NAS9 - 14970
T-1314/4
MA - 129TA

Final Technical Report

NASA Contract NAS9-14970
June 1, 1976 - May 31, 1977

D. A. Landgrebe, Purdue University
Principal Investigator

J. D. Erickson, NASA/JSC
Technical Monitor

Volume I of III

Submitted by
The Laboratory for Applications of Remote Sensing
Purdue University West Lafayette, Indiana
1977

NAS9-14970
T-1314/4
MA-129TA
III DRA
NASA-CR-151454
77-10182

Final Technical Report

(E77-10182) [APPLICATIONS OF REMOTE SENSING, VOLUME 1] Final Technical Report, 1 Jun. 1976 - 31 May 1977 (Purdue Univ.) 196 p HC A09/MF A01 CSCL 05B

N77-30548

Unclas
G3/43 00182

NASA Contract NAS9-14970

June 1, 1976 - May 31, 1977

D. A. Landgrebe, Purdue University
Principal Investigator

J. D. Erickson, NASA/JSC
Technical Monitor

[redacted]

Volume I of III

Original photography may be purchased from
EROS Data Center
[redacted]
Sioux Falls, SD 57198

Submitted by

The Laboratory for Applications of Remote Sensing
Purdue University West Lafayette, Indiana

1977

PURDUE
UNIVERSITY LABORATORY FOR APPLICATIONS
OF REMOTE SENSING

June 28, 1977

National Aeronautics and Space
Administration
Scientific and Technical
Information Facility, Code KS
Washington, D.C. 20546

Dear Sirs:

The report listed below is submitted in accord with Contract NAS9-14970, Article XVI - Reports of Work and Documentation.

"Final Technical Report", Volumes I, II, and III, NASA Contract NAS9-14970, June 1, 1976 - May 31, 1977 by the Laboratory for Applications of Remote Sensing.

We have also provided a space below for you to indicate that this document has been placed in the public domain in accord with Article XX - Limitations on Earth Resources Data.

Very truly yours,

D. A. Landgrebe

D. A. Landgrebe
Director, LARS

DAL:jh
Enclosures 1 copy of the 3 volumes

______ This is to certify the above document has been placed in the public domain.

Signed ______________________

Title ______________________

Table of Contents

RESEARCH TASKS

Volume I

Volume II

Volume III

Preface

This report provides a summary of results for the past year's effort under contract NAS9-14970. The contract called for work on a wide variety of separate and distinct but related tasks. Below is a list of the tasks as contained in the original Work Statement.

As a result of this contract a large volume of results has been generated. Technical and research reports previously submitted or presently being published are listed in Table I.

Because of the diversity present in the task list for the contract, each major subdivision of this report has been written to be relatively self-contained. We hope this will facilitate use of the report by readers with different interests.

Many of the tasks below are continuing; as a result, the discussion presented relative to them constitutes a progress report. Task 2.1 and 2.2 are not continuing, thus the discussion relative to them is more complete.

The various tasks have been managed by various Purdue staff members during the year. It is appropriate that the contributions of these people be recognized.

	Task	Purdue Task Manager
2.1	Test of Boundary Finding/Per Field Classification (ECHO)	P.H. Swain
2.2	Stratification of Scene Characteristics	P.H. Swain
	2.2a Stratification by Machine Clustering	
	2.2b Digitization and Registration of Ancillary Data	
	2.2c Crop Inventory Using Full-Frame Classification	
2.3	LACIE Field Measurements	M.E. Bauer
2.4	Scanner System Parameter Selection	P.A. Anuta
2.5	Technology Transfer	J.C. Lindenlaub
2.6	Large Area Crop Inventory Design	M.E. Bauer
2.7	Forestry Applications Project	R.P. Mroczynski
2.8	Regional Applications Project	R.A. Weismiller
2.9	Interpretation of Thermal Band Data	L.F. Silva
2.10	Super Site Data Management	L.F. Silva
2.11	Soil Classification and Survey	M.F. Baumgardner
2.12	Improved Analysis Techniques for Multitemporal Data	P.H. Swain

The efforts of Dr. J.D. Erickson, the contract Technical Monitor and Mr. M.C. Trichel are especially to be noted and greatfully acknowledged.

Table I. Technical and Research Reports

052977 Davis. The Focus Series: A Collection of Single-Concept Remote Sensing Educational Materials.

042777 Russell and Lindenlaub. Disseminating Technological Information on Remote Sensing to Potential Users.

120776 Vanderbilt, Silva and Bauer. A Laser Technique for Characterizing the Geometry of Plant Canopies.

091576 DeWitt and Robinson. Description and Evaluation of a Bidirectional Reflectance Factor Reflectometer.

070676 Kristof, Russell, Cary, Lube and Weismiller. Determining Land Use Patterns Through Man-Machine Analysis of LANDSAT Data - A Tutorial Simulation.

111076 Russell. Systematically Disseminating Technological Information to Potential Users.

110976 Landgrebe, Simmons and Biehl. An Empirical Study of Scanner System Parameters.

090776 Svedlow, McGillem, and Anuta. Analytical and Experimental Design and Analysis of an Optimal Processor for Image Registration.

082776 Montgomery, Baumgardner and Weismiller. An Investigation of the Relationship Between Spectral Reflectance and the Chemical, Physical, and Genetic Characteristics of Soils.

062176 Wiersma and Landgrebe. The Use of Spatial Characteristics for the Improvement of Multispectral Classification of Remotely Sensed Data.

052576 Lindenlaub and Lube. Matrix of Education of Training Materials in Remote Sensing.

051576 Bauer and Davis. Stratification of Landsat Data by Clustering.

*032576 Todd, Mausel, and Baumgardner. Urban Land Use Monitoring from Computer-Implemented Processing of Airborne Multispectral Data.

*031276 Mausel, Todd and Baumgardner. An Analysis of Metropolitan Land-Use by Machine Processing of Earth Resources Technology Satellite Data.

052075 DeWitt and Robinson. Description and Operation of a Field Rated ERTS-Band Transmissometer.

* Papers reporting earlier SRT work which has just appeared in print.

OTHER REPORTS

Biehl and Simmons.
Field Measurements Data Library Catalog, Volume I, 1974-75 Crop.
October 29, 1976.

Biehl and Simmons.
Field Measurements Data Library Catalog, Volume II, 1975-76 Crop.
December 1, 1976.

2.1 Test of Boundry Finding/Per field Classification

INTRODUCTION

One might wish to adopt a new classification technique for one of three reasons:

- It provides greater accuracy than the currently implemented technique..
- The new technique is less expensive than the current technique.
- The classification results produced by the new technique are easier to use and interpret than those currently produced.

The ECHO* classification algorithms developed at the Laboratory for Applications of Remote Sensing (LARS) have shown promise in all three areas listed above.[1]

Contemporary classifiers for analysis of remotely sensed data compare the spectral measurements from each feature of each point to class statistics, computing a likelihood of discriminant function associated with each class, and categorizing the point according to the class with the largest discriminant function value. Each point is classified individually on the basis of its spectral measurements alone. One premise of this technique is that the objects of interest are large in comparison to the size of a point. If this were not so, a large proportion of points would be composites of several classes, making statistical pattern classification unreliable since pre-specified categories would be inadequate to describe actual states of nature. From this premise it follows that objects are represented by arrays of points, and that a statistical dependence exists between consecutive points. Contemporary classifiers fail to exploit the statistical dependence between adjacent points when assigning classes.

*ECHO stands for Extraction and Classification of Homogeneous Objects

Giving no consideration to the spectral response of adjacent points contributes to a "salt and pepper" effect in classification results. While the results produced may be accurate, they may be too complex, and can even be confusing when attempting to produce a useful analysis product such as a forest inventory map.[2] Consequently time and effort have been invested in developing smoothing programs which "clean-up" the "salt and pepper" effects in classification results.[2]

The ECHO processors benefit from spatial information by first aggregating into groups points whose spectral responses are not significantly different in a statistical sense, and then applying a maximum likelihood classification rule to these homogeneous groups. Homogeneous objects are identified in a three step process. First, cells are formed by systematically partitioning the data into N by N sized blocks of pixels. The statistics of each cell are then compared to a homogeneity criterion. Points which do not comprise homogeneous groups are classified on a point-by-point basis, just as contemporary classifiers classify all points. Statistics of adjoining homogeneous cells are then compared to annexation thresholds. Adjoining cells which appear to belong to the same statistical population are combined into a single object.

Two separate ECHO algorithms have been developed. The first, Supervised ECHO, makes use of pre-specified class statistics to identify homogeneous objects. The second, Nonsupervised ECHO, identifies homogeneous objects without the use of class statistics. Consequently, those objects identified by the Nonsupervised algorithm may be used to aid in the training process.

Past Work at LARS

Much of the background research on the ECHO concept was performed at LARS during 1975 and is documented in the Final Report for 1975,[3] R. L. Ketting's doctoral thesis,[1] a LARS Information Note,[4] and in symposium proceedings.[5]

The partitioning of a data set into homogeneous objects can occur in one of two general approaches. One can utilize a disjunctive approach, starting with large areas and repeatedly subdividing them until each unit appears to be homogeneous. Alternatively, one can choose a conjunctive approach where adjacent units which appear to belong to a single object are combined. One advantage to the conjunctive approach is, if second order statistics are to be used, the second order statistics of the combined area can be calculated from the second order statistics of each constituent unit. The statistics for each subdivision of the initial partition made under the disjunctive approach must be recalculated starting with the individual pixel values, however. Hence the conjunctive approach is potentially faster than the disjunctive approach.

Prior to the 1975 work, Rodd's conjunctive partitioning algorithm[6] was combined with a minimum distance sample classifier and an improvement in classification accuracy over conventional classification was observed, however processing time was increased.[4] Gupta and Wintz,[7] added a second order statistics test to Rodd's first order test, but obtained essentially the same results at an increase in processing time. Robertson[8] implemented a disjunctive partitioning algorithm with a minimum distance classifier which acheived the same classification accuracy as the conventional point-by-point classifier with an order of magnitude increase in computer time.

During 1975 new statistical criteria were applied to the partitioning algorithm producing a "Supervised" field extraction routine to add to the Nonsupervised routine. In addition, the sample classifier utilized by ECHO was altered from a minimum distance strategy to a maximum likelihood strategy. Experimental results were obtained for two aircraft and two LANDSAT-1 data sets which indicated that ECHO algorithms improved classification accuracies for the four tested areas and that they reduced the CPU time required to perform the classifications.

FORTRAN programs were produced implementing the Supervised and Nonsupervised algorithms in two phases. Channel selection, cell size, and cell homogeneity criteria were specified in the first phase of the Supervised processor, annexation in the second. For the Nonsupervised processor, only channel selection and cell size were specified in phase one, cell homogeneity and annexation thresholds were specified in phase two. The Nonsupervised processor did not have the capability to "split" cells; i.e. the constituent points in a singular cell had to be classified as a small sample rather than point-by-point.

Documentation of these programs consisted of commented program listings and a thesis.[3]

Objectives of the 1976-1977 Task

The objectives of the FY77 work were to:

1) Deliver to NASA documented FORTRAN programs implementing the ECHO algorithms, and to make improvements to those algorithms, where appropriate.

2) Systematically test the algorithms on MSS data for agricultural regions as observed by the LANDSAT satellites, aircraft scanners, and on the simulated Thematic Mapper data.

3) Provide products enabling the determination of the utility of the object maps to a LACIE Analyst Interpreter in the selection and labeling of training fields.

DESCRIPTION OF WORK

A. Documented Fortran Programs

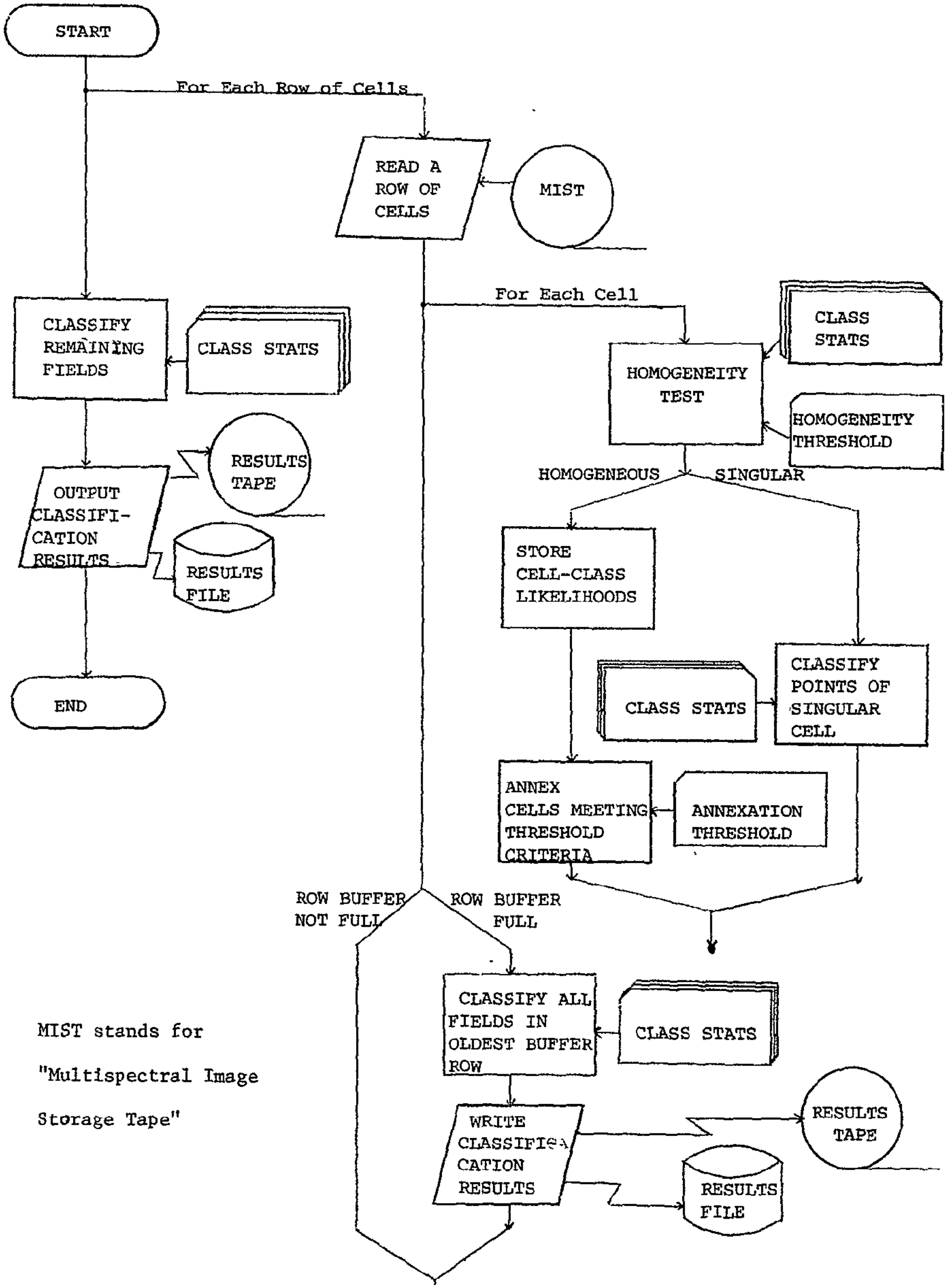
1. Supervised Processor

FORTRAN listings and program abstracts for the Supervised ECHO processor are presented in Appendix A. This processor requires class statistics for field extraction as well as for classification. The general flow of this algorithm is presented in Figure 1. The software is designed to function in either a single phase or a two phase mode. The two phase approach accomplishes channel selection, cell size, and cell homogeneity criteria in phase one; annexation tests take place during phase two. The intermediate results between phases are stored on magnetic tapes. The final results are written either on disk or on tape. Functioning in the two phase mode allows the analyst to vary the annexation threshold and, thus, to produce several results outputs without repeating the expensive cell homogeneity test.

The single step approach allows the analyst to specify all the necessary input parameters on a single control card deck and produce classification results without the necessity of an intermediate tape. When the analyst wishes to produce classifications with varying annexation thresholds for a particular area, it is less expensive in terms of computer time to utilize the two phase approach. When a single annexation threshold is desired, it is more efficient to use the single phase approach. The Supervised ECHO processor determines whether a single phase classification, the first phase of a two phase classification, or the second phase of a two phase classification is to be generated based on the control cards specified.

FIGURE 1

SUPERVISED ECHO FLOW



MIST stands for "Multispectral Image Storage Tape"

Input and output requirements for each processing mode of the Supervised processor are summarized in Table 2.1-1 and Table 2.1-2.

TABLE 2.1-1

Supervised ECHO

Single Phase Mode

Input	Output
Cell Width Parameter	Results File
Cell Homogeneity Threshold	Classification Map (optional)
Annexation Threshold	Singular Cell Map (optional)
Channel Selection	
Specification of the Area to Classify	
Multispectral Image Storage Tape (MIST)	
Specification of Results Destination	
Class Statistics	

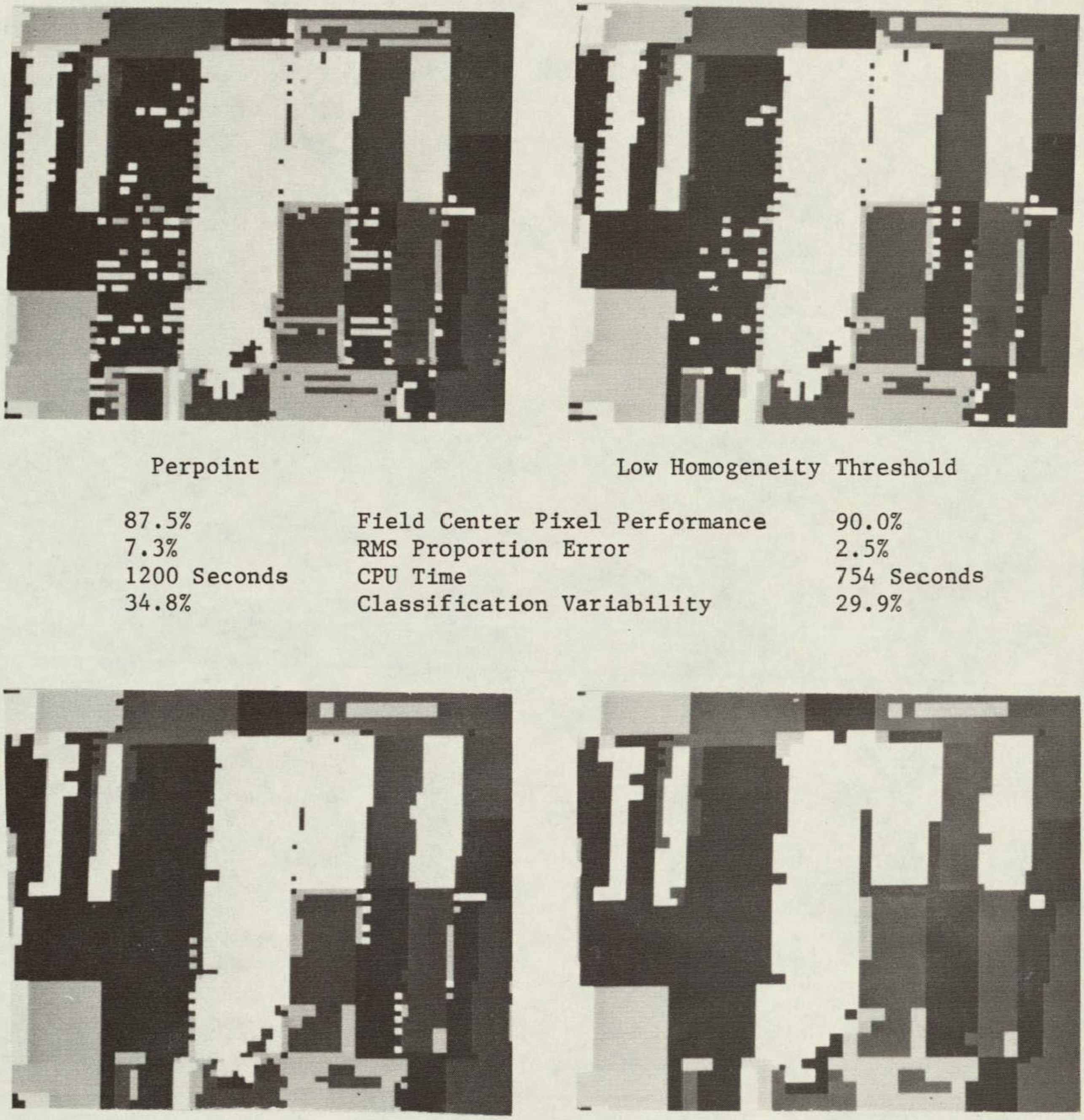
TABLE 2.1-2

Two Phase Mode

Supervised ECHO

Input	Output
Phase One	
Cell Width Parameter	Intermediate Results Tape
Cell Homogeneity Threshold	Singular Cell Map (optional)
Channel Selection	
Specification of the Area to Classify	
Multispectral Image Storage Tape	
Intermediate Tape and File Specifications	
Class Statistics	
Phase Two	
Intermediate Results Tape File	Results File Classification Map (optional)
Annexation Threshold	
Specification of Results Destination	

Figure 2
Effects of Cell Homogeneity Threshold



Perpoint

Low Homogeneity Threshold

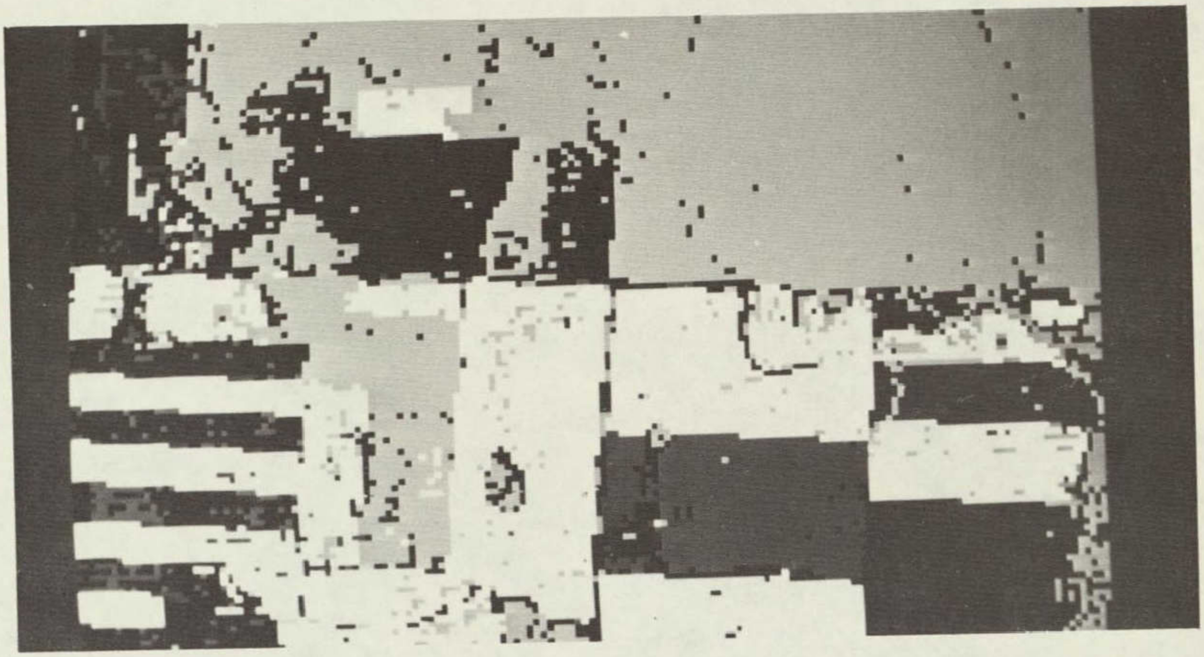
Perpoint		Low Homogeneity Threshold
87.5%	Field Center Pixel Performance	90.0%
7.3%	RMS Proportion Error	2.5%
1200 Seconds	CPU Time	754 Seconds
34.8%	Classification Variability	29.9%

Medium Homogeneity Threshold

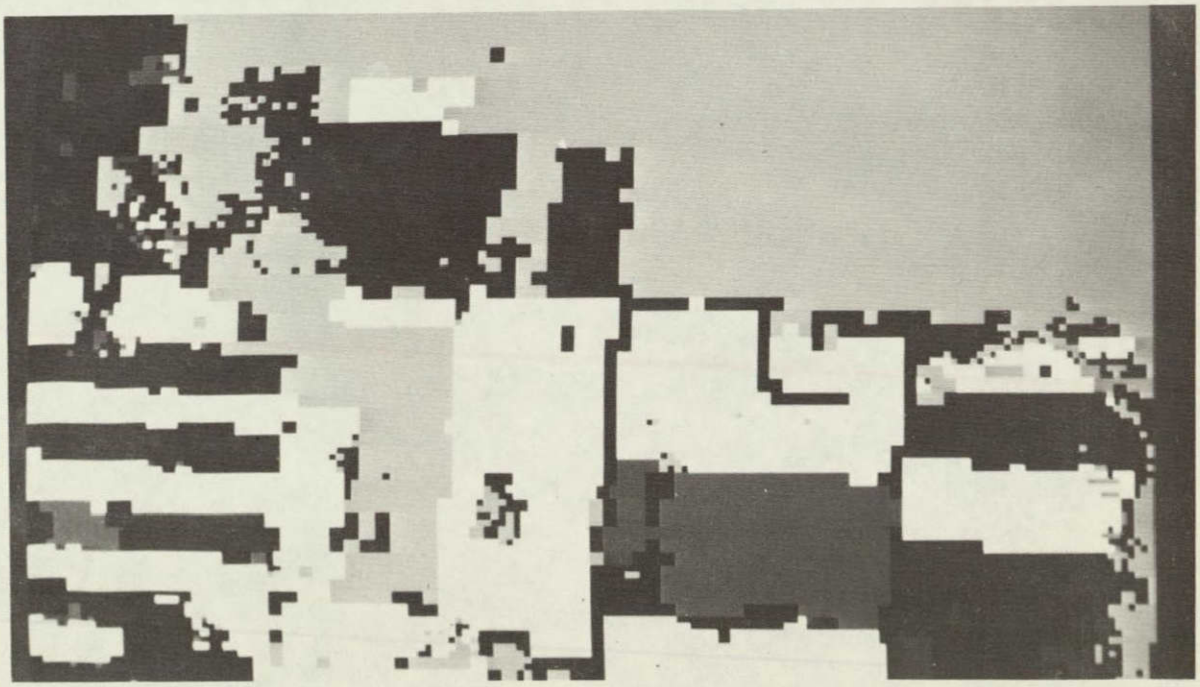
Large Homogeneity Threshold

Medium Homogeneity Threshold		Large Homogeneity Threshold
91.4%	Field Center Pixel Performance	90.7%
2.5%	RMS Proportion Error	2.8%
494 Seconds	CPU Time	370 Seconds
22.5%	Classification Variability	18.6%

Figure 3



Photographs of classification results of aircraft data collected over Williams County, Kansas. Notice the "salt and pepper" in the perpoint results (above). The ECHO results (below) have both higher classification performance and lower classification variability. The ECHO classification was produced using 40% of the CPU time required to produce the perpoint classification.



2. Nonsupervised Processor

Output products for this task included those allowing the determination of the utility of the resulting object map to the LACIE Analyst Interpreter for identifying and labeling training fields. Since the Supervised processor utilizes class statistics for the identification of homogeneous objects, it has only limited value as a training aid. The Nonsupervised processor, on the other hand, does not require class statistics for object extraction, and thus, may prove quite beneficial as a training aid, serving, in a sense, as a spatial clustering algorithm.

The Nonsupervised processor utilizes a homogeneity test which compares σ_{ij}/μ_{ij} to a threshold t_i; where i refers to feature or channel and j refers to the cell. If the standard deviation divided by the mean for feature i exceeds the user-specified threshold t_i, the cell is "singular" and elements of the cell will be classified on a point-by-point basis. Cells are annexed to fields on the basis of a two-step test, 1) that the channel variances of the field are equivalent to the channel variances of the cell and 2) that the channel means of the field are equivalent to the channel means of the cell. Should either of these criteria not be met for any channel, annexation will not take place.

The research software was designed to run in two phases. The first phase merely calculated the cell mean and covariance matrices and wrote them on tape; the second phase proceeded to perform field extraction followed by classification. This process has two disadvantages. First, although an object map could be produced, it had to be produced in phase two, the same phase that required a class statistics deck for input in order to perform the classification. Second, since only cell mean and covariance matrices were written on the intermediate tape, when singular cells were identified in phase two, they had to be classified as small samples, a cell at a time, rather than on a point-by-point basis, because data values for individual points were not available to the phase two classifier.

The software was restructured to move the field extraction algorithm into phase one of the processor and produce an intermediate tape which contains:

1. class means and covariance matrices for each homogeneous object identified.
2. an object map containing the mean for channel i of object j in every pixel of object j and the original data values for those pixels belonging to singular cells.
3. a tag array identifying whether a pixel falls in a singular cell (and should be classified individually), or in a homogeneous object (and hence should receive a class assignment based on the sample classification of the object to which it belongs).

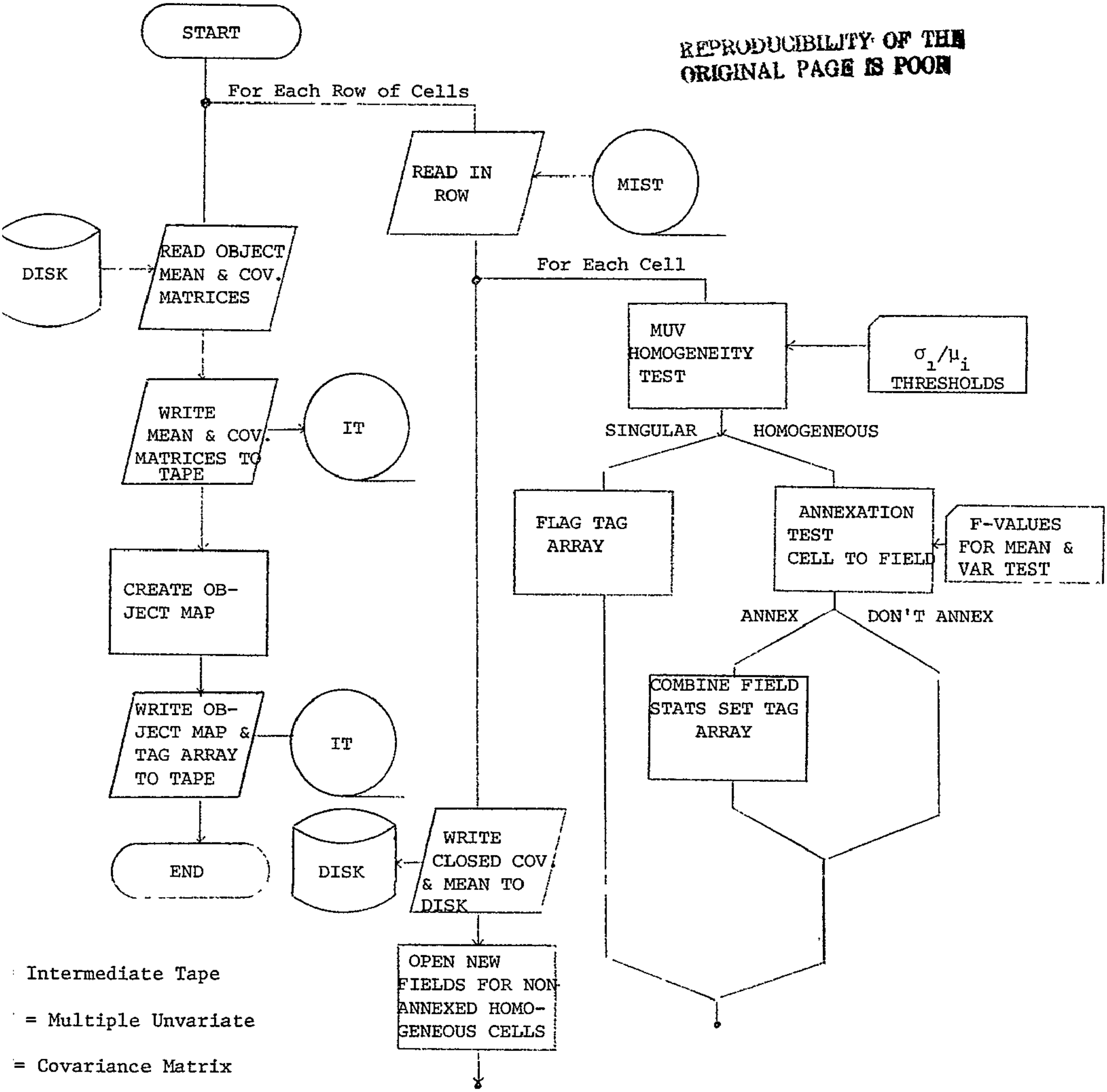
The second phase of the restructured Nonsupervised processor reads the intermediate tape and the class statistics deck, performs a maximum likelihood sample classification (utilizing second order statistics where possible) on the objects identified, and a maximum likelihood point-by-point classification on points falling in singular cells. Figure 4A presents the general processing flow for phase one (field extraction) of the Nonsupervised processor. The general processing flow for phase two (classification) is presented in Figure 4B. Inputs to outputs from the Nonsupervised ECHO processor's two phases are described in Table 2.1-3.

FIGURE 4A

GENERAL FLOW OF

NONSUPERVISED ECHO

PHASE 1


START
For Each Row of Cells
READ IN ROW
MIST
DISK
READ OBJECT MEAN & COV. MATRICES
For Each Cell
MUV HOMOGENEITY TEST
σ_i/μ_i THRESHOLDS
WRITE MEAN & COV. MATRICES TO TAPE
IT
SINGULAR
HOMOGENEOUS
FLAG TAG ARRAY
ANNEXATION TEST CELL TO FIELD
F-VALUES FOR MEAN & VAR TEST
CREATE OB-JECT MAP
ANNEX
DON'T ANNEX
COMBINE FIELD STATS SET TAG ARRAY
WRITE OB-JECT MAP & TAG ARRAY TO TAPE
IT
END
DISK
WRITE CLOSED COV & MEAN TO DISK
OPEN NEW FIELDS FOR NON-ANNEXED HOMO-GENEOUS CELLS

Intermediate Tape

= Multiple Unvariate

= Covariance Matrix

FIGURE 4B

GENERAL FLOW OF

NONSUPERVISED ECHO

PHASE 2

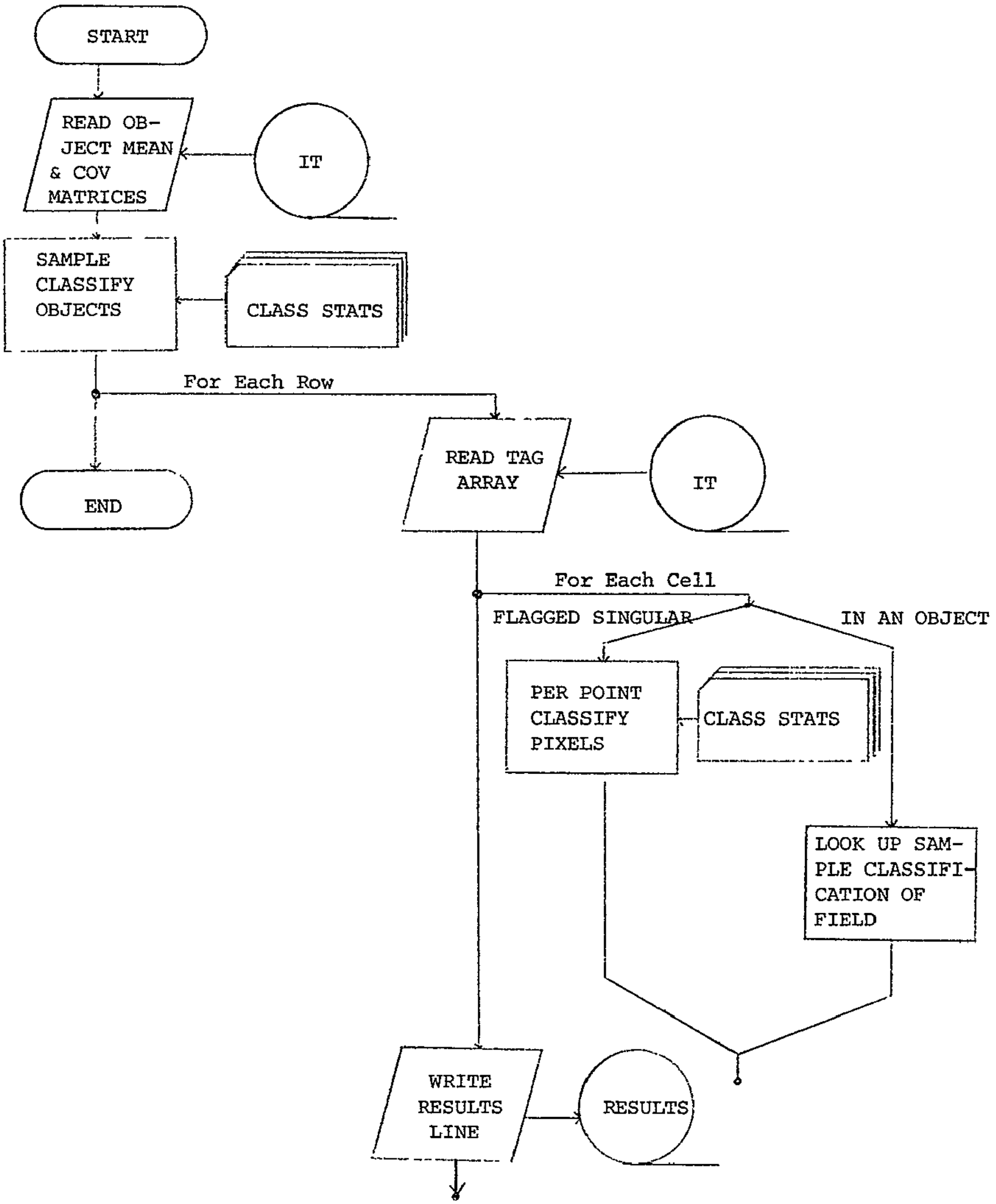


TABLE 2.1-3

NONSUPERVISED ECHO

PROCESSOR

INPUT	OUTPUT
Phase One:	
Channel Selection Cell Width Cell Homogeneity Thresholds (σ_i/μ_i for channel i) Annexation Thresholds (mean and variance) Multispectral Image Storage Tape Intermediate Results Tape and File Specification	Intermediate Tape containing object map, object statistics and pixel tag array.
Phase Two:	
Class Statistics Intermediate Tape from Phase One Specification of Results Tape and File	Results File

B. Tests of the ECHO algorithms

1. Data Sets

The second obejctive of Task 2.1 is to test the ECHO algorithms on MSS data for agricultural regions. Data sets are to include LANDSAT, aircraft, and simulated Thematic Mapper data. Ten LANDSAT, three aircraft and eight simulated Thematic Mapper (2 sites and 4 resolutions) data sets were selected for the analysis. One bitemporal data set was examined separately. These data sets are summarized in Table 2.1-4

TABLE 2.1-4

Data Sets

(LANDSAT)

CHANNEL NUMBER	WAVE BAND µM
1,5,9	.5-.6
2,6,10	.6-.7
3,7,11	.7-.8
4,8,12	.8-1.1

Area	Channels Used	Data Collected	Data Set
Graham County, Kansas	9, 10, 11, 12	5/26/74	LACIE/SRS
Grant County, Kansas	5, 6, 7, 8	5/19/74	LACIE/SRS
Haskell County, Kansas	9, 10, 11, 12	5/27/74	LACIE/SRS
Kearny County, Kansas	9, 10, 11, 12	5/27/74	LACIE/SRS
Huntington County, Indiana	1, 2, 3, 4,	7/7/73	CITARS
Shelby County, Indiana	1, 2, 3, 4	9/7/73	CITARS
White County, Indiana	1, 2, 3, 4	8/21/73	CITARS
Livingston County, Illinois	1, 2, 3, 4	7/16/73	CITARS
Fayette County, Illinois	1, 2, 3, 4	8/21/73	CITARS
Lee County, Illinois	1, 2, 3, 4	8/5/73	CITARS

TABLE 2.1-4 (Continued)

Simulated Thematic Mapper

TM CHANNEL NUMBER	WAVE BAND µM
1	.45-.52
2	.52-.60
3	.63-.69
4	.74-.80
5	.80-.91
6	1.55-1.75
7	10.4-12.5
8	.74-.91

Area	Channels Used	Date	Resolution
Williams County, ND	1, 2, 3, 6, 8	8/15/75	30m
Williams County, ND	1, 2, 3, 6, 8	8/15/75	40m
Williams County, ND	1, 2, 3, 6, 8	8/15/75	50m
Williams County, ND	1, 2, 3, 6, 8	8/15/75	60m
Finney County, Kansas	2, 3, 5, 6, 7	7/6/75	30m
Finney County, Kansas	2, 3, 5, 6, 7	7/6/75	40m
Finney County, Kansas	2, 3, 5, 6, 7	7/6/75	50m
Finney County, Kansas	2, 3, 5, 6, 7	7/6/75	60m

TABLE 2.1-4 (Continued)

Aircraft

Area	Wavebands Used (μM)	Data Collected
Williams County, ND	.4-.49, .59-.64, .65-.69,	8/15/75
Finney County, Kansas	.82-.88, 1.53-1.62, 10.1-11.0	7/6/75
Tippecanoe County, IN	.52-.57, .61-.70, .72-.92, 1.0-1.4, 1.5-1.8	8/13/71

Bitemporal

Area	Wavebands Used for Each Date	Dates	Data Set
Grant County, Kansas	.5-.6, .6-.7,	5/9/74	LACIE/SRS
	.7-.8, .8-1.1	6/14/74	

2. Training Procedures

a. LANDSAT Data Sets

The LACIE training sets were created using ground truth information provided by JSC for both test and training fields. Odd-numbered fields appearing in each subclass were used for training; even-numbered fields were used for test. For Graham and Grant counties, statistics were generated by using the STATISTICS processor. For Haskell and Kearny counties statistics were secured by clustering the training fields of each class into four subclasses which were then pooled on the basis of output from the SEPARABILITY processor. It should be noted that both training and test fields for the LACIE/SRS data are large enough to inset field boundaries approximately two pixels inside estimated field boundaries. This inset allows for any image misregistration which may occur between any two dates on the multitemporal input runs. Proportion estimates for the 1974 LACIE/SRS segments were provided in ground truth packets provided by JSC.

The CITARS training sets were originally created by a supervised procedure using ground truth information provided by the Agricultural Stabilization and Conservation Services (ASCS) to select both the training and test fields.[9] Those training sets were used without change in the ECHO tests.

