and prints the eigenvalues, eigenvectors, and factor analysis data for a given mean vector and covariance matrix. The calculations are identical to the factor analysis done in the FACTOR option. The equations are presented in the engineering description of the FACTOR option.

EDTSIG


ENTER WITH
I1 - UNIT FOR FIRST SIGNATURE FILE
I2 - UNIT FOR SECOND SIGNATURE FILE
PROMPT USER TO SELECT OPTION FROM LIST
READ SELECTED OPTION INTO IAN AND PRINT IT
IAN = 'BEGFIL'?
YES
CAL BEGFIL
NO
IAN = 'SAVSIG'?
YES
CALL SAVSIG
NO
IAN = 'REDSIG'?
YES
CALL REDSIG
NO
IAN = 'WRTSIG'?
YES
CALL WRTSIG
NO
IAN = 'LISFIL'?
YES
CALL LISFIL
NO
IAN = 'ADDSIG'?
YES
CALL ADDSIG
NO
IAN = 'EIGSIG'?
YES
CALL EIGSIG
NO
IAN = 'PRTSIG'?
YES
CALL MATPRT
NO
IAN = 'QUIT'?
YES
RETURN
NO
PRINT "IAN IS NOT A VALID CHOICE"

Using the FACTOR Option

The FACTOR option computes the number of pixels, the mean vector, the covariance matrix, the eigenvalues, the eigenvectors, and the angles between each eigenvector and the mean vector of the block or cluster of data being considered.

If the pixels have been classified, the classes for which the statistics are to be computed may be selected by listing the corresponding characters that would be seen in an IMAGE print. The list of characters BDF, for example, would select only the pixels which were classified as either B, D, or F.

If all pixels are wanted, or if they have not been classified, the character + must be used.

After the desired classes have been specified, the user is required to specify whether or not he wishes to print the statistics for the classes. A response of 'YES' will cause the statistics for the classes to be printed. A response of 'NO' will suppress the print.

Next the user must select the initial values for the statistics from the options:

ZERO - sets initial statistics to correspond to having no data.

FILE - reads initial values for statistics from a signature file. The unit number for the file and the name of the signature must be input.

If the user elects to print the statistics for the class, the output is:

MEAN - the mean vector

SIGMAS - the standard deviations as computed from the diagonal terms of the covariance matrix

COVMAT - the normalized covariance matrix with correlation coefficients above the diagonal

ARRAY - the first column is the eigenvalues ordered with the largest first; the second column is the ratio of each eigenvalue to the sum of the eigenvalues (tells how big each is in a relative sense); the third column contains sums of values in the second column up to that particular row (when the numbers approach 1 this means that most of the effects have been accounted for); the fourth column contains the angle (in degrees) between each eigenvector and the mean vector.

EIGENV - a matrix whose columns are the eigenvectors of the covariance matrix

Next the user is asked if he wishes to save the "signature" of the class for future reference. The mean vector and covariance matrix from FACTOR constitute the signature of the class. A response of 'YES' will save the signature. Prior to saving a signature, it is necessary to identify the file name and number and to have performed the BEGFIL suboption of EDTSIG option before FACTOR. If this has been done, the signature may be saved under the previously identified file name and number.

In order to return control to the ASTEP driver, it is necessary to input a blank for the classification character.

FACTOR OPTION
SAMPLE INPUT AND CORRESPONDING OUTPUT:

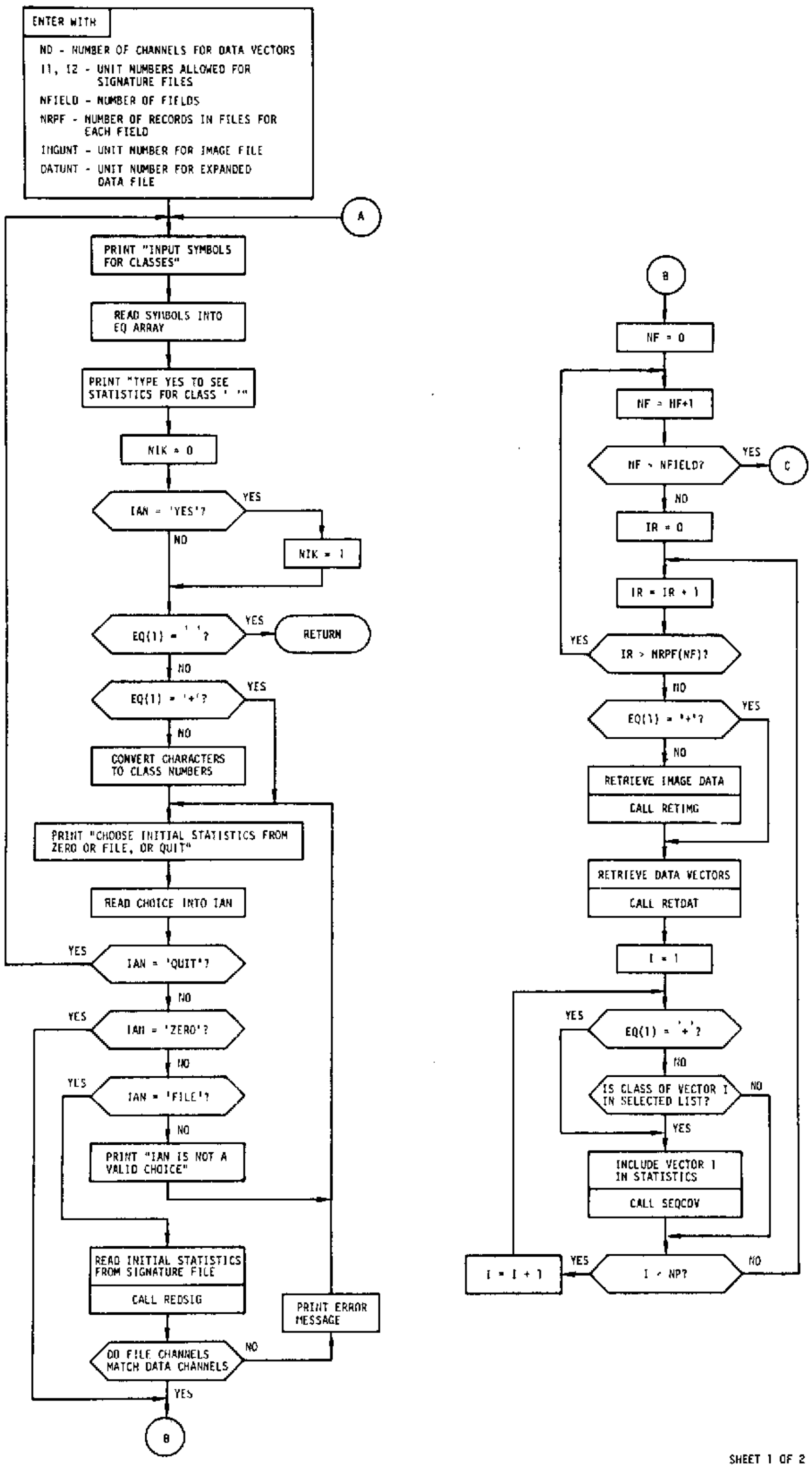
```
ENTER ASTEP OPTION OR TYPE A BLANK
>FACTOR


                         FACTOR OPTION
                         ===============


INPUT SYMBOLS FOR CLASSES.
>+
TYPE YES TO PRINT STATISTICS FOR CLASS +
>YES

FACTOR ANALYSIS FOR +
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE. OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
                      MEAN          1 BY  4
              1          2            3            4

  1      82.052     90.765       71.291       87.997

                      SIGMAS        1 BY  4
              1          2            3            4

  1       5.823      9.020       17.322       14.646

                      COVMAT        4 BY  4
              1          2            3            4

  1      33.906       .438         .290        -.672
  2      23.013     81.353         .925        -.412
  3      29.227    144.567      300.036        -.488
  4     -57.312    -54.411     -123.701      214.493

                      ARRAY         4 BY  4
              1          2            3            4

  1     458.090       .727         .727       70.496
  2     145.131       .230         .958       49.916
  3      23.191       .037         .995       47.164
  4       3.377       .005        1.000       96.193
```

```
                        EIGENV        4 BY  4
          1            2            3            4

 1       .141        -.246         .805        -.520
 2       .375         .256         .492        -.743
 3       .764         .459        -.242        -.383
 4      -.505         .815         .225        -.175

 TYPE YES TO SAVE SIGNATURE
 >YES

 CHOOSE FILE NUMBER FROM  1  2
 >1

 INPUT NAME TO SAVE DATA UNDER
 >SIGALL
 SIGNATURE SIGALL HAS BEEN SAVED ON UNIT 1
 SIGALL ND =  4 K =   1  6  9 12
 NUM(1) =  1681
 INPUT SYMBOLS FOR CLASSES.
 >A
 TYPE YES TO PRINT STATISTICS FOR CLASS A
 >YES


 FACTOR ANALYSIS FOR A
 CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
 >ZERO
 ZERO HAS BEEN CHOSEN.
                        MEAN          1 BY  4
          1            2            3            4

 1     85.655      126.034      125.379       90.138

                        SIGMAS        1 BY  4
          1            2            3            4

 1      3.949        8.313        8.658        3.691

                        COVMAT        4 BY  4
          1            2            3            4

 1     15.591         .686         .528         .562
 2     22.512       69.106         .844         .662
 3     18.064       60.736       74.958         .487
 4      8.192       20.317       15.553       13.623
```

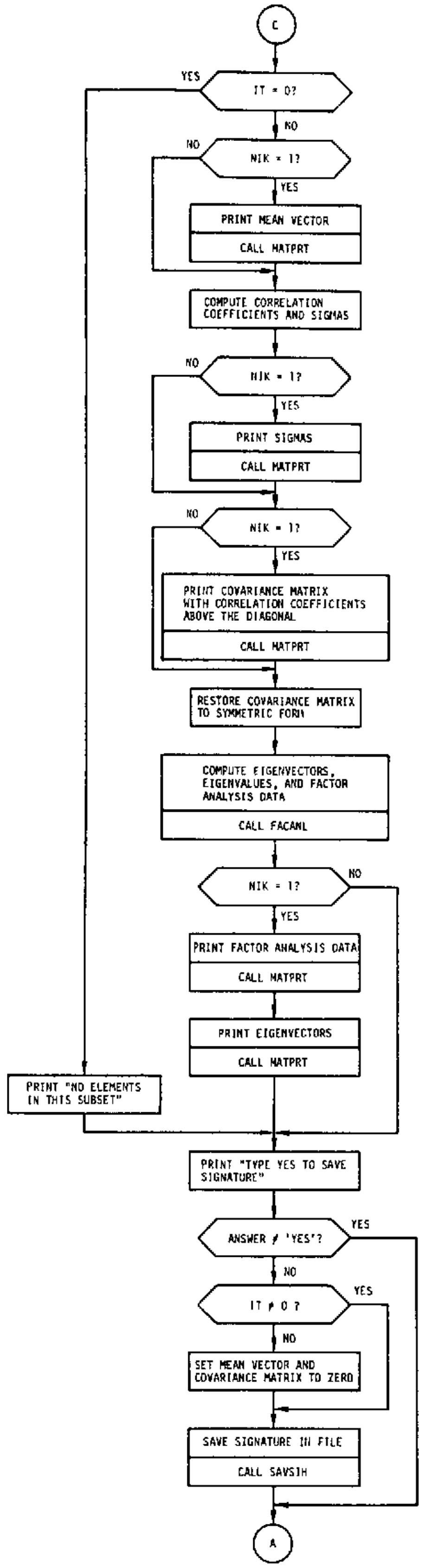
```
                  ARRAY      4 BY  4
        1            2           3          4

 1   144.810       .836        .836      17.697
 2    16.325       .094        .930      77.437
 3     6.466       .037        .967      87.561
 4     5.676       .033       1.000      77.993

                  EIGENV     4 BY  4
        1            2           3          4

 1      .224       .473        .826        .208
 2      .667       .370       -.242       -.600
 3      .682      -.652        .109        .313
 4      .198       .463       -.497        .707

TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1

INPUT NAME TO SAVE DATA UNDER
>SIGA
SIGNATURE SIGA    HAS BEEN SAVED ON UNIT 1
SIGA    ND =   4 K =    1  6  9 12
NUM(1) =      29
INPUT SYMBOLS FOR CLASSES.
>B
TYPE YES TO PRINT STATISTICS FOR CLASS B
>NO
FACTOR ANALYSIS FOR B
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1

INPUT NAME TO SAVE DATA UNDER
>SIGB
SIGNATURE SIGB    HAS BEEN SAVED ON UNIT 1
SIGB    ND =   4 K =    1  6  9 12
NUM(1) =     438
INPUT SYMBOLS FOR CLASSES.
>B
TYPE YES TO PRINT STATISTICS FOR CLASS B
>NO
FACTOR ANALYSIS FOR B
```

```
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>FILE
FILE HAS BEEN CHOSEN.

INPUT NAME TO SEARCH ON
>SIGA

CHOOSE FILE NUMBER FROM  1  2
>1

SIGA   ND =  4 K =   1  6  9 12
NUM(1) =     29
TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1

INPUT NAME TO SAVE DATA UNDER
>SIGAB
SIGNATURE SIGAB  HAS BEEN SAVED ON UNIT 1
SIGAB  ND =  4 K =   1  6  9 12
NUM(1) =    467
INPUT SYMBOLS FOR CLASSES.
>AB
TYPE YES TO PRINT STATISTICS FOR CLASS AB
>NO
FACTOR ANALYSIS FOR AB
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
TYPE YES TO SAVE SIGNATURE
>NO
INPUT SYMBOLS FOR CLASSES.
>
 THE OPTION FACTOR REQUIRED      3.4762 SECONDS OF CPU TIME.

 ----------------------------------------------------------------
```

FAKTOR



SHEET 1 OF 2


C
IT = 0?
YES
NO
NIK = 1?
NO
YES
PRINT MEAN VECTOR
CALL MATPRT
COMPUTE CORRELATION COEFFICIENTS AND SIGMAS
NIK = 1?
NO
YES
PRINT SIGMAS
CALL MATPRT
NIK = 1?
NO
YES
PRINT COVARIANCE MATRIX WITH CORRELATION COEFFICIENTS ABOVE THE DIAGONAL
CALL MATPRT
RESTORE COVARIANCE MATRIX TO SYMMETRIC FORM
COMPUTE EIGENVECTORS, EIGENVALUES, AND FACTOR ANALYSIS DATA
CALL FACANL
NIK = 1?
NO
YES
PRINT FACTOR ANALYSIS DATA
CALL MATPRT
PRINT EIGENVECTORS
CALL MATPRT
PRINT "NO ELEMENTS IN THIS SUBSET"
PRINT "TYPE YES TO SAVE SIGNATURE"
ANSWER ≠ 'YES'?
YES
NO
IT ≠ 0 ?
YES
NO
SET MEAN VECTOR AND COVARIANCE MATRIX TO ZERO
SAVE SIGNATURE IN FILE
CALL SAVSIH
A

SHEET 2 OF 2

Using the FEATSL Option

The purpose of this algorithm is to determine a linear transformation which can be used to reduce the dimension of the data to be processed from n to k, where k is less than n. Within FEATSL there are three suboptions. Suboption SUBSP determines a k by n matrix B ($n \leq 24$; $nk \leq 144$) which maximizes the B-average divergence (References 4 and 5). Suboption REPLCE determines a subset of k out of n channels ($k \leq n \leq 24$) which maximize the average divergence in accordance with the Without Replacement Procedure (Reference 6). Suboption CANON determines the best k linear combinations of n channels where $k < n \leq 24$ (Reference 7). The user can determine an adequate value of k by displaying the "Class Separation to be Gained Map". Upon entering FEATSL, the user must retrieve all signatures (class covariances and means) to be processed. This is accomplished by automatically having FEATSL call option REDSIH, so that the user retrieves all signatures in accordance with option REDSIH.

Once the signatures have been retrieved, the user must select the desired suboption to be executed. If suboption REPLCE is selected, the user must only input by namelist ($INFEAT) the following parameter:

KDIM - the number of channels to be selected.

Then suboption REPLCE is executed and the best KDIM channels are determined in accordance with the "Without Replacement Procedure".

If suboption SUBSP is selected, the user must first input by namelist ($INFEAT) the following parameter:

KDIM - the dimension of the feature space (i.e. k where $k \leq 6$)

Since it is desired to find a k by n transformation matrix B of rank k = KDIM which maximizes the B-average divergence, the user must now initialize this matrix. This is accomplished by the user typing in one of the following four initialization suboptions.

Suboption	Function
CHANAL	User selects KDIM distinct channels to initialize B.

Suboption	Function
VECTOR	Each element of the B-matrix is initialized by the user.
DEFALT	B-matrix is automatically initialized using internal program logic.
RESTRT	Allows user to exceed maximum number of iterations (400) for a given problem, or to alter the convergence tolerance. (Normally, the user need not execute this suboption).

All additional parameters to be defined below are also input by namelist ($INFEAT). If initialization suboption CHANAL is desired, the user must input

ICHAN - the desired distinct KDIM channels

For example, assuming twelve dimensional signatures, if KDIM=3, and

ICHAN = 1,10,12

then the program sets the element in the first row and first column of B to 1.0; the element in the second row and tenth column of B to 1.0; the element in the third row and twelfth column of B to 1.0 and all other elements of B are set to 0.0. Thus, in this example, B effectively selects the first, tenth, and twelfth components of each observation vector.

If initialization suboption VECTOR is selected, the user must input

BMX - the desired non-zero elements of the B-matrix. (The user must be certain that the resulting B-matrix has rank KDIM).

For example, if n=12 (the dimension of the signatures) and

BMX(2) = 2.5, BMX(15) = 1.5, BMX(36) = 1.0

then the program sets the element in the first row and second column of B to 2.5; the element in the second row and third column of B to 1.5, and the element in the third row and twelfth column of B to 1.0. All other elements of the B-matrix are set to 0.0.

If the initialization suboption DEFALT is chosen, the program logic automatically initializes the B-matrix so that a matrix of rank KDIM results. The suboption RESTRT can only be chosen after one of the initialization suboptions CHANAL, VECTOR, or DEFALT has been chosen and nominal program execution of suboption SUBSP has terminated. This occurs whenever a convergence tolerance test is passed (the change in the absolute value of the difference of the B-average divergence corresponding to two successive evaluations of the gradient is less than .1) or whenever the convergence tolerance test is not passed prior to 400 distinct evaluations of the B-average divergence. In either case, the user must input

TOL - the desired convergence tolerance.

Then the iteration counter is set back to 1, and suboption SUBSP is executed until nominal program termination occurs.

If suboption REPLCE is chosen, the best KDIM channels are displayed such that the first one displayed is the "best" single channel in that the average divergence is maximized, the second channel displayed when combined with the displayed first channel is the "best" pair of channels,.........., and the last channel displayed when combined with the previously displayed channels are the best KDIM channels. Also displayed on the same line as the i'th best channel is the average divergence for the combination of channels one through i, and the ratio of this average divergence to the average divergence computed using all the available channels. Displaying the KDIM best channels constitutes nominal termination of suboption REPLCE.

If suboption CANON is selected, the user must only input KDIM, the number of linear combinations of the n-channels to be determined. The k by n B-matrix is automatically initialized to correspond to the best k out of n channels as determined by the "Without Replacement Procedure". This is accomplished by transforming the data with a non-singular transformation (Reference 7). This non-singular transformation corresponds to a permutation of the original channels so that the resulting channel i does not necessarily correspond to the original channel i. Thus, it is recommended that suboption CANON be the last suboption executed within the FEATSL option.

In each of the suboptions, the user is allowed to input the interclass weights so that the weighted B-average divergence may be maximized. The user is asked to "type yes to input interclass weights". If anything but a yes is typed, the weights remain unchanged (preset to 1.0). If the user's response is yes, the weights may be input by namelist $INFEAT. The weights are input into a 10 by 10 array A. Note that A(i,j) represents the weight for the i-jth class pair. Since A(i,j) = A(j,i), an input restriction is that i be less than j. All interclass weights not input are set to 1 or to 0, depending on the value of the namelist parameter IFLG (1 or 0 respectively, default value is 1).

Upon nominal termination of any suboption, the following parameters are displayed

MAX. - The average divergence computed using all the channels

AVER. DIVER - The average divergence computed using the optimal B-matrix (suboption SUBSP) or the best KDIM channels (Suboption REPLCE)

RATIO - The ratio of AVER. DIVERG over MAX.

Since a given KDIM channel combination constitutes a B-matrix, both the optimal B-matrix or the best KDIM channel combination will be referred to below as "the B-matrix". At this point, the user is given the option to display the pairwise interclass divergences. If the user types YES the program will display for all distinct class pairs, the interclass divergence computed using all the channels, the interclass divergence computed using the B-matrix, and the corresponding ratio of these interclass divergences. Then the user is given the option to display the "Separability to be Gained Map". If the response is YES, the following parameters may be input by namelist $INPUT.

MAXX - Maximum value of the interclass divergence computed using all the channels to be displayed by the Separability to be Gained Map. (Preset to 1000)

ICODE - Determines size of the Separability to be Gained Map. (Preset to 1)

= 0; batch run

= 1; interactive run

ILABLX - Allows user to type in a label for the X-axis

ILABLY - Allows user to type in a label for the Y-axis.

At this point the user can display the transformed signatures (means and covariances) by typing YES. The transformed signatures for all the classes, computed in the subspace spanned by the rows of the B-matrix are displayed. The transformed signatures can also be stored on a signature file for future use by other options in ASTEP. The B-matrix (transformation matrix) can be displayed and/or stored on a signature file in a similar manner.

When stored on a signature file, the kxn B-matrix is transposed and stored in the covariance matrix position on the file. The remaining space in the nxn covariance matrix position is filled with zeros. The mean vector position on the file is filled with zeros. The k-dimension of the B-matrix is placed in the location normally used for the number of pixels used in the signature generation NUM(1). The dimension in the signature file, ND, is set to n. The original n channel numbers are stored in the signature file as K(I), (I=1,...,n).

When the B-matrix is retrieved in the TRNSFM option, these channel numbers are compared with the channel numbers on DATUNT. If the channel numbers disagree, the user is informed that a discrepancy exists and is asked for another B-matrix name.

If suboption SUBSP has been executed, the user may now choose any of the available FEATSL suboptions. For example, the user may wish to find the optimal B-matrix for a different value of KDIM, and thus would reselect suboption SUBSP. If suboption REPLCE has been executed, then upon completing the output options the user is given the option to continue suboption REPLCE. If the user's response is YES, he must input a 1-digit integer corresponding to the number of additional channels to be selected. For example, if 2 is input, the best KDIM + 2 channels will be obtained. When the user eventually elects not to continue suboption REPLCE, control is returned to the FEATSL driver and the user may select any of the available FEATSL suboptions.

FEATSL OPTION

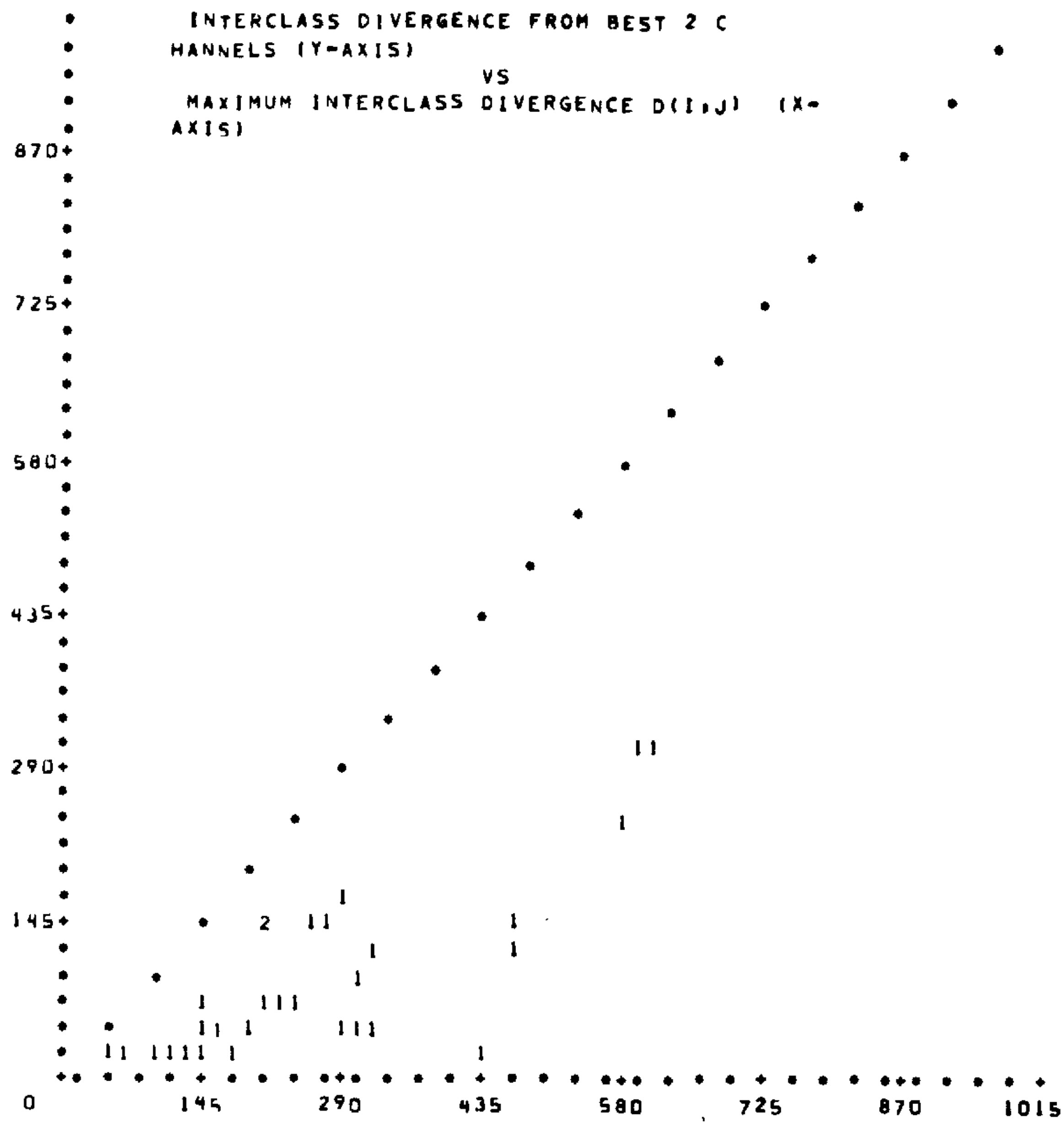
SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>FEATSL

                    FEATSL  OPTION
                    ==============

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>2

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  2
>YES

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>SIG1

SIG1   NO = 12 K =    1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    100

                MEAN        1 BY 12
            1          2          3          4          5          6

 1     84.830     79.750     61.550     62.320     85.630     87.920
            7          8          9         10         11         12

 1     64.400     83.560     69.880     81.810     93.860     73.750


                COVMAT      12 BY 12
            1          2          3          4          5          6

 1      6.730      5.400      3.950      3.860      7.690      7.030
 2      5.400      7.590      3.850      4.180      8.250      7.940
 3      3.950      3.850      3.780      2.740      5.400      5.140
 4      3.860      4.180      2.740      3.780      6.100      5.830
 5      7.690      8.250      5.400      6.100     14.050     11.920
 6      7.030      7.940      5.140      5.830     11.920     13.030
 7      4.680      5.080      3.560      3.850      7.540      7.630
 8      6.940      7.940      5.190      6.050     12.070     11.980
 9      4.970      5.960      3.800      4.600      8.900      8.790
10      5.610      6.590      4.470      5.140     10.430     10.730
11      5.650      5.910      4.550      4.540     10.210     11.670
12      2.300      2.740      1.590      2.170      5.060      6.140
```

	7	8	9	10	11	12
1	4.680	6.940	4.970	5.610	5.650	2.300
2	5.080	7.940	5.960	6.590	5.910	2.740
3	3.560	5.190	3.800	4.470	4.550	1.590
4	3.850	6.050	4.600	5.140	4.540	2.170
5	7.540	12.070	8.900	10.430	10.210	5.060
6	7.630	11.980	8.790	10.730	11.670	6.140
7	5.890	7.780	5.980	7.260	7.480	3.580
8	7.780	14.190	10.050	11.740	9.600	5.420
9	5.980	10.050	9.170	9.280	6.130	3.710
10	7.260	11.740	9.280	13.110	10.390	5.790
11	7.480	9.600	6.130	10.390	28.440	9.370
12	3.580	5.420	3.710	5.790	9.370	9.720

```
>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>2

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  2
>NO

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>SIG2

SIG2    ND = 12 K =    1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    100
>SIG3

SIG3    ND = 12 K =    1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    100
>SIG4

SIG4    ND = 12 K =    1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    100
>SIG5

SIG5    ND = 12 K =    1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    100
>SIG6

SIG6    ND = 12 K =    1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    100
>SIG7

SIG7    ND = 12 K =    1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    100
>SIG8

SIG8    ND = 12 K =    1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    100
>SIG9
```

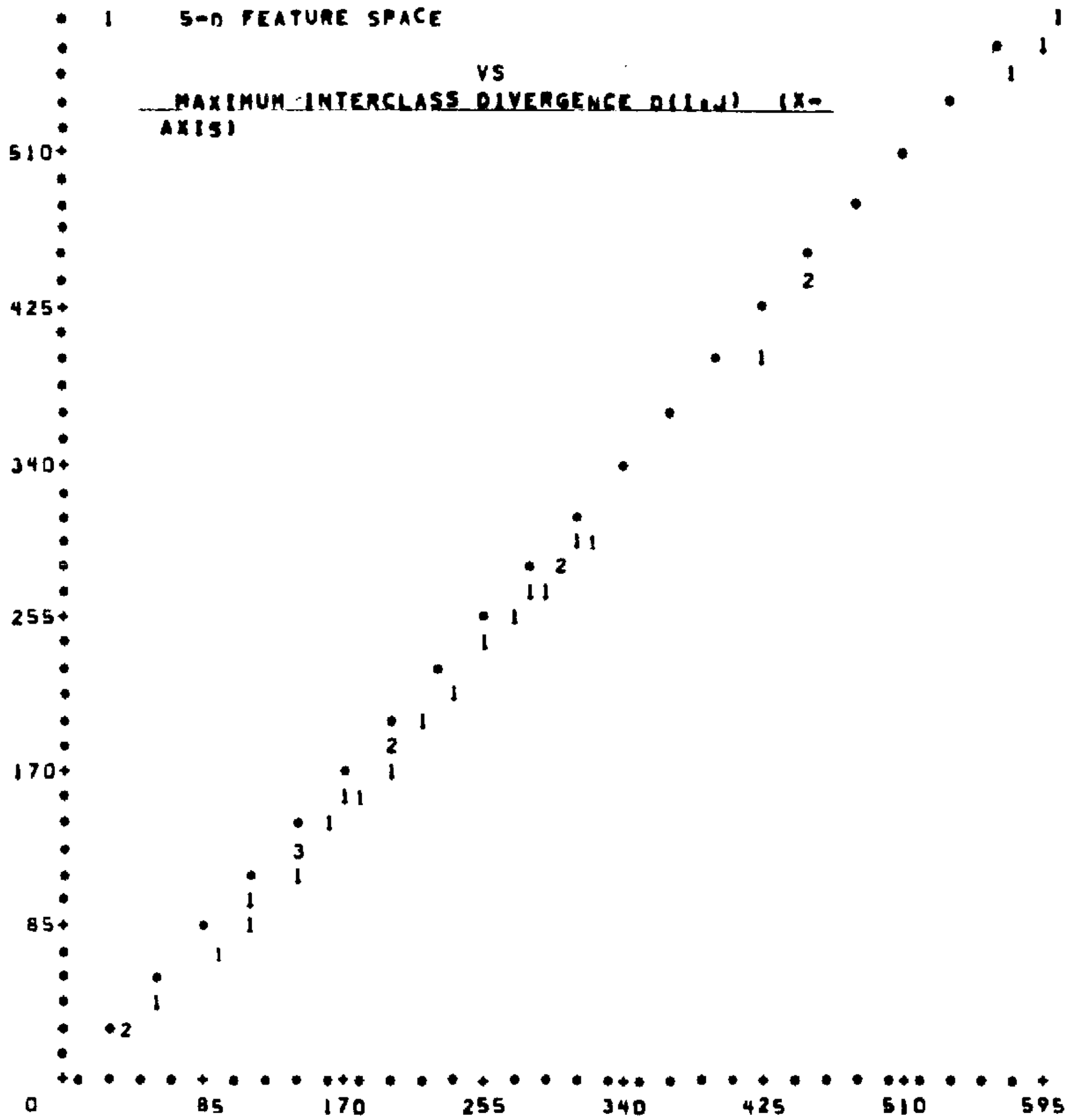
```
SIG9   ND = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(I) =   100
>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
  9 SIGNATURES HAVE BEEN RETRIEVED

CHOOSE FEATURE SELECTION OPTION FROM
  SUBSP   REPLCE  QUIT
  CANON
>REPLCE

$INFEAT      ICHAN,BMX,KDIM,TOL,A(I,J),IFLG
NUMBER  OF  CHANNELS  SELECTED  WILL  BE   2
TYPE YES IF INPUTS ARE OK
>YES

TYPE YES TO INPUT INTERCLASS WEIGHTS
>NO

CHANNEL  SELECTED=12           AVERAGE  DIVERGENCE=   64.513           RATIO= .1739
CHANNEL  SELECTED= 9           AVERAGE  DIVERGENCE=  157.368           RATIO= .4243

MAX.=   370.917    AVER. DIVERG=   155.368    RATIO=  .4189


TYPE YES TO DISPLAY INTERCLASS DIVERGENCES
>NO

TYPE YES TO DISPLAY SEPARABILITY TO BE GAINED MAP
>YES

$INPUT       MAXX, ILABLX, ILABLY, ICODE

MAXX = 1000      ICODE = 1

 MAXIMUM INTERCLASS DIVERGENCE D(I,J)  (X-AXIS)

 INTERCLASS DIVERGENCE FROM BEST 2 CHANNELS (Y-AXIS)

TYPE YES IF INPUTS ARE OK
>YES
```


INTERCLASS DIVERGENCE FROM BEST 2 C
HANNELS (Y-AXIS)
VS
MAXIMUM INTERCLASS DIVERGENCE D(I,J) (X-
AXIS)
870
725
580
435
290
145
0
145
290
435
580
725
870
1015

```
TYPE YES TO DISPLAY TRANSFORMED COVARIANCES AND MEAN VECTORS
>NO
TYPE YES TO SAVE TRANSFORMED SIGNATURES ON SIGNATURE FILE
>NO

TYPE YES TO DISPLAY FEATURE SPACE MATRIX
>NO
TYPE YES TO SAVE B-MATRIX ON SIGNATURE FILE
>NO

TYPE YES TO CONTINUE THE WITHOUT REPLACEMENT PROCEDURE
>NO

CHOOSE FEATURE SELECTION OPTION FROM
  SUBSP   REPLCE  QUIT
  CANON
>SUBSP

TYPE  IN  DIMENSION  OF  FEATURE  SPACE

$INFEAT      ICHAN,BMX,KDIM,TOL,A(I,J),IFLG
SELECTED  DIMENSION  EQUALS       5
TYPE YES IF INPUTS ARE OK
>YES

 INITIALIZE THF B-MATRIX BY SELECTING ONE OF THE FOLLOWING OPTIONS
CHANAL    VECTOR    DEFALT    RESTRT
>DEFALT

TYPE YES TO INPUT INTERCLASS WEIGHTS
>NO

MAX.=   370.917    AVER. DIVERG=   362.202    RATIO=  .9765


TYPE YES TO DISPLAY INTERCLASS DIVERGENCES
>YES

    CLASS          MAXIMUM          COMPUTED          RATIO

   1    2            30.9              27.4           .8871
   1    3           134.8             130.8           .9706
   1    4           298.7             290.6           .9729
   1    5           191.9             184.6           .9618
   1    6           188.5             182.1           .9663
   1    7           107.2             103.7           .9679
   1    8           247.2             244.8           .9903
   1    9           206.4             196.6           .9522
   2    3           164.7             156.6           .9510
   2    4           444.8             435.2           .9783
   2    5            99.9              87.4           .8748
   2    6            81.6              67.7           .8299
   2    7           152.3             144.1           .9458
```

2	8	592.4	585.5	.9884
2	9	314.1	301.6	.9602
3	4	131.7	128.9	.9782
3	5	128.4	120.2	.9362
3	6	170.2	155.6	.9146
3	7	131.6	126.5	.9612
3	8	444.3	435.5	.9802
3	9	186.9	173.2	.9269
4	5	1012.4	1007.1	.9948
4	6	1228.0	1216.3	.9905
4	7	292.8	285.6	.9755
4	8	410.9	393.1	.9567
4	9	281.5	270.9	.9624
5	6	26.4	24.9	.9420
5	7	275.6	266.1	.9655
5	8	1655.3	1650.4	.9971
5	9	581.9	564.2	.9695
6	7	260.4	253.3	.9727
6	8	1748.9	1743.2	.9968
6	9	566.2	546.4	.9651
7	8	225.1	218.5	.9710
7	9	47.2	43.2	.9152
8	9	291.8	277.2	.9499

```
TYPE YES TO DISPLAY SEPARABILITY TO BE GAINED MAP
>YES

$INPUT       MAXX, ILABLX, ILABLY, ICODE

MAXX =  600      ICODE = 1

 MAXIMUM INTERCLASS DIVERGENCE D(I,J)  (X-AXIS)

 5-D FEATURE SPACE

TYPE YES IF INPUTS ARE OK
>YES
```


1 5-D FEATURE SPACE
VS
MAXIMUM INTERCLASS DIVERGENCE D(I,J) (X-AXIS)
510
425
340
255
170
85
0
85
170
255
340
425
510
595

```
TYPE YES TO DISPLAY TRANSFORMED COVARIANCES AND MEAN VECTORS
>NO
TYPE YES TO SAVE TRANSFORMED SIGNATURES ON SIGNATURE FILE
>NO

TYPE YES TO DISPLAY FEATURE SPACE MATRIX
>NO
TYPE YES TO SAVE B-MATRIX ON SIGNATURE FILE
>NO

CHOOSE FEATURE SELECTION OPTION FROM
  SUBSP   REPLCE  QUIT
  CANON
>QUIT
 THE OPTION FEATSL REQUIRED     58.1468 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

FEATSL ENGINEERING DESCRIPTION

The purpose of option FEATSL is to determine a linear transformation which reduces the dimension of the data to be processed from n to k, where k is less than n.

It is desired to maximize the average divergence D_B, for m-distinct classes.

$$D_B = \tilde{c}_1 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} D_B(i,j)$$

where $\tilde{c}_1$: a constant equal to $2/(m^2-m)$

$D_B(i,j)$: the transformed interclass divergence for classes i and j.

Define the weighted average divergence to be

$$D_B = \tilde{c}_1 \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} c_{ij} D_B(i,j)$$

where $c_{ij} = c_{ji}$ are suitably chosen weights.

The weighted average divergence may be written as

$$D_B = 1/2\ \tilde{c}_1\ \mathrm{tr} \left\{ \sum_{i=1}^{m} (B \Lambda_i B^T)^{-1} (B\ S_i\ B^T) \right\} - k\ \tilde{c}_1\ \tilde{c}_2$$

where

tr : denotes the trace of a matrix

B : a k by n matrix of rank k

Λ_i : an n by n positive definite covariance matrix for the i'th class

μ_i : an n-dimensional vector representing the mean of the i'th class

δ_{ij}: $\mu_i - \mu_j$

$$S_i \;:\; \sum_{\substack{j=1 \\ j \neq i}}^{m} c_{ij}[\Lambda_j + \delta_{ij}\delta_{ij}^T]$$

$$\tilde{c}_2 \;:\; \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} c_{ij}$$

Note that the weights are "absorbed" by the matrices S_i, $i=1,\ldots,m$, so that functionally the weights do not affect the problem. However, numerically the weights will affect the solution, i.e., that B which maximizes D_B.

Thus, for a given set of weights c_{ij}, it is desired to maximize the weighted average divergence D_B. Unfortunately, the best B cannot be obtained analytically and must be obtained numerically. Thus, the gradient of D_B must be computed. It is readily verified that

$$\left(\frac{\partial D_B}{\partial B}\right)^T = \tilde{c}_1 \sum_{j=1}^{m} \left[S_i B^T - \Lambda_i\; B^T\; (B\; \Lambda_i\; B^T)^{-1} (B\; S_i\; B^T)\right] (B\; \Lambda_i\; B^T)^{-1}$$

The above expression is a matrix corresponding to the transpose of the partial derivative of D_B with respect to B. By using the above expression for the gradient and iteratively changing the B-matrix in accordance with Davidon's algorithm (Reference 8) it is possible to satisfy, while increasing D_B,

$$\left(\frac{\partial D_B}{\partial B}\right)^T = 0$$

i.e., it is possible to obtain a B-matrix which maximizes D_B. However, this may not necessarily be a global maximum. Thus, it is recommended that the best k out of n channels be chosen as the initial guess for the B-matrix. This will increase the probability that the maximum found iteratively is the global maximum.

An output option, available to the user, is to display the so-called interclass divergence resulting from the best linear transformation B (again, B is a k by n matrix which may correspond to the best k channels). The B-interclass divergence between classes i and j is defined as

$$D_B(i,j) = 1/2 \, \mathrm{tr} \left\{ (B\Lambda_i B^T)^{-1} B(\Lambda_j + \delta_{ij}\delta_{ij})B^T \right.$$

$$\left. + (B\Lambda_j B^T)^{-1} \, B(\Lambda_i + \delta_{ij}\delta_{ij})B^T \right\} - 1$$

The average divergence between classes i and j computed using all the channels is

$$D(i,j) = 1/2 \ \mathrm{tr} \left\{ \Lambda_i^{-1}(\Lambda_j + \delta_{ij}\delta_{ij}^T) + \Lambda_j^{-1}(\Lambda_i + \delta_{ij}\delta_{ij}^T) \right\} -1$$

It is noted that:

$$D(i,j) \geq D_B(i,j)$$

and

$$D_B = \left(\frac{2}{(m)\ (m-1)}\right) \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} D_B(i,j)$$

Thus, defining

$$D = \left(\frac{2}{(m)\ (m-1)}\right) \sum_{i=1}^{m-1} \sum_{j=i+1}^{m} D\ (i,j)$$

it follows

$$D \geq D_B$$

The program always displays the ratio

$$\frac{D_B}{D} \leq 1$$

and the user has the option to display

$$\frac{D_B(i,j)}{D\ (i,j)}$$

for all distinct class pairs.

It is noted that the B-interclass divergence $D_B(i,j)$ is a measure of the "separability" between classes i and j, with in general, the larger the interclass divergence, the easier it is to distinguish between these classes. Since $D(i,j) - D_B(i,j) \geq 0$, this difference is a measure of the "separability loss" for this class pair by performing the transformation $y = Bx$. Thus, it is desired to make the difference $D(i,j) - D_B(i,j)$ small for each distinct class pair. For a given k and "best" B obtained from any suboption, the user can graphically display the concept of separability by displaying what is called a "Separability to be Gained Map". Thus, consider a coordinate system whose (y-coordinate) ordinate (for a fixed value of k) is $D_B(i,j)$. The abscissa (x-coordinate) is the value of $D(i,j)$. The Separability to be Gained Map displays for each distinct class pair a point whose x-coordinate is $D(i,j)$ and whose y-coordinate is $D_B(i,j)$. Also displayed is a diagonal line corresponding to $D(i,j) = D_B(i,j)$. Thus, the distance of a given point from the diagonal line represents the separation to be gained for this class pair. Ideally, the B-matrix should make this separation to be gained small for all distinct class pairs.

Canonical Formulation

Note that since B is a k by n matrix, kn distinct elements or parameters must be determined by the optimization program, so that D_B is maximized. In this case, the Davidon technique iteratively updates an array H of dimension kn by kn to obtain the best B. Thus, to conserve computer storage the product p - k^2n^2 must not be too large. Note that when k = 12 and n = 24, p = 82,944, and if H is a double precision variable, then 165,888 decimal words of computer storage must be allocated for H. This is clearly prohibitive; yet it is desired to be able to solve the feature selection problem when $k \leq 12$ and n = 24. This can be accomplished as described below.

Let Q be a nonsingular k by k matrix. Then QB is a k by n matrix of rank k, and it is readily verified that $D_B = D_{QB}$. In particular, if B = (R S) where R is a k by k nonsingular matrix, let $Q = R^{-1}$. Then it follows that

$$QB = (I_k \quad R^{-1}S)$$

where I_k is a k by k identity matrix and $R^{-1}S$ is a k by n-k matrix. Thus, letting

$$\hat{B} = (I_k \quad R^{-1}S)$$

where $\hat{B}$ is a k by n matrix, it follows that only k(n-k) parameters must be determined - namely the elements of the matrix $T = R^{-1}S$, where T is a k by n-k matrix. The optimization problem is reformulated as: find $\hat{B}$ such that $D_{\hat{B}}$ is a maximum. It is readily verified that if $k \leq 12$ and n = 24, then at most 144 parameters must be determined, so that $(144)^2$ = 20,736 words of computer storage must be allocated for H. It has been experimentally determined that the H matrix need not be double precision, so that only 20,736 words of computer storage must be allocated for H. Also, only a given row of the H-matrix need be used at a time, so that it it convenient to store the H-matrix on a temporary storage device. Thus, the following procedure was applied for solving problems when $k \leq 12$ and $n \leq 24$:

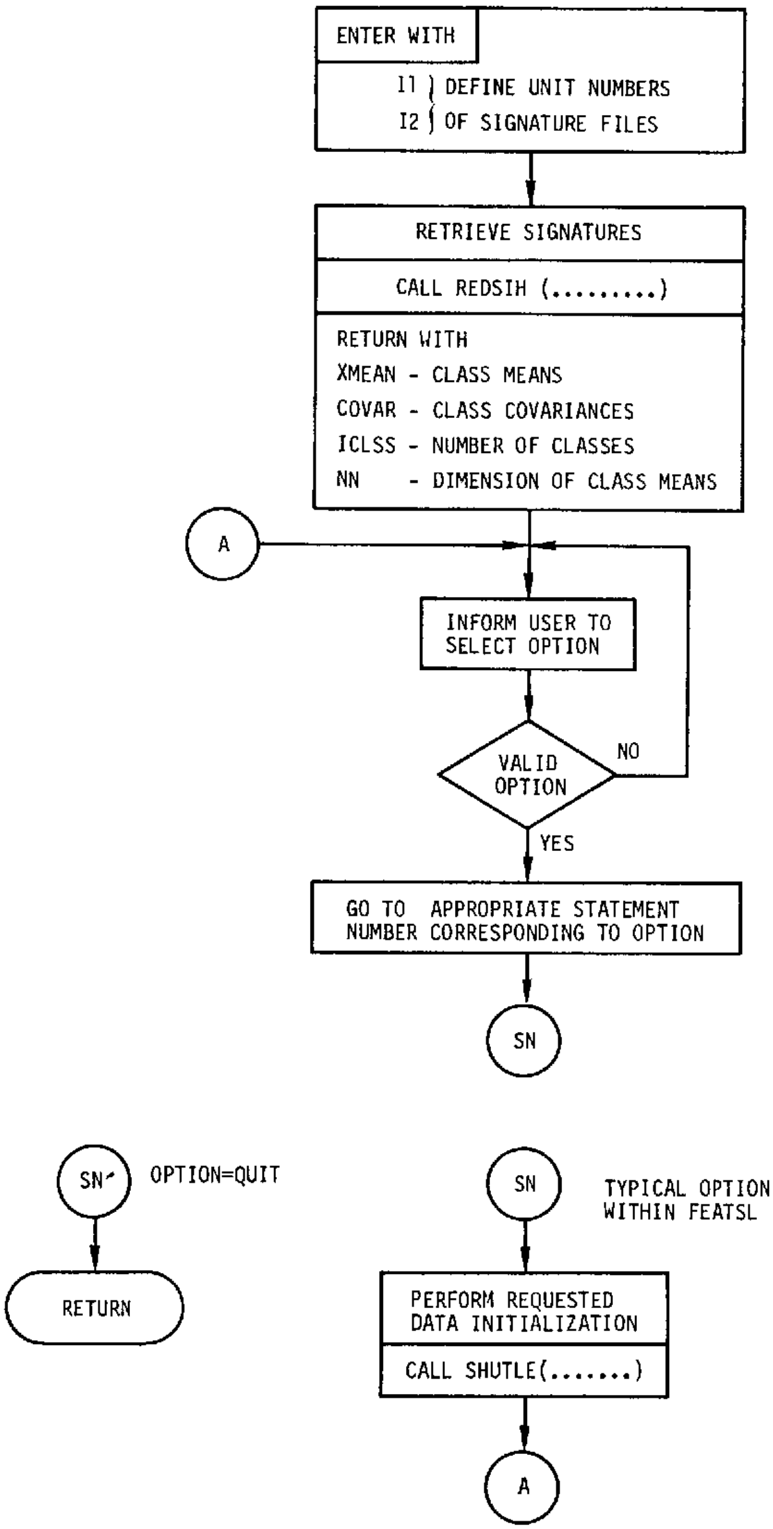
1) Write or read the first 72 rows of H from temporary storage device

2) Write or read the last 72 rows of H from temporary storage device

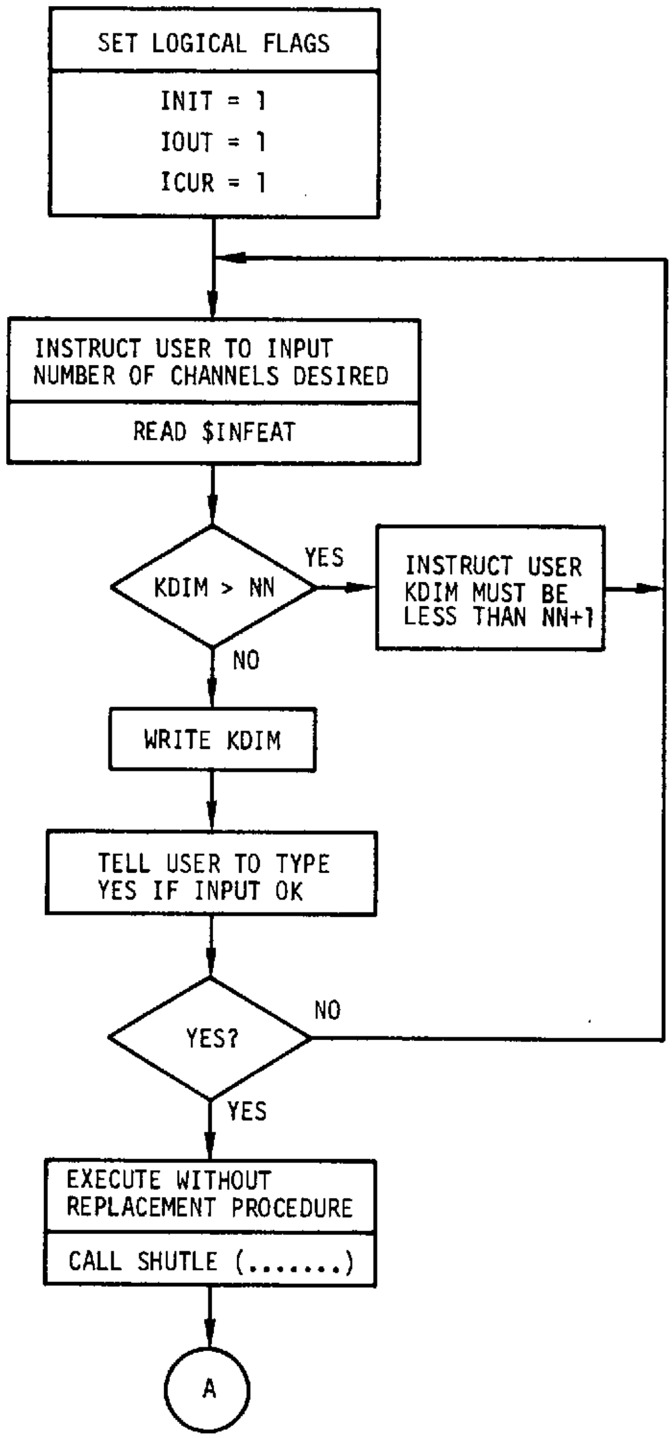
3) Write or read the statistics Λ_i, S_i, and μ_i from temporary storage device

Note that if H is small, i.e., less than 73 rows, then step 2 is omitted. Also, by judiciously using all 3 steps only 10,368 decimal words of data storage are necessary to accomodate the H-matrix and all the statistics for as many as 10 distinct classes with covariances as large as 24 by 24.

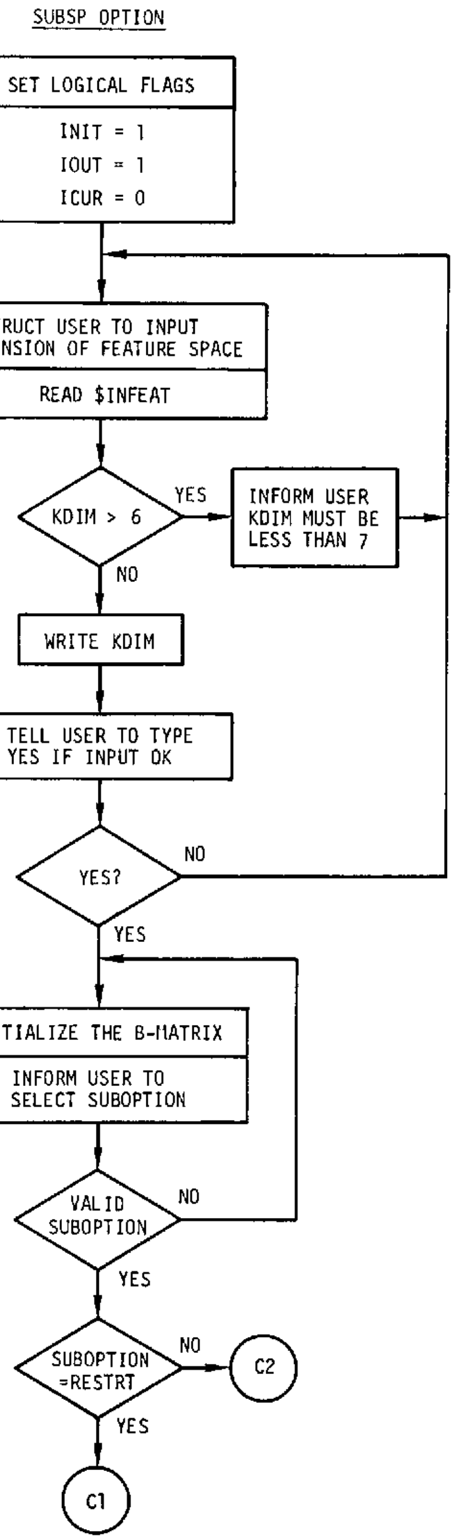
FEATSL


ENTER WITH
I1 } DEFINE UNIT NUMBERS
I2 } OF SIGNATURE FILES
RETRIEVE SIGNATURES
CALL REDSIH (.........)
RETURN WITH
XMEAN - CLASS MEANS
COVAR - CLASS COVARIANCES
ICLSS - NUMBER OF CLASSES
NN - DIMENSION OF CLASS MEANS
A
INFORM USER TO SELECT OPTION
VALID OPTION
NO
YES
GO TO APPROPRIATE STATEMENT NUMBER CORRESPONDING TO OPTION
SN
SN'
OPTION=QUIT
RETURN
SN
TYPICAL OPTION WITHIN FEATSL
PERFORM REQUESTED DATA INITIALIZATION
CALL SHUTLE(.......)
A

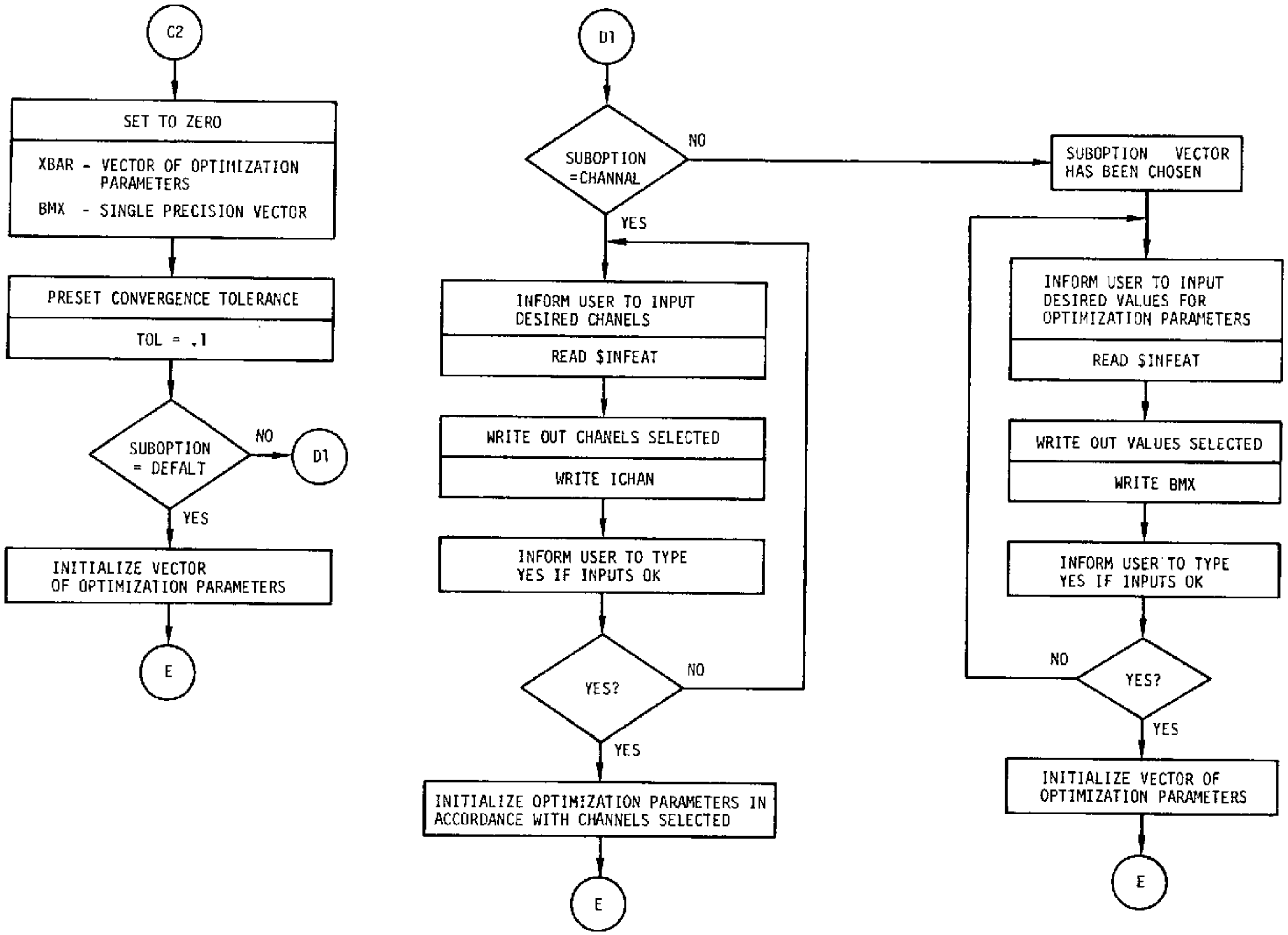
REPLCE OPTION



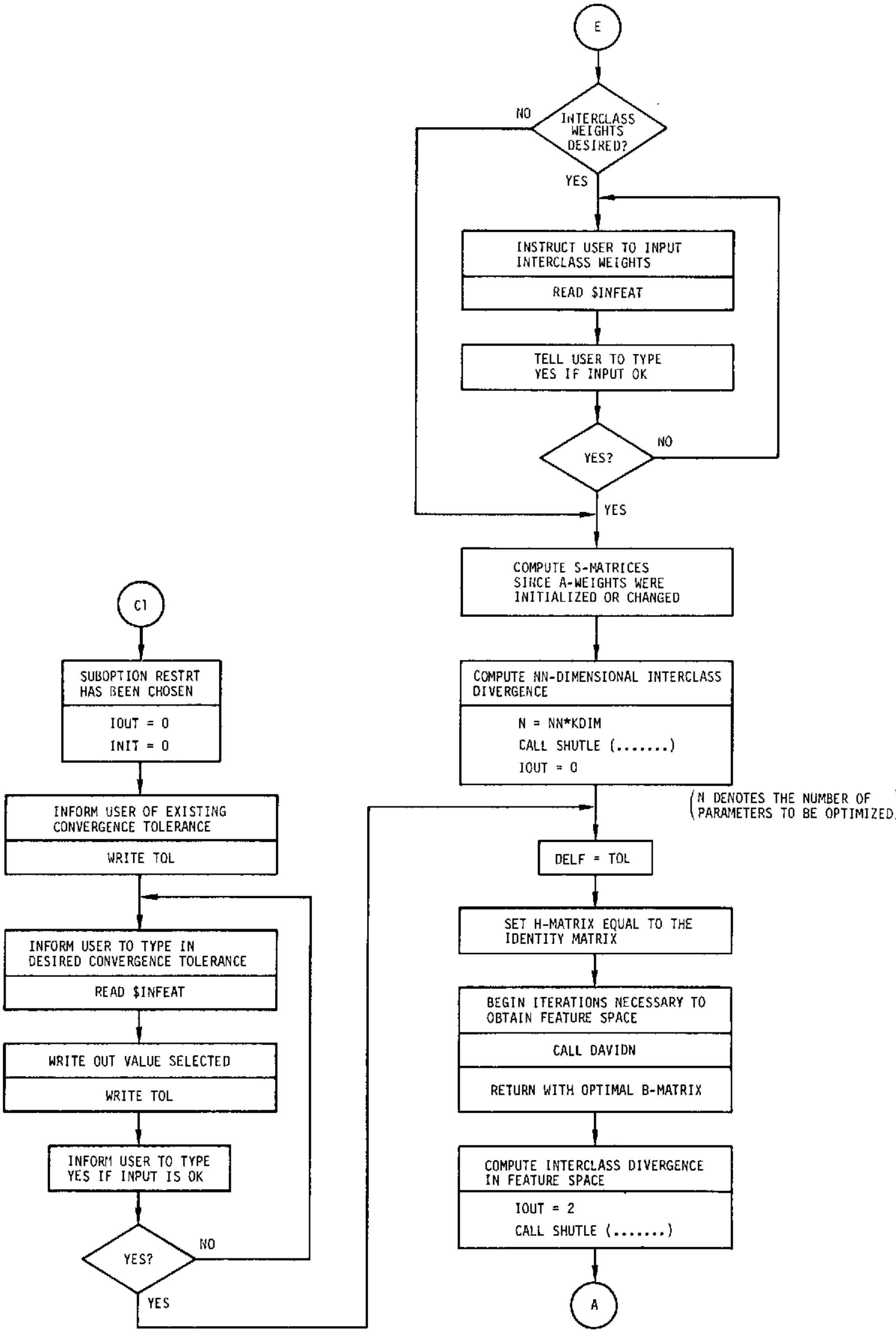
FEATSL 2 of 6
(REPLCE 1)



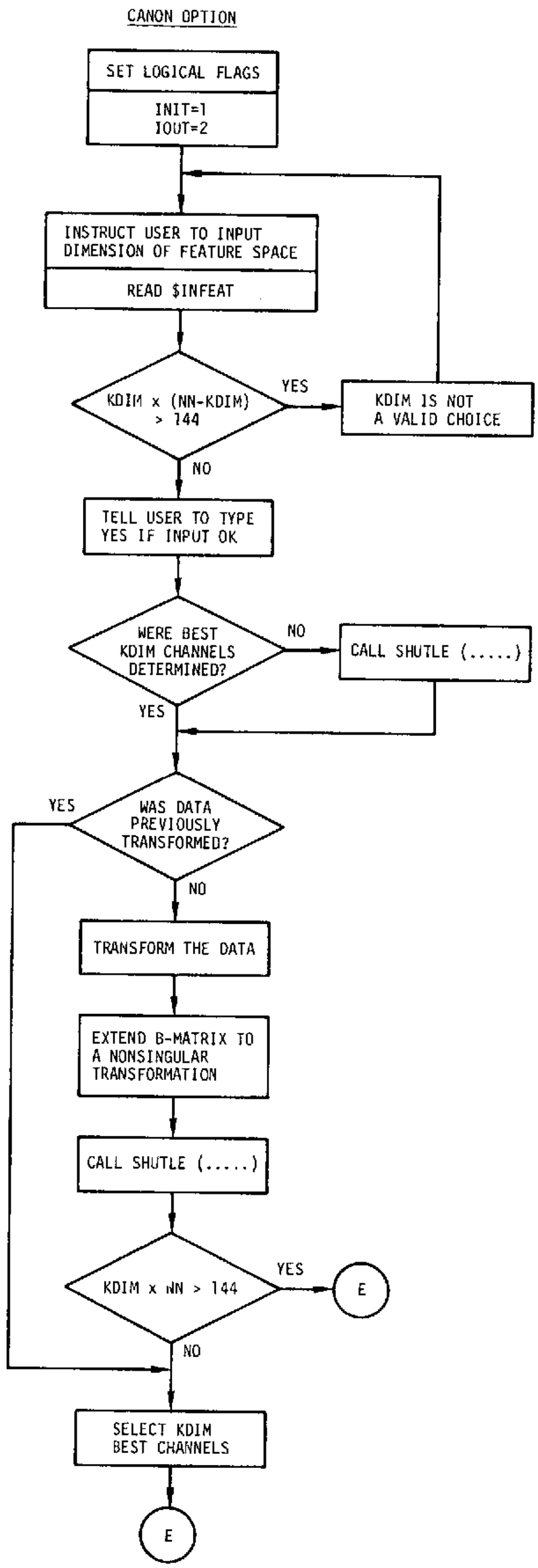
FEATSL 3 of 6
(SUBSP 1)



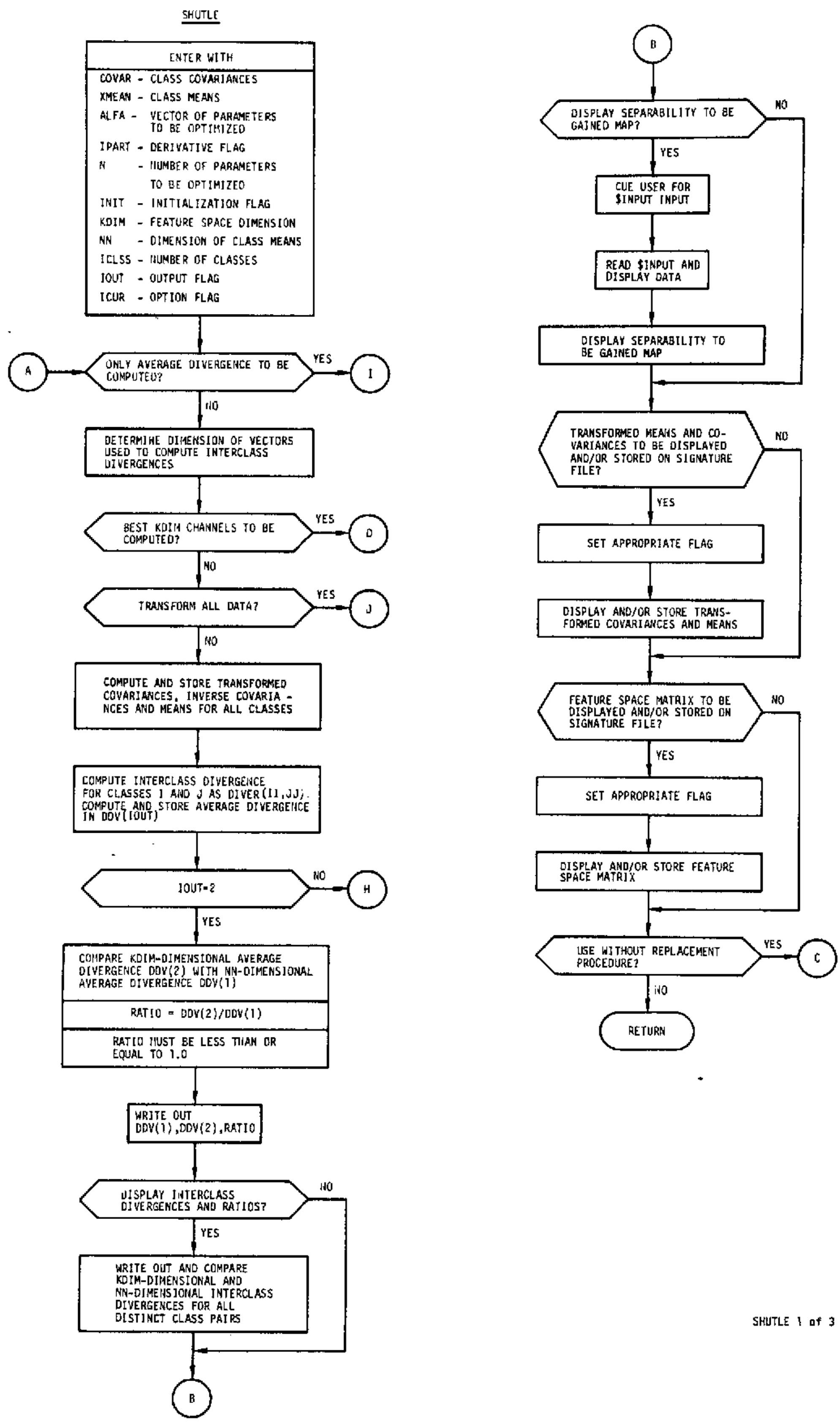
FEATSL 4 of 6
(SUBSP2)