Six counties in the CITARS experiment were used as test sites. A data set free of clouds which occured late in the growing season was required for each of the six counties. Dates from July, August or early September were selected. The classification results for all of the CITARS experiments are catalogued on a series of LARS tapes. The desired data sets were located on the catalogued CITARS tapes and the statistics which had been used for the CITARS experiment were obtained by using the LARSYS PUNCHSTATISTICS processor. The pooling of classes was determined by running the PRINTRESULTS processor and requesting training field results. This combination of requests produces a table of available classes in the statistics deck and also the informational names under which they were classified. By using this list it was possible to reconstruct the combinations of spectral classes and pooled spectral classes which were needed to reproduce the original classification.

Training and test fields for each CITARS classification were secured and appropriate control cards added to evaluate the ECHO classifications of CITARS data sets. The proportion estimates used for the CITARS evaluation are estimates of the proportions of the various classes for the entire county made by the SRS.

b. Simulated Thematic Mapper Data Sets

Training sets from the simulated Thematic Mapper tests performed at LARS in 1976 were used for ECHO analysis. These training sets were generated by selecting fields of known cover types ana clustering each informational class separately to define subclasses.

Color infrared photographic mosaic prints were made from photographic data collected concurrently with the scanner data. Informational class information provided by ground observations was transfered to clear plastic overlays on the mosaic print. The analyst could then easily locate the corresponding fields in his cluster maps and assign the field coordinates to the informational classes.

Statistics were calculated for each training area and compared using the SEPARABILITY processor. Similar classes were combined, where indicated, and the data set was used to classify the flightline. Training areas were not excluded from the test fields since the test fields had been preselected for the entire flightline.

Two to four subclasses were found in each informational class. The Kansas flight was an exception. Because of severe line-to-line changes in signal level in the original 6 meter scanner data, the analyst was forced to create more spectral classes to account for the within-class variations due to excessive noise. This was most apparent in the 30 and 40 meter resolution data. The effect was reduced but not eliminated in the 50 and 60 meter resolution data. Alternate fields were used for training and test decks, respectively. The procedure was repeated for each of the four resolution sizes. As resolution size increased, the number of spectral classes decreased.

The entire training set selection procedure was repeated for each resolution size so that any effects on training set selection which might be caused by data resolution would be included in the analysis results. An example is the increasing difficulty and eventual impossibility of selecting samples from small, or narrow, fields as the resolution size increases.

c. Aircraft

Six Meter NASA Aircraft Scanner Data

The two 6-meter aircraft scanner data sets used to generate the simulated Thematic Mapper data set were used. The same training and test fields used for the Thematic Mapper were available in the six meter data set. Because of the vary large number of data points in the full data set, only the first two miles (one third of the total flight line) were used for the ECHO evaluations.

These data sets were not corrected for sun and scanner angle effects. To compensate for these angle effects, training fields were distributed across the width of the flight line. At least six fields in each informational class were used in the training set. Fields from informational classes were clustered together into five spectral classes. All spectral classes from all informational classes were combined into a statistics deck and appropriate pooling was done based on SEPARABILITY results. The classification results were strongly influenced by angle effects.

Flightline 210, Corn Blight Watch Experiment

Training and test fields from the Michigan Scanner data set were located on the LARS display system using ground information collected during the Corn Blight Watch Experiment. At least six fields were selected to represent each informational class. Each informational class was then clustered to produce five spectral subclasses. All spectral classes were combined and processed with SEPARABILITY to select the subset of the original 12 channels which was to be used for classification. A subset of five channels was chosen and the training set was again processed with SEPARABILITY to determine which classes should be pooled.

d. Bitemporal Data Set

A bitemporal data set from Grant County, Kansas was created by selecting data from a spring date and a mid-summer date. Training and test fields were available from the LACIE data and were edited to reduce the number of single line training fields by about 30 percent. Data from the two dates were clustered together within each informational class with four spectral classes requested. Classes were grouped as indicated by the CLUSTER algorithm and classes with fewer than 50 points were deleted.

3. Dependent Variables

There are six variables which were monitored to evaluate the ECHO algorithm:

- CPU time
- Field center pixel classification performance
- Training field classification performance
- Full field classification performance
- RMS proportion estimate error
- Classification variability

These variables are related to the stated reasons for adopting a new classification technique listed in the introduction. The CPU time required to perform a classification is one way to measure the cost of classification. Field center pixel, full field, and training field performances and RMS proportion estimate error are all ways to evaluate the accuracy of the classifier. Classification variability is a measure of "salt and pepper effect" in classification results.

The CPU time required to execute each of the ECHO classifications has been recorded so that the effects of varying the cell homogeneity and annexation thresholds may be monitored. The CPU times required to perform the perpoint classifications have been adjusted to reflect the increased efficiency of the LARSYS perpoint classifier which is coded in assembly language. Thus, the CPU time recorded for a perpoint classification is what a FORTRAN classifier would have required to perform the classification.

The indices of classification performance were applied in several ways. Classification accuracy (identification) was evaluated utilizing field center pixel, "full field" and test field sample performance for all data sets. Proportion estimation was carried out for the LANDSAT and Simulated Thematic Mapper data sets.

The training performance is the overall classification accuracy (number of training pixels correctly classified divided by the total number of training pixels) of the pixels used to calculate the class statistics. Field center pixel performance is the overall classification accuracy of field center pixels. Field center pixels are pixels inset at least one pixel from the field boundaries. For the registered LACIE/SRS data the field center pixels are inset at least two pixels from the field boundary. Although this procedure insures that the pixels examined are not mixture pixels, it has the unfortunate effect of eliminating smaller fields from consideration. The third measure of classification accuracy, "full field" performance, includes those pixels on the boundaries of the fields in the classification performance. The "full field" pixels were generated by expanding the field center pixel boundaries one pixel in all directions.

The RMS error of informational class proportion estimates for each flightline was found by calculating the percent of the flightline classified as a particular class and comparing it with the ground-collected estimate using equation (1).

$$\text{RMS Error} = \sqrt{\frac{\sum_{i=1}^{N} (C_i - C'_i)^2}{N}} \qquad (1)$$

where, N = number of informational classes

C_i = percent classified as informational class i

C'_i = percent of class i estimated from ground-collected data

RMS error is calculated for the LANDSAT and Thematic Mapper data runs. The Agricultural Stabilization and Conservation Service (ASCS) provided the ground truth proportion estimates for the simulated Thematic

Mapper data set. Proportion estimates for the 1974 LACIE/SRS segments were provided in ground truth packets received from JSC. The SRS county proportion estimates were used to calculate RMS proportion error for the CITARS data set.

Average variability is a measure of the rate of change from one information class to another. It should reflect the degree to which ECHO reduces the "salt and pepper" effect which is sometimes present in per-point classifications. Variability is calculated by systematically selecting 50 lines of the classified area, counting the number of information class changes, and dividing by the number of opportunities for class changes.

$$\text{Variability} = NCC/(50*(NS-1)) \qquad (2)$$

Where

NCC = the number of class changes over the 50 selected lines

NS = the number of classified pixels/line

4. Results

This section outlines the results of the tests performed on the Supervised processor. Results are discussed separately for each scanner type, LANDSAT, simulated Thematic Mapper, and aircraft, followed by a summary of results for all three data types.

a. LANDSAT Results

Training and test information for the 10 data sets comprising the LANDSAT test data were drawn from the 1974 Kansas LACIE/SRS data sets and the 1973 CITARS data sets. One bitemporal and four unitemporal analyses were performed on the LACIE/SRS data sets. Six unitemporal analyses were performed on the CITARS data sets.

i. LACIE/SRS Results

Figures 5 through 10 present the average results for the four LACIE/SRS sites examined at a cell width of two for six cell homogeneity and four annexation parameters. Figure 5 plots the average CPU time in seconds required by the perpoint classifier (represented by the line of P's) versus the CPU time required by the average of the four LACIE/SRS data sets at each of 24 ECHO parameter settings for 2 by 2 pixel cells. The horizontal axis is the cell homogeneity threshold. As this threshold increases, it becomes more likely a cell will be classified as a unit, less likely that a cell will be split and its constituent pixels classified individually. The dependent variable, CPU time, appears on the vertical axis. The cell to field annexation parameter for each cell homogeneity threshold appears on the plot. When two or more annexation thresholds acheive the same performance, a star appears in that position on the plot. It can be seen from Figure 5 that as the cell homogeneity parameter increases, the CPU time required to classify the area decreases. Additionally, as the annexation parameter increases from zero to four, the CPU time required for the classification is reduced. These results are expected because fewer

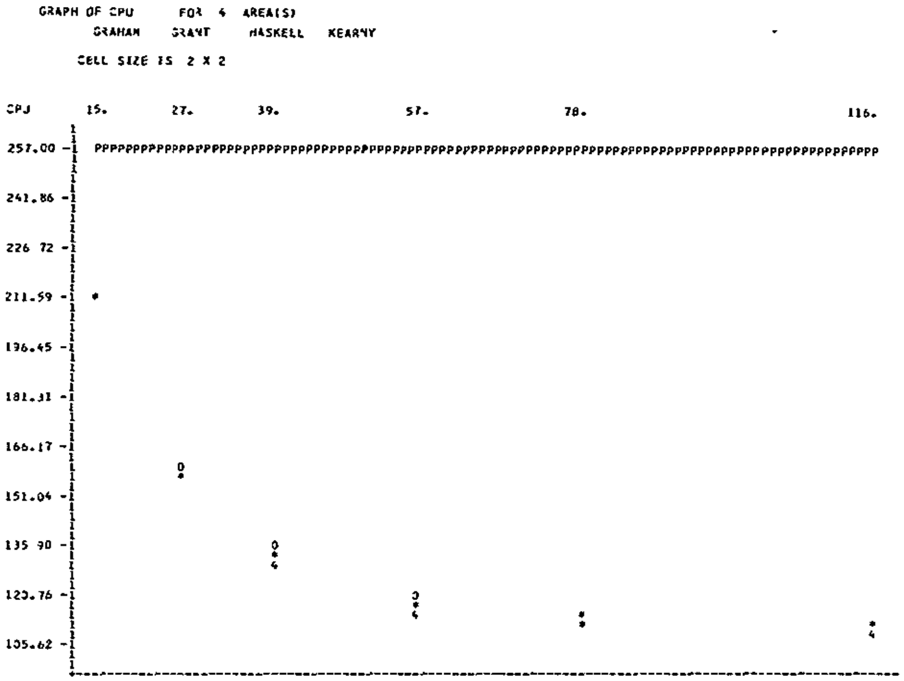


FIGURE 5

LACIE/SRS CPU TIME

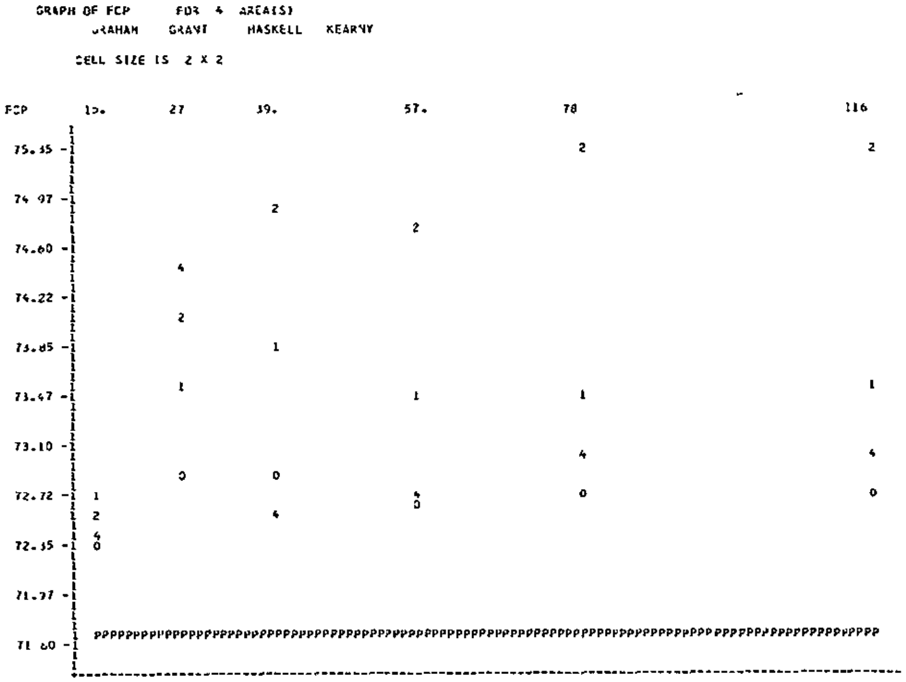


FIGURE 6

LACIE/SRS FIELD CENTER PIXEL PERFORMANCE

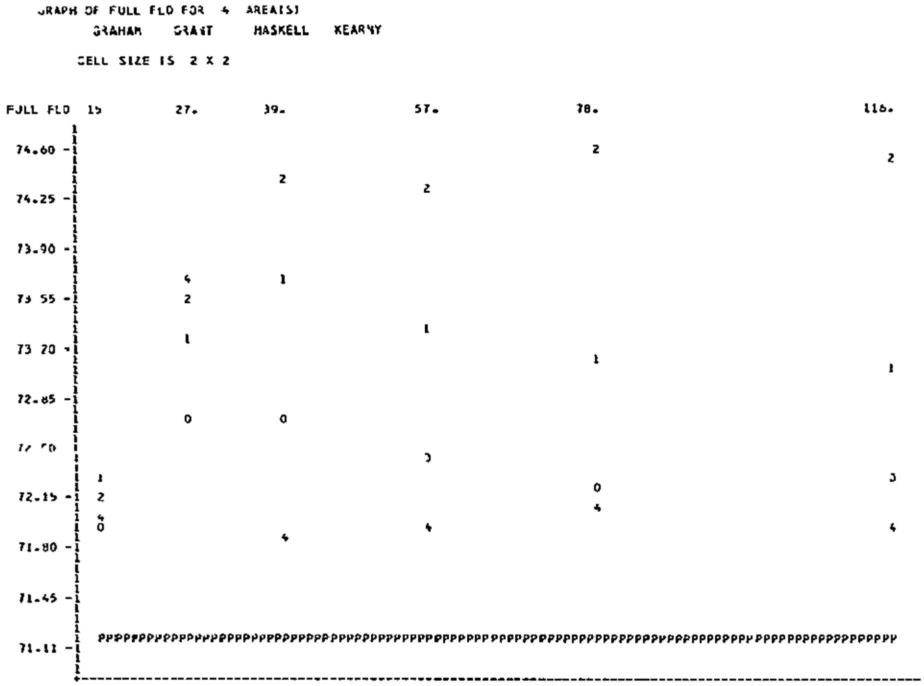


FIGURE 7

LACIE/SRS FULL FIELD PERFORMANCE

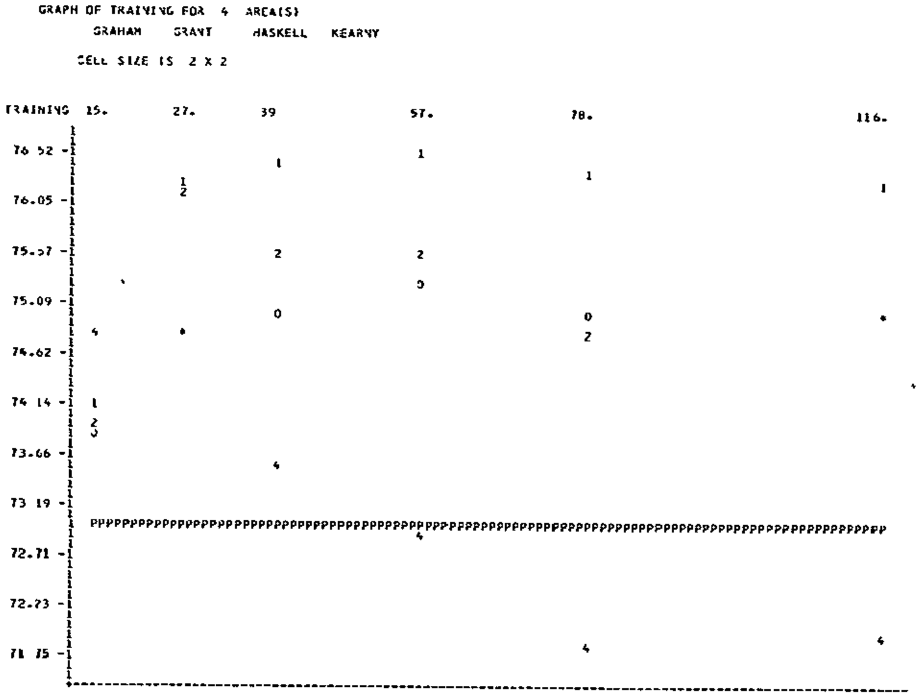


FIGURE 8

LACIE/SRS TRAINING PERFORMANCE

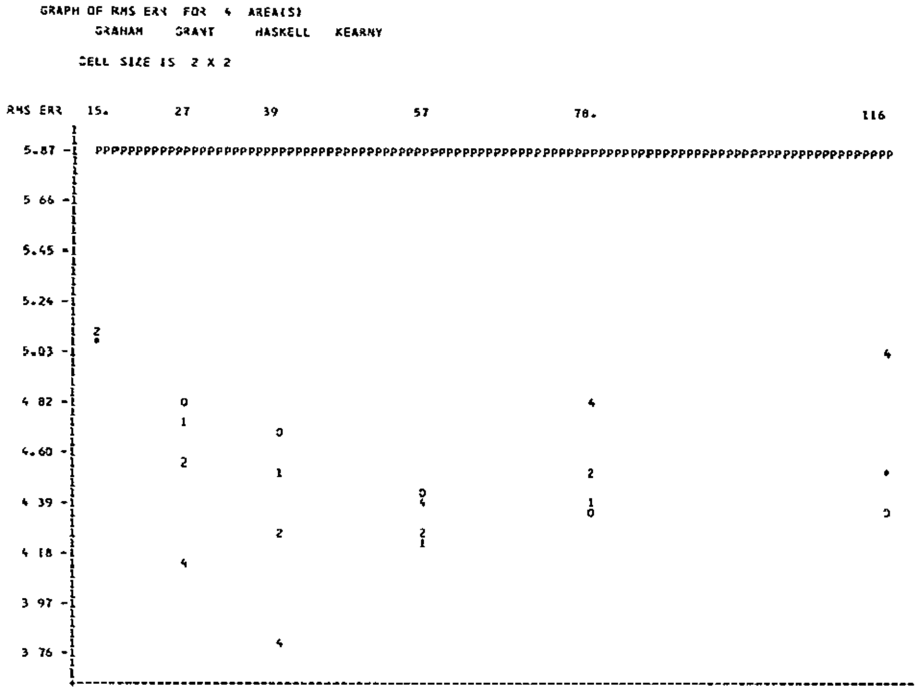


FIGURE 9

LACIE/SRS RMS PROPORTION ESTIMATION ERROR

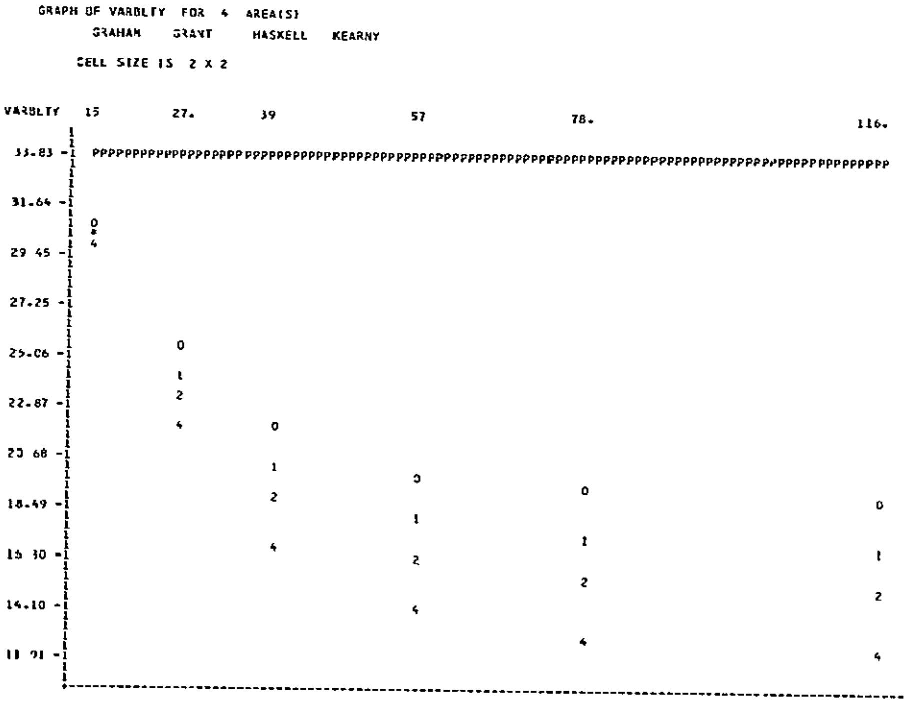


FIGURE 10

LACIE/SR CLASSIFICATION VARIABILITY

TABLE 2.1-5

EFFECT OF ECHO PARAMETERS ON INDIVIDUAL 1974 LACIE DATA SETS

ABBREVIATIONS:

CPU	CENTRAL PROCESSING UNIT TIME
FCP	FIELD CENTER PIXEL PERFORMANCE
FULL	FULL FIELD PERFORMANCE
TRAIN	TRAINING FIELD PERFORMANCE
RMS	RMS PROPORTION ERROR
VAR	CLASSFICATION VARIABILITY
CELW	CELL WIDTH
HOM	CELL HOMOGENEITY THRESHOLD
ANN	ANNEXATION THRESHOLD

ANALYSIS OF VARIANCE RESULTS (ENTRIES ARE SIGNIFICANCE LEVELS):

				CELW	CELW	HOM
	CELW	HOM	ANN	X	X	X
				HOM	ANN	ANN
Graham County, Kansas						
CPU	.1	.1	7.5	.1	24.7	99.9
FCP	.1	.1	18.7	.1	.1	18.9
FULL	.1	.1	99.9	.1	1.6	33.6
TRAIN	.1	.1	.1	.1	.1	.2
RMS	.1	.1	.1	.1	.1	99.9
VAR	.1	.1	.1	.1	.1	.1
Grant County, Kansas						
CPU	.1	.1	.8	.1	2.2	99.9
FCP	.1	.1	.1	.1	.1	99.9
FULL	.1	.1	.1	.1	.1	99.9
TRAIN	.1	.1	.1	.1	.1	34.9
RMS	.1	.1	99.9	.1	99.9	41.8
VAR	.1	.1	.1	.1	.1	40.3
Haskell County, Kansas						
CPU	.1	.1	.2	.1	.1	33.0
FCP	.1	.1	.1	.1	.1	25.5
FULL	.1	.1	.3	.1	.1	33.5
TRAIN	.1	.1	.1	.1	.1	29.5
RMS	.1	.1	.1	.1	.1	9.1
VAR	.1	.1	.1	.1	.1	27.1

TABLE 2.1-5 (Continued)

EFFECT OF ECHO PARAMETERS ON INDIVIDUAL 1974 LACIE DATA SETS

	CELW	HOM	ANN	CELW X HOM	CELW X ANN	HOM X ANN
Kearny County, Kansas						
CPU	.1	.1	.1	.1	.4	99.9
FCP	.1	.1	6.5	.1	.1	99.9
FULL	.1	.1	1.6	.1	.1	99.9
TRAIN	.1	.1	.3	.1	.1	13.0
RMS	.1	.1	.1	.1	15.3	2.0
VAR	.1	.1	.1	.1	.1	.1

Table 2.1-6

EFFECT OF ECHO PARAMETERS ON FOUR LACIE DATA SETS

	Cell Width	Cell Homogeneity Threshold	Annexation Threshold	Cell Width X Cell Homogeneity Threshold	Cell Width X Annexation Threshold	Cell Homogeneity Threshold X Annexation Threshold
Degrees of Freedom	2,6	5,15	3,9			
CPU Time		1	1	.1	99.9	99.9
Field Center Pixel Performance	5	5		.1	.4	99.9
Full Field Performance	5	1	25	.1	.6	99.9
Training Performance	10	1		.1	.1	99.9
RMS Error		25		.1	99.9	99.9
Classification Variability	5	1	1	.1	.1	13.6

Entires are Significance Levels

classifications are necessary when there are fewer cells which must be classified on a point-by-point basis (due to the increased cell homogeneity parameter), and fewer classifications are required when a scene is composed of a few large objects rather than many small objects (the effect of increasing the annexation parameter). Both of these effects are statistically significant at a 1% confidence level.

The statistical significance levels of the various ECHO parameters on each of the six dependent variables (CPU time, field center pixel performance, "full field" performance, training field performance, RMS proportion error, and classification variability) for each of the LACIE data sets are presented in Table 2.1-5. Table 2.1-6 presents the statistical significance levels of the ECHO parameters on the dependent variables for the four LACIE data sets combined.

Parameter Considerations

For the LACIE data runs, cell widths were sampled at 2, $\sqrt{\text{average field size}}/2 + 1$, and $\sqrt{\text{average fields size}}$; hence cell widths sampled were 2, 5 or 6, and 8 or 9. The field center pixel performances at the 2 by 2 cell size tended to be superior to those at 5 by 5 or 6 by 6; RMS proportion estimation errors of 2 by 2 cells were about equivalent to those of the 5 by 5 or 6 by 6 cells; and less computer time was required for the 5 by 5 or 6 by 6 sized cells than the 2 by 2 cells.

Graham County was also sampled at cell sizes 3 by 3 and 4 by 4 to observe what effects intermediate cell sizes might have. Results for the optimal values of these parameters are presented below:

Graham County, Kansas

Optimal Results

Cell Size	CPU Range (Sec)	FCP Performance %	RMS Error %
2x2	109-73	93.9	0.1
3x3	101-59	94.3	0.1
4x4	95-50	93.9	0.0
6x6	98-51	93.3	0.7
Perpoint	121	89.9	1.2

These results tend to indicate that cell width may be optimized for the LACIE/SRS runs when it is in the neighborhood of the fourth root of the average field size. This value is between three and four for Graham County.

For the 2 by 2 tests (performed on all four LACIE/SRS sites) the optimal cell homogeneity parameter, based on the three fields performance criteria, appears to be 78. However, on the basis of RMS error, the optimal homogeneity parameter tested is 39.

Due to the two pixel inset used to avoid mixture pixels and registration error for the LACIE/SRS field coordinates, small fields were not used for training or test purposes. Since only relatively large fields are included in the training decks, the effects that increasing the cell homogeneity parameter may have on the small fields is not reflected in the field center pixel performance results. Consequently, for the LACIE results, 39 is probably closer to the optimal homogeneity parameter than 78. Support for this conclusion is provided by the results in Figure 8. The "full field" performance measures the performance of the large test fields, including the points which contain the field boundaries. "Full field" test fields are composed of the pixels contained in the field center pixel test fields plus those pixels on the borders of the field ("fringe pixels"). Since for the LACIE data sets, the performance of the field center pixels improves by .5% between cell homogeneity values 39 and 78 while the full field performance only improved by .2%, it would appear that the performance of the "fringe pixels" is degraded between homogeneity parameters 39 and 78. Those small fields not included in the test decks will be composed primarily of "fringe pixels".

It can be noted by examining Figures 6 through 9 that the effect of annexation increases as the cell homogeneity parameter increases. This result is expected since a cell A only has the opportunity to be annexed to adjoining cell B when both cells A and B are homogeneous. The higher the cell homogeneity parameter, the more likely it becomes that both A and B will be homogeneous.

Figure 10 is typical of all the variability results: as the homogeneity and annexation parameters increase, the classification variability decreases.

LACIE/SRS Conclusions

1. CPU time and classification variablity decrease as cell homogeneity and cell to field annexation parameters increase (1% significance level).

2. The optimal cell width parameter may fall between two and average field size/2 + 1. The fourth root of the average field size may be an appropriate value.

3. For these data sets, the optimal annexation value appears to be in the neighborhood of two based on field center pixel and full field performances.

4. For a cell width of two, the optimal cell homogeneity parameter is in the neighborhood of 39, based on RMS proportion estimate error.

ii. CITARS Results

Figures 11 through 16 present plots for the 2 by 2 cell size Supervised ECHO results achieved from the CITARS data sets. Characteristic differences between the CITARS and the LACIE/SRS data sets include:

* The CITARS data set has a much smaller average field size than the LACIE/SRS set.

* The information classes are different. CITARS information classes are Corn, Soybeans and Other; LACIE/SRS classes are Wheat and Other.

* The ground truth proportion estimates used for the LACIE/SRS sites were for the area of the LACIE segment. The ground truth proportion estimates used for the CITARS sites are for the whole county in which the data set lies, not for that area of the county which was actually sampled.

Analysis of variance results for the six CITARS data sets are presented in Tables 2.1-7 and 2.1-8. Table 2.1-7 presents results for each data set individually. Table 2.1-8 presents the analysis of variance results for the combined CITARS data sets.

As with the 1974 LACIE/SRS data sets, the CPU time decreased as cell homogeneity and annexation parameters increase.

However, though Figure 5 (CPU time for the LACIE/SRS data set) and Figure 11 (CITARS CPU time) have the same general shape, there is one important difference: the CPU time required to perform the ECHO classification is a higher percentage of the CPU time required to perform the perpoint classification for each cell homogeneity value. This is a reflection of the smaller field size in the CITARS data set. Since fields are smaller, more boundaries and fewer homogeneous cells exist. As a result, more cells must be classified on a point-by-point basis and more CPU time

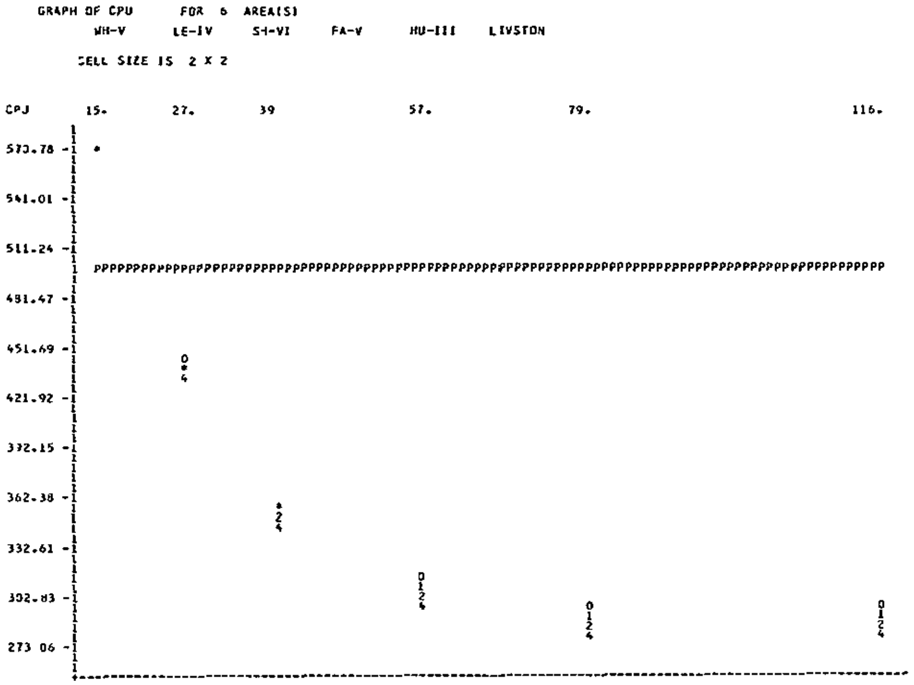


FIGURE 11

CITARS CPU TIMES

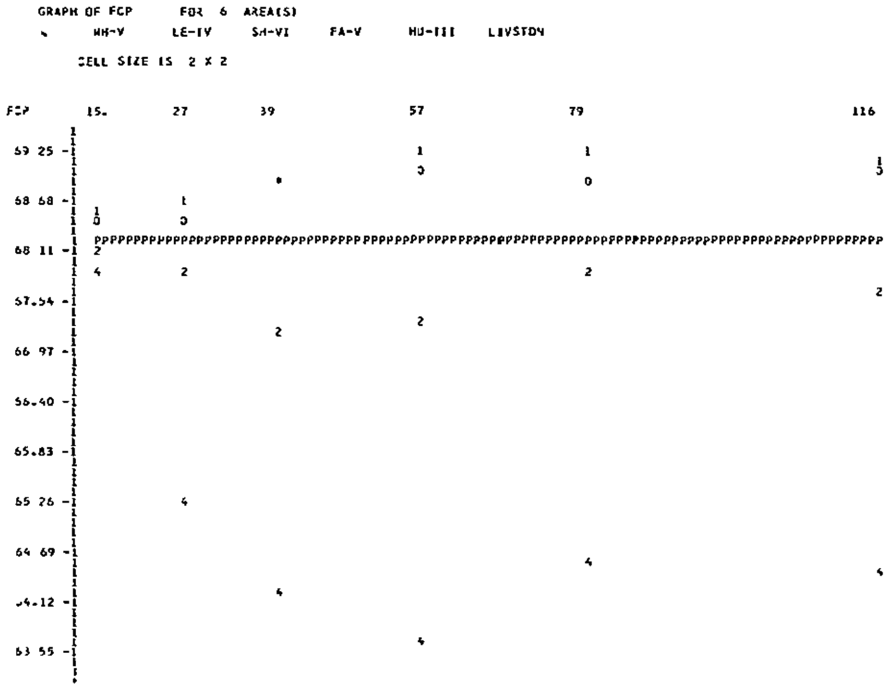


FIGURE 12

CITARS FIELD CENTER PIXEL PERFORMANCE

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

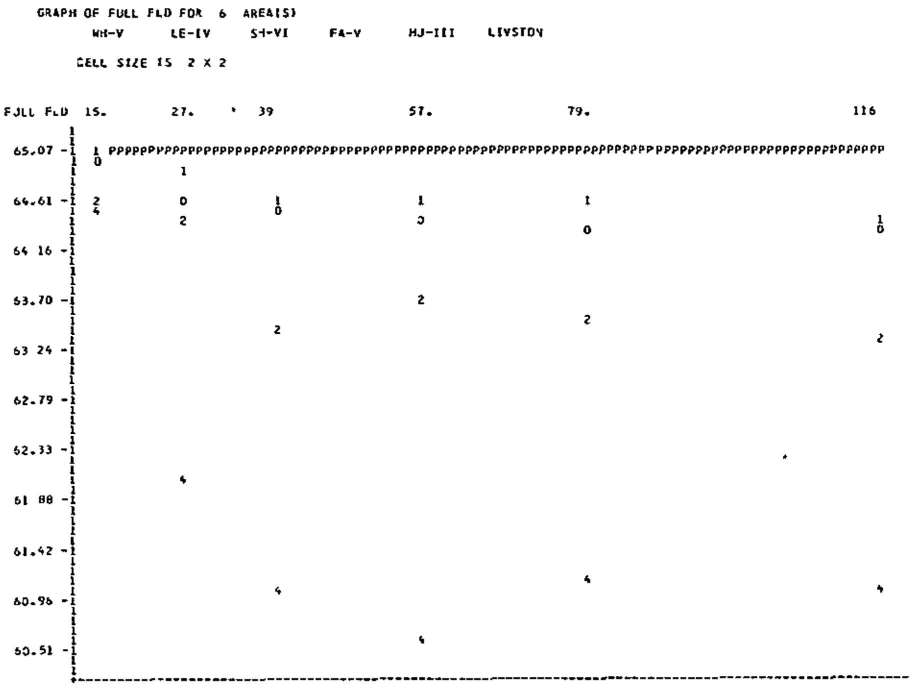


FIGURE 13

CITARS FULL FIELD PERFORMANCE

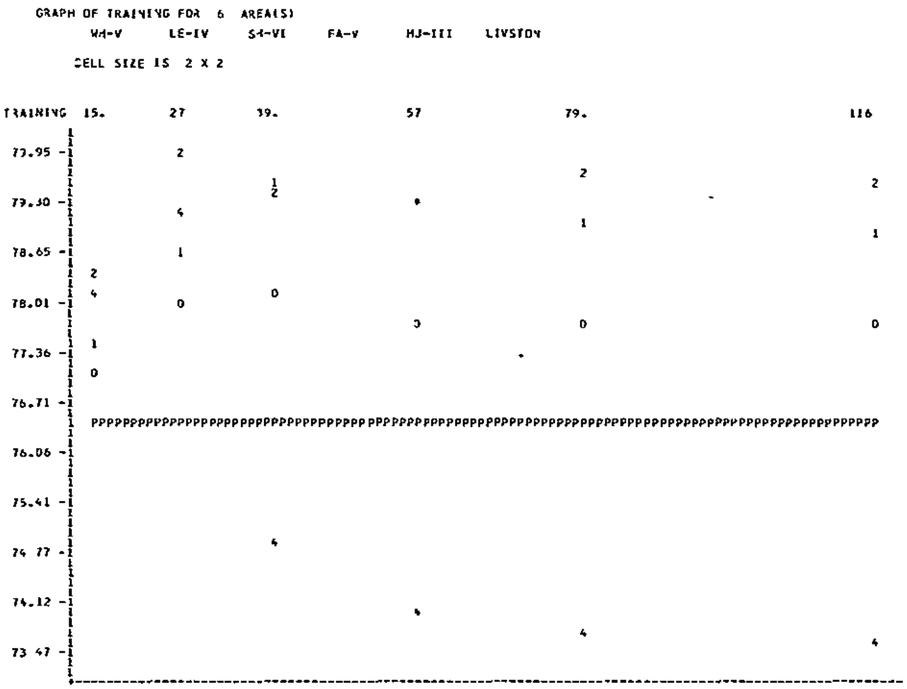


FIGURE 14

CITARS TRAINING ACCURACY

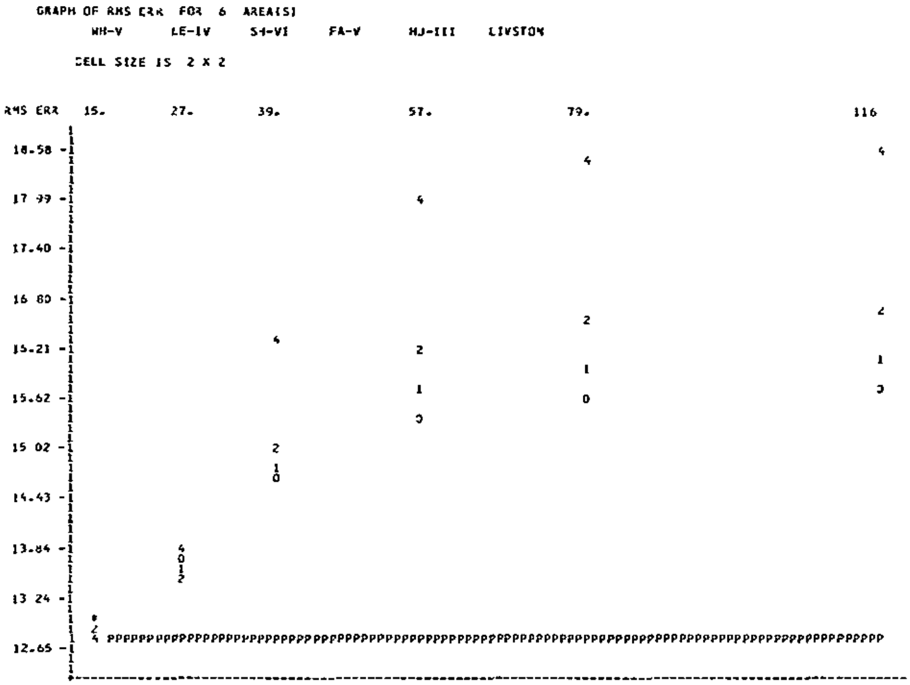


FIGURE 15

CITARS RMS PROPORTION ESTIMATE ERROR

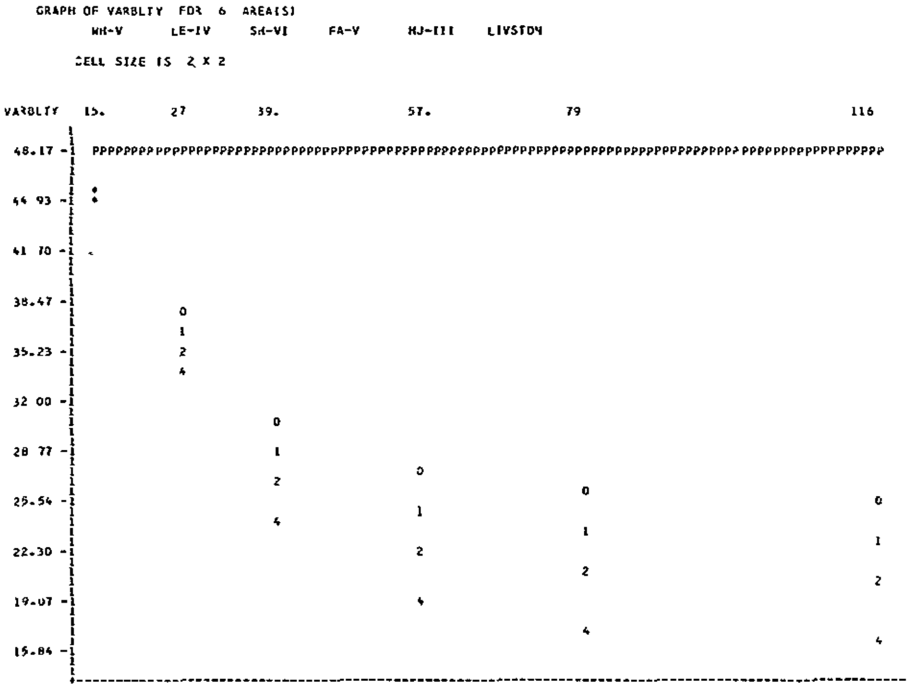


FIGURE 16

CITARS CLASSIFICATION VARIABILITY

TABLE 2.1-7

EFFECT OF ECHO PARAMETERS ON INDIVIDUAL CITARS DATA SETS

ABBREVIATIONS:

CPU	CENTRAL PROCESSING UNIT (COMPUTER) TIME
FCP	FIELD CENTER PIXEL PERFORMANCE
FULL	FULL FIELD PERFORMANCE
TRAIN	TRAINING FIELD PERFORMANCE
RMS	RMS PROPORTION ESTIMATE ERROR
VAR	CLASSIFICATION VARIABILITY
CELW	CELL WIDTH
HOM	CELL HOMOGENEITY THRESHOLD
ANN	ANNEXATION THRESHOLD

ANALYSIS OF VARIANCE RESULTS (ENTRIES ARE SIGNIFICANCE LEVELS):

	CELW	HOM	ANN	CELW X HOM	CELW X ANN	HOM X ANN
Huntington County, Indiana						
CPU	.1	.1	.1	.1	.1	12.7
FCP	.7	.1	99.9	.1	.3	5.9
FULL	37.4	.1	37.1	.1	1.3	20.6
TRAIN	.1	.1	.1	.1	.1	12.9
RMS	.1	.1	99.9	.1	32.7	39.2
VAR	.1	.1	.1	.1	.1	.1
Shelby County, Indiana						
CPU	.1	.1	.1	.1	.1	44.7
FCP	.1	.1	.1	.1	.1	.1
FULL	.1	.1	.1	.1	.1	.1
TRAIN	.1	.4	.1	27.2	.1	.8
RMS	.1	.1	.1	.1	.1	.1
VAR	.1	.1	.1	.1	.1	.1
White County, Indiana						
CPU	.1	.1	.1	.1	.4	99.9
FCP	.1	.1	.1	.1	.1	99.9
FULL	.1	.1	.1	.1	.1	22.8
TRAIN	.1	.1	.1	.1	.1	99.9
RMS	.1	.1	.1	.1	.1	43.1
VAR	.1	.1	.1	1 .1	.1	.2

TABLE 2.1-7 (Continued)

EFFECT OF ECHO PARAMETERS ON INDIVIDUAL CITARS DATA SETS

	CELW	HOM	ANN	CELW X HOM	CELW X ANN	HOM X ANN
Livingston County, Illinois						
CPU	.1	.1	.1	.1	.1	23.0
FCP	.1	14.8	.1	.1	.1	22.1
FULL	.1	.1	.1	.1	.1	3.2
TRAIN	.1	.1	.1	.1	.1	18.7
RMS	.1	.1	.1	.1	1.8	12.9
VAR	.1	.1	.1	.1	.1	.1
Fayette County, Illinois						
CPU	.1	.1	1.1	.1	3.2	99.9
FCP	.1	.1	.1	.1	.1	17.8
FULL	.1	.1	.1	.1	1.8	17.1
TRAIN	.1	.1	2.9	12.6	.4	99.9
RMS	.1	.1	.1	.1	.1	.1
VAR	.1	.1	.1	.1	.1	.3
Lee County, Illinois						
CPU	.1	.1	.1	.1	.1	17.3
FCP	.1	.1	34.9	39.4	.1	99.9
FULL	.1	.1	99.9	7.5	.1	99.9
TRAIN	.1	.1	.1	.1	.2	11.0
RMS	.2	.1	.1	.1	.1	.1
VAR	.1	.1	.1	.1	.1	1.8

Table 2.1-8

EFFECT OF ECHO PARAMETERS ON SIX CITARS DATA SETS

	Cell Width	Cell Homogeneity Threshold	Annexation Threshold	Cell Width X Cell Homogeneity Threshold	Cell Width X Annexation Threshold	Cell Homogeneity Threshold X Annexation Threshold
Degrees of Freedom	2,10	5,25	3,15			
CPU Time	1	1	1	.1	19.3	99.9
Field Center Pixel Performance	10	10	25	.1	.1	34.1
Full Field Performance	10	5	25	.1	.1	99.9
Training Performance	1	1	25	.1	.1	12.0
RMS Error	99	10	25	.1	.1	28.2
Classification Variability	1	1	1	.1	.1	.1

Entries are Significance Levels

is required to perform those point-by-point classifications. For the smallest cell homogeneity parameter, ECHO takes more time on the average than the perpoint classifier. This is due to a great majority of the cells being split. For a cell that is split, the ECHO processor must first gather statistics and perform the homogeneity test, at some expense in CPU time. Then each point of the singular cell must be classified on a point-by-point basis which takes approximately the same CPU time as the perpoint classifier. Hence, when the great majority of cells are singular, ECHO is less efficient, in terms of CPU time, than the perpoint classifier.