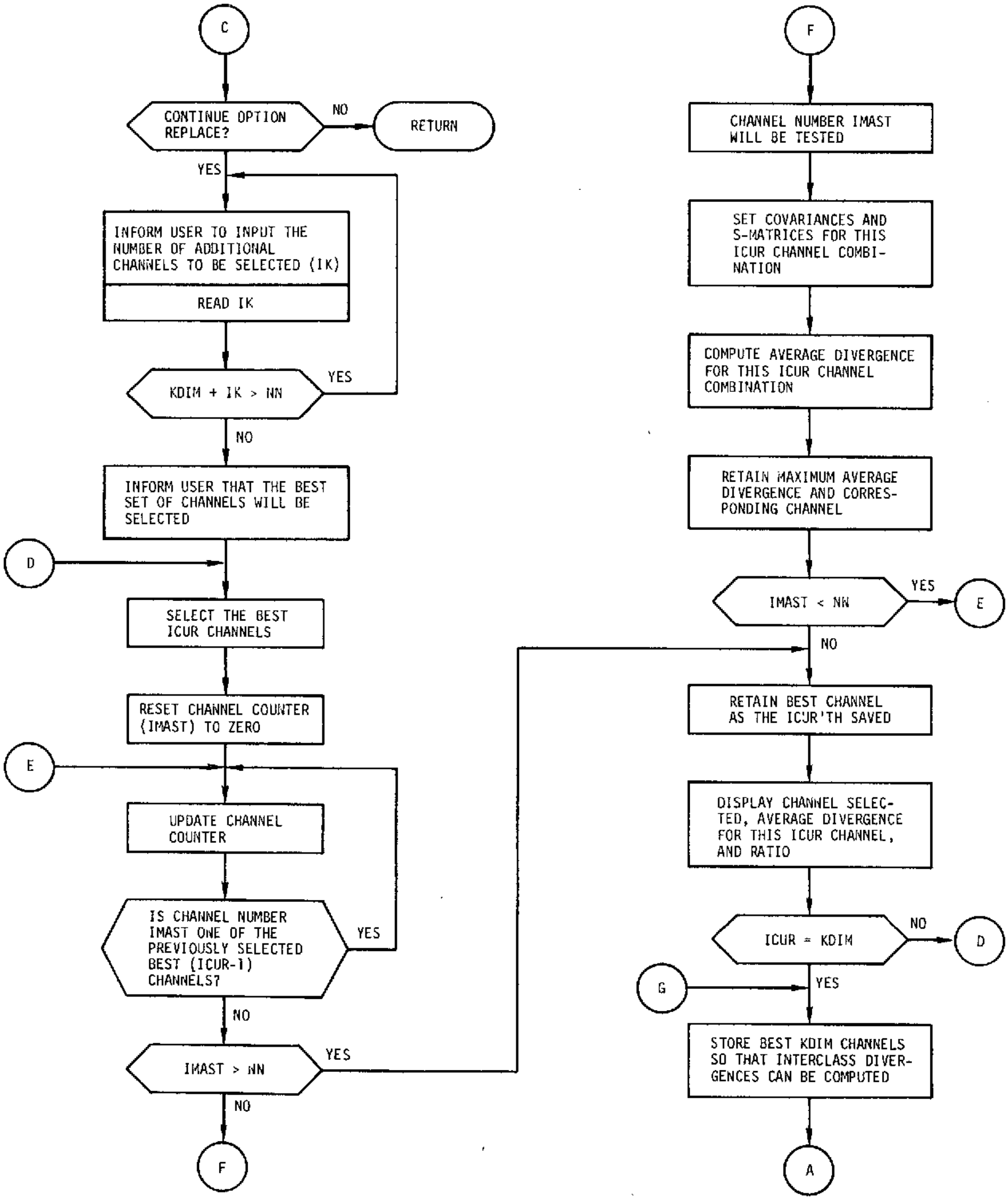


FEATSL 5 of 6
(SUBSP 3)

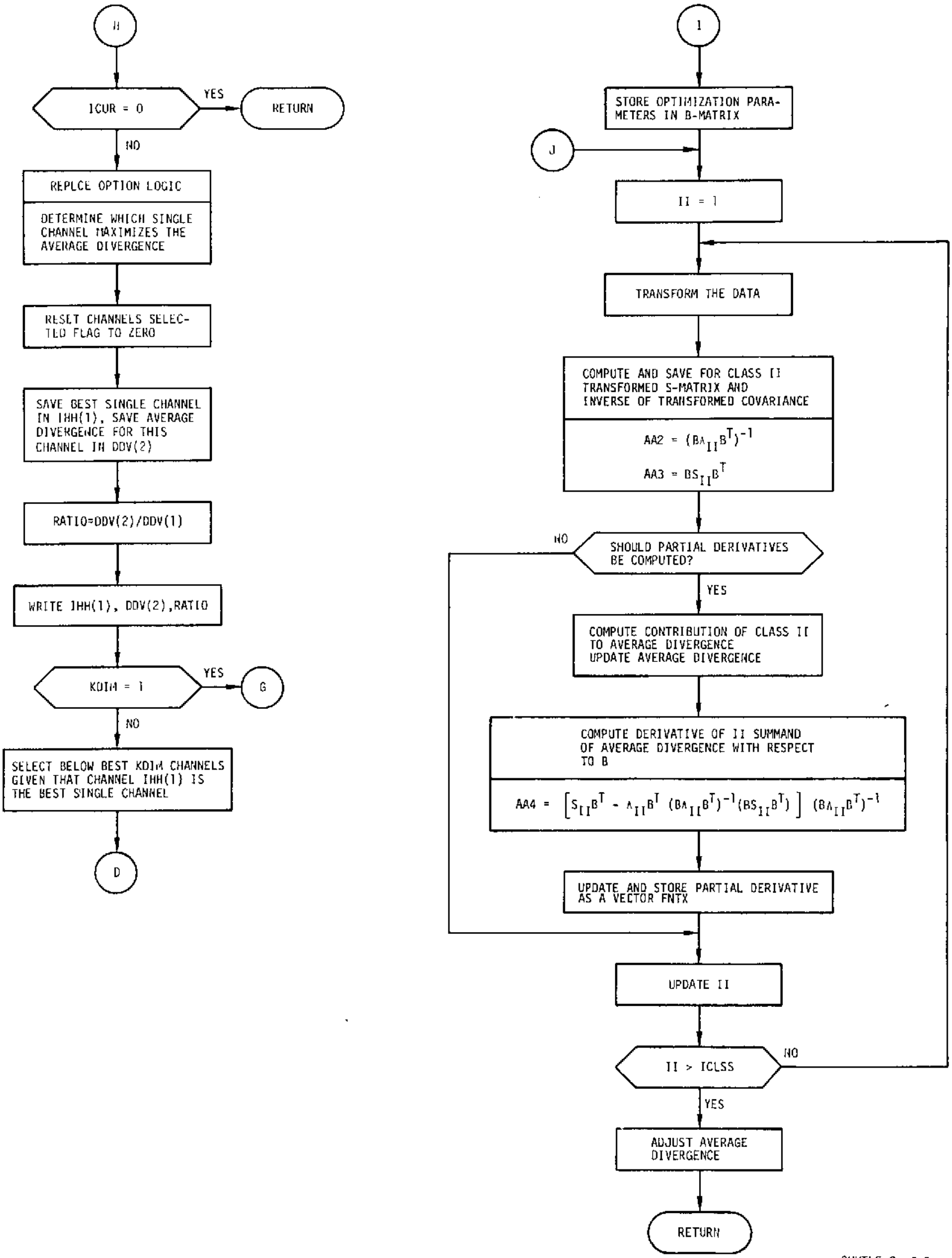


FEATSL 6 of 6
(CANON 1)


SHUTLE
ENTER WITH
COVAR - CLASS COVARIANCES
XMEAN - CLASS MEANS
ALFA - VECTOR OF PARAMETERS TO BE OPTIMIZED
IPART - DERIVATIVE FLAG
N - NUMBER OF PARAMETERS TO BE OPTIMIZED
INIT - INITIALIZATION FLAG
KDIM - FEATURE SPACE DIMENSION
NN - DIMENSION OF CLASS MEANS
ICLSS - NUMBER OF CLASSES
IOUT - OUTPUT FLAG
ICUR - OPTION FLAG
A
ONLY AVERAGE DIVERGENCE TO BE COMPUTED?
YES
I
NO
DETERMINE DIMENSION OF VECTORS USED TO COMPUTE INTERCLASS DIVERGENCES
BEST KDIM CHANNELS TO BE COMPUTED?
YES
D
NO
TRANSFORM ALL DATA?
YES
J
NO
COMPUTE AND STORE TRANSFORMED COVARIANCES, INVERSE COVARIA - NCES AND MEANS FOR ALL CLASSES
COMPUTE INTERCLASS DIVERGENCE FOR CLASSES I AND J AS DIVER(II,JJ). COMPUTE AND STORE AVERAGE DIVERGENCE IN DDV(IOUT)
IOUT=2
NO
H
YES
COMPARE KDIM-DIMENSIONAL AVERAGE DIVERGENCE DDV(2) WITH NN-DIMENSIONAL AVERAGE DIVERGENCE DDV(1)
RATIO = DDV(2)/DDV(1)
RATIO MUST BE LESS THAN OR EQUAL TO 1.0
WRITE OUT DDV(1),DDV(2),RATIO
DISPLAY INTERCLASS DIVERGENCES AND RATIOS?
NO
YES
WRITE OUT AND COMPARE KDIM-DIMENSIONAL AND NN-DIMENSIONAL INTERCLASS DIVERGENCES FOR ALL DISTINCT CLASS PAIRS
B
B
DISPLAY SEPARABILITY TO BE GAINED MAP?
NO
YES
CUE USER FOR $INPUT INPUT
READ $INPUT AND DISPLAY DATA
DISPLAY SEPARABILITY TO BE GAINED MAP
TRANSFORMED MEANS AND CO-VARIANCES TO BE DISPLAYED AND/OR STORED ON SIGNATURE FILE?
NO
YES
SET APPROPRIATE FLAG
DISPLAY AND/OR STORE TRANS-FORMED COVARIANCES AND MEANS
FEATURE SPACE MATRIX TO BE DISPLAYED AND/OR STORED ON SIGNATURE FILE?
NO
YES
SET APPROPRIATE FLAG
DISPLAY AND/OR STORE FEATURE SPACE MATRIX
USE WITHOUT REPLACEMENT PROCEDURE?
YES
C
NO
RETURN
SHUTLE 1 of 3

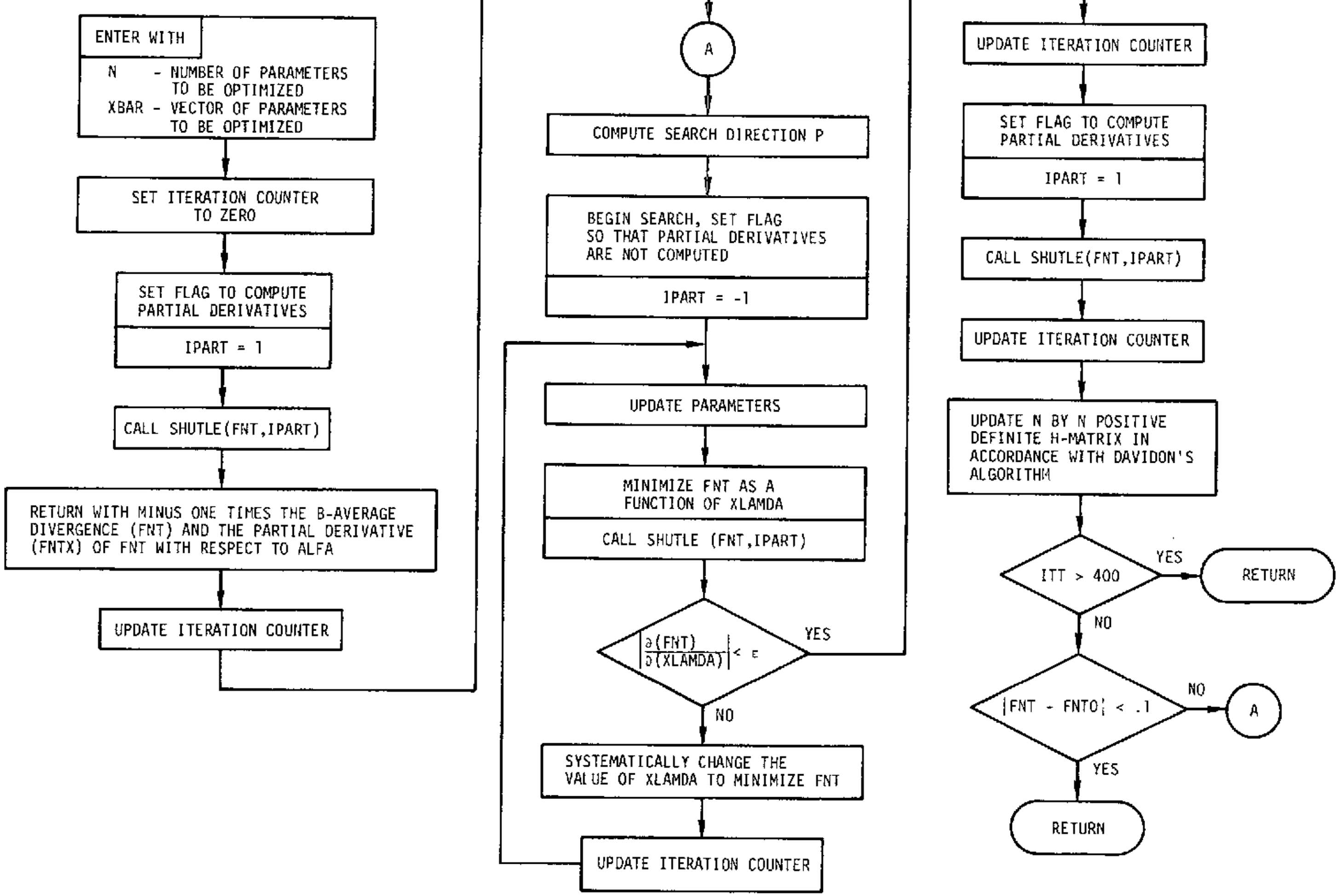

C
CONTINUE OPTION REPLACE?
NO
RETURN
YES
INFORM USER TO INPUT THE NUMBER OF ADDITIONAL CHANNELS TO BE SELECTED (IK)
READ IK
KDIM + IK > NN
YES
NO
INFORM USER THAT THE BEST SET OF CHANNELS WILL BE SELECTED
D
SELECT THE BEST ICUR CHANNELS
RESET CHANNEL COUNTER (IMAST) TO ZERO
E
UPDATE CHANNEL COUNTER
IS CHANNEL NUMBER IMAST ONE OF THE PREVIOUSLY SELECTED BEST (ICUR-1) CHANNELS?
YES
NO
IMAST > NN
YES
NO
F
F
CHANNEL NUMBER IMAST WILL BE TESTED
SET COVARIANCES AND S-MATRICES FOR THIS ICUR CHANNEL COMBINATION
COMPUTE AVERAGE DIVERGENCE FOR THIS ICUR CHANNEL COMBINATION
RETAIN MAXIMUM AVERAGE DIVERGENCE AND CORRESPONDING CHANNEL
IMAST < NN
YES
E
NO
RETAIN BEST CHANNEL AS THE ICUR'TH SAVED
DISPLAY CHANNEL SELECTED, AVERAGE DIVERGENCE FOR THIS ICUR CHANNEL, AND RATIO
ICUR = KDIM
NO
D
YES
G
STORE BEST KDIM CHANNELS SO THAT INTERCLASS DIVERGENCES CAN BE COMPUTED
A

SHUTLE 2 of 3


H
ICUR = 0
YES
RETURN
NO
REPLCE OPTION LOGIC
DETERMINE WHICH SINGLE CHANNEL MAXIMIZES THE AVERAGE DIVERGENCE
RESET CHANNELS SELECTED FLAG TO ZERO
SAVE BEST SINGLE CHANNEL IN IHH(1), SAVE AVERAGE DIVERGENCE FOR THIS CHANNEL IN DDV(2)
RATIO=DDV(2)/DDV(1)
WRITE IHH(1), DDV(2),RATIO
KDIM = 1
YES
G
NO
SELECT BELOW BEST KDIM CHANNELS GIVEN THAT CHANNEL IHH(1) IS THE BEST SINGLE CHANNEL
D
I
STORE OPTIMIZATION PARAMETERS IN B-MATRIX
J
II = 1
TRANSFORM THE DATA
COMPUTE AND SAVE FOR CLASS II TRANSFORMED S-MATRIX AND INVERSE OF TRANSFORMED COVARIANCE
$AA2 = (BA_{II}B^T)^{-1}$
$AA3 = BS_{II}B^T$
NO
SHOULD PARTIAL DERIVATIVES BE COMPUTED?
YES
COMPUTE CONTRIBUTION OF CLASS II TO AVERAGE DIVERGENCE
UPDATE AVERAGE DIVERGENCE
COMPUTE DERIVATIVE OF II SUMMAND OF AVERAGE DIVERGENCE WITH RESPECT TO B
$AA4 = \left[S_{II}B^T - A_{II}B^T\,(BA_{II}B^T)^{-1}(BS_{II}B^T)\right]\,(BA_{II}B^T)^{-1}$
UPDATE AND STORE PARTIAL DERIVATIVE AS A VECTOR FNTX
UPDATE II
II > ICLSS
NO
YES
ADJUST AVERAGE DIVERGENCE
RETURN

SHUTLE 3 of 3

DAVION


ENTER WITH
N - NUMBER OF PARAMETERS TO BE OPTIMIZED
XBAR - VECTOR OF PARAMETERS TO BE OPTIMIZED
SET ITERATION COUNTER TO ZERO
SET FLAG TO COMPUTE PARTIAL DERIVATIVES
IPART = 1
CALL SHUTLE(FNT,IPART)
RETURN WITH MINUS ONE TIMES THE B-AVERAGE DIVERGENCE (FNT) AND THE PARTIAL DERIVATIVE (FNTX) OF FNT WITH RESPECT TO ALFA
UPDATE ITERATION COUNTER
A
COMPUTE SEARCH DIRECTION P
BEGIN SEARCH, SET FLAG SO THAT PARTIAL DERIVATIVES ARE NOT COMPUTED
IPART = -1
UPDATE PARAMETERS
MINIMIZE FNT AS A FUNCTION OF XLAMDA
CALL SHUTLE (FNT,IPART)
$\left|\frac{\partial(FNT)}{\partial(XLAMDA)}\right| < \varepsilon$
YES
NO
SYSTEMATICALLY CHANGE THE VALUE OF XLAMDA TO MINIMIZE FNT
UPDATE ITERATION COUNTER
UPDATE ITERATION COUNTER
SET FLAG TO COMPUTE PARTIAL DERIVATIVES
IPART = 1
CALL SHUTLE(FNT,IPART)
UPDATE ITERATION COUNTER
UPDATE N BY N POSITIVE DEFINITE H-MATRIX IN ACCORDANCE WITH DAVIDON'S ALGORITHM
ITT > 400
YES
RETURN
NO
$|FNT - FNTO| < .1$
NO
A
YES
RETURN

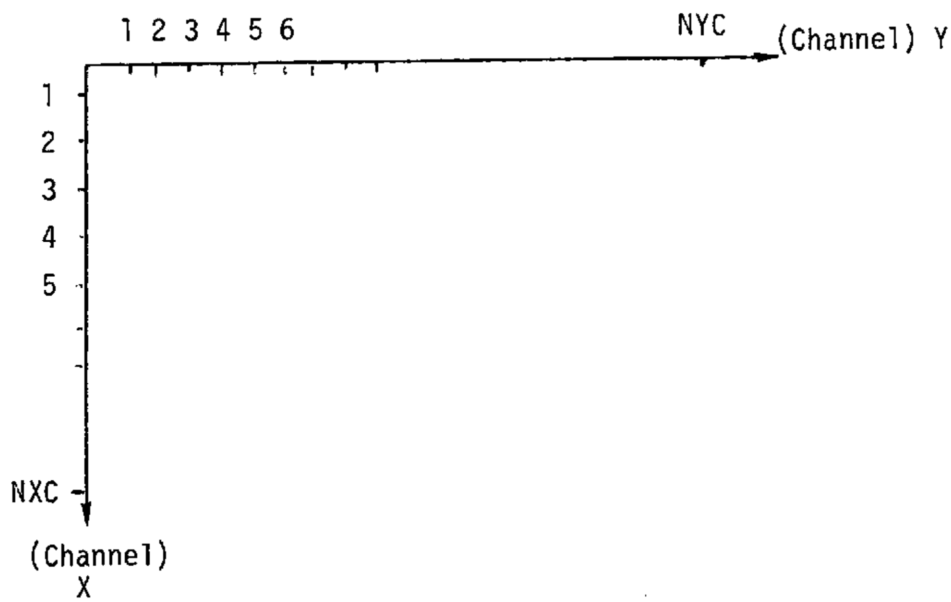
Using the HSGRAM Option

The HSGRAM option computes and displays the histogram of up to three channels of an input data set. The user selects:

1) The class within the data set to be displayed. Inputting a '+' sign indicates all classes.

2) The specific channels to be considered. It is to be noted the order in which the channel output is displayed is determined by the order of their input. That is, the first channel corresponds to the X direction, the second to the Y direction, and the third to the Z planes. The following example shows the orientation of the X and Y directions.

Z PLANE 2 ZL (lower limit) = nnn.nn ZU (upper limit) - nnn.nn



3) The limits and the number of subdivisions for each direction in the following way:

   - o The three values of each variable XMIN and XMAX are the minimums and maximums acceptable for each of the X, Y, and Z directions.

   - o The three values of NXC, NYC, and NZP specify the number of subdivisions for the acceptable regions of the X, Y, and Z directions.

The following is a more detailed description of each input variable:

XMIN - 3 values which set the lowest values of X, Y, and Z which will be considered

XMAX - 3 values which set the highest values of X, Y, and Z which will be considered

NXC - number of cells which are to be in the X-direction - maximum of 20

NYC - number of cells which are to be in the Y-direction - maximum of 20

NZP - number of planes in the Z-direction. The printing will consist of one plane at a time. There is no limit of Z planes.

KCH - which components of the data vector which are to be used in making the histogram (note that these channel numbers do not relate to the original packed data set, but represent the sequential numbers of channels in the extracted, unpacked data set. For example, if channels 1, 6, 9, 10, and 12 were extracted, KCH values 1, 2, and 4 would specify the channels 1, 6, 10 of the original data).

The output consists of a histogram for each Z plane and the summary values

NX - total number of data points in the set

NP - number of points included within the histogram limits

To exit this option one types a blank when asked for a new symbol.

HSGRAM OPTION

SAMPLES OF INPUT AND CORRESPONDING OUTPUT:

SAMPLE 1. A one dimensional histogram.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>HSGRAM


                    HSGRAM  OPTION
                    ==============


INPUT SYMBOLS FOR CLASSES.
>C
HISTOGRAM FOR C
$INHIST   XMIN,XMAX,NZP,NXC,NYC,KCH
$INHIST
XMIN    =     .63000000E+02,   .00000000E+00,   .00000000E+00
XMAX    =     .79000000E+02,   .25500000E+03,   .25500000E+03
NZP     =               +1
NXC     =              +17
NYC     =               +1
KCH     =               +3,              +4,              +1

$END
TYPE YES IF INPUTS ARE OK
>YES
```

```
Z PLANE  1  ZL =      .00  ZU =  255.00

      1

 1
 2    3
 3   17
 4   52
 5  167
 6  255
 7  301
 8  314
 9  279
10  225
11  184
12  127
13  101
14   43
15   21
16    3
17
NX =  5661   NP =  2092
 THE OPTION HSGRAM REQUIRED      1.7804 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

SAMPLE 2. A two dimensional histogram.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>HSGRAM

                    HSGRAM  OPTION
                    ==============

INPUT SYMBOLS FOR CLASSES.
>C
HISTOGRAM FOR C
$INHIST   XMIN,XMAX,NZP,NXC,NYC,KCH
$INHIST
XMIN    =     .63000000E+02,    .65000000E+02,    .00000000E+00
XMAX    =     .79000000E+02,    .81000000E+02,    .25500000E+03
NZP     =               +1
NXC     =              +17
NYC     =              +17
KCH     =               +3,               +4,               +1

$END
TYPE YES IF INPUTS ARE OK
>YES
```

Z PLANE 1 ZL = .00 ZU = 255.00

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2												1	1		1		
3					2	1	1		1	4	4	1			3		
4			1	2	3	4	6	4	1	8	9	8	4	1	1		
5		1	5	8	11	7	20	13	17	33	26	13	2	5	4	1	
6		6	2	10	20	26	27	31	35	37	27	13	5	9	3	2	
7		4	6	16	22	29	31	24	26	50	50	16	12	3	3	4	
8	1	5	3	5	16	27	32	43	32	39	50	23	11	5	12	4	
9		5	6	13	28	19	19	29	23	38	45	12	9	10	6	5	
10	2	4	12	12	12	12	24	25	19	27	31	20	5	6	3	1	
11		5	9	8	15	9	9	12	17	22	38	17	9	4	2	1	
12	1	2	3	3	7	5	12	7	12	22	22	9	7	4	2	2	
13	1		3	9	2	6	4	8	5	26	12	12	5	2	2	1	
14		3	3	1		2	1	4	6	5	6	4		2			
15	1	1	2	1	1				2	2	3	2	2	1			
16		1	1														
17																	

NX = 5661 NP = 2029
THE OPTION HSGRAM REQUIRED 1.8980 SECONDS OF CPU TIME.

SAMPLE 3. An example of a three dimensional histogram with seventeen Z planes.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>HSGRAM

                         HSGRAM  OPTION
                         *************

INPUT SYMBOLS FOR CLASSES.
>C
HISTOGRAM FOR C
$INHIST   XMIN,XMAX,NZP,NXC,NYC,KCH
$INHIST
XMIN     =      .63000000E+02,    .65000000E+02,    .80000000E+02
XMAX     =      .79000000E+02,    .81000000E+02,    .96000000E+02
NZP      =                +17
NXC      =                +17
NYC      =                +17
KCH      =                 +3,               +4,               +1

$END
TYPE YES IF INPUTS ARE OK
>YES
```

Z PLANE 1 ZL = 80.00 ZU = 80.94

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3																	
4																	
5																	
6																	
7																	
8																	
9																	
10																	
11																	
12																	
13																	
14																	
15																	
16																	
17																	

Z PLANE 2 ZL = 80.94 ZU = 81.88

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3																	
4																	
5										1							
6										1	1						
7											1						
8										1	1						
9																	
10																	
11												1					
12																	
13																	
14										1							
15																	
16																	
17																	

Z PLANE 3 ZL = 81.88 ZU = 82.82

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3																	
4																	
5											1						
6																	
7					1					1				1			
8											1	1			1		
9																	
10								1			1						
11								1			1						
12																	
13						1											
14										1							
15																	
16																	
17																	

Z PLANE 4 ZL = 82.82 ZU = 83.76

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3											1						
4							1					1	1				
5										1	1		2	1			
6							1		3	3							
7										1	1						
8											1						
9						1			1		1			2			
10									2								
11		1							1	2							
12							1		1		1						
13						1				2	1						
14																	
15																	
16																	
17																	

Z PLANE 5 ZL = 83.76 ZU = 84.71

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3									1			1					
4							1			1		1			1		
5						1	1	1	2	2	2	1				1	
6					2	1	2	3	4	6	4	1	1	1		1	
7					1	2	1	1	1	5	5	1	1				
8							1	5	2	1	2	1	1		1		
9					1		1	2	2	3	4						
10					1			1	1		3	4		1			
11				1				3		1	2				1		
12					1				1	5	1	1					
13						1				1		2					
14												1					
15												1					
16																	
17																	

Z PLANE 6 ZL = 84.71 ZU = 85.65

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2													1				
3							1				1				2		
4										2	2	2	1				
5						2	2	2	3	3	3	3		1			
6		1		1	1	3	1	3	4	6	7	1	1	3	1		
7				1		2	3	2	7	6	4	1	3			2	
8					1	2	3	7	4	11	12	4	1	2	1	1	
9							2	5	1	8	4	2	1	1	1		
10					1	2	1	1	2	3	2	3			2		
11						1	1	1	1	3	4	2	1				
12						1	1		2	2	3	1	2				
13									1		2	1					
14											1						
15	1									1							
16																	
17																	

Z PLANE 7 ZL = 85.65 ZU = 86.59

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3					1					1							
4					1	2	2	1		3	1	4	1				
5			1	4	5	1	3	1		8	6	2			1		
6		1	1	2	2	4	5	4	9	6	4	4	1	1	1		
7						3	2	3	2	14	9	4	2	1		1	
8				1		2	3	6	5	5	14	9	3		2	1	
9		1			4	3	3	1	3	8	14	4	1	1	1	1	
10			1	1		1	4	7	1	9	7	4	2	1			
11				1	1		4	1	3	4	7	3	2	3		1	
12					1		1	1	2	2	4	2	1	1			
13							1	3	1	4	2	2	2	1			
14							1		1		1	1					
15											2						
16																	
17																	

Z PLANE 8 ZL = 86.59 ZU = 87.53

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3						1					1						
4				2		1	1				2		1	1			
5			1	1	5		5	3	4	6	3	3		1			
6				2	6	8	7	7	5	7	6	2		1			
7		2	2	6	5	4	6	7	3	2	11	2	1				
8		1	1	2	1	6	9	5	8	6	3	3	1	1	2	1	
9				3	6	5	3	5	3	6	10	3	5	3	1	1	
10		1	1	4	4	3	2	8	2	8	12	3		2			
11				1	2	1		1	3	1	4	5					
12			1		1		2	2	2	4	5	1	1	1	1		
13	1			1				2	2	9	2	3		1	1		
14			1					2				1		1			
15													2				
16																	
17																	

Z PLANE 9 ZL = 87.53 ZU = 88.47

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3																	
4					1			1			1						
5				1		1	3	1	2	3	4	1			1		
6		1	1	1	2	5	6	9	5	2	3	1	2	1	1		
7			2	4	9	9	7	7	5	7	6	5	1		3	1	
8		2	1	1	4	9	5	10	2	11	6	1	1		1		
9		1	1	1	6	5	3	9	3	7	6	1	1	1		1	
10		1	2	3	2	5	11	4	3	3	1	2	1	2	1	1	
11		2	1		2	1	1	2	3	3	4	1			1		
12				2	1	1	1	2	2	2	3		1	1	1	1	
13			1	1	1			1	1	5	1	2	1				
14		1							1	1	2			1			
15									1	1							
16																	
17																	

Z PLANE 10 ZL = 88.47 ZU = 89.41

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3					1					1							
4					1	1	1	1			1						
5			1	1	1	1	3	1	3	4	2	1					
6		1			6	1	3	5	2	1	2	1					
7		2	1	5	3	5	6	2	6	8	3	1	1				
8				1	4	2	3	3	4	2	3	1	2	2			
9		2	2	2	5	1	2	2		1	4	1				1	
10	2	1	5	2	2	1	3	2	4		1	1	1				
11		1	3	2	4	2	2	2	1	2	7	1	1	1			
12	1		1	1	1		1	2	2	5	4	1	1				
13			1	4				1		2		2			1		
14						1			1		1						
15												1					
16																	
17																	

Z PLANE 11 ZL = 89.41 ZU = 90.35

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3										1	1				1		
4			1					1	1	1	1						
5		1	1	1			3	3	2	1	2			1			
6		1		2		3	1		3	2							
7					2	3	2		1	3	6	1					
8			1		3	2	7	3	4	1	3	2			3	1	
9		1	1	2	3		2	1	6	2			1		1		
10			2	2	1		1	1	1								
11			1		2	3			2	3	6		3				
12		1	1		1	2	2			1		2				1	
13					1		2				1						
14		1	1					1		2		1					
15		1	1	1							1						
16			1														
17																	

Z PLANE 12 ZL = 90.35 ZU = 91.29

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2															1		
3										1							
4										1	1						
5			1						1	1		2					
6												2		1		1	
7					1	1	3			2	3	1					
8		1			1	3	1	4	2				1		1		
9				4	1	4	1	2	2	1				1	2	1	
10		1	1		1		2		2	1	1	1					
11		1	4		2				1	1		3	1				
12		1															
13				2		1	1			1	1						
14				1							1						
15			1		1									1			
16																	
17																	

Z PLANE 13 ZL = 91.29 ZU = 92.24

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2												1					
3																	
4																	
5						1		1		2	1				1		
6		1		1	1	1	1			2		1		1			
7			1					1	1	1	1		1				
8					1	1			1	1	1						
9			2	1	1				2		1						
10									1								
11				2	1	1	1			1							
12					1	1	2						1				
13				1		2					1		1				
14						1			1								
15																	
16																	
17																	

Z PLANE 14 ZL = 92.24 ZU = 93.18

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3																	
4																	
5										1				1	1		
6				1						1							
7													1				
8					1						2						
9							2	1		1							
10											1	1					
11								1			1						
12														1			
13											1						
14			1														
15																	
16		1															
17																	

Z PLANE 15 ZL = 93.18 ZU = 94.12

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3																	
4																	
5																	
6																	
7							1	1					1				
8		1															
9								1		1		1		1			
10										1	1						
11									1		1		1				
12											1						
13										1			1			1	
14																	
15																	
16																	
17																	

Z PLANE 16 ZL = 94.12 ZU = 95.06

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
1																	
2																	
3																	
4																	
5											1						
6																	
7																	
8	1										1		1				
9					1												
10												1	1				
11												1					
12										1		1					
13			1					1									
14		1															
15									1								
16																	
17																	

```
Z PLANE 17  ZL =    95.06  ZU =    96.00

      1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17

  1
  2
  3
  4
  5
  6
  7
  8
  9
 10
 11
 12
 13
 14
 15
 16
 17
NX =  5661   NP =  2013
 THE OPTION HSGRAM REQUIRED      20.9006 SECONDS OF CPU TIME.

 ----------------------------------------------------------
```

HSGRAM ENGINEERING DESCRIPTION

HSGRAM computes and displays the histogram of the input data set. It has the capability of computing one, two, or three dimensional histograms. The histogram is generated by dividing the space into a number of rectangular cells. The number, size and range of the cells is specified by the user. The routine then counts the number of data points that fall into each cell.

Define

$$\left.\begin{matrix} X_i \\ X_{i+1} \end{matrix}\right\} \quad \text{boundaries of } i^{th} \text{ cell in X direction}$$

$$\left.\begin{matrix} Y_j \\ Y_{j+1} \end{matrix}\right\} \quad \text{boundaries of } j^{th} \text{ cell in Y direction}$$

$$\left.\begin{matrix} Z_\ell \\ Z_{\ell+1} \end{matrix}\right\} \quad \text{boundaries of } \ell^{th} \text{ cell in Z direction}$$

v_k = for k = 1, 2, 3 the three channels of the data to be used in developing the histogram

then if

$$X_i \leq v_1 < X_{i+1} \quad \text{and}$$

$$Y_j \leq v_2 < Y_{j+1} \quad \text{and}$$

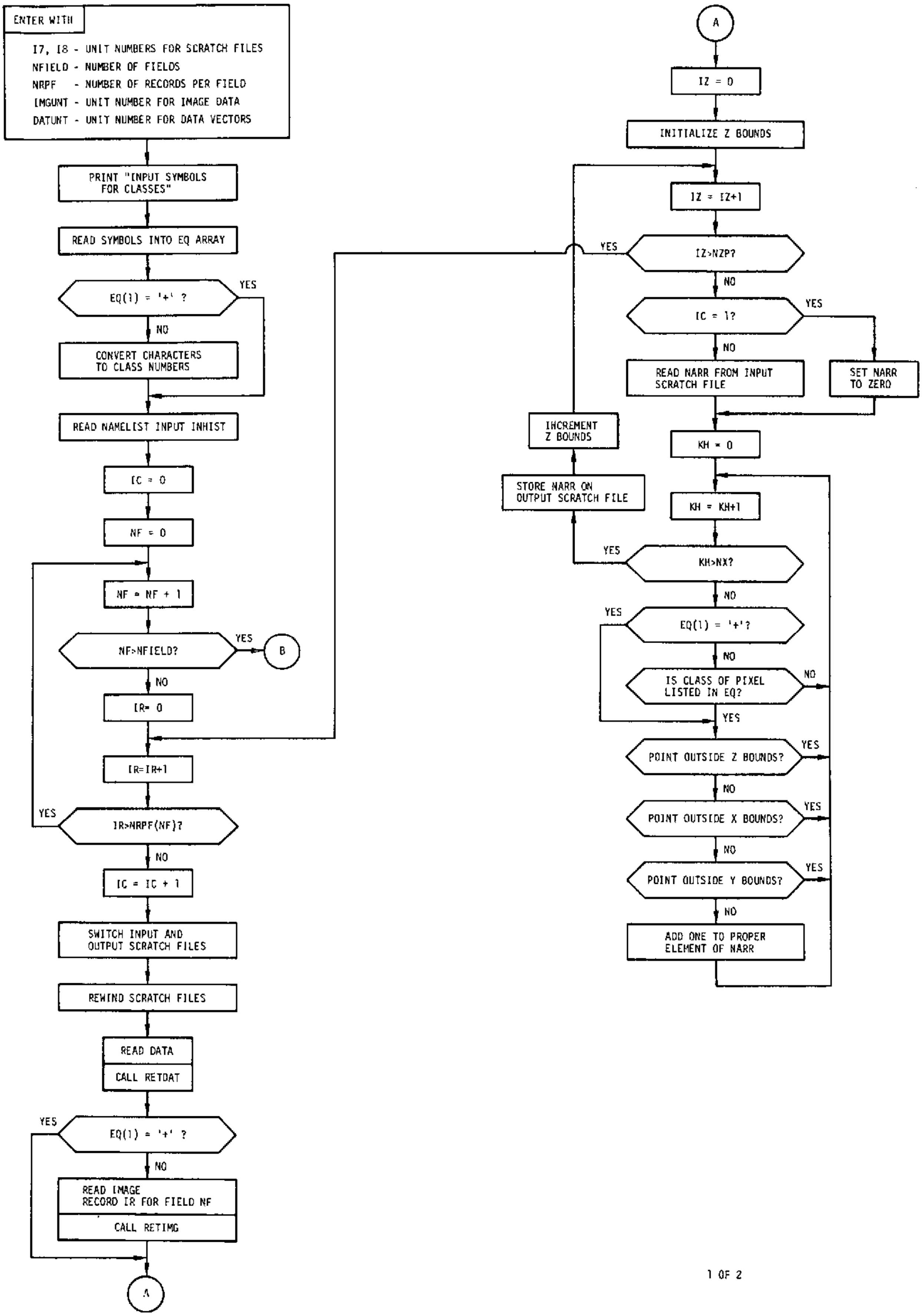
$$Z_\ell \leq v_3 < Z_{\ell+1}$$

then the counter for the $(i, j, \ell)^{th}$ cell is incremented by one.

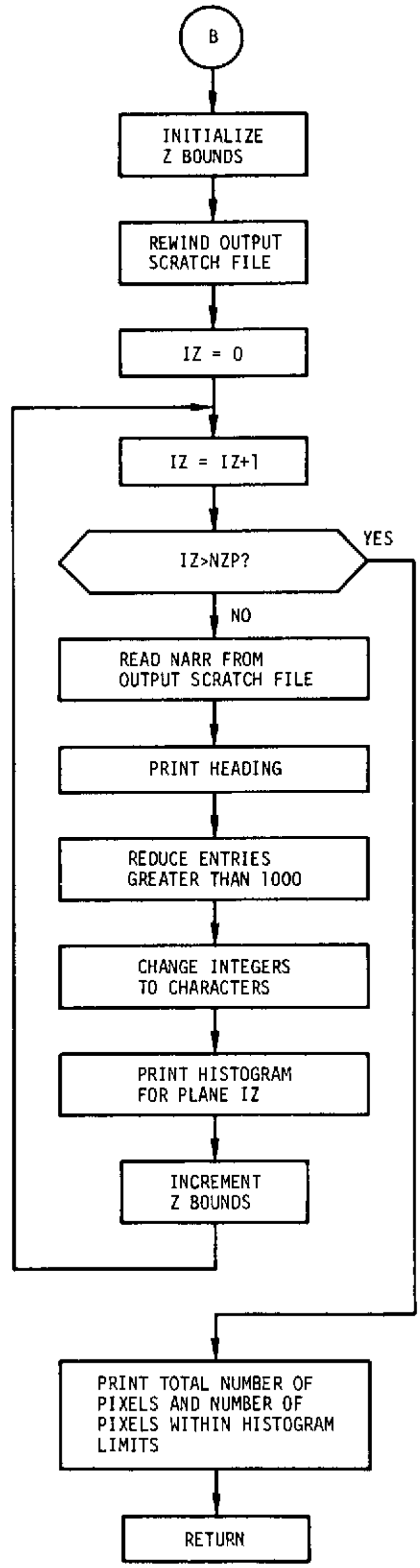
In the computation of the histogram only those sample vectors are used which correspond to the classes of interest. HSGRAM allows the user to input the alphabetic characters which define the classes to be used in computing the histogram.

HSGRAM also has the capability of permuting the data axis so that any channel may be placed on any one of the three output axis X, Y, or Z.

HSGRAM


ENTER WITH
I7, I8 - UNIT NUMBERS FOR SCRATCH FILES
NFIELD - NUMBER OF FIELDS
NRPF - NUMBER OF RECORDS PER FIELD
IMGUNT - UNIT NUMBER FOR IMAGE DATA
DATUNT - UNIT NUMBER FOR DATA VECTORS
PRINT "INPUT SYMBOLS FOR CLASSES"
READ SYMBOLS INTO EQ ARRAY
EQ(1) = '+' ?
YES
NO
CONVERT CHARACTERS TO CLASS NUMBERS
READ NAMELIST INPUT INHIST
IC = 0
NF = 0
NF = NF + 1
NF>NFIELD?
YES
B
NO
IR= 0
IR=IR+1
IR>NRPF(NF)?
YES
NO
IC = IC + 1
SWITCH INPUT AND OUTPUT SCRATCH FILES
REWIND SCRATCH FILES
READ DATA
CALL RETDAT
EQ(1) = '+' ?
YES
NO
READ IMAGE RECORD IR FOR FIELD NF
CALL RETIMG
A
A
IZ = 0
INITIALIZE Z BOUNDS
IZ = IZ+1
IZ>NZP?
YES
NO
IC = 1?
YES
NO
READ NARR FROM INPUT SCRATCH FILE
SET NARR TO ZERO
INCREMENT Z BOUNDS
STORE NARR ON OUTPUT SCRATCH FILE
KH = 0
KH = KH+1
KH>NX?
YES
NO
EQ(1) = '+'?
YES
NO
IS CLASS OF PIXEL LISTED IN EQ?
NO
YES
POINT OUTSIDE Z BOUNDS?
YES
NO
POINT OUTSIDE X BOUNDS?
YES
NO
POINT OUTSIDE Y BOUNDS?
YES
NO
ADD ONE TO PROPER ELEMENT OF NARR

1 OF 2


B
INITIALIZE Z BOUNDS
REWIND OUTPUT SCRATCH FILE
IZ = 0
IZ = IZ+1
IZ>NZP?
YES
NO
READ NARR FROM OUTPUT SCRATCH FILE
PRINT HEADING
REDUCE ENTRIES GREATER THAN 1000
CHANGE INTEGERS TO CHARACTERS
PRINT HISTOGRAM FOR PLANE IZ
INCREMENT Z BOUNDS
PRINT TOTAL NUMBER OF PIXELS AND NUMBER OF PIXELS WITHIN HISTOGRAM LIMITS
RETURN

HSGRAM
2 OF 2

Using the IMAGES Option

The IMAGES option can be used to display classification data as an image composed of characters representing each pixel. The size of the image display is limited to 120 pixels per scan line for batch runs and to about 60 pixels per scan line for interactive runs, depending on the width of the paper. If an attempt is made to print an image with more than 120 pixels per scan line, the IMAGES option will print only the first 120 pixels for each scan line.

The image is labeled with the scan line numbers on the left and the pixel numbers within a scan line at the top. The largest values of these numbers are limited by the print format to six digits for the scan lines and to four digits for the pixels.

There are nine suboptions in IMAGES. Five of these are used to print the images, and the other four are used for information and control.

ALLCLS - prints one image for each field showing all classes. The only input is the suboption name, but the print is affected by the threshold, THRVAL. Any pixel for which the distance value is greater than THRVAL is omitted (printed as a blank). The fields printed are those listed in IFIELD, or all fields if IFIELD has not been input.

ECHCLS - prints an image for each class separately for each field. The only input is the suboption name, but the values of THRVAL and IFIELD affect these images just as they do in ALLCLS. In addition, classes with fewer than MINPIX pixels are omitted.

SUBSET - prints one image for each field showing only a subset of the classes. In addition to the suboption name, a list of characters must be input to define the name of the subset and the names of the classes to be included in the subset. (In this context the name of a class is the same as the symbol used to show it in an image.) If the name of the subset is left blank, then each class in the subset will be printed with its own name. The values of THRVAL and IFIELD affect SUBSET just as they do ALLCLS.

BORDER - prints one image for each field showing only the border pixels for all classes. A border pixel is defined to be different from at least one of four pixels with which it is compared. These are (1) the pixel above, (2) the pixel below, (3) the pixel to the left, and (4) the pixel to the right. If one of these possibilities is undefined because the pixel being tested is on the edge of the image, then it is assumed to match the test pixel. The only input is the suboption name, but the value of IFIELD affects the output just as it does in ALLCLS.

INSIDE - prints one image for each field showing only the inside pixels for all classes. An inside pixel is any pixel that is not a border pixel. It is in the same class as the pixel above, the pixel below, the pixel to the left, and the pixel to the right. The only input to INSIDE is the suboption name, but it is also affected by IFIELD just as in ALLCLS.

THRESH - allows the following control variables to be set with the namelist $INTHRE:

THRVAL - the threshold value used to reject pixels.

MINPIX - the number of pixels below which a class is ignored in ECHCLS.

IFIELD - list of fields to be printed.

The table below shows which print options are affected by each of the namelist inputs. An X indicates that an option is affected.

NAMELIST INPUT

PRINT OPTION	THRVAL	MINPIX	IFIELD
ALLCLS	X		X
ECHCLS	X	X	X
SUBSET	X		X
BORDER			X
INSIDE			X

SYMBOL - allows the symbols printed for each class to be reset. In IMAGES the characters used to print an image are initially set to the letters of the alphabet, but they may be reordered or changed to other symbols in SYMBOL. The input to SYMBOL consists of the option name and a list of characters in the order of the classes that they are to represent. This list is then printed to be verified by the user before control is returned to IMAGES.

STATUS - prints the current values of all input control variables. It prints the namelist variables (THRVAL, MINPIX, and IFIELD), the image file unit number, and the class number, class symbol, and number of pixels for each class. The only input is the name of the suboption.

IMQUIT - returns control to the ASTEP driver.

IMAGES OPTION

SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>IMAGES

                    IMAGES  OPTION
                    ==============

 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
$INTHRE
THRVAL  =       .20000000E+04
MINPIX  =                 +1
IFIELD  =                 +0,             +0,             +0,             +0,
                          +0,             +0,             +0,             +0,
                          +0,             +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>SYMBOL
 TYPE THE STRING OF  4 IMAGE SYMBOLS DESIRED.
>CPWS
 CLASS SYMBOL   ABCD
 IMAGE SYMBOL   CPWS
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>STATUS
IMUNIT   THRVAL    MINPIX    IFIELD
  12     2000.0       1        0  0  0  0  0  0  0  0  0  0
       CLASS     CLASS     IMAGE      NUMBER
       NUMBER    SYMBOL    SYMBOL    OF PIXELS
         1         A         C          465
         2         B         P          456
         3         C         W          358
         4         D         S          402

 SELECT IMAGES OPTION FROM

    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>ALLCLS
```

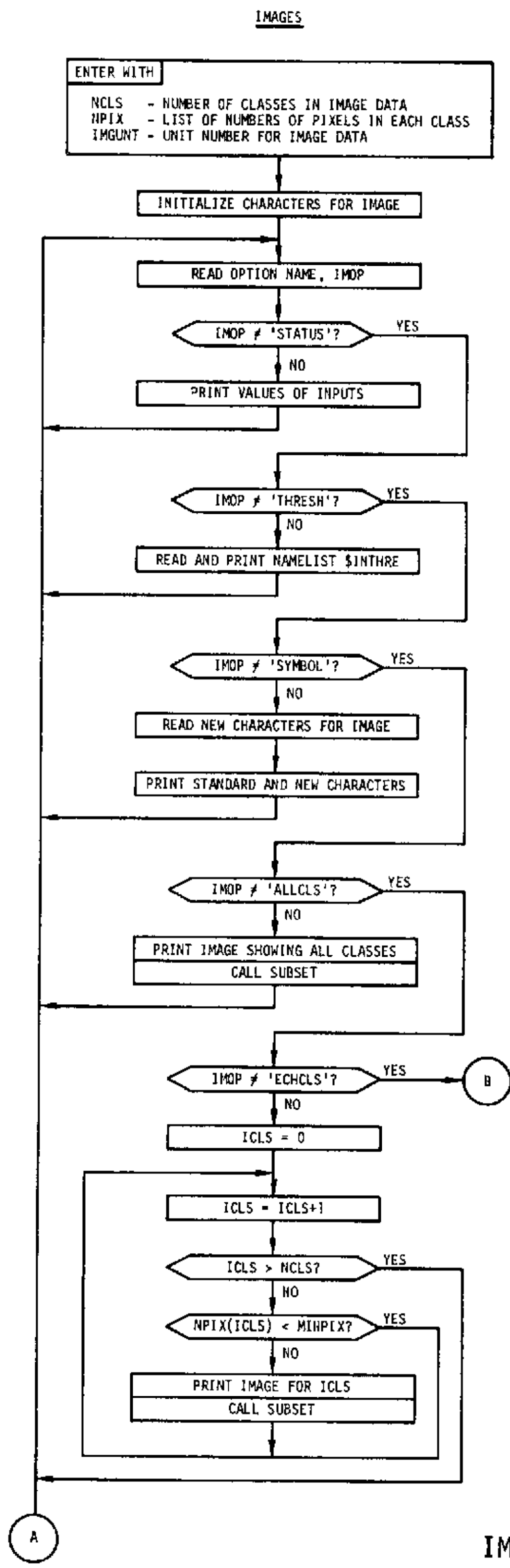
```
IMAGE FOR FIELD  1

          11111222223333344444555556666677777888889
          02468024680246802468024680246802468024680

   600    CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   602    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   604    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   606    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   608    CCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPPP
   610    CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
   612    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   614    CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
   616    CCCCCCCCCCCCCCCCCWCPPPPPPPPPPPPPPPPPPPPPP
   618    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   620    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   622    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   624    CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   626    CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   628    CCCCCCCCCCCCCCCCCCSPPPPPPPCPPPPPPPPPPPPPC
   630    CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPC
   632    CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPC
   634    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPCP
   636    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPCP
   638    CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPCCCC
   640    CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPCCPPPCWW
   642    CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPSWSSCCCWSSW
   644    CCCCCCCCCCCCCCCCCCCPPPPPPPPCCCCPPPPCCSCCW
   646    WWWWWWWWWWWWWWWWWWWCSSSSSSSSSSSSSSSSSSSSS
   648    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS
   650    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS
   652    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS
   654    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS
   656    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS
   658    WWWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSS
   660    WWWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSS
   662    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS
   664    WWWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSS
   666    WWWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSS
   668    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS
   670    WWWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSS
   672    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS
   674    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS
   676    WWWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSS
   678    WWWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSS
   680    WWWWWWWWWWWWWWWWWWWSSSSSSSSSSSSSSSSSSSSSS


SELECT IMAGES OPTION FROM
   STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
   ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>BORDER
```

IMAGE FOR ALL CLASSES FOR FIELD 1 WITH ONLY BORDER PIXELS PRINTED

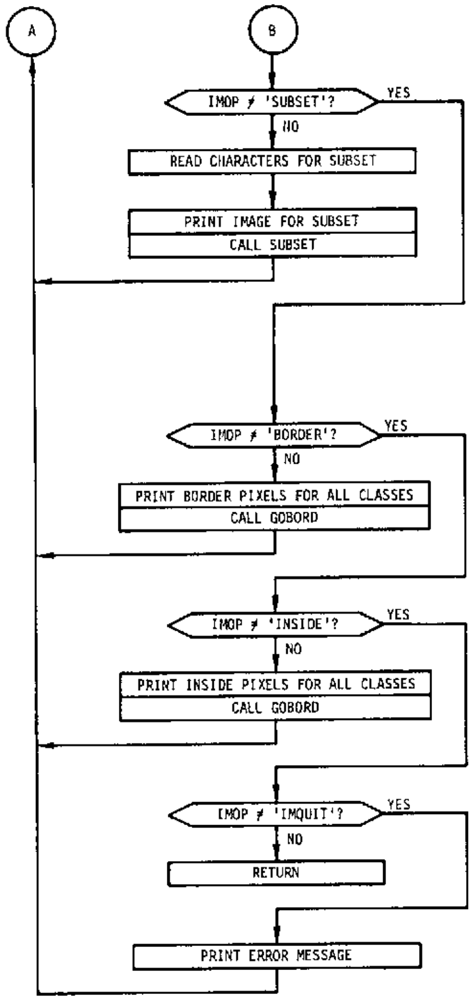
```
      11111222223333344444555556666677777888889
      02468024680246802468024680246802468024680

600                    CSP
602                     CP
604                     CP
606                    CCP
608                   CSP
610                   CSCP
612                    CCP
614                   CSCP
616                   CWCP
618                    CCP
620                     CP
622                     CP
624                    CSP
626                    CSP      P             P
628                    CSP     PCP           PC
630                    CSP      P            PC
632                    CSCP                  PC
634                     CP                  PCP
636                     CP                 PPCP
638                    CSCP            PP PCCCC
640                     CP            PPPCCPPPCWW
642                    CSP         PPPSWSSCCCWSSW
644   CCCCCCCCCCCCCCCCCCCPPPPPPPPCCCCPPPPCCSCCW
646   WWWWWWWWWWWWWWWWWWWCSSSSSSSSSSSSSSSSS SSS
648                     WS
650                     WS
652                     WS
654                     WS
656                     WS
658                      WS
660                      WS
662                     WS
664                      WS
666                      WS
668                     WS
670                      WS
672                     WS
674                     WS
676                      WS
678                      WS
680                     WS
```

```
SELECT IMAGES OPTION FROM
   STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
   ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>INSIDE
```

```
IMAGE FOR ALL CLASSES FOR FIELD  1 WITH ONLY INSIDE
PIXELS PRINTED

         11111222223333344444555556666677777888889
         02468024680246802468024680246802468024680

    600  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
    602  CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
    604  CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
    606  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
    608  CCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPPP
    610  CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
    612  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
    614  CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
    616  CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
    618  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
    620  CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
    622  CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
    624  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
    626  CCCCCCCCCCCCCCCCC   PPPPPP PPPPPPPPPPPPP
    628  CCCCCCCCCCCCCCCCC   PPPPP   PPPPPPPPPPP
    630  CCCCCCCCCCCCCCCCC   PPPPPP PPPPPPPPPPPP
    632  CCCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPP
    634  CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPP
    636  CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPP
    638  CCCCCCCCCCCCCCCCC    PPPPPPPPPPPP  P
    640  CCCCCCCCCCCCCCCCCC  PPPPPPPPPP
    642  CCCCCCCCCCCCCCCCC   PPPPPPP
    644
    646                                       S
    648  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS
    650  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS
    652  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS
    654  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS
    656  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS
    658  WWWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSS
    660  WWWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSS
    662  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS
    664  WWWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSS
    666  WWWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSS
    668  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS
    670  WWWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSS
    672  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS
    674  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS
    676  WWWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSS
    678  WWWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSS
    680  WWWWWWWWWWWWWWWWWW  SSSSSSSSSSSSSSSSSSSSS

 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>IMQUIT
 THE OPTION IMAGES REQUIRED       2.9966 SECONDS OF CPU TIME.

 -----------------------------------------------------------
```

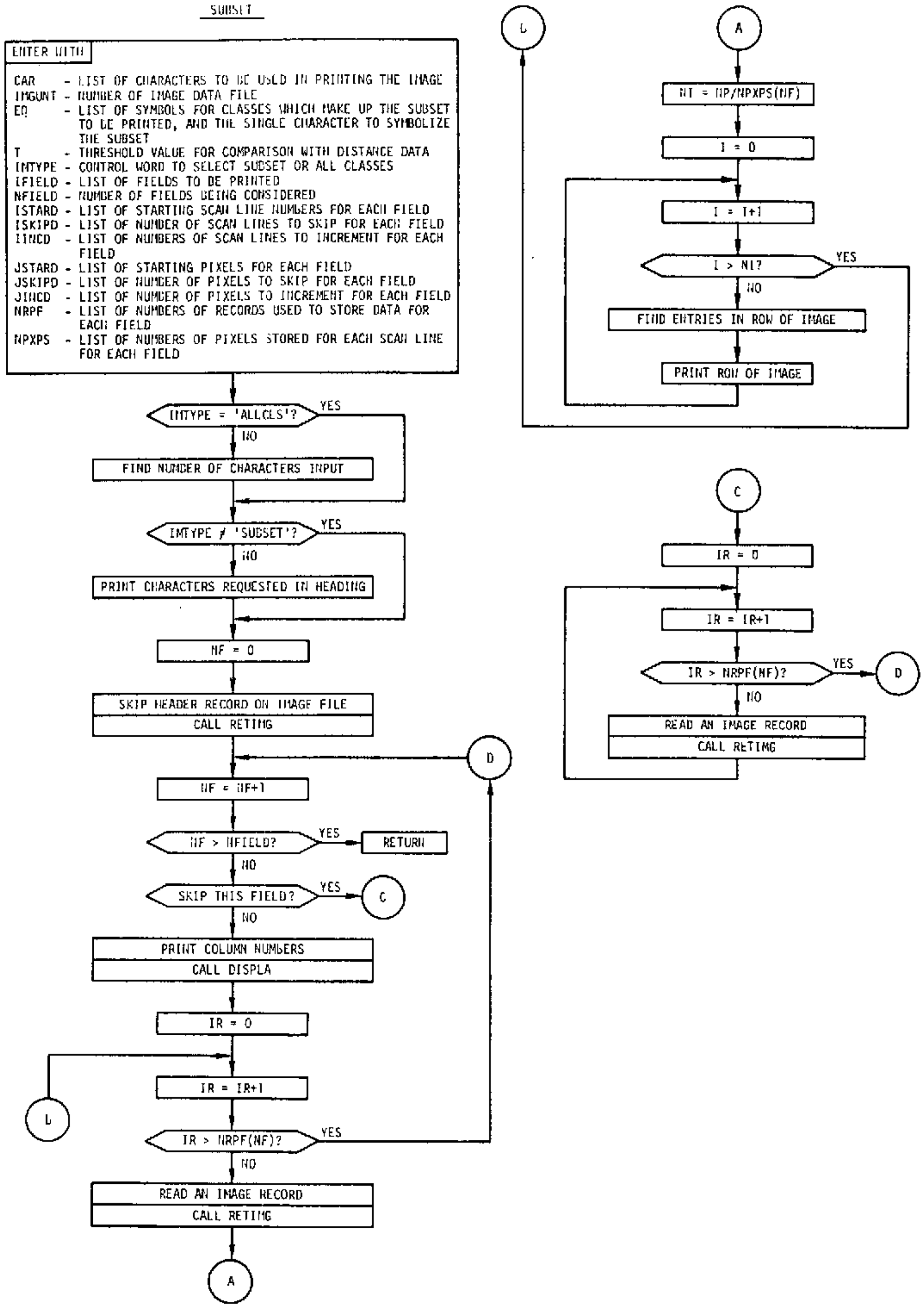

IMAGES
ENTER WITH
NCLS - NUMBER OF CLASSES IN IMAGE DATA
NPIX - LIST OF NUMBERS OF PIXELS IN EACH CLASS
IMGUNT - UNIT NUMBER FOR IMAGE DATA
INITIALIZE CHARACTERS FOR IMAGE
READ OPTION NAME, IMOP
IMOP ≠ 'STATUS'?
YES
NO
PRINT VALUES OF INPUTS
IMOP ≠ 'THRESH'?
YES
NO
READ AND PRINT NAMELIST $INTHRE
IMOP ≠ 'SYMBOL'?
YES
NO
READ NEW CHARACTERS FOR IMAGE
PRINT STANDARD AND NEW CHARACTERS
IMOP ≠ 'ALLCLS'?
YES
NO
PRINT IMAGE SHOWING ALL CLASSES
CALL SUBSET
IMOP ≠ 'ECHCLS'?
YES
B
NO
ICLS = 0
ICLS = ICLS+1
ICLS > NCLS?
YES
NO
NPIX(ICLS) < MINPIX?
YES
NO
PRINT IMAGE FOR ICLS
CALL SUBSET
A

IMAGES 1 of 2

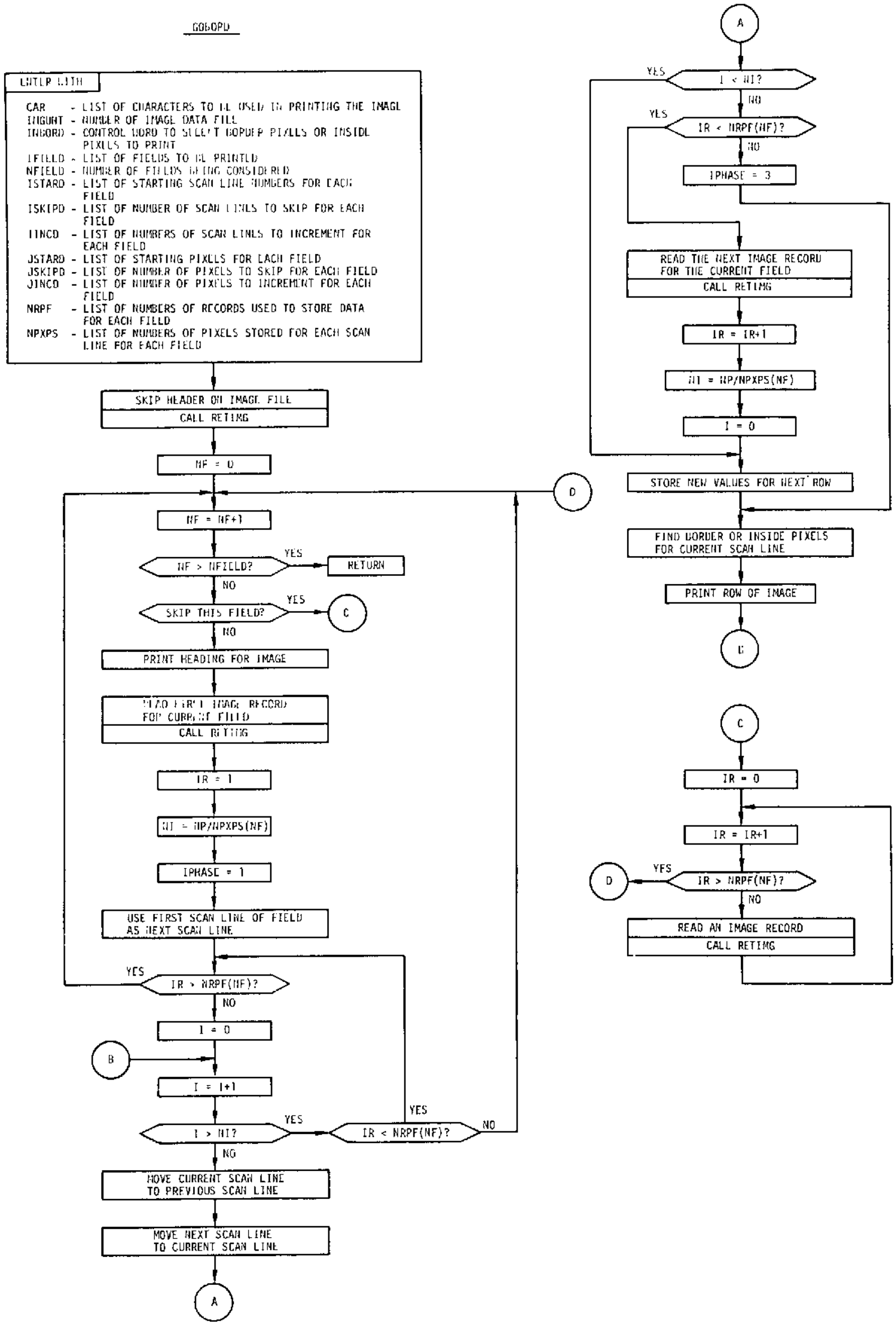

A
B
IMOP ≠ 'SUBSET'?
YES
NO
READ CHARACTERS FOR SUBSET
PRINT IMAGE FOR SUBSET
CALL SUBSET
IMOP ≠ 'BORDER'?
YES
NO
PRINT BORDER PIXELS FOR ALL CLASSES
CALL GOBORD
IMOP ≠ 'INSIDE'?
YES
NO
PRINT INSIDE PIXELS FOR ALL CLASSES
CALL GOBORD
IMOP ≠ 'IMQUIT'?
YES
NO
RETURN
PRINT ERROR MESSAGE

IMAGES 2 of 2

SUBSET

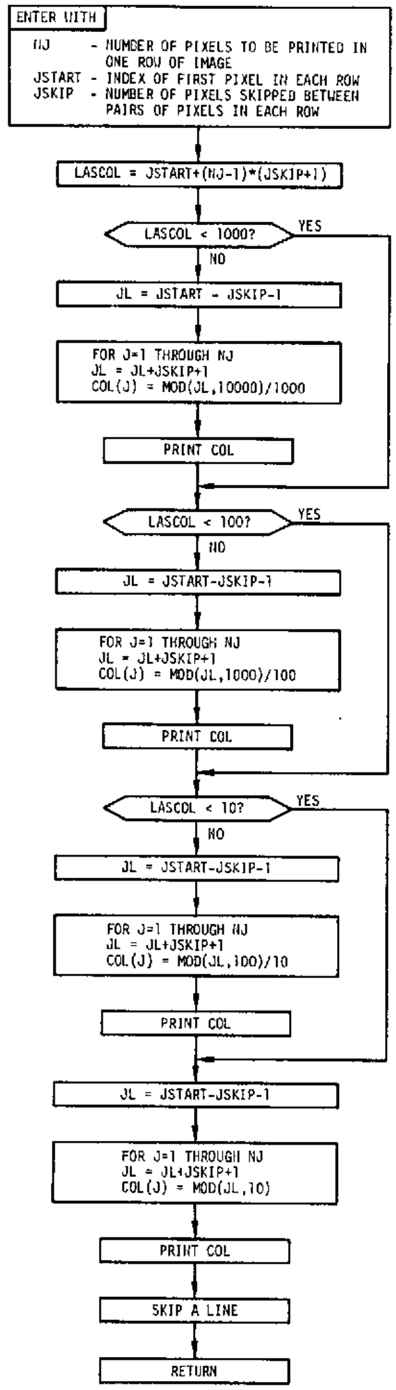

ENTER WITH
CAR - LIST OF CHARACTERS TO BE USED IN PRINTING THE IMAGE
IMGUNT - NUMBER OF IMAGE DATA FILE
EQ - LIST OF SYMBOLS FOR CLASSES WHICH MAKE UP THE SUBSET TO BE PRINTED, AND THE SINGLE CHARACTER TO SYMBOLIZE THE SUBSET
T - THRESHOLD VALUE FOR COMPARISON WITH DISTANCE DATA
IMTYPE - CONTROL WORD TO SELECT SUBSET OR ALL CLASSES
IFIELD - LIST OF FIELDS TO BE PRINTED
NFIELD - NUMBER OF FIELDS BEING CONSIDERED
ISTARD - LIST OF STARTING SCAN LINE NUMBERS FOR EACH FIELD
ISKIPD - LIST OF NUMBER OF SCAN LINES TO SKIP FOR EACH FIELD
IINCD - LIST OF NUMBERS OF SCAN LINES TO INCREMENT FOR EACH FIELD
JSTARD - LIST OF STARTING PIXELS FOR EACH FIELD
JSKIPD - LIST OF NUMBER OF PIXELS TO SKIP FOR EACH FIELD
JINCD - LIST OF NUMBER OF PIXELS TO INCREMENT FOR EACH FIELD
NRPF - LIST OF NUMBERS OF RECORDS USED TO STORE DATA FOR EACH FIELD
NPXPS - LIST OF NUMBERS OF PIXELS STORED FOR EACH SCAN LINE FOR EACH FIELD
IMTYPE = 'ALLCLS'?
YES
NO
FIND NUMBER OF CHARACTERS INPUT
IMTYPE ≠ 'SUBSET'?
YES
NO
PRINT CHARACTERS REQUESTED IN HEADING
NF = 0
SKIP HEADER RECORD ON IMAGE FILE
CALL RETIMG
D
NF = NF+1
NF > NFIELD?
YES
RETURN
NO
SKIP THIS FIELD?
YES
C
NO
PRINT COLUMN NUMBERS
CALL DISPLA
IR = 0
B
IR = IR+1
IR > NRPF(NF)?
YES
NO
READ AN IMAGE RECORD
CALL RETIMG
A
B
A
NI = NP/NPXPS(NF)
I = 0
I = I+1
I > NI?
YES
NO
FIND ENTRIES IN ROW OF IMAGE
PRINT ROW OF IMAGE
C
IR = 0
IR = IR+1
IR > NRPF(NF)?
YES
D
NO
READ AN IMAGE RECORD
CALL RETIMG

GOBOPD



ENTER WITH
CAR - LIST OF CHARACTERS TO BE USED IN PRINTING THE IMAGE
INGUNT - NUMBER OF IMAGE DATA FILE
INBORD - CONTROL WORD TO SELECT BORDER PIXELS OR INSIDE PIXELS TO PRINT
IFIELD - LIST OF FIELDS TO BE PRINTED
NFIELD - NUMBER OF FIELDS BEING CONSIDERED
ISTARD - LIST OF STARTING SCAN LINE NUMBERS FOR EACH FIELD
ISKIPD - LIST OF NUMBER OF SCAN LINES TO SKIP FOR EACH FIELD
IINCD - LIST OF NUMBERS OF SCAN LINES TO INCREMENT FOR EACH FIELD
JSTARD - LIST OF STARTING PIXELS FOR EACH FIELD
JSKIPD - LIST OF NUMBER OF PIXELS TO SKIP FOR EACH FIELD
JINCD - LIST OF NUMBER OF PIXELS TO INCREMENT FOR EACH FIELD
NRPF - LIST OF NUMBERS OF RECORDS USED TO STORE DATA FOR EACH FIELD
NPXPS - LIST OF NUMBERS OF PIXELS STORED FOR EACH SCAN LINE FOR EACH FIELD
SKIP HEADER ON IMAGE FILE
CALL RETIMG
NF = 0
NF = NF+1
NF > NFIELD?
YES
RETURN
NO
SKIP THIS FIELD?
YES
C
NO
PRINT HEADING FOR IMAGE
READ FIRST IMAGE RECORD FOR CURRENT FIELD
CALL RETIMG
IR = 1
NI = NP/NPXPS(NF)
IPHASE = 1
USE FIRST SCAN LINE OF FIELD AS NEXT SCAN LINE
YES
IR > NRPF(NF)?
NO
I = 0
B
I = I+1
I > NI?
YES
IR < NRPF(NF)?
YES
NO
D
NO
MOVE CURRENT SCAN LINE TO PREVIOUS SCAN LINE
MOVE NEXT SCAN LINE TO CURRENT SCAN LINE
A
A
YES
I < NI?
NO
YES
IR < NRPF(NF)?
NO
IPHASE = 3
READ THE NEXT IMAGE RECORD FOR THE CURRENT FIELD
CALL RETIMG
IR = IR+1
NI = NP/NPXPS(NF)
I = 0
STORE NEW VALUES FOR NEXT ROW
FIND BORDER OR INSIDE PIXELS FOR CURRENT SCAN LINE
PRINT ROW OF IMAGE
B
C
IR = 0
IR = IR+1
D
YES
IR > NRPF(NF)?
NO
READ AN IMAGE RECORD
CALL RETIMG

DISPLA


ENTER WITH
NJ - NUMBER OF PIXELS TO BE PRINTED IN ONE ROW OF IMAGE
JSTART - INDEX OF FIRST PIXEL IN EACH ROW
JSKIP - NUMBER OF PIXELS SKIPPED BETWEEN PAIRS OF PIXELS IN EACH ROW
LASCOL = JSTART+(NJ-1)*(JSKIP+1)
LASCOL < 1000?
YES
NO
JL = JSTART - JSKIP-1
FOR J=1 THROUGH NJ
JL = JL+JSKIP+1
COL(J) = MOD(JL,10000)/1000
PRINT COL
LASCOL < 100?
YES
NO
JL = JSTART-JSKIP-1
FOR J=1 THROUGH NJ
JL = JL+JSKIP+1
COL(J) = MOD(JL,1000)/100
PRINT COL
LASCOL < 10?
YES
NO
JL = JSTART-JSKIP-1
FOR J=1 THROUGH NJ
JL = JL+JSKIP+1
COL(J) = MOD(JL,100)/10
PRINT COL
JL = JSTART-JSKIP-1
FOR J=1 THROUGH NJ
JL = JL+JSKIP+1
COL(J) = MOD(JL,10)
PRINT COL
SKIP A LINE
RETURN

Using the INTHDR Option

The INTHDR requires no user inputs. Its purpose is to allow a user to process a data tape created by DATDEF which has been saved. INTHDR reads the header record on the tape and initializes the appropriate variables in the program. It also displays the user inputs that were input to DATDEF when the data was originally extracted from the raw observation data tape.

INTHDR OPTION

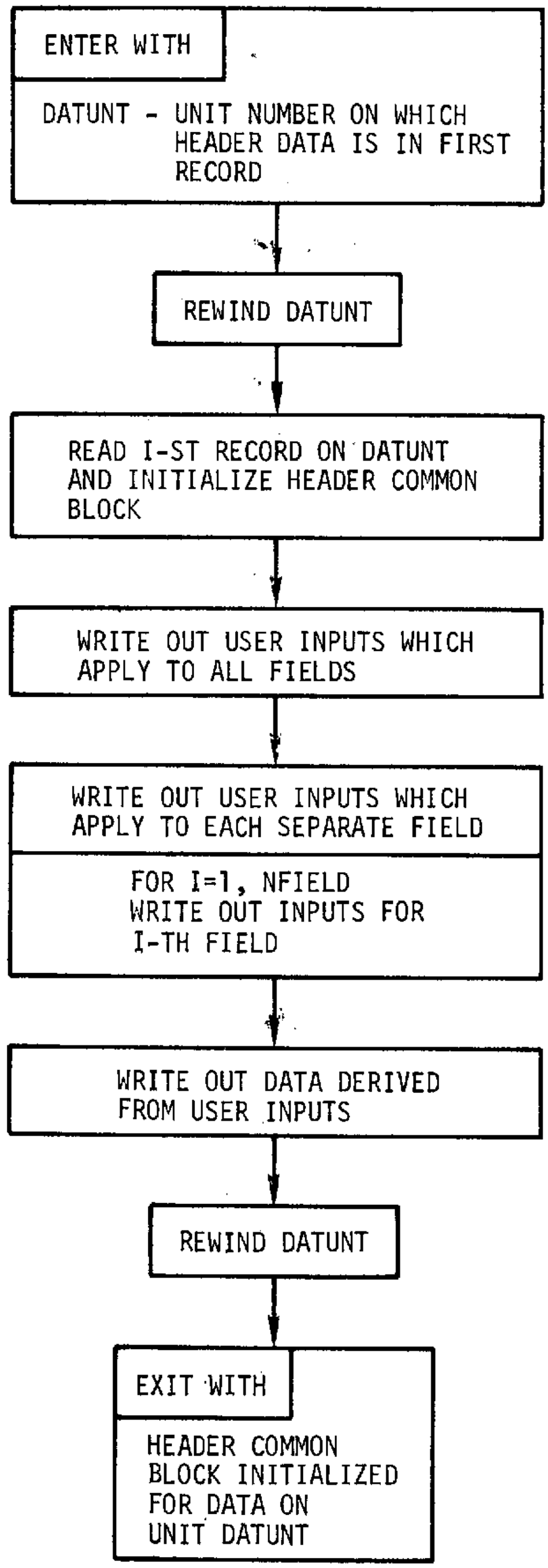
SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>INTHDR

                        INTHDR  OPTION
                        ==============

***          UNPACKED DATA TAPE ON UNIT  4              ***
          1 FIELDS  4 CHANNELS  1  6  9 12
ITPFMT 1 ITPNO 1 IBUF1  6669 IBUF2  8335 NBUFSZ10000 NRT  >
             NO TRANSFORMATION

FIELD DATA
FIELD  ISTART ISKIP  IINC   JSTART JSKIP  JINC   NPXPS  NRPF
  1     600     1     80     10      1     80     41      2

 THE OPTION INTHDR REQUIRED        .0388 SECONDS OF CPU TIME.
 ----------------------------------------------------------------
```

INTHDR ENGINEERING DESCRIPTION

INTHDR does not require an engineering description - see flow chart.