By examining the CITARS field center pixel performance (Figure 12) in light of the LACIE/SRS results, two conclusions can be drawn:

1. The improvement in classification accuracy the ECHO procedure provides is reduced for the CITARS data sets, as compared to the LACIE set.

2. The optimal annexation parameter is smaller for the CITARS data sets than for the LACIE data sets.

It is evident by comparing Figures 12 and 14, that the CITARS training set is not representative of the test fields for the ECHO data sets. The performance of the training fields using ECHO is markedly superior to the performance of those same fields using the perpoint classifier. However, the field center pixel performance of the ECHO algorithm is only marginally superior in magnitude to the performance of the perpoint classifier. This indicates that on the data which the statistics truly represent, ECHO performance is superior; when the first and second order statistics fail to properly represent the classes, especially in areas with small field sizes, ECHO's performance may be no better than that of the perpoint classifier.

The average RMS error achieved by ECHO for the six CITARS data sets is inferior to the average RMS error achieved by the perpoint classifier using the same class statistics. On a flightline by flightline basis, ECHO's RMS proportion estimate error was higher than the perpoint RMS error for three of the data sets and lower than the perpoint RMS error for the other three data sets. This may be partially due to one or more of the following:

* The RMS error values are generally high. The proportion estimates used for "truth" are the SRS estimates for the counties as a whole, not for the individual areas classified. Hence, if an area classified is not a good approximation of the county as a whole, the validity of the RMS results must be questioned.
* The training fields do not appear to be a good representation of the test fields. If the training statistics do not represent the scene as a whole, ECHO performance may suffer to a greater extent than the perpoint performance.
* When the average field size is small, there will be a high proportion of "mixture" cells (cells containing points of more than one class). Should a cell containing pixels of differing classes pass the homogeneity criteria, it is likely to be categorized into the class with the highest variance. Thus as the cell homogeneity criteria becomes larger, the RMS error would tend to increase. (An observed result - see Figure 15.)

The effect of cell width on RMS error was not significant when all the CITARS data sets are considered at once. However, cell width produced significantly different results in terms of RMS error for each of the six data sets taken individually. The effects of increasing cell width were opposite for different data sets.

Conclusions

* As the field size becomes smaller, the improvement in classification accuracy provided by the supervised ECHO algorithm is reduced. At the same time, due to the increased number of singular cells, the CPU time which Supervised ECHO requires increases.

* When the training set is representative of the data, ECHO's classification performance continues to be superior to that of the per-point classifier for scenes with average fields at or below 16 to 20 pixels.

* Due to the mixture-cell, large-variance-class problem, ECHO may or may not degrade RMS proportion Estimation for areas with relatively small fields.

iii. Bitemporal Results

Figures 17 through 22 illustrate the Supervised ECHO results for the Bitemporal analysis. Figures 17a through 22a are the results of a unitemporal analysis of the same area. The training for the bitemporal analysis was done independently of the training for the unitemporal analysis. Comparing Figures 18 and 18a it can be seen that the ECHO field center pixel performance is a greater improvement over the perpoint classifer's performance in the unitemporal case. This may partially be a result of some misregistration between dates making object extraction more difficult and partially a function of the different training statistics used for the separate classifications.

With the larger number of channels in the multidate classification, the optimal homogeneity threshold is higher than it would be for the single date classification. This is expected as, for the Supervised algorithm, the homogeneity criteria approximates to a chi-squared statistic with (number of channels) times (cell width squared) degrees of freedom. For the same significance level of test, the cell homogeneity criteria should be higher for the eight channel data than for the four channel data.

Based on the field center pixel performance and the RMS proportion error results, the optimal settings of the homogeneity and annexation parameters appear to be in the neighborhood of 180 and 1, respectively.

Note that although the multitemporal analysis does not greatly improve the field center pixel classification accuracy, it does (especially for the perpoint classifier) substantially reduce the RMS proportion estimate error. In the Bitemporal analysis the perpoint RMS error is only .2%, leaving ECHO little room for improvement. Note also that for several ECHO parameter settings in the unitemporal analysis, ECHO performs with an RMS

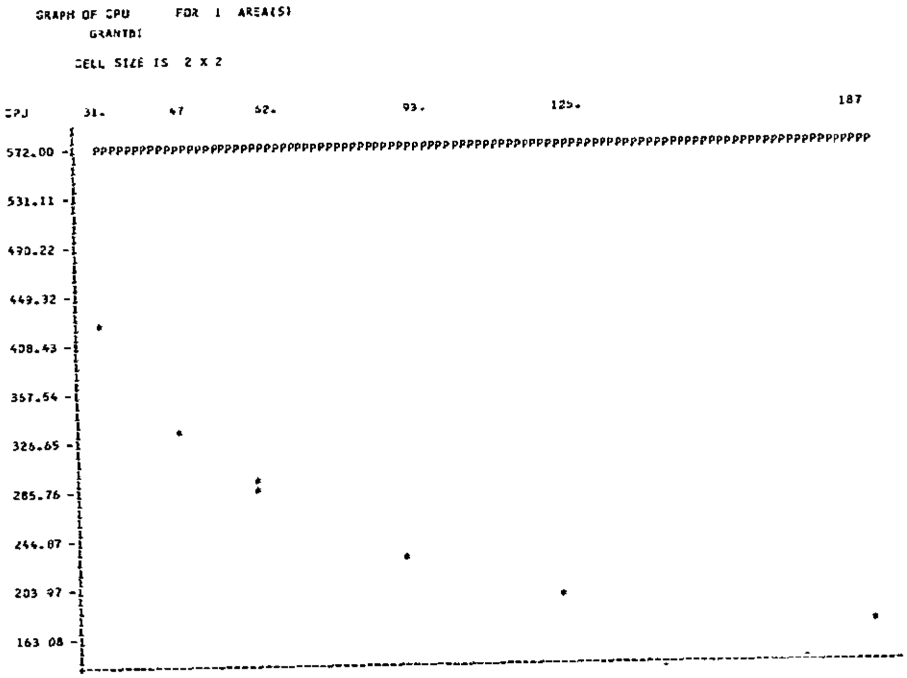


FIGURE 17

GRANT COUNTY BITEMPORAL CPU TIMES

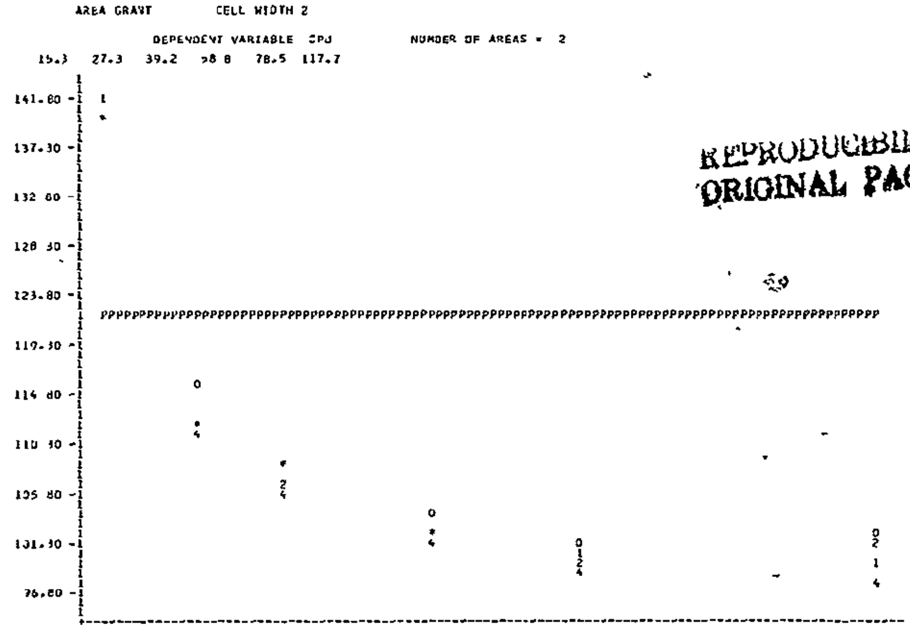


FIGURE 17a

GRANT COUNTY UNITEMPORAL CPU TIMES

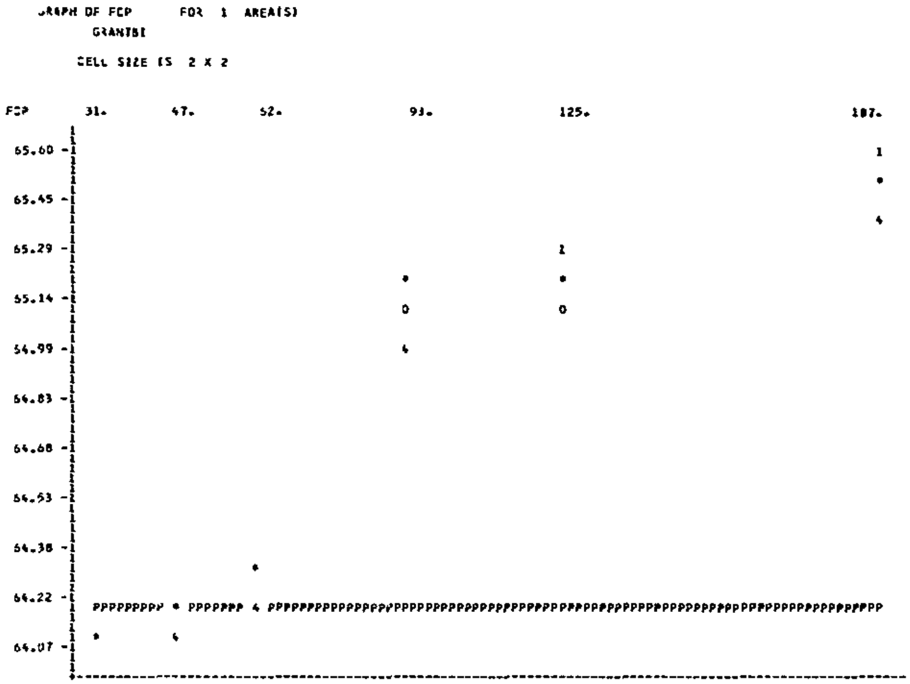


FIGURE 18

GRANT COUNTY BITEMPORAL FIELD CENTER PIXEL PERFORMANCE

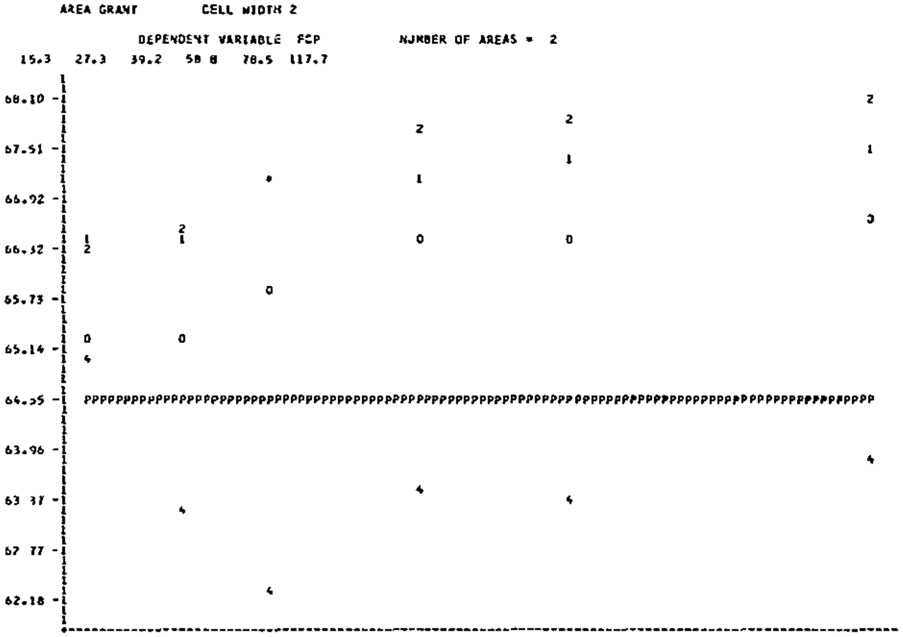


FIGURE 18a

GRANT COUNTY UNITEMPORAL FIELD CENTER PIXEL PERFORMANCE

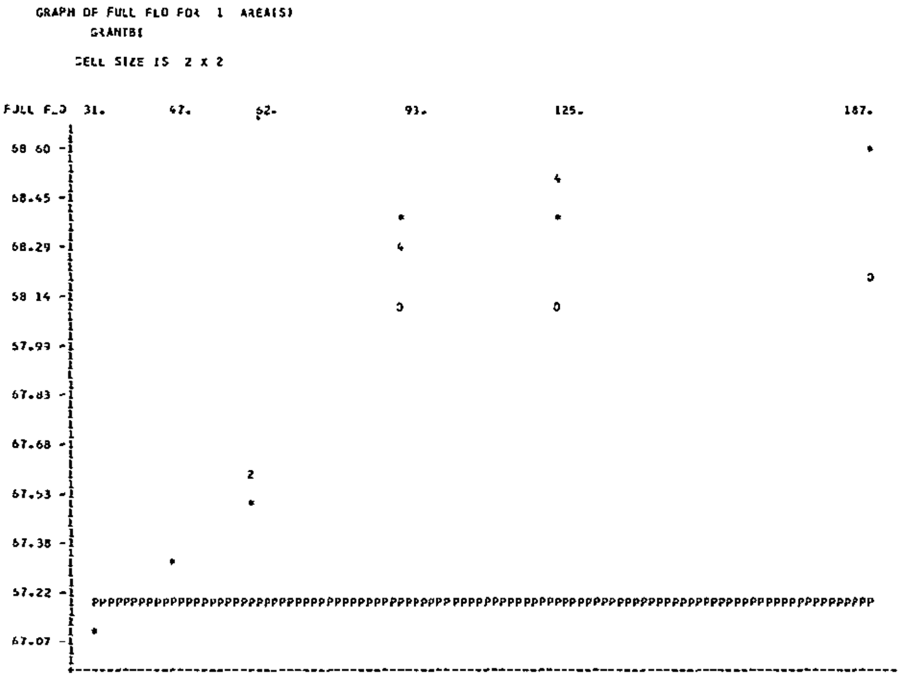


FIGURE 19

GRANT COUNTY BITEMPORAL FULL FIELD PERFORMANCE

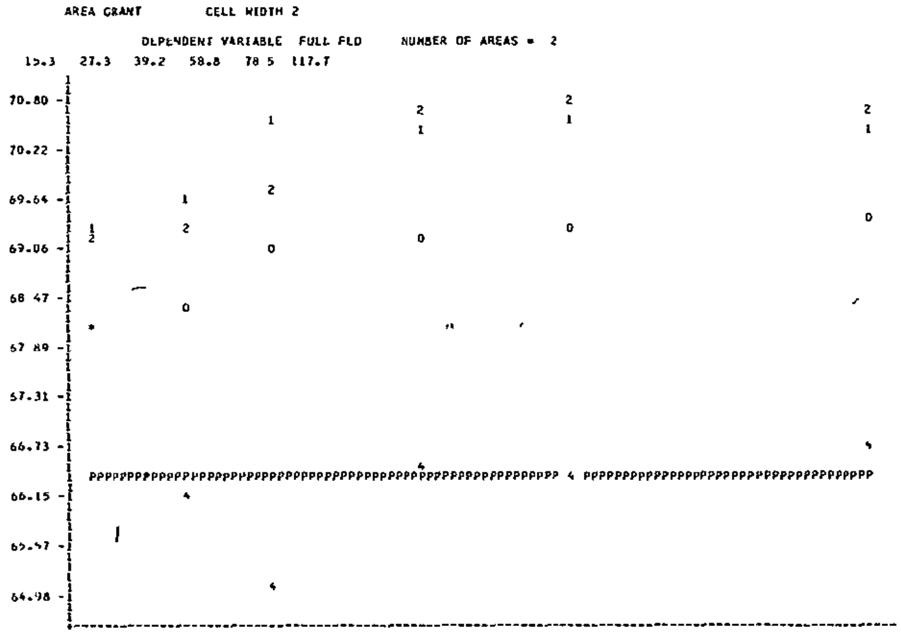


FIGURE 19a

GRANT COUNTY UNITEMPORAL FULL FIELD PERFORMANCE

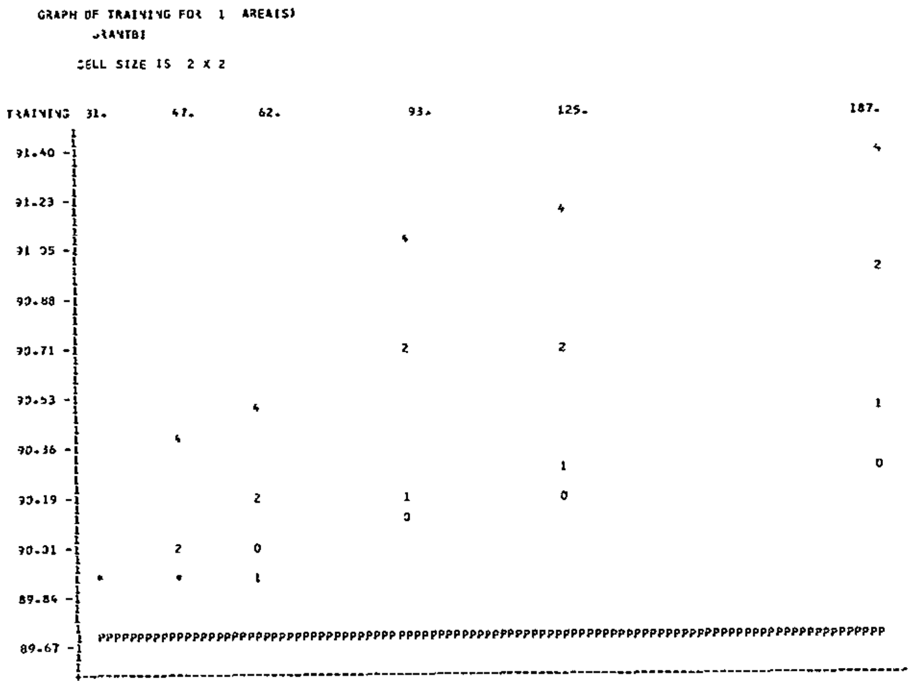


FIGURE 20

GRANT COUNTY BITEMPORAL TRAINING FIELD PERFORMANCE

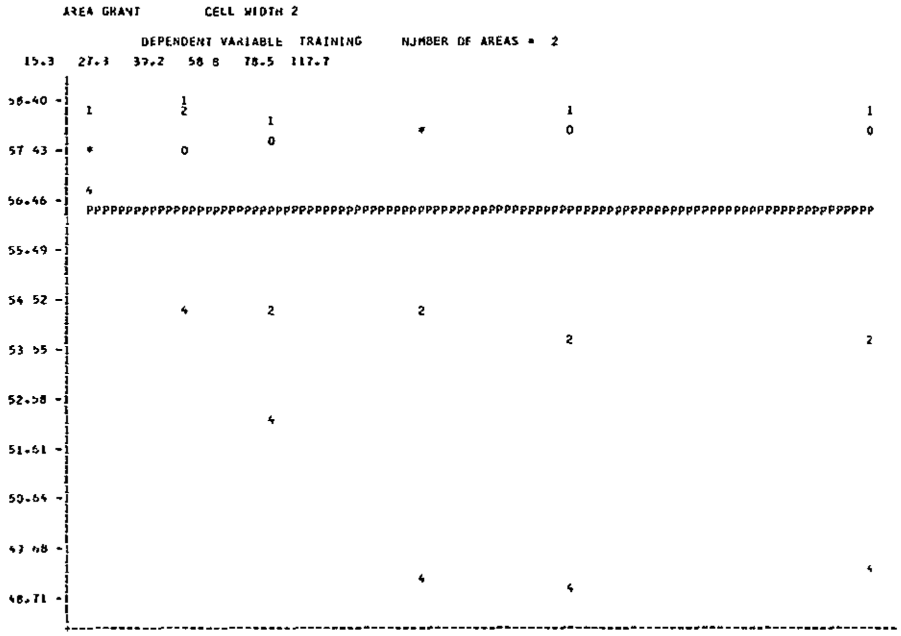


FIGURE 20a

GRANT COUNTY UNITEMPORAL TRAINING FIELD PERFORMANCE

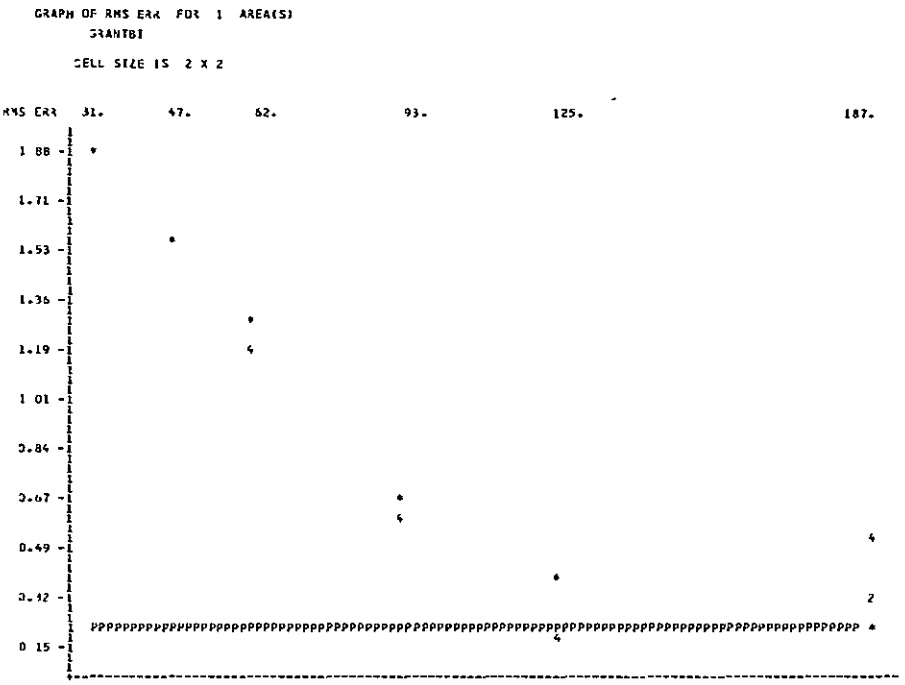
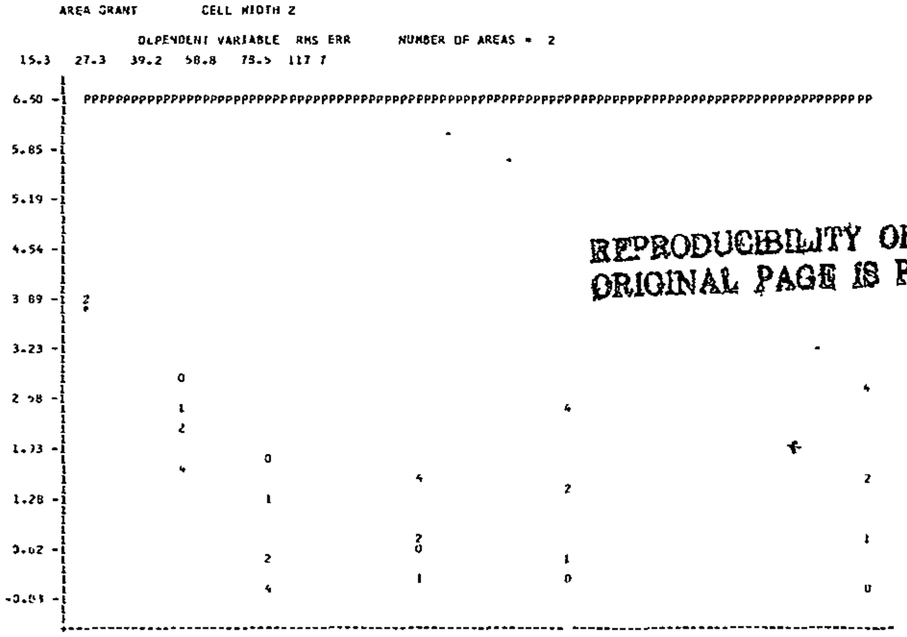


FIGURE 21

GRANT COUNTY BITEMPORAL RMS PROPORTION ESTIMATE ERROR



REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

FIGURE 21a

GRANT COUNTY UNITEMPORAL RMS PROPORTION ESTIMATE ERROR

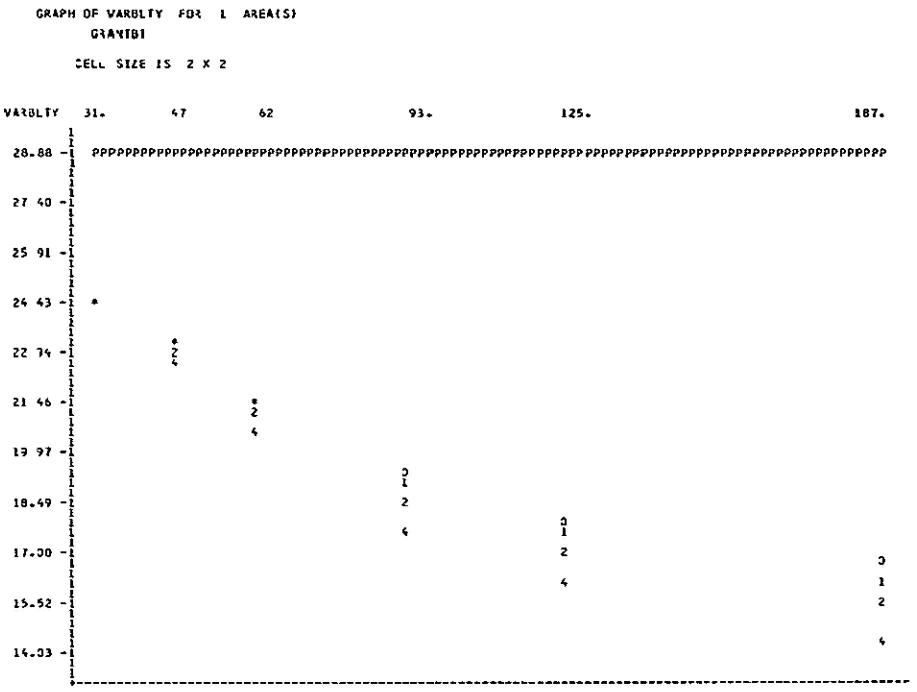


FIGURE 22

GRANT COUNTY BITEMPORAL CLASSIFICATION VARIABILITY

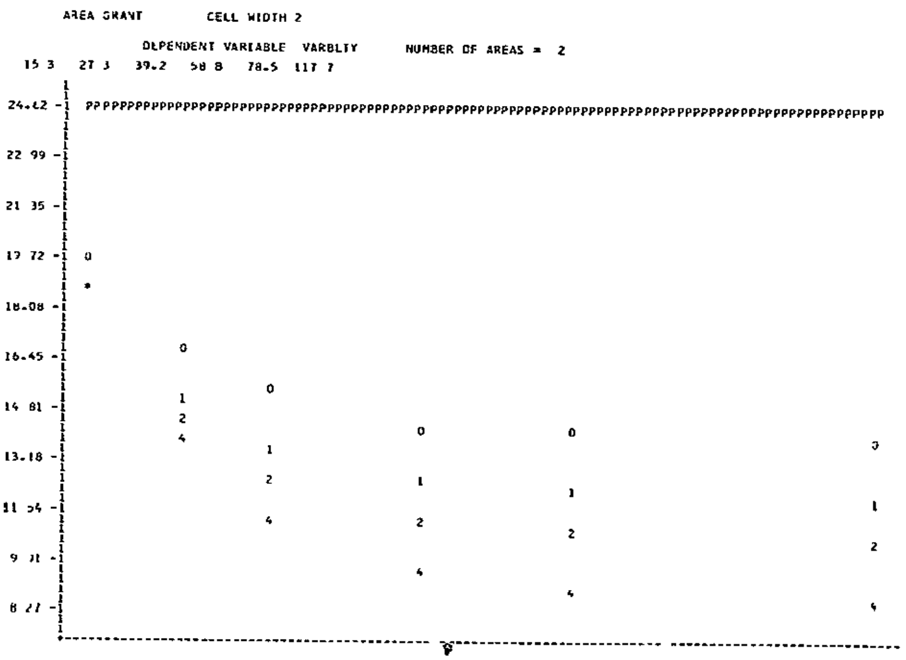


FIGURE 22a

GRANT COUNTY UNITEMPORAL CLASSIFICATION VARIABILITY proportion estimate error of under .5% while the perpoint classifier is functioning with a 6.5% RMS error.

In the multitemporal case, ECHO requires less CPU time than the perpoint classifier for all parameter settings. This is not true in the unitemporal case.

Bitemporal Conclusions

* The Supervised ECHO processor produced classifications superior to those produced by the perpoint classifier in terms of CPU time and field center pixel accuracy for the frame analyzed.

* The optimal homogeneity parameter for bitemporal analysis appears to be much larger than for the unitemporal case.

* The CPU time required and classification variability produced by ECHO are smaller than those of the perpoint classifier.

iv. Overall LANDSAT Results

Table 2.1-9 summarizes the effects of the ECHO parameter settings on the six dependent variables. Figures 23 through 28 plot the average ECHO performance at the various ECHO parameter settings versus the average perpoint performance for the ten LANDSAT data sets.

The cell width setting effects are significant for all dependent variables for all LANDSAT data sets considered individually, and are significant for all dependent variables except RMS proportion error when all LANDSAT data sets are considered together, using data sets as blocks.

When the LACIE and CITARS LANDSAT runs are considered separately, cell width is significant for CPU time, training performance, and classification variability for the CITARS data sets and for CPU time, field center pixel performance, "full field" performance, and classification variability in the LACIE data sets.

The statistically significant (5% significance level) overall cell width effects for the LANDSAT data sets are:

- As cell width increases, CPU time decreases.
- As cell width increases, field center pixel, "full field" and training field performances decrease.
- As cell width increases, classification variability decreases.

Though not statistically significant, the trend is for RMS error to increase as cell width increases. Two things should be noted about the cell width overall results:

* It is not uniformly true that all cell width increases, field center pixel, "full field" and training field performances decrease.

Table 2.1-9

EFFECT OF ECHO PARAMETERS ON SIX VARIABLES USING DATA SET

AS A BLOCK (ALL TEN LANDSAT DATA SETS)

	Cell Width	Cell Homogeneity Threshold	Annexation Threshold	Cell Width X Cell Homogeneity Threshold	Cell Width X Annexation Threshold	Cell Homogeneity Threshold X Annexation Threshold
Degrees of Freedom	2,18	5,45	3,27			
CPU Time	5	1	1	.1	99.9	99.9
Field Center Pixel Performance	1	1	25	.1	.1	31.5
Full Field Performance	1	1	25	.1	.1	99.9
Training Performance	1	1	10	.1	.1	1.1
RMS Error	25	10	25	.1	3.2	99.9
Classification Variability	1	1	1	.1	.1	.1

Entries are Significance Levels

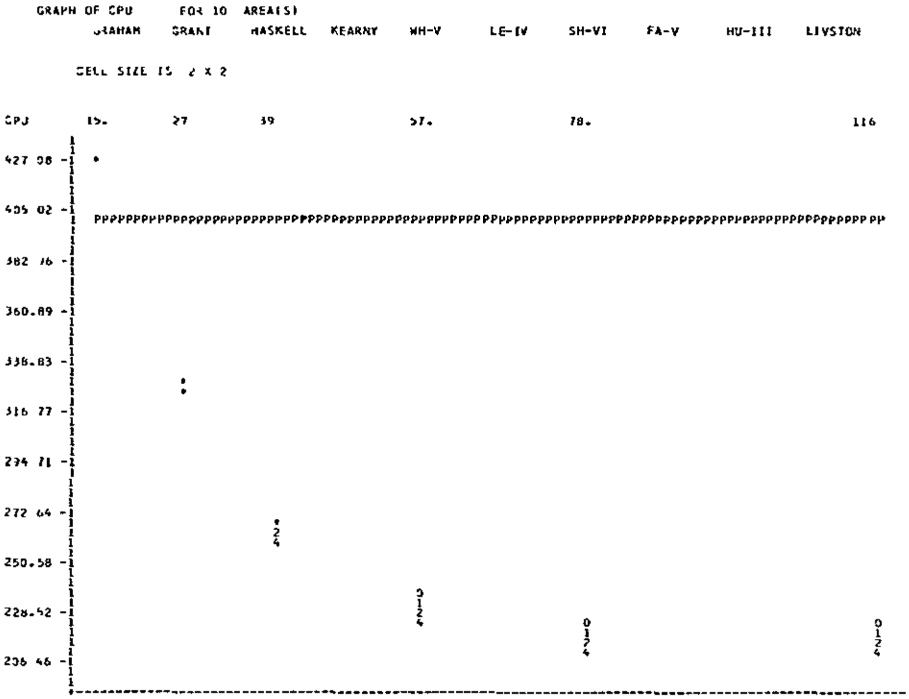


FIGURE 23

LANDSAT CPU TIMES

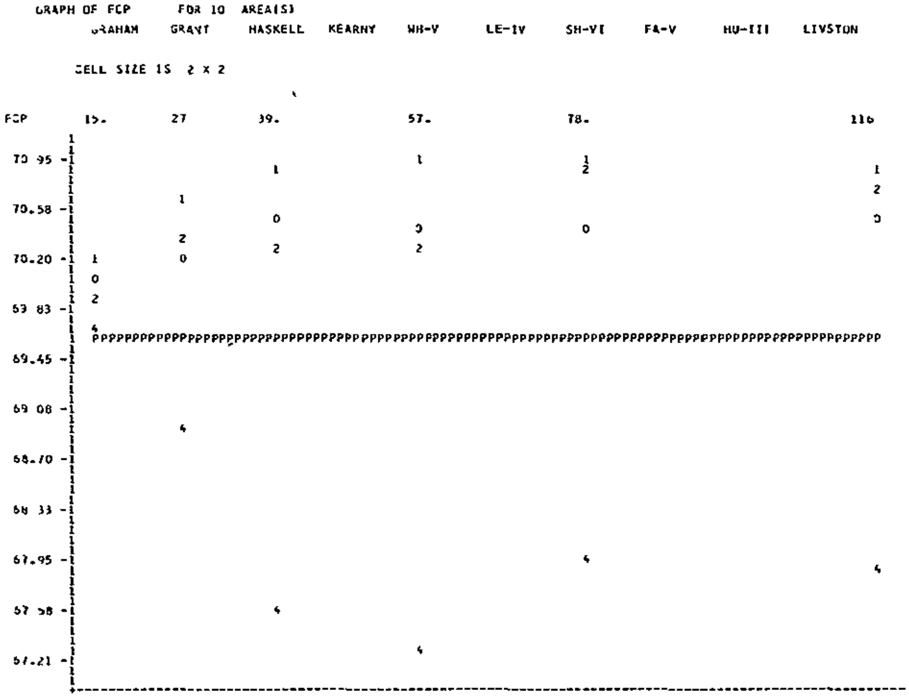


FIGURE 24

LANDSAT FIELD CENTER PIXEL PERFORMANCE

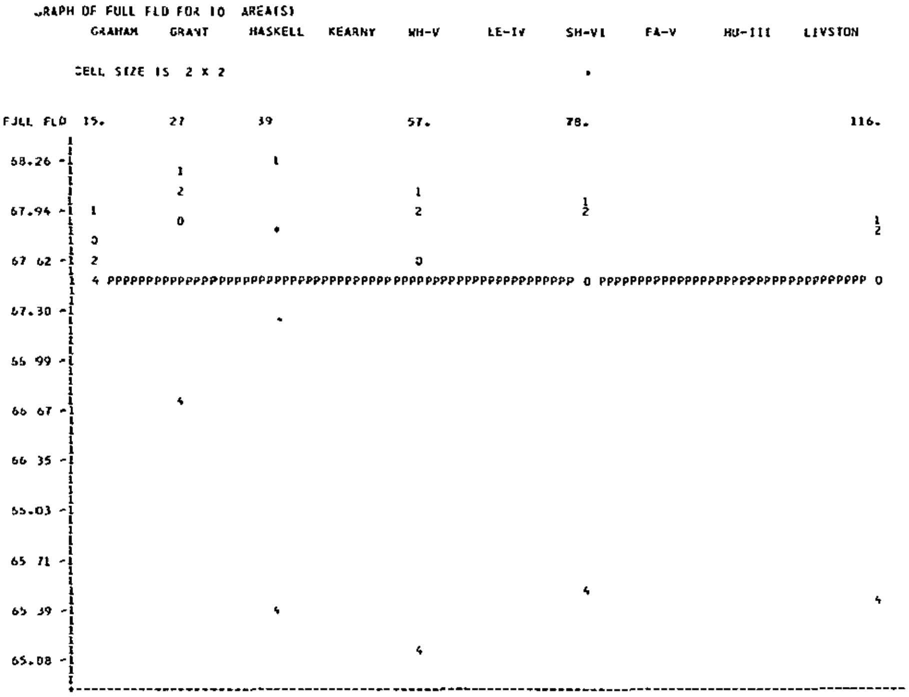


FIGURE 25

LANDSAT FULL FIELD PERFORMANCE

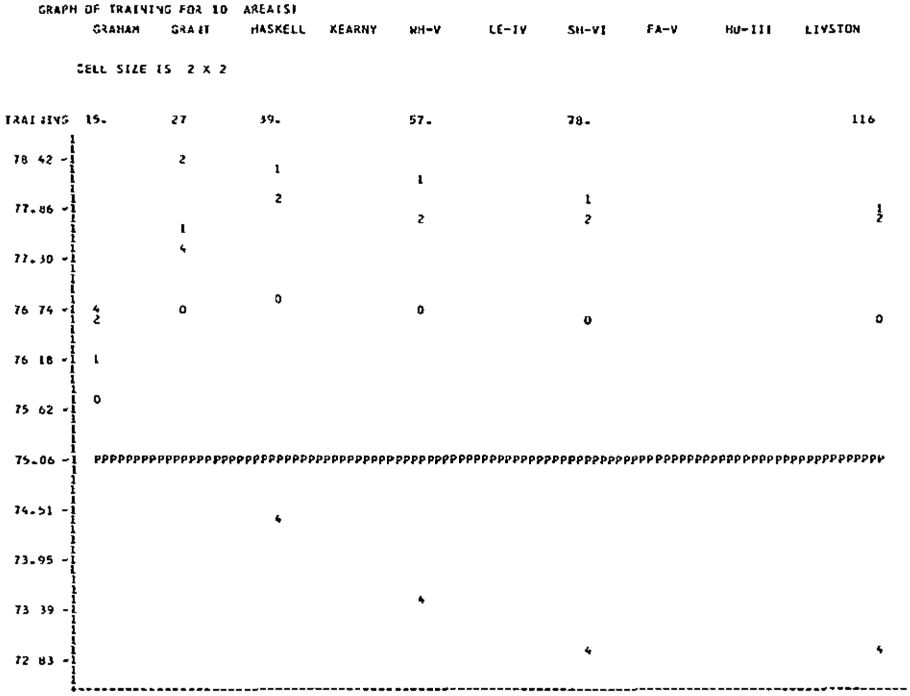


FIGURE 26

LANDSAT TRAINING FIELD PERFORMANCE

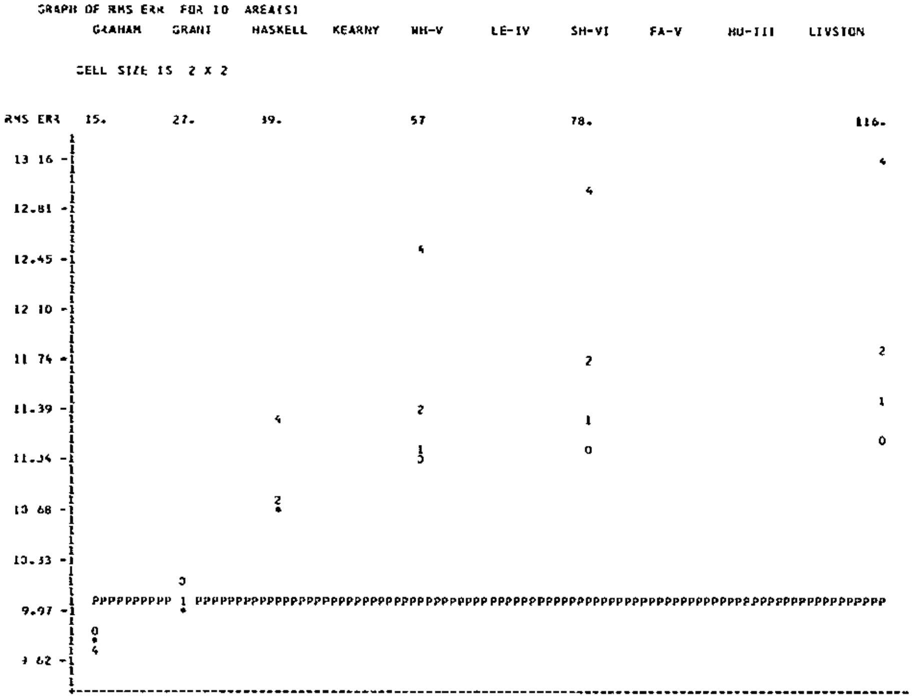


FIGURE 27

LANDSAT RMS PROPORTION ESTIMATE ERRORS

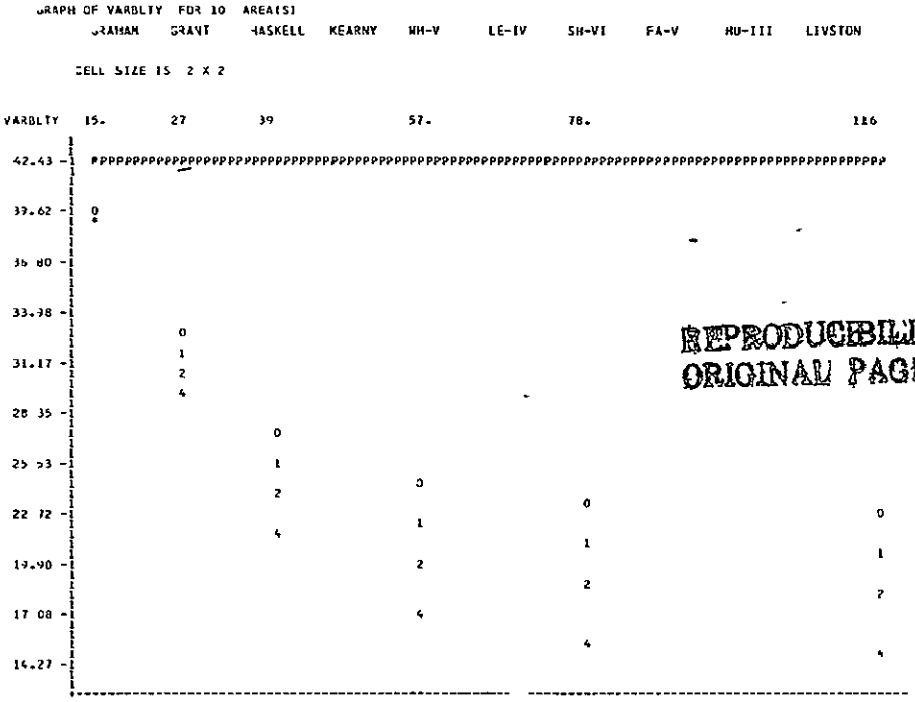


FIGURE 28

LANDSAT CLASSIFICATION VARIABILITY

* Since not all cell widths were sampled between two and the $\sqrt{\text{average field size}}/2 + 1$, it may be true that the optimal cell width for a data set lies between these two parameters. Indications from the Graham County analysis are that the optimal cell width may be in the neighborhood of the fourth root of the average field size.

The cell homogeneity threshold is significant for all dependent variables both when all LANDSAT data sets are considered individually and when all LANDSAT data sets are considered together.

The cell homogeneity threshold is significant for CPU time, "full field" performance, training performance, and classification variability for the combined CITARS data sets and is significant for CPU time, field center pixel performance, "full field" performance, training performance and classification variability in the combined LACIE data sets.

The statistically significant overall effects of the cell homogeneity parameter are:

- As the homogeneity parameter increases, CPU time and variability decrease.
- As the homogeneity parameter increases, field center pixel, "full field" and training performances decrease. Figures 24 through 26 indicate, however, that for the first two or three homogeneity threshold values, as the homogeneity threshold increases, the performances also increase.

As the homogeneity parameter increases, RMS error shows a trend towards increasing (Figure 27) though this trendis not significant at a 5% confidence level.

The annexation parameter is significant in 86% of the 60 cases in the LANDSAT data sets. It is significant only for CPU time and classification variability when the CITARS or the LACIE or all the LANDSAT data sets are considered together. This indicates that the proper annexation parameter is highly data dependent. As the annexation threshold increases, CPU time and classification variability tend to decrease.

The interaction of cell width and the cell homogeneity threshold is significant in 95% of the 60 cases of 6 dependent variables and 10 LANDSAT data sets. This interaction is significant for all dependent variables when the CITARS or the LACIE or all LANDSAT data sets are considered together.

The interaction of cell width and annexation parameter is significant in 93% of the 60 LANDSAT cases. When all the LANDSAT data sets are considered together, the cell width-annexation interaction is significant for all dependent variables except CPU time. This is also true for the CITARS data sets. However, for the LACIE data sets, the interaction is not significant for RMS proportion error.

The cell homogeneity-annexation interaction was significant in 30% of the 60 individual LANDSAT data set ANOVAs, being significant in 90% of the data sets for variability and 50% of the data sets for RMS proportion error. When the 10 LANDSAT data sets are considered together, the cell homogeneity-annexation interaction is significant for training performance and variability. When considering only the CITARS or only the LACIE data sets, the interaction is significant for variability alone.

ECHO-Perpoint LANDSAT Comparison

When the optimal parameter setting for each dependent variable was selected from the ECHO results and compared to the perpoint results for the 10 LANDSAT data sets the following results were obtained:

- ECHO was faster than the perpoint classifier (.1% confidence level).
- ECHO's field center pixel, "full field", and training field performances were superior to those of the perpoint classifier (1.3, 5.7, .1% confidence levels, respectively).
- ECHO had a lower RMS proportion estimate error (4.4% confidence level).
- ECHO had less classification variability than the perpoint classifier (.1% confidence level).

b. Simulated Thematic Mapper Results

The analyses of variance for the effects of the ECHO parameters on the eight simulated Thematic Mapper data sets (two sites, four resolutions) are presented in Table 2.1-10. Results for the eight data sets considered together are presented in Table 2.1-11.

Cell width is significant for all dependent variables and all Thematic Mapper data sets individually. When the Thematic Mapper data sets are considered together, using site as a block and resolution as a fixed factor, cell width is significant only for training performance and classification variability. As cell width increases, classification variability decreases. As cell width increases, training performance decreases. Trends which can be noted, but which are not statistically significant at the sample size used include: as cell width increases, CPU time, field center pixel performance, and "full field" performance decrease. It may be true that a cell width between two and the square root of the average field size divided by two would be superior to those cell sizes tested in terms of CPU time, RMS error, and field center pixel, "full field" and training field performances.

The cell homogeneity parameter is significant for all dependent variables in all Thematic Mapper data sets individually. When the data sets are considered together, the cell homogeneity parameter is significant for CPU time, "full field" performance, training performance and variability. As the cell homogeneity parameter increases, the CPU time and classification variability decreases (1% significance level). "Full field" and training field performance decrease over the range of homogeneity threshold tested (1% confidence level). The graph indicates that RMS proportion error increases as the cell homogeneity criteria increases. However, this is not a consistent trend in the eight data sets and is not statistically significant.

TABLE 2.1-10

EFFECT OF ECHO PARAMETERS ON INDIVIDUAL SIMULATED THEMATIC MAPPER DATA SETS

ABBREVIATIONS:

CPU	CENTRAL PROCESSING UNIT (COMPUTER) TIME
FCP	FIELD CENTER PIXEL PERFORMANCE
FULL	FULL FIELD PERFORMANCE
TRAIN	TRAINING FIELD PERFORMANCE
RMS	RMS PROPORTION ERROR
VAR	CLASSIFICATION VARIABILITY
CELW	CELL WIDTH
HOM	CELL HOMOGENEITY THRESHOLD
ANN	ANNEXATION THRESHOLD

				CELW	CELW	HOM
	CELW	HOM	ANN	X	X	X
				HOM	ANN	ANN
Williams County, North Dakota - 30 meter resolution						
CPU	.1	.1	99.9	.1	99.9	31.8
FCP	.1	.1	.1	.1	.1	20.5
FULL	.1	.1	.1	.1	.1	28.7
TRAIN	.1	.1	4.9	.1	2.3	99.9
RMS	.1	1.0	4.4	.1	4.1	44.7
VAR	.1	.1	.1	.1	.1	46.3
Williams County, North Dakota - 40 meter resolution						
CPU	.1	.1	99.9	1.6	99.9	99.9
FCP	.1	.1	.1	.1	.1	99.9
FULL	1.9	.1	8.9	.1	3.0	40.5
TRAIN	.1	.1	99.9	.1	99.9	99.9
RMS	.1	.1	99.9	.1	99.9	27.1
VAR	.1	.1	99.9	.5	99.9	46.0
Williams County, North Dakota - 50 meter resolution						
CPU	.1	.1	99.9	.1	99.9	99.9
FCP	.1	.1	99.9	.1	9.1	99.9
FULL	.1	.1	99.9	.1	36.3	99.9
TRAIN	.1	.1	5.6	.1	42.7	34.2
RMS	.1	.1	.1	.1	.8	.1
VAR	.1	.1	.1	.1	.1	15.5

TABLE 2.1-10 (Continued)

EFFECT OF ECHO PARAMETERS ON INDIVIDUAL SIMULATED THEMATIC MAPPER DATA SETS

				CELW	CELW	HOM
	CELW	HOM	ANN	X	X	X
				HOM	ANN	ANN
Williams County, North Dakota - 60 meter resolution						
CPU	.1	.1	99.9	.1	99.9	99.9
FCP	.1	.1	19.8	.1	8.7	99.9
FULL	.1	.1	2.1	.1	.7	99.9
TRAIN	.1	.1	14.6	.1	99.9	99.9
RMS	.3	.1	99.9	9.2	26.4	99.9
VAR	.1	.1	.1	.1	.1	11.6
Finney County, Kansas - 30 meter resolution						
CPU	.1	.1	39.7	.1	15.0	3.4
FCP	.1	.1	.1	.1	.1	29.4
FULL	.1	.1	.1	.1	.1	38.6
TRAIN	.1	.1	.1	.1	.1	99.9
RMS	.1	.1	16.7	.1	44.4	24.9
VAR	.1	.1	.1	.1	.1	37.5
Finney County, Kansas - 40 meter resolution						
CPU	.1	.1	10.4	.1	1.5	9.4
FCP	.1	.1	.1	.1	.1	46.2
FULL	.1	.1	.1	.1	.1	47.4
TRAIN	.1	.1	.1	.1	.1	99.9
RMS	.1	.1	7.8	.1	4.4	99.9
VAR	.1	.1	.1	.1	.1	43.4
Finney County, Kansas - 50 meter resolution						
CPU	.1	.1	99.9	.1	99.9	99.9
FCP	.1	.1	9.6	.1	2.7	99.9
FULL	.1	.1	.3	.1	17.3	29.7
TRAIN	.1	.1	7.5	.1	8.7	99.9
RMS	.1	.1	.1	.1	.1	2.3
VAR	.1	.1	.1	.1	.1	44.0
Finney County, Kansas - 60 meter resolution						
CPU	.1	.1	99.9	.1	99.9	99.9
FCP	.1	.1	.1	.1	.1	47.0
FULL	.1	.1	.7	.1	.1	99.9
TRAIN	.1	.1	.7	.1	.2	99.9
RMS	.1	.1	16.9	.1	13.4	47.3
VAR	.1	.1	.1	.1	.1	99.9

Table 2.1-11

EFFECT OF ECHO ON SIX VARIABLES USING SITE AS A BLOCK AND RESOLUTION AS A FIXED FACTOR

(Eight Thematic Mapper Data Sets)

	Cell Width	Cell Homogeneity Threshold	Annexation Threshold	Cell Width X Cell Homogeneity Threshold	Cell Width X Annexation Threshold	Cell Homogeneity Threshold X Annexation Threshold
Degrees of Freedom	2,2	5,5	3,3			
CPU Time	10	1		.1	99	99
Field Center Pixel Performance	25		25	.1	99	99
Full Field Performance	25	1	25	.1	99	99
Training Performance	5	1		.1	99	99
RMS Error		25	10	.1	99	99
Classification Variability	1	1	1	.1	.1	99

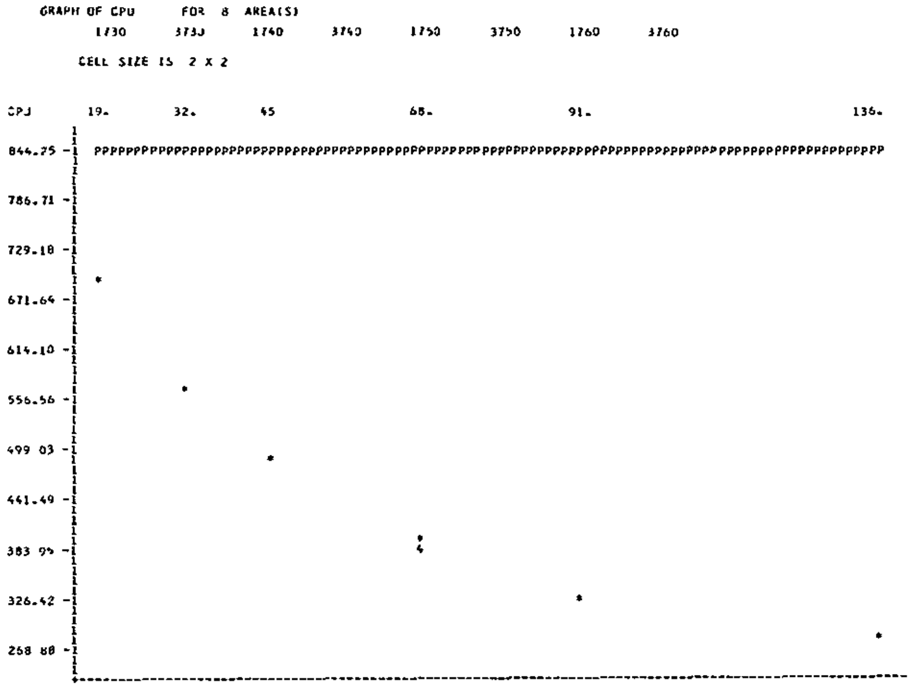


Figure 29
Simulated Thematic Mapper CPU Time

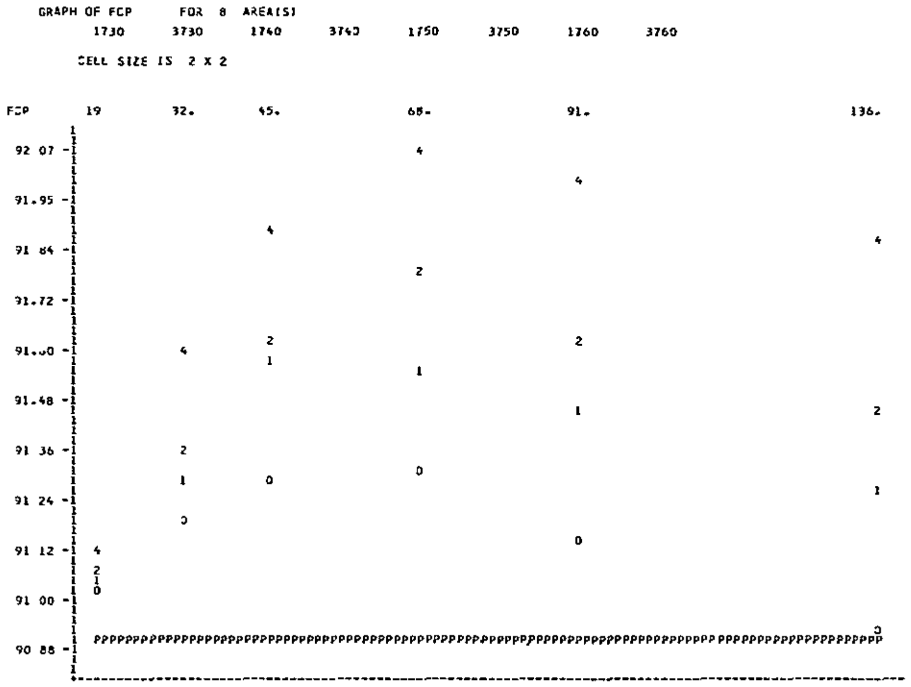


Figure 30
Simulated Thematic Mapper Field Center Pixel Performance

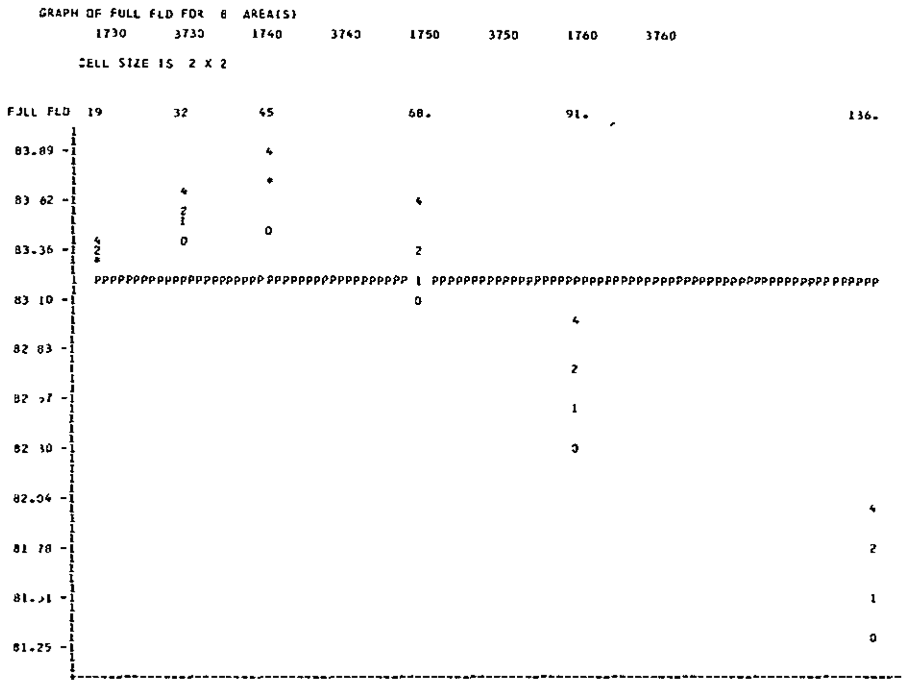


Figure 31
Simulated Thematic Mapper Full Field Performances

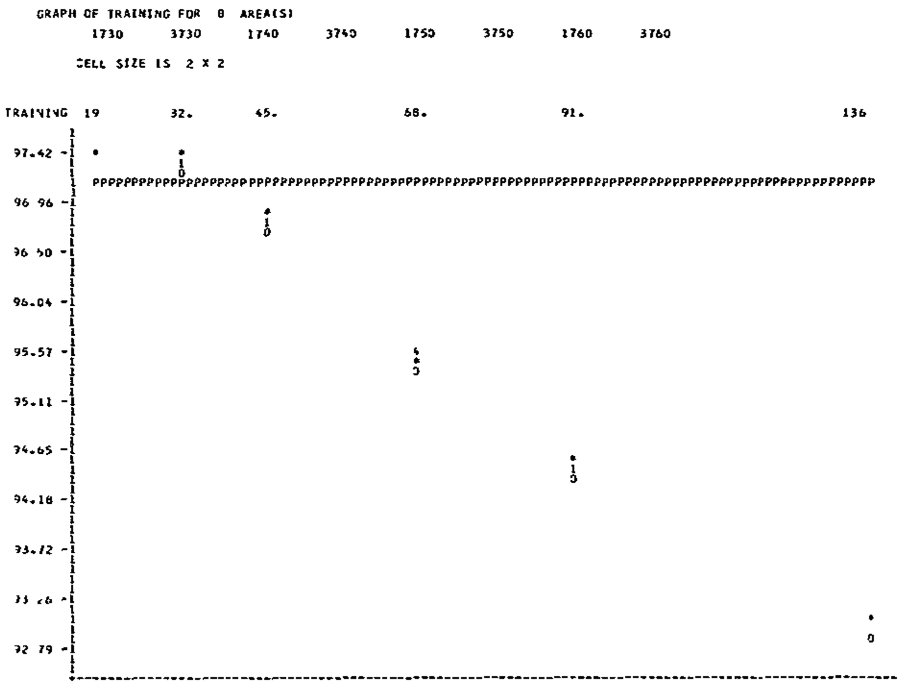


Figure 32
Simulated Thematic Mapper Training Performances

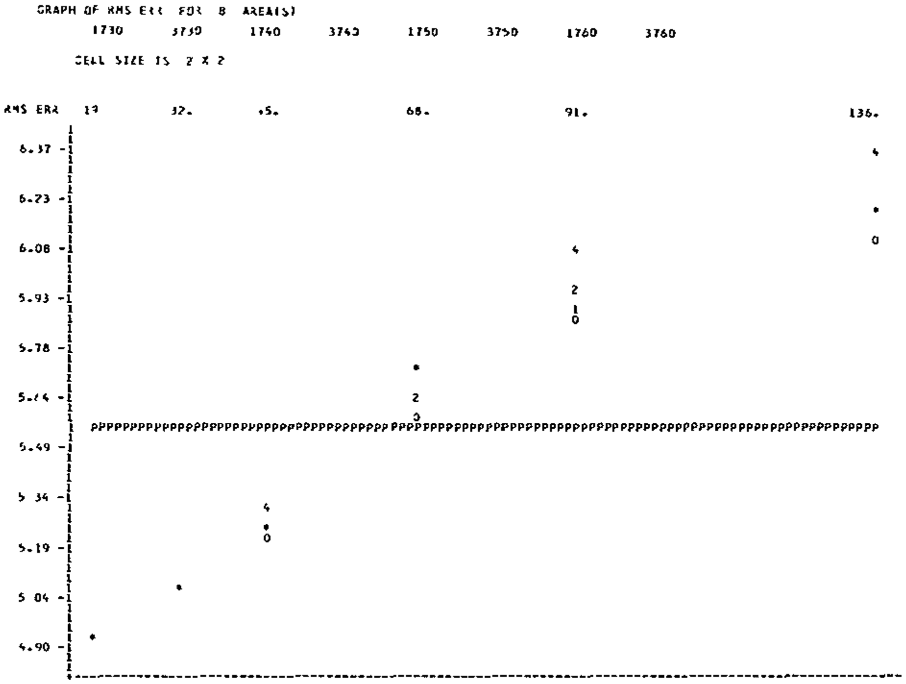


Figure 33
Simulated Thematic Mapper RMS Proportion Errors

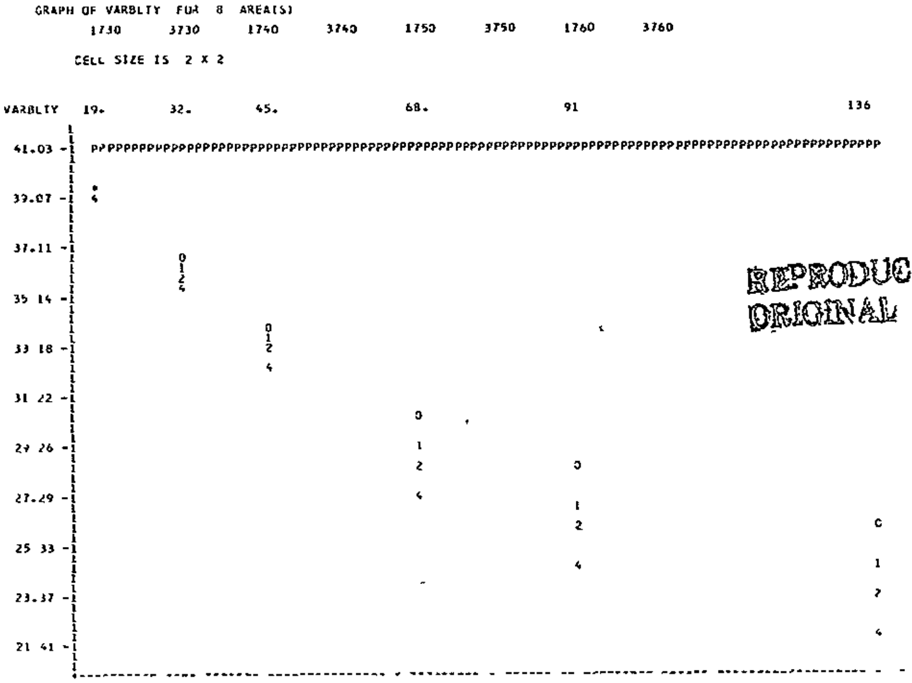


Figure 34
Simulat… Thematic Mapper Classification Variability

The annexation parameter is significant in 63% of the 48 cases involving 6 dependent variables and 2 sites and 4 resolutions of thematic mapper data. It is never significant for CPU time in the individual data sets. When the Thematic Mapper data sets are considered together, the annexation parameter is significant only for variability. As the annexation parameter increases, classification variability decreases.

The interaction of cell width and cell homogeneity is significant for all dependent variables both when the Thematic Mapper data sets are considered together or individually.

The interaction of cell width and annexation is also significant in 63% of the 48 cases in the Thematic Mapper data sets. It is significant only for variability when all the Thematic Mapper data sets are considered together.

The interaction of cell homogeneity and annexation is significant in 8% of the 48 cases of the individual Thematic Mapper data sets and is not significant for any dependent variable when the Thematic Mapper data sets are considered together.

Comparison of the ECHO and the Perpoint Classifier

F-values for the analysis of variance comparing the effects of classification method (ECHO versus perpoint) and resolution size (30, 40, 50 and 60 meters) to the six variables for the optimum parameter setting of the Supervised ECHO classifier for each of the dependent variables is presented in Table 2.1-12.

Table 2.1-12

	Method	Resolution
CPU Time	1204.39*	9.86**
Field Center Pixel Performance	147.63***	1.25
Full Field Performance	74.21***	1.47
Training Field Performance	15.21+	.62
RMS Proportion Error	2.62	.21
Classification Variability	13547.44**	7.23***
Degrees of Freedom	(1,1)	(3,3)

* significant at 1% confidence level

** significant at 5% confidence level

*** significant at a 10% confidence level

+ significant at a 25% confidence level

Classification variability is reduced as resolution is increased. This result is expected since the field which is 10 pixels wide in the 30 meter data is only 5 pixels wide in the 60 meter data. The CPU time required to perform the classification is increased as the resolution increases.

Results based on classification method are hampered by having only (1,1) degrees of freedom. ECHO requires less CPU time and has less classification variability than the perpoint classifier at a 1% confidence level.

ECHO field center pixel and "full field" performance appear to be superior to the perpoint classifier at a 10% confidence level.

More repetitions are necessary before it can be deomonstrated that ECHO RMS error is significantly less than that of the perpoint classifier for the Thematic Mapper data.

c. Aircraft Results

Tables 2.1-13 and 2.1-14 summarize the analysis of variance results for the effect of the ECHO parameters on the three aircraft data sets taken individually and taken as a block. Figures 35 through 39 present the CPU time, field center pixel performance, "full field" performance, training field performance, and classification variability. RMS values were not calculated because ground truth proportion estimates were not available for all three aircraft sites. The aircraft data sets were sampled only at the 2 by 2 cell size.

As the cell homogeneity parameter increases, field center pixel and training field performance increase (5% confidence levels). As with the LANDSAT and simulated Thematic Mapper data sets, classification variability decreases as annexation and cell selection threshold increase.

The effect of the annexation parameter increases as the cell homogeneity parameter increases for training field performance, classification variability and CPU time.

ECHO - Perpoint Comparison

Though only three aircraft data sets were tested, the Supervised ECHO processor produced results which had significantly less variability than the results produced by the perpoint classifier (5% confidence level). The comparison between the perpoint and the ECHO processor is approaching statistical significance for the other four variables. The following statements can be made for the aircraft data sets at a 15% confidence level.

* ECHO requires less computer time than the perpoint classifier.
* ECHO has higher field center pixel, "full field" and training performances than the perpoint classifier.

For the aircraft data sets, the cell homogeneity value appears to maximize performance for a value in the neighborhood of 70 and an annexation threshold of 4 seems to be optimal.

Table 2.1-13

INDIVIDUAL AIRCRAFT DATA SETS

(Entries are Significance Levels)

Finney County, North Dakota*

	Annexation Threshold	Cell Homogeneity Threshold
CPU Time	.2	.1
Field Center Pixel Performance	.1	.1
Full Field Performance	.1	.1
Training Field Performance	.1	.1
Classification Variability	.1	.1

Williams County, Kansas*

	Annexation Threshold	Cell Homogeneity Threshold
CPU Time	11.2	.1
Field Center Pixel Performance	.2	.1
Full Field Performance	.1	.1
Training Field Performance	.5	.1
Classification Variability	.1	.1

Tippecanoe County, Indiana*

	Annexation Threshold	Cell Homogeneity Threshold
CPU Time	.5	.1
Field Center Pixel Performance	2.4	4.0
Full Field Performance	1.7	21.5
Training Field Performance	.1	.1
Classification Variability	.1	.1

* Degrees of freedom for the Annexation Threshold test are 3,15 and Degrees of freedom for the Cell Homogeneity Threshold are 5,15.

Table 2.1-14

RESULTS FOR ECHO CLASSIFICATIONS OF

AIRCRAFT DATA, USING ALL THREE DATA SETS

	Cell Homogeneity	Annexation Threshold	Interaction
Degrees of Freedom	5,10	3,6	15,30
Field Center Pixel Performance	3.92***	2.516^{o}	1.609^{o}
Full Field Performance	0.92	2.88	1.75***
Training Field Performance	6.84*	4.64***	5.47*
Classification Variability	18.80*	63.33*	21.78*
CPU Time	1.81^{o}	1.78^{o}	3.20*

* sig 1%

** sig 5%

*** sig 10% ← Table entries are F values. Sig. level indicated by symbols.

o sig 25%

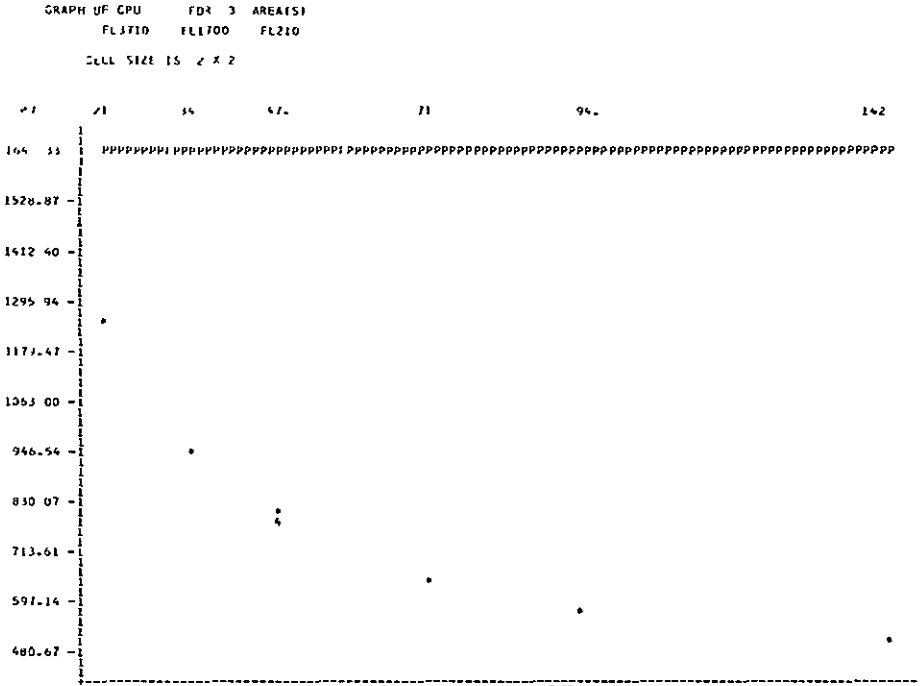


Figure 35
Aircraft CPU time

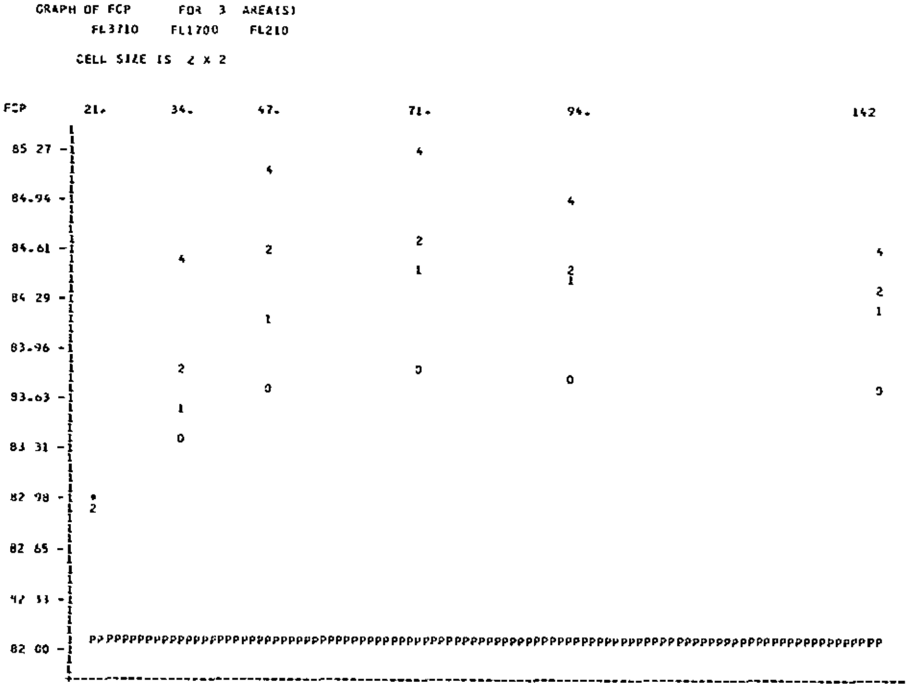


Figure 36
Aircraft Field Center Pixel Performance

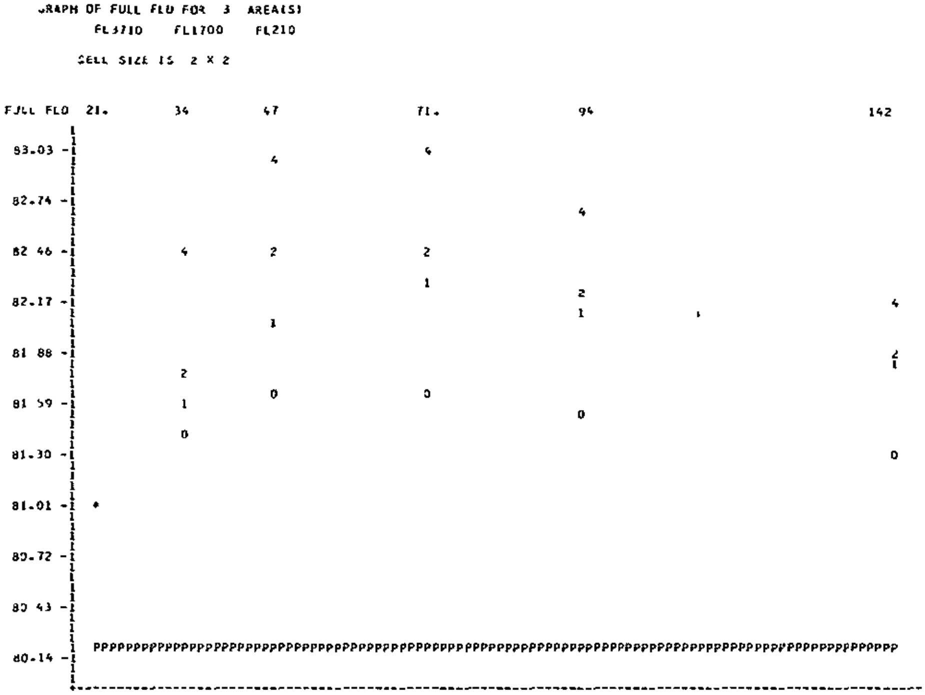


Figure 37
Aircraft Full Field Performance

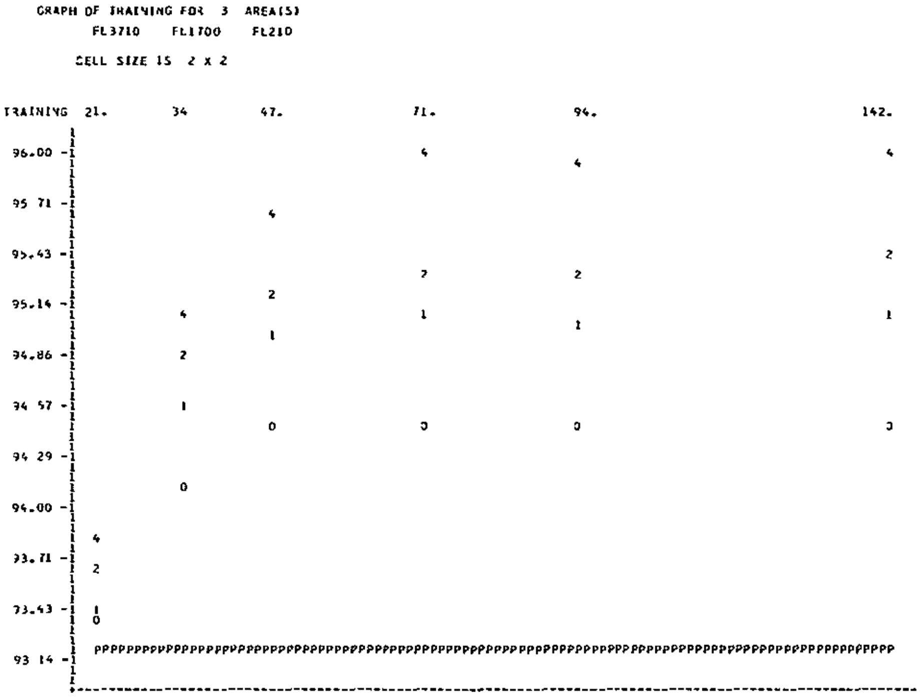


Figure 38
Aircraft Training Field Performance

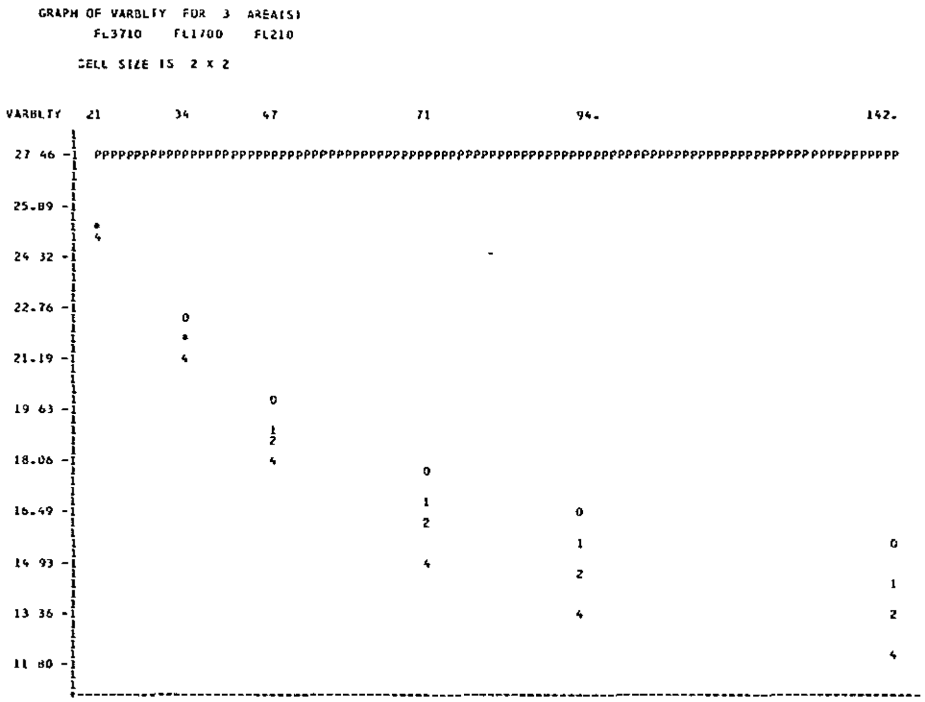


Figure 39
Aircraft Classification Variability

C. Products for LACIE/AI Evaluation

Magnetic computer tapes containing examples of object maps produced by the Nonsupervised field extraction algorithm are provided in Universal format. Object map tapes are multispectral image storage tapes which have had the data values altered for pixels falling within objects identified by the algorithm. The mean response for the pixels comprising the object replaces the individual response for each constituent pixel of the object.

SUMMARY

The ECHO processor developed at LARS has been implemented in documented FORTRAN programs.

The Supervised ECHO processor has been tested over LANDSAT, simulated Thematic Mapper and aircraft data sets. Evaluations of those tests indicate that:

* ECHO is less expensive than the perpoint classification algorithm (statistically significant for LANDSAT and simulated Thematic Mapper data sets at a 1% confidence level and at a 15% confidence level for the aircraft data sets).

* ECHO provides better field center pixel and "full field" classification performance than the perpoint classifier (5% confidence level on LANDSAT data, 10% confidence level on simulated Thematic Mapper data, 15% confidence level on aircraft data).

* ECHO RMS proportion error is significantly less than that of the perpoint classifier at optimal parameter settings for the LANDSAT data. Proportion estimation for the Thematic Mapper data set was not significantly different for the two classification methods.

* ECHO classification results are less variable than the classification results of the perpoint classifier. This result is statistically significant at a 5% confidence level for all data sets.

The effects of the three ECHO parameters on six variables were monitored. In general, cell homogeneity had the strongest effect on the six variables. Cell width had a strong effect on the variables when the data sets were considered individually. However, when the data sets were considered together, the effect of cell width was not significant for RMS error and was less significant than the cell homogeneity parameter for field center pixel and "full field" performances, indicating the effects of cell width are sometimes opposite in direction for differing data sets. The annexation threshold had a significant effect on classification variability and CPU time but a weak effect on the other variables.