INTHDR



INTHDR 1 of 1

Using the ITRCLU Option

Upon entering ITRCLU the user must define values for the parameters

T1	=	threshold for cluster splitting, units of the standard deviation
T2	=	threshold for cluster combining, if ISODAT = 0 T2 has units of the data and if ISODAT $\neq$ 0 T2 has no units
NMIN	=	small cluster elimination threshold (number of points)
NVMMAX	=	maximum number of clusters to be allowed ($\leq$ 20)
SEP	=	the number of sigmas, in cluster splitting, to separate the two new clusters from the mean of the original cluster along the pertinent channel
ISODAT	=	flag defining distance measure in cluster combining algorithm, if ISODAT = 0 use unweighted euclidean distance and if ISODAT $\neq$ 0 use weighted distance measure
IDISF	=	flag defining distance measure used in vector assignments to clusters, if IDISF = 1, use euclidean measure and if IDISF = 2, use L1 measure
S	=	threshold used in grouping the data into strips, units of the data - if S < 0. the strip forming logic is bypassed
P	=	percentage threshold (0. < P < 1.) for initial cluster splitting prior to using input split combine sequence - if P < 0. initial cluster splitting logic is bypassed
IP	=	print control flag, if IP = 0 no print, if IP $\neq$ 0 print merger and split messages, if IP = 2 print cluster means, variances, and weights at the end of each iteration

The default values for these parameters are

T1	=	4.5
T2	=	3.2
NMIN	=	30
NVMMAX	=	20
SEP	=	1.0
ISODAT	=	1
IDISF	=	2
S	=	1.0
P	=	.5
IP	=	0

The user must then enter the split(S) combine(C) sequence. This sequence is controlled by the ordering of the characters S and C. A blank card results in the default sequence

SSSSSCSCSCCC

The user must then define the cluster mean and weights initialization procedure. The options are

ZERO - all values 0, this forces the algorithm to be self starting

OLD - use means and weights from last previous clustering (either ADPCLU or ITRCLU). For example, if one exits ITRCLU and calls IMAGES, then reenters ITRCLU the previous means and weights remain available. One may continue to sequence through the clustering options and image display with the previous results available to restart via OLD.

NEW - allows user to input starting values or to change any of the current parameters. The parameters are

NVM = number of clusters

NVG = weight for each cluster, may be ignored for ITRCLU

VM = cluster means, one-dimensional array of number of channels x NVM values representing a matrix of mean vectors input by columns.

Upon completion of the clustering a run summary is displayed. This output is a description of the clusters formed. It lists the cluster number, assigns a symbol to those points in the cluster, describes the size, gives the statistics (mean and sigma) of the distances of the points to the cluster center, and gives the L1 distances between the vector used as a center to form the cluster and the mean vector of the resulting cluster.

The user then must select one of the suboptions MEANS, SIGMAS, ANGDIS, or QUIT. MEANS, SIGMAS, and ANGDIS are for output only and require no input parameters. QUIT returns control to ASTEP.

The MEANS suboption displays an m x n array where m is the number of data channels and n is the number of clusters. The columns are the mean vectors for the clusters formed.

The SIGMAS suboption displays an m x n array of the individual sigmas for each channel and cluster. The columns are the channel sigmas for the clusters formed.

The output of the ANGDIS suboption is an n x n array. The diagonal of this array will be zero. Angles (in degrees) between a pair of mean vectors are given above the diagonal. The distances (given in channel units) between the vectors are given below the diagonal. Depending upon the value of the distance flag IDISF (=1 or 2), a euclidean or L1 distance measure is used.

ITRCLU OPTION
SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>ITRCLU

              ITRCLU  OPTION
              ==============

$INITRC   T1,T2,NMIN,NVMMAX,SEP,ISODAT,IDISF,S,P,IP
$INITRC
T1      =      .80000000E+01
T2      =      .55000000E+01
NMIN    =                 +5
NVMMAX  =                +20
SEP     =      .15000000E+01
ISODAT  =                 +1
IDISF   =                 +1
S       =     -.10000000E+01
P       =     -.10000000E+01
IP      =                 +2

$END
TYPE YES IF INPUTS OK
>YES
ENTER SPLIT/COMBINE (SC) SEQUENCE
>SSSSSCSCS
CHOOSE VALUES FOR INITIALIZATION FROM
ZERO  OLD  NEW
>ZERO
    ***** ITERATION   1-S *****
               NUMPTS      1 BY  1
        1

 1 1681
               MEANS       4 BY  1
          1

 1     82.052
 2     90.765
 3     71.291
 4     87.997

               SIGMAS      4 BY  1
          1

 1      5.823
 2      9.020
 3     17.322
 4     14.646
```

IC DIS 1 BY 1
ALL ZEROES.

ITERATION 1, CLUSTER 1, POSITION 3, SPLIT, NVM = 2

***** ITERATION 2-5 *****

NUMPTS 1 BY 2

	1	2
1	708	973

MEANS 4 BY 2

	1	2
1	85.064	79.860
2	98.819	84.905
3	87.650	59.388
4	79.394	94.256

SIGMAS 4 BY 2

	1	2
1	4.666	5.599
2	8.270	3.041
3	14.971	4.337
4	5.876	15.890

IC DIS 2 BY 2

	1	2
1	.000	4.837
2	4.837	.000

ITERATION 2, CLUSTER 1, POSITION 3, SPLIT, NVM = 3
ITERATION 2, CLUSTER 2, POSITION 4, SPLIT, NVM = 4

***** ITERATION 3-5 *****

NUMPTS 1 BY 4

	1	2	3	4
1	324	431	601	325

MEANS 4 BY 4

	1	2	3	4
1	81.287	74.800	87.061	83.166
2	103.728	82.847	92.554	85.037
3	101.932	56.039	71.376	60.815
4	78.701	110.524	81.611	79.197

SIGMAS 4 BY 4

	1	2	3	4
1	3.099	2.155	3.703	4.043
2	10.028	1.427	3.610	2.108
3	10.101	1.895	6.344	4.041
4	7.307	7.094	4.760	5.947

IC DIS 4 BY 4

	1	2	3	4
1	.000	12.897	4.601	7.631
2	12.897	.000	9.025	5.989
3	4.601	9.025	.000	3.605
4	7.631	5.989	3.605	.000

ITERATION 3, CLUSTER 1, POSITION 3, SPLIT, NVM = 5

***** ITERATION 4-5 *****

NUMPTS 1 BY 5

	1	2	3	4	5
1	142	441	436	457	205

MEANS 4 BY 5

	1	2	3	4	5
1	82.803	74.780	88.376	83.641	80.181
2	111.049	82.828	94.280	85.794	97.400
3	110.430	56.054	74.057	60.947	94.137
4	83.972	110.152	80.562	81.136	74.229

SIGMAS 4 BY 5

	1	2	3	4	5
1	3.456	2.166	2.721	3.657	2.411
2	10.029	1.448	2.106	2.090	5.301
3	9.117	1.936	2.848	2.455	5.093
4	5.429	7.429	3.814	5.671	6.137

IC DIS 5 BY 5

	1	2	3	4	5
1	.000	15.747	8.254	11.838	3.591
2	15.747	.000	12.809	6.152	14.445
3	8.254	12.809	.000	6.573	6.373
4	11.838	6.152	6.573	.000	10.147
5	3.591	14.445	6.373	10.147	.000

ITERATION 4, CLUSTER 1, POSITION 2, SPLIT, NVM = 6

***** ITERATION 5-5 *****

NUMPTS 1 BY 6

	1	2	3	4	5	6
1	60	438	430	473	215	65

MEANS 4 BY 6

	1	2	3	4	5	6
1	85.350	74.737	88.453	83.609	79.921	81.662
2	120.767	82.826	94.484	85.871	97.395	105.662
3	116.400	56.037	74.393	61.110	95.298	106.615
4	88.517	110.251	80.444	81.345	73.874	82.631

SIGMAS 4 BY 6

	1	2	3	4	5	6
1	3.344	2.090	2.693	3.664	1.854	2.406
2	8.108	1.439	1.979	2.163	4.646	3.114
3	11.180	1.906	2.785	2.679	5.070	3.272
4	3.895	7.355	3.638	5.815	5.364	3.672

IC DIS 6 BY 6

	1	2	3	4	5	6
1	.000	18.083	10.266	13.191	6.113	3.971
2	18.083	.000	13.335	6.151	15.223	23.755
3	10.266	13.335	.000	6.587	6.976	11.902
4	13.191	6.151	6.587	.000	10.151	17.170
5	6.113	15.223	6.976	10.151	.000	4.125
6	3.971	23.755	11.902	17.170	4.125	.000

ITERATION 5, CLUSTER 1, POSITION 3, SPLIT, NVM = 7

***** ITERATION 6-C *****

NUMPTS 1 BY 7

	1	2	3	4	5	6	7
1	23	438	427	476	190	109	18

MEANS 4 BY 7

	1	2	3	4	5	6
1	86.435	74.737	88.513	83.586	79.842	81.725
2	128.304	82.826	94.503	85.908	96.521	106.890
3	127.696	56.037	74.429	61.162	94.558	106.495
4	91.261	110.251	80.382	81.395	73.153	82.385

	7
1	85.889
2	117.389
3	105.111
4	88.167

SIGMAS 4 BY 7

	1	2	3	4	5	6
1	3.422	2.090	2.606	3.664	1.816	2.589
2	7.760	1.439	1.964	2.211	4.013	4.806
3	8.215	1.906	2.761	2.750	4.840	4.428
4	2.832	7.355	3.525	5.857	5.230	3.894

	7
1	3.008
2	4.408
3	7.395
4	3.015

IC DIS 7 BY 7

	1	2	3	4	5	6
1	.000	23.444	14.573	17.527	9.444	5.856
2	23.444	.000	13.482	6.128	15.362	20.532
3	14.573	13.482	.000	6.541	7.040	10.367
4	17.527	6.128	6.541	.000	10.041	14.513
5	9.444	15.362	7.040	10.041	.000	4.143
6	5.856	20.532	10.367	14.513	4.143	.000
7	3.610	20.025	10.639	14.133	6.980	3.215

	7
1	3.610
2	20.025
3	10.639
4	14.133
5	6.980
6	3.215
7	.000

ITERATION 6 CLUSTERS 1 AND 7 MERGED, NVM = 6
ITERATION 6 CLUSTERS 5 AND 6 MERGED, NVM = 5

***** ITERATION 7-5 *****

NUMPTS 1 BY 5

	1	2	3	4	5
1	62	438	429	476	276

MEANS 4 BY 5

	1	2	3	4	5
1	85.597	74.737	88.471	83.586	80.239
2	120.161	82.826	94.482	85.908	99.362
3	116.871	56.037	74.471	61.162	97.790
4	88.935	110.251	80.319	81.395	75.786

SIGMAS 4 BY 5

	1	2	3	4	5
1	3.292	2.090	2.673	3.664	1.945
2	8.462	1.439	1.984	2.211	5.609
3	10.412	1.906	2.825	2.750	6.365
4	3.501	7.355	3.634	5.857	5.984

IC DIS 5 BY 5

	1	2	3	4	5
1	.000	18.323	10.351	13.199	5.229
2	18.323	.000	13.340	6.128	14.561
3	10.351	13.340	.000	6.484	6.808
4	13.199	6.128	6.484	.000	9.681
5	5.229	14.561	6.808	9.681	.000

ITERATION 7, CLUSTER 1, POSITION 3, SPLIT, NVM = 6

***** ITERATION 8=C *****

NUMPTS 1 BY 6

	1	2	3	4	5	6
1	25	438	428	476	264	50

MEANS 4 BY 6

	1	2	3	4	5	6
1	85.840	74.737	88.488	83.586	80.106	84.800
2	127.240	82.826	94.491	85.908	98.780	114.120
3	126.960	56.037	74.446	61.162	97.439	108.460
4	90.560	110.251	80.348	81.395	75.303	87.100

SIGMAS 4 BY 6

	1	2	3	4	5	6
1	3.965	2.090	2.652	3.664	1.851	2.763
2	8.298	1.439	1.979	2.211	5.126	4.378
3	8.279	1.906	2.782	2.750	6.293	5.407
4	3.664	7.355	3.589	5.857	5.729	3.202

IC DIS 6 BY 6

	1	2	3	4	5	6
1	.000	22.655	13.914	16.957	7.165	3.675
2	22.655	.000	13.404	6.128	14.624	21.504
3	13.914	13.404	.000	6.513	6.896	11.278
4	16.957	6.128	6.513	.000	9.673	15.315
5	7.165	14.624	6.896	9.673	.000	5.094
6	3.675	21.504	11.278	15.315	5.094	.000

ITERATION 8 CLUSTERS 1 AND 6 MERGED, NVM = 5

***** ITERATION 9=S *****

NUMPTS 1 BY 5

	1	2	3	4	5
1	80	438	428	476	259

MEANS 4 BY 5

	1	2	3	4	5
1	84.800	74.737	88.488	83.586	80.116
2	117.663	82.826	94.491	85.908	98.656
3	114.775	56.037	74.446	61.162	97.062
4	87.887	110.251	80.348	81.395	75.166

SIGMAS 4 BY 5

	1	2	3	4	5
1	3.328	2.090	2.652	3.664	1.907
2	8.904	1.439	1.979	2.211	5.106
3	10.198	1.906	2.782	2.750	5.925
4	3.910	7.355	3.589	5.857	5.690

IC DIS 5 BY 5

	1	2	3	4	5
1	.000	17.441	9.663	12.477	4.886
2	17.441	.000	13.404	6.128	14.826
3	9.663	13.404	.000	6.513	6.922
4	12.477	6.128	6.513	.000	9.817
5	4.886	14.826	6.922	9.817	.000

ITERATION 9, CLUSTER 1, POSITION 3, SPLIT, NVM = 6

***** ITERATION 10 *****

NUMPTS 1 BY 6

	1	2	3	4	5	6
1	29	438	428	476	247	63

MEANS 4 BY 6

	1	2	3	4	5	6
1	85.655	74.737	88.488	83.586	79.988	84.016
2	126.034	82.826	94.491	85.908	98.166	112.111
3	125.379	56.037	74.446	61.162	96.903	107.143
4	90.138	110.251	80.348	81.395	74.785	85.921

SIGMAS 4 BY 6

	1	2	3	4	5	6
1	3.949	2.090	2.652	3.664	1.804	2.779
2	8.313	1.439	1.979	2.211	4.691	4.646
3	8.658	1.906	2.782	2.750	5.888	5.794
4	3.691	7.355	3.589	5.857	5.532	3.535

```
                 IC DIS       6 BY   6
          1          2          3          4          5          6

  1     .000     21.836     13.273     16.268      7.202      3.641
  2   21.836       .000     13.404      6.128     14.896     20.059
  3   13.273     13.404       .000      6.513      6.992     10.258
  4   16.268      6.128      6.513       .000      9.832     14.161
  5    7.202     14.896      6.992      9.832       .000      4.645
  6    3.641     20.059     10.258     14.161      4.645       .000

CLUSTER  SYMBOL  SIZE  R MEAN  R SIGMA   DIFF
   1       A      29   15.12    6.06    16.17
   2       B     438    6.95    3.97      .00
   3       C     428    4.89    2.75      .00
   4       D     476    6.87    3.59      .00
   5       E     247    8.46    4.36     1.16
   6       F      63   12.47    3.57    15.97
CHOOSE OPTION FROM
  MEANS   ANGDIS  QUIT    SIGMAS
>MEANS
                 MEANS        4 BY   6
          1          2          3          4          5          6

  1   85.655     74.737     88.488     83.586     79.988     84.016
  2  126.034     82.826     94.491     85.908     98.166    112.111
  3  125.379     56.037     74.446     61.162     96.903    107.143
  4   90.138    110.251     80.348     81.395     74.785     85.921

CHOOSE OPTION FROM
  MEANS   ANGDIS  QUIT    SIGMAS
>SIGMAS
                 SIGMAS       4 BY   6
          1          2          3          4          5          6

  1    3.949      2.090      2.652      3.664      1.804      2.779
  2    8.313      1.439      1.979      2.211      4.691      4.646
  3    8.658      1.906      2.782      2.750      5.888      5.794
  4    3.691      7.355      3.589      5.857      5.532      3.535

CHOOSE OPTION FROM
  MEANS   ANGDIS  QUIT    SIGMAS
>QUIT
 THE OPTION ITRCLU REQUIRED      27.1456 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

ITRCLU ENGINEERING DESCRIPTION

The iterative clustering algorithm, ITRCLU, develops the cluster means by using several passes through the data. For any one pass the data points are assigned to the nearest cluster means. Depending upon the value of S, the strip formulation logic is exercised for each iteration prior to the assignments to the nearest cluster means. Certain partial sums are computed, which at the end of the pass will represent the new cluster means and variances. During the assignments of data points to clusters, the means defining the current clusters are not modified.

At the end of each pass or iteration the new cluster means, channel variances, and populations are available. Those clusters whose number of points is less than a threshold, NMIN, are now eliminated. The algorithm then enters a cluster splitting or a cluster combining (merging) phase. The user can initialize the cluster splitting-combining operation by requesting that split iterations be performed until a specified percentage of clusters are stable (do not need to be split). The sequence for all other iterations is controlled by the input split combine sequence.

Strip Formulation.- If

$V_j(i)$ = the ith component of the jth vector to be assigned

S = strip refinement parameter (>0.)

then, the local group or strip is defined by the vectors $V_{j+\ell}$, $\ell=0,1,\ldots,L$, where L is the last ℓ for which

$$|V_j(i) - V_{j+\ell}(i)| \leq S$$

is valid for all i. After generating the local subgroup, its mean and weight are computed.

Cluster Splitting.-

In splitting a cluster, the channel with the largest variance (σ_j^2) is determined. If the standard deviation σ_j exceeds the threshold T1 (system parameter), the cluster is split along channel j alone into two subclusters. Assuming an n-channel vector space, let

m_i, i=1,...,n denote the mean vector for the initial cluster

m_i',i=1,...,n denote the mean vector for the first subcluster

m_i'',i=1,...,n denote the mean vector for the second subcluster

SEP denote a user-specified system parameter defining the separation of the new cluster means from that of the original cluster

Then the splitting process generates the two subclusters m_i' and m_i'' in a manner such that

$$m_i' = m_i ; i \neq j$$

$$m_i' = m_i + SEP * \sigma_j \quad ; i=j$$

$$m_i'' = m_i ; i \neq j$$

$$m_i'' = m_i - SEP * \sigma_j \quad ; i=j$$

Cluster Combining.-

There are two combining (merging) options depending on the value of the ISODAT flag.

If ISODAT = 0, cluster combining operates by computing the euclidean distance measure between the nearest pair of clusters. If this distance is less than the threshold T2, the two means are averaged into one. The nearest distance between clusters is recomputed and the combining process continues until all the cluster means are separated by T2 or more.

If ISODAT≠0, each cluster is limited to combine with at most one other cluster. The process begins with computing the minimum weighted distance between the first cluster and each of the other clusters. If this distance is less than T2, then the two respective means are averaged together. The mean averaging effectively combines two clusters into one cluster for the next pass of the data. The distance computations and thresholding continue until all of the original clusters are tested. Assuming an n-channel vector space let

m_i' and m_i'', i=1,...,n represents two mean vectors

σ_i' and σ_i'', i=1,...,n represents the individual channel sigmas associated with the two clusters then the distance d between the two clusters is

$$d = \left[\sum_{i=1}^{n} \frac{(m_i' - m_i'')^2}{\sigma_i' \sigma_i''} \right]^{1/2}$$

Special Cluster Splitting Tests.-

If $P > 0$, the algorithm forces cluster splits each iteration (up to a maximum of 10 splits) until a certain test is satisfied. The algorithm then resorts to the input split combine sequence for the remaining iterations. The test is - let

pass = number of clusters for which all of the individual channel sigmas are less than T1

num = number of clusters

then the algorithm will force splits until

$$\frac{\text{pass}}{\text{num}} \geq P$$

During the last iteration, a distance for the image display is associated with each pixel or strip of pixels. The value of this distance is equal to the distance (euclidean or L1 depending on the value of IDISF) of the mean of the strip to the cluster mean it is assigned to. Later in the IMAGE option, the distance for each pixel from its cluster mean is compared to a user input threshold. All pixels whose distance exceeds the threshold are displayed as blanks.

The ANGDIS option computes and displays the angles and distance between all pairs of mean vectors resulting from the clustering. Define

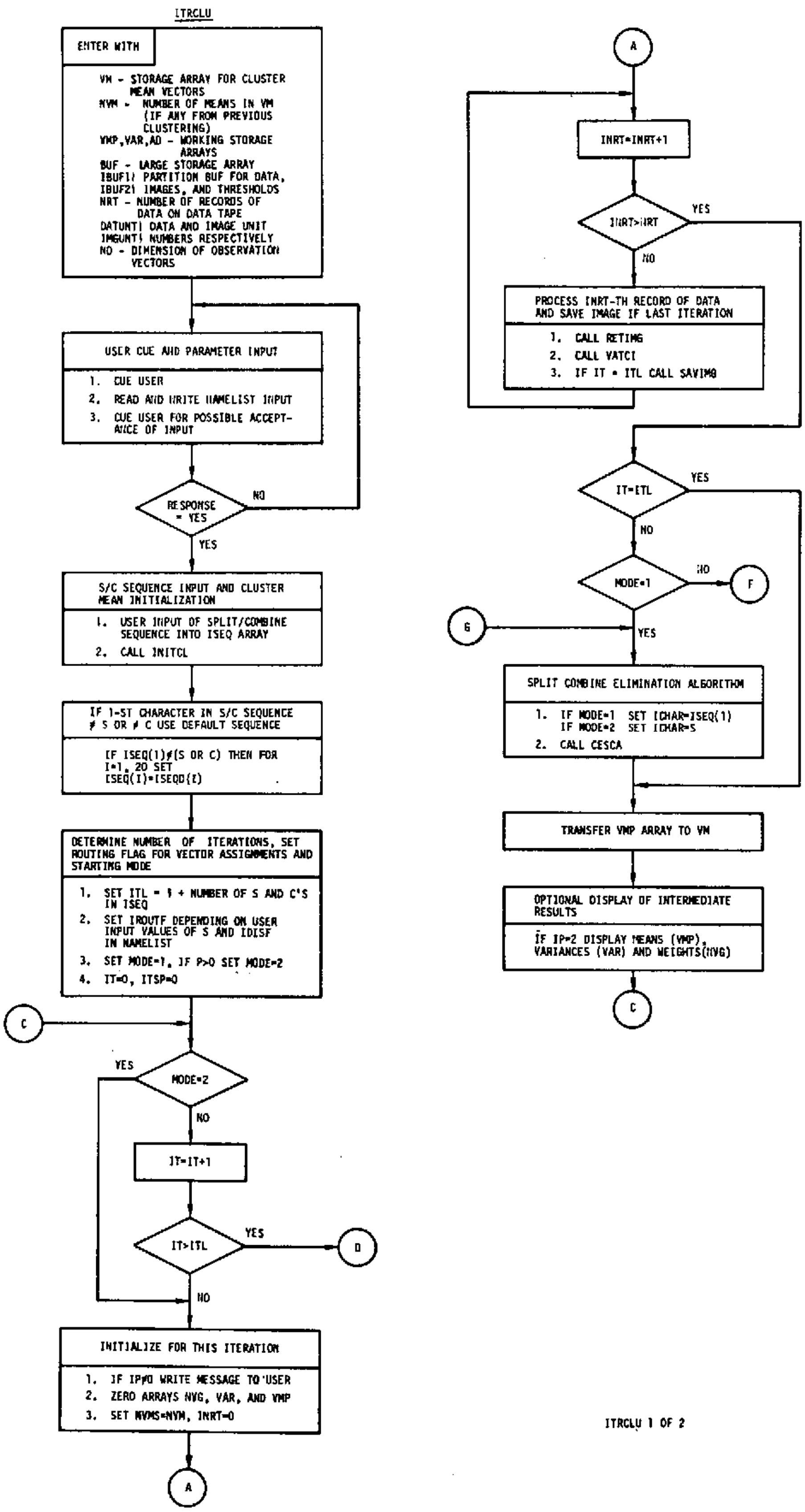
M_i = mean vector of the i^{th} cluster

$m_i(k)$ = k-th component of the i^{th} mean vector

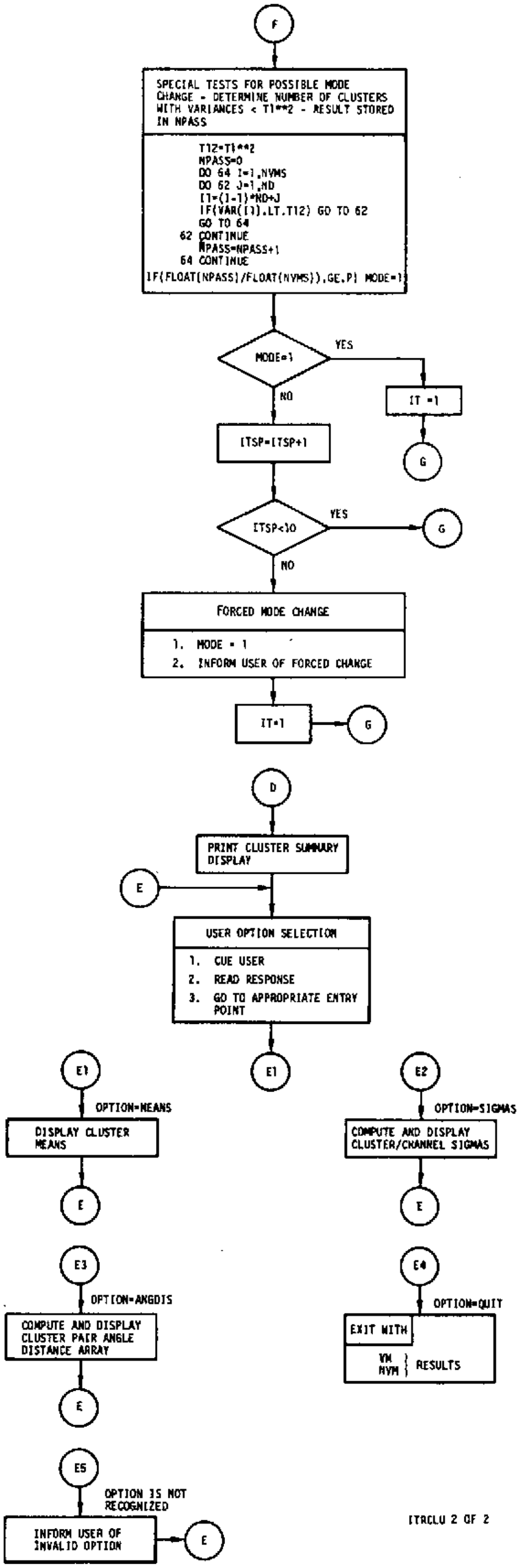
$$
dij = \begin{cases} i > j \text{ and IDISF} = 1 \\ \qquad dij = |M_i - M_j| \\ i > j \text{ and IDISF} = 2 \\ \qquad dij = \sum_k |m_i(k) - m_j(k)| \\ i = j \quad dij = 0 \\ i < j \quad dij = \frac{360}{2\pi} \cos^{-1} \frac{M_i \cdot M_j}{|M_i||M_j|} \end{cases}
$$

then the matrix D = [dij] is displayed.

The iterative clustering algorithm was initially reported in Reference 9. These ideas were applied and modified for use with multi-spectral data in References 10 and 11.

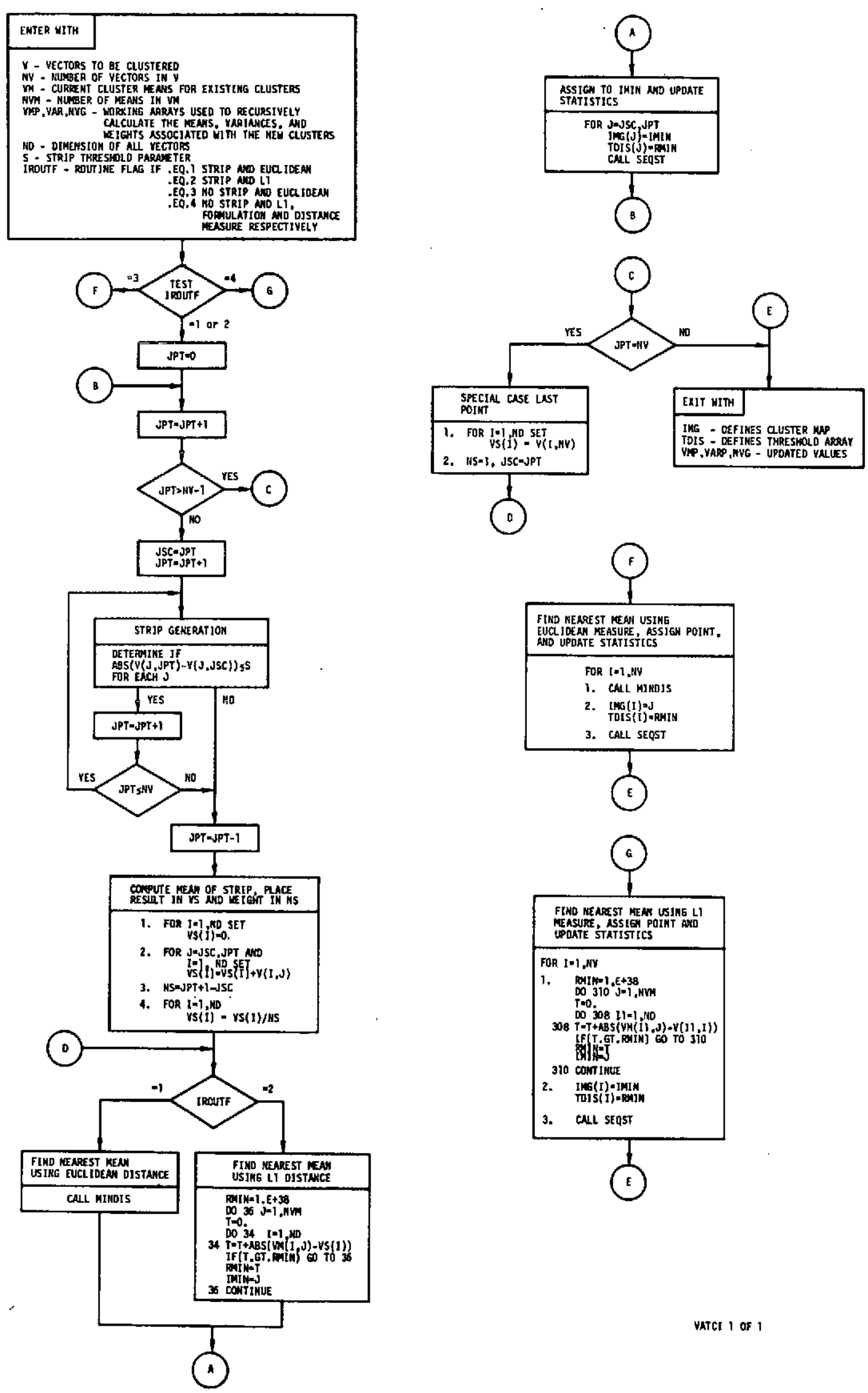

ITRCLU
ENTER WITH
VM - STORAGE ARRAY FOR CLUSTER MEAN VECTORS
NVM - NUMBER OF MEANS IN VM (IF ANY FROM PREVIOUS CLUSTERING)
VMP,VAR,AD - WORKING STORAGE ARRAYS
BUF - LARGE STORAGE ARRAY
IBUF1} PARTITION BUF FOR DATA,
IBUF2} IMAGES, AND THRESHOLDS
NRT - NUMBER OF RECORDS OF DATA ON DATA TAPE
DATUNT} DATA AND IMAGE UNIT
IMGUNT} NUMBERS RESPECTIVELY
NO - DIMENSION OF OBSERVATION VECTORS
USER CUE AND PARAMETER INPUT
1. CUE USER
2. READ AND WRITE NAMELIST INPUT
3. CUE USER FOR POSSIBLE ACCEPTANCE OF INPUT
RESPONSE = YES
NO
YES
S/C SEQUENCE INPUT AND CLUSTER MEAN INITIALIZATION
1. USER INPUT OF SPLIT/COMBINE SEQUENCE INTO ISEQ ARRAY
2. CALL INITCL
IF 1-ST CHARACTER IN S/C SEQUENCE ≠ S OR ≠ C USE DEFAULT SEQUENCE
IF ISEQ(1)≠(S OR C) THEN FOR I=1, 20 SET ISEQ(I)=ISEQD(I)
DETERMINE NUMBER OF ITERATIONS, SET ROUTING FLAG FOR VECTOR ASSIGNMENTS AND STARTING MODE
1. SET ITL = 1 + NUMBER OF S AND C'S IN ISEQ
2. SET IROUTF DEPENDING ON USER INPUT VALUES OF S AND IDISF IN NAMELIST
3. SET MODE=1, IF P>0 SET MODE=2
4. IT=0, ITSP=0
C
MODE=2
YES
NO
IT=IT+1
IT>ITL
YES
D
NO
INITIALIZE FOR THIS ITERATION
1. IF IP≠0 WRITE MESSAGE TO USER
2. ZERO ARRAYS NVG, VAR, AND VMP
3. SET NVMS=NVM, INRT=0
A
A
INRT=INRT+1
INRT>NRT
YES
NO
PROCESS INRT-TH RECORD OF DATA AND SAVE IMAGE IF LAST ITERATION
1. CALL RETIMG
2. CALL VATCI
3. IF IT = ITL CALL SAVIMG
IT=ITL
YES
NO
MODE=1
NO
F
G
YES
SPLIT COMBINE ELIMINATION ALGORITHM
1. IF MODE=1 SET ICHAR=ISEQ(1)
IF MODE=2 SET ICHAR=S
2. CALL CESCA
TRANSFER VMP ARRAY TO VM
OPTIONAL DISPLAY OF INTERMEDIATE RESULTS
IF IP=2 DISPLAY MEANS (VMP), VARIANCES (VAR) AND WEIGHTS(NVG)
C

ITRCLU 1 OF 2


F
SPECIAL TESTS FOR POSSIBLE MODE CHANGE - DETERMINE NUMBER OF CLUSTERS WITH VARIANCES < T1**2 - RESULT STORED IN NPASS
T12=T1**2
NPASS=0
DO 64 I=1,NVMS
DO 62 J=1,ND
I1=(I-1)*ND+J
IF(VAR(I1).LT.T12) GO TO 62
GO TO 64
62 CONTINUE
NPASS=NPASS+1
64 CONTINUE
IF(FLOAT(NPASS)/FLOAT(NVMS)).GE.P) MODE=1
MODE=1
YES
NO
IT =1
G
ITSP=ITSP+1
ITSP<10
YES
G
NO
FORCED MODE CHANGE
1. MODE = 1
2. INFORM USER OF FORCED CHANGE
IT=1
G
D
PRINT CLUSTER SUMMARY DISPLAY
E
USER OPTION SELECTION
1. CUE USER
2. READ RESPONSE
3. GO TO APPROPRIATE ENTRY POINT
E1
E1
OPTION=MEANS
DISPLAY CLUSTER MEANS
E
E2
OPTION=SIGMAS
COMPUTE AND DISPLAY CLUSTER/CHANNEL SIGMAS
E
E3
OPTION=ANGDIS
COMPUTE AND DISPLAY CLUSTER PAIR ANGLE DISTANCE ARRAY
E
E4
OPTION=QUIT
EXIT WITH
VM
NVM
RESULTS
E5
OPTION IS NOT RECOGNIZED
INFORM USER OF INVALID OPTION
E

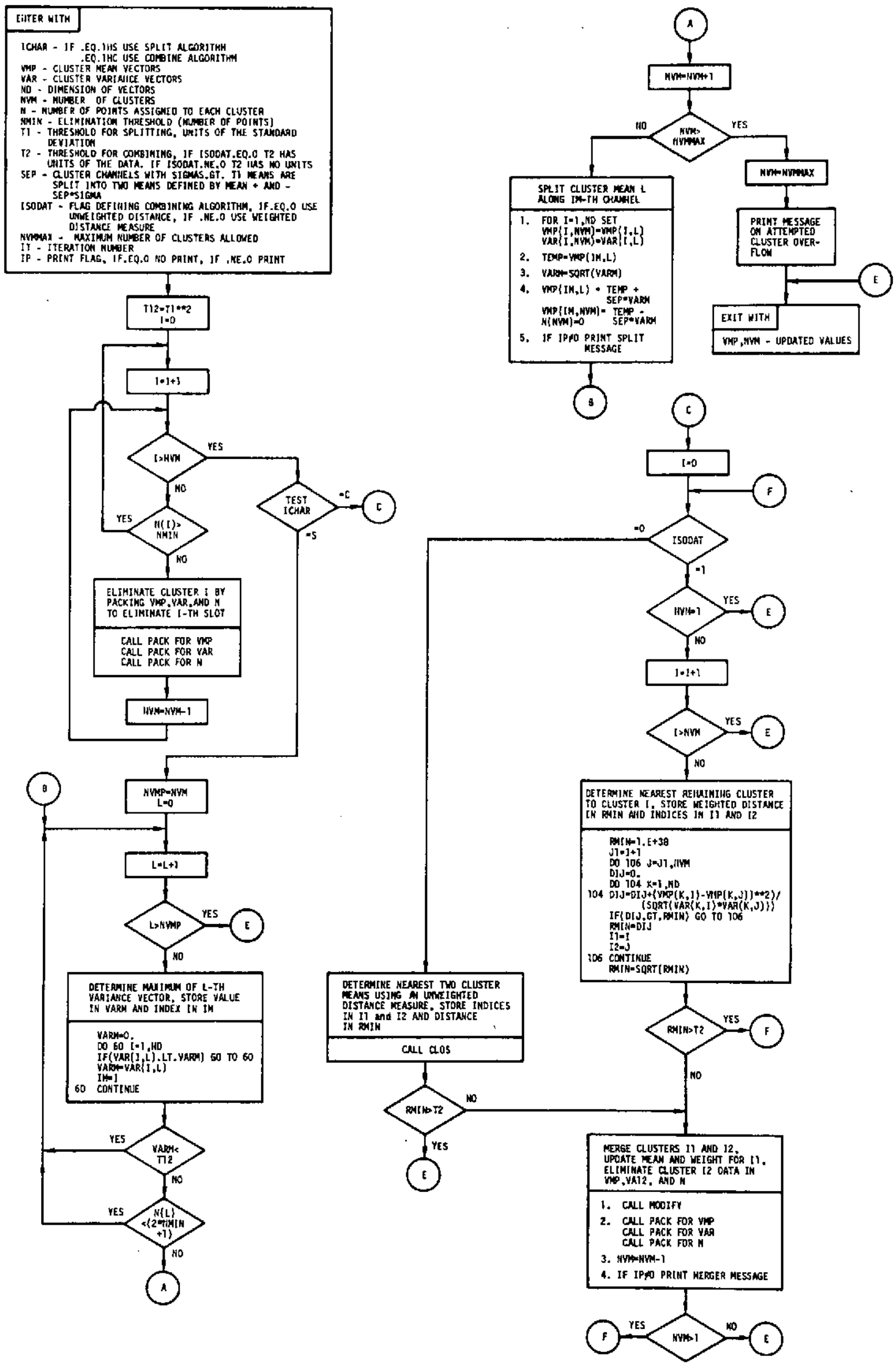
ITRCLU 2 OF 2

VATCI



VATCI 1 OF 1

CESCA



CESCA 1 OF 1

ENTER WITH

V - SET OF VECTORS
VT - VECTOR
ND - DIMENSION OF EACH VECTOR
NV - NUMBER OF VECTORS IN V

↓

DETERMINE NEAREST VECTOR IN V TO VT USING EUCLIDEAN DISTANCE MEASURE

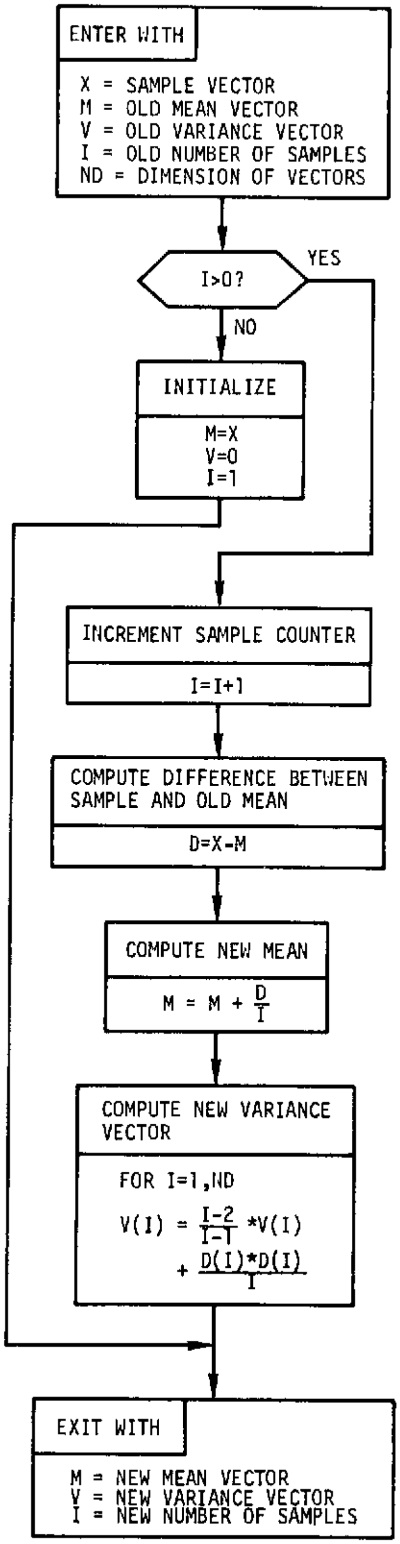
1. RMIN = 1. E+38
2. FOR I=1, NV
   a) COMPUTE DISTANCE BETWEEN VT AND I-TH VECTOR IN V - STORE DISTANCE IN D
   b) IF D < RMIN SET J=I AND RMIN=D

↓

EXIT WITH

RMIN - DISTANCE TO NEAREST VECTOR
J - INDEX TO VECTOR

SEQST



SEQST 1 OF 1

CLOS

ENTER WITH

V- ARRAY OF VECTORS
ND - DIMENSION OF EACH VECTOR IN
NV - NUMBER OF VECTORS IN V

↓

DETERMINE NEAREST TWO VECTORS IN V USING AN EUCLIDEAN MEASURE

1. RMIN=1.E+38
2. FOR I=2, NV
   SET K1=I-1
   K2=I AND
   a) DETERMINE NEAREST VECTOR IN V FROM K2 → NV TO VECTOR K1 - CALL MINDIS
   b) IF NEAREST DISTANCE D<RMIN SET RMIN=D AND UPDATE INDICES J1 AND J2 TO VECTOR PAIR

↓

EXIT WITH

RMIN - DISTANCE BETWEEN NEAREST PAIR
J1,J2 - INDICES TO PAIR

Using the MAXLIK Option

Upon entering the MAXLIK option the user must select one of the sub-options REDSIG, INPSIG, PROCSS or QUIT.

REDSIG allows user to retrieve previously saved spectral signatures (mean vectors and covariance matrices). For one or more signatures to be retrieved, the file number on which signature(s) were saved must be input. Entry of 0 terminates the REDSIG suboption. The routine used to retrieve signatures (REDSIG) contains an option to display each of the retrieved signatures as they are located on the signature file. A user response of YES will cause the signatures to be displayed - any other response will bypass the display. Following this, user must enter a name under which given signature was saved, and then the actual retrieval (and display if requested) takes place. Up to 12 signatures may be retrieved this way. An entry of NOMORE for signature name causes the retrieval process to terminate. The retrieved signatures may then be used by the PROCSS sub-option.

INPSIG allows the user, as an alternate to REDSIG, to input the signature data directly. Its parameters are

NVM = number of signatures (means and covariances to be entered)

ND = dimension of each mean vector

VM = the mean vectors, a one-dimensional array of ND x NVM values representing the matrix of mean vectors input by columns.

COV = the covariance matrices, a one-dimensional array of (NDxND)xNVM values representing the NVM ND x ND full matrices.

Once the signatures have been retrieved (or input) PROCSS is called to process the data. This option first prints out the interclass distance array. This array gives the maximum likelihood measure of the distances between the mean vectors of the various signatures. Then the observation data is processed. Upon completion a class summary is displayed. This summary gives the class number (which corresponds to the order in which the signatures were input), the assigned symbol, and the class size resulting from the processing.

QUIT returns control to the ASTEP driver.

MAXLIK OPTION

SAMPLE 1 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>MAXLIK


                    MAXLIK  OPTION
                    **************


CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT   INPSIG

>INPSIG


 $ININPS   ND,NVM,VM,COV

 ND= 4 NVM= 5

 SIGNATURE  1

                   MEAN        4 BY  1
             1

   1    83.600
   2    85.900
   3    61.100
   4    81.400


                   COVMAT      4 BY  4
             1          2          3          4

   1     1.000       .000       .000       .000
   2      .000      1.000       .000       .000
   3      .000       .000      1.000       .000
   4      .000       .000       .000      1.000


 SIGNATURE  2

                   MEAN        4 BY  1
             1

   1    88.600
   2    94.500
   3    74.300
   4    80.300
```

COVMAT 4 BY 4

	1	2	3	4
1	1.000	.000	.000	.000
2	.000	1.000	.000	.000
3	.000	.000	1.000	.000
4	.000	.000	.000	1.000

SIGNATURE 3

MEAN 4 BY 1

	1
1	74.700
2	82.800
3	56.000
4	110.300

COVMAT 4 BY 4

	1	2	3	4
1	1.000	.000	.000	.000
2	.000	1.000	.000	.000
3	.000	.000	1.000	.000
4	.000	.000	.000	1.000

SIGNATURE 4

MEAN 4 BY 1

	1
1	80.100
2	97.400
3	94.200
4	74.200

COVMAT 4 BY 4

	1	2	3	4
1	1.000	.000	.000	.000
2	.000	1.000	.000	.000
3	.000	.000	1.000	.000
4	.000	.000	.000	1.000

SIGNATURE 5

```
                    MEAN        4 BY  1
         1

  1     82.800
  2    111.000
  3    110.400
  4     83.900


                    COVMAT      4 BY  4
         1             2             3             4

  1      1.000          .000          .000          .000
  2       .000         1.000          .000          .000
  3       .000          .000         1.000          .000
  4       .000          .000          .000         1.000


TYPE YES IF INPUTS ARE OK
>YES

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>PROCSS
                    IC DIS      5 BY  5
         1             2             3             4             5

  1       .000       274.410       950.040      1291.950      3067.390
  2    274.410          .000      1564.990       513.880      1622.060
  3    950.040      1564.990          .000      3004.770      4517.170
  4   1291.950       513.880      3004.770          .000       548.780
  5   3067.390      1622.060      4517.170       548.780          .000

CLASS  SYMBOL  SIZE
  1      A      476
  2      B      428
  3      C      437
  4      D      212
  5      E      128

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>QUIT
 THE OPTION MAXLIK REQUIRED        5.6084 SECONDS OF CPU TIME.

 -----------------------------------------------------------
```

MAXLIK OPTION
SAMPLE 2 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>MAXLIK

                    MAXLIK  OPTION
                    **************

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>REDSIG

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>1

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  1
>YES

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>SIGA

SIGA   ND =  4 K =   1  6  9 12
NUM(1) =   438

                MEAN        1 BY  4
         1            2            3            4

 1     74.737     82.826     56.037    110.251


                COVMAT      4 BY  4
         1            2            3            4

 1      4.368       .643      1.428     -2.900
 2       .643      2.071      1.082      1.009
 3      1.428      1.082      3.633     -3.927
 4     -2.900      1.009     -3.927     54.102

>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>1
```

```
INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  1
>NO

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>SIGB

SIGB    ND =   4 K =    1   6   9 12
NUM(1) =    428
>SIGC

SIGC    ND =   4 K =    1   6   9 12
NUM(1) =    476
>SIGDE

SIGDE   ND =   4 K =    1   6   9 12
NUM(1) =    339
>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
   4 SIGNATURES HAVE BEEN RETRIEVED

CHOOSE MAXLIK OPTION FROM
   REDSIG   PROCSS   QUIT     INPSIG

>PROCSS
                      IC DIS       4 BY  4
           1             2            3            4

  1      7.016     135.893      38.095     635.658
  2    169.933       7.156      36.054     111.352
  3     36.748      33.383       8.552     227.064
  4    202.652      47.109      63.208      11.890

CLASS   SYMBOL   SIZE
   1       A      440
   2       B      420
   3       C      475
   4       D      346

CHOOSE MAXLIK OPTION FROM
   REDSIG   PROCSS   QUIT     INPSIG

>QUIT
 THE OPTION MAXLIK REQUIRED       5.1126 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

MAXLIK ENGINEERING DESCRIPTION

PROCSS is the only MAXLIK option requiring an engineering description. PROCSS classifies the data vectors via a maximum likelihood algorithm.

Define

M_i = mean vector for i^{th} class

Λ_i = covariance matrix for i^{th} class

X = sample data vector

then, for each sample vector X the i for which

$$(X - M_i)^T \Lambda_i^{-1} (X - M_i) + \ell n |\Lambda_i| \tag{1}$$

is minimized is determined.

An alphabetic image array or map of the resulting classifications is generated. The correspondence between the class numbers and alphabetic characters is i=1 is an A, i=2 is a B etc. The threshold distance for the image display for each sample data vector or pixel is the chi-squared variable

$$(X - M_i)^T \Lambda_i^{-1} (X - M_i)$$

where the i is the one which yields the minimum of the expression (1).

Prior to classifying the data, PROCSS generates and displays an inter-class distance array defined by

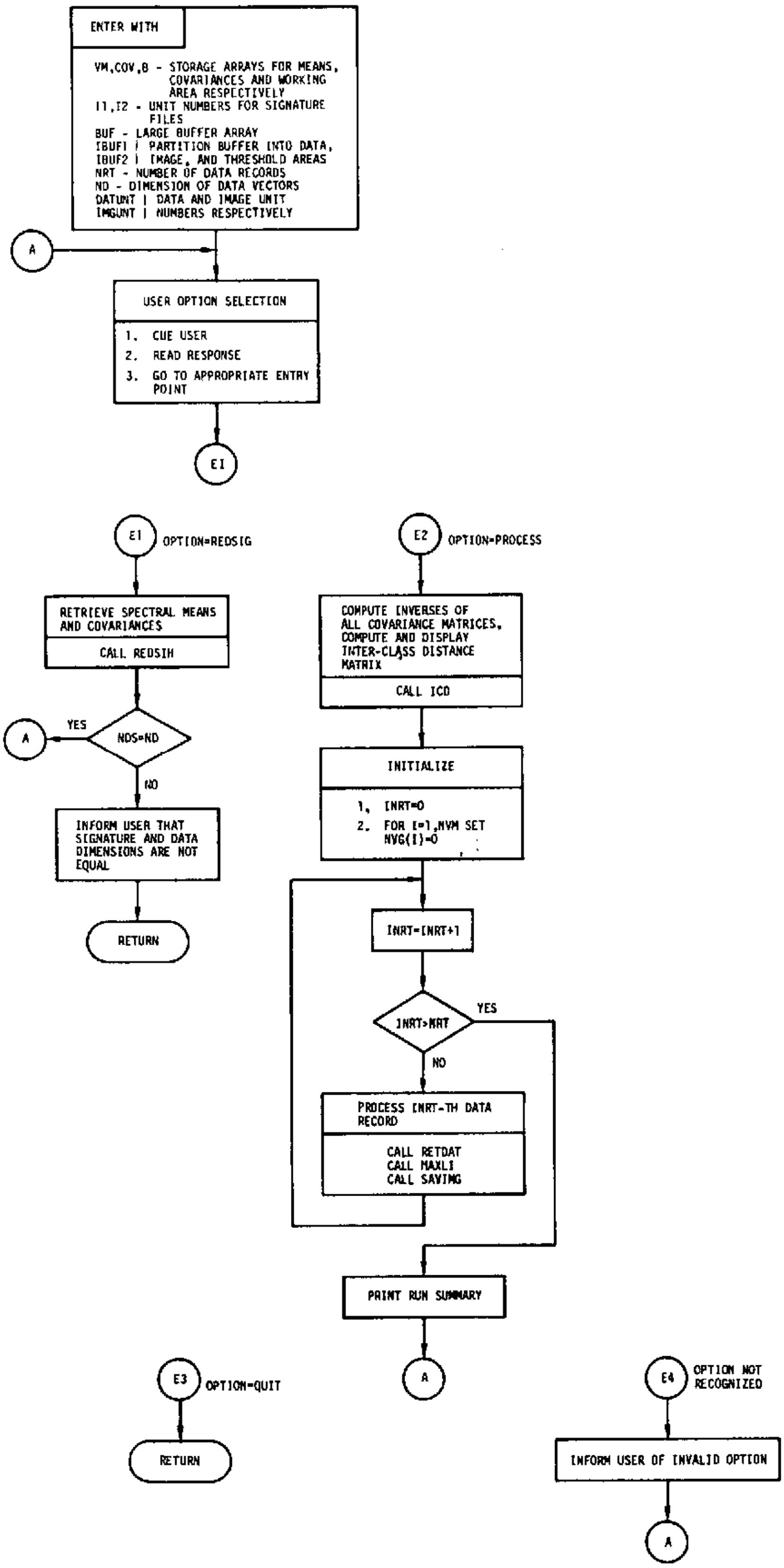
d_{ij} = distance of class i from class j in the maximum likelihood sense

$$= (M_i - M_j)^T \Lambda_i^{-1} (M_i - M_j) + \ell n|\Lambda_i|$$

The matrix $D = (d_{ij})$ is then displayed.

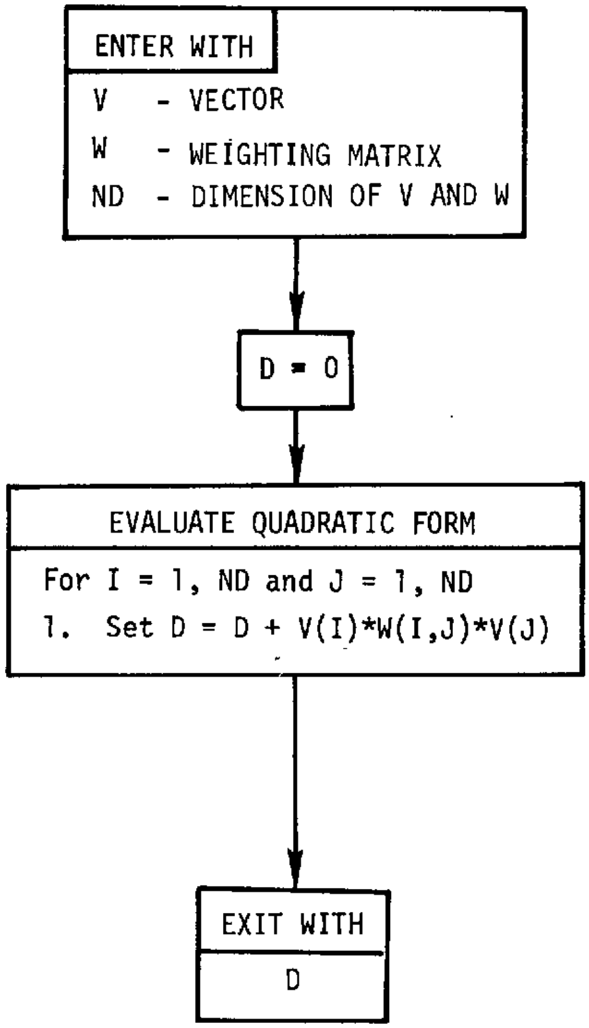
The maximum likelihood algorithm is reported in Reference 12.

MAXLIK



MAXLIK 1 OF 1

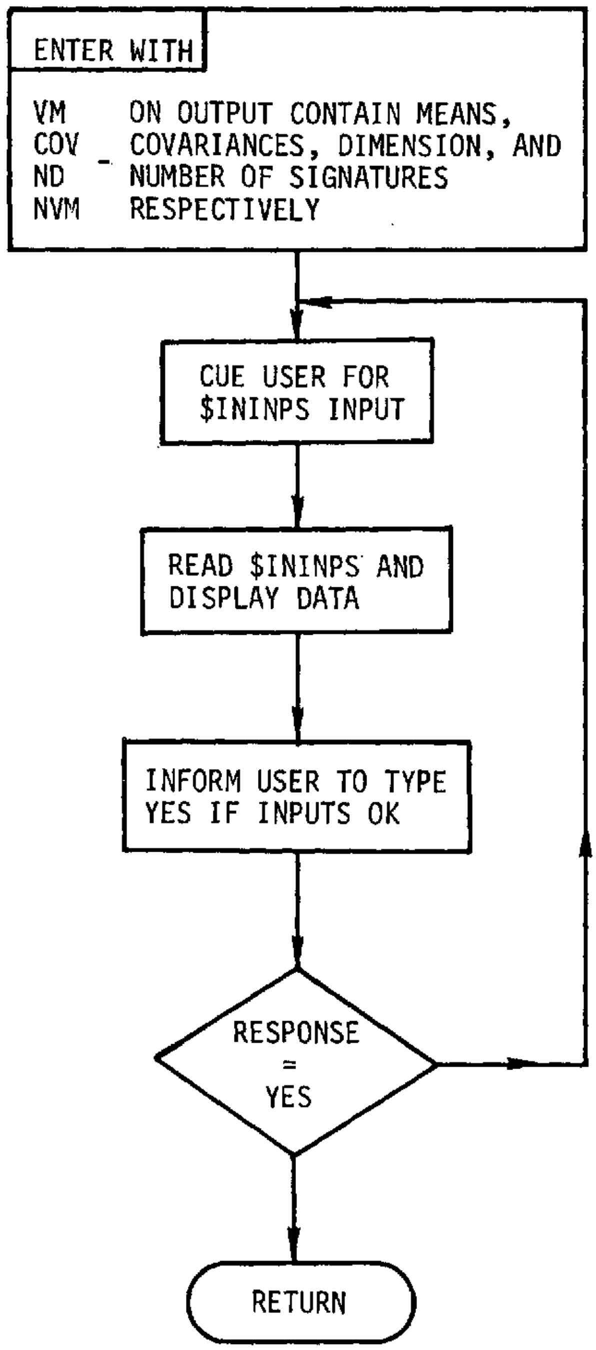
QF



ENTER WITH
V - VECTOR
W - WEIGHTING MATRIX
ND - DIMENSION OF V AND W
D = 0
EVALUATE QUADRATIC FORM
For I = 1, ND and J = 1, ND
1. Set D = D + V(I)*W(I,J)*V(J)
EXIT WITH
D

QF 1 of 1

INPSIG



INPSIG 1 of 1

ENTER WITH

VM, COV - APRIORI MEANS AND COVARIANCES
ND - DIMENSION OF EACH VM AND COV
NVM - NUMBER OF MEANS AND COVARIANCES

↓

COMPUTE INVERSE OF EACH COVARIANCE

1. COMPUTE INVERSE OF EACH MATRIX IN COV ARRAY AND STORE BACK IN COV
2. STORE LN OF EACH DETERMINATE IN DET ARRAY

↓

COMPUTE INTER-CLASS DISTANCES

FOR I=1, NVM AND J=1, NVM

1. SET VD(K)=VM(K,I)-VM(K,J), K=1,ND
2. CALL QF
3. SET D(I,J)=DET(I)+T

↓

DISPLAY D

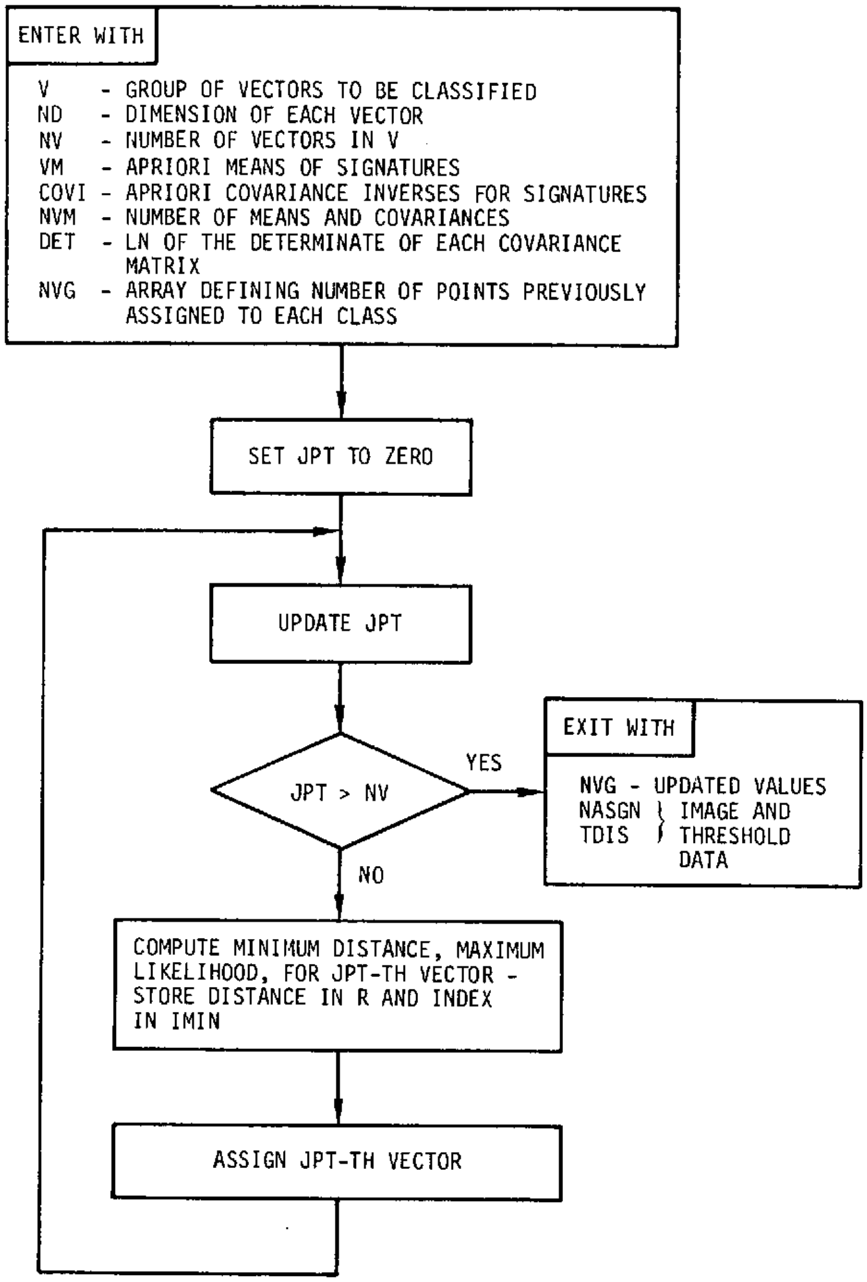
PRINT MATRIX D

↓

EXIT WITH

COV - INVERSES OF INPUT MATRICES
DET - LN OF |COV|

MAXLI


ENTER WITH
V - GROUP OF VECTORS TO BE CLASSIFIED
ND - DIMENSION OF EACH VECTOR
NV - NUMBER OF VECTORS IN V
VM - APRIORI MEANS OF SIGNATURES
COVI - APRIORI COVARIANCE INVERSES FOR SIGNATURES
NVM - NUMBER OF MEANS AND COVARIANCES
DET - LN OF THE DETERMINATE OF EACH COVARIANCE MATRIX
NVG - ARRAY DEFINING NUMBER OF POINTS PREVIOUSLY ASSIGNED TO EACH CLASS
SET JPT TO ZERO
UPDATE JPT
JPT > NV
YES
NO
EXIT WITH
NVG - UPDATED VALUES
NASGN
TDIS
IMAGE AND THRESHOLD DATA
COMPUTE MINIMUM DISTANCE, MAXIMUM LIKELIHOOD, FOR JPT-TH VECTOR - STORE DISTANCE IN R AND INDEX IN IMIN
ASSIGN JPT-TH VECTOR

Using the NEWS Option

The NEWS option provides a convenient method for communicating to the user minor changes to the ASTEP program by reading information from unit 15 (file name NEWS with qualifier TRW-T33710) and printing it. This data may be written or modified by UPNEWS option. The latest information is always printed first.

After requesting NEWS, the user is asked to input either the number of lines to be printed or, ALL to indicate all of the remaining file is to be printed, or QUIT to indicate that no more print is requested. If the number of lines option is requested, the user will again be queried for additional lines, ALL or QUIT. The process is repeated until the file is complete, or ALL or QUIT is input.

Three sample uses of the NEWS option follow. In the first sample, the ALL response is used to print all lines in the NEWS file. In the second sample, only the first three lines are printed. Then ALL is requested to print the remainder of the file. In the third sample, NEWS is entered and three lines are printed, followed by two more lines, and QUIT to indicate that no further printing is requested.

NEWS OPTION

SAMPLE 1 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>NEWS

                    NEWS    OPTION
                    ==============

NEWS   OPTION REQUIRES FILE NAME NEWS TO BE ASSIGNED
IF NOT ASSIGNED INDICATE QUIT,
QUIT FROM ASTEP, ASSIGN NEWS, AND RE-EXECUTE ASTEP.
INPUT ADDITIONAL LINES TO BE PRINTED, ALL, OR QUIT
>ALL
THIS IS A SAMPLE NEWS FILE.                                            1
IT WILL DEMONSTRATE THE ALL AND LINE COUNT                             2
OPTIONS OF NEWS.                                                       3
UPON FIRST ENTRY TO NEWS, THE ENTIRE FILE WILL                         4
BE PRINTED.                                                            5
THE SECOND ENTRY PRINTS THE FIRST 3 LINES FOLLOWED                     6
BY THE REMAINDER OF THE FILE.                                          7
THE LAST ENTRY PRINTS THE FIRST 3 LINES FOLLOWED                       8
BY 2 ADDITIONAL LINES.                                                 9
 THE OPTION NEWS   REQUIRED       .0496 SECONDS OF CPU TIME.

 -------------------------------------------------------------
```

NEWS OPTION

SAMPLE 2 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>NEWS

                    NEWS    OPTION
                    ==============

NEWS   OPTION REQUIRES FILE NAME NEWS TO BE ASSIGNED
IF NOT ASSIGNED INDICATE QUIT,
QUIT FROM ASTEP, ASSIGN NEWS, AND RE-EXECUTE ASTEP.
INPUT ADDITIONAL LINES TO BE PRINTED, ALL, OR QUIT
>3
THIS IS A SAMPLE NEWS FILE.                                          1
IT WILL DEMONSTRATE THE ALL AND LINE COUNT                           2
OPTIONS OF NEWS.                                                     3
INPUT ADDITIONAL LINES TO BE PRINTED, ALL, OR QUIT
>ALL
UPON FIRST ENTRY TO NEWS, THE ENTIRE FILE WILL                       4
BE PRINTED.                                                          5
THE SECOND ENTRY PRINTS THE FIRST 3 LINES FOLLOWED                   6
BY THE REMAINDER OF THE FILE.                                        7
THE LAST ENTRY PRINTS THE FIRST 3 LINES FOLLOWED                     8
BY 2 ADDITIONAL LINES.                                               9
 THE OPTION NEWS   REQUIRED       .0494 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

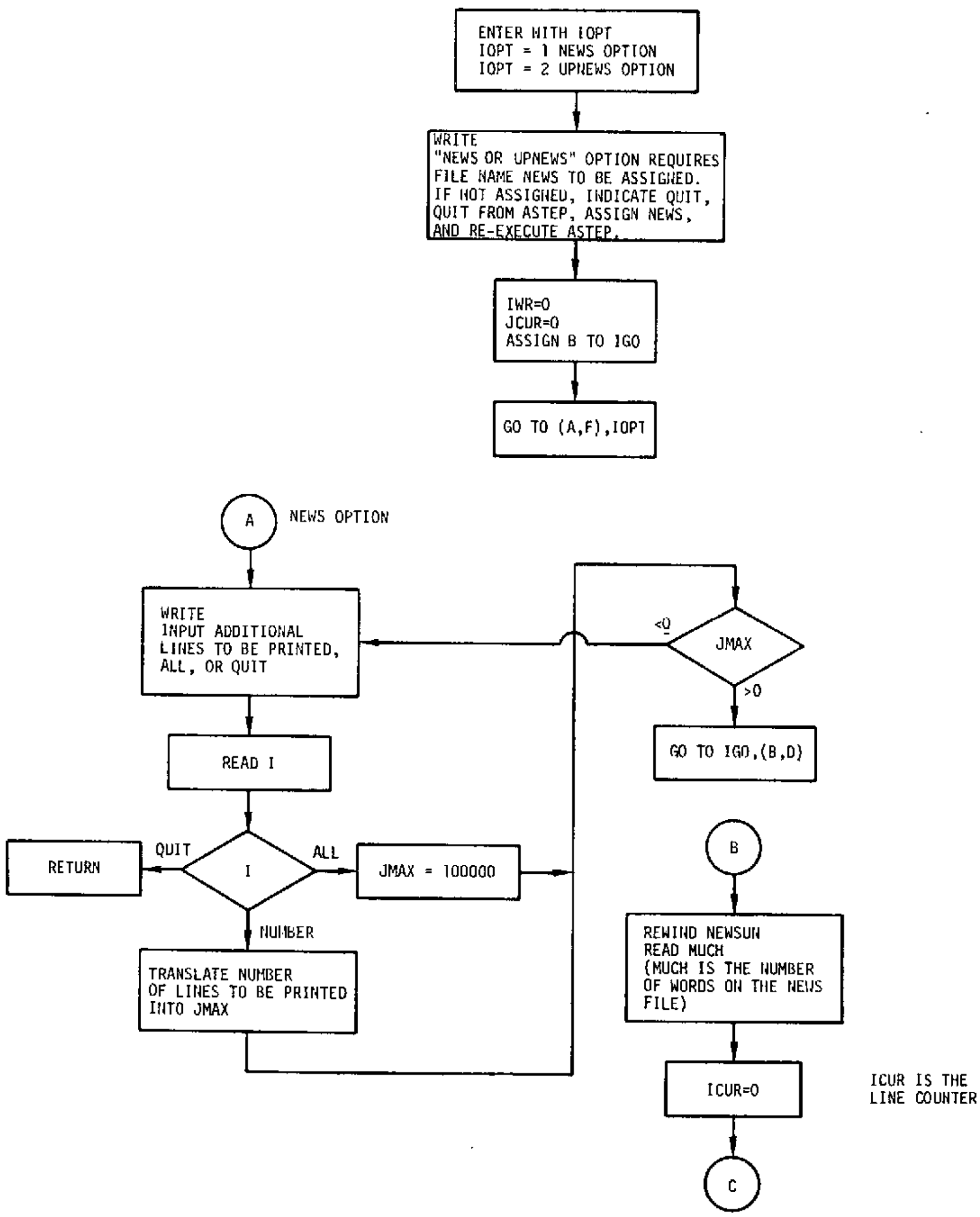
NEWS OPTION

SAMPLE 3 INPUT AND CORRESPONDING OUTPUT:

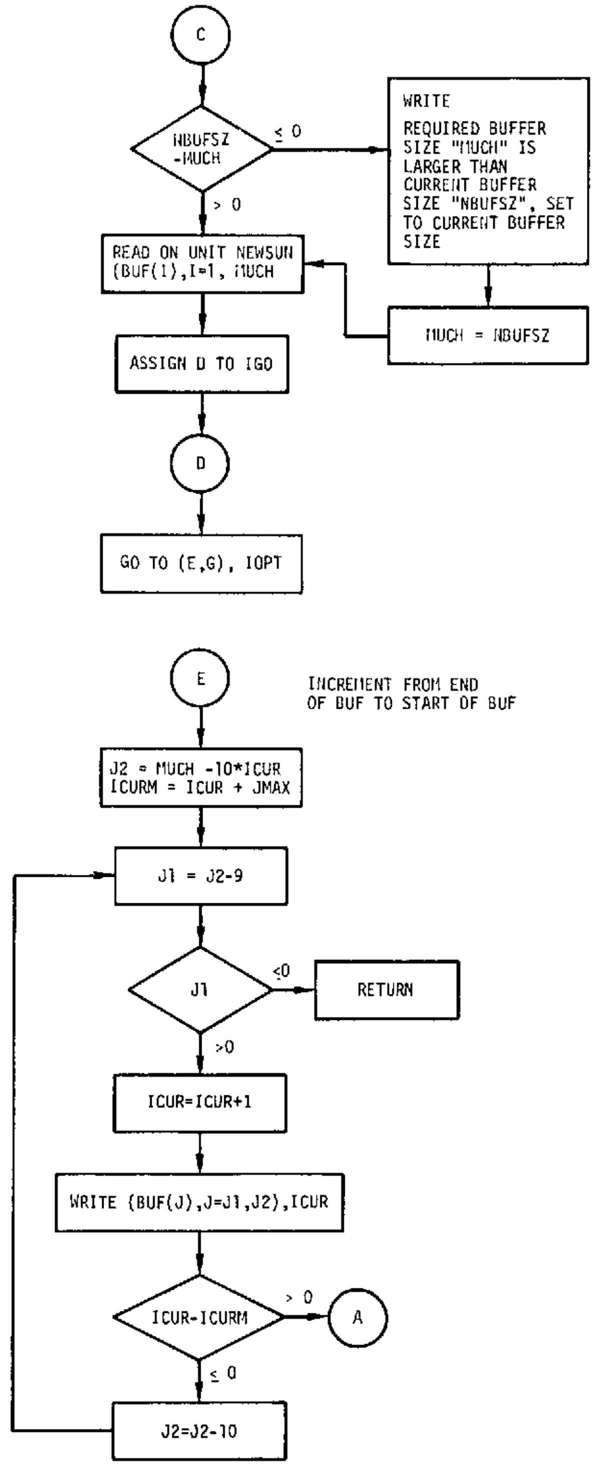
```
ENTER ASTEP OPTION OR TYPE A BLANK
>NEWS

                    NEWS    OPTION
                    **************

NEWS   OPTION REQUIRES FILE NAME NEWS TO BE ASSIGNED
IF NOT ASSIGNED INDICATE QUIT,
QUIT FROM ASTEP, ASSIGN NEWS, AND RE-EXECUTE ASTEP.
INPUT ADDITIONAL LINES TO BE PRINTED, ALL, OR QUIT
>3
THIS IS A SAMPLE NEWS FILE.                                              1
IT WILL DEMONSTRATE THE ALL AND LINE COUNT                               2
OPTIONS OF NEWS.                                                         3
INPUT ADDITIONAL LINES TO BE PRINTED, ALL, OR QUIT
>2
UPON FIRST ENTRY TO NEWS, THE ENTIRE FILE WILL                           4
BE PRINTED.                                                              5
INPUT ADDITIONAL LINES TO BE PRINTED, ALL, OR QUIT
>QUIT
 THE OPTION NEWS   REQUIRED       .0466 SECONDS OF CPU TIME.

 ---------------------------------------------------------------
```

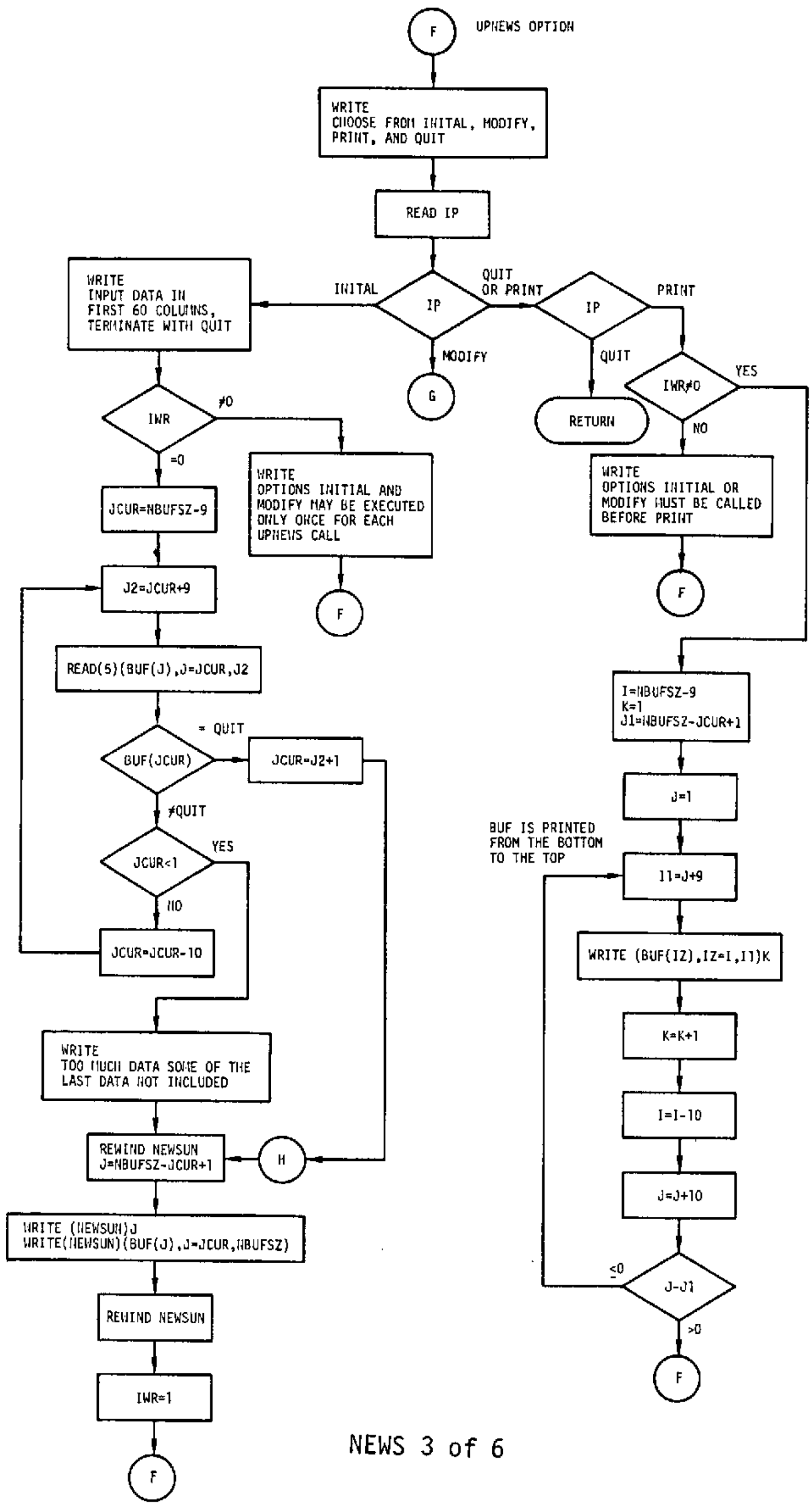
NEWS ENGINEERING DESCRIPTION

NEWS does not require an engineering description - see the following flow diagram.

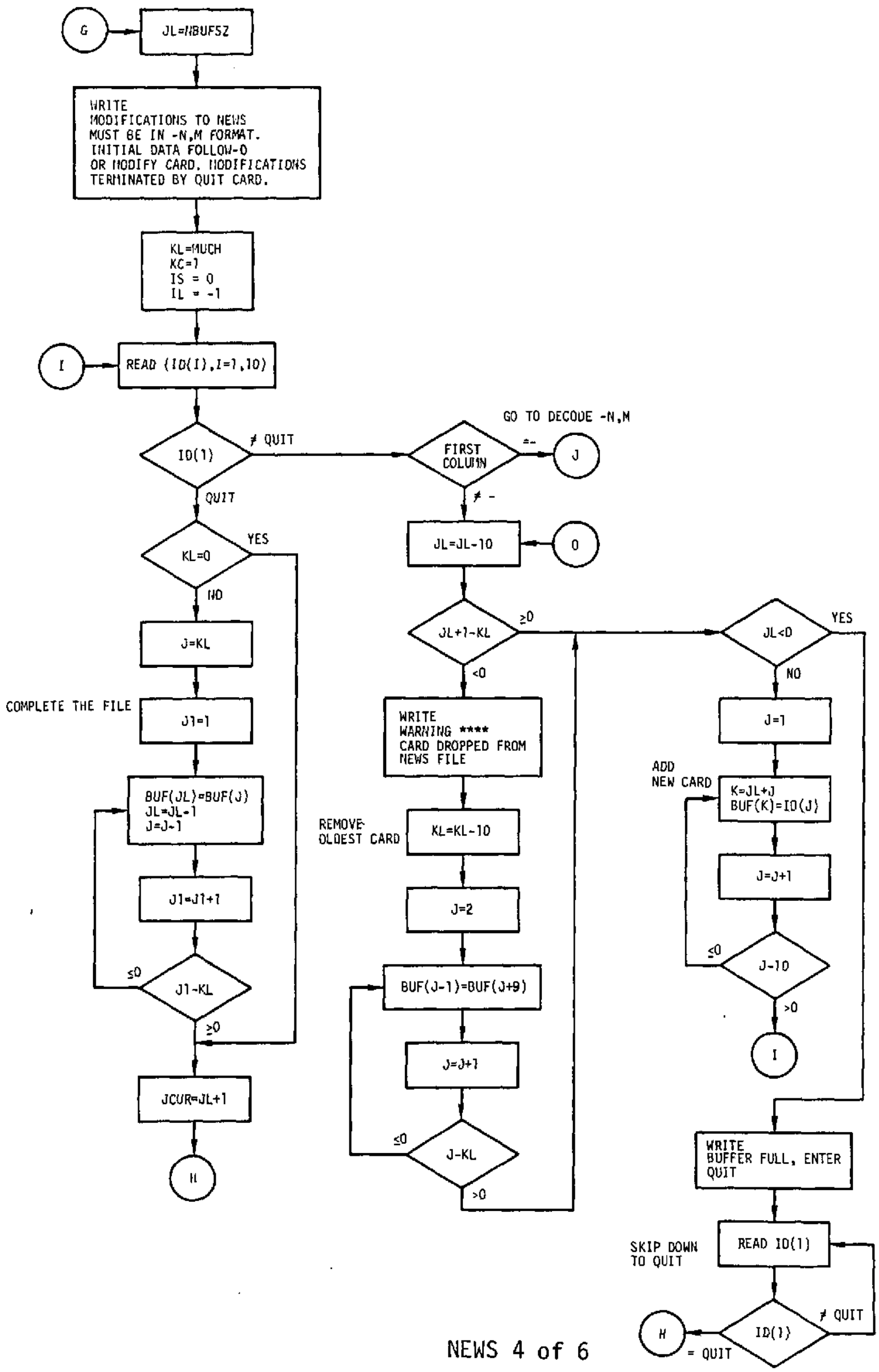

ENTER WITH IOPT
IOPT = 1 NEWS OPTION
IOPT = 2 UPNEWS OPTION
WRITE
"NEWS OR UPNEWS" OPTION REQUIRES FILE NAME NEWS TO BE ASSIGNED. IF NOT ASSIGNED, INDICATE QUIT, QUIT FROM ASTEP, ASSIGN NEWS, AND RE-EXECUTE ASTEP.
IWR=0
JCUR=0
ASSIGN B TO IGO
GO TO (A,F),IOPT
A
NEWS OPTION
WRITE
INPUT ADDITIONAL LINES TO BE PRINTED, ALL, OR QUIT
READ I
I
QUIT
RETURN
ALL
JMAX = 100000
NUMBER
TRANSLATE NUMBER OF LINES TO BE PRINTED INTO JMAX
JMAX
<0
>0
GO TO IGO,(B,D)
B
REWIND NEWSUN
READ MUCH
(MUCH IS THE NUMBER OF WORDS ON THE NEWS FILE)
ICUR=0
ICUR IS THE LINE COUNTER
C

NEWS 1 of 6

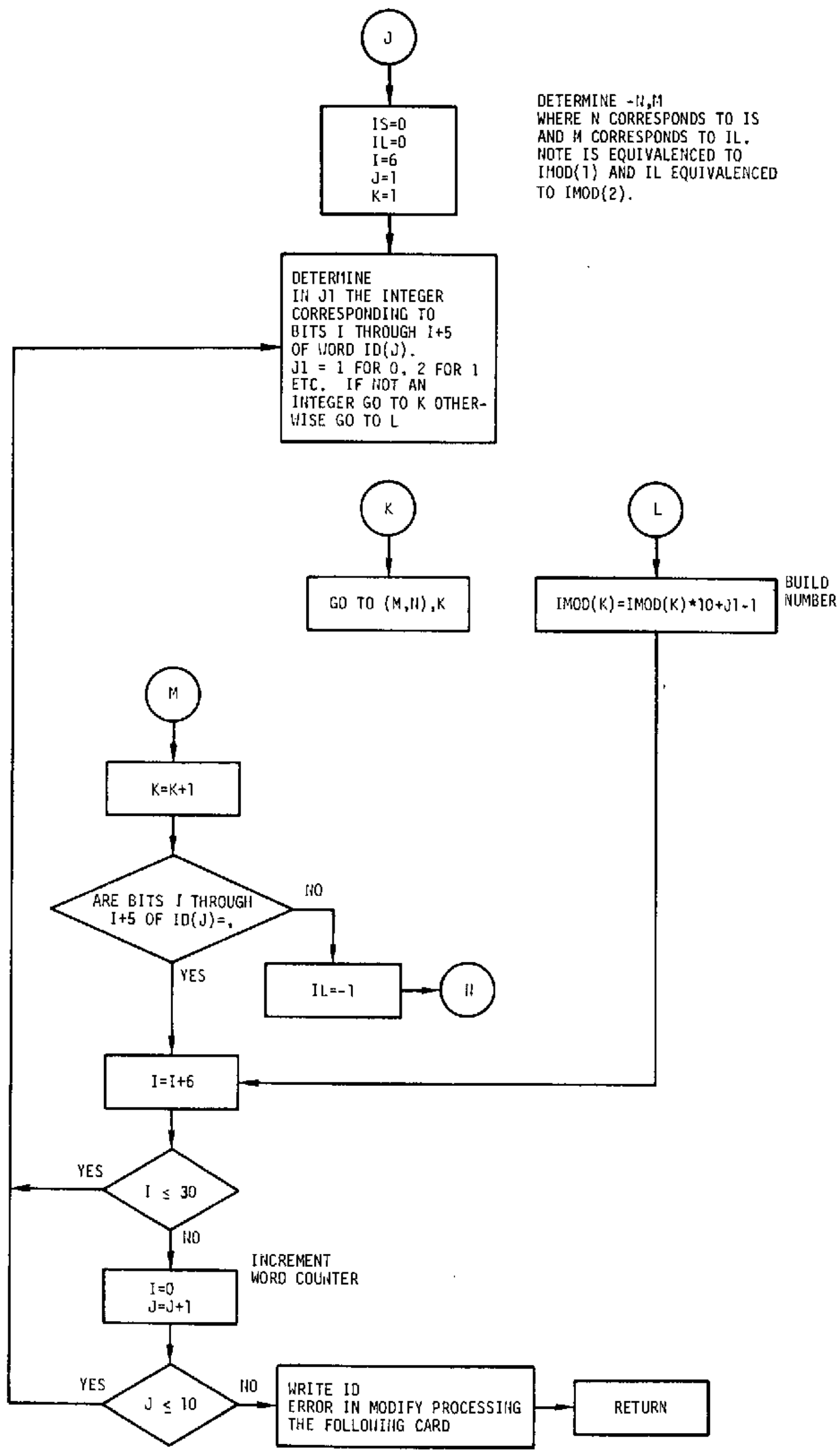

C
NBUFSZ -MUCH
≤ 0
> 0
WRITE
REQUIRED BUFFER SIZE "MUCH" IS LARGER THAN CURRENT BUFFER SIZE "NBUFSZ", SET TO CURRENT BUFFER SIZE
READ ON UNIT NEWSUN (BUF(I),I=1, MUCH
MUCH = NBUFSZ
ASSIGN D TO IGO
D
GO TO (E,G), IOPT
E
INCREMENT FROM END OF BUF TO START OF BUF
J2 = MUCH -10*ICUR
ICURM = ICUR + JMAX
J1 = J2-9
J1
<0
>0
RETURN
ICUR=ICUR+1
WRITE (BUF(J),J=J1,J2),ICUR
ICUR-ICURM
> 0
≤ 0
A
J2=J2-10


F
UPNEWS OPTION
WRITE
CHOOSE FROM INITAL, MODIFY, PRINT, AND QUIT
READ IP
INITAL
IP
QUIT OR PRINT
IP
PRINT
MODIFY
G
QUIT
RETURN
IWR≠0
YES
NO
WRITE
INPUT DATA IN FIRST 60 COLUMNS, TERMINATE WITH QUIT
IWR
≠0
=0
WRITE
OPTIONS INITIAL AND MODIFY MAY BE EXECUTED ONLY ONCE FOR EACH UPNEWS CALL
F
WRITE
OPTIONS INITIAL OR MODIFY MUST BE CALLED BEFORE PRINT
F
JCUR=NBUFSZ-9
J2=JCUR+9
READ(5)(BUF(J),J=JCUR,J2
= QUIT
BUF(JCUR)
JCUR=J2+1
≠QUIT
JCUR<1
YES
NO
JCUR=JCUR-10
WRITE
TOO MUCH DATA SOME OF THE LAST DATA NOT INCLUDED
REWIND NEWSUN
J=NBUFSZ-JCUR+1
H
WRITE (NEWSUN)J
WRITE(NEWSUN)(BUF(J),J=JCUR,NBUFSZ)
REWIND NEWSUN
IWR=1
F
I=NBUFSZ-9
K=1
J1=NBUFSZ-JCUR+1
J=1
BUF IS PRINTED FROM THE BOTTOM TO THE TOP
I1=J+9
WRITE (BUF(IZ),IZ=I,I1)K
K=K+1
I=I-10
J=J+10
≤0
J-J1
>0
F

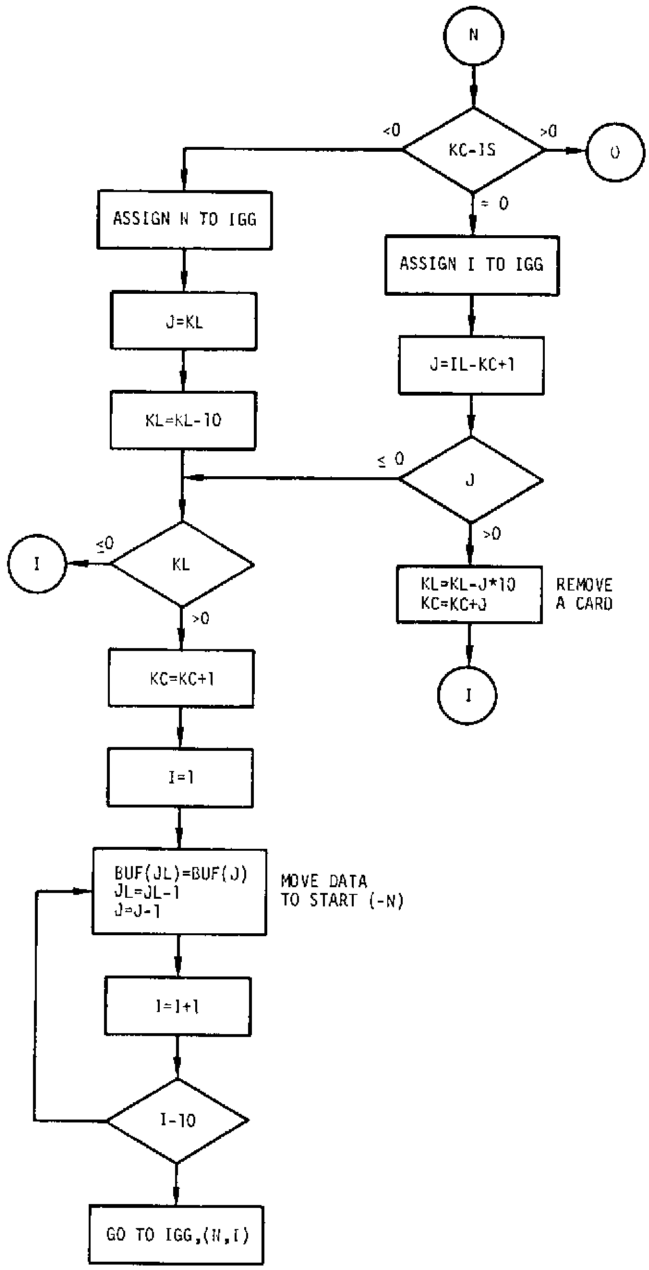
NEWS 3 of 6


G
JL=NBUFSZ
WRITE
MODIFICATIONS TO NEWS
MUST BE IN -N,M FORMAT.
INITIAL DATA FOLLOW-0
OR MODIFY CARD. MODIFICATIONS
TERMINATED BY QUIT CARD.
KL=MUCH
KC=1
IS = 0
IL = -1
I
READ (ID(I),I=1,10)
ID(1)
≠ QUIT
QUIT
FIRST COLUMN
GO TO DECODE -N,M
J
≠ -
KL=0
YES
NO
JL=JL-10
0
J=KL
JL+1-KL
≥0
<0
JL<D
YES
NO
COMPLETE THE FILE
J1=1
WRITE
WARNING ****
CARD DROPPED FROM
NEWS FILE
J=1
ADD
NEW CARD
K=JL+J
BUF(K)=ID(J)
BUF(JL)=BUF(J)
JL=JL-1
J=J-1
REMOVE
OLDEST CARD
KL=KL-10
J=J+1
J1=J1+1
J=2
≤0
J-10
>0
J1-KL
≥0
BUF(J-1)=BUF(J+9)
J=J+1
JCUR=JL+1
≤0
J-KL
>0
H
WRITE
BUFFER FULL, ENTER
QUIT
SKIP DOWN
TO QUIT
READ ID(1)
≠ QUIT
H
ID(1)
= QUIT

NEWS 4 of 6


J
IS=0
IL=0
I=6
J=1
K=1
DETERMINE -N,M
WHERE N CORRESPONDS TO IS
AND M CORRESPONDS TO IL.
NOTE IS EQUIVALENCED TO
IMOD(1) AND IL EQUIVALENCED
TO IMOD(2).
DETERMINE
IN J1 THE INTEGER
CORRESPONDING TO
BITS I THROUGH I+5
OF WORD ID(J).
J1 = 1 FOR 0, 2 FOR 1
ETC. IF NOT AN
INTEGER GO TO K OTHER-
WISE GO TO L
K
GO TO (M,N),K
L
IMOD(K)=IMOD(K)*10+J1-1
BUILD
NUMBER
M
K=K+1
ARE BITS I THROUGH
I+5 OF ID(J)=,
NO
IL=-1
N
YES
I=I+6
YES
I ≤ 30
NO
INCREMENT
WORD COUNTER
I=0
J=J+1
YES
J ≤ 10
NO
WRITE ID
ERROR IN MODIFY PROCESSING
THE FOLLOWING CARD
RETURN

NEWS 5 of 6


N
KC-IS
<0
>0
= 0
O
ASSIGN N TO IGG
ASSIGN I TO IGG
J=KL
J=IL-KC+1
KL=KL-10
≤ 0
J
>0
KL=KL-J*10
KC=KC+J
REMOVE
A CARD
I
≤0
KL
>0
I
KC=KC+1
I=1
BUF(JL)=BUF(J)
JL=JL-1
J=J-1
MOVE DATA
TO START (-N)
I=I+1
I-10
GO TO IGG,(N,I)

NEWS 6 of 6

Using the QUANTZ Option

Upon entering QUANTZ the user must define values for the parameters

XMIN XMAX	=	minimum and maximum intensity levels of interest in the single channel to be used
NQ	=	number of equal divisions desired between XMIN and XMAX
KCH	=	the channel component of the data to be considered, for example if channels 1,6,9, and 12 have been extracted via DATDEF and KCH = 3 then channel 9 will be used in the quantization process

The default values for these parameters are

XMIN = 0.

XMAX = 255.

NQ = 8

KCH = 1

At the completion of the image generation process QUANTZ displays the largest and smallest samples encountered in the data. Upon exiting QUANTZ a grey scale character map is available for display by IMAGES, the upper and lower bounds for each class are included.

QUANTZ OPTION

SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK

>QUANTZ


                QUANTZ  OPTION
                ==============


$INQUAN   XMIN,XMAX,NQ,KCH
$INQUAN
XMIN     =       .00000000E+00
XMAX     =       .25500000E+03
NQ       =                 +16
KCH      =                  +2

$END
TYPE YES IF INPUTS OK
>YES
LARGEST VALUE = 144.0   SMALLEST VALUE =  44.0

CLASS  SYMBOL  SIZE  LW. BND.   UP. BND.
  1      A       0       .00       .00
  2      B       0       .00     15.94
  3      C       0     15.94     31.87
  4      D       9     31.87     47.81
  5      E    1721     47.81     63.75
  6      F     980     63.75     79.69
  7      G     142     79.69     95.62
  8      H     172     95.62    111.56
  9      I      40    111.56    127.50
 10      J      11    127.50    143.44
 11      K       1    143.44    159.37
 12      L       0    159.37    175.31
 13      M       0    175.31    191.25
 14      N       0    191.25    207.19
 15      O       0    207.19    223.12
 16      P       0    223.12    239.06
 17      Q       0    239.06    255.00
 18      R       0    255.00
 THE OPTION QUANTZ REQUIRED        .5488 SECONDS OF CPU TIME.

 -----------------------------------------------------------
```

QUANTZ ENGINEERING DESCRIPTION

QUANTZ generates an alphabetic image array or map via a quantization procedure on one channel of the data. The size of the quantization cells, their number, and range are specified by the user.

Define

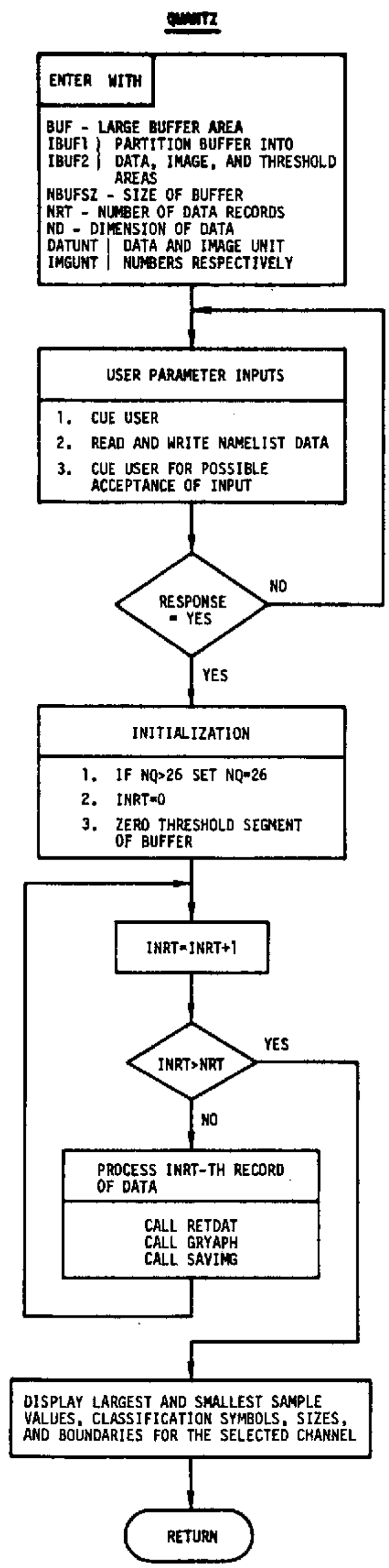
$$\left.\begin{matrix} X_i \\ \\ X_{i+1} \end{matrix}\right\} \text{ boundaries of } i^{th} \text{ cell}$$

v sample from the channel of data to be quantized

Then the quantization character, in the image array, assigned to v is

$X_i \leq v < X_{i+1}$	character code i (with 1=B, 2=C, etc.)
$v < X_1 = XMIN$	character A
$v \geq X_{LAST} = XMAX$	the "NQ+2"nd letter of the alphabet

The distance values for the image display of the image generated by QUANTZ are all zero.


QUANTZ
ENTER WITH
BUF - LARGE BUFFER AREA
IBUF1 } PARTITION BUFFER INTO
IBUF2 } DATA, IMAGE, AND THRESHOLD AREAS
NBUFSZ - SIZE OF BUFFER
NRT - NUMBER OF DATA RECORDS
ND - DIMENSION OF DATA
DATUNT | DATA AND IMAGE UNIT
IMGUNT | NUMBERS RESPECTIVELY
USER PARAMETER INPUTS
1. CUE USER
2. READ AND WRITE NAMELIST DATA
3. CUE USER FOR POSSIBLE ACCEPTANCE OF INPUT
RESPONSE = YES
NO
YES
INITIALIZATION
1. IF NQ>26 SET NQ=26
2. INRT=0
3. ZERO THRESHOLD SEGMENT OF BUFFER
INRT=INRT+1
INRT>NRT
YES
NO
PROCESS INRT-TH RECORD OF DATA
CALL RETDAT
CALL GRYAPH
CALL SAVIMG
DISPLAY LARGEST AND SMALLEST SAMPLE VALUES, CLASSIFICATION SYMBOLS, SIZES, AND BOUNDARIES FOR THE SELECTED CHANNEL
RETURN

QUANTZ 1 OF 1

GRYAPH

ENTER WITH

X - DATA VECTOR ARRAY
ND - DIMENSION OF EACH VECTOR IN X
NX - NUMBER OF VECTORS IN X
XMIN,XMAX,DX - DEFINE LIMITS AND QUANTIZATION INCREMENT OVER WHICH DATA WILL BE QUANTIZED
KCH - CHANNEL POSITION IN EACH VECTOR OF X TO BE QUANTIZED
XS,XL - PREVIOUS VALUES OF SMALLEST AND LARGEST SAMPLE

```
EPS=1.E-7
```

GENERATE GRAY SCALE IMAGE (ALPHABETIC CHARACTER) IMAGE VIA QUANTIZATION OF KCH-TH CHANNEL POSITION

```
      DO 1040 J=1,NX
      K=(J-1)*ND+KCH
      XK=X(K)
      IF(XK.LT.XS) XS=XK
      IF(XK.GT.XL) XL=XK
      IF (XK.LT.XMIN) GO TO 1000
      IF (XK.GT.XMAX) GO TO 1020
      I=INT((XK-XMIN-EPS)/DX)+2
      IMG(J)=I
      NVG(I) = NVG(I) + 1
      GO TO 1040
 1000 IMG(J)=1
      NVG(1) = NVG(1) + 1
      GO TO 1040
 1020 IMG(J) = NVM
      NVG( NVM ) = NVG( NVM ) + 1
 1040 CONTINUE
```

EXIT WITH

XS,XL - UPDATED VALUES
IMG - DEFINES IMAGE

Using the TRNFLD Option

The TRNFLD option is used to classify data in a way that makes the computation of statistics for training fields convenient. Each pixel is given a class number equal to the number of the field containing the pixel. With three fields, for example, all pixels in field 1 are put into class 1, all pixels in field 2 are put into class 2, and all pixels in field 3 are put into class 3. This makes it possible to compute statistics in FACTOR for each field by asking first for A, then for B, and finally for C.

To use the TRNFLD option it is necessary to input the option name only. No other card inputs are required.

A data file must be available on DATUNT for TRNFLD to read, and file must be assigned on IMGUNT for TRNFLD to write.

Printer output from TRNFLD shows the class numbers assigned to each field, the corresponding class symbols, and the number of pixels in each class.

TRNFLD OPTION

SAMPLE INPUT AND CORRESPONDING OUTPUT:

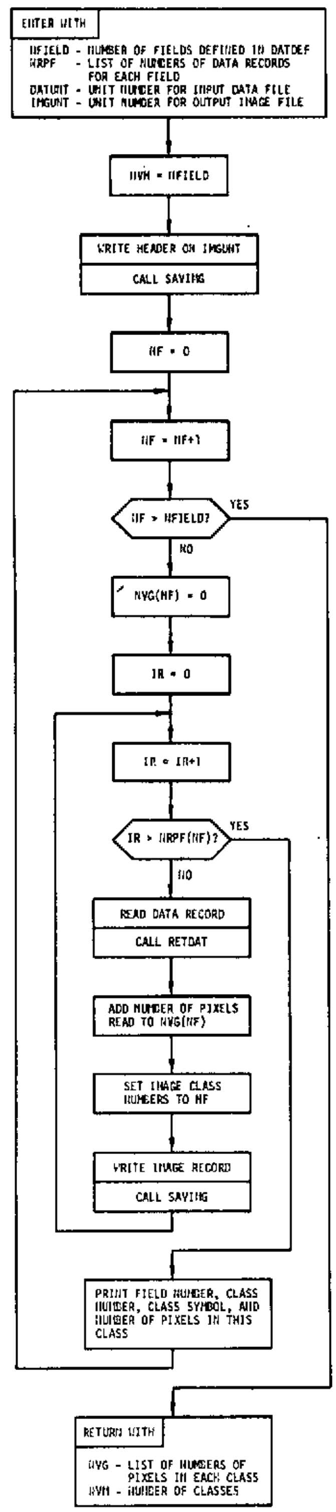
```
ENTER ASTEP OPTION OR TYPE A BLANK
>TRNFLD

                    TRNFLD  OPTION
                    ==============


          FIELD      CLASS      SYMBOL      PIXELS
            1          1          A           105
            2          2          B           207
            3          3          C           133
            4          4          D            77
            5          5          E            65
            6          6          F            56
            7          7          G           195
            8          8          I            81
            9          9          J           119
           10         10          K           140
 THE OPTION TRNFLD REQUIRED         .3812 SECONDS OF CPU TIME.

 ---------------------------------------------------------------
```

TRNFLD ENGINEERING DESCRIPTION

TRNFLD reads each data record on the input file DATUNT and writes a corresponding image record on the output file IMGUNT. The class number written on an image record is equal to the field number for that record.

The TRNFLD subroutine uses the RETDAT and SAVIMG subroutines.

TRNFLD



ENTER WITH
NFIELD - NUMBER OF FIELDS DEFINED IN DATDEF
NRPF - LIST OF NUMBERS OF DATA RECORDS FOR EACH FIELD
DATUNT - UNIT NUMBER FOR INPUT DATA FILE
IMGUNT - UNIT NUMBER FOR OUTPUT IMAGE FILE
NVM = NFIELD
WRITE HEADER ON IMGUNT
CALL SAVING
NF = 0
NF = NF+1
NF > NFIELD?
YES
NO
NVG(NF) = 0
IR = 0
IR = IR+1
IR > NRPF(NF)?
YES
NO
READ DATA RECORD
CALL RETDAT
ADD NUMBER OF PIXELS READ TO NVG(NF)
SET IMAGE CLASS NUMBERS TO NF
WRITE IMAGE RECORD
CALL SAVING
PRINT FIELD NUMBER, CLASS NUMBER, CLASS SYMBOL, AND NUMBER OF PIXELS IN THIS CLASS
RETURN WITH
NVG - LIST OF NUMBERS OF PIXELS IN EACH CLASS
NVM - NUMBER OF CLASSES

Using the TRNSFM Option

The TRNSFM option is used to scale and offset the data vectors or to perform a linear transformation on them, or both. The two suboptions available are SCALE and TRANS. The first input to TRNSFM is the desired suboption name.

If SCALE is chosen, then the namelist $INTRNS must be input. The variables are:

DATUNT - unit number for input data
NEWDAT - unit number for output data (NEWDAT replaces DATUNT)
A - offset constant for each channel
B - scaling constant for each channel

As with all namelist inputs to ASTEP, the values are printed for inspection. Either YES may be input to continue the execution of the program, or NO may be input to correct the values.

If TRANS is chosen, then the same namelist must be input, and additional inputs are required to retrieve a transformation matrix from a signature file. These inputs are:

- File number to search for matrix
- YES to print matrix or NO not to print
- Name of matrix
- NOMORE to stop searching file
- 0 to leave REDSIG

After the transformation is complete, the value of DATUNT is set to the value of NEWDAT and the channel numbers and number of channels are changed to enable other ASTEP options to use the transformed data as input. Control is then returned to the ASTEP driver.

TRNSFM OPTION

SAMPLE INPUT AND CORRESPONDING OUTPUT

```
ENTER ASTEP OPTION OR TYPE A BLANK
>TRNSFM

                    TRNSFM  OPTION
                    ==============

 CHOOSE FROM SCALE OR TRANS.
>TRANS

 $INTRNS  DATUNT,NEWDAT,A,B
$INTRNS
DATUNT  =                 +4
NEWDAT  =                +20
A       =      .00000000E+00
B       =      .10000000E+01

$END

 TYPE YES IF INPUTS ARE CORRECT.
>YES

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>2

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  2
>YES

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>BMATX2

BMATX2 ND = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =      4

               MEAN        1 BY 12
                ALL ZEROES.
```

COVMAT 12 BY 12

	1	2	3	4	5	6
1	.000	.000	1.000	.000	.000	.000
2	.000	.000	.000	.000	.000	.000
3	.000	.000	.000	.000	.000	.000
4	.000	.000	.000	.000	.000	.000
5	.000	.000	.000	.000	.000	.000
6	.000	.000	.000	.000	.000	.000
7	.000	.000	.000	.000	.000	.000
8	1.000	.000	.000	.000	.000	.000
9	.000	.000	.000	1.000	.000	.000
10	.000	.000	.000	.000	.000	.000
11	.000	.000	.000	.000	.000	.000
12	.000	1.000	.000	.000	.000	.000

	7	8	9	10	11	12
1	.000	.000	.000	.000	.000	.000
2	.000	.000	.000	.000	.000	.000
3	.000	.000	.000	.000	.000	.000
4	.000	.000	.000	.000	.000	.000
5	.000	.000	.000	.000	.000	.000
6	.000	.000	.000	.000	.000	.000
7	.000	.000	.000	.000	.000	.000
8	.000	.000	.000	.000	.000	.000
9	.000	.000	.000	.000	.000	.000
10	.000	.000	.000	.000	.000	.000
11	.000	.000	.000	.000	.000	.000
12	.000	.000	.000	.000	.000	.000

```
>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
  1 SIGNATURES HAVE BEEN RETRIEVED
WRITE 2788 WORDS ON UNIT 20
WRITE 2788 WORDS ON UNIT 20
WRITE 1148 WORDS ON UNIT 20

 THE OPTION TRNSFM REQUIRED      5.6206 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

TRNSFM ENGINEERING DESCRIPTION

The TRNSFM option applies a linear transformation to each of the data vectors and creates a new data file. The general transformation is obtained in the TRANS suboption and can be considered in two steps:

1) Offset and scale the data

$$U = A+BV$$

where

V - an input data vector

A - an offsetting (or translating) vector with the same value in all components

B - a diagonal scaling matrix with all diagonal entries equal

U - the resulting offset and scaled data vector

2) Transform the resulting vector with a matrix

$$W = TU \quad \text{or} \quad W = TA+TBV$$

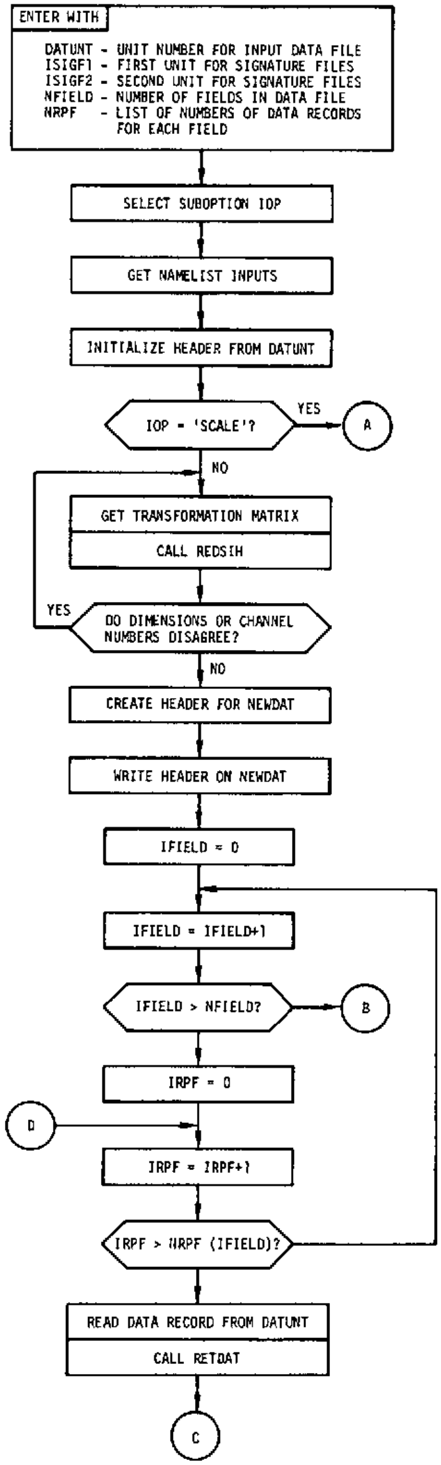
where

T - a given transformation matrix which may reduce the dimension of the data

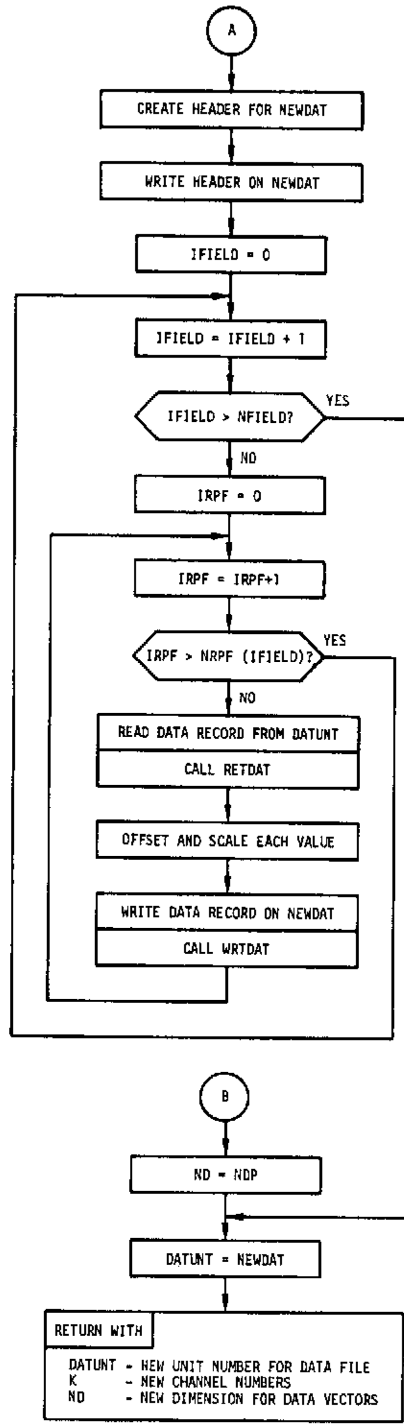
W - final data vector to be output on the new data file

If the second step only is required, then the TRANS suboption is used and A is set to zero and B is set to an identity. If the first step only is required, then the SCALE option is used.

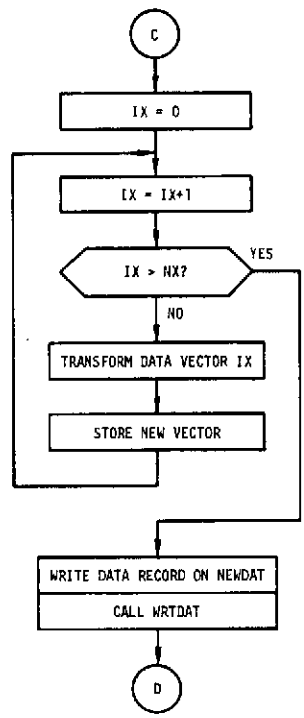
TRNSFM



TRNSFM 1 of 3


A
CREATE HEADER FOR NEWDAT
WRITE HEADER ON NEWDAT
IFIELD = 0
IFIELD = IFIELD + 1
IFIELD > NFIELD?
YES
NO
IRPF = 0
IRPF = IRPF+1
IRPF > NRPF (IFIELD)?
YES
NO
READ DATA RECORD FROM DATUNT
CALL RETDAT
OFFSET AND SCALE EACH VALUE
WRITE DATA RECORD ON NEWDAT
CALL WRTDAT
B
ND = NDP
DATUNT = NEWDAT
RETURN WITH
DATUNT - NEW UNIT NUMBER FOR DATA FILE
K - NEW CHANNEL NUMBERS
ND - NEW DIMENSION FOR DATA VECTORS

TRNSFM 2 of 3


C
IX = 0
IX = IX+1
IX > NX?
YES
NO
TRANSFORM DATA VECTOR IX
STORE NEW VECTOR
WRITE DATA RECORD ON NEWDAT
CALL WRTDAT
D

TRNSFM 3 of 3

Using the UNITS Option

The purpose of the UNITS option is to allow the user to display, set, and manipulate the unit numbers used by ASTEP. Upon entering UNITS, the available image unit numbers (AVIMUN) are displayed. Next, the current unit assignments (INUNIT) are displayed. The unit names and definitions are:

IMGUNT	-	image unit, used by all options which require an input image or generate an output image as the unit number (with the exception of the inputs to DIFIMG as noted in the example)
DATUNT	-	data unit used by all options which read the reformatted data tape, created by DATDEF, as the unit number
OBSUNT	-	observation unit, DATDEF reads the raw packed observation data tape from this unit
ISIGF1	-	signature file 1 unit, the unit number of the first signature file
ISIGF2	-	signature file 2 unit, the unit number of the second signature file
IHISF1, IHISF2	-	histogram files 1 and 2, temporary scratch units used by the HSGRAM option
IMG1, IMG2	-	image 1 and 2 units, used by DIFIMG as the unit numbers of the first and second input images. UNITS/CYCLE used as in the example automatically sets up these values and the values for IMGUNT correctly
OBS1	-	observation unit 1, used by the CPYDAT option of DATDEF as the output unit for copying a subset of raw packed observation data tape

The user must then select one of the suboptions:

CYCLE

CHANGE

QUIT

The CYCLE suboption is a convenience device to be used primarily in conjunction with the DIFIMG option. Whenever an image is generated by any of the image generating options in ASTEP, the image is stored on unit IMGUNT.

There are three available image unit numbers which may be denoted as N1, N2, and N3. These have values of 3, 12, and 9 respectively. These values are set in the ASTEP driver and cannot be changed by user inputs. The CYCLE suboption causes IMGUNT to be cycled from its current value to the next available image unit number. For example if IMGUNT = N2, selection of CYCLE will result in IMGUNT = N3.

The units IMG1 and IMG2 are the input image units for the DIFIMG option. The DIFIMG option computes an image of the differences between the images on IMG1 and IMG2 and stores the image on IMGUNT. The CYCLE suboption does not cycle IMG1 and IMG2, but sets them to N2 and N3 respectively on the second call to CYCLE. They are then set to N2 and N3 on every third call to CYCLE thereafter.

All of the options in ASTEP will assume that the image of interest is currently on or will be next saved on the unit numbers

N1 - nominal value, used prior to any entry to UNITS with selection of CYCLE

N2 - value resulting after first entry to UNITS with selection of CYCLE

N3 - value resulting after second entry to UNITS with selection of CYCLE

N1 - value resulting after third entry to UNITS with selection of CYCLE

N2 - etc.

An example of the use of CYCLE in conjunction with DIFIMG is

ADPCLU (inputs to ADPCLU)	generates first image on unit IMGUNT where IMGUNT = N1
IMAGES (inputs to IMAGE)	display of first image - saved on unit IMGUNT (=N1)
UNITS CYCLE	cycles IMGUNT such that IMGUNT = N2
MAXLIK (inputs to MAXLIK)	generates second image on unit IMGUNT

IMAGES (inputs to IMAGE)	display of second image - saved on unit IMGUNT (=N2)
UNITS CYCLE	cycles IMGUNT such that IMGUNT = N3 and sets IMG1 = N1, IMG2 = N2
DIFIMG (inputs to DIFIMG)	differences the images stored on IMG1 and IMG2 and generates a third image saved on unit IMGUNT
IMAGES	display of third image - saved on unit IMGUNT (=N3)
UNITS CYCLE	cycles IMGUNT such that IMGUNT = N1 for the next image to be generated

In this example any of the image generating options could be used in place of ADPCLU or MAXLIK. After the last UNITS - CYCLE, the next image generated will override the first image saved on unit N1.

The CHANGE suboption allows the user to specify the unit numbers for any of the units used by the program and contained in the INUNIT array displayed by the UNITS option. However, it should be noted that the CYCLE suboption, if used after the CHANGE suboption, will override any values for IMGUNT, IMG1, and IMG2 specified previously in CHANGE. This occurs because, as noted previously, the user cannot change the values of N1, N2, and N3 and the CYCLE suboption cycles IMGUNT through N1, N2, and N3 and sets IMG1 and IMG2 to values of N1 and N2 as previously described.

QUIT returns control to ASTEP. The selection of UNITS with QUIT is only used to examine the unit assignments without cycling (CYCLE) or changing (CHANGE) them. CYCLE and CHANGE upon completion return control to ASTEP.

UNITS OPTION
SAMPLE 1 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK

>UNITS

                         UNITS   OPTION
                         ===============

$AVIMUN
N       =               +3,               +12,              +9

$END
$INUNIT
IMGUNT  =               +9
DATUNT  =               +4
OBSUNT  =               +7
ISIGF1  =               +1
ISIGF2  =               +2
IHISF1  =               +10
IHISF2  =               +11
IMG1    =               +3
IMG2    =               +12
OBS1    =               +8

$END
CHOOSE OPTION FROM
  CYCLE    CHANGE   QUIT
>CYCLE
$INUNIT
IMGUNT  =               +3
DATUNT  =               +4
OBSUNT  =               +7
ISIGF1  =               +1
ISIGF2  =               +2
IHISF1  =               +10
IHISF2  =               +11
IMG1    =               +3
IMG2    =               +12
OBS1    =               +8

$END
 THE OPTION UNITS  REQUIRED        .0544 SECONDS OF CPU TIME.

 ----------------------------------------------------------------
```

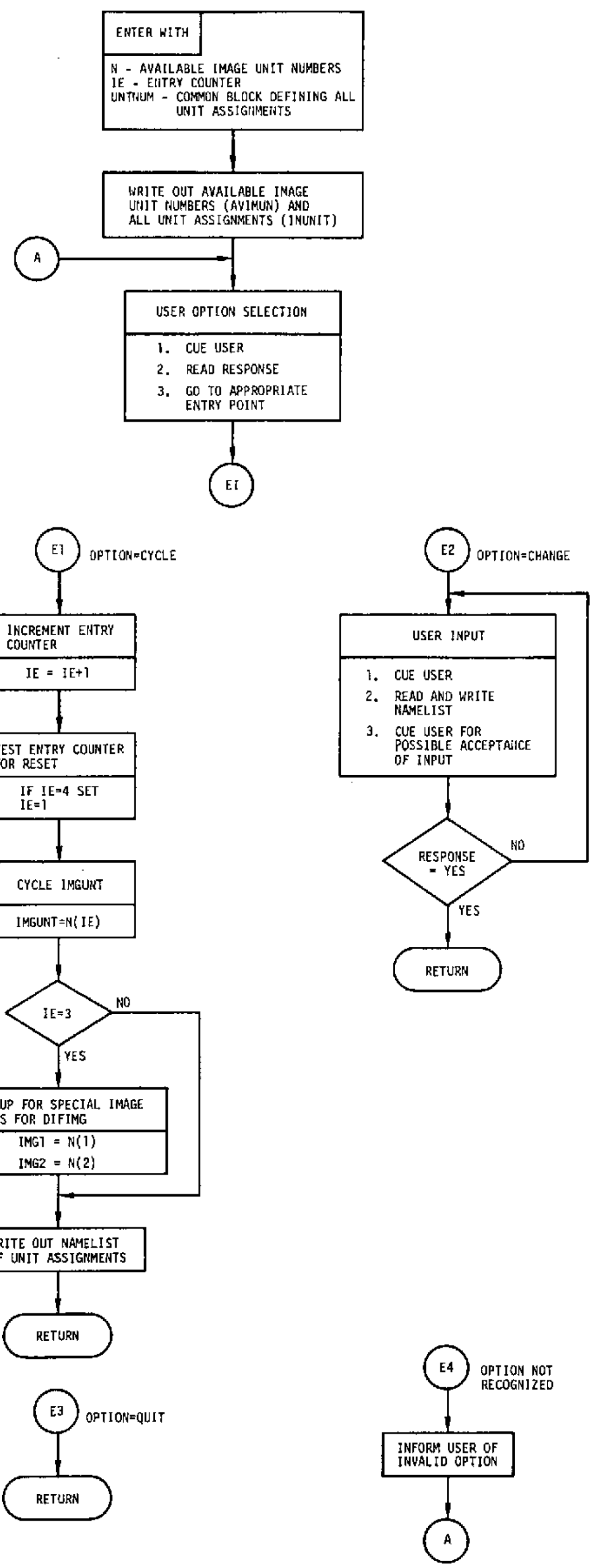
UNITS OPTION
SAMPLE 2 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>UNITS

                    UNITS   OPTION
                    **************

$AVIMUN
N          =             +3,               +12,                +9

$END
$INUNIT
IMGUNT     =             +9
DATUNT     =             +4
OBSUNT     =             +7
ISIGF1     =             +1
ISIGF2     =             +2
IHISF1     =             +10
IHISF2     =             +11
IMG1       =             +3
IMG2       =             +12
OBS1       =             +8

$END
CHOOSE OPTION FROM
  CYCLE   CHANGE   QUIT
>CHANGE
$INUNIT   IMGUNT,DATUNT,OBSUNT,ISIGF1,ISIGF2,IHISF1,IHISF2,IMG1,IMG2,OBS1
$INUNIT
IMGUNT     =             +12
DATUNT     =             +4
OBSUNT     =             +7
ISIGF1     =             +1
ISIGF2     =             +2
IHISF1     =             +10
IHISF2     =             +11
IMG1       =             +3
IMG2       =             +12
OBS1       =             +8

$END
TYPE YES IF INPUTS OK
>YES
 THE OPTION UNITS  REQUIRED       .0952 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

UNITS ENGINEERING DESCRIPTION

UNITS does not require an engineering description - see flow chart.

UNITS


ENTER WITH
N - AVAILABLE IMAGE UNIT NUMBERS
IE - ENTRY COUNTER
UNTNUM - COMMON BLOCK DEFINING ALL UNIT ASSIGNMENTS
WRITE OUT AVAILABLE IMAGE UNIT NUMBERS (AVIMUN) AND ALL UNIT ASSIGNMENTS (INUNIT)
A
USER OPTION SELECTION
1. CUE USER
2. READ RESPONSE
3. GO TO APPROPRIATE ENTRY POINT
EI
E1
OPTION=CYCLE
INCREMENT ENTRY COUNTER
IE = IE+1
TEST ENTRY COUNTER FOR RESET
IF IE=4 SET IE=1
CYCLE IMGUNT
IMGUNT=N(IE)
IE=3
NO
YES
SET UP FOR SPECIAL IMAGE UNITS FOR DIFIMG
IMG1 = N(1)
IMG2 = N(2)
WRITE OUT NAMELIST OF UNIT ASSIGNMENTS
RETURN
E2
OPTION=CHANGE
USER INPUT
1. CUE USER
2. READ AND WRITE NAMELIST
3. CUE USER FOR POSSIBLE ACCEPTANCE OF INPUT
RESPONSE = YES
NO
YES
RETURN
E3
OPTION=QUIT
RETURN
E4
OPTION NOT RECOGNIZED
INFORM USER OF INVALID OPTION
A

Using the UPNEWS Option

The UPNEWS option is used to create or modify the NEWS file (file name NEWS, qualifier TRW-T33710 mounted on unit 15). This option should be used only by ASTEP developers since the file is permanently changed.

After UPNEWS is requested, the user is asked for INITAL, MODIFY, PRINT, or QUIT. If the user responds INITAL, the data to be stored is input next using the first 60 columns. The data is terminated with a QUIT beginning in column 1. The user then is asked to input INITAL, MODIFY, PRINT or QUIT. INITAL and/or MODIFY may be executed only once for each call to UPNEWS.

If MODIFY is executed, the user inputs modifications in -N,M format similar to standard symbolic card modifications. Initial cards may be input immediately after the MODIFY card or after -0. The input example should clarify the -N,M format. The alters must be made in sequential order. The MODIFY option is terminated with a QUIT card and the user is again asked for INITAL, MODIFY, PRINT or QUIT.

For the INITAL and MODIFY options the buffer size is checked to assure that the user does not overflow the buffer.

If the user responds PRINT, the user must have executed MODIFY or INITAL first. The PRINT option prints the entire file with the sequence numbers and then asks for INITAL, MODIFY, PRINT or QUIT. If the user responds QUIT, the UPNEWS option is complete. Two samples of the UPNEWS option are presented to illustrate its use. In sample 1, the INITAL option is used to create a news file. The generation of the file is completed with the option QUIT. After the file has been generated, it is printed using the print option. In sample 2, the MODIFY option is selected and a card is added to the beginning of the file, the second card is replaced, a card is added after the third card, and the fifth and sixth cards are deleted. The PRINT option is then selected to print the results and the sample is ended with the QUIT option.

UPNEWS OPTION

SAMPLE 1 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>UPNEWS

                    UPNEWS  OPTION
                    ==============

UPNEWS OPTION REQUIRES FILE NAME NEWS TO BE ASSIGNED
IF NOT ASSIGNED INDICATE QUIT,
QUIT FROM ASTEP, ASSIGN NEWS, AND RE-EXECUTE ASTEP.
CHOOSE FROM INITAL, MODIFY, PRINT, AND QUIT
>INITAL
INPUT DATA IN FIRST 60 COLUMNS, TERMINATE WITH QUIT
>  THIS IS AN EXAMPLE OF A NEWS FILE.
>  THIS LINE WILL BE REPLACED.
>  ADD A LINE AFTER THIS ONE,
>  AND DELETE THE 2 LINES FOLLOWING THIS ONE.
>  TO BE DELETED.
>  TO BE DELETED.
>  THE QUIT CARD TERMINATES THIS NEWS FILE INPUT.
>QUIT
CHOOSE FROM INITAL, MODIFY, PRINT, AND QUIT
>PRINT
  THIS IS AN EXAMPLE OF A NEWS FILE.                         1
  THIS LINE WILL BE REPLACED.                                2
  ADD A LINE AFTER THIS ONE,                                 3
  AND DELETE THE 2 LINES FOLLOWING THIS ONE.                 4
  TO BE DELETED.                                             5
  TO BE DELETED.                                             6
  THE QUIT CARD TERMINATES THIS NEWS FILE INPUT.             7
CHOOSE FROM INITAL, MODIFY, PRINT, AND QUIT
>QUIT
 THE OPTION UPNEWS REQUIRED       .2430 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

UPNEWS OPTION

SAMPLE 2 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>UPNEWS

                    UPNEWS  OPTION
                    **************

UPNEWS OPTION REQUIRES FILE NAME NEWS TO BE ASSIGNED
IF NOT ASSIGNED INDICATE QUIT,
QUIT FROM ASTEP, ASSIGN NEWS, AND RE-EXECUTE ASTEP.
CHOOSE FROM INITAL, MODIFY, PRINT, AND QUIT
>MODIFY
MODIFICATIONS TO NEWS MUST BE IN -N,M FORMAT.
INITIAL DATA FOLLOW -0 OR MODIFY CARD.
MODIFICATIONS TERMINATED BY QUIT CARD.
>  THIS CARD WILL APPEAR AT THE BEGINNING OF THE NEWS FILE.
>-2,2
>  LINE 2 HAS BEEN REPLACED.
>-3
>  THIS CARD IS A SAMPLE ADDITION.
>-5,6
>QUIT
CHOOSE FROM INITAL, MODIFY, PRINT, AND QUIT
>PRINT
  THIS CARD WILL APPEAR AT THE BEGINNING OF THE NEWS FILE.    1
  THIS IS AN EXAMPLE OF A NEWS FILE.                          2
  LINE 2 HAS BEEN REPLACED.                                   3
  ADD A LINE AFTER THIS ONE.                                  4
  THIS CARD IS A SAMPLE ADDITION.                             5
  AND DELETE THE 2 LINES FOLLOWING THIS ONE.                  6
  THE QUIT CARD TERMINATES THIS NEWS FILE INPUT.              7
CHOOSE FROM INITAL, MODIFY, PRINT, AND QUIT
>QUIT
 THE OPTION UPNEWS REQUIRED       .2380 SECONDS OF CPU TIME.

 -----------------------------------------------------------
```

UPNEWS ENGINEERING DESCRIPTION

UPNEWS does not require an engineering description - see the <u>NEWS</u> flow diagram.