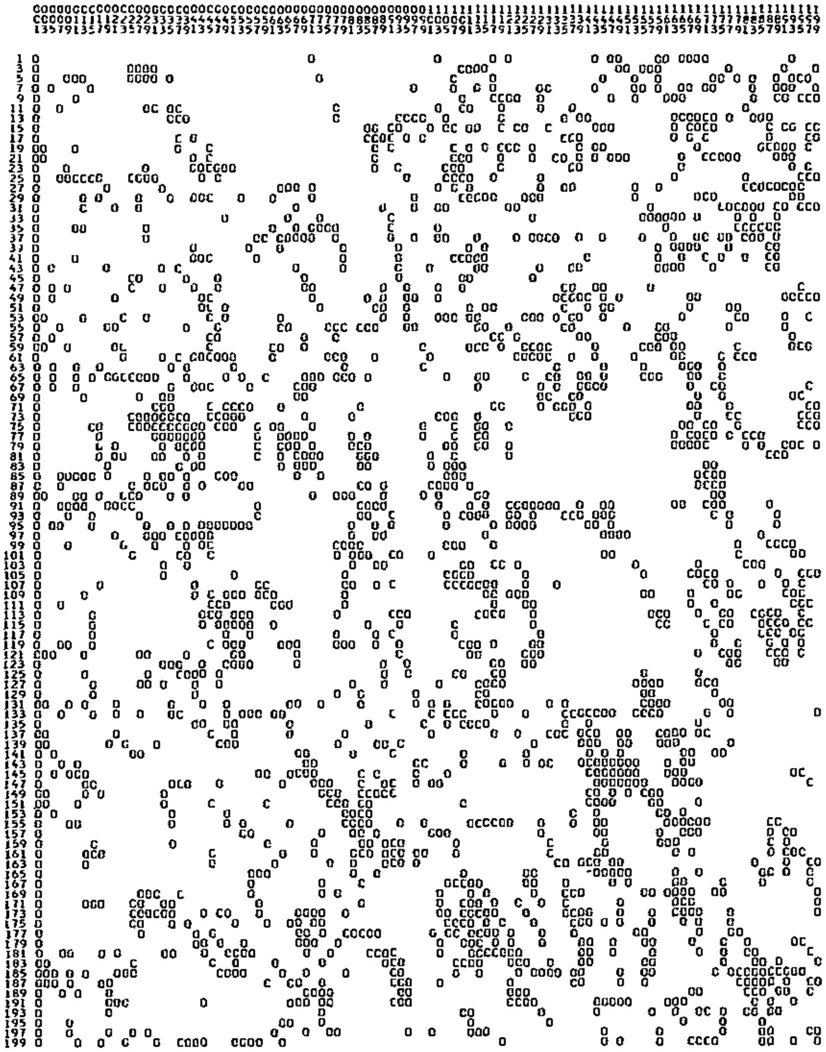
CONCLUSIONS

ECHO successfully exploits the redundancy of states characteristics of sampled imagery of ground scenes to achieve better classification accuracy, reduce the number of classifications required, and reduce the variability of the classification results. The information required to produce ECHO classifications are cell size, cell homogeneity, and cell-to-field annexation parameters, input data, and a class-conditional marginal density statistics deck.

Future research should be directed towards developing methods for utilizing the information produced by the ECHO processors to aid in the training process. For example, the singular cell map produced by the Supervised ECHO processor (see Figure 40) provides information on training adequacy. Singular cells are represented on this map by 0's. For the Supervised processor, cells are identified as being singular when the likelihood of the cell belonging to the most likely of the available classes falls below a threshold. Therefore, when a cell is categorized as "singular", it is either because it contains pixels from more than one class, or because the spectral class of the field is not represented in the available class statistics. Large groups of contiguous singular cells will occur when one or more spectral classes have been omitted. The singular cell map may indicate where additional training statistics should be collected.

The intermediate tape produced by the Nonsupervised ECHO processor contains the statistics for each homogeneous object identified by the field extraction routine. It is possible that this spectral-spatial cluster will contain all the information an analyst will need to produce a class statistics deck superior to what would be produced by traditional training procedures for a smaller investment of analyst time than is now required.

```
ECHOP1                         LABORATORY FOR APPLICATIONS OF REMOTE SENSING                 MAY   9,1977
PHIL ALENDUFF                              PURCUE UNIVERSITY                                 C3 5C 14 PM
                                                                                             LARSYS VERSION 3

                    RUN NUMBER.. ...... . 740285C0          DATE DATA TAKEN .. JULY 19,1974
                    FLIGHT LINE. . 1726-16425   KAN         TIME DATA TAKEN.   .      HOURS
                    DATA TAPE/FILE NUMBER.  2689/ 3         PLATFORM ALTITUDE. 3C62000 FEET
                    REFORMATTING DATE  OCT   7,1975         GROUND HEADING ....    0 DEGREES

                                             CHANNELS USED
                                            ---------------

            CHANNEL 10    SPECTRAL BAND  0.60 TO  0.70 MICROMETERS    CALIBRATION CODE = 1    CO = 0.C
            CHANNEL 11    SPECTRAL BAND  0.70 TO  0 8C MICROMETERS    CALIBRATION CODE = 1    CO = 0 0
            CHANNEL 12    SPECTRAL BAND  0.80 TO  1.10 MICROMETERS    CALIBRATION CODE = 1    CO = C.C
AREA PROCESSED.....LINES       1-  200  (BY  1)
                   CCLUMNS     1-  200  (BY  1)
```



```
EOF RECORD WRITTEN.

IDIO3  CPU TIME USED WAS   94.625 SECONDS.      (LARSMN)
```

Figure 40
Example of the Supervised ECHO Singular Cell Map

Lastly, background research should be performed to determine what training techniques are optimal for a field classifier. ECHO performance may be enhanced by utilizing training techniques developed specifically for it, rather than techniques which are optimal for the perpoint classifier.

References

1. Kettig, R. L. 1975.

   Computer Classification of Remotely Sensed Multispectral Image Data by Extraction and Classification of Homogeneous Objects.

   Ph.D. Thesis, School of Electrical Engineering, Purdue University, West Lafayette, Indiana.

2. Davis, W. A. and Peet, F. G. 1977.

   A Method of Smoothing Digital Thematic Maps.

   Remote Sensing of Environment, Volume 6, pp. 45-49

3. Landgrebe, D. A. 1975.

   Final Report, NASA Contract NAS9-14016, June 1, 1974-May 31, 1975.

   pp. X-1 - X-11.

4. Kettig, R. L. and Landgrebe, D. A. 1973.

   Automatic Boundary Finding and Sample Classification of Remotely Sensed Multispectral Data.

   LARS Information Note 041773, Laboratory for Applications of Remote Sensing, Purdue University, West Lafayette, Indiana.

5. Kettig, R. L. and Landgrebe, D. A. 1975.

   Classification of Multispectral Image Data by Extraction and Classification of Homogeneous Objects.

   Proceedings Symposium on Machine Processing of Remotely Sensed Data Purdue University, IEEE Catalog No. 75CH1009-0-C.

6. Rodd, E. M. 1972.

   Closed Boundary Field Selection in Multispectral Digital Images.

   IBM Publication No. 320.2420.

7. Gupta, J. N. and Wintz, P. A. 1973.

   Closed Boundary Finding, Feature Selection, and Classification Approach to Multi-Image Modeling.

LARS Information Note 062773, Laboratory for Applications of Remote Sensing, Purdue University, West Lafayette, Indiana.

8. Robertson, T. V. 1973.

Extraction and Classification of Objects in Multispectral Images. Proceedings of the Conference on Machine Processing of Remotely Sensed Data, Purdue University, West Lafayette, Indiana, Section 3B, pp. 27-34.

9. Bauer, M. E., Cary, T. K., Davis, B. J., and Swain, P. H. 1975.

Crop Identification Technology Assessment for Remote Sensing (CITARS), Volume VI: Data Processing at the Laboratory for Applications of Remote Sensing, Purdue University, West Lafayette, Indiana. National Aeronautics and Space Administration, Houston, Texas. JSC 09389.

APPENDIX A

TASK 2.1

PROGRAM ABSTRACTS AND LISTINGS FOR THE SUPERVISED ECHO PROCESSOR

LARS Program Abstract ______

MODULE IDENTIFICATION

Module Name: SECSUP Function Name: SECSUP

Purpose: Supervised ECHO supervisor routine.

System/Language: CMS/FORTRAN

Author: P. D. Alenduff Date: 03/30/77

Latest Revisor: ______ Date: ______

MODULE ABSTRACT

SECSUP is the supervisor routine for the supervised ECHO classifier.

PURDUE UNIVERSITY
Laboratory for Applications of Remote Sensing
1220 Potter Drive
West Lafayette, Indiana 47906

1. Module Usage

SECSUP

Calling sequence:

CALL SECSUP

SECSUP is the supervisor routine for the supervised ECHO classifier. It is called with no arguments.

2. Internal Description

The program SECSUP basically controls the program sequencing for the many subprograms involved in the supervised ECHO classifier. Two subroutines are called by SECSUP: SECRDR and SECINT.

SECRDR is the control card reader for reading the user input data cards and is called first. SECINT is the initialization routine for the SECHO function and is called to handle utilization of all input/output operations as well as classification processing. For more detailed descriptions of these modules, see their program abstracts.

Two common blocks are used in SECSUP. GLOCOM, the main LARSYS common block and SECCOM, the common block for the supervised ECHO function are included.

3. Input Description

Not applicable

4. Output Description

Two messages are produced and written to unit TYPEWR, the console.

```
SUPERVISED ECHO FUNCTION REQUESTED
```

Signifies beginning of the function.

SUPERVISED ECHO FUNCTION COMPLETED

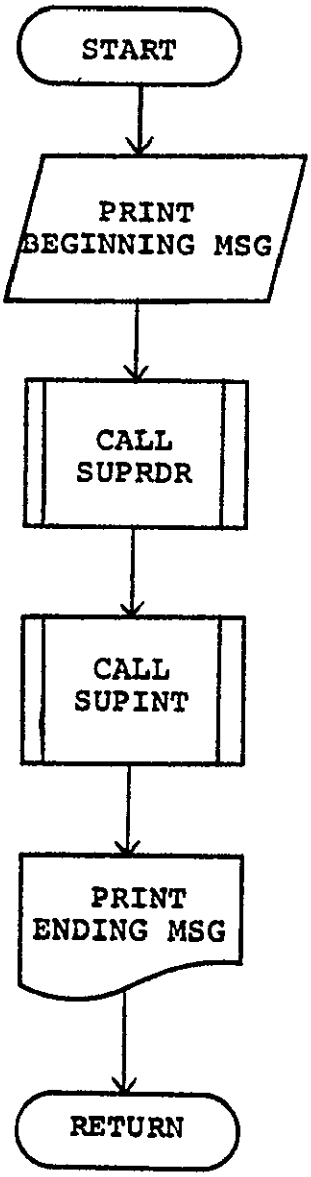
Signifies end of the function.

5. Supplemental Information

Two COMMON blocks are included in SECSUP:

```
GLOCOM
SECCOM
```

6. Flowchart


START
PRINT
BEGINNING MSG
CALL
SUPRDR
CALL
SUPINT
PRINT
ENDING MSG
RETURN

```
FILE SECSUP
                C      SECSUP  LARS    XXXX                                          SEC00010
                C                                                                    SEC00020
                C********************************************************************SEC00030
                C                                                                    SEC00040
                C   SECSUP - SUPERVISOR ROUTINE FOR SUPERVISED ECHO FUNCTION         SEC00050
                C            WRITTEN 01/20/77 BY P.D. ALENDUFF                       SEC00060
                C                                                                    SEC00070
                C********************************************************************SEC00080
                C                                                                    SEC00090
 0001                 SUBROUTINE SECSUP                                              SEC00100
                C                                                                    SEC00110
 0002                 IMPLICIT INTEGER * 4 (A-Z)                                     SEC00120
                C                                                                    SEC00130
 0003                 COMMON /GLOCOM/ BLANK, CARD(20), CHKOUT, COPFIL, CLASSR, CLASSX,  SEC00140
                     1    CLUSTX, CONPUT, CPYOUT, CRDRDR, CRDSEQ, DATAPE,            SEC00150
                     2    DUPLTP, DUPRUN, ERRMSG, FBPNT,                             SEC00160
                     3    FILESV, FLDBND, HDATA, HEAD(88), ID(200), IMAGEX,          SEC00170
                     4    IMARK, KEYBD, MAPTAP, MAXCHA, MAXCLS,                      SEC00180
                     5    PAGSIZ, PNCH, POINT, PRESUX, PRNTR, READIN,                SEC00190
                     6    RESTRT, RUNFIL, RUNTAB(10,3),                              SEC00200
                     7    SDATA, SEPARX, SEPTPX, SPARE(10), TEMPAS(30),              SEC00210
                     8    TPSTAT(6), TTFLDX ,TYPEWR,                                 SEC00220
                     9    TOP, ARRAY(12500)                                          SEC00230
                C                                                                    SEC00240
 0004                 REAL * 8 ARRAY                                                 SEC00250
 0005                 REAL * 4 FRQCAL(5,30)                                          SEC00260
 0006                 INTEGER * 4 COMENT(16), DATE(5), HED1(16), HED2(16), TIME(5)   SEC00270
 0007                 INTEGER * 2 BLANK2                                             SEC00280
 0008                 LOGICAL * 4 CHKOUT                                             SEC00290
 0009                 LOGICAL * 1 BLANK1                                             SEC00300
 0010                 EQUIVALENCE (DATSAV,ID(1)), (CURRUN,ID(3)), (FRQCAL(1),ID(51)),  SEC00310
                     1        (HED1(1),HEAD(8)), (DATE(1),HEAD(26)), (HED2(1),HEAD(39)),SEC00320
                     2        (TIME(1),HEAD(58)), (COMENT(1),HEAD(72)),              SEC00330
                     3        (MAPSAV,TPSTAT(1)),                                    SEC00340
                     4        (SEPSCR,TPSTAT(2)), (DUPIN,TPSTAT(3)), (DASTAT,TPSTAT(4)),SEC00350
                     5        (COPSER,TPSTAT(5)), (TRAOUT,TPSTAT(6)),                SEC00360
                     6        (BLANK,BLANK2,BLANK1)                                  SEC00370
                C                                                                    SEC00380
                C                                                                    SEC00390
 0011                 COMMON /SECCOM/ BUFROZ,CSELEC,CSET(3,30),CSET3(3,30),THRES,CELSIZ,SEC00400
                     1 CELWTH,INFO(17),INPUT, INTTAP, INTFIL, JPTS, LINES, NOCLS, NOFET,SEC00410
                     2 NOFET3,NOPOOL,NWORD,NWORD2, NVR, OUTPUT, POLNAM(2,60), PREFIX(2),SEC00420
                     3 PTS, RQTAPE, RQFILE, SYMCNT, VARSZ3, CSEL(30), CSEL3(30),     SEC00430
                     3 FETVEC(30), FETVC3(30), POLPTR(2,60), POLSTK(60),             SEC00440
                     4 CDFLAG, CLSMAP, CSET1(3,30), OBJMAP, PHASE1, PHASE2, POLNM1(60), SEC00450
                     5 PRSTAT, SYM(60), SYMMTX(60)                                   SEC00460
                C                                                                    SEC00470
 0012                 REAL * 4 CSELEC, CSET, CSET3, THRES                            SEC00480
                C                                                                    SEC00490
 0013                 INTEGER * 2 CSEL, CSEL3, FETVEC, FETVC3, POLPTR, POLSTK        SEC00500
                C                                                                    SEC00510
 0014                 LOGICAL * 1 CDFLAG, CLSMAP, CSET1, OBJMAP, PHASE1, PHASE2, POLNM1,SEC00520
                     1 PRSTAT                                                        SEC00530
                C                                                                    SEC00540
                C                                                                    SEC00550
                C     CALL READER AFTER INITIAL MESSAGE PRODUCED                     SEC00560
                C                                                                    SEC00570
 0015                 WRITE (TYPEWR,100)                                             SEC00580
 0016             100 FORMAT(' I0000  SUPERVISED ECHO FUNCTION REQUESTED    (SECSUP)')  SEC00590
 0017                 CALL SECRDR                                                    SEC00600
                C                                                                    SEC00610
                C CALL PROCESSING ROUTINE THEN PRINT TERMINATION MSG                 SEC00620
                C                                                                    SEC00630
 0018                 CALL SECINT                                                    SEC00640
                C                                                                    SEC00650
 0019                 WRITE(TYPEWR,9000)                                             SEC00660
 0020            9000 FORMAT(' I0000  SUPERVISED ECHO FUNCTION COMPLETED   (SECSUP)')  SEC00670
 0021                 RETURN                                                         SEC00680
 0022                 END                                                            SEC00690
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

LARS Program Abstract

MODULE IDENTIFICATION

Module Name: SECCOM   Function Name: SECSUP

Purpose: Common block for supervised ECHO function

System/Language: CMS/FORTRAN

Author: P. D. Alenduff   Date: 03/30/77

Latest Revisor: Date:

MODULE ABSTRACT

SECCOM is the common block for the supervised ECHO classifier.

SECCOM

Variable Description

BUFROZ	INTEGER*4 Number of rows of cells in buffer.
CSELEC	REAL*4 Cell selection parameter.
CSET (3,30)	REAL*4 Channel calibration code vector from statistics.
CSET3 (3,30)	REAL*4 Channel calibration code vector from channels card.
THRES	REAL*4 Annexation threshold.
CELSIZ	INTEGER*4 Number of data points in a cell.
CELWTH	INTEGER*4 Number of rows in cell.
INFO (16)	INTEGER*4 Array for storing field description information and areas to be classified.
INPUT	INTEGER*4 Unit number for input tape.
INTTAP	INTEGER*4 Tape number of intermediate tape.
INTFIL	INTEGER*4 File number of intermediate tape.
JPTS	INTEGER*4 Number of cells horizontally in data area.
LINES	INTEGER*4 Number of rows of cells in area to be processed.
NOCLS	INTEGER*4 Number of classes in statistics deck.

NOFET	INTEGER*4 Number of channels in statistics deck.
NOFET3	INTEGER*4 Number of channels to be used in classification.
NOPOOL	INTEGER*4 Number of pools to be used in classification.
NWORD	INTEGER*4 Integer array word counter.
NWORD2	INTEGER*4 Integer word counter.
NVR	INTEGER*4 Number of channels on data tape.
OUTPUT	INTEGER*4 Unit number of output file.
POLNAM (2,60)	INTEGER*4 Names of statistics classes.
PREFIX(2)	INTEGER*4 Prefix information used in writing results.
RQTAPE	INTEGER*4 Tape number of the results tape.
RQFILE	INTEGER*4 File number of the results file.
SYMCNT	INTEGER*4 Number of user supplied symbols.
VARS23	INTEGER*4 Number of words to store half of a triangular covariance matrix for classes (channels * (channels + 1))/2.
CSEL(30)	INTEGER*2 Channel select vector from statistics.

CSEL3(30)	INTEGER*2 Channel select vector from channels card.
FETVEC(30)	INTEGER*2 Array of channel numbers from statistics deck.
FETVC3(30)	INTEGER*2 Array of channel numbers from channels card.
POLPTR(2,60)	INTEGER*2 Array of classification pools and pooling information.
POLSTK(60)	INTEGER*2 Array of stacked class numbers organized by pool request.
CDFLAG	LOGICAL*1 Logical variable signalling input of statistics on cards in control card deck.
CLSMAP	LOGICAL*1 Logical variable requesting production of classification map.
CSET1(3,30)	LOGICAL*1 Logical array indicating storage of calibration values by user.
OBJMAP	LOGICAL*1 Logical variable requesting production of singular cell map in phase 2 (annexation).
PHASE1	LOGICAL*1 Logical variable requesting initial cell processing is to be carried out.
PHASE2	LOGICAL*1 Logical variable indicating annexation of cells is to be carried out.
POLNM1(60)	LOGICAL*1 Logical array signalling that the name for POOL (I) has been stored.

PRSTAT	LOGICAL*1 Logical variable indicating request to print statistics information.
SYM(60)	LOGICAL*1 Array of user supplied symbols.
SYMMTX(60)	LOGICAL*1 Array of default symbols.

```
FILE SECCOM
                C     SECCOM   LARS   XXXX                                      SUP00010
                C                                                               SUP00020
                C***********************************************************************SUP00030
                C                                                               SUP00040
                C SECCOM - COMMON BLOCK FOR SUPERVISED ECHO                     SUP00050
                C         WRITTEN 11/22/76 BY P.D. ALENDUFF                     SUP00060
                C                                                               SUP00070
                C***********************************************************************SUP00080
                C                                                               SUP00090
                C VARIABLES USED IN SECCOM - COMMON BLOCK FOR SUPERVISED ECHO    SUP00100
                C                                                               SUP00110
  0001                BLOCK DATA                                                SUP00120
                C                                                               SUP00130
  0002                IMPLICIT INTEGER * 4 (A-Z)                                SUP00140
                C                                                               SUP00150
  0003                COMMON /SECCOM/ BUFROZ,CSELEC,CSET(3,30),CSET3(3,30),THRES,CELSIZ,SUP00160
                     1 CELWTH,INFO(17),INPUT, INTTAP, INTFIL, JPTS, LINES, NOCLS, NOFET,SUP00170
                     2 NOFET3,NOPOOL,NWORD,NWORD2, NVR, OUTPUT, POLNAM(2,60), PREFIX(2),SUP00180
                     3 PTS, RQTAPE, RQFILE, SYMCNT, VARSZ3, CSEL(30), CSEL3(30),        SUP00190
                     3 FETVEC(30), FETVC3(30), POLPTR(2,60), POLSTK(60),                SUP00200
                     4 CDFLAG, CLSMAP, CSET1(3,30), OBJMAP, PHASE1, PHASE2, POLNM1(60), SUP00210
                     5 PRSTAT, SYM(60), SYMMTX(60)                                      SUP00220
                C                                                               SUP00230
  0004                REAL * 4 CSELEC, CSET, CSET3, THRES                       SUP00240
                C                                                               SUP00250
  0005                INTEGER * 2 CSEL, CSEL3, FETVEC, FETVC3, POLPTR, POLSTK   SUP00260
                C                                                               SUP00270
  0006                LOGICAL * 1 CDFLAG, CLSMAP, CSET1, OBJMAP, PHASE1, PHASE2, POLNM1,SUP00280
                     1 PRSTAT                                                           SUP00290
                C                                                               SUP00300
                C                                                               SUP00310
  0007                DATA SYMMTX/'1','2','3','4','5','6','7','8','9','A','B',          SUP00320
                     1'C','D','E','F','G','H','I','J','K','L','M','N',                  SUP00330
                     2'O','P','Q','R','S','T','U','V','W','X','Y','Z',                  SUP00340
                     3'0','+','=','*','$','/','&','(',')',                              SUP00350
                     4'1','2','3','4','5','6','7','8','9','A','B','C',                  SUP00360
                     5'D','E','F','G'/                                                  SUP00370
                C                                                               SUP00380
  0008                END                                                       SUP00390
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

LARS Program Abstract ________

MODULE IDENTIFICATION

Module Name: SECRDR Function Name: SECSUP

Purpose: Control card reader for Supervised ECHO Function

System/Language: CMS/FORTRAN

Author: P. D. Alenduff Date: 3/30/77

Latest Revisor: ________ Date: ________

MODULE ABSTRACT

SECRDR is the control card reader for the Supervised ECHO function. After user requests are read, many different checks are made to detect any control card errors. The input or output classification or intermediate tapes are then mounted on the appropriate devices, and an options list is printed to document user selections.

1. Module Usage

SECRDR

Calling sequence:

CALL SECRDR

SECRDR is called without parameters to read user option selections and detect any errors on specified combinations of options. Upon error detection, either corrected data is requested or the function is terminated following an information message detailing the error. Output and/or intermediate tapes and the multispectral image storage tape are mounted and a list of selected options is produced.

2. Internal Description

SECRDR first performs initialization on all variables stored in SECCOM common block to set up any default options and clear any previously selected options. Control card decoding is heavily dependent upon LARSYS control card processing routines CTLWRD and BCDVAL. CTLWRD is called with a list of acceptable keywords to read input control cards. The number of the matching keyword is returned to SECRDR and a branch is made to the code that will appropriately decode the remainder of the control card. Other special purpose LARSYS routines used for control card decoding include CHANEL for decoding the CHANNELS card, and POLSCN for decoding the CLASSES card.

After encountering a DATA or END card, a check is made for a request of at least one channel on the CHANNELS card. The cell width parameter is then checked to be greater than or equal to 2. The cell selection option must also be non-negative, and annexation threshold THRES must be zero or positive. If both an object map and classification map are requested, only an object map is produced. At this point, tape and disk parameters are checked. If OPTIONS INTERMEDIATE (start processing with the intermediate tape) is selected, a tape number must be supplied. If an intermediate tape number is supplied, a file number must also be specified.

Upon specification of OPTIONS INTERMEDIATE, processing begins with the intermediate tape and a results file is produced. If, however, no OPTIONS INTERMEDIATE request is made and both an intermediate tape number and a results location is specified, a conflict is recognized and EXECUTION is terminated. If a results location is needed and none is specified, more data is requested. A file number must be specified when a tape location is selected. Specifications of both tape and disk options for classification results information causes termination.

After the error checking sequence is finished, tapes are mounted on the appropriate devices by calls to MTAPE. Initialization, if requested, is performed on the results or intermediate tape. A list of the selected options is then produced. Both GLOCOM and SECCOM common blocks are used by SECRDR.

3. Input Description

Input data is read from unit READIN in GLOCOM which is either CRDRDR, the card reader, or KEYBD, the console keyboard, when cards are typed in by the user. This selection is accomplished by the LARSYS system.

4. Output Description

Output is written to both PRNTR, the printer, and TYPEWR, an object map and classification map are requested, only an object map is produced. At this point, tape and disk parameters are checked. If OPTIONS INTERMEDIATE (start processing with the intermediate tape) is selected, a tape number must be supplied. If an intermediate tape number is supplied, a file number must also be specified.

Upon specification of OPTIONS INTERMEDIATE, processing begins with the intermediate tape and a results file is produced. If, however, no OPTIONS INTERMEDIATE request is made and both an intermediate tape number and a results location is specified, a conflict is recognized and EXECUTION is terminated. If a results location is needed and none is specified, more data is requested. A file number must be specified when a tape location is selected. Specifications of both tape and disk options for classification results information causes termination.

After the error checking sequence is finished, tapes are mounted on the appropriate devices by calls to MTAPE. Initialization, if requested, is performed on the results or intermediate tape. A list of the selected options is then produced. Both GLOCOM and SECCOM common blocks are used by SECRDR.

3. Input Description

Input data is read from unit READIN in GLOCOM which is either CRDRDR, the card reader, or KEYBD, the console keyboard, when cards are typed in by the user. This selection is accomplished by the LARSYS system.

4. Output Description

Output is written to both PRNTR, the printer, and TYPEWR, the console device. A variety of error messages is produced by SECRDR, and brief list follows:

ERROR IN TAPE OR FILE SPECIFICATION-TYPE CORRECT CARD

A non-numeric character was entered as either a tape or file number.

ERROR IN PARAMETER VALUE SPECIFICATION-TYPE CORRECT CARD

A non-numeric character was specified as the cell size, cell selection, or annexation value.

YOU HAVE ENTERED X SYMBOLS. THE MAXIMUM ALLOWED IS 60. EXCESS SYMBOLS WILL NOT BE USED.

The maximum number of symbols that can be stored is 60. Only the first 60 can be used.

CELL SIZE MUST BE GREATER THAN OR EQUAL TO TWO-DEFAULT OF 2 ASSUMED-TYPE CORRECT CARD

The CELL SIZE (X) parameter cannot be less than 2. A corrected card must be supplied.

CELL SELECTION PARAMETER MUST BE GREATER THAN OR EQUAL TO ZERO-TYPE CORRECT CARD

The CELL SELECT (X) entry cannot be negative, a new card is requested.

ANNEXATION THRESHOLD MUST BE NON-NEGATIVE. TYPE A CORRECTED ANNEXATION THRESHOLD CARD.

The THRES specified is negative. The user is requested to correct the error.

BOTH SINGULAR CELL AND CLASSIFICATION MAPS REQUESTED. ONLY OBJECT MAP WILL BE PRODUCED

Only one map can be produced during annexation. If both are requested, only the singular cell map will be printed.

NO INTERMEDIATE TAPE SUPPLIED FOR OPTIONS INTERMEDIATE. TYPE IN INTERMEDIATE CARD

When requesting that processing start from an intermediate tape, an intermediate tape number must be specified.

INTERMEDIATE FILE OR INITIALIZE MUST BE SPECIFIED. TYPE ADDITIONAL INTERMEDIATE CARD

When producing an intermediate tape, some indication of file number must be given.

BOTH INTERMEDIATE TAPE AND RESULTS LOCATION SPECIFIED WITHOUT OPTIONS INTERMEDIATE-JOB TERMINATED

It is impossible to determine which part of the classification should be performed when this set of options is supplied. Execution is terminated.

EITHER INTERMEDIATE TAPE OR RESULTS LOCATION MUST BE SPECIFIED WITHOUT OPTIONS INTERMEDIATE. TYPE IN ADDITIONAL CARD

Some output file is needed for processing. Additional information is requested.

EITHER RESULTS FILE OR INITIALIZE MUST BE REQUESTED. TYPE IN ADDITIONAL RESULTS CARD

When producing a results tape, some indication of file number must be given.

NO RESULTS DESTINATION SPECIFIED-TYPE IN RESULTS CARD

Either tape and file or disk must be specified when results will be produced. Additional information is requested.

BOTH RESULTS TAPE PARAMETERS AND DISK SPECIFIED-FUNCTION TERMINATED

Either tape or disk can be selected for results but not both.

BOTH FILE AND INITIALIZE REQUESTED FOR INTERMEDIATE TAPE-FILE REQUEST IGNORED

The intermediate output tape is initialized. Only file 1 can be initialized.

BOTH FILE AND INITIALIZE REQUESTED FOR RESULTS TAPE-FILE REQUEST IGNORED

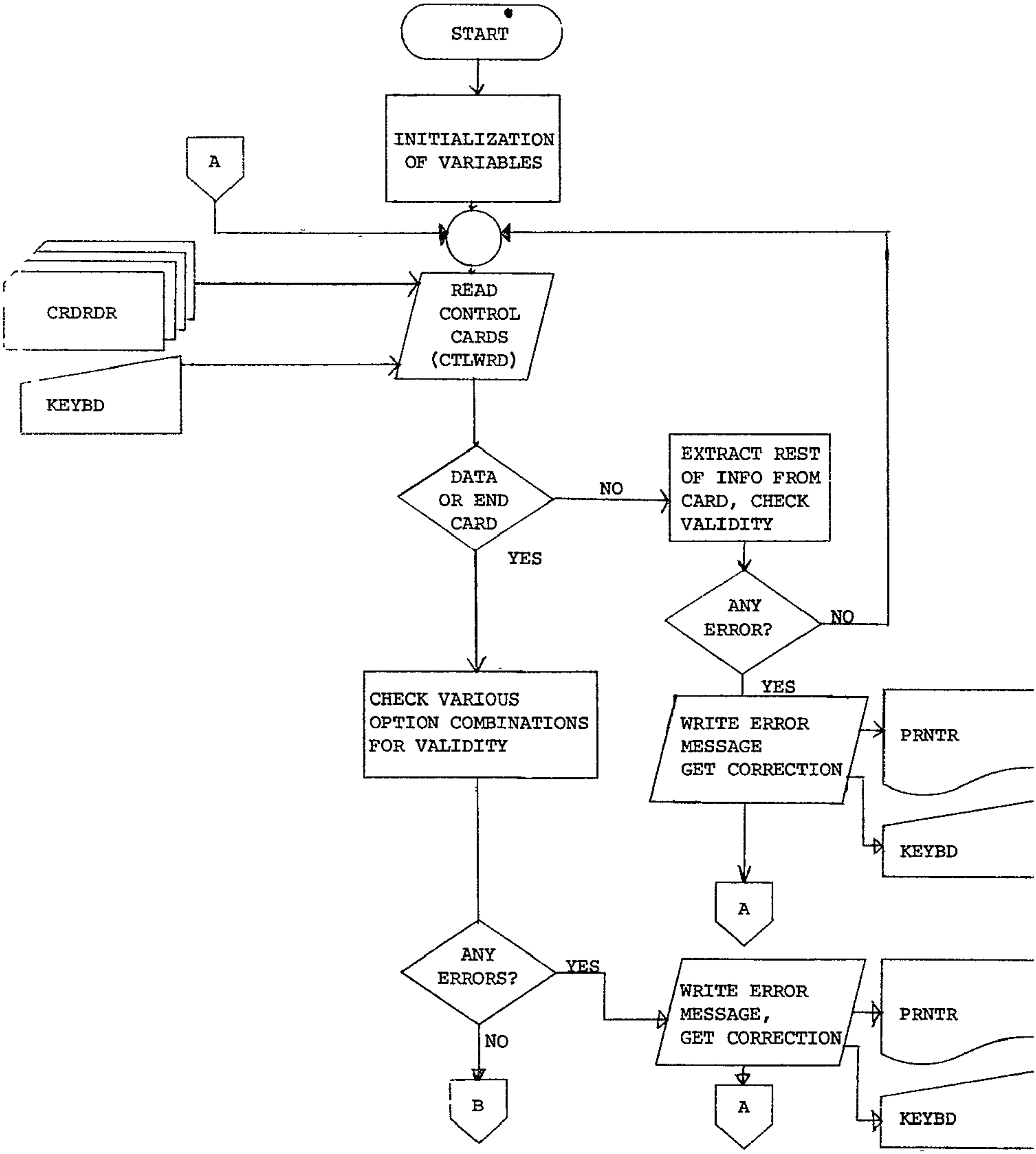
The results tape is initialized. Only file 1 can be initialized.

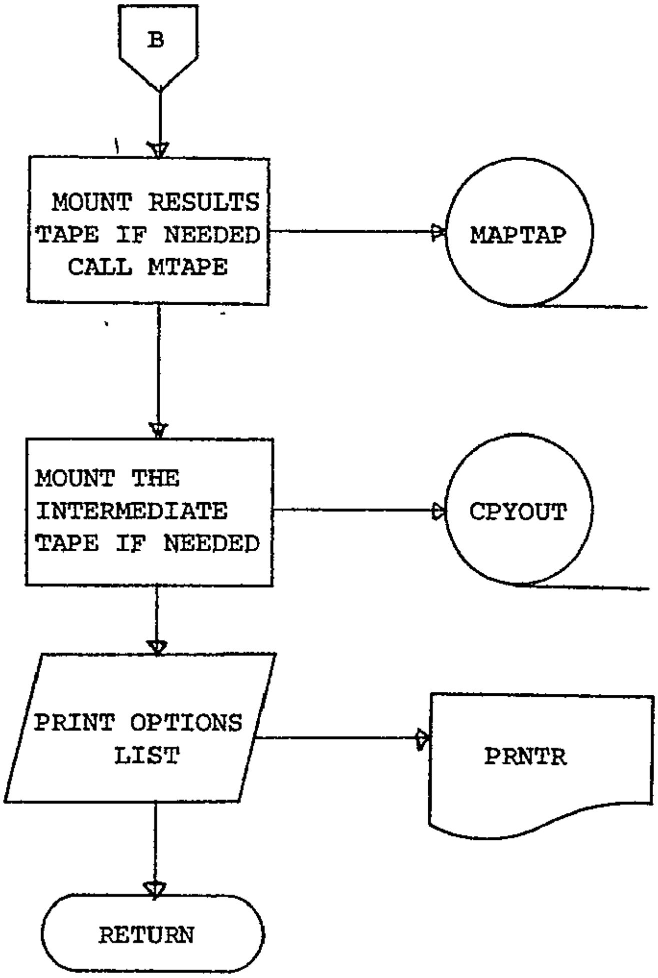
In addition to these messages, the list of options selected is printed on the unit PRNTR.

5. Supplemental Information

Common blocks GLOCOM and SECCOM are used by SECRDR.

6. FLOWCHART


START
INITIALIZATION OF VARIABLES
A
CRDRDR
KEYBD
READ CONTROL CARDS (CTLWRD)
DATA OR END CARD
NO
YES
EXTRACT REST OF INFO FROM CARD, CHECK VALIDITY
ANY ERROR?
NO
YES
WRITE ERROR MESSAGE GET CORRECTION
PRNTR
KEYBD
A
CHECK VARIOUS OPTION COMBINATIONS FOR VALIDITY
ANY ERRORS?
YES
NO
B
WRITE ERROR MESSAGE, GET CORRECTION
PRNTR
KEYBD
A


B
MOUNT RESULTS TAPE IF NEEDED CALL MTAPE
MAPTAP
MOUNT THE INTERMEDIATE TAPE IF NEEDED
CPYOUT
PRINT OPTIONS LIST
PRNTR
RETURN

```
 FORTRAN IV G LEVEL  20.7              SECRDR           DATE = 77129         15018006              PAGE 0001
FILE SECRDR
                C     SECRDR     LARS    XXXX                                                SUP00010
                C                                                                            SUP00020
                C*****************************************************************************SUP00030
                C                                                                            SUP00040
                C   SECRDR - CONTROL CARD READER FOR SUPERVISED ECHO                         SUP00050
                C                                                                            SUP00060
                C*****************************************************************************SUP00070
                C                                                                            SUP00080
   0001               SUBROUTINE SECRDR                                                      SUP00090
                C                                                                            SUP00100
   0002               IMPLICIT INTEGER * 4 (A-Z)                                             SUP00110
                C                                                                            SUP00120
   0003               COMMON /GLCCOM/ BLANK, CARD(20), CHKOUT, COPFIL, CLASSR, CLASSX,       SUP00130
                     1     CLUSTX, CONPUT, CPYCUT, CRDRDR, CRDSEQ, DATAPE,                   SUP00140
                     2     DUPLTP, DUPRUN, ERRMSG, FBPNT,                                    SUP00150
                     3     FILESV, FLDBND, HDATA, HEAD(88), ID(200), IMAGEX,                 SUP00160
                     4     IMARK, KEYBD, MAPTAP, MAXCHA, MAXCLS,                             SUP00170
                     5     PAGSIZ, PNCH, POINT, PRESUX, PRNTR, READIN,                       SUP00180
                     6     RESTRT, RUNFIL, RUNTAB(10,3),                                     SUP00190
                     7     SDATA, SEPARX, SEPTPX, SPARE(10), TEMPAS(30),                     SUP00200
                     8     TPSTAT(6), TTFLDX ,TYPEWR,                                        SUP00210
                     9     TOP, ARRAY(12500)                                                 SUP00220
                C                                                                            SUP00230
   0004               REAL * 8 ARRAY                                                         SUP00240
   0005               REAL * 4 FROCAL(5,30)                                                  SUP00250
   0006               INTEGER * 4 COMENT(16), DATE(5), HED1(16), HED2(16), TIME(5)           SUP00260
   0007               INTEGER * 2 BLANK2                                                     SUP00270
   0008               LOGICAL * 4 CHKOUT                                                     SUP00280
   0009               LOGICAL * 1 BLANK1                                                     SUP00290
   0010               EQUIVALENCE (DATSAV,ID(1)), (CURRUN,ID(3)), (FROCAL(1),ID(5)),         SUP00300
                     1          (HED1(1),HEAD(8)), (DATE(1),HEAD(26)), (HED2(1),HEAD(39)),SUP00310
                     2          (TIME(1),HEAD(58)), (COMENT(1),HEAD(72)),                    SUP00320
                     3          (MAPSAV,TPSTAT(1)),                                          SUP00330
                     4          (SEPSCR,TPSTAT(2)), (DUPIN,TPSTAT(3)), (CASTAT,TPSTAT(4)),SUP00340
                     5          (COPSER,TPSTAT(5)), (TRAOUT,TPSTAT(6)),                      SUP00350
                     6          (BLANK,BLANK2,BLANK1)                                        SUP00360
                C                                                                            SUP00370
                C                                                                            SUP00380
   0011               COMMON /SECCOM/ BUFROZ,CSELEC,CSET(3,30),CSET3(3,30),THRES,CELSIZ,SUP00390
                     1 CELWTH,INFO(17),INPUT, INTTAP, INTFIL, JPTS, LINES, NOCLS, NOFET,SUP00400
                     2 NOFET3,NOPOOL,NWORD,NWORDZ, NVR, OUTPUT, POLNAM(2,60), PREFIX(2),SUP00410
                     3 PTS, ROTAPE, ROFILE, SYMCNT, VARSZ3, CSEL(30), CSEL3(30),         SUP00420
                     3 FETVEC(30), FETVC3(30), POLPTR(2,60), POLSTK(60),                 SUP00430
                     4 CDFLAG, CLSMAP, CSETI(3,30), OBJMAP, PHASE1, PHASE2, POLNMI(60),  SUP00440
                     5 PRSTAT, SYM(60), SYMMTX(60)                                       SUP00450
                C                                                                            SUP00460
   0012               REAL * 4 CSELEC, CSET, CSET3, THRES                                    SUP00470
                C                                                                            SUP00480
   0013               INTEGER * 2 CSEL, CSEL3, FETVEC, FETVC3, POLPTR, POLSTK                SUP00490
                C                                                                            SUP00500
   0014               LOGICAL * 1 CDFLAG, CLSMAP, CSETI, OBJMAP, PHASE1, PHASE2, POLNM1,SUP00510
                     1 PRSTAT                                                            SUP00520
                C                                                                            SUP00530
                C                                                                            SUP00540
                C                                                                            SUP00550
                C                                                                            SUP00560
                C                                                                            SUP00570
                C LOCAL VARIABLE DEFINITION                                                  SUP00580
                C                                                                            SUP00590
   0015               REAL * 4 RWORK(60)                                                     SUP00600
                C                                                                            SUP00610
   0016               INTEGER * 4 SUPLST(12), TAPLST(4), CELLST(2), ANNLST, OPTLST,          SUP00620
                     1 PRNLST(3), CRDLST, IWORK(60)                                      SUP00630
                C                                                                            SUP00640
   0017               LOGICAL * 1 LWORK(1),LSYM(1)                                           SUP00650
                C                                                                            SUP00660
   0018               EQUIVALENCE (IWORK(1),LWORK(1),RWORK(1)),(LSYM(1),SYM(1))              SUP00670
                C                                                                            SUP00680
   0019               DATA SUPLST/'RESU','INTE','CELL','ANNE','OPTI','PRIN','CLAS',          SUP00690
                     1 'CARD','SYMB','CHAN','DATA','END '/                              SUP00700
   0020               DATA TAPLST/'TAPE','FILE','INIT','DISK'/                               SUP00710
   0021               DATA CELLST/'SIZE','SELE'/                                             SUP00720
   0022               DATA ANNLST/'THRE'/, OPTLST/'INTE'/,CRDLST/'READ'/                     SUP00730
   0023               DATA PRNLST/'SING','CLAS','STAT'/                                      SUP00740
                C                                                                            SUP00750
                C INITIALIZATION                                                             SUP00760
```