Using the UVWRIT Option

Upon entering the UVWRIT option the user must select one of the sub-options MSSDAT or IMAGE. Following this selection user is asked if he wants to have the universal output records printed before they are output onto a magnetic tape. User enters YES or NO and this completes the user controlled input. Input and output unit numbers are set internally, but they may be changed by the UNITS option.

UVWRIT OPTION

SAMPLE 1 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>UVWRIT

                    UVWRIT  OPTION
                    **************

CHOOSE OPTION FROM
  MSSDAT  IMAGE
>MSSDAT

MSSDAT OPTION HAS BEEN SELECTED
PRINT UNIVERSAL FORMAT OUTPUT - YES OR NO
>YES
PRINT UNIVERSAL OUTPUT SELECTED
```

```
                    F I E L D   N U M B E R   1
                    ---------------------------

NUMBER OF SUBFRAME STATUS BITS BEING USED IS      0


                         UNIVERSAL HEADER RECORD
                         -----------------------

START   NO OF
BYTE    BYTES    DESCRIPTION

1        32      COMPUTING SYSTEM ID                *
33       20      TAPE LIBRARY ID                    *
53       8       SENSOR ID                          *5192
61       3       DATE OF TAPE GENERATION  DD MM YY  *    0    0    0
64       1       TAPE SEQUENCE ID                   *         0
65       2       MISSION NUMBER                     *         0
67       2       SITE                               *         0
69       1       LINE                               *         0
70       1       RUN                                *         0
71       2       ORBIT                              *         0
73       8       TIME OF FIRST SCAN  DD MM YY       *    0    0    0
                                     HH MM SS .HS   *    0    0    0    0
81       8       CHANNELS ACTIVE                    *

1000010010010000000000000000000000000000000000000000000000000000

89       1       PROCESSING FLAG                    *         1
90       1       NO.CHANNELS ON THIS TAPE           *         4
91       1       NO. OF BITS/PIXEL                  *         8
92       2       FIRST VIDEO PIXEL WITHIN SCAN      *         1
94       2       FIRST CAL. ELEMENT WITHIN SCAN     *         0
96       2       NO. PIXELS/SCAN/CHANNEL            *        41
98       2       NO. CALIB./SCAN/CHANNEL  (AREA 1)  *         0
100      2       PHYSICAL RECORD SIZE IN BYTES      *      2080
102      1       NO. CHANNELS/PHYSICAL RECORD       *         0
103      1       PHYS. RECORDS/SCAN/CHANNEL         *         0
104      1       NO. RECORDS TO A DATA SET          *         1
105      2       ANCILLARY DATA LENGTH IN BYTES     *        95
107      1       DATA ORDER INDICATOR               *         0
108      2       START PIXEL NUMBER                 *        10
110      2       STOP PIXEL NUMBER                  *        90
112      512     COEFFS.EXPS. A0-E0-A1-E1           *

         0          0          0          0
         0          0          0          0
         0          0          0          0
         0          0          0          0

624      64      COLOR CODE 0=NA,1=R,2=G,3=B        *
                                                              0
                                                              0



                                                              0
                                                              0

688      64      SCALE FACTOR                       *
                                                              0
                                                              0
                                                              0
                                                              0

752      1       OFFSET CONSTANT                    *         0
753      1       WORD SIZE OF GENERATING COMPUTER   *        36
1778     1       NO OF DATA SETS/PHYSICAL RECORD    *        11
1779     2       START OF SECOND CALIB. AREA        *         0
1781     2       NO. CAL. ELE. IN SECOND CAL. AREA  *         0
1783     1       FIRST CAL. SOURCE INDICATOR        *         0
1783     1       SECOND CAL. SOURCE INDICATOR       *         0
1784     1       FILL ZEROS                         *00000000
1785     2       NO. CHANNELS IN FIRST RECORD       *         4
1787     2       BYTES/SCAN/CHANNEL                 *        41
1789     2       PIXEL SKIP FACTOR                  *         2
1791     2       SCAN SKIP FACTOR                   *         2
2789     1       N THOUSAND SCAN LINES PER FRAME    *         0
2790     3       ALTITUDE IN METERS                 *         0
2793     2       GROUND SPEED IN METERS PER SECOND  *         0
2795     1       SCAN TYPE -LINEAR OR SMOOTHED      *         0
2796     1       ANGLE OF ARC IN DEGREES            *         0
2797     1       CAMERA   0 = 70 MM,  1 = 5 INCH    *         0
2798     1       INPUT DEV. 0 = 9TRK,  1 = HDT      *         0
2799     1       TRUNCATION  0=LO, 1=HI, 2=NO       *         0
2800     8       CHANNELS REQUESTED                 *
0000000000000000000000000000000000000000000000000000000000000000
2808     1       PROCESSING MODE  0=SER. 1=CONC.    *         0
2874     1       COLOR 0=NO COLOR,1=ASSIGN,2=FALSE  *         0
2875     1       IMAGE FORMAT                       *         0
2876     1       REPEAT OF PIXELS PER SCAN          *         0
2877     1       REPEAT OF SCAN                     *         0
2878     4       PARTIAL SCAN -  FROM PIXEL NO.     *         0
                              -  TO  PIXEL  NO.     *         0
2882     2       SCAN RATE IN SCANS PER SECOND      *         0
2885     2       ANGLE OF DRIFT                     *      .000
2889     1       PIXEL  SIZE                        *         0
```

```
OUTPUT UNIVERSAL RECORD
-----------------------
MSSDAT OPTION
NWORDS =      640
NO. OF DATA SETS (SCANS) =     11
TOTAL NO. OF BYTES IN THIS RECORD =      2851
```

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	81	81	82	85	83	84	82	83	86	85	86	83	85	84	82	82	82	80	83	80	76	76	77
120	75	78	75	74	75	76	76	74	74	74	75	73	74	74	72	72	78	74	78	82	80	81	81	81	82	83	87	84	84	84
150	84	84	82	86	84	85	86	84	81	81	82	81	83	82	81	82	84	82	83	83	82	83	81	82	82	83	85	82	83	87
180	59	67	57	60	57	58	62	62	60	64	61	61	61	59	59	62	62	74	56	57	57	56	56	55	56	55	58	58	55	58
210	59	58	58	58	55	55	56	57	57	57	81	79	77	72	71	76	74	77	76	79	78	80	79	85	85	79	84	86	77	107
240	110	106	108	113	117	112	122	116	109	111	106	107	106	108	110	116	123	122	123	121	122	0	0	0	0	0	0	0	0	0
270	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
330	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	82	83	83	83
360	85	86	85	85	87	82	85	84	82	85	84	83	83	82	89	76	75	74	74	75	73	74	74	74	74	73	74	74	73	74
390	74	73	71	71	73	72	71	82	85	84	86	86	87	86	86	87	85	88	88	88	88	86	88	87	93	87	82	83	80	82
420	82	82	82	84	82	86	83	82	86	83	82	81	82	81	82	81	83	82	57	60	58	59	60	58	61	61	62	60	61	63
450	60	60	60	64	60	71	63	54	56	55	54	56	55	55	67	55	57	57	56	64	57	56	58	55	55	65	54	53	54	84
480	76	79	80	78	80	73	75	77	81	82	83	90	88	83	83	82	89	90	106	111	113	114	116	117	122	122	121	113	122	119
510	97	108	116	116	111	122	122	118	120	119	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
540	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
570	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0
600	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	81	85	85	87	86	88	86	88	86	89	89	88	87	87	85
630	83	87	82	89	80	76	78	77	75	76	76	74	75	75	76	76	79	76	74	74	73	74	72	73	73	75	82	86	86	89
660	85	86	86	86	85	86	88	88	87	88	89	86	89	91	86	82	83	83	82	83	83	82	83	84	84	84	84	84	82	82
690	84	83	83	84	82	82	82	59	60	60	62	61	69	60	61	60	64	63	62	63	59	63	60	60	71	58	57	67	54	56
720	56	56	57	54	54	56	58	57	58	58	58	56	57	56	55	56	57	54	85	84	89	84	79	81	79	76	75	72	79	85
750	87	88	85	86	81	85	108	117	114	113	124	114	116	119	122	122	120	123	119	115	111	115	123	115	111	117	110	120	118	0
780	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
810	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
840	0	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
870	0	0	0	0	79	81	83	84	84	87	85	86	87	87	85	86	87	84	85	82	86	82	88	77	76	76	75	76	73	74
900	74	75	73	73	74	74	75	74	74	72	74	76	77	74	73	82	82	86	86	88	87	87	67	87	85	87	89	90	87	84
930	86	86	89	87	82	83	83	83	83	83	85	84	83	84	85	86	84	85	84	84	84	83	88	87	84	82	59	57	63	59
960	62	60	60	62	65	63	62	62	62	62	61	62	60	67	60	58	58	57	56	56	56	55	54	56	57	55	56	55	56	56
990	56	54	56	61	54	56	56	82	82	80	85	83	80	83	79	74	70	78	84	88	87	78	76	75	74	91	109	109	106	109
1020	119	115	122	123	120	119	119	125	124	123	122	125	122	119	120	111	121	117	0	0	0	0	0	0	0	0	0	0	0	0
1050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5	0	0
1110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	82	82	85	86	86	84	85
1140	88	90	87	90	82	85	85	84	83	84	87	82	78	77	76	77	74	74	76	76	75	74	76	73	75	73	73	76	73	74
1170	75	75	75	73	83	82	87	86	86	87	87	88	87	87	84	86	84	85	85	85	84	93	84	84	84	83	82	82	83	83
1200	82	84	81	84	84	83	83	82	83	83	81	84	83	81	83	60	58	61	61	62	61	62	64	65	65	61	61	59	61	69
1230	63	61	69	58	58	57	55	57	56	56	55	55	55	55	54	56	56	54	56	58	55	56	66	56	55	60	83	79	83	83
1260	80	77	76	74	69	68	75	75	77	79	75	72	81	76	107	109	111	107	109	106	110	121	116	122	118	119	116	122	116	112
1290	107	114	114	119	112	114	110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1350	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0
1380	0	0	0	0	0	0	0	0	0	0	0	0	84	84	86	85	86	91	89	87	90	87	86	86	84	86	85	85	86	86
1410	86	77	77	76	76	75	77	76	75	75	75	76	74	73	73	74	74	72	73	71	74	72	75	86	85	86	87	88	89	88
1440	88	85	89	85	85	86	84	84	89	87	96	83	83	83	82	84	84	82	83	81	84	85	83	85	83	84	84	85	82	83
1470	83	82	84	82	63	62	60	61	67	63	66	63	64	63	62	60	62	59	58	66	60	71	59	56	57	56	58	58	58	58
1500	55	56	54	57	56	55	56	57	59	55	56	56	55	56	56	81	77	77	78	74	75	76	78	75	74	82	75	69	74	76
1530	74	78	78	97	110	116	106	102	98	108	102	109	113	117	111	112	112	113	107	106	107	116	117	107	115	114	0	0	0	0
1560	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1590	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1620	0	0	0	0	0	7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1650	0	81	82	83	87	83	89	86	87	87	83	86	84	84	83	85	83	84	81	90	76	77	77	74	75	76	74	74	74	73
1680	75	73	76	72	72	76	74	74	73	74	74	74	84	82	86	85	87	87	88	88	84	88	86	87	83	84	86	83	88	89
1710	85	83	82	83	81	82	83	83	81	81	83	84	82	84	83	83	85	83	84	82	82	83	82	61	59	61	61	61	63	63
1740	60	63	59	59	60	58	58	59	56	59	70	58	56	56	56	58	57	56	56	54	57	55	55	56	65	59	55	57	57	53
1770	57	55	54	56	77	81	81	78	84	80	76	82	86	88	80	74	72	75	77	79	78	81	91	114	113	103	91	102	104	101
1800	106	110	109	109	112	109	112	116	107	113	110	107	112	120	117	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1830	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1860	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8	0	0	0	0	0	0
1890	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	80	83	83	84	84	85	85	88	88	86
1920	86	85	84	84	86	83	83	85	87	78	77	76	77	75	77	74	76	75	74	75	71	76	73	73	76	74	74	73	74	75
1950	75	83	84	83	83	86	87	88	91	88	89	88	85	85	85	87	86	83	95	83	83	83	81	81	82	81	82	83	83	82
1980	82	82	83	81	82	82	82	83	83	84	84	85	58	58	58	61	57	63	62	63	63	61	62	59	59	61	58	63	58	74
2010	57	56	56	56	57	57	58	58	55	58	57	56	56	56	52	56	54	55	54	54	56	56	58	81	77	82	79	83	81	89
2040	86	81	88	79	80	74	81	76	84	78	79	91	109	111	105	96	98	103	108	107	113	115	109	110	107	113	112	107	106	116
2070	107	112	110	110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2130	0	0	0	0	0	0	0	0	0	0	0	0	9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2160	0	0	0	0	0	0	0	0	0	78	80	81	85	88	86	86	86	87	85	86	85	84	81	82	83	83	80	97	79	76
2190	74	75	74	73	70	75	73	75	75	75	75	74	74	74	73	73	74	70	74	74	81	81	86	88	86	87	88	88	88	88
2220	86	85	83	83	86	87	82	98	83	80	79	80	81	81	81	83	81	83	82	82	81	81	82	81	82	83	82	83	82	86
2250	84	60	55	66	66	61	59	62	62	60	62	59	58	56	58	58	61	55	82	56	54	55	54	54	56	55	55	54	52	55
2280	53	55	54	56	53	54	54	53	56	55	57	55	71	73	73	64	68	75	78	82	77	79	82	76	73	70	78	80	78	81
2310	74	94	97	106	101	103	97	110	109	112	112	113	110	110	107	107	113	113	112	110	112	112	106	0	0	0	0	0	0	0
2340	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2370	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2400	0	0	10	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	78	80
2430	94	90	90	89	88	86	87	85	86	83	86	83	83	84	86	81	90	77	77	75	76	74	72	75	74	74	73	74	73	72
2460	74	73	72	73	73	73	72	73	76	80	82	88	89	89	89	87	89	88	87	88	86	83	86	87	87	67	92	84	79	82
2490	81	82	81	82	84	82	82	81	83	83	82	82	84	82	84	85	81	82	84	86	60	59	69	69	66	66	60	63	61	62
2520	60	62	57	63	64	59	62	71	61	55	54	57	56	56	55	54	57	55	54	55	55	55	54	53	54	55	53	55	55	57
2550	58	78	81	69	72	72	75	83	81	85	86	85	84	76	79	82	87	81	74	75	100	100	112	108	111	112	114	118	121	116
2580	116	115	113	110	111	111	110	117	121	113	116	109	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2610	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2640	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11	0	0	0	0	0	0	0	0
2670	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	78	81	89	84	88	89	87	84	85	86	82	83	83
2700	83	81	83	87	84	84	76	75	74	76	75	74	74	74	73	74	73	75	71	71	72	71	72	73	70	74	73	72	81	82
2730	87	89	86	88	88	89	88	88	89	87	86	88	88	87	87	88	82	80	80	83	83	83	81	85	85	82	84	82	84	84
2760	83	83	83	82	83	83	83	85	84	59	59	69	66	65	65	69	62	62	64	59	63	57	62	61	64	61	66	56	53	55
2790	55	56	55	56	55	55	56	56	53	57	54	63	59	55	55	53	53	56	54	55	80	75	68	75	77	74	76	80	83	82
2820	87	84	83	84	79	89	79	83	90	102	106	110	112	121	122	120	122	121	121	113	118	114	116	112	114	115	116	118	113	119
2850	110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

OUTPUT UNIVERSAL RECORD

MSSDAT OPTION
NWORDS = 690
NO. OF DATA SETS (SCANS) = 11
TOTAL NO. OF BYTES IN THIS RECORD = 2851

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	83	85	87	84	93	90	90	84	88	86	85	86	84	87	86	82	86	85	87	78	76	76	77
120	75	76	74	77	73	75	73	74	74	76	76	73	78	73	74	76	71	76	82	84	87	87	91	91	87	89	89	90	89	88
150	89	89	86	85	89	86	88	80	81	81	81	81	82	83	83	82	84	84	83	89	84	84	82	85	83	83	82	82	83	80
180	60	65	67	69	68	61	65	63	63	62	64	64	69	59	62	63	63	66	53	56	55	57	56	57	59	67	59	55	54	55
210	56	55	54	54	54	55	54	58	55	57	78	81	78	76	74	73	75	82	83	85	86	82	90	82	84	82	87	90	76	101
240	108	105	110	111	119	120	119	118	118	111	118	117	110	112	116	107	109	113	103	104	95	0	0	0	0	0	0	0	0	0
270	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
330	13	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	79	81	81	83
360	88	86	90	83	83	84	83	83	83	79	82	81	80	81	85	82	74	75	76	74	74	74	74	75	79	73	74	73	73	79
390	76	73	75	74	73	73	72	79	81	83	86	91	87	89	88	87	88	87	85	89	86	86	85	87	87	93	81	82	81	80
420	80	81	85	81	81	84	83	83	82	85	82	84	85	84	82	84	83	84	57	59	58	62	67	66	64	62	60	60	58	58
450	61	60	58	60	60	59	72	59	56	55	54	56	56	54	67	54	55	55	54	56	56	58	56	62	56	54	55	55	53	77
480	77	80	78	77	72	74	78	85	86	90	84	90	88	85	86	84	91	82	77	108	109	112	110	122	119	118	109	121	108	117
510	113	109	104	96	106	96	106	105	106	113	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
540	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
570	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0
600	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	81	84	83	85	86	88	87	87	83	86	87	82	85	83	83
630	84	80	82	88	81	75	76	75	74	75	74	74	73	75	73	76	74	73	75	72	75	75	73	72	74	72	83	81	85	86
660	88	88	85	88	88	89	87	85	87	86	88	88	89	85	77	86	83	83	83	82	82	83	82	89	80	83	82	81	82	83
690	82	84	81	82	81	83	85	59	59	58	58	63	64	60	58	65	61	59	59	60	60	60	58	64	58	78	58	55	57	64
720	55	53	56	53	55	54	53	55	54	55	53	55	53	55	56	53	54	56	80	80	78	83	75	74	79	83	84	87	90	87
750	86	90	83	90	89	92	82	107	116	117	121	116	122	114	122	115	114	115	107	104	107	102	107	117	109	113	107	110	111	0
780	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
810	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
840	0	0	0	0	0	0	0	0	15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
870	0	0	0	0	85	85	84	88	86	87	86	85	85	85	86	84	86	85	85	85	84	82	90	80	75	76	75	74	72	74
900	75	78	76	72	75	74	75	76	74	74	73	73	72	73	79	86	85	84	86	89	87	86	86	88	87	87	86	86	88	86
930	87	90	87	94	84	83	82	83	82	84	82	89	87	82	82	82	82	82	83	82	83	81	81	84	81	92	59	59	59	59
960	60	62	60	60	61	60	61	60	59	62	61	60	67	62	73	58	55	57	56	56	56	57	64	59	59	56	54	54	55	54
990	53	56	55	53	60	54	70	87	81	86	81	80	74	77	83	65	86	80	88	85	87	84	85	94	93	83	107	113	114	120
1020	117	118	116	106	107	109	106	101	102	98	98	98	99	98	97	106	104	98	0	0	0	0	0	0	0	0	0	0	0	0
1050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16	0	0
1110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	89	89	85	86	83	86	83
1140	85	86	84	85	84	83	85	87	63	84	84	90	82	77	78	75	76	76	74	75	77	73	76	74	75	75	73	73	74	73
1170	73	77	75	79	84	85	83	85	85	85	85	87	88	88	86	85	86	87	87	87	89	85	94	85	81	83	83	84	83	73
1200	86	82	83	84	84	82	80	80	82	80	81	85	83	82	88	59	58	58	56	58	59	58	69	59	59	60	60	59	62	69
1230	62	63	63	76	60	57	58	56	57	56	55	57	57	55	54	56	55	55	51	56	53	57	56	59	60	67	62	63	75	64
1260	83	76	77	80	81	87	83	81	84	83	85	85	90	91	81	107	115	119	116	119	115	112	111	114	106	107	106	103	95	94
1290	94	93	94	94	93	90	90	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1350	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	17	0	0	0	0	0	0	0	0	0	0	0	0	0
1380	0	0	0	0	0	0	0	0	0	0	0	0	82	85	83	84	85	83	84	86	85	86	86	83	81	81	82	79	78	88
1410	82	85	78	77	76	75	74	76	75	75	75	74	74	72	74	73	74	73	73	73	72	77	75	84	84	84	85	85	84	84
1440	85	85	87	85	87	85	84	84	86	86	85	94	86	83	82	84	83	82	84	83	81	82	82	83	82	82	80	82	81	83
1470	82	86	83	85	57	60	58	60	57	58	60	57	61	58	62	62	61	58	59	62	61	59	73	60	56	57	57	57	58	57
1500	55	55	56	54	57	55	56	54	55	55	58	58	69	60	65	83	78	82	80	79	73	70	76	78	85	80	79	81	85	88
1530	89	88	84	81	107	109	116	120	115	106	114	114	117	111	107	109	105	102	90	94	93	90	91	87	91	90	0	0	0	0
1560	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1590	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1620	0	0	0	0	0	14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1650	0	82	85	85	86	85	85	86	85	86	85	86	84	86	83	85	86	84	81	88	76	77	74	77	75	73	74	74	76	77
1680	75	73	74	72	74	74	73	76	74	75	78	78	84	82	87	85	83	86	83	87	87	87	89	86	85	84	87	87	87	86
1710	87	84	84	82	84	81	83	82	82	85	82	82	81	82	80	85	83	81	83	82	86	85	84	60	58	59	60	58	61	58
1740	62	61	62	63	61	61	62	62	61	65	61	62	56	58	58	58	54	55	56	54	58	58	55	51	55	58	61	58	56	59
1770	59	60	65	56	80	76	80	81	74	73	76	76	78	83	82	84	78	78	83	82	86	87	80	109	104	100	96	106	116	112
1800	106	103	108	104	107	103	94	96	93	96	90	85	90	91	96	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1830	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1860	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	0	0	0	0	0
1890	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	84	83	95	88	86	89	92	85	87	87
1920	86	85	85	84	86	84	86	82	90	80	77	77	77	76	76	74	77	74	72	74	75	73	73	73	73	71	76	74	77	73
1950	77	87	85	96	87	85	88	88	86	88	89	88	87	85	87	86	87	90	88	90	87	84	83	85	82	83	82	83	81	81
1980	83	81	84	82	84	81	83	85	84	84	86	84	62	64	71	62	58	69	64	60	61	64	63	62	61	61	64	62	66	69
2010	65	59	63	58	60	60	58	58	59	54	54	57	65	60	58	57	56	60	63	61	61	62	56	77	73	74	79	77	72	74
2040	76	81	84	78	79	77	79	81	79	81	83	90	113	96	103	100	100	100	97	97	103	111	106	104	97	94	89	95	94	90
2070	89	88	86	91	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2130	0	0	0	0	0	0	0	0	0	0	0	0	0	20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2160	0	0	0	0	0	0	0	0	0	84	65	84	87	86	87	85	85	65	86	83	84	85	83	84	84	83	85	89	82	77
2190	76	77	78	74	76	75	74	75	75	74	73	78	77	75	76	76	78	76	78	77	86	85	84	87	86	87	85	85	84	85
2220	86	84	87	86	86	86	87	86	89	86	84	82	85	83	82	81	81	81	81	83	82	84	85	83	82	85	84	86	86	83
2250	81	62	59	59	60	61	61	60	60	58	61	59	60	63	60	62	60	66	62	69	59	58	58	57	58	58	57	55	54	57
2280	55	56	60	59	61	58	60	62	69	63	61	63	63	83	82	84	83	81	78	75	77	77	83	84	80	79	80	84	83	77
2310	75	97	102	106	109	103	107	106	107	104	102	102	103	96	90	86	88	94	91	89	85	80	83	0	0	0	0	0	0	0
2340	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2370	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2400	0	0	21	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	84	88
2430	86	87	86	85	84	85	86	85	82	84	82	85	83	82	83	81	88	78	77	74	77	77	78	74	78	74	75	76	73	78
2460	76	78	76	75	75	77	79	77	80	83	85	86	86	87	87	84	85	86	87	86	85	84	86	87	87	87	87	90	85	84
2490	84	85	83	84	82	84	83	83	82	86	86	86	87	86	84	84	85	85	90	87	61	64	63	63	61	63	60	61	61	62
2520	60	61	61	63	61	62	61	64	69	58	59	58	59	58	58	57	57	58	58	55	61	61	62	64	63	58	63	62	61	79
2550	66	74	71	75	78	79	70	75	77	78	79	74	78	75	80	87	82	85	74	91	110	107	114	111	116	115	107	101	106	99
2580	109	97	90	93	91	93	93	90	85	90	79	83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2610	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2640	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	22	0	0	0	0	0	0	0	0
2670	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	84	86	87	87	88	87	87	85	85	85	84	86	83
2700	82	83	82	82	83	87	84	77	77	76	77	74	77	77	75	75	76	82	78	87	80	78	78	78	76	85	80	79	84	83
2730	86	87	85	85	87	85	86	86	88	86	85	85	87	87	88	85	92	85	85	86	83	83	83	84	84	84	87	84	95	99
2760	92	90	86	86	88	86	96	92	91	58	63	61	61	64	62	61	61	63	60	61	59	60	60	60	60	60	62	69	60	57
2790	58	66	57	66	57	58	57	65	59	73	90	69	70	63	63	66	70	76	75	73	82	77	75	78	72	70	75	74	78	74
2820	79	75	74	77	82	83	85	84	75	100	99	106	107	106	107	106	106	98	97	94	97	94	96	91	90	96	91	84	89	87
2850	82	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

OUTPUT UNIVERSAL RECORD

MSSDAT OPTION
NWORDS = 640
NO. OF DATA SETS (SCANS) = 11.
TOTAL NO. OF BYTES IN THIS RECORD = 2851

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	23	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	85	87	87	87	86	88	87	84	88	85	84	86	83	85	85	84	83	81	88	78	76	77	77
120	76	77	76	77	76	78	79	77	76	76	76	76	77	79	80	81	78	83	88	87	89	89	89	87	88	88	89	86	87	86
150	86	89	89	89	90	87	94	84	83	83	84	83	83	84	86	84	86	86	87	84	86	84	84	88	86	91	89	82	101	65
180	64	67	61	64	64	63	64	64	64	63	66	63	65	64	61	65	65	66	59	56	58	59	60	58	60	61	62	64	64	63
210	60	61	58	60	66	67	69	66	60	88	83	81	84	83	76	78	78	73	74	78	82	77	75	76	82	84	88	84	81	98
240	97	106	105	104	99	101	93	90	89	86	94	90	99	93	93	89	80	85	81	87	90	0	0	0	0	0	0	0	0	0
270	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
330	24	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	80	82	79	82
360	80	83	83	82	79	81	79	89	87	90	93	90	89	91	79	80	86	88	87	87	87	86	86	87	87	88	89	89	88	88
390	86	88	87	88	87	86	86	99	100	99	98	104	108	109	103	96	98	121	132	137	133	139	134	142	110	89	94	94	91	93
420	93	91	92	94	93	93	94	93	93	94	93	93	94	96	98	91	91	91	103	101	100	100	104	109	110	102	98	103	127	134
450	130	131	139	127	146	106	70	68	74	71	72	73	71	70	74	74	76	76	73	78	74	75	74	74	75	76	71	68	72	74
480	75	78	73	76	80	81	73	74	77	90	89	90	88	98	97	92	89	77	89	83	77	81	75	77	82	79	78	80	78	78
510	77	78	81	76	83	82	83	78	82	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
540	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
570	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	26	0	0	0	0	0	0	0	0	0	0
600	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	82	82	81	83	80	85	82	83	80	80	82	81	84	87	86
630	85	82	83	81	88	90	87	88	90	88	89	89	87	88	89	89	91	87	88	87	88	88	89	87	87	88	99	101	96	100
660	103	107	101	98	99	99	98	106	112	118	115	116	102	106	96	94	95	94	93	95	94	94	94	93	92	92	94	92	94	94
690	94	94	95	94	92	92	92	102	103	96	103	102	103	99	100	100	99	98	108	105	100	103	114	106	108	81	72	70	74	70
720	73	74	71	74	74	72	75	77	73	75	74	75	75	76	73	74	67	70	81	80	78	77	79	81	74	74	73	75	75	79
750	84	92	90	90	88	81	82	81	87	83	79	83	79	82	78	79	79	80	77	80	80	80	84	83	79	89	81	76	80	0
780	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
810	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
840	0	0	0	0	0	0	0	0	26	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
870	0	0	0	0	79	81	80	79	82	82	82	80	78	79	80	80	82	87	88	86	80	89	81	87	87	90	90	87	88	86
900	88	87	88	89	87	88	86	88	88	88	90	87	88	87	84	96	99	98	100	109	111	104	100	99	102	105	104	118	126	130
930	113	112	133	97	93	95	93	95	94	94	95	94	94	94	94	97	95	93	94	96	96	96	93	95	91	92	95	100	98	97
960	105	98	102	98	101	101	102	106	112	113	124	111	111	127	86	69	72	74	71	70	74	72	73	76	73	75	74	76	72	75
990	75	76	77	73	77	71	69	70	76	74	73	81	81	75	75	77	74	79	77	83	91	91	89	79	95	87	77	77	79	77
1020	86	80	81	80	79	81	82	77	83	83	80	81	81	79	80	80	79	79	0	0	0	0	0	0	0	0	0	0	0	0
1050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27	0	0	0
1110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	78	78	79	80	82	83	89
1140	83	80	79	80	81	79	79	77	81	82	76	84	84	90	88	89	89	87	89	89	88	89	90	89	91	88	89	88	92	89
1170	88	89	87	88	96	95	100	105	113	110	116	101	100	102	106	108	103	102	99	103	110	116	104	92	95	93	94	94	95	94
1200	97	95	95	95	97	97	95	96	95	97	97	96	98	93	93	98	93	98	104	100	91	108	96	98	103	106	105	101	99	101
1230	100	106	117	91	73	73	72	71	72	74	74	75	74	74	77	78	77	75	75	78	77	78	76	78	71	69	75	76	74	78
1260	86	83	83	81	77	76	85	83	82	80	76	76	87	81	81	81	76	79	79	82	78	81	82	80	83	82	82	79	85	78
1290	82	82	82	81	80	81	47	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1350	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	28	0	0	0	0	0	0	0	0	0	0	0	0	0
1380	0	0	0	0	0	0	0	0	0	0	0	0	80	78	81	85	85	88	89	89	83	83	84	79	79	80	80	81	86	85
1410	78	84	89	89	90	88	89	87	88	89	88	91	89	90	90	90	89	89	90	90	89	88	86	96	97	95	118	117	117	127
1440	118	105	121	110	105	103	100	102	117	123	121	100	93	93	94	94	94	95	95	96	97	96	96	97	95	96	96	95	98	97
1470	96	95	93	91	95	100	95	107	95	96	110	116	108	123	114	109	106	104	102	119	118	121	95	75	76	71	74	73	74	73
1500	77	79	75	76	78	77	76	76	77	76	78	75	78	71	69	76	74	72	82	88	90	92	89	82	90	89	83	83	78	75
1530	83	89	90	79	83	77	82	77	80	77	85	83	81	80	80	79	79	83	79	81	78	82	78	83	76	77	0	0	0	0
1560	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1590	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1620	0	0	0	0	0	29	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1650	0	82	82	82	84	85	86	83	84	84	82	81	79	80	81	80	80	89	90	92	86	92	91	91	90	92	87	88	92	90
1680	89	90	89	89	89	88	91	90	88	89	89	89	95	97	103	110	107	107	112	110	106	107	103	100	97	97	102	109	127	141
1710	115	95	97	92	95	94	96	95	94	97	96	95	96	96	96	94	96	97	95	96	95	95	92	91	95	100	99	88	92	107
1740	109	110	108	103	102	102	98	99	113	127	149	109	77	75	74	73	74	74	73	73	78	78	76	78	78	75	76	78	77	76
1770	73	77	73	72	73	74	75	82	86	79	83	90	84	83	82	79	79	77	78	81	90	91	90	78	79	83	78	82	78	81
1800	83	81	80	81	81	77	80	78	79	80	83	83	85	81	83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1830	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1860	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30	0	0	0	0	0	0
1890	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	77	82	79	85	85	82	85	82	79	80
1920	80	80	78	79	79	80	82	82	82	81	88	91	91	90	91	91	90	91	89	89	89	88	90	90	88	90	87	88	88	90
1950	98	95	99	108	111	120	110	115	104	103	104	102	101	95	102	98	102	114	111	106	94	96	96	96	94	96	96	95	96	96
1980	95	93	96	97	94	96	96	95	96	95	95	94	92	95	106	97	104	100	112	105	109	102	103	102	97	99	95	105	120	119
2010	110	81	75	76	75	75	76	74	74	76	75	76	74	75	78	76	78	74	73	76	75	74	72	71	74	76	85	88	89	86
2040	82	80	80	86	80	74	78	82	84	84	88	87	83	85	84	85	78	79	82	86	84	82	81	82	77	76	78	77	81	86
2070	80	89	82	83	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2100	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2130	0	0	0	0	0	0	0	0	0	0	0	0	0	31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2160	0	0	0	0	0	0	0	0	0	83	83	90	83	87	87	85	82	81	81	80	79	79	78	81	86	87	78	85	79	90
2190	92	88	91	91	88	90	90	89	89	90	89	90	92	91	88	91	89	88	91	90	101	114	117	113	115	118	105	97	104	100
2220	100	97	102	99	124	116	135	108	108	95	95	94	95	95	93	93	94	94	94	95	96	97	95	94	94	94	93	92	93	99
2250	94	97	109	110	109	114	115	105	100	105	101	103	100	100	98	120	108	127	109	113	81	74	74	76	73	79	71	76	75	71
2280	75	76	77	79	75	75	73	72	72	73	73	72	77	83	85	82	87	85	80	75	80	78	79	74	80	81	90	89	95	79
2310	87	85	78	81	76	82	77	81	75	78	85	79	77	77	81	78	82	81	80	80	85	78	82	0	0	0	0	0	0	0
2340	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2370	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2400	0	0	32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	83	83
2430	82	82	84	83	86	83	82	82	80	82	84	84	86	86	84	87	87	83	90	89	93	89	89	89	90	89	89	92	89	90
2460	91	91	89	90	89	90	88	89	90	102	105	103	97	101	108	108	101	99	100	108	108	114	115	123	112	119	124	111	96	95
2490	94	94	91	92	94	94	96	94	94	96	95	96	94	93	96	95	95	93	96	91	99	103	100	99	99	102	101	97	99	101
2520	107	111	110	135	117	107	126	124	113	75	76	74	73	71	71	73	76	76	76	75	77	77	76	77	76	76	73	74	73	72
2550	72	81	80	74	73	74	75	86	77	75	79	78	81	88	90	90	91	92	90	89	79	83	81	77	73	74	78	81	78	78
2580	77	79	77	79	81	80	81	87	82	86	79	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2610	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
2640	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	33	0	0	0	0	0	0	0	0
2670	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	78	81	80	80	79	80	79	78	77	77	80	86	83
2700	79	82	78	79	81	82	77	87	92	90	88	87	87	86	90	90	86	89	86	87	88	89	89	87	91	88	91	88	102	98
2730	92	93	92	94	95	94	94	94	122	116	113	101	101	99	107	107	106	94	93	94	93	91	93	93	93	93	92	93	92	90
2760	92	93	97	97	96	94	95	94	90	93	98	91	89	91	93	92	92	93	98	114	118	109	103	103	99	111	110	112	82	72
2790	74	74	73	72	72	75	72	71	73	76	72	72	78	78	74	76	74	73	74	71	85	74	68	68	63	67	73	74	70	75
2820	91	91	95	86	83	78	85	88	84	80	78	77	74	72	77	78	74	76	75	80	78	79	84	80	79	85	83	84	86	82
2850	78	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

OUTPUT UNIVERSAL RECORD

NBSDAT OPTION

NWORDS = 690

NO. OF DATA SETS (SCANS) = 6

TOTAL NO. OF BYTES IN THIS RECORD = 2079

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
60	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
90	0	0	0	0	0	0	0	79	82	81	79	81	80	81	80	76	77	78	77	80	79	78	78	78	79	79	79	88	89	88
120	90	90	90	88	87	88	87	88	88	87	89	88	89	88	88	86	87	85	97	97	92	93	90	93	96	94	92	96	99	103
150	101	98	97	97	100	97	98	94	92	92	93	94	96	94	98	92	91	92	94	93	95	96	95	95	96	99	94	92	91	88
180	90	92	89	89	93	91	94	94	96	97	107	102	101	98	99	104	101	102	79	72	70	73	75	74	73	74	73	69	76	76
210	75	75	77	80	74	78	73	72	72	71	71	73	68	69	68	73	72	72	70	76	78	83	80	75	76	82	76	79	74	76
240	77	79	76	75	78	78	84	78	75	81	82	80	79	81	81	83	82	84	89	78	84	0	0	0	0	0	0	0	0	0
270	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
300	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
330	35	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	80	82	79	80
360	81	81	79	76	79	78	79	77	80	77	77	80	79	77	79	83	88	89	89	90	89	88	90	90	88	89	91	89	90	89
390	89	87	88	87	87	85	86	99	98	88	91	92	91	98	96	91	96	96	98	99	96	99	99	99	96	97	95	93	93	93
420	92	93	94	94	93	95	96	95	93	93	96	98	96	98	93	96	92	93	92	90	87	88	89	87	89	92	91	93	96	99
450	100	94	98	99	101	95	103	77	73	72	74	73	79	72	73	76	73	78	78	76	76	78	77	77	73	76	71	73	69	75
480	70	66	68	66	70	74	70	75	70	74	74	76	78	81	78	78	76	75	80	77	77	74	79	75	83	80	78	86	81	81
510	80	83	79	83	82	82	79	85	76	82	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
540	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
570	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	36	0	0	0	0	0	0	0	0	0	0
600	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	80	91	82	82	82	81	79	79	78	81	79	80	80	80	81
630	82	81	87	81	79	86	90	92	89	89	89	87	89	88	91	92	90	88	89	90	89	89	90	88	89	89	97	95	91	92
660	93	93	96	92	91	96	99	98	97	102	104	107	123	110	104	94	91	93	93	90	90	90	92	91	94	95	95	94	95	96
690	96	95	99	93	93	93	92	91	90	86	95	88	91	87	87	91	94	96	99	95	101	98	106	128	111	108	73	74	70	71
720	71	73	69	70	71	72	76	76	74	75	75	75	76	75	74	73	74	72	78	73	67	67	67	69	71	71	70	76	79	73
750	72	78	77	83	91	91	87	79	77	75	76	77	75	79	81	80	78	82	82	79	81	81	82	85	85	82	86	86	82	0
780	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
810	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
840	0	0	0	0	0	0	0	0	37	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
870	0	0	0	0	80	81	82	81	82	79	78	76	76	80	79	79	77	82	80	80	80	79	79	86	89	89	91	86	88	88
900	90	90	86	90	89	90	90	90	91	89	90	87	85	89	87	97	94	90	95	95	97	92	92	91	98	110	109	103	116	109
930	103	103	106	100	94	94	94	94	94	93	95	93	97	96	98	95	95	96	96	97	97	95	94	92	93	95	96	90	90	96
960	95	93	87	89	88	96	104	100	98	115	96	105	105	107	94	75	73	72	73	72	73	72	76	77	72	78	79	74	79	77
990	78	77	74	74	69	73	75	73	68	69	70	67	73	69	67	66	76	79	80	78	86	79	83	81	80	78	74	76	79	74
1020	75	81	76	76	76	83	87	75	75	79	81	82	78	80	83	80	81	77	0	0	0	0	0	0	0	0	0	0	0	0
1050	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1080	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	38	0	0
1110	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	81	82	81	80	81	79	78
1140	79	79	78	79	81	79	82	77	81	79	77	80	81	90	89	89	91	88	90	90	91	90	90	91	89	91	91	91	89	89
1170	89	89	84	88	105	94	91	92	92	95	89	89	88	94	102	105	99	96	101	101	100	97	100	94	97	94	92	95	95	94
1200	96	97	96	97	97	95	97	97	95	96	96	94	96	94	96	99	89	85	87	89	93	82	86	86	89	96	96	94	90	97
1230	95	103	98	98	74	74	73	73	72	74	74	74	77	76	74	78	75	78	76	77	76	74	75	71	70	74	77	69	68	66
1260	64	69	66	67	64	68	75	77	73	76	75	83	79	79	77	77	81	83	80	77	82	81	84	82	83	80	82	81	80	80
1290	83	85	77	83	84	80	84	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1320	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1350	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0
1380	0	0	0	0	0	0	0	0	0	0	0	0	79	80	78	79	81	80	81	78	80	78	77	83	84	77	82	83	83	86
1410	89	85	90	92	93	90	90	89	90	92	90	88	93	93	87	90	91	90	90	87	91	88	91	98	92	91	91	97	95	94
1440	93	90	99	109	118	108	106	107	115	119	132	112	94	95	94	95	96	98	94	96	98	95	98	96	96	98	97	96	97	96
1470	94	98	95	96	93	89	87	79	74	95	95	92	85	99	103	112	104	105	104	112	122	132	112	74	73	74	76	73	74	74
1500	75	75	75	78	77	75	78	78	77	78	75	73	74	74	71	75	70	68	68	69	70	68	66	66	75	78	86	82	77	78
1530	87	86	94	90	80	81	80	79	86	80	85	77	80	83	84	85	78	80	81	81	88	86	82	86	88	81	0	0	0	0
1560	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1590	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1620	0	0	0	0	0	40	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1650	0	80	82	80	80	82	82	80	81	80	81	82	82	84	81	80	80	80	80	80	79	90	89	92	89	89	91	91	92	90
1680	93	99	91	91	90	90	91	89	93	88	90	91	96	93	90	94	94	92	90	91	92	97	108	104	104	100	101	100	102	101
1710	101	95	94	95	94	96	95	96	96	97	96	98	99	96	96	96	99	99	97	96	99	98	96	90	91	93	90	90	88	88
1740	94	92	100	104	103	106	102	100	100	106	101	107	77	75	74	76	76	77	74	80	80	76	78	82	78	74	79	78	78	78
1770	77	76	78	74	71	69	68	66	68	66	64	69	68	75	79	79	83	78	81	78	81	79	81	77	78	82	81	82	77	80
1800	84	79	76	82	76	79	83	81	84	86	80	85	88	85	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1830	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1860	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	41	0	0	0	0	0
1890	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	78	82	80	80	79	81	81	80	79	78
1920	78	80	78	81	80	77	80	80	76	86	91	89	91	91	90	90	91	89	93	91	91	93	88	89	90	87	91	87	88	89
1950	87	94	95	91	91	91	89	90	87	91	97	98	95	98	98	101	98	101	101	96	95	96	94	94	97	94	96	97	96	97
1980	99	103	98	99	95	95	94	94	93	96	96	97	92	91	90	89	88	86	87	85	92	95	97	96	100	96	105	100	103	100
2010	100	78	73	75	74	76	77	75	76	76	77	80	83	80	75	76	72	76	75	75	75	74	73	67	67	63	66	66	62	69
2040	61	64	75	72	73	74	77	82	75	78	80	74	82	75	77	80	78	77	79	79	81	81	85	85	82	87	82	79	81	77
2070	84	80	82	88	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

NUMBER OF UNIVERSAL RECORDS WRITTEN IS 4

THE OPTION UVWRIT REQUIRED 12.6206 SECONDS OF CPU TIME.

UVWRIT OPTION

SAMPLE 2 INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>UVWRIT

                    UVWRIT  OPTION
                    **************

CHOOSE OPTION FROM
  MSSDAT  IMAGE
>IMAGE

IMAGE  OPTION HAS BEEN SELECTED
PRINT UNIVERSAL FORMAT OUTPUT - YES OR NO
>NO
```

```
                    F I E L D   N U M B E R   1
                    ---------------------------

NUMBER OF SUBFRAME STATUS BITS BEING USED IS      0


                         UNIVERSAL HEADER RECORD
                         -----------------------

START  NO OF
BYTE   BYTES   DESCRIPTION

1       32     COMPUTING SYSTEM ID                 *
33      20     TAPE LIBRARY ID                     *
53      8      SENSOR ID                           *S192
61      3      DATE OF TAPE GENERATION  DD MM YY   *   0    0    0
64      1      TAPE SEQUENCE ID                    *        0
65      2      MISSION NUMBER                      *        0
67      2      SITE                                *        0
69      1      LINE                                *        0
70      1      RUN                                 *        0
71      2      ORBIT                               *        0
73      8      TIME OF FIRST SCAN  DD MM YY        *   0    0    0
                                   HH MM SS .MS    *   0    0    0    0
81      8      CHANNELS ACTIVE                     *

1000000000000000000000000000000000000000000000000000000000000000

89      1      PROCESSING FLAG                     *        1
90      1      NO.CHANNELS ON THIS TAPE            *        1
91      1      NO. OF BITS/PIXEL                   *        8
92      2      FIRST VIDEO PIXEL WITHIN SCAN       *        1
94      2      FIRST CAL. ELEMENT WITHIN SCAN      *        0
96      2      NO. PIXELS/SCAN/CHANNEL             *       41
98      2      NO. CALIB./SCAN/CHANNEL  (AREA 1)   *        0
100     2      PHYSICAL RECORD SIZE IN BYTES       *     3040
102     1      NO. CHANNELS/PHYSICAL RECORD        *        0
103     1      PHYS. RECORDS/SCAN/CHANNEL          *        0
104     1      NO. RECORDS TO A DATA SET           *        1
105     2      ANCILLARY DATA LENGTH IN BYTES      *       95
107     1      DATA ORDER INDICATOR                *        0
108     2      START PIXEL NUMBER                  *       10
110     2      STOP PIXEL NUMBER                   *       90
112     512    COEFFS,EXPS. A0-E0-A1-E1            *

        0            0            0            0

624     64     COLOR CODE 0=NA,1=R,2=G,3=B         *
                                                            0

688     64     SCALE FACTOR                        *
                                                            0


752     1      OFFSET CONSTANT                     *        0
763     1      WORD SIZE OF GENERATING COMPUTER    *       36
1778    1      NO OF DATA SETS/PHYSICAL RECORD     *       22
1779    2      START OF SECOND CALIB. AREA         *        0
1781    2      NO. CAL. ELE. IN SECOND CAL. AREA   *        0
1783    1      FIRST CAL. SOURCE INDICATOR         *        0
1783    1      SECOND CAL. SOURCE INDICATOR        *        0
1784    1      FILL ZEROS                          *00000000
1785    2      NO. CHANNELS IN FIRST RECORD        *        1
1787    2      BYTES/SCAN/CHANNEL                  *       41
1789    2      PIXEL SKIP FACTOR                   *        2
1791    2      SCAN SKIP FACTOR                    *        2
2759    1      N THOUSAND SCAN LINES PER FRAME     *        0
2790    3      ALTITUDE IN METERS                  *        0
2793    2      GROUND SPEED IN METERS PER SECOND   *        0
2795    1      SCAN TYPE =LINEAR OR SMOOTHED       *        0
2796    1      ANGLE OF ARC IN DEGREES             *        0
2797    1      CAMERA   0 = 70 MM,  1 = 5 INCH     *        0
2798    1      INPUT DEV. 0 = 9TRK,  1 = HDT       *        0
2799    1      TRUNCATION  0=LO, 1=HI, 2=NO        *        0
2800    8      CHANNELS REQUESTED                  *
1000000000000000000000000000000000000000000000000000000000000000
2808    1      PROCESSING MODE  0=SER, 1=CONC.     *        0
2874    1      COLOR 0=NO COLOR,1=ASSIGN,2=FALSE   *        0
2875    1      IMAGE FORMAT                        *        0
2876    1      REPEAT OF PIXELS PER SCAN           *        0
2877    1      REPEAT OF SCAN                      *        0
2878    4      PARTIAL SCAN -  FROM PIXEL NO.      *        0
                            -  TO  PIXEL  NO.      *        0
2882    2      SCAN RATE IN SCANS PER SECOND       *        0
2885    2      ANGLE OF DRIFT                      *     .000
2889    1      PIXEL  SIZE                         *        0


NUMBER OF UNIVERSAL RECORDS WRITTEN IS       2
THE OPTION UVWRIT REQUIRED       4.2710 SECONDS OF CPU TIME.

------------------------------------------------------------------
```

UVWRIT ENGINEERING DESCRIPTION

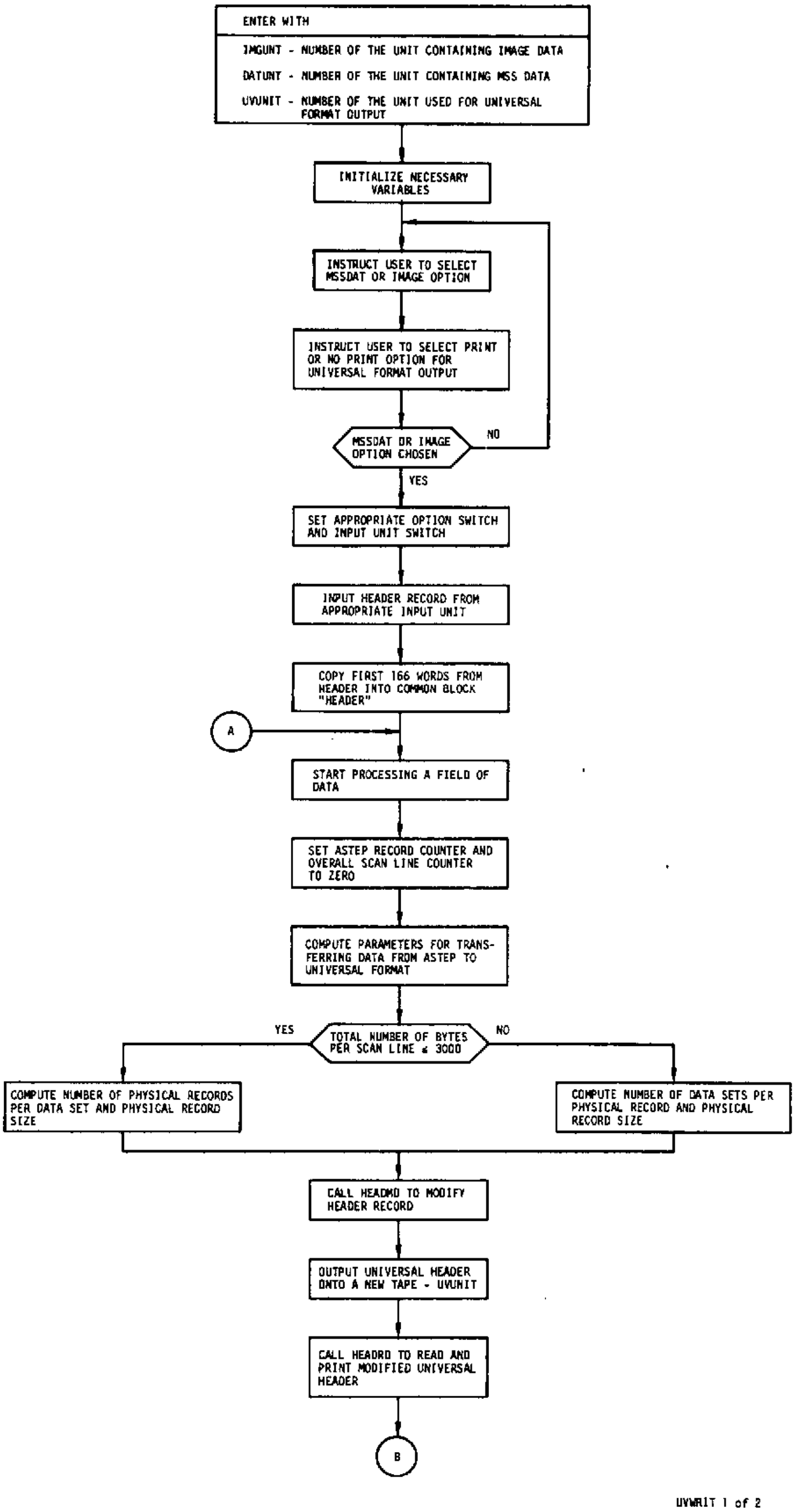
The purpose of the UVWRIT option is to convert data (multispectral data or image data) from internal ASTEP format to Universal format (Reference 1).

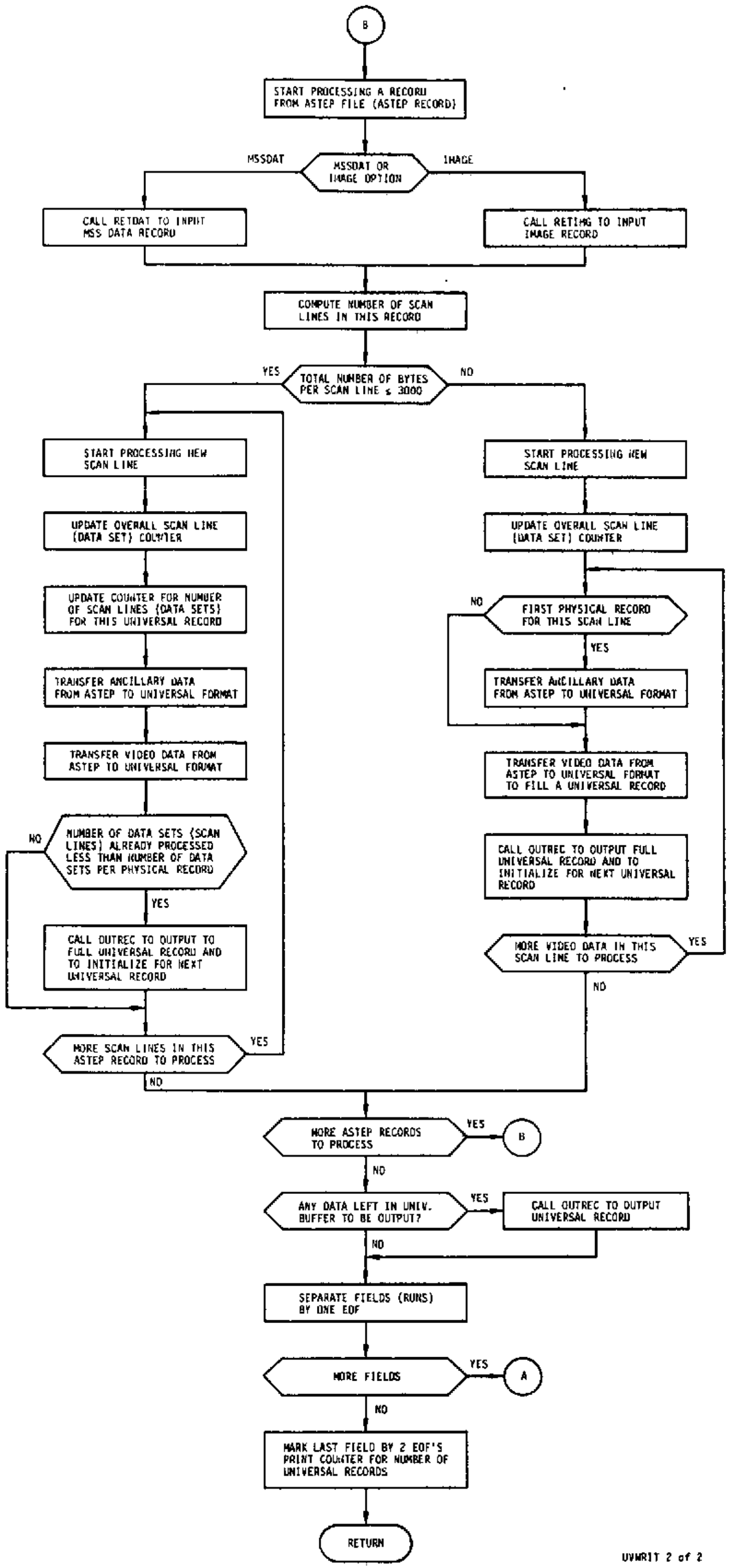
The MSS data coming into ASTEP may be in either LARS, ERTS or Universal format. It is converted in the DATDEF option into ASTEP internal format. The image data resulting from any of the classification options in ASTEP is also stored in internal ASTEP format. UVWRIT takes MSS data or image data (depending on the suboption selected by user), converts it to the universal format, and outputs it onto a magnetic tape.

A new universal header is created using information obtained from the ASTEP header. In the case when original data coming into ASTEP is in universal format, the existing universal header is modified when necessary to reflect the changes to the original data structure (i.e., scan line length, skip factors, etc.).

Next, all the records from ASTEP file are converted into universal format and stored on a magnetic tape.

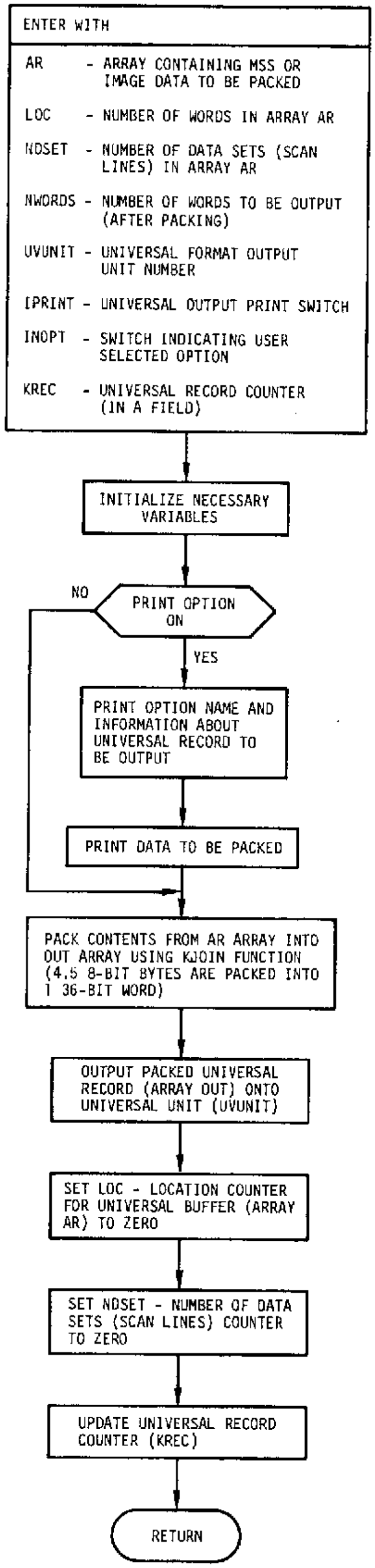
All information in the universal format is in a packed form based on an 8-bit bytes structure. The length of the header record is 3060 bytes (680 36-bit words). The length of each physical record following the header is variable, not exceeding 3000 bytes of information per record. Each record in universal format may contain one or more scan lines, or part of a scan line, depending on the number of pixels and number of channels per scan line. For more information on universal format user should refer to the Format Control Book (Reference 1).


ENTER WITH
IMGUNT - NUMBER OF THE UNIT CONTAINING IMAGE DATA
DATUNT - NUMBER OF THE UNIT CONTAINING MSS DATA
UVUNIT - NUMBER OF THE UNIT USED FOR UNIVERSAL FORMAT OUTPUT
INITIALIZE NECESSARY VARIABLES
INSTRUCT USER TO SELECT MSSDAT OR IMAGE OPTION
INSTRUCT USER TO SELECT PRINT OR NO PRINT OPTION FOR UNIVERSAL FORMAT OUTPUT
MSSDAT OR IMAGE OPTION CHOSEN
NO
YES
SET APPROPRIATE OPTION SWITCH AND INPUT UNIT SWITCH
INPUT HEADER RECORD FROM APPROPRIATE INPUT UNIT
COPY FIRST 166 WORDS FROM HEADER INTO COMMON BLOCK "HEADER"
A
START PROCESSING A FIELD OF DATA
SET ASTEP RECORD COUNTER AND OVERALL SCAN LINE COUNTER TO ZERO
COMPUTE PARAMETERS FOR TRANSFERRING DATA FROM ASTEP TO UNIVERSAL FORMAT
YES
TOTAL NUMBER OF BYTES PER SCAN LINE ≤ 3000
NO
COMPUTE NUMBER OF PHYSICAL RECORDS PER DATA SET AND PHYSICAL RECORD SIZE
COMPUTE NUMBER OF DATA SETS PER PHYSICAL RECORD AND PHYSICAL RECORD SIZE
CALL HEADMD TO MODIFY HEADER RECORD
OUTPUT UNIVERSAL HEADER ONTO A NEW TAPE - UVUNIT
CALL HEADRD TO READ AND PRINT MODIFIED UNIVERSAL HEADER
B

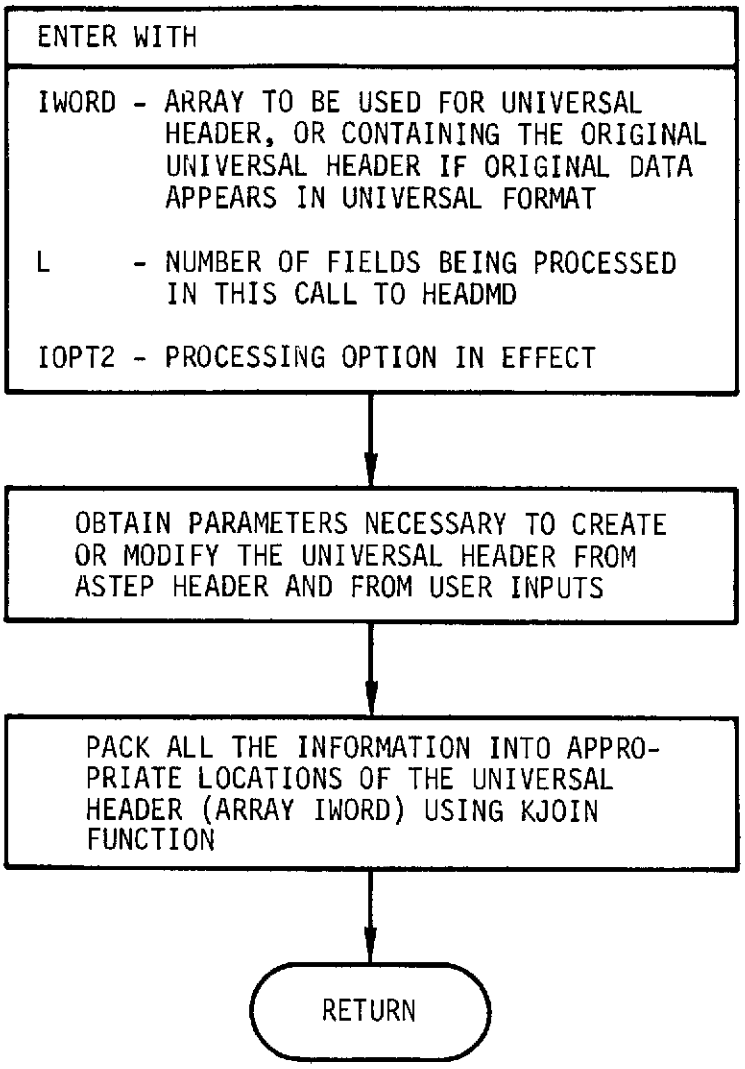

B
START PROCESSING A RECORD FROM ASTEP FILE (ASTEP RECORD)
MSSDAT
MSSDAT OR IMAGE OPTION
IMAGE
CALL RETDAT TO INPUT MSS DATA RECORD
CALL RETIMG TO INPUT IMAGE RECORD
COMPUTE NUMBER OF SCAN LINES IN THIS RECORD
YES
TOTAL NUMBER OF BYTES PER SCAN LINE ≤ 3000
NO
START PROCESSING NEW SCAN LINE
UPDATE OVERALL SCAN LINE (DATA SET) COUNTER
UPDATE COUNTER FOR NUMBER OF SCAN LINES (DATA SETS) FOR THIS UNIVERSAL RECORD
TRANSFER ANCILLARY DATA FROM ASTEP TO UNIVERSAL FORMAT
TRANSFER VIDEO DATA FROM ASTEP TO UNIVERSAL FORMAT
NO
NUMBER OF DATA SETS (SCAN LINES) ALREADY PROCESSED LESS THAN NUMBER OF DATA SETS PER PHYSICAL RECORD
YES
CALL OUTREC TO OUTPUT TO FULL UNIVERSAL RECORD AND TO INITIALIZE FOR NEXT UNIVERSAL RECORD
MORE SCAN LINES IN THIS ASTEP RECORD TO PROCESS
YES
NO
START PROCESSING NEW SCAN LINE
UPDATE OVERALL SCAN LINE (DATA SET) COUNTER
NO
FIRST PHYSICAL RECORD FOR THIS SCAN LINE
YES
TRANSFER ANCILLARY DATA FROM ASTEP TO UNIVERSAL FORMAT
TRANSFER VIDEO DATA FROM ASTEP TO UNIVERSAL FORMAT TO FILL A UNIVERSAL RECORD
CALL OUTREC TO OUTPUT FULL UNIVERSAL RECORD AND TO INITIALIZE FOR NEXT UNIVERSAL RECORD
MORE VIDEO DATA IN THIS SCAN LINE TO PROCESS
YES
NO
MORE ASTEP RECORDS TO PROCESS
YES
B
NO
ANY DATA LEFT IN UNIV. BUFFER TO BE OUTPUT?
YES
CALL OUTREC TO OUTPUT UNIVERSAL RECORD
NO
SEPARATE FIELDS (RUNS) BY ONE EOF
MORE FIELDS
YES
A
NO
MARK LAST FIELD BY 2 EOF'S PRINT COUNTER FOR NUMBER OF UNIVERSAL RECORDS
RETURN

UVWRIT 2 of 2

OUTREC


ENTER WITH
AR - ARRAY CONTAINING MSS OR IMAGE DATA TO BE PACKED
LOC - NUMBER OF WORDS IN ARRAY AR
NDSET - NUMBER OF DATA SETS (SCAN LINES) IN ARRAY AR
NWORDS - NUMBER OF WORDS TO BE OUTPUT (AFTER PACKING)
UVUNIT - UNIVERSAL FORMAT OUTPUT UNIT NUMBER
IPRINT - UNIVERSAL OUTPUT PRINT SWITCH
INOPT - SWITCH INDICATING USER SELECTED OPTION
KREC - UNIVERSAL RECORD COUNTER (IN A FIELD)
INITIALIZE NECESSARY VARIABLES
PRINT OPTION ON
NO
YES
PRINT OPTION NAME AND INFORMATION ABOUT UNIVERSAL RECORD TO BE OUTPUT
PRINT DATA TO BE PACKED
PACK CONTENTS FROM AR ARRAY INTO OUT ARRAY USING KJOIN FUNCTION (4.5 8-BIT BYTES ARE PACKED INTO 1 36-BIT WORD)
OUTPUT PACKED UNIVERSAL RECORD (ARRAY OUT) ONTO UNIVERSAL UNIT (UVUNIT)
SET LOC - LOCATION COUNTER FOR UNIVERSAL BUFFER (ARRAY AR) TO ZERO
SET NDSET - NUMBER OF DATA SETS (SCAN LINES) COUNTER TO ZERO
UPDATE UNIVERSAL RECORD COUNTER (KREC)
RETURN

HEADMD


ENTER WITH
IWORD - ARRAY TO BE USED FOR UNIVERSAL HEADER, OR CONTAINING THE ORIGINAL UNIVERSAL HEADER IF ORIGINAL DATA APPEARS IN UNIVERSAL FORMAT
L - NUMBER OF FIELDS BEING PROCESSED IN THIS CALL TO HEADMD
IOPT2 - PROCESSING OPTION IN EFFECT
OBTAIN PARAMETERS NECESSARY TO CREATE OR MODIFY THE UNIVERSAL HEADER FROM ASTEP HEADER AND FROM USER INPUTS
PACK ALL THE INFORMATION INTO APPROPRIATE LOCATIONS OF THE UNIVERSAL HEADER (ARRAY IWORD) USING KJOIN FUNCTION
RETURN

Using the QUIT Option

The QUIT option returns control to the operating system. This option is normally executed when the run is to be terminated. Following execution of the QUIT option, the total CPU time required by the various options executed in the run is printed out.