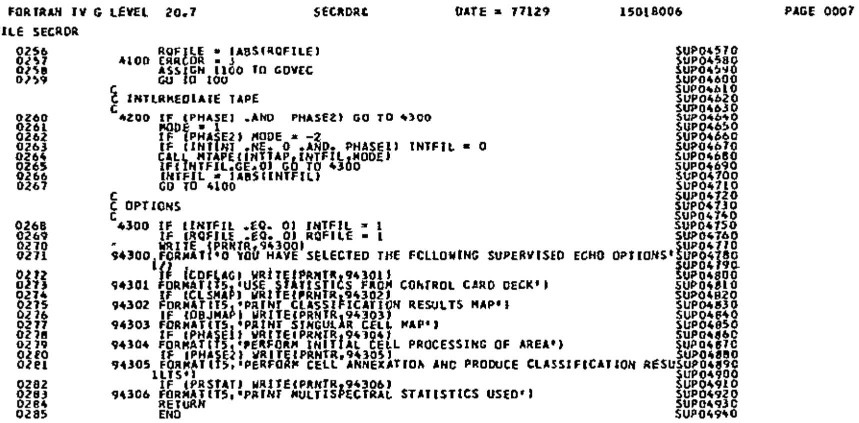
```
 FORTRAN IV G LEVEL  20.7              SECRDRS          DATE = 77129         15018006              PAGE 0002
FILE SECRDR                                                                                          -
                C                                                                            SUP00770
   0024               STOPFG = 0                                                             SUP00780
   0025               OUTPUT = 0                                                             SUP00790
   0026               SYMCNT = 0                                                             SUP00800
   0027               NOPOOL = 0                                                             SUP00810
   0028               CSELEC = 0.                                                            SUP00820
   0029               CELWTH = 0                                                             SUP00830
   0030               OPTINT = 0                                                             SUP00840
   0031               ROTAPE = -1                                                            SUP00850
   0032               ROFILE = -1                                                            SUP00860
   0033               ROINIT = 0                                                             SUP00870
   0034               INTTAP = -1                                                            SUP00880
   0035               INTFIL = -1                                                            SUP00890
   0036               INTINT = 0                                                             SUP00900
   0037               THRES = 0.0                                                            SUP00910
   0038               NOFET3 = 0                                                             SUP00920
   0039               DO 20 I = 1, 30                                                        SUP00930
   0040               CSEL3(I) = 0                                                           SUP00940
   0041               FETVC3(I) = 0                                                          SUP00950
   0042               DO 10 J = 1, 3                                                         SUP00960
   0043               CSET3(J,I) = -50000.                                                   SUP00970
   0044               CSETI(J,I) = .FALSE                                                    SUP00980
   0045            10 CONTINUE                                                               SUP00990
   0046            20 CONTINUE                                                               SUP01000
   0047               DO 30 I = 1, 60                                                        SUP01010
   0048               SYM(I) = SYMMTX(I)                                                     SUP01020
   0049               POLSTK(I) = 0                                                          SUP01030
   0050               POLNMI(I) = .TRUE.                                                     SUP01040
   0051               POLPTR(1,I) = 0                                                        SUP01050
   0052            30 CONTINUE                                                               SUP01060
   0053               PHASE1 = .FALSE.                                                       SUP01070
   0054               PHASE2 = .FALSE.                                                       SUP01080
   0055               CDFLAG = .FALSE.                                                       SUP01090
   0056               CLSMAP = .FALSE.                                                       SUP01100
   0057               OBJMAP = .FALSE.                                                       SUP01110
   0058               PRSTAT = .FALSE                                                        SUP01120
                C                                                                            SUP01130
                C BEGIN CONTROL CARD PROCESSING                                              SUP01140
                C                                                                            SUP01150
   0059               IF (CONPUT .EQ. 2) WRITE (TYPEWR,50)                                   SUP01160
   0060            50 FORMAT(' I****  TYPE IN SUPERVISED ECHO CONTROL CARDS    (SECRDR)'SUP01170
                     1)                                                                  SUP01180
   0061               ERRCOR = 0                                                             SUP01190
   0062               ASSIGN 100 TO GOVEC                                                    SUP01200
   0063            90 GO TO GOVEC,(100,1100)                                                 SUP01210
   0064           100 CALL CTLWRD(CARD,CCL,SUPLST,12,CODE,READIN,ERRCOR)                     SUP01220
   0065               IF (CODE .EQ. 11) SPARE(2) = 1                                         SUP01230
   0066               IF (ERRCOR .EQ. 4) CALL RTMAIN                                         SUP01240
   0067               IF (ERRCOR .EQ. 2) CALL ERPRNT (139,'STOP')                            SUP01250
   0068               GO TO (200,300,400,500,600,700,800,850,900,1000,1100,1050),CODE        SUP01260
                C                                                                            SUP01270
                C ERROR ON CONTROL CARD                                                      SUP01280
                C                                                                            SUP01290
   0069           110 WRITE (TYPEWR,9110) CARD                                               SUP01300
   0070          9110 FORMAT(5X,20A4)                                                        SUP01310
   0071           115 CALL ERPRNT (ERRNUM,'GO')                                              SUP01320
   0072           120 IF (ERRCOR .NE. 3) ERRCOR = 1                                          SUP01330
   0073               GO TO 100                                                              SUP01340
                C                                                                            SUP01350
                C RESULTS CARD                                                               SUP01360
                C                                                                            SUP01370
   0074           200 ENTRY = 2                                                              SUP01380
   0075               GO TO 310                                                              SUP01390
                C                                                                            SUP01400
                C INTERMEDIATE TAPE CARD                                                     SUP01410
                C                                                                            SUP01420
   0076           300 ENTRY = 1                                                              SUP01430
   0077           310 LSTSZ = 2 + ENTRY                                                      SUP01440
   0078           315 IF (COL .EQ. 72) GO TO 90                                              SUP01450
   0079               CALL CTLPRM (CARD, COL, TAPLST, 4, CODE, &120)                         SUP01460
   0080               GO TO (320,330,340,350), CODE                                          SUP01470
                C                                                                            SUP01480
                C TAPE SPECIFICATION                                                         SUP01490
                C                                                                            SUP01500
   0081           320 VECSZ = 1                                                              SUP01510
   0082               CALL IVAL (CARD,COL,IWORK,VECSZ,&321)                                  SUP01520
 FORTRAN IV G LEVEL  20.7               SECRDR&           DATE = 77129          15018006              PAGE 0003
FILE SECRDR
  0083             IF (VECSZ .EQ. 0) GO TO 321                                            SUP01530
  0084             IF (ENTRY .EQ. 1) INTTAP = IWORK(1)                                    SUP01540
  0085             IF (ENTRY .EQ. 2) ROTAPE = IWORK(1)                                    SUP01550
  0086             GO TO 315                                                              SUP01560
             C                                                                           .SUP01570
             C ERROR ON TAPE SPECIFICATION                                                SUP01580
             C                                                                            SUP01590
  0087         321 WRITE (TYPEWR,9348) CARD                                               SUP01600
  0088             WRITE (PRNTR, 9348) CARD                                               SUP01610
  0089        9348 FORMAT(/5X,20A4//7X,'E***  ERROR IN TAPE OR FILE SPECIFICATION - TSUP01620
                  1YPE CORRECT CARD    (SECRDR)')                                         SUP01630
  0090             GO TO 120                                                              SUP01640
             C                                                                            SUP01650
             C FILE SPECIFICATION                                                         SUP01660
             C                                                                            SUP01670
  0091         330 VECSZ = 1                                                              SUP01680
  0092             CALL IVAL(CARD,COL,IWORK,VECSZ,&321)                                   SUP01690
  0093             IF (VECSZ .EQ. 0) GO TO 321                                            SUP01700
  0094             IF (ENTRY .EQ. 1) INTFIL = IWORK(1)                                    SUP01710
  0095             IF (ENTRY .EQ. 2) ROFILE = IWORK(1)                                    SUP01720
  0096             GO TO 315                                                              SUP01730
             C                                                                            SUP01740
             C INITIALIZE REQUEST                                                         SUP01750
             C                                                                            SUP01760
  0097         340 IF (ENTRY .EQ. 1) INTINT = 1                                           SUP01770
  0098             IF (ENTRY .EQ. 2) ROINIT = 1                                           SUP01780
  0099             GO TO 315                                                              SUP01790
             C                                                                            SUP01800
             C RESULTS DISK                                                               SUP01810
             C                                                                            SUP01820
  0100         350 OUTPUT = CLASSR                                                        SUP01830
  0101             GO TO 315                                                              SUP01840
             C                                                                            SUP01850
             C CELL CARD                                                                  SUP01860
             C                                                                            SUP01870
  0102         400 IF (COL .EQ. 72) GO TO 90                                              SUP01880
  0103             CALL CTLPRM (CARD, COL, CELLST, 2, CODE, &120)                         SUP01890
  0104             GO TO (410,420), CODE                                                  SUP01900
             C                                                                            SUP01910
             C CELL SIZE                                                                  SUP01920
             C                                                                            SUP01930
  0105         410 VECSZ = 1                                                              SUP01940
  0106             CALL IVAL(CARD,COL,IWORK,VECSZ,&411)                                   SUP01950
  0107             CELWTH = IWORK(1)                                                      SUP01960
  0108             GO TO 400                                                              SUP01970
             C                                                                            SUP01980
             C ERROR                                                                      SUP01990
             C                                                                            SUP02000
  0109         411 WRITE (TYPEWR,9411) CARD                                               SUP02010
  0110             WRITE (PRNTR,9411) CARD                                                SUP02020
  0111        9411 FORMAT(/5X,20A4//7X,'E***  ERROR IN PARAMETER VALUE SPECIFICATION SUP02030
                  1 - TYPE CORRECT CARD    (SECRDR)')                                     SUP02040
  0112             GO TO 120                                                              SUP02050
             C                                                                            SUP02060
             C CELL SELECTION PARM                                                        SUP02070
             C                                                                            SUP02080
  0113         420 VECSZ = 1                                                              SUP02090
  0114             CALL FVAL(CARD,COL,RWORK,VECSZ,&411)                                   SUP02100
  0115             CSELEC = RWORK(1)                                                      SUP02110
  0116             GO TO 400                                                              SUP02120
             C                                                                            SUP02130
             C ANNEXATION CARD                                                            SUP02140
             C                                                                            SUP02150
  0117         500 IF (COL .EQ. 72) GO TO 90                                              SUP02160
  0118             CALL CTLPRM(CARD,COL,ANNLST,1,CODE,&120)                               SUP02170
  0119             VECSZ = 1                                                              SUP02180
  0120             CALL FVAL(CARD,COL,RWORK,VECSZ,&411)                                   SUP02190
  0121             IF (VECSZ .EQ. 0) GO TO 411                                            SUP02200
  0122             THRES = RWORK(1)                                                       SUP02210
  0123             GO TO 500                                                              SUP02220
             C                                                                            SUP02230
             C OPTIONS CARD                                                               SUP02240
             C                                                                            SUP02250
  0124         600 IF (COL .EQ. 72) GO TO 90                                              SUP02260
  0125             CALL CTLPRM(CARD,COL,OPTLST,1,CODE,&120)                               SUP02270
  0126             OPTINT = 1                                                             SUP02280
```

```
 FORTRAN IV G LEVEL  20 7               SECRDR&           DATE = 77129          15018006              PAGE 0004
FILE SECRDR
  0127             GO TO 600                                                              SUP02290
             C                                                                            SUP02300
             C PRINT CARD                                                                 SUP02310
             C                                                                            SUP02320
  0128         700 IF (COL .EQ. 72) GO TO 90                                              SUP02330
  0129             CALL CTLPRM (CARD,COL,PRNLST,3,CODE,&120)                              SUP02340
  0130             IF (CODE .EQ. 1) OBJMAP = .TRUE.                                       SUP02350
  0131             IF (CODE .EQ. 2) CLSMAP = .TRUE.                                       SUP02360
  0132             IF (CODE .EQ. 3) PRSTAT = .TRUE.                                       SUP02370
  0133             GO TO 700                                                              SUP02380
             C                                                                            SUP02390
             C CLASSES CARD                                                               SUP02400
             C                                                                            SUP02410
  0134         800 CALL POLSGN (POLNAM,POLPTR,POLSTK,POLNMI,NOPOOL,STKPTR,COL,CODE,       SUP02420
                  1 &810)                                                                 SUP02430
  0135             GO TO 90                                                               SUP02440
             C                                                                            SUP02450
             C ERROR ON CLASSES CARD                                                      SUP02460
             C                                                                            SUP02470
  0136         810 STOPFG = 1                                                             SUP02480
  0137             CALL ERPRNT (341,'GOTO',&90)                                           SUP02490
             C                                                                            SUP02500
             C CARDS READSTATS                                                            SUP02510
             C                                                                            SUP02520
  0138         850 IF (COL .EQ. 72) GO TO 90                                              SUP02530
  0139             CALL CTLPRM(CARD,COL,'READ',1,CODE,&120)                               SUP02540
  0140             COFLAG = .TRUE.                                                        SUP02550
  0141             GO TO 850                                                              SUP02560
             C                                                                            SUP02570
             C SYMBOLS CARD                                                               SUP02580
             C                                                                            SUP02590
  0142         900 I = 34                                                                 SUP02600
  0143             CALL BCDVAL(CARD,COL,WORK,I,&940)                                      SUP02610
  0144             K = SYMCNT + I                                                         SUP02620
  0145             IF (K .GT. 60) GO TO 950                                               SUP02630
  0146         902 DO 905 J = 1,I                                                         SUP02640
  0147         905 LSYM((SYMCNT+J)*4) = LWORK(4*J-3)                                      SUP02650
  0148             SYMCNT = K                                                             SUP02660
  0149             GO TO 90                                                               SUP02670
  0150         940 ERRNUM = 447                                                           SUP02680
  0151             GO TO 110                                                              SUP02690
             C                                                                            SUP02700
             C TOO MANY SYMBOLS SPECIFIED                                                 SUP02710
             C                                                                            SUP02720
  0152         950 WRITE (TYPEWR,9950)K                                                   SUP02730
  0153             WRITE(PRNTR,9950)K                                                     SUP02740
  0154        9950 FORMAT(' I0073  YOU HAVE ENTERED',I3,' SYMBOLS. THE MAXIMUM ALLOWESUP02750
                  1D IS 60.'/9X,'EXCESS SYMBOLS WILL NOT BE USED    (SECRDR)')            SUP02760
  0155             I = I - (K-60)                                                         SUP02770
  0156             GO TO 902                                                              SUP02780
             C                                                                            SUP02790
             C CHANNELS CARD                                                              SUP02800
             C                                                                            SUP02810
  0157        1000 CALL CHANEL(CARD,COL,NCR,CSEL3,CSET3,FETVC3,&1070)                     SUP02820
  0158             DO 1060 I = 1,3                                                        SUP02830
  0159             DO 1060 J = 1,30                                                       SUP02840
  0160        1060 IF (CSET3(I,J) .NE. -50000 ) CSET1(I,J) = .TRUE.                       SUP02850
  0161             NOFET3 = NCR                                                           SUP02860
  0162             GO TO 90                                                               SUP02870
  0163        1070 ERRNUM = 171                                                           SUP02880
  0164             GO TO 110                                                              SUP02890
             C                                                                            SUP02900
             C DATA OR END CARD                                                           SUP02910
             C                                                                            SUP02920
  0165        1050 SPARE(2) = 1                                                           SUP02930
  0166        1100 IF (STOPFG .EQ. 1) CALL RTMAIN                                         SUP02940
             C                                                                            SUP02950
             C CHECK IF NUMBER OF CHAN GT 0                                               SUP02960
             C                                                                            SUP02970
  0167             IF (NOFET3 .GT. 0) GO TO 1120                                          SUP02980
  0168             ERRNUM = 343                                                           SUP02990
  0169             ASSIGN 1100 TO GOVEC                                                   SUP03000
  0170             ERRCOR = 3                                                             SUP03010
  0171             GO TO 115                                                              SUP03020
             C                                                                            SUP03030
             C CELL SIZE GT 0 IF SPECIFIED                                                SUP03040
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
FORTRAN IV G LEVEL  20 7               SECRDRE          DATE = 77129          15018006                PAGE 0005
FILE SECRDR

                C                                                                           SUP03050
 0172            1120 IF (CELWTH .GE. 2) GO TO 1145                                         SUP03060
 0173                 CELWTH = 2                                                            SUP03070
 0174                 WRITE (TYPEWR,91120)                                                  SUP03080
 0175                 WRITE (PRNTR,91120)                                                   SUP0309C
 0176           91120 FORMAT(' E***  CELL SIZE MUST BE GREATER THAN OR EQUAL TO 2 - DEFASUP03100
                    1ULT OF 2 ASSUMED  (SECRDR)')                                           SUP03110
 0177                 GO TO 1145                                                            SUP03120
 0178            1125 ASSIGN 1100 TO GOVEC                                                  SUP03130
 0179                 ERRCOR = 1                                                            SUP03140
 0180                 GO TO 100                                                             SUP03150
                C                                                                           SUP03160
                C BOTH MAPS                                                                 SUP03170
                C                                                                           SUP03180
 0181            1145 IF (.NCT. CLSMAP  OR. .NOT  OBJMAP) GO TO 1150                        SUP03190
 0182                 WRITE (TYPEWR,91145)                                                  SUP0320C
 0183                 WRITE (PRNTR,91145)                                                   SUP03210
 0184           91145 FORMAT(' E***  BOTH SINGULAR CELL AND CLASSIFICATION MAPS REQUESTESUP03220
                    1D - ONLY SINGULAR CELL MAP'/9X,'WILL BE PRODUCED    (SECRDR)')         SUP0323C
 0185                 CLSMAP = .FALSE.                                                      SUP03240
                C                                                                           SUP03250
                C*******************************************************************SUP03260
                C                                                                           SUP03270
                C BEGIN TAPE PARAMETER CHECKING FOR RESULTS AND INTERMEDIATE TAPE            SUP0328C
                C                                                                           SUP0329C
                C*******************************************************************SUP03300
                C                                                                           SUP03310
 0186            1150 IF (OPTINT .NE. 0) PHASE2 = .TRUE.                                    SUP03320
 0187                 IF (OPTINT .EQ. 0 .AND.RQTAPE .EQ. -1.AND CUTPUT.EQ.0) PHASE1 =       SUP0333C
                    1 .TRUE.                                                                SUP03340
 0188                 IF(OPTINT.NE 0.OR (RQTAPE.EQ -1.AND OUTPUT.EQ 0))                     SUP03350
                    1 GO TO 1151                                                            SUP03360
 0189                 PHASE1 = .TRUE.                                                       SUP03370
 0190                 PHASE2 = .TRUE                                                        SUP03380
 0191            1151 CONTINUE                                                              SUP03390
                C                                                                           SUP03400
                C CSELEC GE 0                                                               SUP03410
                C                                                                           SUP03420
 0192            1130 IF (CSELEC .GE. 0.0 .OR..NOT PHASE1) GO TO 1140                       SUP03430
 0193                 WRITE (TYPEWR,91130)                                                  SUP03440
 0194                 WRITE (PRNTR,91130)                                                   SUP03450
 0195           91130 FORMAT(' E***  CELL SELECTION PARAMETER MUST BE GREATER THAN OR '/SUP0346C
                    19X,'EQUAL TO ZERO - TYPE CORRECT CARD    (SECRDR)')                    SUP0347C
 0196                 GO TO 1125                                                            SUP03480
                C                                                                           SUP03490
                C CHECK FOR ANN GT 0                                                        SUP03500
                C                                                                           SUP03510
 0197            1140 IF (THRES  GT. 0.0 .OR  .NOT.PHASE2) GO TO 1152                       SUP03520
 0198                 WRITE(TYPEWR,91140)                                                   SUP0353C
 0199                 WRITE(PRNTR,91140)                                                    SUP0354C
 0200           91140 FORMAT(' E***  ANNEXATION THRESHOLD MUST BE NON-NEGATIVE.'            SUP0355C
                    1 /8X,'TYPE A CORRECTED ANNEXATION THRESHOLD CARD   (SECRDR)')          SUP0356C
 0201                 GO TO 1125                                                            SUP0357C
                C                                                                           SUP03580
                C TAPE PARAMS                                                               SUP03590
                C                                                                           SUP03600
 0202            1152 IF(PHASE1.AND PHASE2)GO TO 1600                                       SUP03610
 0203                 IF (INITAP .NE  -1) GO TO 1155                                        SUP0362C
 0204                 WRITE(TYPEWR,91150)                                                   SUP03630
 0205                 WRITE(PRNTR,91150)                                                    SUP03640
 0206           91150 FORMAT(' E***  NO INTERMEDIATE TAPE SUPPLIED FO',                     SUP03650
                    1'R OPTIONS INTERMEDIATE.'/8X,'TYPE IN INTERMEDIATE CARD    (SECRDRSUP0366C
                    2)')                                                                    SUP0367C
 0207                 GO TO 1125                                                            SUP03680
                C                                                                           SUP03690
                C CHECK FOR INTERMEDIATE FILE OR INIT                                       SUP03700
                C                                                                           SUP03710
 0208            1155 IF (INTFIL .NE. -1  OR. INTINT .NE  0) GO TO 1560                     SUP03720
 0209                 WRITE(PRNTR,91155)                                                    SUP03730
 0210                 WRITE(TYPEWR,91155)                                                   SUP03740
 0211           91155 FORMAT(' E***  INTERMEDIATE FILE CR INITIALIZE MUST BE SPECIFIED 'SUP0375C
                    1/8X,'TYPE ADDITICNAL INTERMEDIATE CARD    (SECRDR)')                   SUP0376C
 0212                 GO TO 1125                                                            SUP03770
                C                                                                           SUP03780
                C CHECK FOR INTERMEDIATE TAPE FOR OUTPUT AND BOTH FILE AND INIT             SUP03790
                C                                                                           SUP03800
```

```
FORTRAN IV G LEVEL  20 7               SECRDRE          DATE = 77129          15018006                PAGE 0006
FILE SECRDR
 0213            1560 IF (.NOT.PHASE1 .OR. INTFIL  EQ. -1 .OR. INTINT  EQ. 0)               SUP0381C
                    1 GO TO 1600                                                            SUP03820
 0214                 WRITE(TYPEWR,91560)                                                   SUP03830
 0215                 WRITE(PRNTR,91560)                                                    SUP03840
 0216           91560 FORMAT(' IC000  BOTH FILE AND INITIALIZE REQUESTED FOR INTERMEDIATSUP0385C
                    1E TAPE -'/8X,'FILE REQUEST IGNOREC     (SECRDR)')                      SUP03860
 0217                 INTFIL = -1                                                           SUP03870
                C                                                                           SUP03880
                C CHECK FOR BOTH INT AND RESULTS                                            SUP03890
                C                                                                           SUP03900
 0218            1600 IF((INTTAP.EQ.-1.CR (RQTAPE.EC.-1.AND OUTPUT EQ 0)).OR.OPTINT NE. SUP03910
                    1 0)GO TO 2100                                                          SUP03920
 0219                 WRITE(TYPEWR,92000)                                                   SUP03930
 0220                 WRITE(PRNTR,92000)                                                    SUP03940
 0221           92000 FORMAT(' E***  BOTH INTERMEDIATE TAPE AND RESULTS LOCATION SPECIFISUP03950
                    1ED WITHOUT OPTIONS'/9X,'INTERMEDIATE - JOB TERMINATED    (SECRDR)'SUP03960
                    2)                                                                      SUP0397C
 0222                 CALL RTMAIN                                                           SUP03980
                C                                                                           SUP03990
                C SEE IF EITHER TAPE WAS REQUESTED                                          SUP04000
                C                                                                           SUP04010
 0223            2100 IF (INTTAP.NE -1 OR RQTAPE NE.-1.CR DUTPUT NE 0.OR.OPTINT.EQ.1)       SUP04020
                    1GO TO 2200                                                             SUP04030
 0224                 WRITE(TYPEWR,92100)                                                   SUP04040
 0225                 WRITE(PRNTR,92100)                                                    SUP04050
 0226           92100 FORMAT(' E***  EITHER INTERMEDIATE TAPE OR RESULTS LOCATION MUST BSUP04060
                    1E SPECIFIED WITHOUT'/8X, 'CPTIONS INTERMECIATE - TYPE IN ADDITIONASUP04070
                    2L CARD     (SECRDR)')                                                  SUP04080
 0227                 GO TO 1125                                                            SUP04090
                C                                                                           SUP04100
                C CHECK FOR RESULTS IF NEEDED                                               SUP04110
                C                                                                           SUP04120
 0228            2200 IF (.NOT. PHASE2) GO TO 4000                                          SUP04130
 0229                 IF (RQTAPE.NE.-1.OR.OUTPUT.NE 0) GO TO 3050                           SUP04140
 0230                 WRITE(TYPEWR,93000)                                                   SUP0415C
 0231                 WRITE(PRNTR,93000)                                                    SUP04160
 0232           93000 FORMAT(' E***  NO RESULTS DESTINATION SPECIFIED - TYPE IN RESULTS SUP04170
                    1 CARD     (SECRDR)')                                                   SUP04180
 0233                 GO TO 1125                                                            SUP04190
                C                                                                           SUP04200
                C CHECK FOR FILE AND INIT                                                   SUP04210
                C                                                                           SUP04220
 0234            3050 IF (RQFILE .EQ. -1 .AND. RQINIT .EQ  0) GO TO 3075                    SUP0423C
 0235                 WRITE(TYPEWR,93050)                                                   SUP04240
 0236                 WRITE(PRNTR,93050)                                                    SUP04250
 0237           93050 FORMAT(' E***  EITHER RESULTS FILE OR INITIALIZE MUST BE',            SUP04260
                    1 ' REQUESTED '/8X,'TYPE IN ADDITICNAL RESULTS CARD     (SECRDR)')      SUP0427C
 0238                 GO TO 1125                                                            SUP04280
                C                                                                           SUP04290
                C CHECK FOR BOTH FILE AND INIT SPECIFIED ON RESULTS REQUEST                 SUP04300
                C                                                                           SUP04310
 0239            3075 IF (RQFILE .EQ. -1 .OR. RQINIT .EC. 0) GO TO 3100                     SUP04320
 0240                 WRITE(TYPEWR,93075)                                                   SUP04330
 0241                 WRITE(PRNTR,93075)                                                    SUP04340
 0242           93075 FORMAT(' I0000  BOTH FILE AND INTIALIZE RECUESTED FOR RESULTS TAPESUP04350
                    1 -'/8X,'FILE REQUEST IGNORED     (SECRDR)')                            SUP0436C
 0243                 RQFILE = -1                                                           SUP04370
                C                                                                           SUP04380
                C CHECK FOR TAPE PARAM AND DISK                                             SUP04390
                C                                                                           SUP04400
 0244            3100 IF (OUTPUT .EQ. 0) GO TO 4000                                         SUP04410
 0245                 IF (RQTAPE .EQ.-1.AND.RQFILE.EQ -1.AND.RQINIT.EQ 0) GO TO 4000        SUP04420
 0246                 WRITE (TYPEWR,93100)                                                  SUP04430
 0247                 WRITE (PRNTR,93100)                                                   SUP0444C
 0248           93100 FORMAT(' E***  BOTH RESULTS TAPE PARAMETERS AND DISK SPECIFIED - FSUP0445C
                    1UNCTION TERMINATED     (SECRDR)')                                      SUP04460
 0249                 CALL RTMAIN                                                           SUP04470
 0250            4000 IF (OUTPUT .EQ  0) OUTPUT = MAPTAP                                    SUP04480
 0251                 IF (CHKOUT) RETURN                                                    SUP04490
                C                                                                           SUP04500
                C MOUNT TAPES  - RESULT                                                     SUP04510
                C                                                                           SUP04520
 0252                 IF (.NOT.PHASE2 .OR. OUTPUT .EQ. CLASSR) GO TO 4200                   SUP04530
 0253                 IF (RQINIT  EQ  1) RQFILE = 0                                         SUP04540
 0254                 CALL MTAPE(RQTAPE,RQFILE,1)                                           SUP04550
 0255                 IF (RQFILE .GE. 0) GO TO 4200                                         SUP04560
```



```
 FORTRAN IV G LEVEL  20.7                SECRDR           DATE = 77129          15018006                PAGE 0007
FILE SECRDR

  0256              RQFILE = IABS(RQFILE)                                                    SUP04570
  0257         4100 ERRCOR = 3                                                               SUP04580
  0258              ASSIGN 1100 TO GOVEC                                                     SUP04590
  0259              GO TO 100                                                                SUP04600
             C                                                                               SUP04610
             C INTERMEDIATE TAPE                                                             SUP04620
             C                                                                               SUP04630
  0260        4200 IF (PHASE1 .AND  PHASE2) GO TO 4300                                       SUP04640
  0261              MODE = 1                                                                 SUP04650
  0262              IF (PHASE2) MODE = -2                                                    SUP04660
  0263              IF (INTINT .NE. 0 .AND. PHASE1) INTFIL = 0                               SUP04670
  0264              CALL MTAPE(INTTAP,INTFIL,MODE)                                           SUP04680
  0265              IF(INTFIL.GE.0) GO TO 4300                                               SUP04690
  0266              INTFIL = IABS(INTFIL)                                                    SUP04700
  0267              GO TO 4100                                                               SUP04710
             C                                                                               SUP04720
             C OPTIONS                                                                       SUP04730
             C                                                                               SUP04740
  0268        4300 IF (INTFIL .EQ. 0) INTFIL = 1                                             SUP04750
  0269              IF (RQFILE .EQ. 0) RQFILE = 1                                            SUP04760
  0270              WRITE (PRNTR,94300)                                                      SUP04770
  0271        94300 FORMAT('0 YOU HAVE SELECTED THE FOLLOWING SUPERVISED ECHO OPTIONS'SUP04780
                  1/)                                                                        SUP04790
  0272              IF (CDFLAG) WRITE(PRNTR,94301)                                           SUP04800
  0273        94301 FORMAT(T5,'USE STATISTICS FROM CONTROL CARD DECK')                       SUP04810
  0274              IF (CLSMAP) WRITE(PRNTR,94302)                                           SUP04820
  0275        94302 FORMAT(T5,'PRINT CLASSIFICATION RESULTS MAP')                            SUP04830
  0276              IF (OBJMAP) WRITE(PRNTR,94303)                                           SUP04840
  0277        94303 FORMAT(T5,'PRINT SINGULAR CELL MAP')                                     SUP04850
  0278              IF (PHASE1) WRITE(PRNTR,94304)                                           SUP04860
  0279        94304 FORMAT(T5,'PERFORM INITIAL CELL PROCESSING OF AREA')                     SUP04870
  0280              IF (PHASE2) WRITE(PRNTR,94305)                                           SUP04880
  0281        94305 FORMAT(T5,'PERFORM CELL ANNEXATION AND PRODUCE CLASSIFICATION RESUSUP04890
                  1LTS')                                                                     SUP04900
  0282              IF (PRSTAT) WRITE(PRNTR,94306)                                           SUP04910
  0283        94306 FORMAT(T5,'PRINT MULTISPECTRAL STATISTICS USED')                         SUP04920
  0284              RETURN                                                                   SUP04930
  0285              END                                                                      SUP04940
```

LARS Program Abstract ________

MODULE IDENTIFICATION

Module Name: SECINT   Function Name: SECSUP

Purpose: Initiator routine for Supervised ECHO classifier.

System/Language: CMS/Fortran

Author: P.D. Alenduff   Date: 4/27/77

Latest Revisor: ________   Date: ________

MODULE ABSTRACT

SECINT carries out required initialization of the rest of variables used by SECSUP and finishes reading any data cards as well as the statistics to be used with the processor. In addition to the array allocation performed, some option information is printed and a loop is entered that carries out the desired stages of classification for each area to be classified.

1. Module Usage

SECINT

```
CALL SECINT
```

The program SECINT is called with no arguments. Any variables to be used or changed are contained in common blocks GLOCOM and SECCOM, both required by SECINT. All information required to perform the classification is gathered by SECINT from the appropriate source (ie., statistics and areas to be classified) and array allocation is carried out using variables supplied to common block SECCOM by the control card reader SECRDR. Statistics are reduced by calling REDSAV to save statistics for only classes and channels requested and statistics information is printed by calling SECPRT. Initial records are then written to the output device. A loop is entered and, for each area to be classified, a sequence of tests is performed. If multispectral data is required, the correct run is requested and existance of requested channels is confirmed. Further allocation is performed and SECH01 is called to perform the needed processing for the area.

2. Internal Description

The first information handled by SECINT is the statistics file. If processing is starting from an intermediate tape, records 1 and 2 are read from the intermediate tape to re-establish information used in initial processing such as channels selected and calibration used, and classes selected and pooling requested. The statistics file is then read from the tape and written onto the unit SDATA, the statistics file on disk. The next card is checked for an "end" card and if any area description cards are required, the function is terminated.

Otherwise, CLASSX, the storage disk file for all classification areas for this file, is rewound and processing loops until all areas are stored on disk. Some initial allocation is then performed as array storage is set up for the covariance and mean matrices for the entire statistics file on disk. REDSTA transfers the file into memory. CLSCHK and FETCHK then check for fatal errors in selection of pooling and channels. The addresses of reduced arrays for storing only needed information are then computed and REDSAV reduces the information to that required.

Initial records are then produced and written to OUTPUT. The entire statistics file from SDATA is then transfered to the output file, and SECPRT is called to print the statistics information, if requested, and to produce the record type 4 with covariance and means matrices of the classes. A loop is then started and executed once for each area to be classified. If processing begins with the data type, the correct input run is requested and calibration information is established. If processing begins with an intermediate type, all areas on the tape in this file are used. Field size and the allocation of all other arrays used by SECHO1 are calculated. Then a record type 5, area identification record, is produced and SECHO1 is called once for each area to be processed. When finished, the final record is written to the output device.

3. Input Description

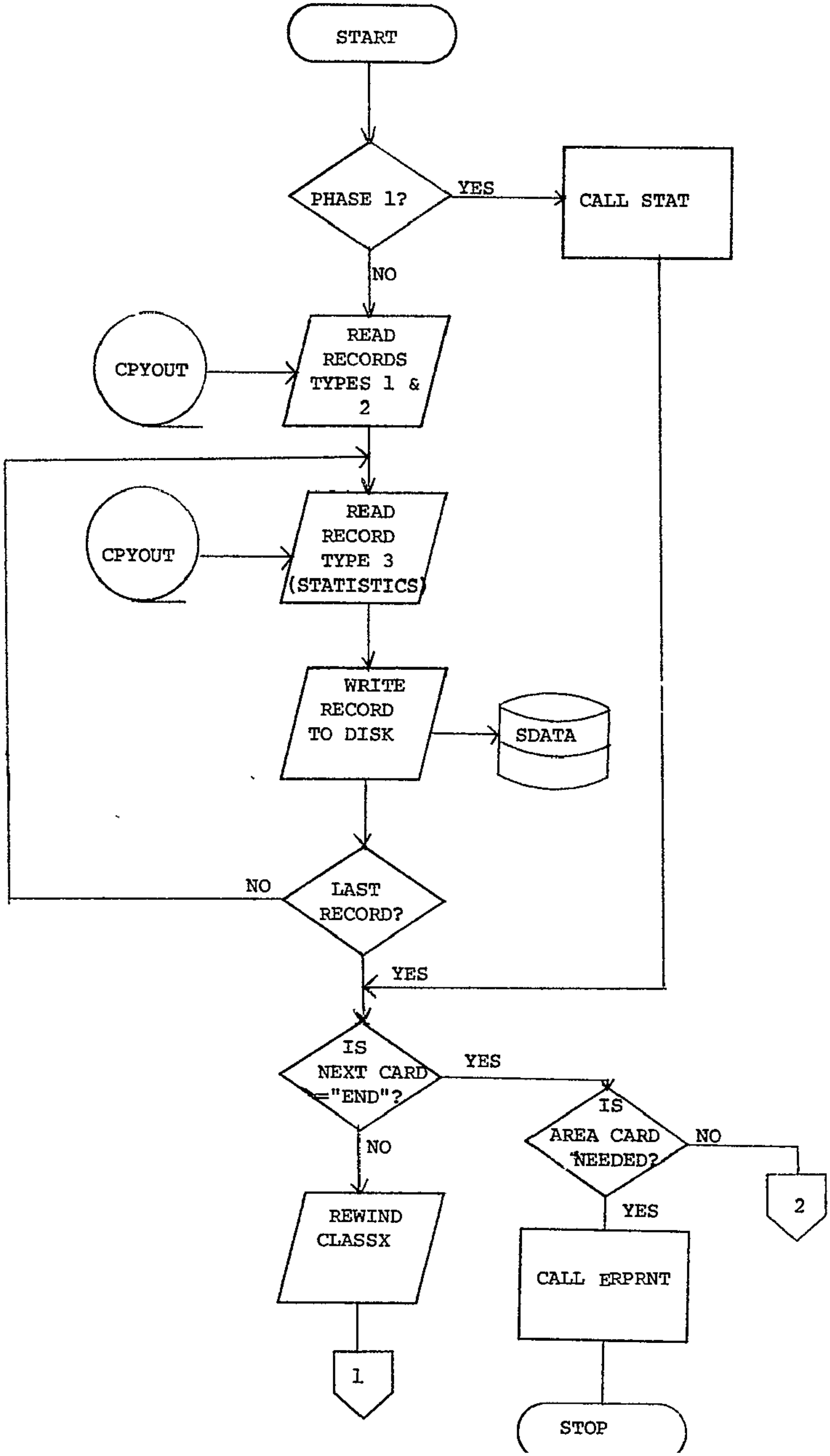
Input cell information for SECINT is read from CPYOUT if an intermediate tape is requested as the starting point of processing. The format is identical to that of a classification results file. SDATA, the statistics file on disk is also read to be copied into the output file. CLASSX, the areas to be classified as stored on disk, is also used to store the list of areas to be processed.

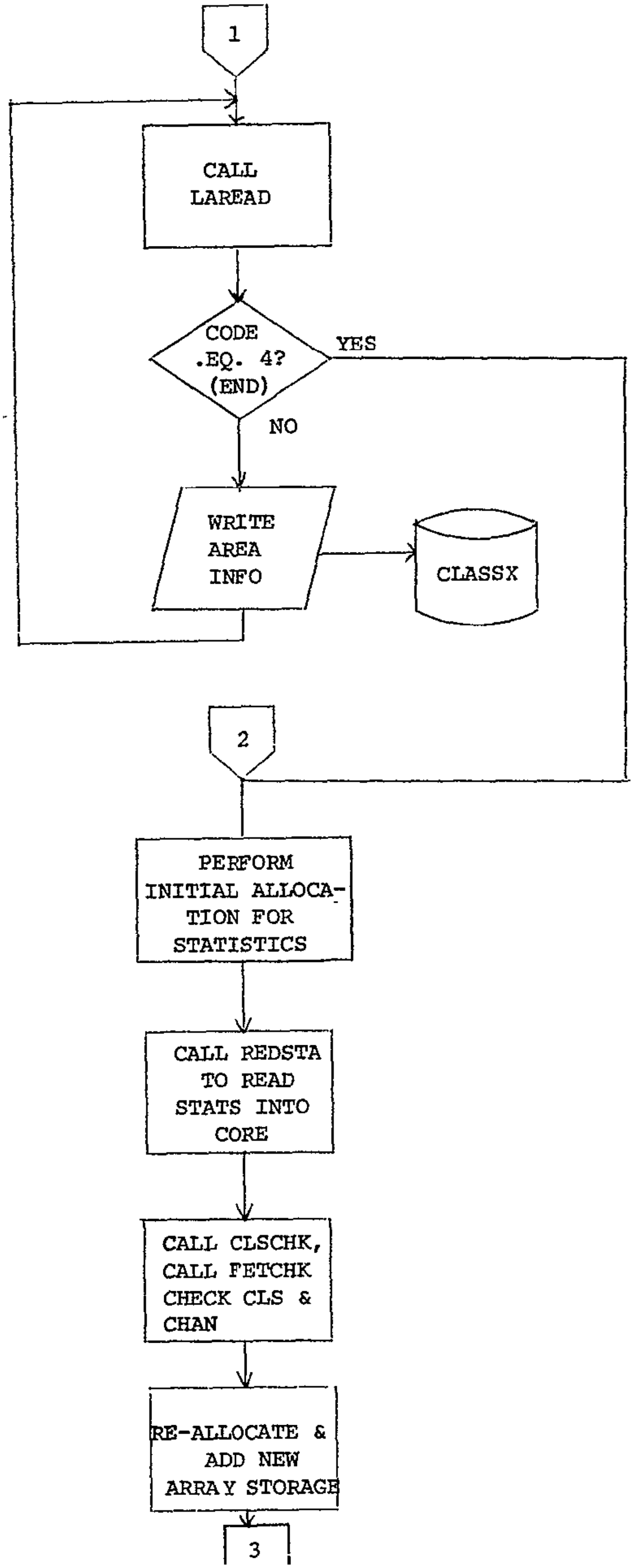
4. Output Description

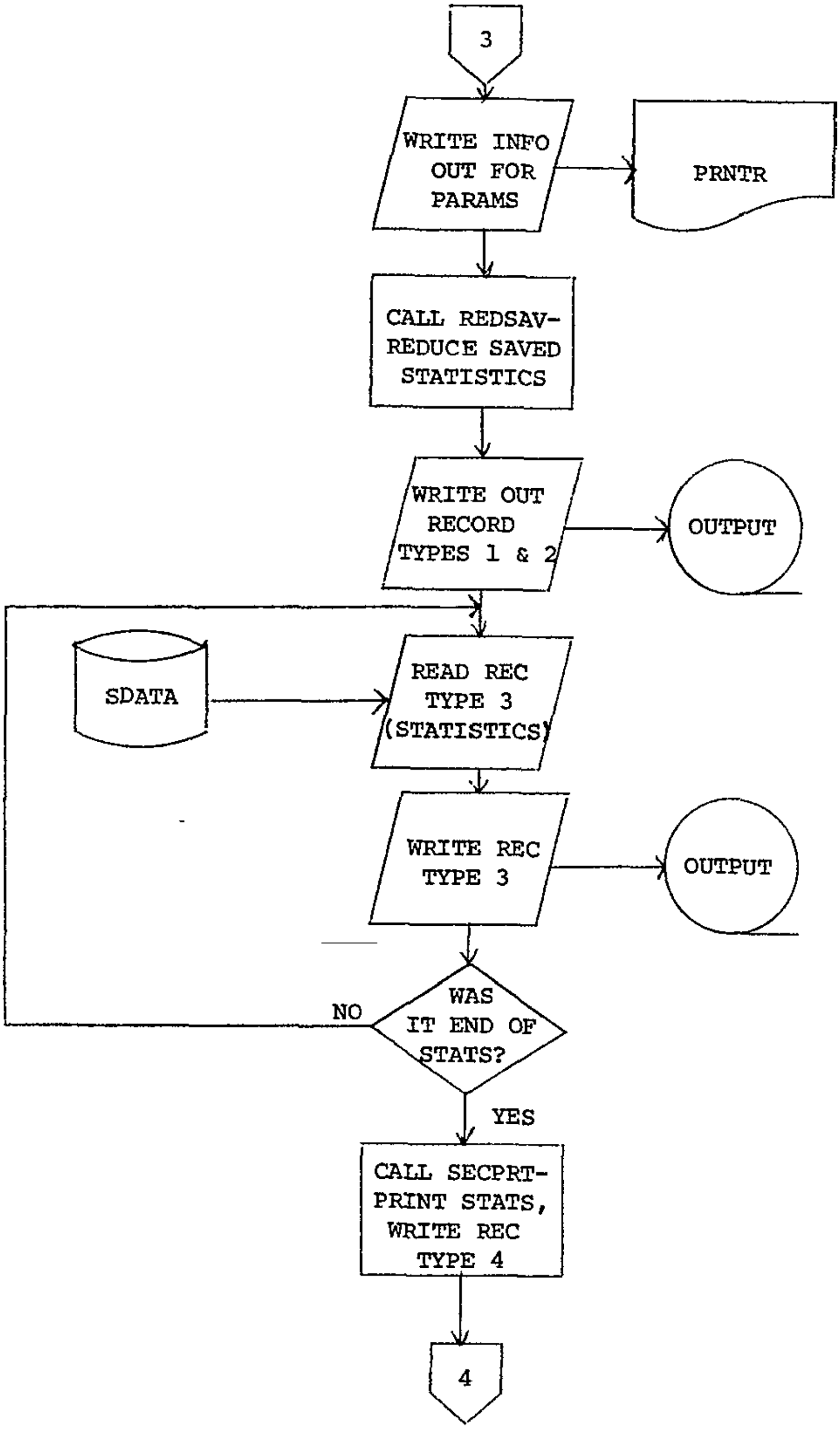
A copy of the statistics file is written to SDATA from the intermediate tape is used as input. CLASSX is written as the list of areas to be classified. The PRNTR is used for messages and output information. The file OUTPUT is the result of processing with the same format as a classification results file.