QUIT OPTION
SAMPLE INPUT AND CORRESPONDING OUTPUT:

```
ENTER ASTEP OPTION OR TYPE A BLANK
>QUIT

                    QUIT     OPTION
                    **************

 ------------------------------------------------------------

 THE OPTIONS IN THIS RUN REQUIRED    100.7416 SECONDS OF CPU TIME.

 +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

4. SUBROUTINE FUNCTIONS AND DEPENDENCIES

Table 4.1 gives a listing of all the subroutines and their basic functions. Table 4.2 displays the subroutine dependencies.

Also included is a list of subroutines common to more than one ASTEP option with their flow diagrams, when appropriate.

Table 4.1 Subroutine Functions

Subroutine	Basic Function
ASTEPX	Main program or driver for the Algorithm Simulation Test and Evaluation Program
ADDSIG	Given two signatures, creates the signature of the union of the two signatures
ADPCLU	Driver for the adaptive clustering option
ANGDIS	Given a set of vectors, computes the distances and angles between each pair of vectors
ANGLE	Computes the angle between two vectors
BEGFIL	Begins a spectral signature file
CESCA	Cluster elimination, splitting, and combining algorithm
CHLSKY	Inverts a positive definite matrix in double precision using a Chlesky factorization
CLOS	Given a set of vectors, determines the pair which are nearest to each other
CLUSTA	Adaptive clustering algorithm, 1st pass through develops cluster means using periodic elimination and merger tests, also has features of strip formulation and sequential search for strip assignment, 2nd pass through develops classification map.
COMPAR	Computes projections and angles used to compare two signature means and covariance matrices
CONVRT	Performs alphanumeric and integer conversions for use with the PLOT routine
CPUTIM	Reads the system clock so that time to execute an ASTEPX option can be computed
CUBIC	Passes a cubic equation through four distinct points and solves for the coordinates of the minimum value of the ordinate

Table 4.1 Subroutine Functions (Continued)

Subroutine	Basic Function
DATDEF	Allows the user to define the data subset of the raw packed MSS observation data to be processed
DAVIDN	The Davidon Iterator, used to obtain the minimum of a function of several variables
DIF	Differs two alphabetic images based upon an equivalence table defining equal alphabetic characters
DIFIMG	Driver for image saving and image comparisons
DISPLA	Displays a line of alphabetic characters
DUMP	Translates into readable form and prints contents of a tape file
EDIST	Computes the distance between two vectors
EDTSIG	Driver for spectral signature file data manipulation routines
EIGEN	Determines eigenvalues and eigenvectors of a symmetric matrix
EIGSIG	Computes the eigenvalues and eigenvectors for the signature in core storage
ERRPRT	Prints messages when error conditions or unexpected end of file are encountered while reading namelist
FACANL	Orders the eigenvalues and corresponding eigenvectors
FAKTOR	Computes mean and covariance matrix of a set of vectors, performs a factor analysis of the results, option to save the mean and covariance matrix in signature file
FEATSL	Driver for the feature selection option
FINT	Computes partial derivatives for use with the feature selection option

Table 4.1 Subroutine Functions (Continued)

Subroutine	Basic Function
FNDVID	Locates the first pixel of video data within a scan line
GOBORD	Prints an image consisting of only border pixels or of only inside pixels
GRYAPH	Generates a gray level image via quantization of one channel and assignment of alphabetic characters to each level
HEADMD	Modifies or creates header record in universal format
HEADRD	Translates and prints certain information from header record in universal format
HSGRAM	Computes and displays a one, two, or three dimensional histogram
ICD	Given a set of mean vectors and covariance matrices defining a set of spectral signatures, computes the inner class distance - in the likelihood sense - between each pair in the set
IMAGES	Alphabetic image and subset display
IMTPRT	Prints a matrix of integers
INITCL	Initializes the mean vectors and weights for the clustering algorithms
INPSIG	Reads in spectral signatures for use in maximum likelihood classification - MAXLIK
INTHDR	Allows user to process a data tape created by DATDEF which has been saved, reads the header record on the tape and initializes the appropriate variables in the program
ITRCLU	Driver for the iterative clustering option

Table 4.1 Subroutine Function (Continued)

Subroutine	Basic Function
KJOIN	Function that extracts a specified field of bits from one computer word and replaces this field in an image of the second word at specified bit position. Calls the following functions: KPOS, MLU, KSL, and KSR
LISFIL	Lists the heading data for each signature saved on the file
MATPRT	General matrix print routine
MAXLI	Classifies data vectors according to a maximum likelihood algorithm
MAXLIK	Driver for maximum likelihood processing
MINDIS	Given a vector and a set of vectors, determines the vector in the set nearest to first vector and the minimum distance
MODIFY	Computes weighted average of two vectors
MPROD	Performs double precision matrix multiplication
MSHIFT	Changes the method of storage for a matrix to be consistent with its dimensions
NCPRLO	Controls processing of data from universal format tape when each scan line requires more than one record, but one or more channels are in each record
NEWS	Provides a convenient method for communicating minor changes to ASTEP
NOPROC	Skips over channels of a scan line which are not to be used while reading data tape in universal format
NRPCLO	Controls processing of data from universal format tape when each channel requires more than one record

Table 4.1 Subroutine Function (Continued)

Subroutine	Basic Function
NSPRLO	Controls processing of data from universal format tape when there is one or more complete scan lines in one record
OUTREC	Packs data into 8-bit bytes and outputs full records in universal format
PACK	Packs a storage array to eliminate a vacated slot, moves all vectors with index greater than index of vacated slot down one position in the array
PLOT	Given a set of x and y coordinates, creates a plot of the data
PRANDB	Skips over ancillary data for each scan line of a universal format data tape
PROCES	Unpacks data from 8-bit bytes and converts it to floating point numbers
PROJEC	Computes the vector projections required in COMPAR option
QF	Given a matrix and a vector, evaluate the quadratic form vector transpose X matrix X vector
QUANTZ	Generates alphabetic image array from data base via quantizing a single channel and assigning characters to each quantization level
REDREC	Reads a record from the input data tape in universal format
REDSIG	Retrieves one spectral signature from the signature files
REDSIH	Retrieves one or more spectral signatures from the signature files
RETDAT	Retrieves record of data from tape

Table 4.1 Subroutine Function (Continued)

Subroutine	Basic Function
RETIMG	Retrieves image and threshold arrays corresponding to a data record
SAVIMG	Saves, on tape, image and threshold arrays corresponding to a data record
SAVSIG	Saves a spectral signature in the signature files
SCALE	Creates the scale for the PLOT routine
SEQCOV	Sequentially calculates the mean and covariance of a set of vectors
SEQST	Sequentially calculates the mean vector and variances of a set of data vectors
SHUTLE	Computes the average and interclass divergences, selects the best k of n channels using a "Without Replacement" procedure
STATIS	Performs and prints factor analysis for given statistics
SUBSET	Constructs and prints an image of a specified subset of classes or all of the classes on an image data file
SYMINV	Inverts a symmetric matrix
THRDST	Updates the mean and variance of the threshold statistics
TRNFLD	Classifies data according to the field number so that training fields can be found conveniently
TRNSFM	Performs scaling or transforming on data vectors
UNIPRO	Unpacks data by converting each eight-bits to a separate integer value and prints the results
UNIRED	Controls the conversion of data from the universal tape format to the internal ASTEP format

Table 4.1 Subroutine Function (Concluded)

Subroutine	Basic Function
UNITS	Driver for the units option
UPPLT	Updates priority or population list based upon number of points assigned to each cluster
UVWRIT	Controls the conversion of data in internal ASTEP format to the universal format
VATCI	Assigns vectors to existing clusters, does not adjust means of existing clusters, updates weights, means, and variances associated with the actual assignments
WRHMTX	Reads or writes matrix onto temporary storage device
WRSTAT	Reads or writes given statistics onto temporary storage device
WRTDAT	Writes the pair of records which are used to make one entry in a data file in an unpacked ASTEP format
WRTSIG	Writes an input spectral signature in the signature core area

Table 4.2 Subroutine Dependencies

Subroutine	Calls
ASTEPX	ADPCLU, COMPAR, DATDEF, DIFIMG, DUMP, EDTSIG, FAKTOR, FEATSL, HSGRAM, IMAGES, INTHDR, ITRCLU, MAXLIK, NEWS, QUANTZ, TRNFLD, TRNSFM, UNITS, UVWRIT
ADDSIG	REDSIG
ADPCLU	ANGDIS, CLUSTA, ERRPRT, INITCL, MATPRT, RETDAT, SAVIMG
ANGDIS	ANGLE, EDIST
ANGLE	
BEGFIL	
CESCA	CLOS, MODIFY, PACK
CHLSKY	
CLOS	MINDIS
CLUSTA	MODIFY, PACK, SEQST, THRDST, UPPLT
COMPAR	ERRPRT, FACANL, PROJEC, REDSIH
CONVRT	
CPUTIM	
CUBIC	
DATDEF	ERRPRT, HEADRD, UNIRED
DAVIDN	CUBIC, FINT, WRHMTX, WRSTAT
DIF	
DIFIMG	DIF, RETIMG, SAVIMG
DISPLA	
DUMP	ERRPRT, UNIPRO
EDIST	
EDTSIG	ADDSIG, BEGFIL, EIGSIG, LISFIL, MATPRT, REDSIG, SAVSIG, WRTSIG
EIGEN	
EIGSIG	FACANL, MATPRT
ERRPRT	
FACANL	EIGEN
FAKTOR	FACANL, MATPRT, REDSIG, RETDAT, RETIMG, SAVSIG, SEQCOV
FEATSL	DAVIDN, ERRPRT, REDSIH, SHUTLE, WRSTAT
FINT	SHUTLE
FNDVID	
GOBORD	DISPLA, RETIMG
GRYAPH	
HEADMD	KJOIN
HEADRD	
HSGRAM	ERRPRT, RETDAT, RETIMG
ICD	MATPRT, QF, SYMINV
IMAGES	ERRPRT, GOBORD, SUBSET
IMTPRT	
INITCL	ERRPRT, MATPRT
INPSIG	ERRPRT, MATPRT
INTHDR	
ITRCLU	ANGDIS, CESCA, ERRPRT, IMTPRT, INITCL, MATPRT, RETDAT, SAVIMG, VATCI
KJOIN	
LISFIL	
MATPRT	
MAXLI	QF
MAXLIK	ICD, INPSIG, MAXLI, REDSIH, RETDAT, SAVIMG
MINDIS	EDIST
MODIFY	
MPROD	
MSHIFT	
NCPRLO	FNDVID, MATPRT, NOPROC, PRANDB, PROCES, REDREC, WRTDAT
NEWS	
NOPROC	
NRPCLO	FNDVID, MATPRT, PRANDB, PROCES, REDREC, WRTDAT
NSPRLO	FNDVID, MATPRT, NOPROC, PRANDB, PROCES, REDREC, WRTDAT
OUTREC	KJOIN
PACK	
PLOT	CONVRT, SCALE
PRANDB	
PROCES	
PROJEC	MATPRT
QF	
QUANTZ	ERRPRT, GRYAPH, RETDAT, SAVIMG
REDREC	
REDSIG	
REDSIH	MATPRT
RETDAT	
RETIMG	
SAVIMG	
SAVSIG	
SCALE	
SEQCOV	
SEQST	
SHUTLE	CHLSKY, ERRPRT, MPROD, MSHIFT, PLOT, SAVSIG, WRSTAT
STATIS	FACANL, MATPRT
SUBSET	DISPLA, RETIMG
SYMINV	
THRDST	
TRNFLD	RETDAT, SAVIMG
TRNSFM	ERRPRT, MATPRT, REDSIH, RETDAT, WRTDAT
UNIPRO	
UNIRED	NCPRLO, NOPROC, NRPCLO, NSPRLO
UNITS	ERRPRT
UPPLT	
UVWRIT	HEADMD, HEADRD, OUTREC, RETDAT, RETIMG
VATCI	MINDIS, SEQST, THRDST
WRHMTX	
WRSTAT	
WRTDAT	
WRTSIG	ERRPRT, MATPRT

COMMON SUBROUTINES

The following subroutines are common to more than one ASTEP option:

ANGDIS*	REDSIG*
ANGLE *	REDSIH*
EDIST*	RETDAT*
EIGEN	RETIMG*
ERRPRT*	SAVIMG*
FACANL	SAVSIG
HEADRD	SEQCOV
INITCL*	SEQST
MATPRT	SYMINV
MODIFY*	THRDST*
PACK *	WRTDAT

Flow diagrams are included here for those subroutines denoted by *. The flow diagrams for these subroutines are included in order to aid the user in understanding the ASTEP Program.

ANGDIS

ENTER WITH

VM - ARRAY OF VECTORS
NVM - NUMBER OF VECTORS IN VM
ND - DIMENSION OF EACH VECTOR
IDIFF- = 1 COMPUTE EUCLIDEAN DISTANCE
≠ 1 COMPUTE L1 DISTANCE
R - WORKING ARRAY

N=NVM-1
R(NVM,NVM)=0.

COMPUTE ANGLE DISTANCE ARRAY
-ANGLES ARE ABOVE DIAGONAL AND DISTANCES ARE BELOW

FOR J=1,N
1) R(J,J)=0.
I1=J+1
2) FOR I=I1,NVM
a) IF IDISF=1 COMPUTE EUCLIDEAN DISTANCE BETWEEN VECTORS I AND J-STORE IN D
b) IF IDISF≠1 COMPUTE L1 DISTANCE AND STORE IN D
c) COMPUTE ANGLE BETWEEN I AND J VECTORS - STORE IN D
d) R(I,J)=D
R(J,I)=A

EXIT WITH

R - ANGLE DISTANCE ARRAY

ANGLE

ENTER WITH

V1, V2 - TWO VECTORS

ND - DIMENSION OF V1 AND V2

A - ON OUTPUT EQUALS ANGLE BETWEEN V1 AND V2

COMPUTE LENGTHS SQUARED AND INNER PRODUCT

$$D1 = \sum_{I=1}^{ND} V1(I)**2$$

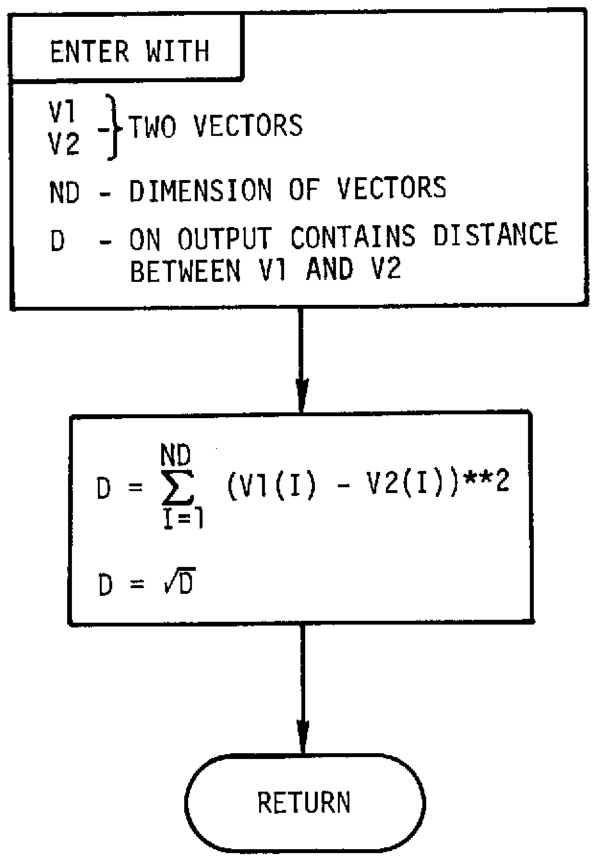
$$D2 = \sum_{I=1}^{ND} V2(I)**2$$

$$A = \sum_{I=1}^{ND} V1(I)*V2(I)$$

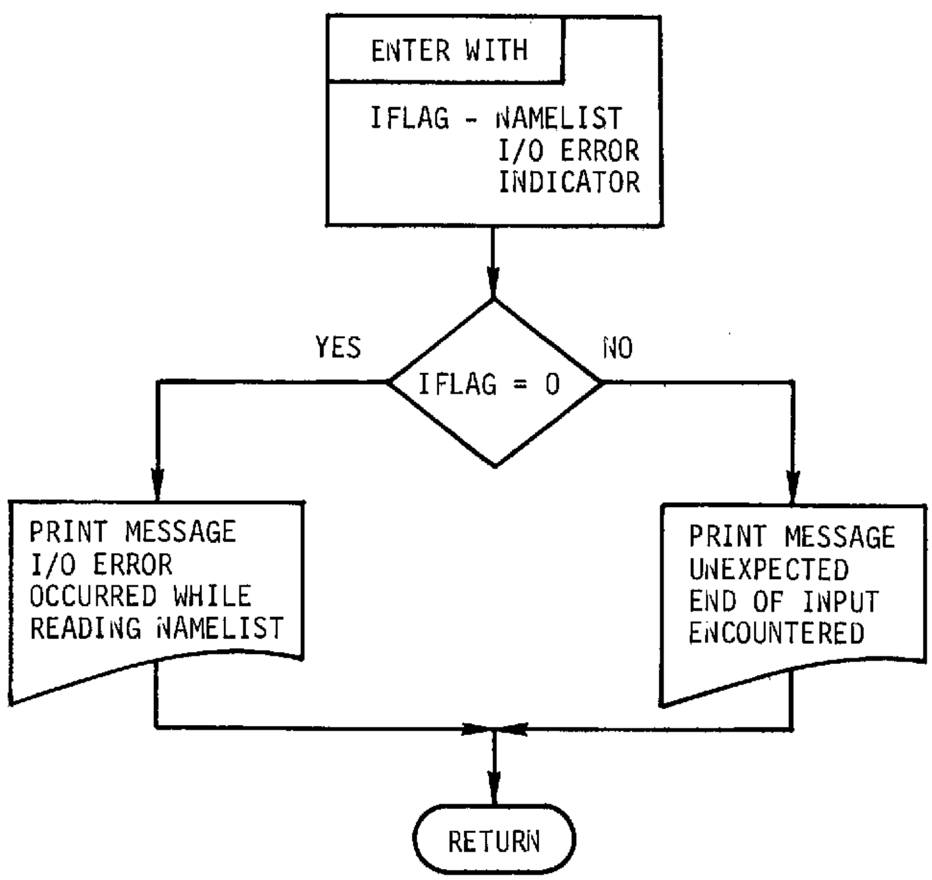
$$A = \frac{360}{2\pi} \cos^{-1} \left(A/(D1*D2)^{1/2}\right)$$

RETURN

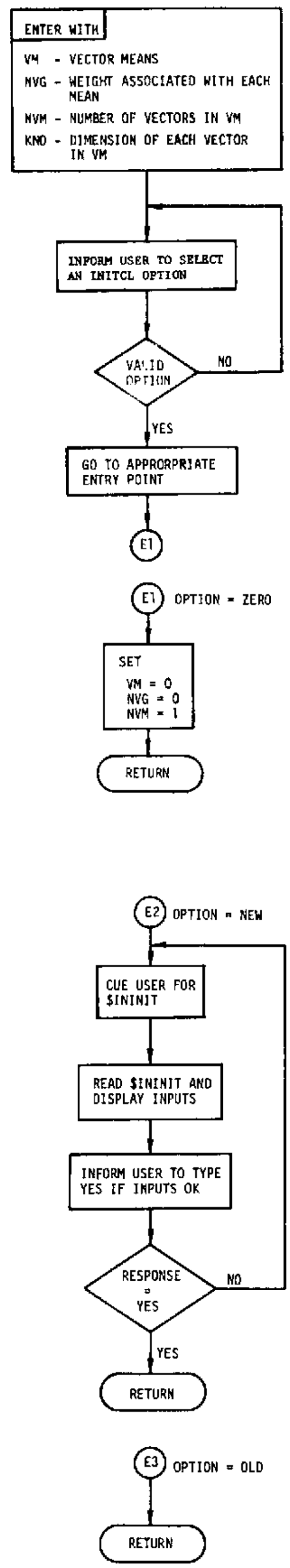
EDIST


ENTER WITH
V1
V2 - TWO VECTORS
ND - DIMENSION OF VECTORS
D - ON OUTPUT CONTAINS DISTANCE BETWEEN V1 AND V2
$D = \sum_{I=1}^{ND} (V1(I) - V2(I))**2$
$D = \sqrt{D}$
RETURN

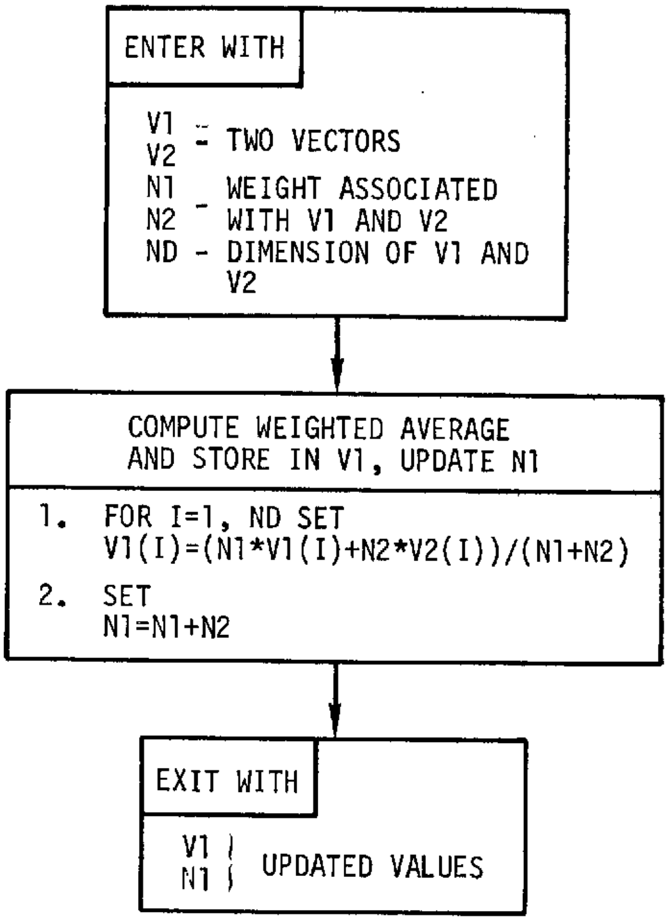
ERRPRT


ENTER WITH
IFLAG - NAMELIST I/O ERROR INDICATOR
IFLAG = 0
YES
NO
PRINT MESSAGE I/O ERROR OCCURRED WHILE READING NAMELIST
PRINT MESSAGE UNEXPECTED END OF INPUT ENCOUNTERED
RETURN

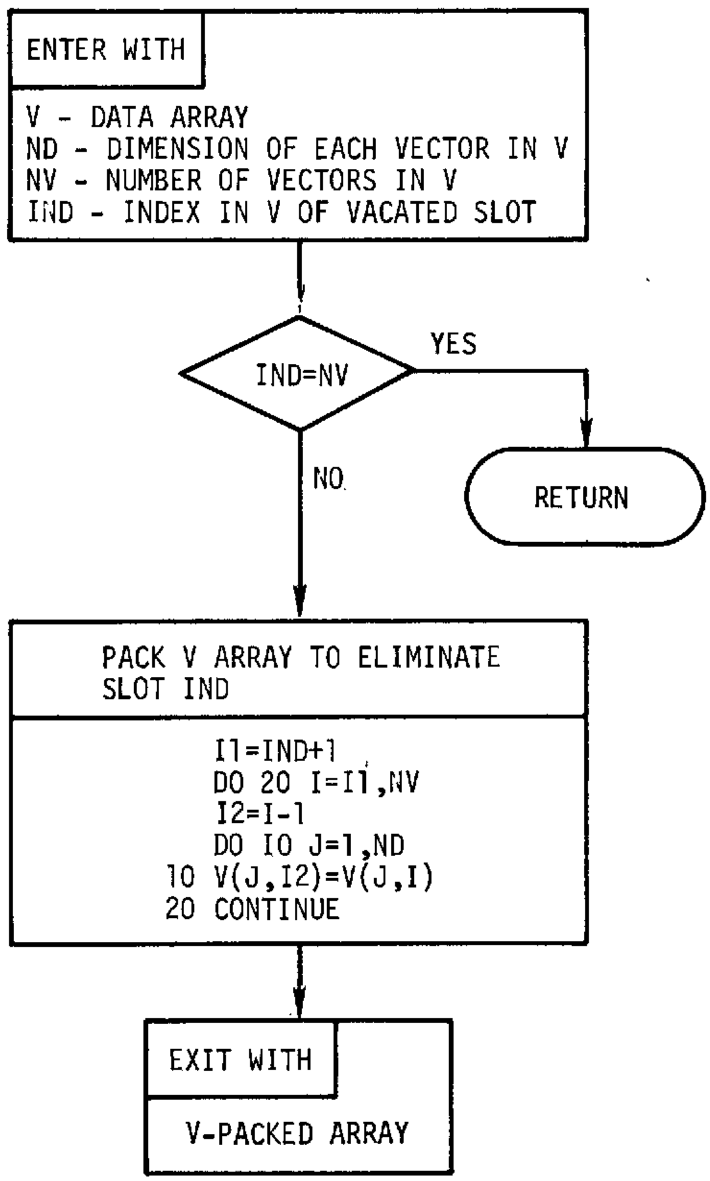
INITCL


ENTER WITH
VM - VECTOR MEANS
NVG - WEIGHT ASSOCIATED WITH EACH MEAN
NVM - NUMBER OF VECTORS IN VM
KNO - DIMENSION OF EACH VECTOR IN VM
INFORM USER TO SELECT AN INITCL OPTION
VALID OPTION
NO
YES
GO TO APPRORPRIATE ENTRY POINT
E1
E1
OPTION = ZERO
SET
VM = 0
NVG = 0
NVM = 1
RETURN
E2
OPTION = NEW
CUE USER FOR $ININIT
READ $ININIT AND DISPLAY INPUTS
INFORM USER TO TYPE YES IF INPUTS OK
RESPONSE = YES
NO
YES
RETURN
E3
OPTION = OLD
RETURN

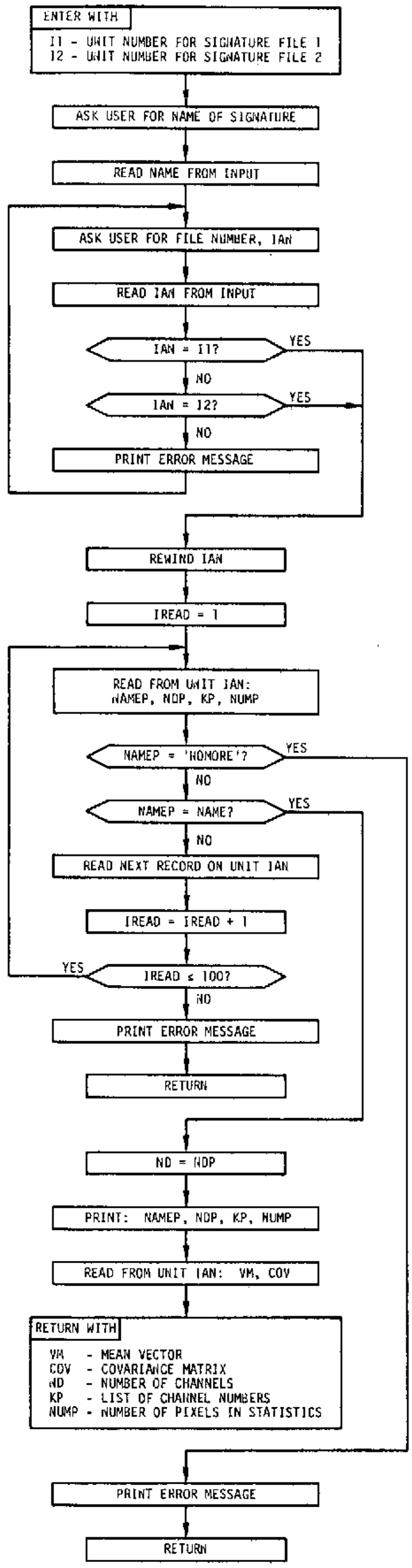
MODIFY


ENTER WITH
V1
V2 - TWO VECTORS
N1
N2 - WEIGHT ASSOCIATED WITH V1 AND V2
ND - DIMENSION OF V1 AND V2
COMPUTE WEIGHTED AVERAGE AND STORE IN V1, UPDATE N1
1. FOR I=1, ND SET
V1(I)=(N1*V1(I)+N2*V2(I))/(N1+N2)
2. SET
N1=N1+N2
EXIT WITH
V1
N1 UPDATED VALUES

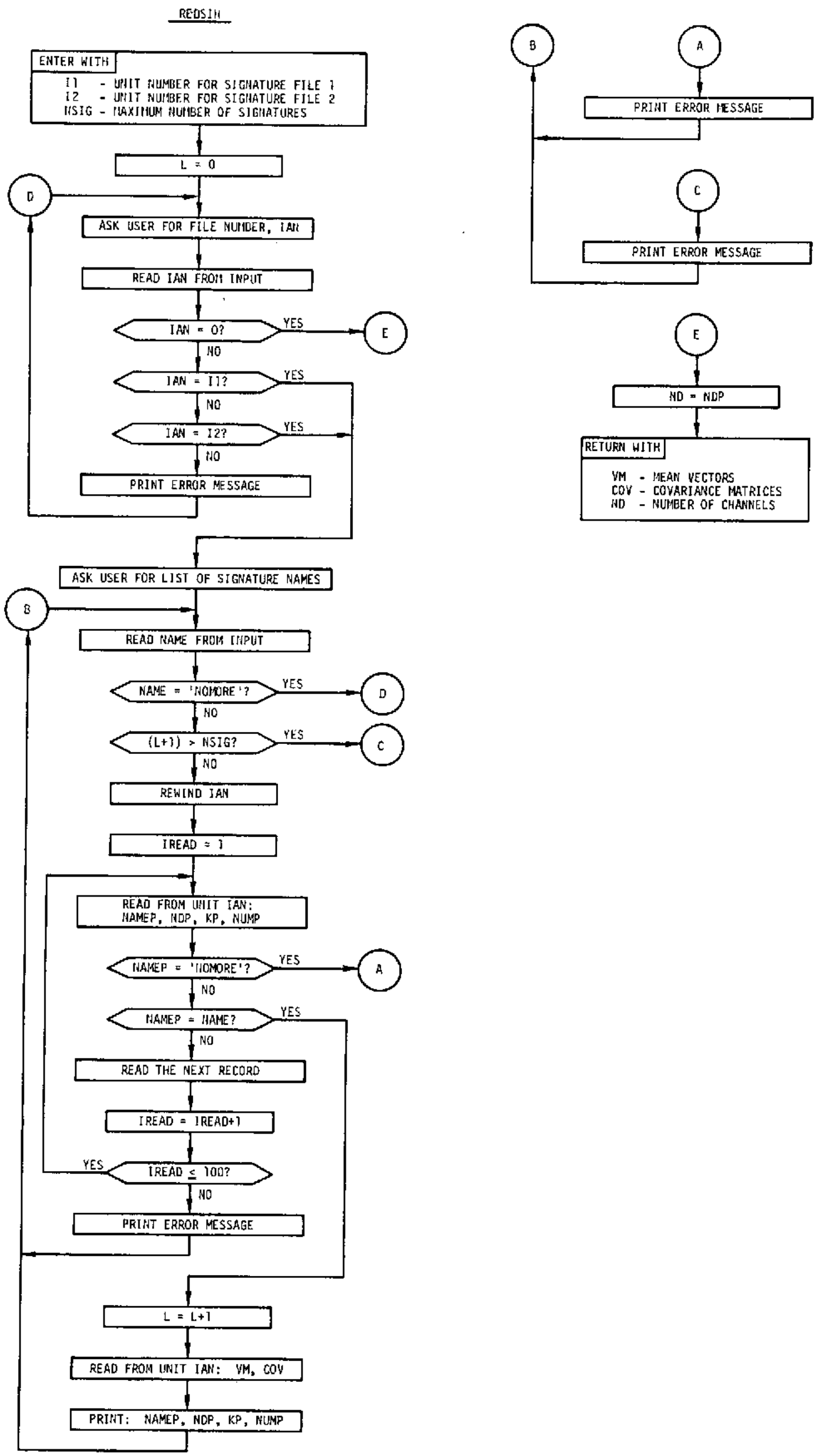
PACK


ENTER WITH
V - DATA ARRAY
ND - DIMENSION OF EACH VECTOR IN V
NV - NUMBER OF VECTORS IN V
IND - INDEX IN V OF VACATED SLOT
IND=NV
YES
NO
RETURN
PACK V ARRAY TO ELIMINATE SLOT IND
I1=IND+1
DO 20 I=I1,NV
I2=I-1
DO 10 J=1,ND
10 V(J,I2)=V(J,I)
20 CONTINUE
EXIT WITH
V-PACKED ARRAY

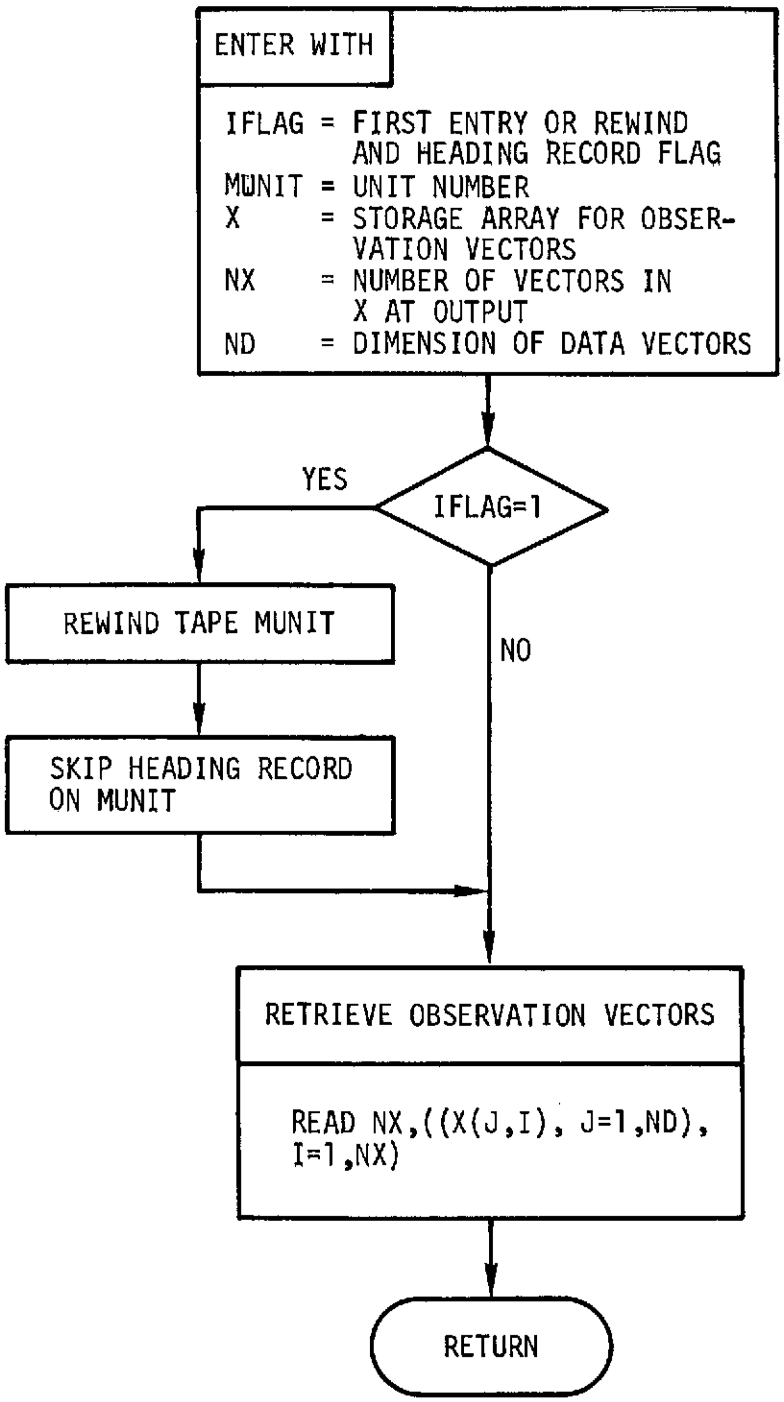
REDSIG



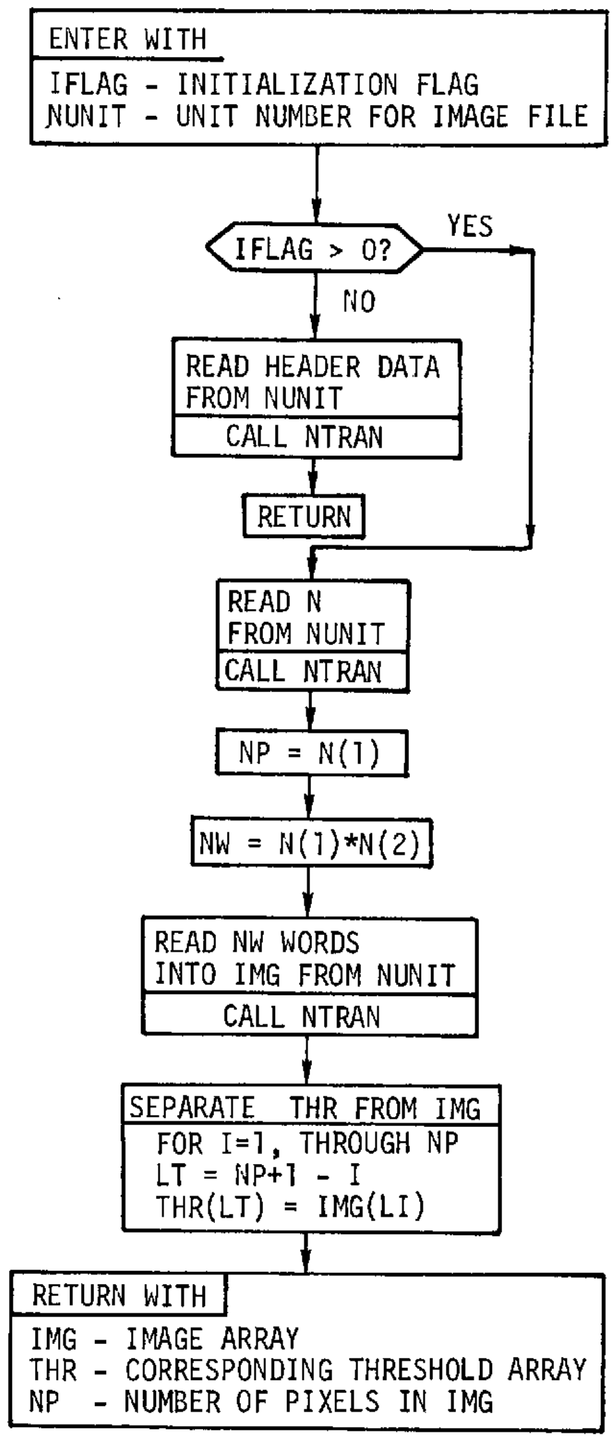
ENTER WITH
I1 - UNIT NUMBER FOR SIGNATURE FILE 1
I2 - UNIT NUMBER FOR SIGNATURE FILE 2
ASK USER FOR NAME OF SIGNATURE
READ NAME FROM INPUT
ASK USER FOR FILE NUMBER, IAN
READ IAN FROM INPUT
IAN = I1?
YES
NO
IAN = I2?
YES
NO
PRINT ERROR MESSAGE
REWIND IAN
IREAD = 1
READ FROM UNIT IAN: NAMEP, NDP, KP, NUMP
NAMEP = 'NOMORE'?
YES
NO
NAMEP = NAME?
YES
NO
READ NEXT RECORD ON UNIT IAN
IREAD = IREAD + 1
YES
IREAD ≤ 100?
NO
PRINT ERROR MESSAGE
RETURN
ND = NDP
PRINT: NAMEP, NDP, KP, NUMP
READ FROM UNIT IAN: VM, COV
RETURN WITH
VM - MEAN VECTOR
COV - COVARIANCE MATRIX
ND - NUMBER OF CHANNELS
KP - LIST OF CHANNEL NUMBERS
NUMP - NUMBER OF PIXELS IN STATISTICS
PRINT ERROR MESSAGE
RETURN


REDSIN
ENTER WITH
I1 - UNIT NUMBER FOR SIGNATURE FILE 1
I2 - UNIT NUMBER FOR SIGNATURE FILE 2
NSIG - MAXIMUM NUMBER OF SIGNATURES
L = 0
D
ASK USER FOR FILE NUMBER, IAN
READ IAN FROM INPUT
IAN = 0?
YES
E
NO
IAN = I1?
YES
NO
IAN = I2?
YES
NO
PRINT ERROR MESSAGE
ASK USER FOR LIST OF SIGNATURE NAMES
B
READ NAME FROM INPUT
NAME = 'NOMORE'?
YES
D
NO
(L+1) > NSIG?
YES
C
NO
REWIND IAN
IREAD = 1
READ FROM UNIT IAN: NAMEP, NDP, KP, NUMP
NAMEP = 'NOMORE'?
YES
A
NO
NAMEP = NAME?
YES
NO
READ THE NEXT RECORD
IREAD = IREAD+1
YES
IREAD ≤ 100?
NO
PRINT ERROR MESSAGE
L = L+1
READ FROM UNIT IAN: VM, COV
PRINT: NAMEP, NDP, KP, NUMP
B
A
PRINT ERROR MESSAGE
C
PRINT ERROR MESSAGE
E
ND = NDP
RETURN WITH
VM - MEAN VECTORS
COV - COVARIANCE MATRICES
ND - NUMBER OF CHANNELS

RETDAT

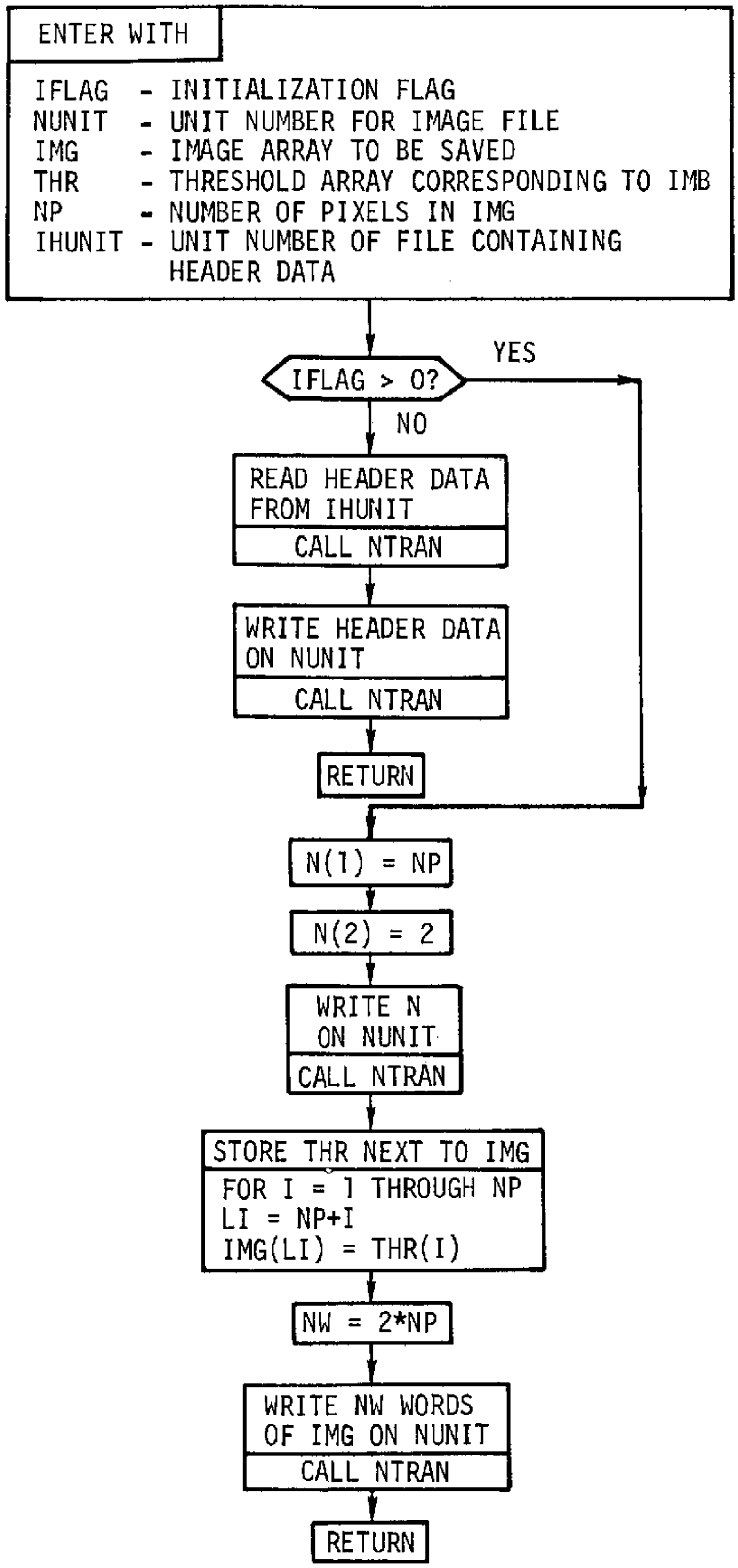

ENTER WITH
IFLAG = FIRST ENTRY OR REWIND AND HEADING RECORD FLAG
MUNIT = UNIT NUMBER
X = STORAGE ARRAY FOR OBSERVATION VECTORS
NX = NUMBER OF VECTORS IN X AT OUTPUT
ND = DIMENSION OF DATA VECTORS
IFLAG=1
YES
NO
REWIND TAPE MUNIT
SKIP HEADING RECORD ON MUNIT
RETRIEVE OBSERVATION VECTORS
READ NX,((X(J,I), J=1,ND), I=1,NX)
RETURN

RETIMG

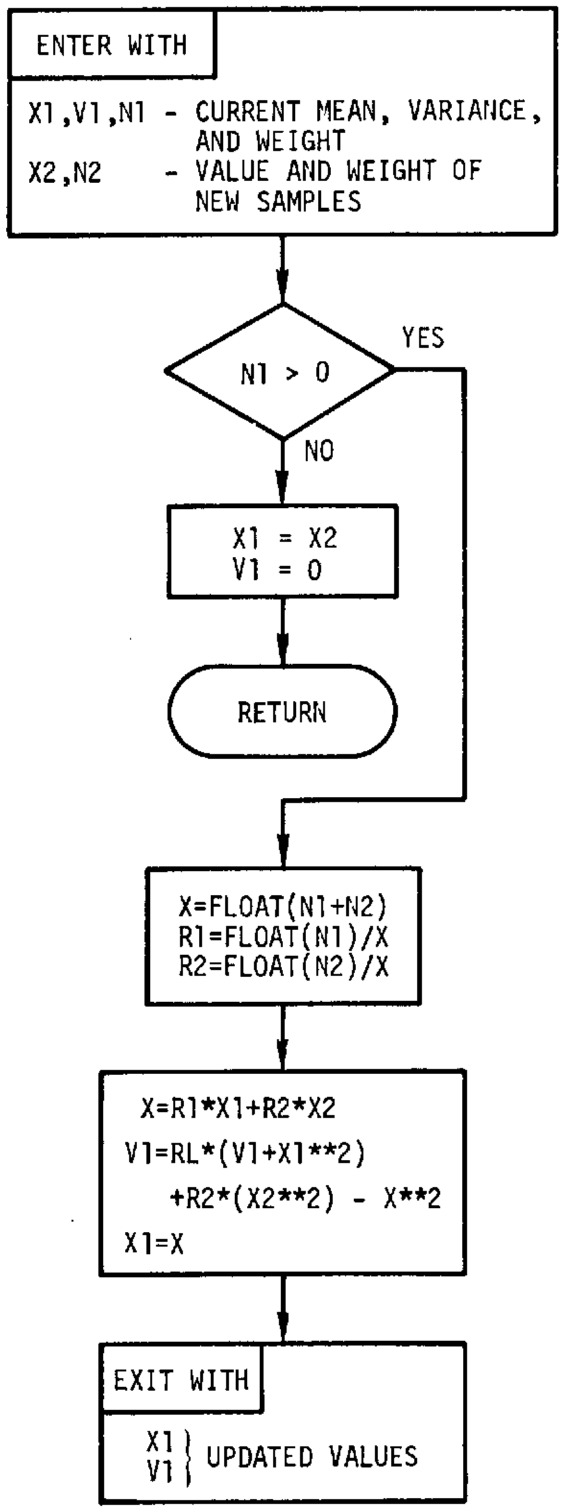


ENTER WITH
IFLAG - INITIALIZATION FLAG
NUNIT - UNIT NUMBER FOR IMAGE FILE
IFLAG > 0?
YES
NO
READ HEADER DATA FROM NUNIT
CALL NTRAN
RETURN
READ N FROM NUNIT
CALL NTRAN
NP = N(1)
NW = N(1)*N(2)
READ NW WORDS INTO IMG FROM NUNIT
CALL NTRAN
SEPARATE THR FROM IMG
FOR I=1, THROUGH NP
LT = NP+1 - I
THR(LT) = IMG(LI)
RETURN WITH
IMG - IMAGE ARRAY
THR - CORRESPONDING THRESHOLD ARRAY
NP - NUMBER OF PIXELS IN IMG

SAVIMG


ENTER WITH
IFLAG - INITIALIZATION FLAG
NUNIT - UNIT NUMBER FOR IMAGE FILE
IMG - IMAGE ARRAY TO BE SAVED
THR - THRESHOLD ARRAY CORRESPONDING TO IMB
NP - NUMBER OF PIXELS IN IMG
IHUNIT - UNIT NUMBER OF FILE CONTAINING HEADER DATA
IFLAG > 0?
YES
NO
READ HEADER DATA FROM IHUNIT
CALL NTRAN
WRITE HEADER DATA ON NUNIT
CALL NTRAN
RETURN
N(1) = NP
N(2) = 2
WRITE N ON NUNIT
CALL NTRAN
STORE THR NEXT TO IMG
FOR I = 1 THROUGH NP
LI = NP+I
IMG(LI) = THR(I)
NW = 2*NP
WRITE NW WORDS OF IMG ON NUNIT
CALL NTRAN
RETURN

THRDST


ENTER WITH
X1,V1,N1 - CURRENT MEAN, VARIANCE, AND WEIGHT
X2,N2 - VALUE AND WEIGHT OF NEW SAMPLES
N1 > 0
YES
NO
X1 = X2
V1 = 0
RETURN
X=FLOAT(N1+N2)
R1=FLOAT(N1)/X
R2=FLOAT(N2)/X
X=R1*X1+R2*X2
V1=RL*(V1+X1**2)
+R2*(X2**2) - X**2
X1=X
EXIT WITH
X1
V1
UPDATED VALUES

5. CONTROL CARDS

This section presents some examples of the control cards required to execute ASTEP in both the batched mode on the EXEC 8 system and the interactive mode on the EXEC 8 system. The control cards required depend upon the particular case to be executed by the user and no example can illustrate all of the possibilities; however, it is hoped that the examples presented will prove helpful as an aid to the user.

In general, the following steps are required to run the program:

Step 1:

Preparation of the data for the run

Step 2:

Selection of program options to be executed and
preparation of program to execute the options

Step 3:

Execution of program options

The first two of these three steps involve assignments of files and/or tapes for data handling within the program, which is accomplished by control cards. The nominal file unit assignments used by ASTEP are given in Table 5.1. The user should be aware of these unit assignments when setting up and executing the program.

In general, Step 1 requires the use of file units 4 and 7. The other unit assignments required depend upon the program options selected in Step 2. Table 5.2 lists the nominal unit assignments required by the various options in the program.

Tables 5.3 - 5.10 illustrate different examples of control card setups for batched runs on the EXEC 8 system. It should be noted that the symbol, @, in the examples is equivalent to a 7/8 punch in column 1 for card input.

Table 5.1 Nominal Unit Assignments

Unit Name	Unit Number	Definition
ISIGF1	1	Signature file 1
ISIGF2	2	Signature file 2
IMGUNT	3	Usual image file
DATUNT	4	Data file in ASTEP format
-	5	Card input file
-	6	Printed output file
OBSUNT	7	Packed MSS data file (LARSC, ERTS, and Universal formats)
OBS1	8	Output file for CPYDAT option
NUNT(3)	9	Special image file used by the DIFIMG option
IHISF1	10	Temporary scratch files used by HSGRAM option
IHISF2	11	
NUNT(2)	12	Special image file used by the DIFIMG option
UVUNIT	14	Output file for the UVWRIT option
NEWSUN	15	File used by NEWS and UPNEWS options
IUB	16	Temporary files used by the FEATSL option
IUC	18	
IUD	19	
NEWDAT*	20	Output file for TRNSFM option

* NOTE: When the TRNSFM option is executed, it sets DATUNT = NEWDAT before returning control to the ASTEP driver.

Table 5.2. Program Options and Required Units

Program Option	Units Required (Nominal Values)
ADPCLU	DATUNT (=4) and IMGUNT (=3)
COMPAR	ISIGF1 (=1) or ISIGF2 (=2) or both
CPYDAT	OBSUNT (=7) and OBS1 (=8)
DATDEF	OBSUNT (=7) and DATUNT (=4)
DIFIMG	NUNT(2) (=12), NUNT(3) (=9), and IMGUNT (=3)
DUMP	UNITNO (Input to DUMP)
EDTSIG	ISIGF1 (=1) or ISIGF2 (=2) or both
FACTOR	DATUNT (=4), IMGUNT (=3), and sometimes ISIGF1 (=1) or ISIGF2 (=2) or both
FEATSL	ISIGF1 (=1) or ISIGF2 (=2) or both
HSGRAM	IHISF1 (=10) and IHISF2 (=11)
IMAGES	IMGUNT (=3) or NUNT(2) (=12) or NUNT(3) (=9)
INTHDR	DATUNT (=4)
ITRCLU	DATUNT (=4) and IMGUNT (=3)
MAXLIK	DATUNT (=4) and IMGUNT (=3)
NEWS	NEWSUN (=15)
QUIT	(No files required)
QUANTZ	DATUNT (=4) and IMGUNT (=3)
TRNFLD	DATUNT (=4) and IMGUNT (=3)
TRNSFM	DATUNT (=4) and NEWDAT (=20)
UNITS	(No files required)
UPNEWS	NEWSUN (=15)
UVWRIT	DATUNT (=4) or IMGUNT (=3) and UVUNIT (=14)

NOTE: When several options are used in combination, some unit numbers may change.

Table 5.3. Example of Control Card Set Required to Use a MSS Data File (Batched Mode - EXEC8 System)

Cards	Explanation
@RUN,/R PSTRWX, 1490T-A025-C, TRW-T49144	Run card
@QUAL TRW-T60655 @ASG,A *LARSC1. @USE 7,*LARSC1.	MSS LARSC1 data is saved on unit 7 under the qualifier symbol TRW-T60655
↕	Other unit assignments required by the ASTEP options to be used should be inserted here
@QUAL TRW-T33710 @XQT *ASTEPT.ASXMAP	Loads ASTEPT which has been previously saved under the qualifier TRW T33710 and begins execution
↑ ASTEP User input cards ↓	The ASTEP options input by user. This set of cards always begins with an ASTEP option and ends with the ASTEP option, QUIT
@FIN	Terminates the input stream begun by the RUN card

Table 5.4. Example of Control Card Set Required to Use an ERTS Physical Tape

Cards	Explanation
@RUN,/R PSTRWX, 1490T-A025-C, TRW-T49144	Run card
@ASG,T ERTS.,8C,T01440 @USE 7,ERTS	ERTS is an arbitrary file name, 8C means that the tape is 7 track (rather than 9 track), T01440 represents the number on the tape reel Assigns ERTS to unit 7
	Other unit assignments required by the ASTEP options to be used should be inserted here
@QUAL TRW-T33710 @XQT *ASTEPT.ASXMAP	Loads ASTEPT which has previously been saved under the qualifier TRW-T33710 and begins execution
ASTEP User input cards	The ASTEP options input by the user. This set of cards always begins with an ASTEP option and ends with the ASTEP option, QUIT.
@REWIND,I ERTS	Rewinds tape and tells operator that the user is finished with the tape.
@FIN	Terminates the input stream begun by the RUN card

Table 5.5. Example of Control Card Set Required to Save a Tape Created by the DATDEF option from a MSS data tape

Cards	Explanation
@RUN,/R PSTRWX, 1490T-A025-C, TRW-T49144	Run card
@ASG,T ERTS.,8C,T01440	ERTS is an arbitrary file name, 8C means that the tape is 7-track (rather than 9-track), T01440 represents the number on the tape reel, ERTS is the input tape
@USE 7,ERTS	Assigns ERTS to unit 7
@ASG,T DATAUP.,8C,XSAVE	Defines the output tape name (DATAUP) and tells the operator that the user wants to save DATAUP
@USE 4,DATAUP	Assigns DATAUP to unit 4
	Other unit assignments required by the ASTEP options to be used should be inserted here
@QUAL TRW-T33710 @XQT *ASTEPT.ASXMAP	Loads ASTEPT which has previously been saved under the qualifier TRW-T33710 and begins execution
ASTEP User input cards	The ASTEP options input by the user. This set of cards always begins with an ASTEP option and ends with the ASTEP option, QUIT
@REWIND,I ERTS @REWIND,I DATAUP	Rewinds the tapes and tells operator that the user is finished with the tapes.
@FIN	Terminates the input stream begun by the RUN card

Table 5.6. Example of Starting With a Tape Created by DATDEF From a MSS Data Tape and Saving an Image Tape

Cards	Explanation
@RUN,/R PSTRWX, 1490T-A025-C, TRW-T49144	Run card
@ASG,T DATAUP.,8C,A12345	DATAUP is an arbitrary file name, 8C means that the tape is 7 track (rather than 9 track), A12345 represents the number on the tape reel, DATAUP is the input tape previously created by the DATDEF option
@USE 4,DATAUP.	Assigns DATAUP to unit 4
@ASG,T IMAGES.,8C,XSAVE	Defines the output tape name (IMAGES) and tells operator that the user wants to save IMAGES
@USE 3,IMAGES.	Assigns IMAGES to unit 3
	Other unit assignments required by the ASTEP options to be used should be inserted here
@QUAL TRW-T33710 @XQT *ASTEP.ASXMAP	Loads ASTEP which has previously been saved under the qualifier TRW-T33710 and begins execution
ASTEP User input cards	The ASTEP options input by the user. This set of cards always begins with an ASTEP option and ends with the ASTEP option, QUIT
@REWIND,I DATAUP @REWIND,I IMAGES	Rewinds tapes and tells operator that the user is finished with the tapes
@FIN	Terminates the input stream begun by the RUN card

Table 5.7. Example of Copying a Subset of a MSS Data Tape on to a File

Cards	Explanation
@RUN,/R PSTRWX, 1490T-A025-C, TRW-T45116	Run card
@ASG,T ERTS.,8C,T01440	ERTS is an arbitrary file name, 8C means that the tape is 7 track (rather than 9 track), T01440 represents the number on the tape reel, ERTS is the input tape
@USE 7,ERTS.	Assigns ERTS to unit 7
@ASG,RP CPYDAT.	Defines the output file name as CPYDAT and designates a public read-only file
@USE 8,CPYDAT.	Assigns CPYDAT to unit 8
	Other unit assignments required by the ASTEP options to be used should be inserted here
@QUAL TRW-T33710 @XQT *ASTEPT.ASXMAP	Loads ASTEPT which has previously been saved under the qualifier TRW-T33710 and begins execution
ASTEP User input cards	The ASTEP options input by the user. This set of cards always begins with an ASTEP option and ends with the ASTEP option,QUIT.
@REWIND,I ERTS	Rewinds tape and tells operator that the user is finished with the tape
@FREE CPYDAT	Catalogs the new file CPYDAT
@FIN	Terminates the input stream begun by the RUN card

Table 5.8. Example of Reading a File Created by the CPYDAT Option with DATDEF

Cards	Explanation
@RUN,/R PSTRWX, 1490T-A025-C, TRW-T63798	Run card
@QUAL TRW-T45116 @ASG,A *CPYDAT. @USE 7,*CPYDAT.	The CPYDAT file, created in example 5.7 under the qualifier TRW-T45116, is assigned to unit 7
@ASG,T TEMP. @USE 4,TEMP.	Assignment of temporary file, TEMP, for use by the DATDEF option Assigns the TEMP file to unit 4
	Other unit assignments required by the ASTEP options to be used should be inserted here
@QUAL TRW-T33710 @XQT *ASTEPT.ASXMAP	Loads ASTEP, which has previously been saved under the qualifier TRW-T33710, and begins execution
ASTEP User input cards	The ASTEP options input by the user. This set of cards always begins with an ASTEP option and ends with the ASTEP option,QUIT.
@FIN	Terminates the input stream begun by the RUN card

Table 5.9. Example of Saving a Signature File

Cards	Explanation
@RUN,/R PSTRWX, 1490T-A025-C, TRW-T49155	Run card
@QUAL TRW-T60655 @ASG,A *LARSC1. @USE 7,*LARSC1.	MSS LARSC1 data is saved on unit 7 under the qualifier TRW-T60655
@ASG,P SIGSIG.	Defines the output file name as SIGSIG and makes it a public (P) file and assigns it the qualifier on the RUN card (TRW-T49155)
@USE 2,SIGSIG	Assigns the SIGSIG file to unit 2
	Other unit assignments required by the ASTEP options to be used should be inserted here
@QUAL TRW-T33710 @XQT *ASTEPT.ASXMAP	Loads ASTEPT, which has been previously saved under the qualifier TRW-T33710 and begins execution
ASTEP User inputs	The ASTEP options input by the user. This set of cards always begins with an ASTEP option and ends with the ASTEP option,QUIT.
@FREE SIGSIG	Catalogs the new file SIGSIG
@FIN	Terminates the input stream begun by the RUN card

Table 5.10. Example of the Use of a Previously Saved Signature File by a Different User Than the User who Created the File, and Creation of a New Signature File

Cards	Explanation
@RUN,/R PSTRWX, 1490T-A025-C, TRW-T63754	Run card
@QUAL TRW-T60655 @ASG,A *LARSC1. @USE 7,*LARSC1.	MSS LARSC1 data is saved on unit 7 under the qualifier TRW T60655.
@QUAL TRW-T49155 @ASG,AX *SIGSIG.	Makes the signature file (SIGSIG), which was created under the qualifier TRW-T49155, available. X denotes exclusive use.
@USE 1,*SIGSIG.	Assigns the SIGSIG file to unit 1
@ASG,P WRTSIG.	Defines the output file name as WRTSIG and makes it a public (P) file and assigns it the qualifier on the RUN card (TRW-T63754)
@USE 2,WRTSIG.	Assigns WRTSIG to unit 2
	Other unit assignments required by the ASTEP options to be used should be inserted here
@QUAL TRW-T33710 @XQT *ASTEP.ASXMAP	Loads ASTEPT, which has been previously saved under the qualifier TRW T33710 and begins execution
ASTEP User input cards	The ASTEP options input by the user. This set of cards always begins with an ASTEP option and ends with the ASTEP option,QUIT.
@FREE WRTSIG	Catalogs the new file WRTSIG
@FIN	Terminates the input stream begun by the RUN card

6. EXAMPLE ASTEP RUNS

This section presents two example ASTEP runs which demonstrate the use of several ASTEP options. Both examples were run from a remote terminal on the UNIVAC 1110 computer using the EXEC 8 system. The interactive mode illustrates user and program responses. In examining the examples, it is noted that the system types the prompting character > in column 1 when it requires a user input.

For both examples, a block of data was selected from the LARSC1 file for analysis. In order for the user to better understand the results of the example ASTEP runs, it is felt that a brief description of the data set selected for the examples would be appropriate.

The LARSC1 file contains 12 channel multispectral scanner data taken from the C1 flight line of test area C in Tippecanoe County, Indiana (Reference 12). For the examples, a block of data was selected corresponding to an area that contained sharp, well defined boundaries between agricultural crops that appeared individually homogeneous in a photograph of the area. The block of data selected consists of elements 10 through 91 on scan lines 600 through 681 of the C1 flight line. An annotated photograph of the area is provided in Figure 6. of Reference 12. Examination of the photograph shows the following:

- The upper left-hand corner is a corn field.
- The upper right-hand corner is a pasture.
- The lower right-hand corner is a soybean field.
- The lower left-hand corner is a wheat field.
- A straight road runs from top to bottom separating the corn and wheat from pasture and soybeans.
- There may also be a road from left to right just below the corn and pasture.

A sketch of the scene is shown in Figure 6.1.

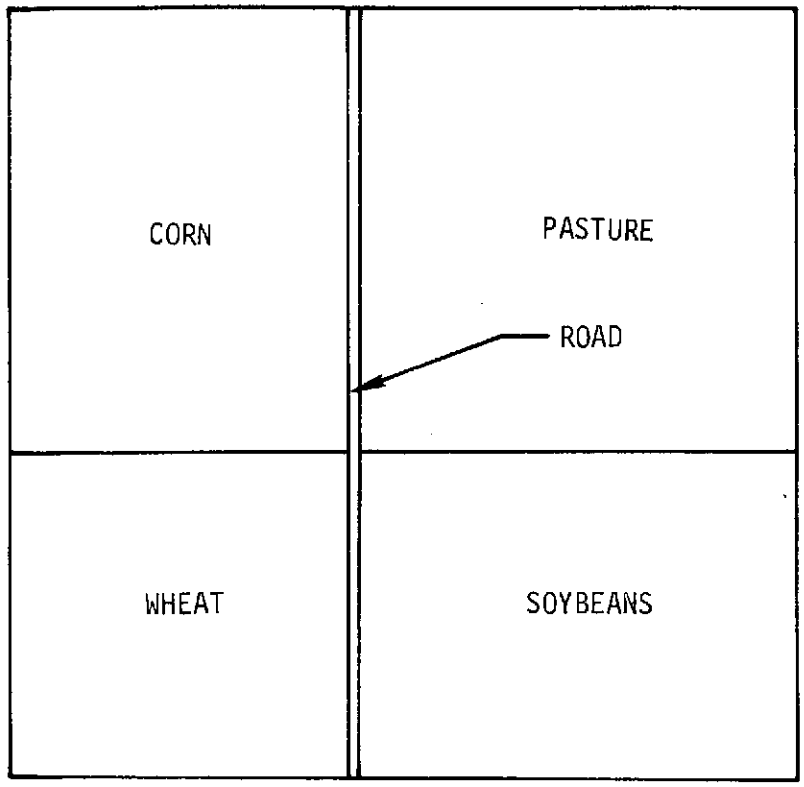


Figure 6.1 Sketch of the ground scene

For this analysis every other element and scan line was skipped such that a total of 1681 data vectors were considered.

EXAMPLE ASTEP RUN 1

In the first example run, the data set is classified using three ASTEP data classification algorithms (ITRCLU, ADPCLU, and MAXLIK). The data is also classified using the QUANTZ option. The results of the various classification options are displayed using the IMAGES option, and the differences between various images are displayed using the DIFIMG and IMAGES options. Based on the preceding analysis of ground truth data, it is expected that the classification schemes in ASTEP would classify the 1681 data points into 4 or 5 major classes.

EXAMPLE ASTEP RUN 2

In the second example run, training fields are selected from the four sub-fields comprising the entire data set. These training fields are shown in Figure 6.2. The signatures (i.e., means and covariances) of the training fields are then computed and saved on a temporary file. Next the entire 12 channel data set is classified using the maximum likelihood classification scheme (MAXLIK). The results of this classification are displayed using the IMAGES option.

Next the best 4 of the original 12 channels and the best 4 linear combinations of the original 12 channels are computed using the FEATSL option. The two transformations for these subsets are computed and saved on a temporary file along with the transformed signatures. The entire 12 channel data set is then transformed and saved on file using both transformations. Both 4 channel data sets are then classified using MAXLIK and the results displayed using the IMAGES option. The differences between the image

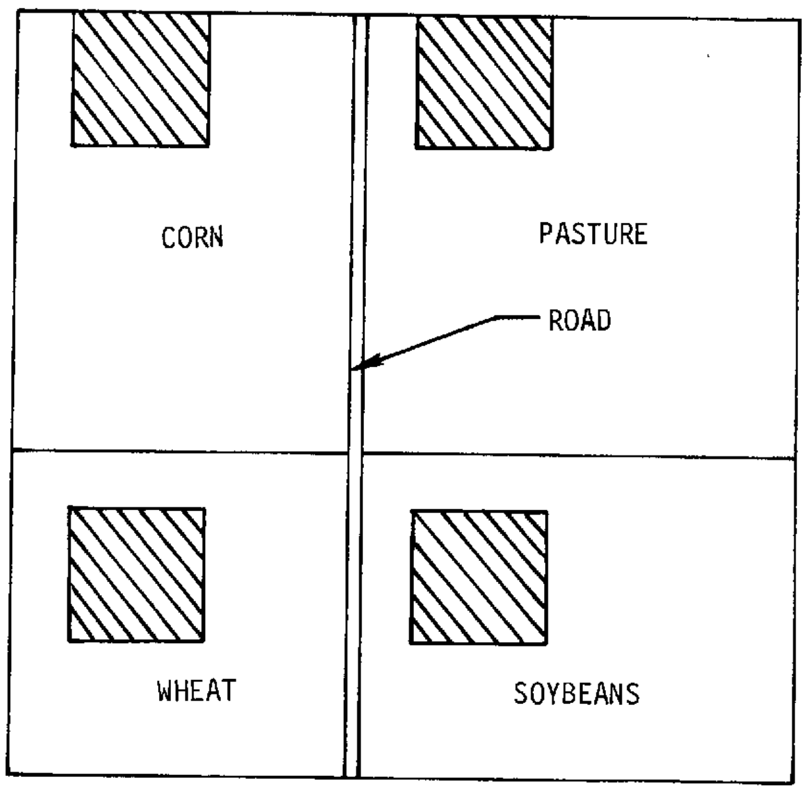


Figure 6.2 Location of training fields in ground scene

corresponding to the 12 channel MAXLIK classification and the two images corresponding to the MAXLIK classification using the best 4 channels and the best 4 linear combinations of the 12 channels are displayed using the DIFIMG and IMAGES options.

The user may note the differences in computation times between the MAXLIK classification of the 12 channel data set and the 4 channel data sets.

The examples presented in this section are not difficult or time consuming to run. Thus, it is suggested that the new user consider duplicating all or part of Example 1 and Example 2 on a terminal in order to become acquainted with the terminal response and methods of input.

The ASTEP program is designed to essentially "talk" the user through a run. If the program requires an input, it will normally type an instruction or list choices. In the case of input, the program will type the first namelist card and the variables which need to be input. As was mentioned previously, when the system requires a user input, the prompting character > will be typed in column 1.