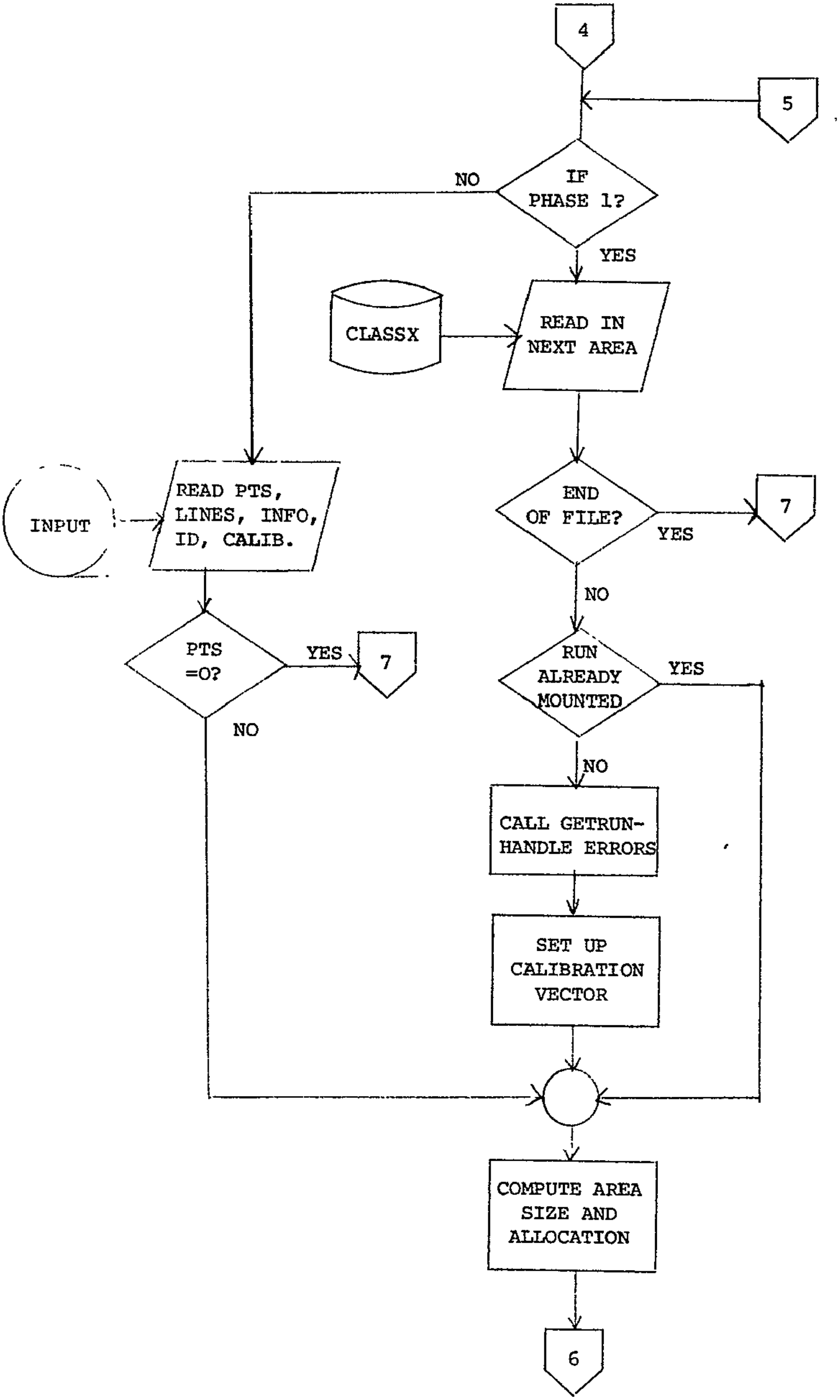
5. Supplemental Information

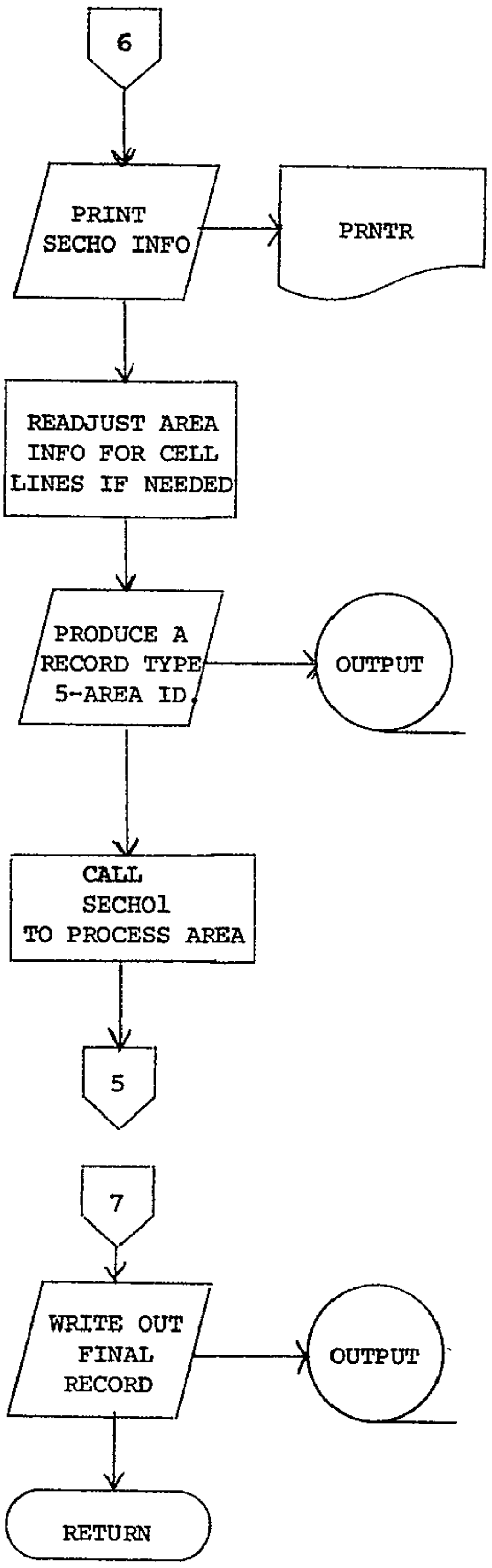
This program uses common blocks GLOCOM and SECCOM. See LARSYS Systems Manual for classification results file format.


START
PHASE 1?
YES
CALL STAT
NO
CPYOUT
READ RECORDS TYPES 1 & 2
CPYOUT
READ RECORD TYPE 3 (STATISTICS)
WRITE RECORD TO DISK
SDATA
NO
LAST RECORD?
YES
IS NEXT CARD ="END"?
YES
IS AREA CARD NEEDED?
NO
2
NO
YES
REWIND CLASSX
CALL ERPRNT
1
STOP


1
CALL LAREAD
CODE .EQ. 4? (END)
YES
NO
WRITE AREA INFO
CLASSX
2
PERFORM INITIAL ALLOCA- TION FOR STATISTICS
CALL REDSTA TO READ STATS INTO CORE
CALL CLSCHK, CALL FETCHK CHECK CLS & CHAN
RE-ALLOCATE & ADD NEW ARRAY STORAGE
3


3
WRITE INFO OUT FOR PARAMS
PRNTR
CALL REDSAV-REDUCE SAVED STATISTICS
WRITE OUT RECORD TYPES 1 & 2
OUTPUT
SDATA
READ REC TYPE 3 (STATISTICS)
WRITE REC TYPE 3
OUTPUT
WAS IT END OF STATS?
NO
YES
CALL SECPRT-PRINT STATS, WRITE REC TYPE 4
4


4
5
IF PHASE 1?
NO
YES
CLASSX
READ IN NEXT AREA
INPUT
READ PTS, LINES, INFO, ID, CALIB.
END OF FILE?
YES
7
NO
PTS =0?
YES
7
NO
RUN ALREADY MOUNTED
YES
NO
CALL GETRUN-HANDLE ERRORS
SET UP CALIBRATION VECTOR
COMPUTE AREA SIZE AND ALLOCATION
6


6
PRINT SECHO INFO
PRNTR
READJUST AREA INFO FOR CELL LINES IF NEEDED
PRODUCE A RECORD TYPE 5-AREA ID.
OUTPUT
CALL SECH01 TO PROCESS AREA
5
7
WRITE OUT FINAL RECORD
OUTPUT
RETURN

```
FILE SECINT
               C   SECINT     LARS    XXXX                                               SEC00010
               C                                                                         SEC00020
               C*************************************************************************SEC00030
               C                                                                         SEC00040
               C PERFORM ECHO SUPERVISED VERSION ON AREA                                  SEC00050
               C                                                                         SEC00060
               C*************************************************************************SEC00070
               C                                                                         SEC00080
               C        WRITTEN  03/4/77 BY P.D. ALENDUFF                                SEC00090
               C                                                                         SEC00100
               C*************************************************************************SEC00110
               C                                                                         SEC00120
  0001                SUBROUTINE SECINT                                                  SEC00130
  0002                IMPLICIT INTEGER * 4 (A-Z)                                         SEC00140
               C                                                                         SEC00150
  0003               COMMON /GLCCOM/ BLANK, CARD(20), CHKOUT, COPFIL, CLASSR, CLASSX,    SEC00160
                    1    CLLSTX, CONPUT, CPYOUT, CRDRDR, CRDSEQ, DATAPE,                 SEC00170
                    2    DUPLIP, DUPRUN, ERRMSG, FBPNT,                                  SEC00180
                    3    FILESV, FLDBND, HDATA, HEAD(88), ID(200), IMAGEX,               SEC00190
                    4    IMARK, KEYBD, MAPTAP, MAXCHA, MAXCLS,                           SEC00200
                    5    PAGSIZ, PNCH, POINT, PRESUX, PRNTR, READIN,                     SEC00210
                    6    RESTRT, RUNFIL, RUNTAB(10,3),                                   SEC00220
                    7    SDATA, SEPARX, SEPTPX, SPARE(10), TEMPAS(30),                   SEC00230
                    8    TPSTAT(6), TTFLDX ,TYPEWR,                                      SEC00240
                    9    TOP, ARRAY(12500)                                               SEC00250
               C                                                                         SEC00260
  0004               REAL * 8 ARRAY                                                      SEC00270
  0005               REAL * 4 FROCAL(5,30)                                               SEC00280
  0006               INTEGER * 4 COMENT(16), DATE(5), HED1(16), HED2(16), TIME(5)        SEC00290
  0007               INTEGER * 2 BLANK2                                                  SEC00300
  0008               LOGICAL * 4 CHKOUT                                                  SEC00310
  0009               LOGICAL * 1 BLANK1                                                  SEC00320
  0010               EQUIVALENCE (CATSAV,ID(1)), (CURRUN,ID(3)), (FROCAL(1),ID(51)),     SEC00330
                    1        (HED1(1),HEAD(8)), (DATE(1),HEAD(26)), (HED2(1),HEAD(39)),SEC00340
                    2        (TIME(1),HEAD(58)), (COMENT(1),HEAD(72)),                   SEC00350
                    3        (MAPSAV,TPSTAT(1)),                                         SEC00360
                    4        (SEPSCR,TPSTAT(2)), (DUPIN,TPSTAT(3)), (DASTAT,TPSTAT(4)),SEC00370
                    5        (CCPSER,TPSTAT(5)), (TRAOUT,TPSTAT(6)),                     SEC00380
                    6        (BLANK,BLANK2,BLANK1)                                       SEC00390
               C                                                                         SEC00400
               C                                                                         SEC00410
  0011               COMMON /SECCOM/ BUFROZ,CSELEC,CSET(3,30),CSET3(3,30),THRES,CELSIZ,SEC00420
                    1 CELWTH,INFO(17),INPUT, INTTAP, INTFIL, JPTS, LINES, NOCLS, NOFET,SEC00430
                    2 NOFET3,NOPOOL,NWCRD,NWORD2, NVR, OUTPUT, POLNAM(2,60), PREFIX(2),SEC00440
                    3 PTS, ROTAPE, ROFILE, SYMCNT, VARSZ3, CSEL(30), CSEL3(30),          SEC00450
                    3 FETVEC(30), FETVC3(30), PCLPTR(2,60), POLSTK(60),                  SEC00460
                    4 CDFLAG, CLSMAP, CSET1(3,30), OBJMAP, PHASE1, PHASE2, POLNM1(60),   SEC00470
                    5 PRSTAT, SYM(60), SYMMTX(60)                                        SEC00480
               C                                                                         SEC00490
  0012               REAL * 4 CSELEC, CSET, CSET3, THRES                                 SEC00500
               C                                                                         SEC00510
  0013               INTEGER * 2 CSEL, CSEL3, FETVEC, FETVC3, POLPTR, POLSTK             SEC00520
               C                                                                         SEC00530
  0014               LOGICAL * 1 CDFLAG, CLSMAP, CSET1, OBJMAP, PHASE1, PHASE2, POLNM1,SEC00540
                    1 PRSTAT                                                             SEC00550
               C                                                                         SEC00560
               C                                                                         SEC00570
               C                                                                         SEC00580
               C                                                                         SEC00590
               C                                                                         SEC00600
               C                                                                         SEC00610
               C  LOCAL VARIABLE DEFINITION                                              SEC00620
               C                                                                         SEC00630
  0015               REAL * 8 BUFFER(6250)                                               SEC00640
  0016               RFAL*4 WORK(60)                                                     SEC00650
  0017               INTEGER * 4 CLSDAT(5), CARD1(19), KEPPTS(60),NOCNT(60),             SEC00660
                    1 IWORK(1)                                                           SEC00670
  0018               INTEGER * 2 BINFLG                                                  SEC00680
  0019               LOGICAL * 1 CHKFLG                                                  SEC00690
  0020               EQUIVALENCE (WORK(1),NOCNT(1)),(CARD(1),CARD1(1)),                  SEC00700
                    1 (IWORK(1),WORK(1))                                                 SEC00710
               C                                                                         SEC00720
  0021               DATA IEOS/'EOS '/,END/'END '/                                       SEC00730
               C                                                                         SEC00740
  0022               INPUT = CPYOUT                                                      SEC00750
  0023               AREAFG = 0                                                          SEC00760
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

```
FILE SECINT
  0024                CALL GTSERL(SERIAL)                                                 SEC00770
  0025                IF (PHASE1) GO TO 200                                               SEC00780
  0026                IF (CHKOUT) GO TO 300                                               SECOC79C
                C                                                                         SEC00800
                C READ STATISTICS TO DISK FROM INTERMEDIATE TAPE                          SEC0081C
                C                                                                         SEC00820
  0027                READ (INPUT,END=9000)I,I,I,I,I,I,SERIAL,MODFLG                      SEC00830
  0028                IF (MODFLG .NE. 0) GO TO 105                                        SEC00840
  0029                READ (INPUT,END=9C00)I,I,NOCLS4,NCFET3,NOFLD,NOPCOL,                SEC0085C
                     1 (FETVC3(I),I=1,NCFET3),(CSEL3(FETVC3(I)),I=1,NOFET3),              SEC00860
                     2(FRQCAL(1,FETVC3(I)),I=1,NCFET3),(FRQCAL(2,FETVC3(I)),I=1,NOFET3),SEC0087C
                     3 ((PCLNAM(I,J),I=1,2),J=1,NOPCOL),((PCLPTR(I,J),I=1,2),J=1,NOPOOL)SEC00880
                     4,(PCLSTK(I),I=1,NCCLS4),(CLSDAT(I),I=3,5),CLSDAT(1),CLSDAT(2)       SEC00890
  0030                GO TO 110                                                           SEC00900
  0031            105 READ (INPUT,END=9000)I,I,NOCLS4,NOFET3,NOFLD,NOPOOL,                SEC00910
                     1 (FETVC3(I),I=1,NOFET3),(CSEL3(FETVC3(I)),I=1,NOFET3),              SEC00920
                     2(FRQCAL(1,FETVC3(I)),I=1,NCFET3),(FRQCAL(2,FETVC3(I)),I=1,NOFET3) SEC0093C
                     3,((PCLNAM(I,J),I=1,2),J=1,NOPCOL),((POLPTR(I,J),I=1,2),J=1,NOPOOL)SEC00940
                     4,(PCLS]K(I),I=1,NOCLS4),(WORK(I),I=1,NOPOOL),(CLSDAT(I),I=3,5),    SEC00950
                     5 CLSDAT(1),CLSDAT(2)                                                SEC00960
  0032                CELWIH = IWORK(1)                                                   SEC00970
  0033                THRES = WORK(2)                                                     SEC00980
                C                                                                         SEC00990
                C READY TO CCPY FROM INPUT TAPE TO DISK                                   SEC01C00
                C                                                                         SEC01010
  0034            110 REWIND SDATA                                                        SEC01020
  0035                READ(INPUT,END=9000)I,I,CARD1,CRDSEQ                                SEC01030
  0036                BINFLG = 0                                                          SEC01040
  0037                IF (CARD1(11) .EQ. IONE) BINFLG = 1                                 SEC01C50
  0038                WRITE(SDATA,9110) CARD1,CRDSEQ                                      SEC01060
  0039           9110 FORMAT(18A4,I8)                                                     SEC01070
  0040            115 READ(INPUT,END=9000) I,I,CARD1,CRDSEQ                               SEC01080
  0041                WRITE(SDATA,9115)CARD1,CRDSEQ                                       SEC01C90
  0042           9115 FORMAT(18A4,I8)                                                     SEC01100
  0043                IF (CARD1(1).NE. IEOS) GO TO 115                                    SEC01110
  0044                GO TO 300                                                           SEC01120
                C                                                                         SEC01130
                C CHECK STATS ON DISK                                                     SEC01140
                C                                                                         SEC01150
  0045            200 CHKFLG = .FALSE.                                                    SEC01160
  0046                UNIT = CRDRDR                                                       SEC01170
  0047                IF (.NCT.COFLAG) UNIT = SDATA                                       SEC01180
  0048                IF (UNIT .EQ. CRDRDR.OR..NOT.CHKOUT) GO TO 210                      SEC01190
  0049                REWIND SDATA                                                        SEC012C0
  0050                READ(SCATA,9200)I                                                   SEC01210
  0051           9200 FORMAT(20A4)                                                        SEC01220
  0052                IF (I.NE. IEOS) GO TO 210                                           SEC01230
  0053                CHKFLG=.TRUE.                                                       SEC0124C
                C                                                                         SEC01250
                C STATS ON CARDS OR DISK                                                  SEC01260
                C                                                                         SEC01270
  0054            210 CALL STAT(UNIT,NOCLS,NOFLD,NOFET,NUM,BINFLG)                        SEC01280
                C                                                                         SEC0129C
                C STATS ARE NOW CN DISK - TRANSFER TO CORE                                SEC01300
                C                                                                         SEC01310
  0055            300 REWIND CLASSX                                                       SEC01320
  0056                IF (PHASE2.AND..NOT. PHASE1) AREAFG = 1                             SEC01330
  0057                IF (CARD(1) .NE. END) GO TO 310                                     SEC01340
  0058                IF (PHASE1) CALL ERPRNT(254,'STOP')                                 SEC01350
  0059                GO TO 350                                                           SEC01360
                C                                                                         SEC01370
                C READ AREA CARDS                                                         SEC01380
                C                                                                         SEC01390
  0060            310 IF (AREAFG .EQ. 1) GO TO 350                                        SEC01400
  0061                CALL LAREAD (INFO,CODE,READIN,0,0)                                  SEC01410
  0062                IF (CODE .EQ. 4) GO TO 350                                          SEC01420
  0063                IF (CODE .NE. 0) CALL ERPRNT(373,'&GOTO',300)                       SEC01430
  0064                IF (CHKOUT) GO TO 310                                               SEC0144C
  0065                WRITE(CLASSX) INFO                                                  SEC0145C
  0066                GO TO 310                                                           SEC01460
                C                                                                         SEC01470
                C CONTINUE PROCESSING                                                     SEC01480
                C                                                                         SEC01490
  0067            350 VARSIZ = NOFET*(NOFET+1)/2                                          SEC01500
  0068                COVAR1 = 1                                                          SEC01510
  0069                AVAR1 = COVAR1 + (VARSIZ*NOCLS+1)/2                                 SEC01520
```

```
      CLSID1 = AVAR1 + (NOFET * NOCLS + 1)/2
      CATBAS = CLSID1 + NOCLS
      CORE = TOP - (DATBAS*8)
      IF (CORE .LE. 0) CALL ERPRNT(371,'STOP')
      CALL REDSTA(ARRAY(COVAR1),ARRAY(AVAR1),ARRAY(CLSID1),KEPPTS,
     1BINFLG,NOCNT,NOCLS,CSET,CSEL,FETVEC,VARSIZ,NOFET)
C
C STATS ARE IN CCRE - CHECK CLASSES AND CHANNELS IF NEEDED
C
C     IF WE JUST GOT THEM OFF TAPE THEN THEY ARE OK
C
      IF (PHASE2.AND..NCT.PHASE1) GO TO 380
      NO = 0
      N = 0
      CALL CLSCHK(ARRAY(CLSID1),NOCLS,NCPOOL,POLPTR,POLSTK,POLNAM,POLMN1
     1,STKPTR,NO)
      CALL FETCHK(MAXCHA,FETVEC,NOFET,NCFET3,FETVC3,CSEL3,CSEL)
      IF (NOFET3 .LE. 0) CALL ERPRNT(372,'STOP')
      N = POLPTR(2,NOPOCL)+ POLPTR(1,NOPOOL) - 1
      DO 360 II=1,N
  360 NOFLD3 = NCFLD3 + NOCNT(POLSTK(II))
C
C ALLOCATE STORAGE AND PRINT DATA
C
  380 VARSZ3 = NOFET3*(NOFET3+1)/2
      COVAR3 = CCVAR1
      AVAR3 = COVAR3 + (VARSZ3*NCPOCL+1)/2
      SCRAR3 = AVAR3 + (NOFET3*NCPOCL + 1)/2
      DETAR3 = SCRAR3 + (NOPOOL+1)/2
      CONAR3 = DETAR3 + (NCPOOL+1)/2
      DATBS3 = CCNAR3 + (NCPCOL+1)/2
      END = CATBS3 + (NOFET3*(NOFET3+2)+1)/2
      CORE = TOP - END*8
      IF (CORE .LT.0)CALL ERPRNT(371,'STOP')
      IF (.NOT. PHASE1) CALL SECRD(ARRAY(COVAR3),ARRAY(AVAR3))
C
C PRINT SUPERVISOR INFORMATION
C
      WRITE(PRNTR,9350)
 9350 FORMAT(//' SUPERVISED ECHO INFORMATION....'/)
      WRITE(PRNTR,9351)SERIAL,NOPCOL,NOFET3,NOFLD
 9351 FORMAT(5X,'CLASSIFICATION STUDY...........',I9,
     1       /5X,'NO.OF PCOLS.....................',I4,
     2       /5X,'NC.OF CHANNELS..................',I4,
     3       /5X,'NO. OF TRAINING FIELDS.........',I4)
C
C REDUCE STATISTICS
C
      CALL REDSAV(ARRAY(COVAR1),ARRAY(AVAR1),ARRAY(COVAR3),ARRAY(AVAR3),
     1 KEPPTS,NOPOOL,NOFET,VARSIZ,NOFET3,POLPTR,POLSTK,FETVEC,FETVC3,
     2 VARSZ3)
C
C WRITE OUT INITIAL RECORDS
C
      IF (CHKOUT) RETURN
      DO 390 I=1,NOPOOL
  390 WORK(I) = 0.0
      II = 3
      JJ = 0
      KK = 1
      I = INTTAP
      J = INIFIL
      PREFIX(1) = 1
      PREFIX(2) = 0
      IF (PHASE2) I = RQTAPE
      IF (PHASE2) J = RQFILE
      WRITE (OUTPUT) PREFIX,I,J,II,JJ,SERIAL,KK,(JJ,IJ=1,6)
      PREFIX(1) = 2
      IF (.NCT. PHASE2) IWORK(1) = CELWTH
      IF (.NOT. PHASE2) WORK(2) = THRES
      WRITE (CUTPUT) PREFIX,NOCLS,NCFET3,NOFLD,NGPCOL,
     1 (FETVC3(I),I=1,NOFET3),(CSEL3(FETVC3(I)),I=1,NOFET3),
     2(FRGCAL(1,FETVC3(I)),I=1,NOFET3),(FRQCAL(2,FETVC3(I)),I=1,NOFET3)
     3,((POLNAM(I,J),I=1,2),J=1,NOPOOL),((POLPTR(I,J),I=1,2),J=1,NOPOOL)
     4,(POLSTK(I),I=1,NOCLS),(WORK(I),I=1,NCPOOL),DATE
C
```

```
  0117            REWIND SDATA                                                      SEC02290
  0118            PREFIX(1) = 3                                                     SEC02300
  0119        400 READ(SCATA,9390,END=9100) CARD                                    SEC02310
  0120       9390 FORMAT(20A4)                                                      SEC02320
  0121            WRITE (OUTPUT) PREFIX,CARD                                        SEC02330
  0122            IF (CARD(1) .EQ. IEOS) GO TO 410                                  SEC02340
  0123            GO TO 400                                                         SEC02350
            C                                                                       SEC02360
            C PRINT STATS, CALC INVERSE COVAR MTX,SCAPRD, VECTOR PRD, DET           SEC02370
            C                                                                       SEC02380
  0124        410 CALL SECPRT(ARRAY(COVAR3),ARRAY(AVAR3),ARRAY(SCRAR3),ARRAY(DETAR3)SEC02390
                 1,ARRAY(CONAR3),ARRAY(DATBS3))                                     SEC02400
  0125        420 IF (.NOT. PHASE1) GO TO 458                                       SEC02410
            C***********************************************************************SEC02420
            C                                                                       SEC02430
            C PHASE1 ONLY                                                           SEC02440
            C                                                                       SEC02450
            C***********************************************************************SEC02460
  0126            REWIND CLASSX                                                     SEC02470
  0127            PREFIX(2) = 0                                                     SEC02480
            C                                                                       SEC02490
            C ENTER LOOP TO READ AND CLASSIFY THE REGIONS.                          SEC02500
            C                                                                       SEC02510
  0128        425 READ (CLASSX, END = 530) INFO                                     SEC02520
  0129        426 PREFIX(2) = PREFIX(2)+1                                           SEC02530
            C                                                                       SEC02540
            C GET RUN NUMBER                                                        SEC02550
            C                                                                       SEC02560
  0130            IF(CURRUN.EQ.INFO(1))GO TO 460                                    SEC02570
  0131            CALL GETRUN(INFO(1),DATAPE,ID,ERROR,RUNTAB,IMARK)                 SEC02580
  0132            IF(ERROR.GT.0) CALL RUNERR(ERROR,INFO(1))                         SEC02590
  0133        430 DO 440 I = 1,NOFET3                                               SEC02600
  0134            IF(FETVC3(I).LE.ID(5))GO TO 440                                   SEC02610
  0135            WRITE(PRNTR,9430)FETVC3(I)                                        SEC02620
  0136            WRITE(TYPEWR,9430)FETVC3(I)                                       SEC02630
  0137       9430 FORMAT (10X,' CHANNEL ',I3)                                       SEC02640
  0138            CALL ERPRNT (178,'STOP')                                          SEC02650
  0139        440 CONTINUE                                                          SEC02660
            C                                                                       SEC02670
            C SET UP CALIBRATION SET VECTOR.                                        SEC02680
            C                                                                       SEC02690
  0140            DO 450 J = 1,30                                                   SEC02700
  0141            DO 450 I = 1,3                                                    SEC02710
  0142        450 IF(CSET3(I,J) .EQ. -50000.) CSET3(I,J) = FRQCAL(I+2,J)            SEC02720
            C                                                                       SEC02730
            C CHECK FOR NEW CALIBRATION                                             SEC02740
            C                                                                       SEC02750
  0143            FLAG = 0                                                          SEC02760
  0144            DO 455 I = 1,NOFET3                                               SEC02770
  0145            DO 455 J = 1,3                                                    SEC02780
  0146        455 IF(ABS(CSET(J,FETVC3(I))-CSET3(J,FETVC3(I))) .GT. 1.02) FLAG = 1  SEC02790
  0147            IF(FLAG.EQ.1) WRITE(TYPEWR,9455)                                  SEC02800
  0148            IF(FLAG.EQ.1) WRITE(PRNTR,9455)                                   SEC02810
  0149       9455 FORMAT(' I0214 CALIBRATION VALUES USED FOR CLASSIFICATION ARE NOT SEC02820
                 1SAME AS THOSE USED IN THE STATISTICS')                            SEC02830
  0150            GO TO 460                                                         SEC02840
            C                                                                       SEC02850
            C***********************************************************************SEC02860
            C                                                                       SEC02870
            C PHASE 2 ONLY - CHECK ALL SUCESSIVE AREAS ON TAPE                      SEC02880
            C                                                                       SEC02890
            C***********************************************************************SEC02900
            C                                                                       SEC02910
  0151        458 READ(INPUT) PREFIX,PTS,LINES,INFO,ID,CSET3                        SEC02920
  0152            IF (PTS .EQ. 0) GO TO 426                                         SEC02930
            C                                                                       SEC02940
            C          COMPUTE FIELD SIZE AND SET UP STORAGE ALLOCATION.            SEC02950
            C                                                                       SEC02960
  0153        460 LSAMNO = ID(6) - 6                                                SEC02970
  0154            IF (INFO(8) .LE. LSAMNO) GO TO 470                                SEC02980
  0155            WRITE (TYPEWR,9460) INFO(8), LSAMNO                               SEC02990
  0156       9460 FORMAT (' I0212 LAST SAMPLE NUMBER (',I4,') EXCEEDS LAST SAMPLE ONSEC03000
                 1 TAPE.  LAST SAMPLE NUMBER RESET TO',I5,'.')                      SEC03010
  0157            WRITE (PRNTR,9460) INFO(8), LSAMNO                                SEC03020
  0158           -INFO(8) = LSAMNO                                                  SEC03030
            C                                                                       SEC03040
```

FILE SECINT

```
C SET UP JPTS AS NO OF CELLS HORIZONTALLY,PTS AS NUMBER OF TRUNCATED PTS
C
  470 JPTS = (INFO(8)-INFO(7)+INFO(9))/(INFO(9)*CELWTH)
      PTS = JPTS * CELWTH
C
C RESET INFO (8) TO CORRESPOND TO TRUNCATION
C
      INFO(8) = INFO(7) + (PTS-1)*INFO(9)
      J = INFO(6) * CELWTH
      INFO(5) = INFO(4) + J*((INFO(5)-INFO(4)+INFO(6))/J) - INFO(6)
      NSR = PTS+6
      CELSIZ = CELWTH*CELWTH
C
C NVR IS NUM OF SAMPLES IN ROATA TO SKIP TO GET NEXT CHAN OF DATA
C      FOR THIS POINT
C
      NVR = NSR*NOFET3
      NWORD = MAXO(NOPOOL,(CELSIZ+1)/2)
      NWORD2 = NWORD*2
C
C ALLOCATE ARRAYS USED IN ACTUAL CELL SPLITTING BY ECHO3
C
C ARRAYS ARE ------        BDATA FOR READING IN DATA
C                          PIXCLS FOR CLASS CF GIVEN PIXEL FROM NON-HOMOGEN
C                                 CELL
C                          CELIKE FOR STORING LIKELIHOOD CODE FOR EACH CELL
C                         ROATA FOR READING IN DATA
C                         PIXCOR FOR CORRELATION (DATA VALUE PRODUCTS) OF PIX
C                         PIXVAL FOR ACTUAL PIXEL DATA VALUES
C                         CELCOR FOR CELL CORRELATION MATRIX VALUES
C                         CELSUM FOR SUM OF ALL CATA VALUES IN CELL
C                         CELCLS FOR CLASS NC OF CELL
C                         DATBS3 FOR REST OF STUFF
C
C*******************************************************************
C
      CELSUM = DATBS3
      VARSZ3 = NCFET3*(NOFET3+1)/2
      IF (.NCT. PHASE1) GO TO 480
      BDATA = DATBS3
      RDATA = BDATA + (NOFET3*ID(6) + 7)/8
      PIXCOR = ROATA + (NVR*CELWTH+1)/2
      PIXVAL = PIXCOR + (VARSZ3*CELSIZ+1)/2
      CELCOR = PIXVAL + (NOFET3*CELSIZ+1)/2
      CELSUM = CELCOR + (VARSZ3+1)/2
  480 CELCLS = CELSUM + (NOFET3+1)/2
      CELIKE = CELCLS + (JPTS+3)/4
      PIXCLS = CELIKE
      PRTBUF = CELIKE + (NWORD*JPTS+1)/2
      DATBS3 = PRTBUF + (PTS+3)/4
C
C PHASE 2 ALLOCATION
C
      IF (.NCT. PHASE2) GO TO 490
      STACK = DATBS3
      CLOSED = STACK + (JPTS+1)/2
      OPEN = CLOSED + (JPTS + 3)/4
      FLOSIZ = OPEN + (JPTS+3)/4
      FLOCLS = FLOSIZ + (JPTS+1)/2
      FLOLIK = FLOCLS + (JPTS+3)/4
      DATBS3 = FLOLIK + (JPTS * NOPOOL+1)/2
C
C
C
  490 IF (DATBS3 .GT. TOP) CALL ERPRNT(139,'STOP')
      BUFROZ = 50000/(JPTS*2*CELSIZ)
      IF (BUFROZ .LE. 1) CALL ERPRNT(139,'STOP')
C
C PRINT OUT PROCESSING PARAMETERS
C
      WRITE(PRNTR,9500) CELWTH,NOFET3,NCPOOL,THRES, CSELEC, BUFROZ
 9500 FORMAT(////'0PROCESSING PARAMETERS'//
     1 '   CELL WIDTH = ',I2//
     2 '   NUMBER OF CHANNELS = ',I3//
     3 '   NUMBER OF POOLED CLASSES = ',I3//
     4 '   ANNEXATION THRESHOLD = ',1P E12.4//
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

```
                 4.'  CELL SELECTION THRESHOLD = ',1P E12.4//                       SEC03810
                 5'  NUMBER OF CELL LINES IN BUFFER = ',I7)                         SEC03820
            C                                                                       SEC03830
            C PRINT RUN HEADER                                                      SEC03840
            C                                                                       SEC03850
0196              CALL GETIME(TIME)                                                 SEC03860
0197              WRITE(PRNTR,HEAD)                                                 SEC03870
0198              CALL HEADER(NOFET3,FETVC3,CSEL3,CSET3,ID,FRQCAL)                  SEC03880
            C***********************************************************************SEC03890
            C                                                                       SEC03900
            C WRITE RECORD TYPE 5.                                                  SEC03910
            C                                                                       SEC03920
            C***********************************************************************SEC03930
            C                                                                       SEC03940
0199          520 LINES = (INFO(5)-INFO(4))/INFO(6) + 1                             SEC03950
            C                                                                       SEC03960
            C CORRECT AREA INFO PUT ON INTERMEDIATE TAPE TO REFLECT CELL LINES      SEC03970
            C                                                                       SEC03980
0200          527 PREFIX(1) = 5                                                     SEC03990
0201              WRITE (OUTPUT) PREFIX,PTS,LINES,INFO,ID,CSET3                     SEC04000
0202              PREFIX(1) = 6                                                     SEC04010
            C                                                                       SEC04020
            C PRINT HEADER.                                                         SEC04030
            C                                                                       SEC04040
0203              WRITE(PRNTR,9520) (INFO(I),I=4,9)                                 SEC04050
0204         9520 FORMAT('0AREA PROCESSED.....LINES ',I6,'-',I5,'  (BY ',I2,')'/   SEC04060
                 & T21,'COLUMNS ',I4,'-',I5,'  (BY ',I2,')'/)                       SEC04070
            C                                                                       SEC04080
            C***********************************************************************SEC04090
            C                                                                       SEC04100
            C CLASSIFY THE REGION.                                                  SEC04110
            C                                                                       SEC04120
            C***********************************************************************SEC04130
            C                                                                       SEC04140
0205              CALL SECHO1(ARRAY(COVAR3),ARRAY(AVAR3),ARRAY(SCRAR3),             SEC04150
                 & ARRAY(DETAR3),ARRAY(CONAR3),                                     SEC04160
                 & ARRAY(BDATA),  ARRAY(RDATA),  ARRAY(PIXCOR), ARRAY(PIXVAL),      SEC04170
                 & ARRAY(CELCOR), ARRAY(CELSUM), ARRAY(CELIKE), ARRAY(CELCLS),      SEC04180
                 & ARRAY(PIXCLS), ARRAY(STACK), ARRAY(OPEN), ARRAY(CLOSED),         SEC04190
                 & ARRAY(FLDSIZ), ARRAY(FLDCLS), ARRAY(FLDLIK), ARRAY(PRTBUF),      SEC04200
                 & BUFFER(1))                                                       SEC04210
0206              IF (.NOT.PHASE1.AND.PHASE2) GO TO 458                             SEC04220
0207              GO TO 425                                                         SEC04230
            C                                                                       SEC04240
            C END OF FILE RECORD                                                    SEC04250
            C                                                                       SEC04260
0208          530 PTS = 0                                                           SEC04270
0209              PREFIX(1) = 8                                                     SEC04280
0210              PREFIX(2) = 0                                                     SEC04290
0211              WRITE (OUTPUT,ERR= 536) PREFIX, PTS, LINES, INFO, ID, CSET3       SEC04300
0212              WRITE(PRNTR,9535)                                                 SEC04310
0213         9535 FORMAT('0EOF RECORD WRITTEN.')                                    SEC04320
0214              GO TO 537                                                         SEC04330
0215          536 WRITE(PRNTR,9536)                                                 SEC04340
0216         9536 FORMAT('0ERROR WHILE WRITING EOF RECORD.')                        SEC04350
0217          537 IF(OUTPUT .EQ. CLASSR) RETURN                                     SEC04360
            C                                                                       SEC04370
            C***********************************************************************SEC04380
            C                                                                       SEC04390
            C FINISH TAPE                                                           SEC04400
            C                                                                       SEC04410
            C***********************************************************************SEC04420
            C                                                                       SEC04430
0218              CALL TOPEF (OUTPUT)                                               SEC04440
0219          540 CALL TOPRF (OUTPUT)                                               SEC04450
0220              I0 = 0                                                            SEC04460
0221              I3 = 3                                                            SEC04470
0222              K = -1                                                            SEC04480
0223              FILESV = FILESV+1                                                 SEC04490
0224              PREFIX(1) = 1                                                     SEC04500
0225              PREFIX(2) = 0                                                     SEC04510
0226              WRITE (OUTPUT) PREFIX, RQTAPE, FILESV, I3, K, (I0,I=1,8)          SEC04520
0227              CALL TOPEF (OUTPUT)                                               SEC04530
0228              CALL TOPRF (OUTPUT)                                               SEC04540
0229              CALL TOPEF (OUTPUT)                                               SEC04550
0230              CALL TOPBF (OUTPUT)                                               SEC04560
```

FILE SECINT

```
               CALL TOPBF (OUTPUT)                                                    SEC04570
         8900 RETURN                                                                  SEC04580
         9000 WRITE(TYPEWR,99000)                                                     SEC04590
              WRITE (PRNTR,99C00)                                                     SEC0460C
        99000 FORMAT(' E***  END OF RECORD READING INTERMEDIATE TAPE -')              SEC04610
              GO TO 9900                                                              SEC04620
         9100 WRITE(TYPEWR,99100)                                                     SEC04630
              WRITE (PRNTR,99100)                                                     SEC04640
        99100 FORMAT(' E***  UNEXPECTED END OF FILE READING STATS FROM DISK -')       SEC04650
         9900 WRITE (TYPEWR,99900)                                                    SEC04660
              WRITE (PRNTR,99900)                                                     SEC04670
        99900 FORMAT('         FUNCTION TERMINATED   (SUPINT)')                        SEC04680
              CALL RTMAIN                                                             SEC04690
              END                                                                     SEC04700
```

LARS Program Abstract ________

MODULE IDENTIFICATION

Module Name: MTAPE   Function Name: ECHO FUNCTIONAL SUPPORT

Purpose: Mounts and positions results or intermediate tapes

System/Language: CMS/FORTRAN

Author: P. D. Alenduff   Date: 4/8/77

Latest Revisor: ________   Date: ________

MODULE ABSTRACT

MTAPE mounts and positions the results or intermediate tape (or a tape to be used as output for copying results files).

1. Module Usage

MTAPE

CALL MTAPE (RQTAPE, RQFILE, MODE)

Input Arguments:

RQTAPE — I*4 Tape number of requested tape. A tape number of 0 is a request for a scratch tape.

RQFILE — I*4 File number of requested file. If RQFILE is = 0, then the tape will be initialized by writing a record type 1 on the results tape with file-type = 0.