```
IEL033
ENTER USERID/PASSWORD:

*DESTROY USERID/PASSWORD ENTRY
*UNIVAC 1100 OPERATING SYSTEM  VER. 31.159.206EA(RSI)*
>@RUN AGTRWL,1490T-A025-C,TRW-T60655
DATE: 042274       TIME: 152239
>@USE A.,TRW-T33710*TRWDAT.
READY
>@ASG,AX A.
READY
>@COPY,A A.,TPF$.
FURPUR 0026-04/22-15:23
 1 ABS
>@FREE A.
READY
>@USE ASTEPT.,TPF$.
READY
>@USE L.,TRW-T60655*LARSCI.
READY
>@ASG,AX L.
READY
>@USE 7,L.
READY
>@ASG,T 1.,F
READY
>@ASG,T 3.,F
READY
>@ASG,T 4.,F
READY
>@ASG,T 9.,F
READY
>@ASG,T 12.,F
READY
>@XQT ASTEPT.ASXMAP
```

In Example 1, a block of data contained in the LARSCI file is classified using three classification algorithms (ITRCLU, ADPCLU, and MAXLIK). The LARSCI file is assigned to unit 7. The temporary files 1, 3, 4, 9, and 12 are assigned because they are required by the various ASTEP options executed in this run.

```
            ALGORITHM  SIMULATION  TEST  AND  EVALUATION  PROGRAM
            ======================================================
                                  (ASTEP)

                         ASTEP VERSION APR  5 ,1974

 CHOOSE PRINT CONTROL FOR RUN AS
   ECHO    NOECHO
 NOECHO
 ----------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>DATDEF
```

The DATDEF option is called first in order to define the block (or blocks) of data in the LARSCI file that the user wishes to process, and to input the data to the program.

```
                    DATDEF  OPTION
                    **************

$INDATD  NFIELD, ITPFMT, ITPNO, A, B, K, IDEVCE
>  $INDATD NFIELD=1,ITPFMT=1,ITPNO=1,A=255.,B=-1.,K=1,6,9,12 $
 NFIELD    1 ITPFMT  1 ITPNO  1 A   255.0 B    -1.0 IDEVCE  D
 CHANNELS SELECTED,  1  6  9 12
TYPE YES IF INPUTS ARE OK
>YES
$INFLDD ISTART,ISKIP,IINC,JSTART,JSKIP,JINC
>  $INFLDD  ISTART=600,ISKIP=1,IINC=80,JSTART=10,JSKIP=1,JINC=80$
INPUT  1 FIELD DATA
FIELD ISTART ISKIP IINC JSTART JSKIP JINC
  1     600     1    80     10     1    80

TYPE YES IF INPUTS ARE OK
>YES
 THE OPTION DATDEF REQUIRED       5.5898 SECONDS OF CPU TIME.

 ----------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>EDTSIG
```

Later in the example, the data is going to be factored and various signatures saved. However, before this can be done, it is required that a temporary file first be opened up. This is done by calling EDTSIG and using BEGFIL to begin a file. In this case, the file is on unit 1. Had a cataloged file containing signature data been assigned to unit 1, then this EDTSIG and BEGFIL operation would not be required.

```
                    EDTSIG  OPTION
                    **************

CHOOSE EDTSIG OPTION FROM
  BEGFIL  SAVSIG  REDSIG  WRTSIG
  LISFIL  ADDSIG  EIGSIG  PRTSIG
  QUIT
>BEGFIL
BEGFIL OPTION HAS BEEN SELECTED.

CHOOSE FILE NUMBER FROM  1  2
>1

CHOOSE EDTSIG OPTION FROM
  BEGFIL  SAVSIG  REDSIG  WRTSIG
  LISFIL  ADDSIG  EIGSIG  PRTSIG
  QUIT
```

```
>QUIT
QUIT    OPTION HAS BEEN SELECTED.
 THE OPTION EDTSIG REQUIRED         .0654 SECONDS OF CPU TIME.

 ----------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>ITRCLU
```

The data is first classified using the iterative clustering algorithm. This algorithm is activated by the ITRCLU option.

```
                    ITRCLU  OPTION
                    **************

$INITRC    T1,T2,NMIN,NVMMAX,SEP,ISODAT,IDISF,S,P,IP
>  $INITRC  T1=8.,T2=8.5,IP=1,
>   NMIN=5,SEP=1.5,IDISF=1,S=-1.0,P=-1.0 $
$INITRC
T1       =      .80000000E+01
T2       =      .85000000E+01
NMIN     =                 +5
NVMMAX   =                +20
SEP      =      .15000000E+01
ISODAT   =                 +1
IDISF    =                 +1
S        =     -.10000000E+01
P        =     -.10000000E+01
IP       =                 +1

$END
TYPE YES IF INPUTS OK
>YES
ENTER SPLIT/COMBINE (SC) SEQUENCE
>SSSSSCSCS
CHOOSE VALUES FOR INITIALIZATION FROM
ZERO  OLD  NEW
>ZERO
    ***** ITERATION   1-S *****
ITERATION   1, CLUSTER   1, POSITION  3, SPLIT, NVM =    2
    ***** ITERATION   2-S *****
ITERATION   2, CLUSTER   1, POSITION  3, SPLIT, NVM =    3
ITERATION   2, CLUSTER   2, POSITION  4, SPLIT, NVM =    4
    ***** ITERATION   3-S *****
ITERATION   3, CLUSTER   1, POSITION  3, SPLIT, NVM =    5
    ***** ITERATION   4-S *****
ITERATION   4, CLUSTER   1, POSITION  2, SPLIT, NVM =    6
    ***** ITERATION   5-S *****
ITERATION   5, CLUSTER   1, POSITION  3, SPLIT, NVM =    7
    ***** ITERATION   6-C *****
ITERATION   6  CLUSTERS   1 AND   7 MERGED, NVM =   6
ITERATION   6  CLUSTERS   5 AND   6 MERGED, NVM =   5
    ***** ITERATION   7-S *****
ITERATION   7, CLUSTER   1, POSITION  3, SPLIT, NVM =    6
    ***** ITERATION   8-C *****
ITERATION   8  CLUSTERS   1 AND   6 MERGED, NVM =   5
    ***** ITERATION   9-S *****
ITERATION   9, CLUSTER   1, POSITION  3, SPLIT, NVM =    6
     ***** ITERATION  10-   *****
 CLUSTER  SYMBOL  SIZE  R MEAN  R SIGMA    DIFF
     1       A      29   15.12     6.06   14.17
     2       B     438    6.95     3.97     .00
     3       C     428    9.89     2.75     .00
     4       D     476    6.07     3.59     .00
     5       E     247    8.46     4.36    1.16
     6       F      63   12.47     3.57   15.97
 CHOOSE OPTION FROM
   MEANS   ANGDIS  QUIT    SIGMAS
 >QUIT
  THE OPTION ITRCLU REQUIRED      29.3729 SECONDS OF CPU TIME.
```

The ITRCLU option clustered the data into six clusters. The various clusters of points are output with the cluster symbol and size.

```
  ----------------------------------------------------------------

 ENTER ASTEP OPTION OR TYPE A BLANK
 >IMAGES
```

The IMAGES option is called to display the results from the ITRCLU option.

```
                     IMAGES  OPTION
                     **************


  SELECT IMAGES OPTION FROM
     STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
     ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
 >STATUS
```

The STATUS suboption displays the class symbol, image symbol, and number of pixels in each class on the image unit (in this case unit 3).

```
 IMUNIT    THRVAL    MINPIX    IFIELD
    3        24.0        1        0  0  0  0  0  0  0  0  0  0
         CLASS     CLASS     IMAGE     NUMBER
         NUMBER    SYMBOL    SYMBOL    OF PIXELS
           1         A         A          29
           2         B         B         438
           3         C         C         428
           4         D         D         476
           5         E         E         247
           6         F         F          63

  SELECT IMAGES OPTION FROM
     STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
     ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
 >THRESH
```

The THRESH suboption is used to change the threshold value to 100. Pixels with a greater value are ignored.

```
 $INTHRE  THRVAL,MINPIX,IFIELD
 >  $INTHRE THRVAL=100.$
```

```
$INTHRE
THRVAL  =       .10000000E+03
MINPIX  =                 +1
IFIELD  =                 +0,           +0,           +0,           +0,
                          +0,           +0,           +0,           +0,
                          +0,           +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>ALLCLS
```

The ALLCLS suboption displays the pixels in all classes in one image. The class of each pixel is indicated by its image symbol.

```
  IMAGE FOR FIELD   1

         11111222223333344444555556666677777888889
         02468024680246802468024680246802468024680

   600   DDDDDDDDDDDDDDDDDDCBBBBBBBBBBBBBBBBBBBBBB
   602   DDDDDDDDDDDDDDDDDCDBBBBBBBBBBBBBBBBBBBBBB
   604   DDDDDDDDDDDDDDDDDCBBBBBBBBBBBBBBBBBBBBBBB
   606   DDDDDDDDDDDDDDDDDDDBBBBBBBBBBBBBBBBBBBBBB
   608   DDDDDDDDDDDDDDDDDCBBBBBBBBBBBBBBBBBBBBBBB
   610   DDDDDDDDDDDDDDDDDCDBBBBBBBBBBBBBBBBBBBBBB
   612   DDDDDDDDDDDDDDDDDDDBBDBBBBBBBBBBBBBBBBBBB
   614   DDDDDDDDDDDDDDDDDCDBBBBBBBBBBBBBBBBBBBBBB
   616   DDDDDDDDDDDDDDDDDCDDBBBBBBBBBBBBBBBBBBBBB
   618   DDDCDDDDDDDDDDDDDCDBBBBBBBBBBBBBBBBBBBBBB
   620   DDCDDDDDDDDDDDDDDDDBBBBBBBBBBBBBBBBBBBBBB
   622   DDDDCCDDDDDDDDDDDDDBBBBBBBBBBBBBBBBBBBBBB
   624   DDDDCDDDDDDDDDDDDDCBBBBBBBBBBBBBBBBBBBBBB
   626   DDDDDDDDDDDDDDDDDDCBBBBBBBBBBBBBBBBBBBBBB
   628   DDDDDDDDDDDDDDDDDDCBBBBBBBBBBBBBBBBBBBBBD
   630   DDDDDDDDDDDDDDDDDDCBBBBBBBBBBBBBBBDDBBDDD
   632   DDDDDDDDDDDDDDDDDDCBBBBBBBBBBBBBBDBBDDDDD
   634   DDDDDDDDDDDDDDDDDDDBBBBBBBBBBBBBBBDBDDDDB
   636   DDCDDDDDDDDDDDDDDDDBBBBBBBBBBBBBBDBBDDDDD
   638   DDDDDDDDDDDDDDDDDDCBBBBBBBBBBBBBDDDDDDDDD
   640   DDDDDDDDDDDDDDDDDDCBBBBBBBBBBBBDDDDDDDDCD
   642   DDDDDDDDDDDDDDDDDDCBBBBBBBBBBDCECDDDDDCCC
   644   DDDDDDDDDDDDDDDDDDCBBBBBBBDDDDDDBDDDDDDDE
   646   EEEEEFFEEEAAAAAAAFDDCCCCCCCCCCCCCCCCCCCCC
   648   EEEEEEEEEEEEFFFFEECCCCCCCCCCCCCCCCCCCCCCC
   650   EEEEFFEEEEEEFFAFFACCCCCCCCCCCCCCCCCCCCCCC
   652   EEEEFFFEEEFFEEEEFAECCCCCCCCCCCCCCCCCCCCCC
   654   EEEFFFFAEAFEEEEFAAECCCCCCCCCCCCCCCCCCCCCC
   656   EEEFFEFFFFEEEEEFAAFCCCCCCCCCCCCCCCCCCCCCC
   658   EEEFFFFEEEEEEEEEAFFCCCCCCCCCCCCCCCCCCCCCC
   660   EFFFFAEEEEEEEEAFAEFCCCCCCCCCCCCCCCCCCCCCC
   662   EEEEEEFEEEEEFFAFAAFCCCCCCCCCCCCCCCCCCCCCC
   664   EEEEEEEEEEAAFEEEFFFCCCCCCCCCCCCCCCCCCCCCC
   666   EEEEEEEEEEEEEEEEEEECCCCCCCCCCCCCCCCCCCCCC
   668   EEEEEEEEEEEEEEEEEEECCCCCCCCCCCCCCCCCCCCCC
   670   EEEEEEEEEEEEEEEFAFFCCCCCCCCCCCCCCCCCCCCCC
   672   EEEEEEEEEEEFEEAEEEECCCCCCCCCCCCCCCCCCCCCC
   674   EEEEEECEEEEEEEEEEEECCCCCCCCCCCCCCCCCCCCCC
   676   EEECEEEEEEEEFFEFAAFCCCCCCCCCCCCCCCCCCCCCC
   678   EEEEEEEEEEEEEEEEEEECCCCCCCCCCCCCCCCCCCCCC
   680   EEEEEEEEEEEEEEEEEEECCCCCCCCCCCCCCCCCCCCCC


 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>IMQUIT
 THE OPTION IMAGES REQUIRED        .5976 SECONDS OF CPU TIME.
```

The IMQUIT suboption terminates the IMAGE option.

```
 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>FACTOR
```

The FACTOR option is called to compute and save the signatures (i.e., the mean vectors and covariance matrices) of the clusters formed previously by ITRCLU. The signatures are saved on unit 1. The signatures for clusters B, C, and D are saved under the names SIGA, SIGB, and SIGC respectively. The signature for clusters A, E, and F are combined and saved under the name SIGDE.

```
                    FACTOR  OPTION
                    **************

INPUT SYMBOLS FOR CLASSES.
>B
TYPE YES TO PRINT STATISTICS FOR CLASS B
>YES

FACTOR ANALYSIS FOR B
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
                  MEAN        1 BY  4
          1          2           3          4

 1     74.737     82.826      56.037     110.261

                  SIGMAS      1 BY  4
          1          2           3          4

 1      2.090      1.439       1.906      7.356
```

```
                   COVMAT      4 BY  4
          1           2           3           4

 1      4.368        .214        .359       -.189
 2       .693       2.071        .394        .098
 3      1.428       1.082       3.633       -.280
 4     -2.900       1.009      -3.927      54.102

                   ARRAY       4 BY  4
          1           2           3           4

 1     54.600        .851        .851      52.113
 2      5.490        .086        .936      41.290
 3      2.712        .042        .979      84.695
 4      1.371        .021       1.000      77.111

                   EIGENV      4 BY  4
          1           2           3           4

 1      -.059        .728       -.683        .006
 2       .017        .397        .375        .859
 3      -.078        .686        .627       -.509
 4       .996        .089        .002       -.064

TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1

INPUT NAME TO SAVE DATA UNDER
>SIGA
SIGNATURE SIGA   HAS BEEN SAVED ON UNIT 1
SIGA   NO =  4 K =   1  6  9 12
NUM(1) =    438
INPUT SYMBOLS FOR CLASSES.
>C
TYPE YES TO PRINT STATISTICS FOR CLASS C
>NO
FACTOR ANALYSIS FOR C
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO

ZERO HAS BEEN CHOSEN.
TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1

INPUT NAME TO SAVE DATA UNDER
>SIGB
SIGNATURE SIGB   HAS BEEN SAVED ON UNIT 1
SIGB   NO =  4 K =   1  6  9 12
NUM(1) =    428
INPUT SYMBOLS FOR CLASSES.
>D
TYPE YES TO PRINT STATISTICS FOR CLASS D
>YES

FACTOR ANALYSIS FOR D
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
                   MEAN        1 BY  4
          1           2           3           4

 1     83.586      85.908      61.162      81.395

                   SIGMAS      1 BY  4
          1           2           3           4

 1      3.464       2.211       2.750       5.867

                   COVMAT      4 BY  4
          1           2           3           4

 1     13.426        .399        .114       -.506
 2      3.191       4.888        .622        .055
 3      1.147       3.779       7.563       -.074
 4    -10.868        .706      -1.198      34.307

                   ARRAY       4 BY  4
          1           2           3           4

 1     39.032        .649        .649      76.597
 2     12.105        .201        .850      15.645
 3      7.607        .126        .976      97.593
 4      1.440        .024       1.000      87.756

                   EIGENV      4 BY  4
          1           2           3           4

 1      -.395        .548       -.661       -.291
 2      -.024        .558        .132        .819
 3      -.052        .632        .698       -.477
 4       .917        .289       -.241       -.131

TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1
```

```
INPUT NAME TO SAVE DATA UNDER
>SIGC
SIGNATURE SIGC    HAS BEEN SAVED ON UNIT 1
SIGC    NO =   4 K =    1   6   9  12
NUM(1) =    476
INPUT SYMBOLS FOR CLASSES.
>AEF
TYPE YES TO PRINT STATISTICS FOR CLASS AEF
>YES

FACTOR ANALYSIS FOR AEF
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
                MEAN          1 BY  4
         1          2           3            4

 1     81.221    103.192    101.292      78.168

                SIGMAS        1 BY  4
         1          2           3            4

 1      3.055     10.186     10.388       7.584

                COVMAT        4 BY  4
         1          2           3            4

 1      9.333       .712        .671         .401
 2     22.156    103.755        .896         .858
 3     10.115     94.765     107.841         .791
 4     13.930     66.302      62.308       57.549

                ARRAY         4 BY  4
         1          2           3            4

 1    248.937       .894        .894       18.924
 2     16.129       .058        .952       84.866
 3      9.646       .035        .986       83.379
 4      3.765       .014       1.000       73.165

                EIGENV        4 BY  4
         1          2           3            4

 1       .131       .188        .413         .885
 2       .630       .183        .627        -.420
 3       .631      -.687       -.311         .182
 4       .433       .683       -.583         .078

TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM   1   2
>1

INPUT NAME TO SAVE DATA UNDER
>SIGDE
SIGNATURE SIGDE   HAS BEEN SAVED ON UNIT 1
SIGDE   NO =   4 K =    1   6   9  12
NUM(1) =    339
INPUT SYMBOLS FOR CLASSES.
>
 THE OPTION FACTOR REQUIRED        2.3770 SECONDS OF CPU TIME.

 ------------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK

>UNITS
```

The UNITS option is called to cycle the image unit number. The image generated by ITRCLU currently resides on image unit number 3.

```
                    UNITS   OPTION
                    **************


$AVINU
N         =                 +1,              +12,              +9

$END
$INUNIT
IMGUNT  =                  +3
DATUNT  =                  +4
OBSUNT  =                  +7
ISIGF1  =                  +1
ISIGF2  =                  +2
IHISF1  =                 +10
IHISF2  =                 +11
IMG1    =                  +0
IMG2    =                  +0
OBS1    =                  +8

$END
CHOOSE OPTION FROM
  CYCLE    CHANGE   QUIT
>CYCLE
```

The CYCLE option cycles the unit number on which the next image generated will be saved to the next available image unit number. In this case, unit 12.

```
$INUNIT
IMGUNT  =                 +12
DATUNT  =                  +4
OBSUNT  =                  +7
ISIGF1  =                  +1
ISIGF2  =                  +2
IHISF1  =                 +10
IHISF2  =                 +11
IMG1    =                  +0
IMG2    =                  +0
OBS1    =                  +8

$END
```

```
THE OPTION UNITS  REQUIRED       .0484 SECONDS OF CPU TIME.
----------------------------------------------------------------

ENTER ,STEP OPTION OR TYPE A BLANK
>ADPCLU
```

The data is classified using the adaptive clustering algorithm. This algorithm is activated by the ADPCLU option.

```
                    ADPCLU  OPTION
                    --------------


$INADPC   C,S,RP,R1,R2,NVMMAX,NPT,NET,NMT,NMIN,IP
>  $INADPC  C=18.,S=2.,RP=15.,R1=3 .,R2=30.,NMIN=5,IP=1
>    NVMMAX=20,NPT=100,NET=500,NMT=100 $
$INADPC
C        =     .18000000E+02
S        =     .20000000E+01
RP       =     .15000000E+02
R1       =     .30000000E+02
R2       =     .30000000E+02
NVMMAX   =               +20
NPT      =              +100
NET      =              +500
NMT      =              +100
NMIN     =                +5
IP       =                +1

$END
TYPE YES IF INPUTS OK
>YES
CHOOSE VALUES FOR INITIALIZATION FROM
ZERO  OLD  NEW
>ZERO
CLUSTER   5 WEIGHT    2 ELIMINATED, JPT =  1000 NVM =   9
CLUSTER   6 WEIGHT    4 ELIMINATED, JPT =  1000 NVM =   8
CLUSTER   6 WEIGHT    3 ELIMINATED, JPT =  1000 NVM =   7
CLUSTER   6 WEIGHT    5 ELIMINATED, JPT =  1000 NVM =   6
CLUSTER   9 WEIGHT    1 ELIMINATED, JPT =  1500 NVM =  10
MERGER J2 =  8 NJ2 =    9 J1 =  5 NJ1 =  169 NVM =  9 JPT =      0
CLUSTER  SYMBOL  SIZE  R MEAN  R SIGMA    DIFF
    1       A       6     .00      .00     .00
    2       B     464   10.72     5.46     .20
    3       C     426    8.02     4.45     .29
    4       D     449   10.18     5.38     .10
    5       E     168   10.31     4.52     .49
    6       F      57   12.70     4.74    4.49
    7       G      22   19.69     5.33    2.77
    8       H      89    8.73     3.45     .18
CHOOSE OPTION FROM
  MEANS   SIGMAS  ANGDIS  QUIT
>QUIT
 THE OPTION ADPCLU REQUIRED       3.1020 SECONDS OF CPU TIME.
```

The ADPCLU option clustered the data into eight clusters. The various clusters of points are output with the cluster symbol and size.

```
 ----------------------------------------------------------------

ENTER ,STEP OPTION OR TYPE A BLANK
>IMAGES
```

The IMAGES option is then called to display the results from the ADPCLU option. All classes are displayed.

```
                    IMAGES  OPTION
                    --------------


 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>STATUS
IMUNIT   THRVAL   MINPIX   IFIELD
  12       24.0      1        0  0  0  0  0  0  0  0  0  0
      CLASS    CLASS    IMAGE    NUMBER
      NUMBER   SYMBOL   SYMBOL   OF PIXELS
         1        A        A         6
         2        B        B       464
         3        C        C       426
         4        D        D       449
         5        E        E       168
         6        F        F        57
         7        G        G        22
         8        H        H        89

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=30.01$
$INTHRE
THRVAL  =     .30010000E+02
MINPIX  =                +1
IFIELD  =                +0,          +0,          +0,          +0,
                         +0,          +0,          +0,          +0,
                         +0,          +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>ALLCLS
```

```
IMAGE FOR FIELD   1

        11111222223333344444555556666677777888889
        02468024680246802468024680246802468024680

  600   BBBBBBBBBBBBBBBBBCDDDDDDDDDDDDDDDDDDDDDDD
  602   BBBBBBBBBBBBBBBBCBDDDDDDDDDDDDDDDDDDDDDDD
  604   BBBBBBBBBBBBBBBBBCDDDDDDDDDDDDDDDDDDDDDDD
  606   BBBBBBBBBBBBBBBBBBDDDDDDDDDDDDDDDDDDDDDDD
  608   BBBBBBBBBBBBBBBBBCDDDDDDDDDDDDDDDDDDDDDDD
  610   BBBBBBBBBBBBBBBBCBDDDDDDDDDDDDDDDDDDDDDDD
  612   BBBBBBBBBBBBBBBBBBDDDDDDDDDDDDDDDDDDDDDDD
  614   BBBBBBBBBBBBBBBBCBDDDDDDDDDDDDDDDDDDDDDDD
  616   BBBBBBBBBBBBBBBBCBDDDDDDDDDDDDDDDDDDDDDDD
  618   BBBCBBBBBBBBBBBBCBDDDDDDDDDDDDDDDDDDDDDDD
  620   BBCBBBBBBBBBBBBBBBDDDDDDDDDDDDDDDDDDDDDDD
  622   BBBCCBBBBBBBBBBBBBDDDDDDDDDDDDDDDDDDDDDDD
  624   BBBCBBBBBBBBBBBBCBDDDDDDDDDDDDDDDDDDDDDDD
  626   BBBBBBBBBBBBBBBBBCDDDDDDDDDDDDDDDDDDDDDDD
  628   BBBBBBBBBBBBBBBBBCDDDDDDDDDDDDDDDDDDDDDD
  630   BBBBBBBBBBBBBBBBBCDDDDDDDDDDDDDDDDDDDDDDB
  632   BBBBBBBBBBBBBBBBBCDDDDDDDDDDDDDDDDDDDDDBB
  634   BBBBBBBBBBBBBBBBBBDDDDDDDDDDDDDDDDDDDBBBD
  636   BBCBBBBBBBBBBBBBBBDDDDDDDDDDDDDDDDDDDBBBD
  638   BBBBBBBBBBBBBBBBBCBDDDDDDDDDDDDDDBBBBBBBB
  640   BBBBBBBBBBBBBBBBBCDDDDDDDDDDDDDDBBBDBBBCB
  642   BBBBBBBBBBBBBBBBBCBDDDDDDDDDC CBBBBBBCCC
  644   BBBBBBBBBBBBBBBBBCDDDDDDBBBBBBDDDBBBBBE
  646   EEEEEFFEEEGGGG G FBCCCCCCCCCCCCCCCCCCCCCBC
  648   EEEEEEEEEEEEFFFFEECCCCCCCCCCCCCCCCCCCCCCC
  650   HEEEEEEEEEEEFGGFFGCCCCCCCCCCCCCCCCCCCCCCC
  652   EHEEFEFEEEEEEEEEEFFECCCCCCCCCCCCCCCCCCCCC
  654   EEHFFFGFEGFEEEEFGGECCCCCCCCCCCCCCCCCCCCCC
  656   HEEE EFFFFEEEEEFG FCCCCCCCCCCCCCCCCCCCCCC
  658   HEEFFFFEEEEEEHEEEFFFCCCCCCCCCCCCCCCCCCCCC
  660   EFFFFFFEEEEEEEEEGFGEFCCCCCCCCCCCCCCCCCCCC
  662   EEEEEEFEEEEEFFFGFGGFCCCCCCCCCCCCCCCCCCCCC
  664   EEHHHHHHHEGFFEEEFFFCCCCCCCCCCCCCCCCCCCCCC
  666   HHHHHHHHHEEEEEEEEEEECCCCCCCCCCCCCCCCCCCCC
  668   EHHHHHHHHHHHEEEEEEEEECCCCCCCCCCCCCCCCCCCC
  670   EHHHHHHHHHEEHEEEGFFCCCCCCCCCCCCCCCCCCCCCC
  672   EHHHHHHHHEEEEFEEEEECCCCCCCCCCCCCCCCCCCCCC
  674   EHHHHHHHHHEEHHEEEEECCCCCCCCCCCCCCCCCCCCCC
  676   HHHHHHHHHEEFEEEFGGFCCCCCCCCCCCCCCCCCCCCCC
  678   HHHHHHHHHEEEEEEEEEEECCCCCCCCCCCCCCCCCCCCC
  680   HHHHHHHHHEEHEEEEEEECCCCCCCCCCCCCCCCCCCCCC
```

Note the similarities and differences between the images generated by the ITRCLU and ADPCLU options.

```
 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>INQUIT
 THE OPTION IMAGES REQUIRED       .6199 SECONDS OF CPU TIME.

 ---------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>UNITS

                  UNITS   OPTION
                  **************


$AVIMUN
N        =              +3,          +12,          +9

$END
$INUNIT
IMGUNT   =              +12
DATUNT   =              +4
OBSUNT   =              +7
ISIGF1   =              +1
ISIGF2   =              +2
IHISF1   =              +10
IHISF2   =              +11
IMG1     =              +0
IMG2     =              +0
OBS1     =              +8

$END
CHOOSE OPTION FROM
  CYCLE   CHANGE   QUIT
>CYCLE
$INUNIT
IMGUNT   =              +9
DATUNT   =              +4
OBSUNT   =              +7
ISIGF1   =              +1
ISIGF2   =              +2
IHISF1   =              +10
IHISF2   =              +11
IMG1     =              +3
IMG2     =              +12
OBS1     =              +8

$END
 THE OPTION UNITS  REQUIRED       .0438 SECONDS OF CPU TIME.

 ---------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
```

In order to compare the images generated by the ITRCLU and ADPCLU options, it is required to use the DIFIMG option. However, first it is necessary to cycle the image unit numbers. This is done by calling the UNITS option.

At this point, the image generated by the ITRCLU option resides on image unit 3. The image generated by the ADPCLU option resides on unit 12.

The CYCLE suboption in the UNITS option cycles the next image number available for the next image generated to unit number 9.

The DIFIMG option is called to generate an image of the differences between the images generated by the ITRCLU and ADPCLU options. A comparison of the subsets between the images indicates that subset A in the first image is the same as subset G in the second image, subset B in the first image is subset D in the second, etc. This information is input to the DIFIMG option through the equivalence lists.

```
>DIFIMG

                    DIFIMG  OPTION
                    **************

INPUT EQUIVALENCE LISTS
>AG
>BD
>CC
>DB
>EEH
>FF
>
CLASS  SYMBOL  SIZE
  1       A       0
  2       B      44
  3       C    1637
CLASS A ARE THOSE PIXELS OF IMAGE 1 THAT ARE NOT IN THE INPUT EQUIVALENCE LIST
CLASS B ARE THOSE PIXELS OF IMAGE 1 DIFFERENT FROM IMAGE 2
CLASS C ARE THOSE PIXELS THE SAME IN IMAGES 1 AND 2

 THE OPTION DIFIMG REQUIRED        .4929 SECONDS OF CPU TIME.

 ---------------------------------------------------------------
```

The IMAGES option is then called to display the results from the DIFIMG option. Only points different from the equivalence lists are displayed.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>IMAGES

                    IMAGES  OPTION
                    **************


 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>STATUS
IMUNIT   THRVAL   MINPIX   IFIELD
   9      29.0       1       0  0  0  0  0  0  0  0  0  0
      CLASS    CLASS    IMAGE     NUMBER
      NUMBER   SYMBOL   SYMBOL   OF PIXELS
        1        A        A          0
        2        B        B         44
        3        C        C       1637

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=1 [illegible]
$INTHRE
THRVAL  =      .10000000E+01
MINPIX  =                +1
IFIELD  =                +0,             +0,             +0,             +0,
                         +0,             +0,             +0,             +0,
                         +0,             +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES
```

The SYMBOL suboption of IMAGES is used to change the image symbols of classes A, B, and C to I, D, and blank respectively. The STATUS suboption is used to demonstrate the change in the image symbols of all the classes to be printed.

```
 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>SYMBOL
 TYPE THE STRING OF  3 IMAGE SYMBOLS DESIRED.
>ID
 CLASS SYMBOL    ABC
 IMAGE SYMBOL    ID
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>STATUS
IMUNIT   THRVAL   MINPIX   IFIELD
   9       1.0       1       0  0  0  0  0  0  0  0  0  0
      CLASS    CLASS    IMAGE     NUMBER
      NUMBER   SYMBOL   SYMBOL   OF PIXELS
        1        A        I          0
        2        B        D         44
        3        C                1637

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>ALLCLS
```

```
IMAGE FOR FIELD  1

      11111222223333344444555556666677777888889
      02468024680246802468024680246802468024680

600
602
604
606
608
610
612                         D
614
616                      D
618
620
622
624                      D
626
628                                           D
630                                      D  D
632                                    D  DD
634                                     D
636                                    D      D
638                      D
640                                      D
642                      D         D D
644                                    DD
646                    D D  D                D
648
650       DD      D
652        D    DD    D
654         DD
656      DD           D
658                  D
660        D
662
664                D
666
668
670                    D
672                D  D
674         D
676     D        D
678
680

SELECT IMAGES OPTION FROM
   STATUS     IMUNIT    THRESH     SYMBOL     ALLCLS
   ECHCLS     SUBSET    BORDER     INSIDE     IMQUIT
>IMQUIT
 THE OPTION IMAGES REQUIRED       1.2320 SECONDS OF CPU TIME.
 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>UNITS
```

At this point in the program, the image resulting from the ITRCLU option is on image unit 3, the image from the ADPCLU option is on unit 12, and the image from the DIFIMG option on unit 9. The UNITS option is called to change the image numbers.

```
                    UNITS   OPTION
                    **************

$AVIMUN
N        =              +3,              +12,              +9

$END
$INUNIT
IMGUNT  =               +9
DATUNT  =               +4
OBSUNT  =               +7
ISIGF1  =               +1
ISIGF2  =               +2
IHISF1  =               +10
IHISF2  =               +11
IMG1    =               +3
IMG2    =               +12
OBS1    =               +8

$END
CHOOSE OPTION FROM
  CYCLE   CHANGE  QUIT
>CHANGE
$INUNIT IMGUNT,DATUNT,OBSUNT,ISIGF1,ISIGF2,IHISF1,IHISF2,IMG1,IMG2,OBS1
>  $INUNIT  IMGUNT=12
$INUNIT
IMGUNT  =               +12
DATUNT  =               +4
OBSUNT  =               +7
ISIGF1  =               +1
ISIGF2  =               +2
IHISF1  =               +10
IHISF2  =               +11
IMG1    =               +3
IMG2    =               +12
OBS1    =               +8

$END
TYPE YES IF INPUTS OK
>YES
 THE OPTION UNITS  REQUIRED        .0784 SECONDS OF CPU TIME.
 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
```

The CHANGE suboption is called to reset the next available image unit number to 12. This is done to save the image on unit 3 from being wiped out by the next image to be generated (by the MAXLIK option).

The user wants to save the image on unit 3 (generated from the results of the ITRCLU option) so it can be used to difference images later.

The MAXLIK option is called to classify the data once again using the signatures previously saved by the FACTOR option. First, the saved signatures are retrieved by using the REDSIG suboption.

```
>MAXLIK

                  MAXLIK  OPTION
                  **************

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>REDSIG

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>1

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  1
>YES

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>SIGA

SIGA   ND =  4 K =   1  6  9 12
NUM(1) =    438

               MEAN       1 BY  4
        1         2          3          4

 1    74.737    82.826     56.037    110.251


               COVMAT     4 BY  4
        1         2          3          4

 1     4.368      .643     1.428     -2.900
 2      .643     2.071     1.082      1.009
 3     1.428     1.082     3.633     -3.927
 4    -2.900     1.009    -3.927     54.102

>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>1

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  1
>NO

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>SIGB

SIGB   ND =  4 K =   1  6  9 12
NUM(1) =    428
>SIGC

SIGC   ND =  4 K =   1  6  9 12
NUM(1) =    476
>SIGDE

SIGDE  ND =  4 K =   1  6  9 12
NUM(1) =    339
>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
  4 SIGNATURES HAVE BEEN RETRIEVED

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG
```

After the signatures have been retrieved, they are used by the PROCSS suboption to classify the data using the maximum likelihood classification scheme. The data is classified into four classes.

```
>PROCSS
               IC DIS     4 BY  4
        1         2          3          4

 1     7.016   135.893     38.095    635.468
 2   169.933     7.156     34.054    111.362
 3    36.748    33.383      8.552    227.064
 4   202.652    47.109     63.208     11.890

CLASS  SYMBOL  SIZE
  1      A      440
  2      B      420
  3      C      475
  4      D      346

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>QUIT
 THE OPTION MAXLIK REQUIRED        4.3550 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

The IMAGES option is then called to display the results from the MAXLIK option. All of the data subsets (i.e., the 4 classes) are displayed.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>IMAGES

                  IMAGES  OPTION
                  **************


 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>STATUS
```

```
IMUNIT   THRVAL   MINPIX   IFIELD
  12        25.0       1       0  0  0  0  0  0  0  0  0  0
     CLASS    CLASS    IMAGE     NUMBER
     NUMBER   SYMBOL   SYMBOL   OF PIXELS
        1        A        A         490
        2        B        B         420
        3        C        C         475
        4        D        D         396

 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     INQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=6.25|$

$INTHRE
THRVAL  =      .62510000E+01
MINPIX  =                +1
IFIELD  =                +0,          +0,          +0,          +0,
                         +0,          +0,          +0,          +0,
                         +0,          +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     INQUIT
>ALLCLS


   IMAGE FOR FIELD  1

        11111222223333344444555556666677777888889
        02468024680246802468024680246802468024680

   600   C    CCCC CCCCCCCCB AAAAAAAAAAAAAAAAAA AA
   602  CCCCCCCCCCCCCCCCC  AAAAAAAAAAAA AA A AAAA
   604  CCCCCCCCCCCCCCCCC   AA AAAAAAAAAAAAAAAAAA
   606  CCCCCCCCCCCCCCCCC  AAAAAAAAAAA AAAAAA  AA
   608  CCCCCCCC   CCCCCC  AAAAAAAAAAAAAAAAAAAAA
   610  CCCCCCCCCLCC CCCC  AAAAAAAAAAAAAAAAAAAAAA
   612  CCCCCCCCC CCCCCC   AAA AAAAAAAAA  AAAAAAA
   614  CCCCCCC CCCCCCCCCB AAAAAAAAAAAAAAAAAAAAAA
   616      CCCCCCCC  CCC    AAAAAAAAAAAAAAAAAAAA
   618   C  CCCCCCCCCCCCC   AAAAAAAAAAAAAAAA AAAA
   620  CC CCCCCCC CCC CCC  AAAAAAAAAAAAAAAAA AAA
   622  CCC   CCCCCCCCCCC C AAAAAAAAAAAAAAA AAAAA
   624   CCC CCCCCCCCCCCCB AAAAAAAAAAAAAAA AAAAA
   626  C CCCCCCCC CCCCCCCB AAAAAAAAAAAAA AAAAAAA
   628  CCCCCCCCCCCCCCCC CB AAAAAA  AAAAA  AA  A
   630  CCCCCCCCCCCCCCCCCCB AAAAAAAAAAAA
   632  CCCCCCCCCCCCCCCCCC  AAAAAAAAAAAAA      C
   634  CCCCCCCCCCCCCCCCCCCAAAAAAAAAAAA A   AC C
   636  CC CC CCCCCCCCCCC    AA AAAAAAA     C C
   638  CCCCCCCCCCCCCCCCCC  AAAAAAAAAAA  C C  CC
   640  C CCCCCCCLCCCCCCC  AAAAAAAAAAA  CC   CC C
   642  C CC CCCCCCCCCCCCC  A AAAAAA      C C
   644  CC CCCCCCCCCCCCCCC AAAAAAACCCC C  C   CC
   646  DDDDDDDDDD          BBBBBBBBBBB BBBBBBB B
   648  DDDDDDDDDDDDD  D D B BBBBBBBBBBBBBBBBBB B
   650  DDDDD DDDDDDD  DD  BBBBBBBBBBB BBBBBBBBB
   652  DDDD  DDDDDDDDDDD  BBBBBBBBBBBBBBBBBBBBB
   654  DDD     DDDDDDDDDDDBBBBBBBBBBBBBBBBBBBBBB
   656  DDDD  DDDDDDDDDD   BB BBBBBBBBBBBBBBBBBBB
   658  DDD   DDDDDDDDDDDDD BBBBBBBBBBBBBBBBBB BB
   660  DD DDDDDDDDDDD D D  BBBBBBBBBBBBBBBBBBBBB
   662  DDDDDDDDDDDDDDD  D  BBBBBBBBBBBBBBBBBBBBB
   664   DDDDDDDDD DD DDDDD BBB BBBBBBBBBBBBBBBBB
   666  DDDDDDDDDDDDDDDDDDD BBBBBBBBBBBBBBBBBB BB
   668  DDDDDDDDDDDDDDDDDDDBBBBBBBBBBBBBBBBBBBBBB
   670  DDDDDDDDDDDDDDDD  D BBBB  BBBBBBBBBB BBBB
   672  DD DDDDDDDDDDDDDDDDBBBBBBBBBBBBBBBBBBBBBB
   674  DDDDDDDDDDDDDDDDDDD BBBBBBBBBBBBBBBBBB  B
   676  DDD DDDDDD DDDDDD   BBBB BBBBBBBBBBBBB BB
   678  DDDDDDDDDDDDDDDDDDD BBBBBB BBB BBBBBBB BB
   680  DDDDD D DDDDDDDDDDDBBBBBBBBBB    BBBBBBB




 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     INQUIT
>INQUIT
 THE OPTION IMAGES REQUIRED        .4912 SECONDS OF CPU TIME.

 ----------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>UNITS

                     UNITS   OPTION
                     **************
```

The UNITS option is called to change the image unit numbers. At this point, the image from the ITRCLU option is on unit 3, the image from the MAXLIK option is on unit 12, and the image from DIFING is on unit 9.

```
$AVIMUN
N        =              +3,            +12,           +9

$END
$INUNIT
IMGUNT  =              +12
DATUNT  =              +4
OBSUNT  =              +7
ISIGF1  =              +1
ISIGF2  =              +2
IHISF1  =              +10
IHISF2  =              +11
IMG1    =              +3
IMG2    =              +12
OBS1    =              +8
```

The DIFIMG option assumes that the two images to be differenced are on units IMG1 and IMG2. The output of DIFIMG will be on IMGUNT.

```
$END
CHOOSE OPTION FROM
  CYCLE   CHANGE  QUIT
>CHANGE
$INUNIT IMGUNT,DATUNT,OBSUNT,ISIGF1,ISIGF2,IHISF1,IHISF2,IMG1,IMG2,OBS1
>  $INUNIT  IMGUNT=9$
$INUNIT
IMGUNT  =              +9
DATUNT  =              +4
OBSUNT  =              +7
ISIGF1  =              +1
ISIGF2  =              +2
IHISF1  =              +10
IHISF2  =              +11
IMG1    =              +3
IMG2    =              +12
OBS1    =              +8
```

The CHANGE suboption is called to reset the next available image unit number to 9. This is done in order to display the differences between the ITRCLU image on unit 3 and the MAXLIK image on unit 12.

```
$END
TYPE YES IF INPUTS OK
>YES
 THE OPTION UNITS  REQUIRED       .0752 SECONDS OF CPU TIME.
 ---------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>DIFIMG

                 DIFIMG  OPTION
                 **************
```

The DIFIMG is called to generate an image of the differences between the images generated by the ITRCLU and MAXLIK options. A comparison of these images indicates the appropriate equivalence lists to be input to DIFIMG.

```
INPUT EQUIVALENCE LISTS
>AD
>BA
>CB
>DC
>ED
>FD
>
CLASS  SYMBOL  SIZE
  1       A       0
  2       B      34
  3       C     1647
CLASS A ARE THOSE PIXELS OF IMAGE 1 THAT ARE NOT IN THE INPUT EQUIVALENCE LIST
CLASS B ARE THOSE PIXELS OF IMAGE 1 DIFFERENT FROM IMAGE 2
CLASS C ARE THOSE PIXELS THE SAME IN IMAGES 1 AND 2

 THE OPTION DIFIMG REQUIRED       .5694 SECONDS OF CPU TIME.

 ---------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>IMAGES

                 IMAGES  OPTION
                 **************
```

The IMAGES option is then called to display the results from the DIFIMG option. Only the points difference from the equivalence lists are displayed.

```
 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>STATUS
IMUNIT   THRVAL   MINPIX    IFIELD
   9      24.0      1          0  0  0  0  0  0  0  0  0  0
      CLASS     CLASS     IMAGE      NUMBER
      NUMBER    SYMBOL    SYMBOL   OF PIXELS
        1         A         A           0
        2         B         B          34
        3         C         C        1647

 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=1.0$
$INTHRE
THRVAL  =    .10000000E+01
MINPIX  =              +1
IFIELD  =              +0,            +0,            +0,            +0,
                       +0,            +0,            +0,            +0,
                       +0,            +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES
```

```
 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>SYMBOL
 TYPE THE STRING OF  3 IMAGE SYMBOLS DESIRED.
>ID
 CLASS SYMBOL   ABC
 IMAGE SYMBOL   ID
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>STATUS
IMUNIT   THRVAL   MINPIX   IFIELD
   9        1.0      1       0  0  0  0  0  0  0  0  0  0
      CLASS    CLASS    IMAGE    NUMBER
      NUMBER   SYMBOL   SYMBOL   OF PIXELS
        1        A        I         0
        2        B        D         34
        3        C                1647

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>ALLCLS


  IMAGE FOR FIELD  1

         1111122222333334444455555666667777788889
         0246802468024680246802468024680246802460

    400
    602                                D
    604                   D
    606
    608
    610
    612                  D    D
    614
    616                  D D
    618
    620    D
    622
    624
    626
    628                          D
    630                                   D  D
    632                                 D  DD
    634                                 DD
    636                    D            D D
    638                   D
    640                  D          D    D   D
    642                   D       DD  D
    644                                 D
    646                  D
    648
    650                  D
    652
    654
    656
    658
    660
    662
    664                   D
    666                   D
    668
    670
    672
    674       D
    676    D
    678
    680




 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>INQUIT
 THE OPTION IMAGES REQUIRED      1.3468 SECONDS OF CPU TIME.

 ----------------------------------------------------------
```

The QUANTZ option is called to generate a gray scale map of the data. Data points are classified according to the magnitude of one component of the data vectors.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>QUANTZ

                     QUANTZ  OPTION
                     ==============

$INQUAN   XMIN,XMAX,NQ,KCH
>  $INQUAN  XMIN=0.,XMAX=255.,NQ=16,KCH=2$
$INQUAN
XMIN     =      .00000000E+00
XMAX     =      .25500000E+03
NQ       =                +16
KCH      =                 +2

$END
TYPE YES IF INPUTS OK
>YES
LARGEST VALUE = 142.0   SMALLEST VALUE =  78.0

CLASS  SYMBOL  SIZE  LW. BND.  UP. BND.
  1       A       0       .00       .00
  2       B       0       .00     15.94
  3       C       0     15.94     31.87
  4       D       0     31.87     47.81
  5       E       0     47.81     63.75
  6       F       4     63.75     79.69
  7       G    1274     79.69     95.62
  8       H     343     95.62    111.56
  9       I      49    111.56    127.50
 10       J      11    127.50    143.44
 11       K       0    143.44    159.37
 12       L       0    159.37    175.31
 13       M       0    175.31    191.25
 14       N       0    191.25    207.19
 15       O       0    207.19    223.12
 16       P       0    223.12    239.06
 17       Q       0    239.06    255.00
 18       R       0    255.00
 THE OPTION QUANTZ REQUIRED       .4224 SECONDS OF CPU TIME.

 ------------------------------------------------------------
```

The IMAGES option is then called to display the results from the QUANTZ option. All subsets are displayed by the use of the ALLCLS suboption.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>IMAGES

                     IMAGES  OPTION
                     ==============

 SELECT IMAGES OPTION FROM
    STATUS    INUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=1.0$
$INTHRE
THRVAL  =      .10000000E+01
MINPIX  =                +1
IFIELD  =                +0,          +0,          +0,          +0,
                         +0,          +0,          +0,          +0,
                         +0,          +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    INUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>ALLCLS
```

```
IMAGE FOR FIELD  1

          11111222223333344444555556666677777888889
          02468024680246802468024680246802468024680

 600      FGGGGGGGGGGGGGGGGGHGGGGGGGGGGGGGGGGGGGGGG
 602      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 604      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 606      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 608      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 610      GGGGGGGGGGGGGGGGGGHGGGGGGGGGGGGGGGGGGGGGG
 612      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 614      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 616      GGGGGGGGGGGGGGGGGGHGFGGGGGGGGGGGGGGGGGGGG
 618      GGGGGGGGGGGGGGGGGGGFGGGGGGGGGGGGGGGGGGGGG
 620      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 622      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 624      FGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 626      GGGGGGGGGGGGGGGGGGHGGGGGGGGGGGGGGGGGGGGGG
 628      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 630      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 632      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 634      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 636      GGHGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 638      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 640      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGG
 642      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGHGGGGGGGHGG
 644      GGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGGH
 646      HHHHHHHHHHIJJJJJJHGGGGGGGGGGGGGGGGGHGGGGG
 648      HHHHHHHHHHHHIIIIHHHGGGGGGGGGGGGGGGGGGGGGG
 650      GHHHHHHHHHHHIIJIIJHGGGGGGGGGGGHGGGHHHGGGG
 652      HGHHIHIHHHHHHHHHHIHGGGGGGHGGGHHGHGHHHHHGG
 654      HHGIIIIIHIHHHHHIIIHGGGGGGGHHHHHGHHGHHHGGG
 656      GHHHHHIHHHHHHHHHIJIGHGGGHGGHHGHHHGHHGHGGG
 658      GHHHIHIHHHHHGHHHIHHGHHGHHGHHGGHHGHHGHGGG
 660      HIIIIIHHHHHHHHIIJHHGGGGGGGGGGGHHGGGGGGGGG
 662      HHHHHHHHHHHHIIIIIIHHGGGGGGGHGGHGHGGHGGGHG
 664      HHGGGGGGGGIIIHHHHHHGGGGGGGGGGGGGGGHHHGGGG
 666      HHGGGGHGGHHHHHHHHHHGGGGGHGGGGGGGGHGGHGGGG
 668      HHGGGGHGGHHHHHHHHHHGGGGGGGGGGHGGGHHGGGGGG
 670      HGGGGGHGGHHHHHHHIHHGGGGGGGGGGGGGGHHGHGGGG
 672      HGGGGHGGGHHHHIHHHHHGGGGGGGGHHHGGHHHHGGGGG
 674      HGGGGGGGGGHHHHHHHHHGHGGGGGHHHHHGHHGHHGHGH
 676      HGGGHGGGGHHIHHHIIJIGGGGHHGHHGHHHHHHHHGHGH
 678      HGGGGGGGGGHHHHHHHHHGGGHGHHHHHHHHHHHHHHHHH
 680      GGGGGGGGGHHGHHHHHHHGHGGHGHHHHHHHGGGGGHHH


SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>SUBSET
INPUT CHARACTER FOR PRINT AND IMAGE SYMBOLS FOR CLASSES IN SUBSET.
> HIJ
THE IMAGE SYMBOLS HIJ                          WILL BE PRINTED AS HIJ
TYPE YES IF INPUTS ARE CORRECT.
>YES
```

The image containing only subsets H, I, and J is generated by using the SUBSET suboption followed by an input string starting with a blank character and followed H, I, and J. All other classes are printed as blanks.

```
                    SUBSET HIJ
                 PRINTED AS HIJ

IMAGE FOR FIELD  1

      11111222223333344444555556666677777888889
      02468024680246802468024680246802468024680

400                      H
402
404
406
408
410                     H
412
414
416                     H
418
420
422
424
426                      H
428
430
432
434
436     H
438
440
442                                      H      H
444                                                H
446   HHHHHHHHHHHIJJJJJJH                   H
448   HHHHHHHHHHHHHIIIIHHH
450    HHHHHHHHHHHHIIJIIJH            H  HHH
452   H HHIHIHHHHHHHHHHHIH       H   HH  H HHHH
454   HH IIIIIHIHHHHHHIIIH       HHHHH  HH HHH
456    HHHHHIHHHHHHHHHHIJI H   H  HH HHH  HH H
458    HHHIHIHHHHHH HHHIHH HHH HH HH   HH HH H
460   HIIIIIHHHHHHHHHIIJHH            HH
462   HHHHHHHHHHHHHIIIIIIHH        H  H H  H   H
464   HH         IIIHHHHHH                HHH
466   HH    H  HHHHHHHHHHH     H         H  H
468   HH    H  HHHHHHHHHHH          H    HH
470   H     H  HHHHHHHHIHH               HH H
472   H    H   HHHHIHHHHHH        HHH  HHHH
474   H         HHHHHHHHHH H     HHHHH HH  HH H H
476   H   H    HHIHHHIIJI     HH HH HHHHHHHHH H H
478   H        HHHHHHHHHHH    H HHHHHHHHHHHHHHHHH
480            HH HHHHHHHH H  H HHHHHHHH      HHH


SELECT IMAGES OPTION FROM
    STATUS    INUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>SUBSET
INPUT CHARACTER FOR PRINT AND IMAGE SYMBOLS FOR CLASSES IN SUBSET.
>XHIJ
THE IMAGE SYMBOLS HIJ                                    WILL BE PRINTED AS X
TYPE YES IF INPUTS ARE CORRECT.
>YES
```

The image of all pixels of classes H, I, and J being printed as the character X is generated using the suboption SUBSET. The first character is used as the image symbol for all of the following classes.

```
                    SUBSET HIJ
                 PRINTED AS X

 IMAGE FOR FIELD  1

      11111222223333344444555556666677777888889
      02468024680246802468024680246802468024680

 600                    X
 602
 604
 606
 608
 610                   X
 612
 614
 616                   X
 618
 620
 622
 624
 626                    X
 628
 630
 632
 634
 636    X
 638
 640
 642                                     X      X
 644                                             X
 646  XXXXXXXXXXXXXXXXXX                  X
 648  XXXXXXXXXXXXXXXXXXX
 650   XXXXXXXXXXXXXXXXXX           X   XXX
 652  X XXXXXXXXXXXXXXXXX       X   XX X XXXX
 654  XX XXXXXXXXXXXXXXXX       XXXXX XX XXX
 656   XXXXXXXXXXXXXXXXXX X   X  XX XXX XX X
 658   XXXXXXXXXXX XXXXXX XXX XX XX  XX XX X
 660  XXXXXXXXXXXXXXXXXXX           XX
 662  XXXXXXXXXXXXXXXXXXXX       X  X X  X   X
 664  XX        XXXXXXXXX               XXX
 666  XX    X  XXXXXXXXXX     X        X  X
 668  XX    X  XXXXXXXXXX          X   XX
 670  X     X  XXXXXXXXXX              XX X
 672  X    X   XXXXXXXXXX        XXX  XXXX
 674  X         XXXXXXXXX X     XXXXX XX XX X X
 676  X   X    XXXXXXXXXX    XX XX XXXXXXXX X X
 678  X        XXXXXXXXXX    X XXXXXXXXXXXXXXXX
 680           XX XXXXXXX X  X XXXXXXXX     XXX



SELECT IMAGES OPTION FROM
   STATUS     IMUNIT    THRESH    SYMBOL    ALLCLS
   ECHCLS     SUBSET    BORDER    INSIDE    IMQUIT
>IMQUIT
 THE OPTION IMAGES REQUIRED      2.1174 SECONDS OF CPU TIME.

-------------------------------------------------------

ENTER  STEP OPTION OR TYPE A BLANK
>QUIT

                  QUIT    OPTION
                  --------------


-------------------------------------------------------

THE OPTIONS IN THIS RUN REQUIRED      53.2160 SECONDS OF CPU TIME.

+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

The QUIT option returns control to the operating system. The input @FIN terminates the input stream begun by the @RUN input.

At the termination of an execution the total CPU time required by all the options in the run is printed.

```
TEL 056
ENTER USERID/PASSWORD:

*DESTROY USERID/PASSWORD ENTRY
*UNIVAC 1100 OPERATING SYSTEM  VER. 31.159.206EA(RSI)*
>@RUN AGTPW1,1490T-A025-C,TRW-T60655
DATE: 042274       TIME: 190412
>@USE A.,TRW-T33710*TRWDAT.
READY
>@ASG,AX A.
READY
>@COPY,A A.,TPF$.
FURPUR 0026-04/22-19:04
 1 ABS
>@FREE A.
READY
>@USE ASTEPT.,TPF$.
READY
>@USE L.,TRW-T60655*LARSC1.
READY
>@ASG,AX L.
READY
>@USE 7,L.
READY
>@ASG,T 1.,F
READY
>@ASG,T 2.,F
READY
>@ASG,T 3.,F
READY
>@ASG,T 4.,F
READY
>@ASG,T 9.,F
READY
>@ASG,T 12.,F
READY
>@ASG,T 16.,F
READY
>@ASG,T 18.,F
READY
>@ASG,T 19.,F
READY
>@ASG,T 20.,F
READY
>@XQT ASTEPT.ASXMAP
```

In Example 2, a block of 12 channel data contained in the LARSC1 flight line is classified using the maximum likelihood classification algorithm (MAXLIK). The data is then reduced to 4 channels by the feature selection algorithm and again classified by MAXLIK. The LARSC1 file is assigned to unit 7. Other temporary files required by the options in the run are assigned.

```
           ALGORITHM  SIMULATION  TEST  AND  EVALUATION  PROGRAM
           =====================================================
                                (ASTEP)

                        ASTEP VERSION APR  5 ,1974

CHOOSE PRINT CONTROL FOR RUN AS
  ECHO    NOECHO
>NOECHO
-------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>EDTSIG

                 EDTSIG  OPTION
                 **************
```

The EDTSIG option is called to open up two temporary files (i.e., files 1 and 2) which are to be used later in the run for the storage of various signature information. This is accomplished by calling EDTSIG and using BEGFIL to begin a file.

```
CHOOSE EDTSIG OPTION FROM
  BEGFIL  SAVSIG  REDSIG  WRTSIG
  LISFIL  ADDSIG  EIGSIG  PRTSIG
  QUIT
>BEGFIL
BEGFIL OPTION HAS BEEN SELECTED.

CHOOSE FILE NUMBER FROM  1  2
>1

CHOOSE EDTSIG OPTION FROM
  BEGFIL  SAVSIG  REDSIG  WRTSIG
  LISFIL  ADDSIG  EIGSIG  PRTSIG
  QUIT
>BEGFIL
BEGFIL OPTION HAS BEEN SELECTED.

CHOOSE FILE NUMBER FROM  1  2
>2

CHOOSE EDTSIG OPTION FROM
  BEGFIL  SAVSIG  REDSIG  WRTSIG
  LISFIL  ADDSIG  EIGSIG  PRTSIG
  QUIT
>QUIT
QUIT   OPTION HAS BEEN SELECTED.
 THE OPTION EDTSIG REQUIRED        .0896 SECONDS OF CPU TIME.

 -----------------------------------------------------------
```


PRECEDING PAGE BLANK NOT FILMED

```
ENTER ASTEP OPTION OR TYPE A BLANK
>DATDEF
```

The DATDEF option is called to define and extract four blocks of data from the LARSC1 flight line. These four blocks (or subfields) are to be used as training fields to compute statistics for corn, pasture, wheat, and soybeans.

```
                    DATDEF  OPTION
                    **************

$INDATD  NFIELD, ITPFMT, ITPNO, A, B, K, IDEVCE
>  $INDATD NFIELD=4,ITPFMT=1,ITPNO=1,K=1,2,3,4,5,6,7,8,9,10,11,12
>  A=255.0,B=-1.0
>  $END
 NFIELD     4 ITPFMT   1 ITPNO   1 A    255.0 B     -1.0 IDEVCE  0
 CHANNELS SELECTED.  1  2  3  4  5  6  7  8  9 10 11 12
TYPE YES IF INPUTS ARE OK
>YES
$INFLDD ISTART,ISKIP,IINC,JSTART,JSKIP,JINC
INPUT  1 FIELD DATA
>  $INFLDD ISTART=600,IINC=20,JSTART=10,JINC=20$
INPUT  2 FIELD DATA
>  $INFLDD ISTART=600,IINC=20,JSTART=50,JINC=20$
INPUT  3 FIELD DATA
>  $INFLDD ISTART=655,IINC=20,JSTART=10,JINC=20$
INPUT  4 FIELD DATA
>  $INFLDD ISTART=655,IINC=20,JSTART=50,JINC=20$
FIELD ISTART ISKIP IINC JSTART JSKIP JINC
  1      600     0    20    10     0    20
  2      600     0    20    50     0    20
  3      655     0    20    10     0    20
  4      655     0    20    50     0    20

TYPE YES IF INPUTS ARE OK
>YES
 THE OPTION DATDEF REQUIRED      11.9620 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>TRNFLD
```

The TRNFLD option classifies each field by field number and writes the information on the image file.

```
                    TRNFLD  OPTION
                    **************


          FIELD      CLASS      SYMBOL      PIXELS
            1          1          A           441
            2          2          B           441
            3          3          C           441
            4          4          D           441
 THE OPTION TRNFLD REQUIRED         .1776 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>FACTOR
```

The FACTOR option is called to compute and save the signatures (i.e., the mean vectors and covariances matrices) of the four training fields. The signatures of fields of classes A, B, C, and D are saved on unit 1 under the names of CORN, PAST, WHEAT, and SOY respectively.

```
                    FACTOR  OPTION
                    **************

INPUT SYMBOLS FOR CLASSES.
>A
TYPE YES TO PRINT STATISTICS FOR CLASS A
>YES

FACTOR ANALYSIS FOR A
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
                 MEAN        1 BY 12
          1           2           3           4           5           6

 1     84.778      78.161      59.785      60.107      81.955      85.809
          7           8           9          10          11          12

 1     61.832      75.946      61.277      71.499     102.576      78.481

                 SIGMAS      1 BY 12
          1           2           3           4           5           6

 1      2.620       2.831       1.770       1.744       3.082       2.568
          7           8           9          10          11          12

 1      1.861       3.351       2.679       2.967       7.139       4.613

                 COVMAT     12 BY 12
          1           2           3           4           5           6

 1      6.864        .717        .756        .683        .779        .669
 2      5.318       8.017        .633        .744        .742        .722
 3      3.507       3.171       3.133        .631        .741        .657
 4      3.120       3.673       1.947       3.040        .688        .698
 5      6.292       6.473       4.045       3.695       9.498        .774
 6      4.501       5.249       2.988       3.126       6.123       6.596
 7      3.092       3.379       2.125       2.295       3.674       3.488
 8      5.579       6.320       3.677       4.030       6.913       6.131
 9      4.289       5.171       7.717       3.497       5.049       4.503
10      4.579       5.056       3.130       3.491       5.998       5.602
11     -2.288      -4.470       -.805      -3.351      -1.019       1.683
12     -2.802      -3.287      -1.358      -2.939      -1.946        .767
```

```
            7          8          9          10         11         12

  1       .634       .698       .611       .589      -.122      -.237
  2       .641       .732       .682       .602      -.221      -.257
  3       .645       .681       .573       .596      -.064      -.170
  4       .706       .758       .749       .675      -.249      -.318
  5       .641       .735       .612       .656      -.046      -.190
  6       .772       .783       .655       .735       .042       .066
  7      3.463       .710       .688       .744       .032      -.002
  8      4.032      9.306       .824       .831      -.208      -.236
  9      3.431      6.733      7.173       .784      -.367      -.362
 10      4.109      7.525      6.232      8.805      -.095      -.105
 11       .431     -4.528     -7.019     -2.020     50.963       .719
 12      -.019     -3.249     -4.372     -1.404     23.007     20.368

                 ARRAY      12 BY  4
            1          2          3          4

  1     69.211       .504       .504     86.227
  2     44.238       .322       .827     11.357
  3      8.308       .061       .887     84.089
  4      4.818       .035       .922     85.150
  5      2.441       .018       .940     91.269
  6      1.882       .019       .954     83.870
  7      1.389       .010       .964     88.646
  8      1.302       .009       .973     91.169
  9      1.188       .009       .982     90.734
 10       .988       .007       .989     88.606
 11       .811       .006       .995     88.291
 12       .651       .005      1.000     87.388

                 EIGENV     12 BY 12
            1          2          3          4          5          6

  1      -.138       .278      -.155       .366       .316       .642
  2      -.178       .288      -.037       .332      -.739      -.029
  3      -.078       .186      -.074       .153       .257       .219
  4      -.118       .168      -.007       .003      -.152       .111
  5      -.140       .369      -.102       .426       .367      -.587
  6      -.072       .340       .103       .044      -.161      -.195
  7      -.061       .221       .065      -.105      -.083       .283
  8      -.196       .346       .075      -.252       .036      -.101
  9      -.209       .239       .085      -.358      -.149       .191
 10      -.143       .338       .166      -.520       .257      -.104
 11       .780       .401      -.444      -.154      -.074       .006
 12       .436       .156       .841       .228       .045       .082
            7          8          9          10         11         12

  1      -.170       .143       .013      -.429      -.014      -.007
  2      -.102       .415       .127       .138      -.092      -.027
  3       .029      -.160       .150       .765      -.308       .296
  4       .221      -.180      -.023       .024       .751       .525
  5       .187       .096      -.302       .025       .119      -.152
  6       .153      -.581       .269      -.403      -.402       .213
  7       .510      -.200       .151       .155       .165      -.686
  8      -.743      -.266       .060       .124       .241      -.248
  9       .090      -.305      -.789       .008      -.257       .089
 10       .169       .537       .371      -.083      -.049       .163
 11      -.029       .024      -.054       .009       .014       .017
 12      -.058       .056      -.088       .043       .045       .026

TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1

INPUT NAME TO SAVE DATA UNDER
>CORN
SIGNATURE CORN    HAS BEEN SAVED ON UNIT 1
CORN    ND = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
INPUT SYMBOLS FOR CLASSES.
>B
TYPE YES TO PRINT STATISTICS FOR CLASS B
>NO
FACTOR ANALYSIS FOR B
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1

INPUT NAME TO SAVE DATA UNDER
>PAST
SIGNATURE PAST    HAS BEEN SAVED ON UNIT 1
PAST    ND = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
INPUT SYMBOLS FOR CLASSES.
>C
TYPE YES TO PRINT STATISTICS FOR CLASS C
>NO
FACTOR ANALYSIS FOR C
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1

INPUT NAME TO SAVE DATA UNDER
>WHEAT
```

```
SIGNATURE WHEAT  HAS BEEN SAVED ON UNIT 1
WHEAT  NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
INPUT SYMBOLS FOR CLASSES.
>D
TYPE YES TO PRINT STATISTICS FOR CLASS D
>NO
FACTOR ANALYSIS FOR D
CHOOSE INITIAL STATISTICS FROM ZERO OR FILE, OR QUIT.
>ZERO
ZERO HAS BEEN CHOSEN.
TYPE YES TO SAVE SIGNATURE
>YES

CHOOSE FILE NUMBER FROM  1  2
>1

INPUT NAME TO SAVE DATA UNDER
>SOY
SIGNATURE SOY     HAS BEEN SAVED ON UNIT 1
SOY     NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
INPUT SYMBOLS FOR CLASSES.
>      LEAVE FACTOR
 THE OPTION FACTOR REQUIRED        8.6936 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>DATDEF

                    DATDEF  OPTION
                    **************
```

The DATDEF option is called again to define and extract the test field from the LARSC1 flight line. The test field contains the training fields defined previously (see Figure 6.2 on page 6-3).

```
$INDATD  NFIELD, ITPFMT, ITPNO, A, B, K, IDEVCE
>  $INDATD NFIELD=1,ITPFMT=1,ITPNO=1,K=1,2,3,4,5,6,7,8,9,10,11,12
>  A=255.0,B=-1.0 $
 NFIELD     1 ITPFMT   1 ITPNO   1 A    255.0 B    -1.0 IDEVCE  0
 CHANNELS SELECTED.  1  2  3  4  5  6  7  8  9 10 11 12
TYPE YES IF INPUTS ARE OK
>YES
$INFLDD ISTART,ISKIP,IINC,JSTART,JSKIP,JINC
INPUT  1 FIELD DATA
>  $INFLDD ISTART=600,ISKIP=1,IINC=80,JSTART=10,JSKIP=1,JINC=80$
FIELD ISTART ISKIP IINC JSTART JSKIP JINC
  1     600     1    80    10      1    80

TYPE YES IF INPUTS ARE OK
>YES
 THE OPTION DATDEF REQUIRED        4.8356 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>UNITS

                    UNITS   OPTION
                    **************
```

The UNITS option is called to see the nominal unit numbers for the files used by ASTEP.

```
$AVINUN
N        =                 +3,            +12,            +9

$END
$INUNIT
IMGUNT   =                 +3
DATUNT   =                 +4
OBSUNT   =                 +7
ISIGF1   =                 +1
ISIGF2   =                 +2
IHISF1   =                 +10
IHISF2   =                 +11
IMG1     =                 +0
IMG2     =                 +0
OBS1     =                 +8

$END
CHOOSE OPTION FROM
  CYCLE   CHANGE  QUIT
>QUIT
 THE OPTION UNITS  REQUIRED        .0278 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>MAXLIK

                    MAXLIK  OPTION
                    **************
```

The MAXLIK option is called to classify the test field using the signatures of the four training fields previously saved by the FACTOR option. First the signatures are retrieved by using the REDSIG suboption.

```
CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>REDSIG

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>1
```

```
INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  1
>NO

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>CORN

CORN   NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
>PAST

PAST   NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
>WHEAT

WHEAT  NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
>SOY

SOY    NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
  4 SIGNATURES HAVE BEEN RETRIEVED

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>PROCSS
                IC DIS       4 BY  4
         1          2           3           4

 1     13.356    167.543    533.847     54.660
 2    163.261     10.057   1568.254    520.152
 3    168.684    692.905     19.421    133.955
 4     64.563    488.320    397.696     10.277

CLASS  SYMBOL  SIZE
  1       A     451
  2       B     469
  3       C     356
  4       D     405

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>QUIT
 THE OPTION MAXLIK REQUIRED      23.0256 SECONDS OF CPU TIME.
 ----------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>IMAGES
```

The IMAGES option is called to display the results of the maximum likelihood classification of the 12 channel test field data. The THRESH suboption is called to set the threshold to a large number so that no pixels will be eliminated from the image.

```
                  IMAGES  OPTION
                  **************
```

```
 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=2000 $
$INTHRE
THRVAL  =      .20000000E+04
MINPIX  =                +1
IFIELD  =                +0,          +0,          +0,          +0,
                         +0,          +0,          +0,          +0,
                         +0,          +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>SYMBOL
```

The SYMBOL suboption is called to change the image symbols from A, B, C, D to C, P, W, S corresponding to corn, pasture, wheat and soybeans respectively.

```
 TYPE THE STRING OF  4 IMAGE SYMBOLS DESIRED.
>CPWS
 CLASS SYMBOL   ABCD
 IMAGE SYMBOL   CPWS
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>STATUS
```

The STATUS suboption is called to display the current values of the class symbols, image symbols, etc.

```
IMUNIT   THRVAL   MINPIX   IFIELD
   3     2000.0      1       0  0  0  0  0  0  0  0  0  0
      CLASS    CLASS    IMAGE    NUMBER
      NUMBER   SYMBOL   SYMBOL   OF PIXELS
        1        A        C        451
        2        B        P        469
        3        C        W        356
        4        D        S        406
```

```
SELECT IMAGES OPTION FROM
    STATUS    IHUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>ALLCLS

  IMAGE FOR FIELD  1

      11111222223333344444555556666677777888889
      02468024680246802468024680246802468024680

600   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
602   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
604   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
606   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
608   CCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPPP
610   CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
612   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
614   CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
616   CCCCCCCCCCCCCCCCC#CPPPPPPPPPPPPPPPPPPPPPP
618   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
620   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
622   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
624   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
626   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
628   CCCCCCCCCCCCCCCCCCSPPPPPPPCPPPPPPPPPPPPPC
630   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPC
632   CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPC
634   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPCP
636   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPCP
638   CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPCCCC
640   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPCCPPPC##
642   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPS#SSCCC#SS#
644   CCCCCCCCCCCCCCCCCCCPPPPPPPPCCCCPPPPCCSCC#
646   ###################CSSSSSSSSSSSSSSSSSSSSS
648   ###################SSSSSSSSSSSSSSSSSSSSSS
650   ###################SSSSSSSSSSSSSSSSSSSSSS
652   ###################SSSSSSSSSSSSSSSSSSSSSS
654   ###################SSSSSSSSSSSSSSSSSSSSSS
656   ###################SSSSSSSSSSSSSSSSSSSSSS
658   ####################SSSSSSSSSSSSSSSSSSSSS
660   ####################SSSSSSSSSSSSSSSSSSSSS
662   ###################SSSSSSSSSSSSSSSSSSSSSS
664   ####################SSSSSSSSSSSSSSSSSSSSS
666   ####################SSSSSSSSSSSSSSSSSSSSS
668   ###################SSSSSSSSSSSSSSSSSSSSSS
670   ####################SSSSSSSSSSSSSSSSSSSSS
672   ###################SSSSSSSSSSSSSSSSSSSSSS
674   ###################SSSSSSSSSSSSSSSSSSSSSS
676   ####################SSSSSSSSSSSSSSSSSSSSS
678   ####################SSSSSSSSSSSSSSSSSSSSS
680   ###################SSSSSSSSSSSSSSSSSSSSSS
```

The ALLCLS suboption is called to display the image of all classes. The BORDER and INSIDE suboptions are called to show the boundary and interior pixels of each class in the image.

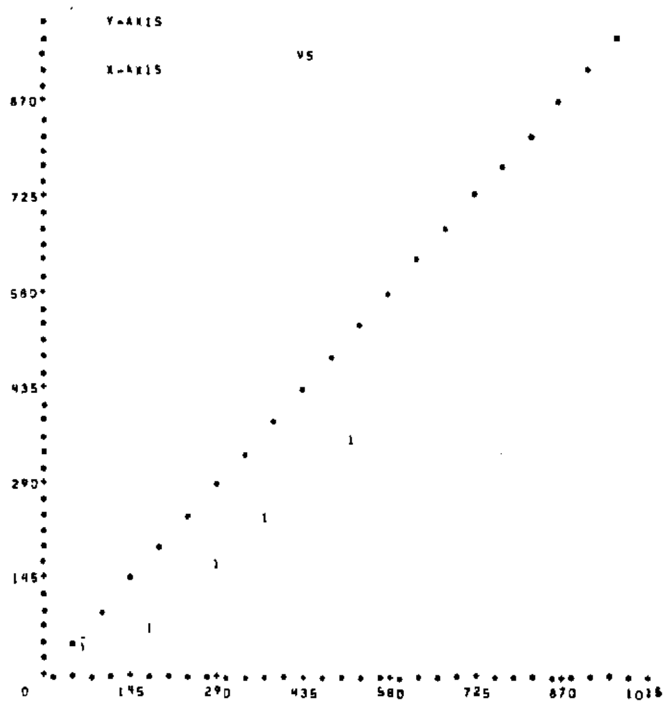
```
SELECT IMAGES OPTION FROM
    STATUS    IHUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>BORDER

  IMAGE FOR ALL CLASSES FOR FIELD  1 WITH ONLY BORDER PIXELS PRINTED

      11111222223333344444555556666677777888889
      02468024680246802468024680246802468024680

600                    CSP
602                     CP
604                     CP
606                    CCP
608                   CSP
610                   CSCP
612                    CCP
614                   CSCP
616                   C#CP
618                    CCP
620                     CP
622                     CP
624                    CSP
626                    CSP      P             P
628                    CSP     PCP           PC
630                    CSP      P            PC
632                    CSCP                  PC
634                     CP                  PCP
636                     CP                 PPCP
638                    CSCP            PP PCCCC
640                     CP          PPPCCPPPC##
642                    CSP       PPPS#SSCCC#SS#
644   CCCCCCCCCCCCCCCCCCCPPPPPPPPCCCCPPPPCCSCC#
646   ###################CSSSSSSSSSSSSSSSSS SSS
648                     #S
650                     #S
652                     #S
654                     #S
656                     #S
658                      #S
660                      #S
662                     #S
664                      #S
666                      #S
668                     #S
670                      #S
672                     #S
674                     #S
676                      #S
678                      #S
680                     #S
```

```
SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>INSIDE

 IMAGE FOR ALL CLASSES FOR FIELD  1 WITH ONLY INSIDE PIXELS PRINTED

       11111222223333344444555556666677777888889
       02468024680246802468024680246802468024680

 600   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 602   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
 604   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
 606   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 608   CCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPPP
 610   CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
 612   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 614   CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
 616   CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
 618   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 620   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
 622   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
 624   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 626   CCCCCCCCCCCCCCCCC   PPPPPP PPPPPPPPPPPPP
 628   CCCCCCCCCCCCCCCCC   PPPPP   PPPPPPPPPPP
 630   CCCCCCCCCCCCCCCCC   PPPPPP PPPPPPPPPPPP
 632   CCCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPP
 634   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPP
 636   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPP
 638   CCCCCCCCCCCCCCCCC    PPPPPPPPPPPP  P
 640   CCCCCCCCCCCCCCCCCC  PPPPPPPPPP
 642   CCCCCCCCCCCCCCCCC   PPPPPPP
 644
 646                                        S
 648   ##################  SSSSSSSSSSSSSSSSSSSSS
 650   ##################  SSSSSSSSSSSSSSSSSSSSS
 652   ##################  SSSSSSSSSSSSSSSSSSSSS
 654   ##################  SSSSSSSSSSSSSSSSSSSSS
 656   ##################  SSSSSSSSSSSSSSSSSSSSS
 658   ###################  SSSSSSSSSSSSSSSSSSSS
 660   ###################  SSSSSSSSSSSSSSSSSSSS
 662   ##################  SSSSSSSSSSSSSSSSSSSSS
 664   ###################  SSSSSSSSSSSSSSSSSSSS
 666   ###################  SSSSSSSSSSSSSSSSSSSS
 668   ##################  SSSSSSSSSSSSSSSSSSSSS
 670   ###################  SSSSSSSSSSSSSSSSSSSS
 672   ##################  SSSSSSSSSSSSSSSSSSSSS
 674   ##################  SSSSSSSSSSSSSSSSSSSSS
 676   ###################  SSSSSSSSSSSSSSSSSSSS
 678   ###################  SSSSSSSSSSSSSSSSSSSS
 680   ##################  SSSSSSSSSSSSSSSSSSSSS


 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>IMQUIT
 THE OPTION IMAGES REQUIRED        2.5902 SECONDS OF CPU TIME.

 ---------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>FEATSL

                    FEATSL  OPTION
                    **************


CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>1

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  1
>NO

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>CORN

CORN    NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
>PAST

PAST    NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
>WHEAT

WHEAT   NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
>SOY

SOY     NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =    441
>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
  4 SIGNATURES HAVE BEEN RETRIEVED
```

The FEATSL option is executed to compute the transformations necessary to reduce the test field data from 12 channels to a smaller number of channels while still maintaining a high degree of data class separation.

The REPLCE suboption is used to determine the best four of the original twelve channels which will result in a minimum loss of class separation in the data. The separability-to-be-gained map is displayed. The data transformation matrix (BMATX2) is computed and saved on unit 2. The signatures of the training fields are transformed with BMATX2 and saved on unit 2 as SIG1, SIG2, SIG3, and SIG4.

```
CHOOSE FEATURE SELECTION OPTION FROM
  SUBS*   REPLCE  QUIT
  CANON
>REPLCE

$INFEAT     ICHAN,BMX,KDIM,TOL,A(I,J),IFLG
>  $INFEAT KDIM=4 $
NUMBER  OF  CHANNELS  SELECTED  WILL  BE    4
TYPE YES IF INPUTS ARE OK
>YES
TYPE YES TO INPUT INTERCLASS WEIGHTS
>NO

CHANNEL  SELECTED= 8          AVERAGE  DIVERGENCE=  148.899          RATIO= .3840
CHANNEL  SELECTED=12          AVERAGE  DIVERGENCE=  219.195          RATIO= .5210
CHANNEL  SELECTED= 1          AVERAGE  DIVERGENCE=  249.965          RATIO= .6261
CHANNEL  SELECTED= 9          AVERAGE  DIVERGENCE=  302.383          RATIO= .7186

MAX.=   420.698    AVER. DIVERG=   298.383    RATIO=  .7093


TYPE YES TO DISPLAY INTERCLASS DIVERGENCES
>YES

   CLASS          MAXIMUM         COMPUTED        RATIO

  1    2           162.4            81.2          .5003
  1    3           354.5           246.4          .6952
  1    4            52.0            38.6          .7417
  2    3          1182.8           901.9          .7625
  2    4           500.0           350.5          .7010
  3    4           272.2           171.7          .6307

TYPE YES TO DISPLAY SEPARABILITY TO BE GAINED MAP
>YES

$INPUT      MAXX, ILABLX, ILABLY, ICODE
>  $INPUT ILABLY=6HY-AXIS,ILABLX=6HX-AXIS $

MAXX = 1000     ICODE = 1

X-AXIS

Y-AXIS

TYPE YES IF INPUTS ARE OK
>YES
```



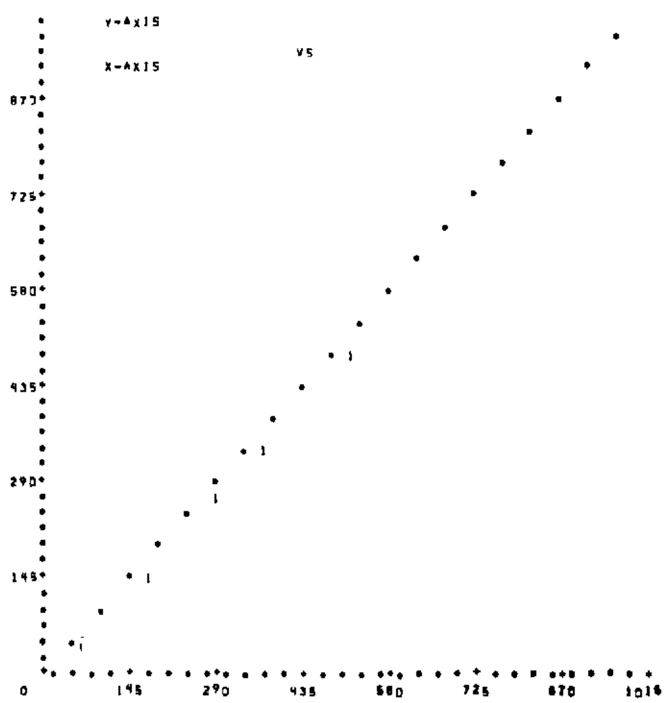

```
TYPE YES TO DISPLAY TRANSFORMED COVARIANCES AND MEAN VECTORS
>YES
TYPE YES TO SAVE TRANSFORMED SIGNATURES ON SIGNATURE FILE
>YES
```

```
CLASS =     1
TRANSFORMED  COVARIANCE
     7.306      -3.249       5.579       6.733
    -3.249      20.368      -2.802      -4.372
     5.579      -2.802       6.864       4.289
     6.733      -4.372       4.289       7.173
TRANSFORMED  MEAN  VECTOR
    75.946      78.481      84.776      61.277

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>SIG1
SIGNATURE SIG1   HAS BEEN SAVED ON UNIT 2
SIG1   ND =  4 K =   1  2  3  4
NUM(1) =      4

CLASS =     2
TRANSFORMED  COVARIANCE
     1.714      -2.313        .273        .821
    -2.313      61.949       -.464      -2.222
      .273       -.464       1.709        .377
      .821      -2.222        .377       1.917
TRANSFORMED  MEAN  VECTOR
    69.490     112.605      74.619      56.846

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>SIG2
SIGNATURE SIG2   HAS BEEN SAVED ON UNIT 2
SIG2   ND =  4 K =   1  2  3  4
NUM(1) =      4

CLASS =     3
TRANSFORMED  COVARIANCE
   109.745      63.251      15.463      77.326
    63.251      47.283       9.378      43.062
    15.463       9.378       5.815       9.783
    77.326      43.062       9.783      62.036
TRANSFORMED  MEAN  VECTOR
   111.968      75.002      80.760      96.023

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>SIG3
SIGNATURE SIG3   HAS BEEN SAVED ON UNIT 2
SIG3   ND =  4 K =   1  2  3  4
NUM(1) =      4

CLASS =     4
TRANSFORMED  COVARIANCE
     5.565       2.741       1.197       3.728
     2.741       9.117        .427       1.903
     1.197        .427       2.147        .952
     3.728       1.903        .952       4.292
TRANSFORMED  MEAN  VECTOR
    89.120      79.283      88.914      73.732

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>SIG4
SIGNATURE SIG4   HAS BEEN SAVED ON UNIT 2
SIG4   ND =  4 K =   1  2  3  4
NUM(1) =      4

TYPE YES TO DISPLAY FEATURE SPACE MATRIX
>YES

FEATURE  SPACE  TRANSFORMATION  MATRIX

       ROW =    1
     .000       .000       .000       .000       .000       .000
     .000      1.000       .000       .000       .000       .000

       ROW =    2
     .000       .000       .000       .000       .000       .000
     .000       .000       .000       .000       .000      1.000

       ROW =    3
    1.000       .000       .000       .000       .000       .000
     .000       .000       .000       .000       .000       .000

       ROW =    4
     .000       .000       .000       .000       .000       .000
     .000       .000      1.000       .000       .000       .000
TYPE YES TO SAVE B-MATRIX ON SIGNATURE FILE
>YES

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>BMATX2
SIGNATURE BMATX2 HAS BEEN SAVED ON UNIT 2
BMATX2 ND = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =      4

TYPE YES TO CONTINUE THE WITHOUT REPLACEMENT PROCEDURE
>NO
```

```
CHOOSE FEATURE SELECTION OPTION FROM
  SUBSP   REPLCE  QUIT
  CANON
>SUBSP

TYPE IN DIMENSION OF FEATURE SPACE

$INFEAT     ICHAN,BNX,KDIM,TOL,A(I,J),IFLG
>  $INFEAT KDIM=4$
SELECTED DIMENSION EQUALS        4
TYPE YES IF INPUTS ARE OK
>YES

INITIALIZE THE B-MATRIX BY SELECTING ONE OF THE FOLLOWING OPTIONS
CHANAL    VECTOR    DEFALT    RESTRT
>DEFALT

TYPE YES TO INPUT INTERCLASS WEIGHTS
>NO

MAX.=   420.648    AVER. DIVERG=   410.219    RATIO=  .9762

TYPE YES TO DISPLAY INTERCLASS DIVERGENCES
>YES
```

CLASS		MAXIMUM	COMPUTED	RATIO
1	2	162.4	151.5	.9327
1	3	354.5	340.9	.9618
1	4	52.0	48.1	.9245
2	3	1182.8	1172.2	.9911
2	4	500.0	493.8	.9875
3	4	272.2	254.8	.9360

```
TYPE YES TO DISPLAY SEPARABILITY TO BE GAINED MAP
>YES

$INPUT      MAXX, ILABLX, ILABLY, ICODE
>  $INPUT ILABLY=6HY-AXIS,ILABLX=6HX-AXIS $

MAXX = 1000     ICODE = 1

X-AXIS

Y-AXIS

TYPE YES IF INPUTS ARE OK
>YES
```




```
TYPE YES TO DISPLAY TRANSFORMED COVARIANCES AND MEAN VECTORS
>YES
TYPE YES TO SAVE TRANSFORMED SIGNATURES ON SIGNATURE FILE
>YES
```

The SUBSP suboption is used to determine the best four linear combinations of the original twelve channels which will result in a minimum loss of class separation in the data. The separability-to-be-gained map is displayed. The data transformation matrix (BMATX1) is computed and saved on unit 2. The signatures of the training fields are transformed with BMATX1 and saved on unit 2 as TSIG1, TSIG2, TSIG3, and TSIG4.

```
CLASS =     1
TRANSFORMED COVARIANCE
    21.813      7.096     -17.437     -11.843
     7.096      5.382       1.391       2.600
   -17.437      1.391      38.148      30.732
   -11.843      2.600      30.732      30.389
TRANSFORMED  MEAN  VECTOR
    80.759     59.546      73.408     131.085

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>TSIG1
SIGNATURE TSIG1  HAS BEEN SAVED ON UNIT 2
TSIG1  ND =  4 K =   1  2  3  4
NUM(1) =      4

CLASS =     2
TRANSFORMED  COVARIANCE
    70.813     17.867     -39.409     -29.603
    17.867      5.774      -9.009      -7.111
   -39.409     -9.009      25.552      18.104
   -29.603     -7.111      18.104      14.367
TRANSFORMED  MEAN  VECTOR
   121.350     67.554      44.875      99.532

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>TSIG2
SIGNATURE TSIG2  HAS BEEN SAVED ON UNIT 2
TSIG2  ND =  4 K =   1  2  3  4
NUM(1) =      4

CLASS =     3
TRANSFORMED  COVARIANCE
    89.954     97.141      99.896      57.027
    97.141    111.813     115.972      70.791
    99.896    115.972     141.880      76.669
    57.027     70.791      76.669      56.379
TRANSFORMED  MEAN  VECTOR
    87.750     88.063     135.112     150.082

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>TSIG3
SIGNATURE TSIG3  HAS BEEN SAVED ON UNIT 2
TSIG3  ND =  4 K =   1  2  3  4
NUM(1) =      4

CLASS =     4
TRANSFORMED  COVARIANCE
    11.909      6.050        .713      -1.018
     6.050      5.427       3.925       1.575
      .713      3.925      10.373       5.975
    -1.018      1.576       5.975       5.132
TRANSFORMED  MEAN  VECTOR
    85.206     68.762      96.012     145.036

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>TSIG4
SIGNATURE TSIG4  HAS BEEN SAVED ON UNIT 2
TSIG4  ND =  4 K =   1  2  3  4
NUM(1) =      4

TYPE YES TO DISPLAY FEATURE SPACE MATRIX
>YES

FEATURE  SPACE  TRANSFORMATION  MATRIX

        ROW =    1
    -.139       .095       -.341        .274        .174       -.196
    -.422       .142        .145        .167        .437        .522

        ROW =    2
    -.358      -.198       -.147       -.198        .407        .100
     .216       .585        .291       -.272        .141        .171

        ROW =    3
     .007       .004       -.002       -.026        .036        .202
     .074       .480        .559        .460       -.247       -.368

        ROW =    4
     .479       .276        .053        .285        .208        .667
     .083       .153        .041       -.094       -.166       -.259
TYPE YES TO SAVE B-MATRIX ON SIGNATURE FILE
>YES

CHOOSE FILE NUMBER FROM  1  2
>2

INPUT NAME TO SAVE DATA UNDER
>BMATX1
SIGNATURE BMATX1 HAS BEEN SAVED ON UNIT 2
BMATX1 ND = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =      4
```

```
CHOOSE FEATURE SELECTION OPTION FROM
  SUBSP   REPLCE  QUIT
  CANON
>QUIT
 THE OPTION FEATSL REQUIRED      51.6262 SECONDS OF CPU TIME.

 -----------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>TRNSFM
```

The TRNSFM option is used to transform the unpacked twelve channel data using the transformation matrix BMATX2, generated by the REPLCE suboption of FEATSL. The transformed data are written on unit 20.

```
                    TRNSFM  OPTION
                    --------------

 CHOOSE FROM SCALE OR TRANS.
>TRANS

 $INTRNS  DATUNT,NEWDAT,A,B
>  $INTRNS $
$INTRNS
DATUNT  =                   +4
NEWDAT  =                   +20
A       =        .00000000E+00
B       =        .10000000E+01

$END

 TYPE YES IF INPUTS ARE CORRECT.
>YES

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>2

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  2
>YES

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>BMATX2

BMATX2 NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(I) =      4

                MEAN         1 BY 12
                 ALL ZEROES.

                COVMAT     12 BY 12
          1          2          3          4          5          6

 1      .000       .000      1.000       .000       .000       .000
 2      .000       .000       .000       .000       .000       .000
 3      .000       .000       .000       .000       .000       .000
 4      .000       .000       .000       .000       .000       .000
 5      .000       .000       .000       .000       .000       .000
 6      .000       .000       .000       .000       .000       .000
 7      .000       .000       .000       .000       .000       .000
 8     1.000       .000       .000       .000       .000       .000
 9      .000       .000       .000      1.000       .000       .000
10      .000       .000       .000       .000       .000       .000
11      .000       .000       .000       .000       .000       .000
12      .000      1.000       .000       .000       .000       .000
          7          8          9         10         11         12

 1      .000       .000       .000       .000       .000       .000
 2      .000       .000       .000       .000       .000       .000
 3      .000       .000       .000       .000       .000       .000
 4      .000       .000       .000       .000       .000       .000
 5      .000       .000       .000       .000       .000       .000
 6      .000       .000       .000       .000       .000       .000
 7      .000       .000       .000       .000       .000       .000
 8      .000       .000       .000       .000       .000       .000
 9      .000       .000       .000       .000       .000       .000
10      .000       .000       .000       .000       .000       .000
11      .000       .000       .000       .000       .000       .000
12      .000       .000       .000       .000       .000       .000

>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
   1 SIGNATURES HAVE BEEN RETRIEVED
WRITE 2788 WORDS ON UNIT 20
WRITE 2788 WORDS ON UNIT 20
WRITE 1198 WORDS ON UNIT 20
 THE OPTION TRNSFM REQUIRED        5.2148 SECONDS OF CPU TIME.

 -----------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>UNITS
```

The UNITS option is executed to cycle the image unit number. The image generated by MAXLIK for the twelve channel data currently resides on unit 3.

```
                    UNITS  OPTION
                    -------------


$AVINUM
N       =              +3,             +12,             +9

$END
```

```
$INUNIT
IMGUNT  =           +3
DATUNT  =           +20
OBSUNT  =           +7
ISIGF1  =           +1
ISIGF2  =           +2
IHISF1  =           +10
IHISF2  =           +11
IMG1    =           +0
IMG2    =           +0
OBS1    =           +8

$END
CHOOSE OPTION FROM
  CYCLE   CHANGE  QUIT
>CYCLE
$INUNIT
IMGUNT  =           +12
DATUNT  =           +20
OBSUNT  =           +7
ISIGF1  =           +1
ISIGF2  =           +2
IHISF1  =           +10
IHISF2  =           +11
IMG1    =           +0
IMG2    =           +0
OBS1    =           +8

$END
 THE OPTION UNITS  REQUIRED       .0486 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>MAXLIK

                     MAXLIK  OPTION
                     --------------

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT     INPSIG

>REDSIG

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>2

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  2
>YES

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>SIG1

SIG1   ND =  4 K =   1  2  3  4
NUM(1) =     4

                 MEAN        1 BY  4
         1          2          3          4

 1     75.946    78.481     84.778     61.277


                 COVMAT      4 BY  4
         1          2          3          4

 1      9.306    -3.249      5.579      6.733
 2     -3.249    20.368     -2.807     -4.372
 3      5.579    -2.802      6.864      4.289
 4      6.733    -4.372      4.289      7.173

>SIG2

SIG2   ND =  4 K =   1  2  3  4
NUM(1) =     4

                 MEAN        1 BY  4
         1          2          3          4

 1     69.490   112.605     74.619     55.846


                 COVMAT      4 BY  4
         1          2          3          4

 1      1.719    -2.313       .273       .821
 2     -2.313    51.949      -.464     -2.222
 3       .273     -.464      1.709       .377
 4       .821    -2.222       .377      1.917

>SIG3

SIG3   ND =  4 K =   1  2  3  4
NUM(1) =     4

                 MEAN        1 BY  4
         1          2          3          4

 1    111.968    75.002     80.740     96.023


                 COVMAT      4 BY  4
         1          2          3          4

 1    109.795    63.251     15.463     77.326
 2     63.261    47.280      9.378     43.062
 3     15.463     9.378      6.818      9.783
 4     77.324    43.062      9.783     42.036
```

The CYCLE suboption cycles the unit number on which the next image generated will be saved to the next available image unit number. In this case, unit 12.

The MAXLIK option is called to classify the data transformed by BMATX2. The transformed signatures SIG1, SIG2, SIG3, and SIG4 are retrieved by the REDSIG suboption. The resulting image data is written on IMGUNT (i.e., unit 12).

```
>SIG4

SIG4   NO =  4 K =   1  2  3  4
NUM(1) =      4

              MEAN        1 BY  4
        1            2           3           4

 1    89.120     79.283     88.914     73.732


              COVMAT      4 BY  4
        1            2           3           4

 1     5.565      2.741      1.197      3.728
 2     2.741      9.117       .427      1.703
 3     1.197       .427      2.147       .952
 4     3.728      1.703       .952      4.292
>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
  4 SIGNATURES HAVE BEEN RETRIEVED

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>PROCSS
              IC DIS      4 BY  4
        1            2           3           4

 1     7.168     75.089    389.626     37.319
 2    92.484      5.317   1349.777    336.359
 3    96.243    385.863     10.317     74.970
 4    48.285    367.501    262.645      4.979

CLASS  SYMBOL  SIZE
  1       A     465
  2       B     456
  3       C     354
  4       D     402

CHOOSE MAXLIK OPTION FROM

  REDSIG  PROCSS  QUIT    INPSIG

>QUIT
 THE OPTION MAXLIK REQUIRED       5.1798 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>IMAGES

                    IMAGES  OPTION
                    **************
```

The IMAGES option is used to display the results of the maximum likelihood classification of the best four channels of data. Again the INSIDE and BORDER suboptions of the IMAGES option are executed.

```
 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=2000. $
$INTHRE
THRVAL  =      .20000000E+04
MINPIX  =                 +1
IFIELD  =                 +0,          +0,          +0,          +0,
                          +0,          +0,          +0,          +0,
                          +0,          +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>SYMBOL
 TYPE THE STRING OF  4 IMAGE SYMBOLS DESIRED.
>CP#S
 CLASS SYMBOL   ABCD
 IMAGE SYMBOL   CP#S
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>STATUS
IMUNIT   THRVAL   MINPIX   IFIELD
  12     2000.0      1       0  0  0  0  0  0  0  0  0  0
      CLASS    CLASS    IMAGE    NUMBER
      NUMBER   SYMBOL   SYMBOL   OF PIXELS
        1        A        C        465
        2        B        P        456
        3        C        #        354
        4        D        S        402
```

```
SELECT IMAGES OPTION FROM

    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>ALLCLS

 IMAGE FOR FIELD  1

        11111222223333344444555556666677777888889
        02468024680246802468024680246802468024680

   600  CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   602  CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   604  CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPP
   606  CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   608  CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   610  CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPP
   612  CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   614  CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPP
   616  CCCCCCCCCCCCCCCCCC#CPPPPPPPPPPPPPPPPPPPPP
   618  CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   620  CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   622  CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   624  CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPP
   626  CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   628  CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPC
   630  CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPC
   632  CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPCCC
   634  CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPCP
   636  CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPCPCPP
   638  CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPCCPPCCCC
   640  CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPCCCCPCCC#C
   642  CCCCCCCCCCCCCCCCCCSCPPPPPPPPPP##SSCCCC#SS
   644  CCCCCCCCCCCCCCCCCCCPPPPPPPCCCCCCPPPCCSCC#
   646  ###################CSSSSSSSSSSSSSSSSSSSSS
   648  ###################SSSSSSSSSSSSSSSSSSSSCS
   650  ###################SSSSSSSSSSSSSSSSSSSSSS
   652  ###################SSSSSSSSSSSSSSSSSSSSSS
   654  ###################SSSSSSSSSSSSSSSSSSSSSS
   656  ###################SSSSSSSSSSSSSSSSSSSSSS
   658  ###################SSSSSSSSSSSSSSSSSSSSSS
   660  ###################SSSSSSSSSSSSSSSSSSSSSS
   662  ###################SSSSSSSSSSSSSSSSSSSSSS
   664  ###################SSSSSSSSSSSSSSSSSSSSSS
   666  ###################SSSSSSSSSSSSSSSSSSSSSS
   668  ###################SSSSSSSSSSSSSSSSSSSSSS
   670  ###################SSSSSSSSSSSSSSSSSSSSSS
   672  ###################SSSSSSSSSSSSSSSSSSSSSS
   674  ###################SSSSSSSSSSSSSSSSSSSSSS
   676  ###################SSSSSSSSSSSSSSSSSSSSSS
   678  ###################SSSSSSSSSSSSSSSSSSSSSS
   680  ###################SSSSSSSSSSSSSSSSSSSSSS

SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>INSIDE

  IMAGE FOR ALL CLASSES FOR FIELD  1 WITH ONLY INSIDE PIXELS PRINTED

        11111222223333344444555556666677777888889
        02468024680246802468024680246802468024680

   600  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
   602  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
   604  CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
   606  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
   608  CCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPPP
   610  CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
   612  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
   614  CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
   616  CCCCCCCCCCCCCCCC     PPPPPPPPPPPPPPPPPPPP
   618  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
   620  CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
   622  CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
   624  CCCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPP
   626  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPP
   628  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPP
   630  CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPP
   632  CCCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPP
   634  CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPP P
   636  CCCCCCCCCCCCCCCCCC PPPPPPPPPPPPPP
   638  CCCCCCCCCCCCCCCCC    PPPPPPPPPP       C
   640  CCCCCCCCCCCCCCCCCC  PPPPPPPPPP       C
   642  CCCCCCCCCCCCCCCCC    PPPPP          C
   644
   646                                       S
   648  ##################  SSSSSSSSSSSSSSSSSS
   650  ##################  SSSSSSSSSSSSSSSSSSS S
   652  ##################  SSSSSSSSSSSSSSSSSSSSS
   654  ##################  SSSSSSSSSSSSSSSSSSSSS
   656  ##################  SSSSSSSSSSSSSSSSSSSSS
   658  ###################  SSSSSSSSSSSSSSSSSSSS
   660  ###################  SSSSSSSSSSSSSSSSSSSS
   662  ###################  SSSSSSSSSSSSSSSSSSSS
   664  ###################  SSSSSSSSSSSSSSSSSSSS
   666  ###################  SSSSSSSSSSSSSSSSSSSS
   668  ##################  SSSSSSSSSSSSSSSSSSSSS
   670  ###################  SSSSSSSSSSSSSSSSSSSS
   672  ##################  SSSSSSSSSSSSSSSSSSSSS
   674  ###################  SSSSSSSSSSSSSSSSSSSS
   676  ###################  SSSSSSSSSSSSSSSSSSSS
   678  ###################  SSSSSSSSSSSSSSSSSSSS
   680  ##################  SSSSSSSSSSSSSSSSSSSSS
```

```
 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     INQUIT
>BORDER

  IMAGE FOR ALL CLASSES FOR FIELD  1 WITH ONLY BORDER PIXELS PRINTED

        11111222223333344444555556666677777888889
        02468024680246802468024680246802468024680

600                      CSP
602                      CCP
604                     CSCP
606                      CCP
608                     CSP
610                     CSCP
612                      CCP
614                     CSCP
616                     C##CP
618                      CCP
620                       CP
622                       CP
624                      CSCP
626                      CSP                    P
628                      CSP                   PC
630                      CSP                  PPC
632                      CSCP                PCCC
634                       CP                P PCP
636                       CP             ##PCPCPP
638                      CSCP          PPCCPPC CC
640                       CP          PCCCCPC C#C
642                      CSCP     PPPP##SSCC C#SS
644     CCCCCCCCCCCCCCCCCCCCPPPP##PCCCCCCPPCCSCC#
646     #################CSSSSSSSSSSSSSSSSSSS SSS
648                       #S                  SCS
650                       #S                   S
652                       #S
654                       #S
656                       #S
658                        #S
660                        #S
662                        #S
664                        #S
666                        #S
668                       #S
670                        #S
672                       #S
674                        #S
676                        #S
678                        #S
680                       #S

 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     INQUIT
>INQUIT
 THE OPTION IMAGES REQUIRED       2.8326 SECONDS OF CPU TIME.

 --------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>UNITS
```

In order to compare the images generated by MAXLIK for the twelve and four channel data, it is required to use the DIFIMG option. However, it is first necessary to cycle the image unit numbers.

```
                    UNITS   OPTION
                    **************

$AVIMUN
N      =                +3,                 +12,                +9

$END
$INUNIT
IMGUNT  =               +12
DATUNT  =               +20
OBSUNT  =                +7
ISIGF1  =                +1
ISIGF2  =                +2
IHISF1  =               +10
IHISF2  =               +11
IMG1    =                +0
IMG2    =                +0
OBS1    =                +8

$END
CHOOSE OPTION FROM
  CYCLE   CHANGE   QUIT
>CYCLE
```

The CYCLE suboption is called to cycle the unit number on which the next image generated will be saved to the next available image unit number. In this case, unit 9.

```
$INUNIT
IMGUNT  =                +9
DATUNT  =               +20
OBSUNT  =                +7
ISIGF1  =                +1
ISIGF2  =                +2
IHISF1  =               +10
IHISF2  =               +11
IMG1    =                +3
IMG2    =               +12
OBS1    =                +8

$END
 THE OPTION UNITS  REQUIRED       .0796 SECONDS OF CPU TIME.

 --------------------------------------------------------------
```

```
ENTER ASTEP OPTION OR TYPE A BLANK
>DIFIMG

                    DIFIMG  OPTION
                    **************

INPUT EQUIVALENCE LISTS
>AA
>BB
>CC
>DD
>
CLASS  SYMBOL  SIZE
  1       A       0
  2       B      27
  3       C    1654
CLASS A ARE THOSE PIXELS OF IMAGE 1 THAT ARE NOT IN THE INPUT EQUIVALENCE LIST
CLASS B ARE THOSE PIXELS OF IMAGE 1 DIFFERENT FROM IMAGE 2
CLASS C ARE THOSE PIXELS THE SAME IN IMAGES 1 AND 2

 THE OPTION DIFIMG REQUIRED        .5270 SECONDS OF CPU TIME.

 -----------------------------------------------------------------
```

The DIFIMG option is called to generate an image of the difference between the images on units 3 and 12. A comparison of the previous images indicates that subset A in the first image is the same as subset A in the second image etc. This information is input to DIFIMG through the equivalence lists.

It is noted that 27 pixels are different between the images and 1654 pixels are the same.

```
ENTER ASTEP OPTION OR TYPE A BLANK
>IMAGES

                    IMAGES  OPTION
                    **************


 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=1 $

$INTHRE
THRVAL  =     .10000000E+01
MINPIX  =                +1
IFIELD  =                +0,          +0,          +0,          +0,
                         +0,          +0,          +0,          +0,
                         +0,          +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>SYMBOL
 TYPE THE STRING OF  3 IMAGE SYMBOLS DESIRED.
>ID
 CLASS SYMBOL    ABC
 IMAGE SYMBOL    ID
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>STATUS
IMUNIT   THRVAL   MINPIX   IFIELD
   9       1.0       1       0  0  0  0  0  0  0  0  0  0
      CLASS    CLASS    IMAGE     NUMBER
      NUMBER   SYMBOL   SYMBOL   OF PIXELS
        1        A        I          0
        2        B        D         27
        3        C                1654

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>ALLCLS
```

The IMAGES option is called to display the results of the DIFIMG option. The SYMBOL option is used to define the image symbols such that only pixels different between the images are displayed.

```
IMAGE FOR FIELD  1

         1111122222333334444455555666667777788888 9
         0246802468024680246802468024680246802468 0

    600
    602
    604                   D
    606
    608
    610
    612
    614
    616                   DD
    618
    620
    622
    624                     D
    626
    628                            D
    630
    632                                       DD
    634
    636                                     D DD
    638                                  DD
    640                                DD   DD   D
    642                     D          D      DD D
    644                            D    D
    646
    648                                        D
    650
    652
    654
    656
    658
    660
    662                     D
    664
    666
    668
    670
    672
    674                     D
    676
    678
    680


 SELECT IMAGES OPTION FROM
    STATUS     IMUNIT     THRESH     SYMBOL     ALLCLS
    ECHCLS     SUBSET     BORDER     INSIDE     IMQUIT
>IMQUIT
 THE OPTION IMAGES REQUIRED        .8808 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>TRNSFM
```

The TRNSFM option is called to transform the twelve channel test field data using the transformation matrix BMATX1, generated by the SUBSP suboption of FEATSL. The transformed data are written on unit 20.

```
                    TRNSFM  OPTION
                    **************


 CHOOSE FROM SCALE OR TRANS.
>TRANS

 $INTRNS  DATUNT,NEWDAT,A,B
>  $INTRNS DATUNT=4 $
$INTRNS
DATUNT  =                  4
NEWDAT  =                 20
A       =      .00000000E+00
B       =      .10000000E+01

$END
 TYPE YES IF INPUTS ARE CORRECT.
>YES

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>2

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  2
>YES

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>BMATX1

BMATX1 NO = 12 K =   1  2  3  4  5  6  7  8  9 10 11 12
NUM(1) =      4

                MEAN        1 BY 12
                 ALL ZEROES.


                COVMAT      12 BY 12
          1         2         3         4         5         6

1      -.139     -.358      .007      .479      .000      .000
2       .095     -.198      .004      .276      .000      .000
3      -.341     -.147     -.002      .053      .000      .000
4       .274     -.198     -.026      .285      .000      .000
```

```
5        .176      .407      .036      .208      .000      .000
6       -.176      .100      .202      .467      .000      .000
7       -.422      .216      .074      .083      .000      .000
8        .142      .685      .480      .153      .000      .000
9        .145      .291      .559      .041      .000      .000
10       .167     -.272      .460     -.094      .000      .000
11       .437      .141     -.297     -.146      .000      .000
12       .522      .171     -.348     -.269      .000      .000
           7         8         9        10        11        12

1        .000      .000      .000      .000      .000      .000
2        .000      .000      .000      .000      .000      .000
3        .000      .000      .000      .000      .000      .000
4        .000      .000      .000      .000      .000      .000
5        .000      .000      .000      .000      .000      .000
6        .000      .000      .000      .000      .000      .000
7        .000      .000      .000      .000      .000      .000
8        .000      .000      .000      .000      .000      .000
9        .000      .000      .000      .000      .000      .000
10       .000      .000      .000      .000      .000      .000
11       .000      .000      .000      .000      .000      .000
12       .000      .000      .000      .000      .000      .000

>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
  1 SIGNATURES HAVE BEEN RETRIEVED
WRITE 2788 WORDS ON UNIT 20
WRITE 2788 WORDS ON UNIT 20
WRITE 1198 WORDS ON UNIT 20
 THE OPTION TRNSFM REQUIRED      3.8964 SECONDS OF CPU TIME.

 --------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>UNITS
```

The UNITS option is called to change the image unit numbers. At this point the image of the MAXLIK classification of the twelve channel data is on unit 3, the image of the four channel classification is on unit 12, and the difference between the images is on unit 9.

The DIFIMG option assumes that the two images to be differenced are on units IMG1 and IMG2. The output of DIFIMG will be on IMGUNT.

```
                       UNITS   OPTION
                       **************

$AVIMUN
N        =              +3,              +12,              +9

$END
$INUNIT
IMGUNT   =              +9
DATUNT   =              +20
OBSUNT   =              +7
ISIGF1   =              +1
ISIGF2   =              +2
IHISF1   =              +10
IHISF2   =              +11
IMG1     =              +3
IMG2     =              +12
OBS1     =              +8

$END
CHOOSE OPTION FROM
  CYCLE   CHANGE   QUIT
>CHANGE
```

The CHANGE suboption is called to reset the next available image unit number to 12.

```
$INUNIT    IMGUNT,DATUNT,OBSUNT,ISIGF1,ISIGF2,IHISF1,IHISF2,IMG1,IMG2,OBS1
>  $INUNIT IMGUNT=12 $
$INUNIT
IMGUNT   =              +12
DATUNT   =              +20
OBSUNT   =              +7
ISIGF1   =              +1
ISIGF2   =              +2
IHISF1   =              +10
IHISF2   =              +11
IMG1     =              +3
IMG2     =              +12
OBS1     =              +8

$END
TYPE YES IF INPUTS OK
>YES
 THE OPTION UNITS  REQUIRED       .7803 SECONDS OF CPU TIME.

 --------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>MAXLIK
```

The MAXLIK option is called to classify the data transformed by BMATX1. The transformed signatures TSIG1, TSIG2, TSIG3, and TSIG4 are retrieved by the REDSIG suboption. The resulting image is written on IMGUNT (i.e., unit 12).

```
                       MAXLIK  OPTION
                       **************


CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>REDSIG

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>2

INPUT YES TO PRINT SIGNATURES RETRIEVED FROM FILE  2
>YES

LIST NAMES FOR SIGNATURES.  END LIST WITH NOMORE.
>TSIG1
```

```
TSIG1  ND =  4 K =   1  2  3  4
NUM(1) =     4

                 MEAN        1 BY  4
          1          2          3          4

 1     80.759     59.646     73.409    131.085

                 COVMAT      4 BY  4
          1          2          3          4

 1     21.813      7.096    -17.437    -11.843
 2      7.096      5.382      1.391      2.600
 3    -17.437      1.391     38.168     30.732
 4    -11.843      2.600     30.732     30.389

>TSIG2

TSIG2  ND =  4 K =   1  2  3  4
NUM(1) =     4

                 MEAN        1 BY  4
          1          2          3          4

 1    121.350     67.554     44.875     99.532

                 COVMAT      4 BY  4
          1          2          3          4

 1     70.813     17.867    -39.404    -29.603
 2     17.867      5.774     -9.009     -7.111
 3    -39.409     -9.009     25.552     18.104
 4    -29.603     -7.111     18.104     14.367

>TSIG3

TSIG3  ND =  4 K =   1  2  3  4
NUM(1) =     4

                 MEAN        1 BY  4
          1          2          3          4

 1     87.750     88.043    135.112    150.087

                 COVMAT      4 BY  4
          1          2          3          4

 1     89.954     97.191     99.896     57.027
 2     97.191    111.813    115.972     70.791
 3     99.896    115.972    141.880     76.669
 4     57.027     70.791     76.669     56.379

>TSIG4

TSIG4  ND =  4 K =   1  2  3  4
NUM(1) =     4

                 MEAN        1 BY  4
          1          2          3          4

 1     85.206     68.762     96.012    145.036

                 COVMAT      4 BY  4
          1          2          3          4

 1     11.909      6.050       .713     -1.018
 2      6.050      5.427      3.925      1.575
 3       .713      3.925     10.373      5.975
 4     -1.018      1.575      5.975      5.132

>NOMORE

CHOOSE FILE NUMBER FROM 1 OR  2 OR CHOOSE 0 TO QUIT.
>0
  4 SIGNATURES HAVE BEEN RETRIEVED

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>PROCSS
                 IC DIS      4 BY  4
          1          2          3          4

 1      7.939    148.453    515.040     95.504
 2    153.199      5.800   1542.098    514.101
 3    153.066    680.366     11.534    120.126
 4     56.898    474.842    373.870      5.348

CLASS  SYMBOL  SIZE
  1       A     463
  2       B     465
  3       C     350
  4       D     413

CHOOSE MAXLIK OPTION FROM
  REDSIG  PROCSS  QUIT    INPSIG

>QUIT
 THE OPTION MAXLIK REQUIRED      4.9290 SECONDS OF CPU TIME.

 ----------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
```

```
>IMAGES
                     IMAGES  OPTION
                     **************

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=2000, $
$INTHRE
THRVAL  =      .20000000E+04
MINPIX  =                 +1
IFIELD  =                 +0,            +0,            +0,            +0,
                          +0,            +0,            +0,            +0,
                          +0,            +0

$END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>SYMBOL
 TYPE THE STRING OF  4 IMAGE SYMBOLS DESIRED.
>CP#S
 CLASS SYMBOL   ABCD
 IMAGE SYMBOL   CP#S
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>STATUS
IMUNIT   THRVAL   MINPIX   IFIELD
  12     2000.0      1        0  0  0  0  0  0  0  0  0  0
      CLASS    CLASS    IMAGE     NUMBER
      NUMBER   SYMBOL   SYMBOL   OF PIXELS
        1        A        C        453
        2        B        P        466
        3        C        #        360
        4        D        S        413

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>ALLCLS

  IMAGE FOR FIELD  1

         11111222223333344444555556666677777888889
         02468024680246802468024680246802468024680

   600   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   602   CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
   604   CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
   606   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   608   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   610   CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
   612   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   614   CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
   616   CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
   618   CCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPPP
   620   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   622   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPPP
   624   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   626   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPP
   628   CCCCCCCCCCCCCCCCCCSPPPPPPPCPPPPPPPPPPPPPC
   630   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPPPPPPPPPPPC
   632   CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPPPPPPPC
   634   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPPCP
   636   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPPPPPPCCP
   638   CCCCCCCCCCCCCCCCCCSCPPPPPPPPPPPPPPCPPCCCC
   640   CCCCCCCCCCCCCCCCCCCPPPPPPPPPPPPPPCCPPCC#C
   642   CCCCCCCCCCCCCCCCCCSPPPPPPPPPPPS#SSCCCCSS#
   644   CCCCCCCCCCCCCCCCCCSPPPPPPPPCCCCCPPPCCSCCS
   646   ###################CSSSSSSSSSSSSSSSSSSSSS
   648   ###################SSSSSSSSSSSSSSSSSSSSSS
   650   ###################SSSSSSSSSSSSSSSSSSSSSS
   652   ###################SSSSSSSSSSSSSSSSSSSSSS
   654   ###################SSSSSSSSSSSSSSSSSSSSSS
   656   ###################SSSSSSSSSSSSSSSSSSSSSS
   658   ###################SSSSSSSSSSSSSSSSSSSSSS
   660   ####################SSSSSSSSSSSSSSSSSSSSS
   662   ###################SSSSSSSSSSSSSSSSSSSSSS
   664   ####################SSSSSSSSSSSSSSSSSSSSS
   666   ####################SSSSSSSSSSSSSSSSSSSSS
   668   ###################SSSSSSSSSSSSSSSSSSSSSS
   670   ###################SSSSSSSSSSSSSSSSSSSSSS
   672   ###################SSSSSSSSSSSSSSSSSSSSSS
   674   ###################SSSSSSSSSSSSSSSSSSSSSS
   676   ###################SSSSSSSSSSSSSSSSSSSSSS
   678   ####################SSSSSSSSSSSSSSSSSSSSS
   680   ###################SSSSSSSSSSSSSSSSSSSSSS


 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>INSIDE
```

The IMAGES option is then called to display the results of the maximum likelihood classification of the best four linear combinations of the original twelve channels. Again the INSIDE and BORDER suboptions of the IMAGE option are executed.

```
IMAGE FOR ALL CLASSES FOR FIELD  1 WITH ONLY INSIDE PIXELS PRINTED

       1111122222333334444455555666667777788888 9
       0246802468024680246802468024680246802468 0

 600   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 602   CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
 604   CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
 606   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 608   CCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPPP
 610   CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
 612   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 614   CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
 616   CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
 618   CCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPPPP
 620   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 622   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPPPPP
 624   CCCCCCCCCCCCCCCCC   PPPPPPPPPPPPPPPPPPPPP
 626   CCCCCCCCCCCCCCCCC   PPPPPP PPPPPPPPPPPPP
 628   CCCCCCCCCCCCCCCCC   PPPPP   PPPPPPPPPPP
 630   CCCCCCCCCCCCCCCCC   PPPPPP PPPPPPPPPPP
 632   CCCCCCCCCCCCCCCCC    PPPPPPPPPPPPPPPPPPP
 634   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPPPPP
 636   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPPPPPP PP
 638   CCCCCCCCCCCCCCCCC    PPPPPPPPPPPPP     C
 640   CCCCCCCCCCCCCCCCCC  PPPPPPPPPPP
 642   CCCCCCCCCCCCCCCCC   PPPPPPP
 644
 646                                      S   S
 648   ##################  SSSSSSSSSSSSSSSSSSSSS
 650   ##################  SSSSSSSSSSSSSSSSSSSSS
 652   ##################  SSSSSSSSSSSSSSSSSSSSS
 654   ##################  SSSSSSSSSSSSSSSSSSSSS
 656   ##################  SSSSSSSSSSSSSSSSSSSSS
 658   ###################  SSSSSSSSSSSSSSSSSSSS
 660   ###################  SSSSSSSSSSSSSSSSSSSS
 662   ##################  SSSSSSSSSSSSSSSSSSSSS
 664   ###################  SSSSSSSSSSSSSSSSSSSS
 666   ###################  SSSSSSSSSSSSSSSSSSSS
 668   ##################  SSSSSSSSSSSSSSSSSSSSS
 670   ##################  SSSSSSSSSSSSSSSSSSSSS
 672   ##################  SSSSSSSSSSSSSSSSSSSSS
 674   ##################  SSSSSSSSSSSSSSSSSSSSS
 676   ##################  SSSSSSSSSSSSSSSSSSSSS
 678   ###################  SSSSSSSSSSSSSSSSSSSS
 680   ##################  SSSSSSSSSSSSSSSSSSSSS

SELECT IMAGES OPTION FROM
   STATUS     INUNIT     THRESH     SYMBOL     ALLCLS
   ECHCLS     SUBSET     BORDER     INSIDE     INQUIT
>BORDER

IMAGE FOR ALL CLASSES FOR FIELD  1 WITH ONLY BORDER PIXELS PRINTED

       1111122222333334444455555666667777788888 9
       0246802468024680246802468024680246802468 0

 600                   CSP
 602                  CSCP
 604                  CSCP
 606                   CCP
 608                  CSP
 610                  CSCP
 612                   CCP
 614                  CSCP
 616                  CSCP
 618                  CSCP
 620                   CCP
 622                    CP
 624                   CSP
 626                   CSP      P             P
 628                   CSP     PCP           PC
 630                   CSP      P            PC
 632                   CSCP                  PC
 634                    CP                  PCP
 636                    CP              P  PCCP
 638                   CSCP            PCPPC CC
 640                    CP          PPPCCPPCCNC
 642                   CSP       PPPSNSSCCCCSSN
 644   CCCCCCCCCCCCCCCCCCSPPPPPPPPCCCCPPPPCCSCCS
 646   ##################CSSSSSSSSSSSSSSSSSS SS
 648                    #S
 650                    #S
 652                    #S
 654                    #S
 656                    #S
 658                     #S
 660                     #S
 662                    #S
 664                     #S
 666                     #S
 668                    #S
 670                    #S
 672                    #S
 674                    #S
 676                    #S
 678                     #S
 680                    #S

SELECT IMAGES OPTION FROM
   STATUS     INUNIT     THRESH     SYMBOL     ALLCLS
   ECHCLS     SUBSET     BORDER     INSIDE     INQUIT
>ECHCLS
```

The ECHCLS suboption is executed to display each class separately.

```
IMAGE FOR FIELD  1
FOR CLASS C ONLY

       11111222223333344444555556666677777888889
       02468024680246802468024680246802468024680

  600  CCCCCCCCCCCCCCCCCC
  602  CCCCCCCCCCCCCCCCC C
  604  CCCCCCCCCCCCCCCCC C
  606  CCCCCCCCCCCCCCCCCCC
  608  CCCCCCCCCCCCCCCCC
  610  CCCCCCCCCCCCCCCCC C
  612  CCCCCCCCCCCCCCCCCCC
  614  CCCCCCCCCCCCCCCCC C
  616  CCCCCCCCCCCCCCCCC C
  618  CCCCCCCCCCCCCCCCC C
  620  CCCCCCCCCCCCCCCCCCC
  622  CCCCCCCCCCCCCCCCCCC
  624  CCCCCCCCCCCCCCCCCC
  626  CCCCCCCCCCCCCCCCCC
  628  CCCCCCCCCCCCCCCCCC        C             C
  630  CCCCCCCCCCCCCCCCCC                      C
  632  CCCCCCCCCCCCCCCCCC C                    C
  634  CCCCCCCCCCCCCCCCCCC                    C
  636  CCCCCCCCCCCCCCCCCCC                   CC
  638  CCCCCCCCCCCCCCCCCC C              C  CCCC
  640  CCCCCCCCCCCCCCCCCCC              CC  CC C
  642  CCCCCCCCCCCCCCCCCC                CCCC
  644  CCCCCCCCCCCCCCCCCC         CCCCC   CC CC
  646                     C
  648
  650
  652
  654
  656
  658
  660
  662
  664
  666
  668
  670
  672
  674
  676
  678
  680
```

```
IMAGE FOR FIELD  1
FOR CLASS P ONLY

       11111222223333344444555556666677777888889
       02468024680246802468024680246802468024680

  600                     PPPPPPPPPPPPPPPPPPPPPP
  602                     PPPPPPPPPPPPPPPPPPPPPP
  604                     PPPPPPPPPPPPPPPPPPPPPP
  606                     PPPPPPPPPPPPPPPPPPPPPP
  608                    PPPPPPPPPPPPPPPPPPPPPPP
  610                     PPPPPPPPPPPPPPPPPPPPPP
  612                     PPPPPPPPPPPPPPPPPPPPPP
  614                     PPPPPPPPPPPPPPPPPPPPPP
  616                     PPPPPPPPPPPPPPPPPPPPPP
  618                     PPPPPPPPPPPPPPPPPPPPPP
  620                     PPPPPPPPPPPPPPPPPPPPPP
  622                     PPPPPPPPPPPPPPPPPPPPPP
  624                     PPPPPPPPPPPPPPPPPPPPPP
  626                     PPPPPPPPPPPPPPPPPPPPPP
  628                     PPPPPPP PPPPPPPPPPPPP
  630                     PPPPPPPPPPPPPPPPPPPPP
  632                      PPPPPPPPPPPPPPPPPPPP
  634                     PPPPPPPPPPPPPPPPPPPP P
  636                     PPPPPPPPPPPPPPPPPPP  P
  638                      PPPPPPPPPPPPPP PP
  640                     PPPPPPPPPPPPPP  PP
  642                     PPPPPPPPPPP
  644                     PPPPPPPP     PPP
  646
  648
  650
  652
  654
  656
  658
  660
  662
  664
  666
  668
  670
  672
  674
  676
  678
  680
```

```
IMAGE FOR FIELD  1
FOR CLASS # ONLY

       11111222223333344444555556666677777888889
       02468024680246802468024680246802468024680

   600
   602
   604
   606
   608
   610
   612
   614
   616
   618
   620
   622
   624
   626
   628
   630
   632
   634
   636
   638
   640                                         #
   642                                 #        #
   644
   646 ###################
   648 ###################
   650 ###################
   652 ###################
   654 ###################
   656 ###################
   658 ####################
   660 ####################
   662 ###################
   664 ####################
   666 ####################
   668 ###################
   670 ###################
   672 ###################
   674 ###################
   676 ###################
   678 ####################
   680 ###################


IMAGE FOR FIELD  1
FOR CLASS S ONLY

       11111222223333344444555556666677777888889
       02468024680246802468024680246802468024680

   600                   S
   602                  S
   604                  S
   606
   608                  S
   610                  S
   612
   614                  S
   616                  S
   618                  S
   620
   622
   624                   S
   626                   S
   628                   S
   630                   S
   632                   S
   634
   636
   638                   S
   640
   642                   S           S SS     SS
   644                  S                   S  S
   646                    SSSSSSSSSSSSSSSSSSSSSS
   648                   SSSSSSSSSSSSSSSSSSSSSSS
   650                   SSSSSSSSSSSSSSSSSSSSSSS
   652                   SSSSSSSSSSSSSSSSSSSSSSS
   654                   SSSSSSSSSSSSSSSSSSSSSSS
   656                   SSSSSSSSSSSSSSSSSSSSSSS
   658                    SSSSSSSSSSSSSSSSSSSSSS
   660                    SSSSSSSSSSSSSSSSSSSSSS
   662                   SSSSSSSSSSSSSSSSSSSSSSS
   664                    SSSSSSSSSSSSSSSSSSSSSS
   666                    SSSSSSSSSSSSSSSSSSSSSS
   668                   SSSSSSSSSSSSSSSSSSSSSSS
   670                   SSSSSSSSSSSSSSSSSSSSSSS
   672                   SSSSSSSSSSSSSSSSSSSSSSS
   674                   SSSSSSSSSSSSSSSSSSSSSSS
   676                   SSSSSSSSSSSSSSSSSSSSSSS
   678                    SSSSSSSSSSSSSSSSSSSSSS
   680                   SSSSSSSSSSSSSSSSSSSSSSS

SELECT IMAGES OPTION FROM
    STATUS    INUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>SUBSET
INPUT CHARACTER FOR PRINT AND IMAGE SYMBOLS FOR CLASSES IN SUBSET.
>XWP
THE IMAGE SYMBOLS WP                        WILL BE PRINTED AS X
TYPE YES IF INPUTS ARE CORRECT.
>YES
```

The SUBSET suboption is executed to display the W and P subsets as X. All other subsets appear as blanks.

```
                     SUBSET #P
                  PRINTED AS X

  IMAGE FOR FIELD  1

      11111222223333344444555556666677777888889
      02468024680246802468024680246802468024680

600                      XXXXXXXXXXXXXXXXXXXXXX
602                      XXXXXXXXXXXXXXXXXXXXXX
604                      XXXXXXXXXXXXXXXXXXXXXX
606                      XXXXXXXXXXXXXXXXXXXXXX
608                      XXXXXXXXXXXXXXXXXXXXXX
610                      XXXXXXXXXXXXXXXXXXXXXX
612                      XXXXXXXXXXXXXXXXXXXXXX
614                      XXXXXXXXXXXXXXXXXXXXXX
616                      XXXXXXXXXXXXXXXXXXXXXX
618                      XXXXXXXXXXXXXXXXXXXXXX
620                      XXXXXXXXXXXXXXXXXXXXXX
622                      XXXXXXXXXXXXXXXXXXXXXX
624                      XXXXXXXXXXXXXXXXXXXXXX
626                      XXXXXXXXXXXXXXXXXXXXXX
628                      XXXXXXX XXXXXXXXXXXXX
630                      XXXXXXXXXXXXXXXXXXXXX
632                       XXXXXXXXXXXXXXXXXXXX
634                      XXXXXXXXXXXXXXXXXXXX X
636                      XXXXXXXXXXXXXXXXXXX  X
638                       XXXXXXXXXXXXXX XX
640                      XXXXXXXXXXXXXX  XX  X
642                      XXXXXXXXXXX X        X
644                      XXXXXXXX     XXX
646   XXXXXXXXXXXXXXXXXXX
648   XXXXXXXXXXXXXXXXXXX
650   XXXXXXXXXXXXXXXXXXX
652   XXXXXXXXXXXXXXXXXXX
654   XXXXXXXXXXXXXXXXXXX
656   XXXXXXXXXXXXXXXXXXX
658   XXXXXXXXXXXXXXXXXXXX
660   XXXXXXXXXXXXXXXXXXXX
662   XXXXXXXXXXXXXXXXXXX
664   XXXXXXXXXXXXXXXXXXXX
666   XXXXXXXXXXXXXXXXXXXX
668   XXXXXXXXXXXXXXXXXXX
670   XXXXXXXXXXXXXXXXXXX
672   XXXXXXXXXXXXXXXXXXX
674   XXXXXXXXXXXXXXXXXXX
676   XXXXXXXXXXXXXXXXXXX
678   XXXXXXXXXXXXXXXXXXXX
680   XXXXXXXXXXXXXXXXXXX

SELECT IMAGES OPTION FROM
    STATUS    INUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    INQUIT
>SUBSET
INPUT CHARACTER FOR PRINT AND IMAGE SYMBOLS FOR CLASSES IN SUBSET.
>OCS
THE IMAGE SYMBOLS CS                              WILL BE PRINTED AS O
TYPE YES IF INPUTS ARE CORRECT.
>YES
```

Next the SUBSET suboption is executed to display the C and S subsets as O. All other subsets appear as blanks.

```
                     SUBSET CS
                  PRINTED AS O

  IMAGE FOR FIELD  1

      11111222223333344444555556666677777888889
      02468024680246802468024680246802468024680

600   OOOOOOOOOOOOOOOOOOO
602   OOOOOOOOOOOOOOOOOOO
604   OOOOOOOOOOOOOOOOOOO
606   OOOOOOOOOOOOOOOOOOO
608   OOOOOOOOOOOOOOOOOO
610   OOOOOOOOOOOOOOOOOOO
612   OOOOOOOOOOOOOOOOOOO
614   OOOOOOOOOOOOOOOOOOO
616   OOOOOOOOOOOOOOOOOOO
618   OOOOOOOOOOOOOOOOOOO
620   OOOOOOOOOOOOOOOOOOO
622   OOOOOOOOOOOOOOOOOOO
624   OOOOOOOOOOOOOOOOOOO
626   OOOOOOOOOOOOOOOOOOO
628   OOOOOOOOOOOOOOOOOOO       O             O
630   OOOOOOOOOOOOOOOOOOO                     O
632   OOOOOOOOOOOOOOOOOOOO                    O
634   OOOOOOOOOOOOOOOOOOO                    O
636   OOOOOOOOOOOOOOOOOOO                   OO
638   OOOOOOOOOOOOOOOOOOOO              O  OOOO
640   OOOOOOOOOOOOOOOOOOO              OO  OO O
642   OOOOOOOOOOOOOOOOOOO           O OOOOOOOO
644   OOOOOOOOOOOOOOOOOOO        OOOOO   OOOOOO
646                      OOOOOOOOOOOOOOOOOOOOOO
648                      OOOOOOOOOOOOOOOOOOOOOO
650                      OOOOOOOOOOOOOOOOOOOOOO
652                      OOOOOOOOOOOOOOOOOOOOOO
654                      OOOOOOOOOOOOOOOOOOOOOO
656                      OOOOOOOOOOOOOOOOOOOOOO
658                       OOOOOOOOOOOOOOOOOOOOO
660                       OOOOOOOOOOOOOOOOOOOOO
662                      OOOOOOOOOOOOOOOOOOOOOO
664                       OOOOOOOOOOOOOOOOOOOOO
666                       OOOOOOOOOOOOOOOOOOOOO
668                      OOOOOOOOOOOOOOOOOOOOOO
670                      OOOOOOOOOOOOOOOOOOOOOO
672                      OOOOOOOOOOOOOOOOOOOOOO
674                      OOOOOOOOOOOOOOOOOOOOOO
676                      OOOOOOOOOOOOOOOOOOOOOO
678                       OOOOOOOOOOOOOOOOOOOOO
680                      OOOOOOOOOOOOOOOOOOOOOO
```

```
SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>IMQUIT
 THE OPTION IMAGES REQUIRED       7.3304 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>UNITS
```

The UNITS option is called to change the unit numbers.

```
                    UNITS   OPTION
                    **************


$AVIMUN
N         =             +3,             +12,             +9

$END
$INUNIT
IMGUNT    =             +12
DATUNT    =             +20
OBSUNT    =             +7
ISIGF1    =             +1
ISIGF2    =             +2
IHISF1    =             +10
IHISF2    =             +11
IMG1      =             +3
IMG2      =             +12
OBS1      =             +8

$END
CHOOSE OPTION FROM
  CYCLE    CHANGE   QUIT
>CHANGE
```

The CHANGE suboption is called to reset the next available image unit number to 9.

```
$INUNIT    IMGUNT,DATUNT,OBSUNT,ISIGF1,ISIGF2,IHISF1,IHISF2,IMG1,IMG2,OBS1
>  $INUNIT IMGUNT=9 $
$INUNIT
IMGUNT    =             +9
DATUNT    =             +20
OBSUNT    =             +7
ISIGF1    =             +1
ISIGF2    =             +2
IHISF1    =             +10
IHISF2    =             +11
IMG1      =             +3
IMG2      =             +12
OBS1      =             +8

$END
TYPE YES IF INPUTS OK
>YES
 THE OPTION UNITS  REQUIRED        .0940 SECONDS OF CPU TIME.
 ------------------------------------------------------------
ENTER ASTEP OPTION OR TYPE A BLANK
>DIFIMG
```

The DIFIMG option is called to generate an image of the difference between the images on units 3 and 12. The generated image will be placed on unit 9. A comparison of the images to be differenced indicates the appropriate equivalence lists to be input to DIFIMG.

```
                    DIFIMG  OPTION
                    **************



INPUT EQUIVALENCE LISTS
>AA
>BB
>CC
>DD
>
CLASS  SYMBOL  SIZE
  1       A       0
  2       B      14
  3       C    1667
```

It is noted that 14 pixels are different between the images and 1667 pixels are the same.

```
CLASS A ARE THOSE PIXELS OF IMAGE 1 THAT ARE NOT IN THE INPUT EQUIVALENCE LIST
CLASS B ARE THOSE PIXELS OF IMAGE 1 DIFFERENT FROM IMAGE 2
CLASS C ARE THOSE PIXELS THE SAME IN IMAGES 1 AND 2

 THE OPTION DIFIMG REQUIRED       .5448 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTEP OPTION OR TYPE A BLANK
>IMAGES
```

The IMAGES option is called to display the results of the DIFIMG option. The SYMBOL suboption is used to define image symbols such that only pixels different between the images are displayed.

```
                    IMAGES  OPTION
                    **************


 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>THRESH
$INTHRE  THRVAL,MINPIX,IFIELD
>  $INTHRE THRVAL=1.0 $
$INTHRE
THRVAL  =      .10000000E+01
MINPIX  =             +1
IFIELD  =             +0,            +0,            +0,            +0,
                      +0,            +0,            +0,            +0,
                      +0,            +0
```

```
*END
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>SYMBOL
 TYPE THE STRING OF  3 IMAGE SYMBOLS DESIRED.
>ID
 CLASS SYMBOL   ABC
 IMAGE SYMBOL   ID
TYPE YES IF INPUTS ARE CORRECT.
>YES

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>STATUS
IMUNIT   THRVAL   MINPIX   IFIELD
   9        1.0      1       0  0  0  0  0  0  0  0  0  0
      CLASS     CLASS     IMAGE      NUMBER
      NUMBER    SYMBOL    SYMBOL   OF PIXELS
        1         A         I          0
        2         B         D         14
        3         C                 1667

 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>ALLCLS

  IMAGE FOR FIELD  1

         11111222223333344444555556666677777888889
         02468024680246802468024680246802468024680

    600
    602                   D
    604                   D
    606
    608
    610
    612
    614
    616                   D
    618                   D
    620
    622
    624
    626
    628
    630
    632
    634
    636                                       D
    638                                   D
    640                                      D  D
    642                                      D
    644                    D            D        D
    646
    648
    650
    652
    654
    656
    658
    660
    662
    664
    666
    668
    670                     D
    672
    674
    676                     D
    678
    680


 SELECT IMAGES OPTION FROM
    STATUS    IMUNIT    THRESH    SYMBOL    ALLCLS
    ECHCLS    SUBSET    BORDER    INSIDE    IMQUIT
>IMQUIT
 THE OPTION IMAGES REQUIRED         .9098 SECONDS OF CPU TIME.

 ------------------------------------------------------------

ENTER ASTER OPTION OR TYPE A BLANK
>QUIT

                  QUIT    OPTION
                  **************


 ------------------------------------------------------------

 THE OPTIONS IN THIS RUN REQUIRED    135.5182 SECONDS OF CPU TIME.

 *-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-*-
```

The QUIT option returns control to the operating system. The input @FIN terminates the input stream begun by the @RUN input.

At the termination of an execution, the total CPU time required by all options in the run is printed.

7. REFERENCES

1. Philco-Ford Corp., WDL Div., Houston Operations", Earth Resources Data Format Control Book", prepared for Data Systems Integration Section, Communications Data Systems Branch, Flight Support Div., Flight Operations Directorate, NASA/JSC Houston, Texas, PHO-TR543, October 1973.

2. MacQueen, J., "Some Methods for Classification and Analysis of Multivariate Observations", in Proc. of 5th Berkely Symposium on Mathematical Statistics and Probability, University of California Press 1967, pp 281-297.

3. Nagy, G. and Tolaba, J., "Nonsupervised Crop Classification Through Airborne Multispectral Observations", IBM Journal of Research and Development, March 1972.

4. J. A. Quirein, "An Iterative Approach to the Feature Selection Classification Problem", TRW Technical Note 99900-H019-RO, December 1972.

5. J. A. Quirein, "Divergence, Some Necessary Conditions for an Extremum", University of Houston, Department of Mathematics Report, November 1972.

6. C. Y. Chang, "Dynamic Programming as Applied to Feature Subset Selection in a Pattern Recognition System", LEC/HASD No. 640-TR-046, November 1971.

7. J. A. Quirein, "Feature Selection", TRW IOC 6531.6-73-32, May 1973.

8. I. Johnson, "Impulsive Orbit Transfer Optimization by an Accelerated Gradient Method", Journal of Spacecraft and Rockets, Vol. 6, No. 5, May 1969.

9. Ball, G. H. and Hall, D. J., "ISODATA, A Novel Method of Data Analysis and Pattern Classification", Standford Research Institute, Menlo Park, California, Tech. Rep. AD699616, April 1965.

10. Kan, Edwin P. F., and Holley W. A., "More on Clustering Techniques with Final Recommendations on ISODATA", LEC/HASD No. 640-TR-112, May 1972.

11. F. L. Barnes, "ISODATA Clustering Sensitivity to Split and Combine Parameters", TRW IOC 6534.6-72-43, October 1972.

12. Fu, K. S., Landgrebe, D. A., and Phillips, T. L., "Information Processing of Remotely Sensed Agriculture Data", Proc. of IEEE, Vol. 57, No. 4, April 1969.