MODE — I*4 Flag indicating usage of MMTAPE. MODE = -1 indicates MMTAPE has been called to mount and position a tape to be used for copying results files onto. Mode = 0 indicates that a results tape is being mounted for reading a results file. In this case, the tape is mounted ring out. Also, if MODE = 0, RQFILE = 0 is invalid and will cause an error when an attempt is made to write on the tape. MODE = 1 indicates a tape is being mounted for writing a new results file (or continuing a suspended classification). MODE = 2 indicates a tape is to be used as an intermediate tape for reading an intermediate results file. The difference between MODE = -1 and MODE = +1 is the DSRN used for the tape. For MODE = -1 and -2 DSRN is CPYOUT and for MODE = +1, DSRN is MAPTAP. (DSRN is MAPTAP for MODE = 0).

Output Arguments:

RQTAPE I*4 When MODE = 0, set to -1 if requested tape file was full and user decided to use disk for results. Otherwise, remains unchanged.

RQFILE I*4 When MODE = 1, set to -1 if requested tape file was full and user decided to use disk for results. Otherwise, sends back current file position of tape.

MTAPE checks the validity of the tape by reading the record type 1 from the tape and verifying the tape and file number as well as checking fro the correct type of file. Any attempt to overwirte an existing file causes MTAPE to ask the user (via the typewriter) if he wishes to overwrite the file, respecify a new results card, or terminate the function. Note, however, that if a request has been made to initialize a tape, no checking is performed on previous contents.

2. Internal Description

See output description.

3. Input Description

The record type 1 of the results tape is read for each file up to and including the file needed. That is, if file 4 is requested the record type 1 is read from files 1-4.

4. Output Description

The following information messages are issued under the circumstances listed. The term filetype means the filetype code from record type 1 of a results file (the program uses variable CHECK for this number).

I0041 is typed when a tape has been mounted and before MTAPE positions it. This message is not typed when the tape is being initialized or when the correct type number was already mounted.

I0043 is typed when MODE = ±1 and file type of the requested file = 0.

I0044 is typed when MODE = ±1 and filetype of the requested file = 1 and the restart flag from GLOCOM (RESTRT) is not = 1.

I0045 is typed when the tape is correctly positioned. This is not typed when initializing a tape.

After I0043 and I0044, the user is asked whether he wishes to overwrite the file, respecify a new results card with a new tape and/or file or disk option, or terminate the function.

I0100 is typed to allow entry of the new results card. This occurs when the user requests to respecify the results card.

I0101 is typed to confirm usage of disk for results and occurs whenever disk is specified on the results card.

The following error messages are typed under the conditions listed.

E361 is written when the tape is being filed forward and a file is encountered with filetype other than zero before the requested file is reached and MODE = 0.

E362 is written when the circumstances for E361 occurs and MODE = 1. It is also written when MODE = 1 and the filetype of the file requested is = -1.

E363 is written if the RESTRT flag is = 1 and the filetype of the requested file is not = 1.

E364 is written when MODE = 1 and the filetype of the file requested = 1.

E365 is written when an EOF is read on the results file. This should never occur with valid results files.

For message texts refer to the User's Manual.

5. Supplemental Information

This section deals with the handling of tapes by MMTAPE.

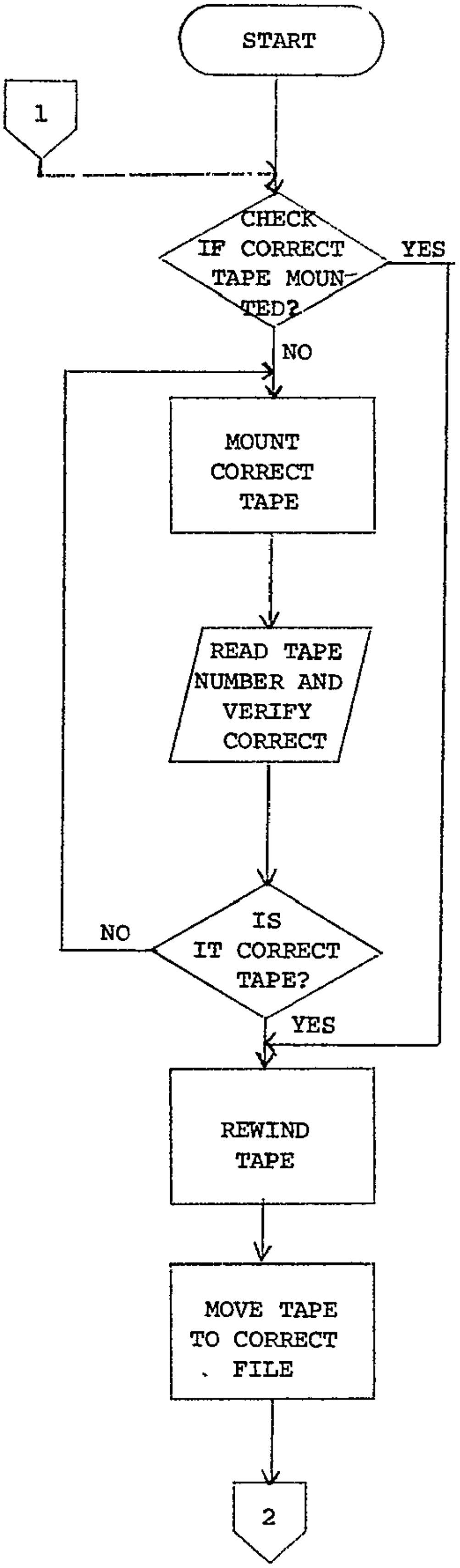
Input:

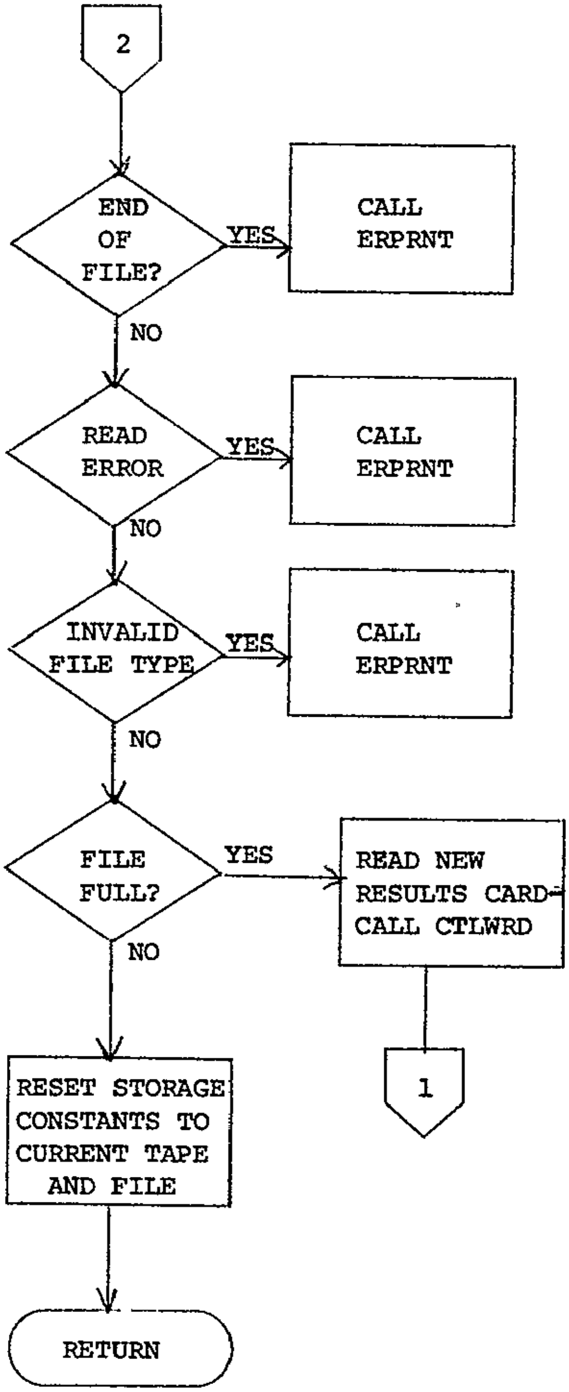
If a tape is mounted on the device and it is the incorrect tape number (as noted from the appropriate status words in GLOCOM), TOPRU is called to unload the tape before the correct tape is mounted. If the correct tape is mounted, MTAPE will check for the ring in if MODE = ±1. If the ring is not in, the tape is unloaded and MOUNT is called to mount the tape with the ring in. If the correct tape is mounted, MTAPE assumes that the file number (as recorded in GLOCOM) is correct and moves the tape backwards or forwards to find the requested file.

Output:

The tape is mounted with ring in for MODE = ±1 and with ring out for MODE = 0 and -2.

The tape is left positioned at the beginning of the requested file. When the tape is initialized a TOPRW is used to do this.


START
1
CHECK IF CORRECT TAPE MOUNTED?
YES
NO
MOUNT CORRECT TAPE
READ TAPE NUMBER AND VERIFY CORRECT
IS IT CORRECT TAPE?
NO
YES
REWIND TAPE
MOVE TAPE TO CORRECT FILE
2


2
END OF FILE?
YES
CALL ERPRNT
NO
READ ERROR
YES
CALL ERPRNT
NO
INVALID FILE TYPE
YES
CALL ERPRNT
NO
FILE FULL?
YES
READ NEW RESULTS CARD-CALL CTLWRD
NO
1
RESET STORAGE CONSTANTS TO CURRENT TAPE AND FILE
RETURN

```
  FORTRAN IV G LEVEL  20.7              MTAPE             DATE = 77129        15025011            PAGE 0001
FILE MTAPE

                  C        MTAPE    LARS  0000
                  C
                  C*****************************************************************
                  C
                  C MTAPE      MOUNTS AND POSITIONS RESULTS TAPES
                  C
                  C*****************************************************************
                  C
  0001                  SUBROUTINE MTAPE (RQTAPE,RQFILE,MODE)
                  C
  0002                  IMPLICIT INTEGER * 4 (A-Z)
  0003                  COMMON /GLCCOM/ BLANK, CARD(20), CHKOUT, COPFIL, CLASSR, CLASSX,
                       1     CLUSIX, CONPUT, CPYOUT, CRDROR, CRDSEQ, DATAPE,
                       2     DUPLTP, DUPRUN, ERRMSG, FBPNT,
                       3     FILESV, FLDBND, HDATA, HEAD(88), ID(200), IMAGEX,
                       4     IMARK, KEYBD, MAPTAP, MAXCHA, MAXCLS,
                       5     PAGSIZ, PNCH, POINT, PRESUX, PRNTR, READIN,
                       6     RESTRT, RUNFIL, RUNTAB(10,3),
                       7     SDATA, SEPARX, SEPTPX, SPARE(10), TEMPAS(30),
                       8     TPSTAT(6), TTFLDX ,TYPEWR,
                       9     TOP, ARRAY(12500)
                  C
  0004                  REAL * 8 ARRAY
  0005                  REAL * 4 FRQCAL(5,30)
  0006                  INTEGER * 4 COMENT(16), DATE(5), HED1(16), HED2(16), TIME(5)
  0007                  INTEGER * 2 BLANK2
  0008                  LOGICAL * 4 CHKOUT
  0009                  LOGICAL * 1 BLANK1
  0010                  EQUIVALENCE (DATSAV,ID(1)), (CURRUN,ID(3)), (FRQCAL(1),ID(51)),
                       1          (HED1(1),HEAD(8)), (DATE(1),HEAD(26)), (HED2(1),HEAD(39)),
                       2          (TIME(1),HEAD(58)), (COMENT(1),HEAD(72)),
                       3          (MAPSAV,TPSTAT(1)),
                       4          (SEPSCR,TPSTAT(2)), (DUPIN,TPSTAT(3)), (DASTAT,TPSTAT(4)),
                       5          (COPSER,TPSTAT(5)), (TRAOLT,TPSTAT(6)),
                       6          (BLANK,BLANK2,BLANK1)
                  C
                  C
  0011                  INTEGER * 4 PREFIX(2), RESCOD(3), CLACOD(4)
  0012                  DATA PREFIX/0,1/, RESULT/'RESU'/, RESCOD/'CVER','RESP','TERM'/,
                       1     CLACOD/'TAPE','FILE','INIT','DISK'/
                  C
                  C*****************************************************************
                  C
                  C ANS     = REPLY FROM USER AS TO WHETHER TO CONTINUE WITH RUN
                  C CHECK   = FLAG NOTING TYPE OF FILE ON RESULTS TAPE
                  C           0 = RESULTS FILE, -1 = CHECK RECORD (WRITTEN AFTER LAST
                  C           RESULTS OF RESTART FILE ON TAPE), 1 = RESTART FILE
                  C ERROR   = ERROR CODE FROM CPFUNC (NOT USED)
                  C I1      = CONSTANT USED WRITING INITIAL RECORD ONTO TAPE
                  C I3      = CONSTANT USED WRITING INITIAL RECORD ONTO TAPE
                  C K       = USED AS DUMMY READING VERSION NUMBER FROM TAPE
                  C MODE    = INDICATES FUNCTION OF MMTAPE, SUPPLIED FROM CALLER
                  C           = 1 MEANS WISH TO GET TAPE READY FOR WRITING CLASSIFICATION
                  C           RESULTS, = 0 MEANS WISH TO GET TAPE READY FOR PRINTING
                  C           RESULTS, = -1 MEANS MOUNT TAPE FOR COPYING CLASSIFICATION
                  C           RESULTS ONTO.
                  C RQFILE = REQUESTED FILE NUMBER IN TAPE TO BE READIED. IF RQFILE
                  C           IS INPUT AS A 0, THEN MMTAPE IS TO WRITE AN INITIAL
                  C           RECORD ONTO THE TAPE.
  0013                  CONTINLE
                  C RQTAPE = REQUESTED TAPE NUMBER TO BE MOUNTED. A ZERO MEANS SCRATCH TAP
                  C           E. IF BOTH RQTAPE AND RQFILE = -1, TAPE HAD RESULTS AND DISK
                  C           WILL BE USED FOR RESULTS.
                  C IFO     = FILE NUMBER AS READ FROM RESULTS TAPE
                  C TNO     = TAPE NUMBER AS READ FROM RESULTS TAPE
                  C UNIT    = DSRN OF TAPE BEING MOUNTED
                  C           THIS WILL BE MAPTAP UNLESS MOUNTING COPYRESULTS OUTPUT TAP
                  C
  0014                1 CONTINLE
                  C*****************************************************************
                  C
  0015                  INITFG = 0
                  C
                  C IF USER REJECTS TAPE/FILE PREVIOUSLY REQUESTED, DISK WILL BE USED
                  C
  0016              100 IF (RQTAPE .EQ. -1 .AND. RQFILE .EQ. -1) RETURN


  FORTRAN IV G LEVEL  20 7              MTAPE C           DATE = 77129        15025011            PAGE 0002
FILE MTAPE

  0017                  UNIT = MAPTAP
  0018                  IF (MODE .LE. -1) GO TO 108
  0019                  FILNO = FILESV
                  C
                  C         IF CORRECT TAPE IS MOUNTED GO TO POSITIONING
                  C
  0020                  IF (MAPSAV  EQ. RQTAPE) GO TO 130
  0021              105 IF (MAPSAV  NE. -1) CALL TOPRU(MAPTAP)
  0022                  GO TO 110
  0023              108 UNIT = CPYOUT
  0024                  FILNO = COPFIL
  0025                  IF (COPSER .NE. -1) CALL TOPRU(CPYOUT)
  0026              110 IF (MODE .EQ. 1) CALL MOUNT (RQTAPE,MAPTAP,'RI')
  0027                  IF (MODE .EQ. 0) CALL MOUNT (RQTAPE,MAPTAP,'RO')
  0028                  IF (MODE .EQ. -1) CALL MOUNT (RQTAPE,CPYOUT,'RI')
  0029                  IF (MODE .EQ. -2) CALL MOUNT (RQTAPE,CPYOUT,'RO')
                  C
                  C         IF CALLED FROM COPYRESULTS, THE UNIT IS COPYOUT
                  C
  0030                  FILNO = 1
                  C
                  C         RQFILE OF 0 MEANS THE CALL IS TO INITIALIZE THE TAPE
                  C
  0031                  IF (RQFILE .NE. 0) GO TO 120
  0032                  I1 = 1
  0033                  I3 = 3
  0034                  WRITE (UNIT) PREFIX, RQTAPE, I1, I3,(RQFILE,I=1,9)
  0035                  CALL TOPRW(UNIT)
  0036                  GO TO 190
  0037              120 WRITE (TYPEWR,9120) RQTAPE, RQFILE
  0038             9120 FORMAT (' I0042  POSITIONING RESULTS TAPE',I5,' TO FILE',
                       1     I4,'     (MTAPE)')
  0039                  CALL TOPRW(UNIT)
                  C
                  C         READ FIRST RECORD AND SEE IF CORRECT TAPE IS MOUNTED
                  C
  0040                  READ (UNIT,ERR=530,END = 500) PREFIX, TNO, IFO, K, CHECK
  0041                  CALL TOPRW(UNIT)
  0042                  IF (TNO  EQ. RQTAPE) GO TO 130
  0043                  CALL CPFUNC (15,'M CP WRONG TAPE',ERROR)
  0044                  CALL TOPRU(UNIT)
  0045                  GO TO 110
                  C
                  C         POSITION RESULTS TAPE TO CORRECT FILE
                  C
  0046              130 IF (RQFILE .GT. 1) GO TO 140
                  C
                  C         IF NEED RINGIN, MAKE SURE IT IS. (TAPE MAY HAVE MOUNTED
                  C         BY PREVIOUS PRINTRESULTS WHICH HAD RING OUT)
                  C
  0047                  IF (MODE .EQ. 0 .OR. MODE .EQ  -2) GO TO 135
  0048                  CALL RINGIN (UNIT,I)
  0049                  IF (I  EQ. 0) GO TO 135
  0050                  CALL TOPRU(UNIT)
  0051                  GO TO 110
  0052              135 FILNO = 1
  0053                  CALL TOPRW(UNIT)
  0054                  GO TO 200
  0055              140 J = RQFILE - FILNO
                  C
                  C         J IS THE NUMBER OF FILES THE TAPE MUST BE MOVED
                  C
  0056                  IF (J) 150, 200, 170
  0057              150 J = -J
                  C
                  C         THE DO 160 LOOP BACKS THE TAPE UP TO THE CORRECT FILE
                  C
  0058                  DO 160 I = 1,J
  0059                  FILNO = FILNO -1
  0060              160 CALL TOPBF(UNIT)
  0061                  GO TO 200
                  C
                  C         THE DO 180 LOOP FORWARD FILES THE TAPE AND CHECKS
                  C         TO BE SURE THAT FILES ARE REALLY THERE TO SPACE OVER
                  C
  0062              170 DO 180 I = 1, J
```

```
 FORTRAN IV G LEVEL  20.7              MTAPE C          DATE = 77129         15025011              PAGE 0003
FILE MTAPE
  0063              READ (LUNIT,ERR=530,END=500) PREFIX, IND, TFO, K, CHECK                 MMT01530
  0064              IF (CHECK .NE. 0) GO TO 210                                             MMT0154C
  0065              FILNO = FILNO + 1                                                       MMT0155C
  0066          180 CALL TCPFF(UNIT)                                                        MMT01560
            C                                                                               MMT01570
            C           IF FILNO AND RQFILE ARE UNEQUAL , THE PROGRAM HAS GOOOFED.          MMT0158C
            C                                                                               MMT0159C
  0067          200 IF (FILNO  NE. RQFILE) CALL ERPRNT (100,'STOP')                         MMT0160C
            C                                                                               MMT01610
            C           NOW READ THE FIRST RECORD OF THE FILF AND CHECK VALIDITY            MMT0162C
            C                                                                               MMT0163C
  0068              READ (LUNIT,ERR=530,END=500)  PREFIX, IND, TFO, K, CHECK                MMT01640
            C                                                                               MMT0165C
            C           CHECK FOR ERRCR OF RESTART OPTION BUT NOT RESTART FILE              MMT0166C
            C                                                                               MMT01670
  0069              IF (RESTRT .EQ  1 .AND. CHECK  NE. 1) CALL ERPRNT (363,'STOP')          MMT01680
  0070              IF (MODE  NE. 0.AND MODE  NE. -2) GO TO 225                             MMT01690
  0071              IF (CHECK  NE  -1) GO TO 220                                            MMT0170C
  0072              I = FILNO - 1                                                           MMT01710
            C                                                                               MMT01720
            C           THE FOLLOWING CODE PROCESSES ERROR FOR INVALID FILE TYPE            MMT0173C
            C                                                                               MMT0174C
  0073          210 CALL ERPRNT (361+IABS(MODE),'GO')                                       MMT01750
  0074              WRITE (PRNTR,9210) I                                                    MMT01760
  0075              WRITE (TYPEWR,9210) I                                                   MMT01770
  0076         9210 FORMAT (10X,'FILE',I3)                                                  MMT0178C
  0077              CALL RTMAIN                                                             MMT0179C
            C                                                                               MMT01800
            C           IF A RESTART FILE,  AND RESTART NCT SPECIFIED, ERRCR                MMT01810
            C                                                                               MMT01820
  0078          220 IF (CHECK  EQ  1) CALL ERPRNT (364,'STOP')                              MMT0183C
  0079              GO TO 300                                                               MMT01840
  0080          225 IF (CHECK  EQ  -1 .AND. RESTRT .EC  0) GO TC 300                        MMT0185C
  0081              IF (RESTRT .NE  1) GO TO 230                                            MMT0186C
  0082              IF (CHECK .EQ  1) GO TO 300                                             MMT01870
            C                                                                               MMT01880
            C           ERROR CF RESTARTING AND NO RESTART FILE                             MMT01890
            C                                                                               MMT0190C
  0083              CALL ERPRNT (260,'STOP')                                                MMT01910
  0084          230 IF (CHECK .NE  0) GO TO 233                                             MMT01920
  0085              WRITE (TYPEWR, 9230)                                                    MMT01930
  0086              WRITE (PRNTR, 9230)                                                     MMT0194C
  0087         9230 FORMAT (' 10043  FILE HAS RESULTS IN IT - DO YOU WISH TC ',             MMT01950
                 1         'OVERWRITE THE FILE, RESPECIFY THE ENTIRE RESULTS CARD,',/ MMT0196C
                 2          ' CR TERMINATE THE JOB (OVERWRITE, RESPECIFY, TERMINATE) 'MMT0197C
                 3        ,'      (MTAPE)')                                                 MMT0198C
  0088          233 IF (CHECK .NE  1) GO TO 235                                             MMT01990
  0089              WRITE (TYPEWR, 9235)                                                    MMT02000
  0090              WRITE (PRNTR, 9235)                                                     MMT0201C
  0091         9235 FORMAT (' 10044  FILE IS A RESTART FILE - CO YOU WISH TC ',             MMT02020
                 1         'OVERWRITE THE FILE, RESPECIFY THE ENTIRE RESULTS CARD,',/  MMT02030
                 2         ' OR TERMINATE THE JOB (OVERWRITE, RESPECIFY, TERMINATE) ', MMT0204C
                 3         '      (MTAPE)')                                                 MMT0205C
  0092          235 COL = C                                                                 MMT0206C
  0093              ERRCOR = 3                                                              MMT02070
  0094              CALL CTLWRD (CARD, CCL, RESCCD, 3, CODE, KEYBD, ERRCOR)                 MMT0208C
  0095              IF (ERRCOR .EQ  4) CALL RTMAIN                                          MMTC209C
  0096              GO TO (300,240,244), CODE                                               MMT0210C
  0097          240 WRITE(TYPEWR,9250)                                                      MMT0211C
  0098              WRITE (KEYBD, 9250)                                                     MMT0212C
  0099         9250 FORMAT (' 10100  ENTER RESULTS CARD       (MMTAPE)')                    MMT0213C
  0100              COL = C                                                                 MMT02140
  0101              ERRCOR = 3                                                              MMT02150
  0102          242 CALL CTLWRC (CARD, COL, RESULT, 1, CODE, KEYBD, ERRCOR)                 MMT0216C
  0103              IF (ERRCOR .EQ. 4) CALL RTMAIN                                          MMT0217C
  0104          243 IF (COL .EC  72) GO TO 100                                              MMT02180
  0105              CALL CTLPRM (CARD, COL, CLACOD, 4, CODE, &245)                          MMT02190
  0106              GO TO (250,260,270,280),CODE                                            MMT0220C
            C                                                                               MMT02210
            C TERMINATE CPTICN                                                              MMT02220
            C                                                                               MMT02230
  0107          244 CALL RTMAIN                                                             MMT02240
            C                                                                               MMT02250
            C RETYPE RESULTS CARD AFTER ERROR IN ENTRY                                      MMT02260
            C                                                                               MMT02270
  0108          245 ERRCOR = 1                                                              MMT02280
```

```
 FORTRAN IV G LEVEL  20 7              MTAPE C          DATE = 77129         15025011              PAGE 0004
FILE MTAPE
  0109              GO TO 242                                                               MMT02290
            C                                                                               MMT02300
            C PROCESS PARAMETER ERRORS                                                      MMT02310
            C                                                                               MMT02320
  0110          248 WRITE (TYPEWR, 9248) CARD                                               MMT02330
  0111         9248 FORMAT (5X,20A4)                                                        MMT0234C
  0112              CALL ERPRNT (ERRNUM, 'GO')                                              MMT02350
  0113              GO TO 242                                                               MMT02360
            C                                                                               MMT02370
            C DECODE TAPE SPECIFICATION                                                     MMT02380
            C                                                                               MMT02390
  0114          250 VECSZ = 1                                                               MMT02400
  0115              CALL IVAL (CARD, COL, RQTAPE, VECSZ, &252)                              MMT02410
  0116              IF (VECSZ .EQ. 0) GO TO 252                                             MMT0242C
  0117              GO TO 243                                                               MMT02430
  0118          252 ERRNUM = 348                                                            MMT02440
  0119              GO TO 248                                                               MMT02450
            C                                                                               MMT02460
            C FILE SPECIFICATION                                                            MMT02470
            C                                                                               MMT02480
  0120          260 VECSZ = 1                                                               MMT02490
  0121              CALL IVAL (CARD, COL, RQFILE, VECSZ, &252)                              MMT02500
  0122              IF (VECSZ  EQ  0) GO TO 252                                             MMT0251C
  0123              GO TO 243                                                               MMT0252C
  0124          265 ERRNUM = 347                                                            MMT0253C
  0125              COL = C                                                                 MMT02540
  0126              ERRCOR = 3                                                              MMT02550
  0127              GO TO 242                                                               MMT02560
            C                                                                               MMT02570
            C INITIALIZE SPECIFICATION                                                      MMT02580
            C                                                                               MMT02590
  0128          270 INITFG = 1                                                              MMT02600
  0129              GO TO 243                                                               MMT02610
            C                                                                               MMT02620
            C DISK SPECIFICATION -- WHEN DISK APPEARS ON CARD, IT OVERRIDES CTHER           MMT02630
            C        PARAMETERS. CANNOT CHECK FCR WHEN BOTH SPECIFIEC ON CARD SINCE         MMT02640
            C        RQTAPE AND RQFILE ARE ALREADY USED (IN MOUNTED TAPE)                   MMT0265C
            C                                                                               MMT02660
  0130          280 RQTAPE = -1                                                             MMT02670
  0131              RQFILE = -1                                                             MMT02680
  0132              WRITE (TYPEWR, 9280)                                                    MMT02690
  0133         9280 FORMAT (' 10101  DISK WILL BE USEC FOR RESULTS      (MMTAPE)')          MMT02700
  0134              GO TO 100                                                               MMT02710
  0135          300 WRITE (TYPEWR,9300)                                                     MMT02720
  0136         930C FORMAT (' 10045  RESULTS TAPE MOUNTED AND POSITIONED',                  MMT0273C
                 1          '      (MMTAPE)')                                               MMT02740
  0137              CALL TCPRF(UNIT)                                                        MMT02750
  0138          390 IF (MODE .LE. -1) GO TO 400                                             MMT0276C
  0139              PAPSAV = RCTAPE                                                         MMT0277C
  0140              FILESV = FILNO                                                          MMT02780
  0141              RETURN                                                                  MMT02790
  0142          400 COPSER = RCTAPE                                                         MMT02800
  0143              COPFIL = FILNO                                                          MMT0281C
  0144              RETURN                                                                  MMT02820
  0145          500 CALL ERPRNT (365,'STOP')                                                MMT02830
  0146          510 REWIND KEYBD                                                            MMT02840
  0147              GO TG 280                                                               MMT02850
  0148          530 CALL ERPRNT (261,'STOP')                                                MMT02860
  0149              END                                                                     MMT0287C
```

LARS Program Abstract ______

MODULE IDENTIFICATION

Module Name: SECH01 Function Name: SECSUP

Purpose: Performs Supervised ECHO Processing on area.

System/Language:

Author: P. D. Alenduff Date: 4/11/77

Latest Revisor: Date:

MODULE ABSTRACT

This routine is called once for each area to be classified and performs all classification on each line of data and produces the output file.

1. Module Usage

SECH01

```
CALL SECH01 (ICLSCV, VECPRD, SCAPRD, DETCOV, CONST, BDATA,
             RDATA, PIXCOR, PIXVAL, CELCOR, CELSUM, CELIKE,
             CELCLS, PIXCLS, STACK, OPEN, CLOSED, FLDSIZ,
             FLDLIK, PRTBUF, BUFFER)
```

Input Arguments:

ICLSCV (VARSZ3, NOPOOL) — R*4 An array dimensioned (channels * (channels + 1)/2) * number of classes, containing the upper half of the triangular covariance matrices for each of the classes used for classification.

VECPRD (NOFET3, NOPOOL) — R*4 An array dimensioned channels * classes containing the vector product of the classes for each channel from the inverse covariance matrix and the determinant of the covariance matrix.

SCAPRD (NOPOOL) — R*4 An array dimensioned "classes" for the scalar product of the mean and covariance matrices for each class.

DETCOV (NOPOOL) — R*4 An array dimensioned "classes" words for the determinant of the covariance matrix for each class.

CONST (NOPOOL) — R*4 An array dimensioned "classes" words containing the constant term used consisting of an expression with determinant and scalar product for each class.

BDATA ((NOFET3*ID(6))/2) — I*2 The array used for reading the byte information of data values from data tape.

RDATA (NVR, CELWTH) — R*4 The array dimensioned "(ID(6)-6) * cell width" words for reading actual data values from tape and calibrating them.

PIXCOR (VARSZ3 * CELSIZ) — R*4 The array dimensioned (channels * (channels + 1)/2) by (cell width)^2 words for storing the cross product terms of the data values of a cell (correlation term).

PIXVAL (NOFET3 *CELSIZ) — R*4 An array dimensioned "channels * (cell width)^2" for storing actual pixel data values for each channel for 1 cell.

CELCOR (VARSZ3) — R*4 An array dimensioned (channels * (channels + 1))/2 words for saving the cross product of the pixel values.

CELSUM (NOFET3) — R*4 An array dimensioned "channels" words used to store the sum of data values for each pixel in the cell for a given channel number.

CELIKE (NWORD, JPTS) — R*4 An array dimensioned "NWORD by columns of cells" for storing the cell likelihoods or pixel values for each cell in the line.

CELCLS (JPTS) I*2 An array dimensioned "column" of cells halfwords used to contain the most likely class number of all cells in a given row.

PIXCLS (NWORD, JPTS) I*2 The array dimensioned NWORD2 by "column" halfwords for storing the class number of each pixel in a given cell for singular cells (equivalent to CELIKE).

STACK (JPTS) I*2 A stack of field numbers for the cell in the given position for a row of cells.

OPEN (JPTS) L*1 An array of "column" bytes used as logical flags to store whether or not a particular column of cells has been opened as a field.

CLOSED (JPTS) L*1 An array of logical values for each column of cells to indicate whether the field for each column has been opened for annexation.

FLDSIZ (JPTS) I*4 The array of "column" halfwords to store the length of the current field.

FLDCLS (JPTS) I*2 The array of "column" halfwords to store the length of the current fields as the number of previously annexed rows in this column.

FLDLIK (NOPOOL, JPTS) R*4 The array of likelihoods dimensioned columns of cells by classes words used to store the likelihood of each field for each class.

PRTBUF (JPTS) I*2 The output print buffer dimensioned "columns of cells" halfwords.

BUFFER (BUFROZ * JPTS) I*2 The buffer array used for annexation of neighboring rows of cells dimensioned BUFROZ * JPTS.

SECH01 performs processing of an area to be classified. If only an intermediate tape is to be produced, cell information is gathered and written to tape. Otherwise all information is used in annexation and classification, and the classification results are written to the output device. Any maps are also produced.

2. Internal Description

Initially, column headers are printed if any map is to be produced. All buffer pointers and cell arrays are initialized so that the arrays OPEN and CLOSE indicate that no annexation and classification has taken place. Then a loop is entered and executed once for each row of cells (two or more data lines). The routine FILBUF is called to gather statistics on one row of cells, either directly from processing the data tape, or from reading the intermediate tape. If no annexation is to be performed, the information is written to the output device and this loop is executed until this entire area is completed and a return ıs executed. Otherwise, a series of calls to LIKRAT for a comparison of the likelihood of the cell being annexed to the cell above, left, and right is performed. The cell is annexed to the most likely cell if the comparison with the annexation threshold is successful. This is continued for all cells in the row. When finished, any fields formed from adjoining cells that are closed and no longer continued are classified and then any new fields are opened in that statistics from the initial cell are copied into the array storing field statistics. Then LIKRAT is called to classify closed areas. A print buffer is then filled with symbols for either the cell map or classification amp and printed out. Statistics are handled for remaining open fields and the row is written to the output tape. This is continued until the area has been processed.

3. Input Description

Not applicable

4. Output Description

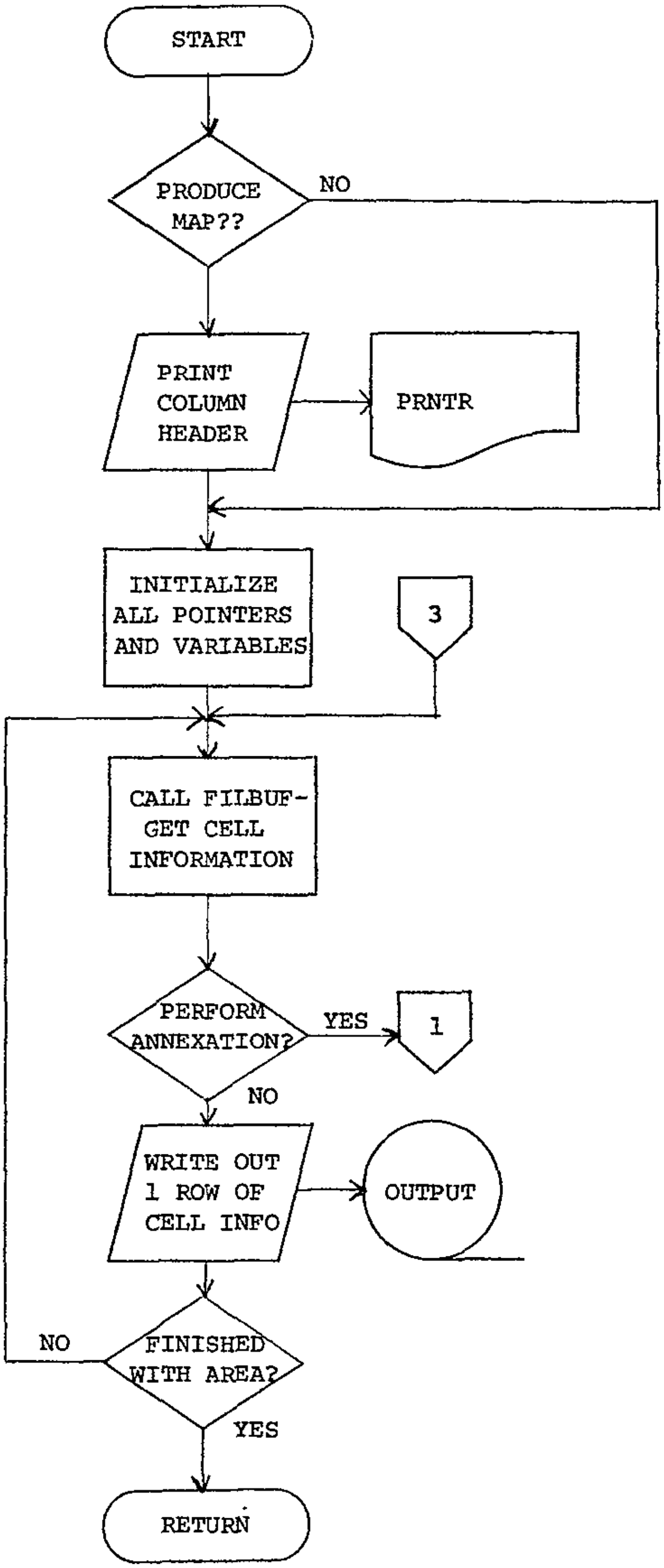
The unit PRNTR receives the map and column header produced if a map is to be printed.

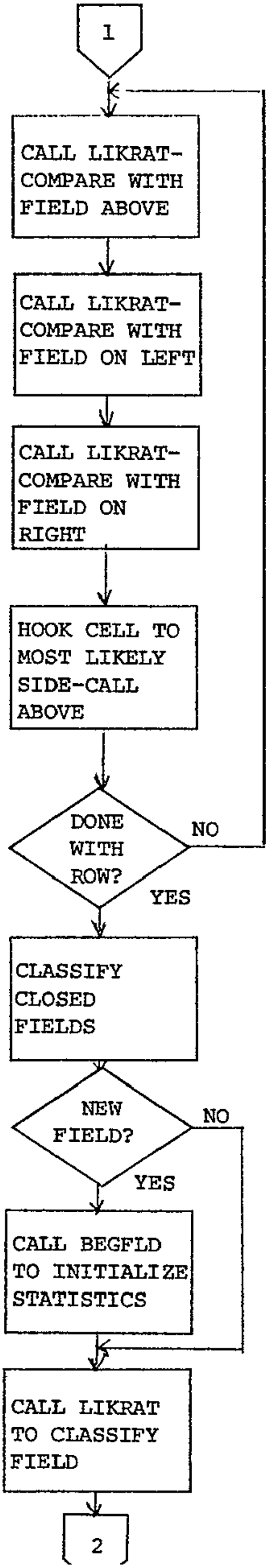
The unit OUTPUT is used as the output unit for either intermediate tape - cell information or classified cell lines. These lınes follow the format of regular classification records.

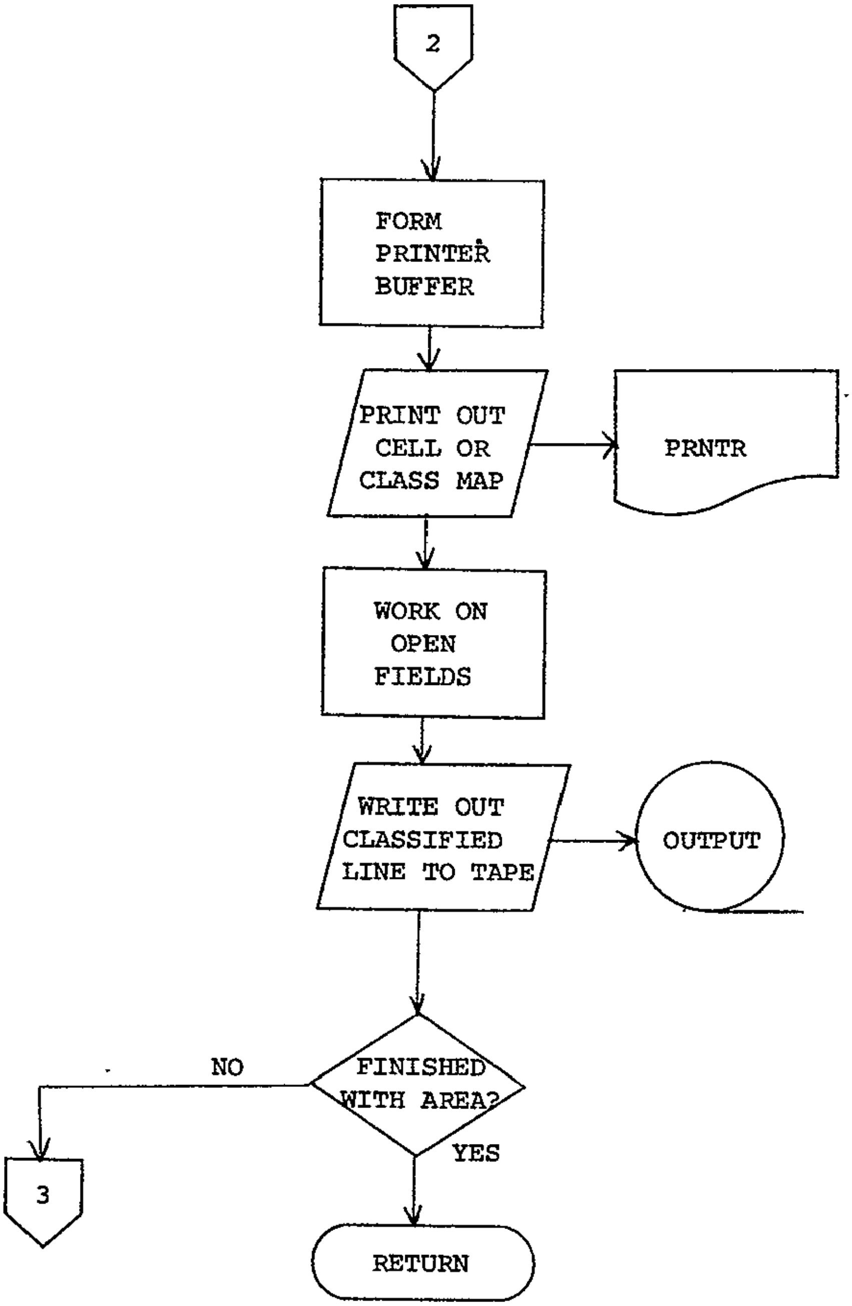
5. Supplemental Information

This program uses common blocks GLOCOM and SECCOM.

6. FLOWCHART


START
PRODUCE MAP??
NO
PRINT COLUMN HEADER
PRNTR
INITIALIZE ALL POINTERS AND VARIABLES
3
CALL FILBUF-GET CELL INFORMATION
PERFORM ANNEXATION?
YES
1
NO
WRITE OUT 1 ROW OF CELL INFO
OUTPUT
NO
FINISHED WITH AREA?
YES
RETURN


1
CALL LIKRAT-COMPARE WITH FIELD ABOVE
CALL LIKRAT-COMPARE WITH FIELD ON LEFT
CALL LIKRAT-COMPARE WITH FIELD ON RIGHT
HOOK CELL TO MOST LIKELY SIDE-CALL ABOVE
DONE WITH ROW?
NO
YES
CLASSIFY CLOSED FIELDS
NEW FIELD?
NO
YES
CALL BEGFLD TO INITIALIZE STATISTICS
CALL LIKRAT TO CLASSIFY FIELD
2

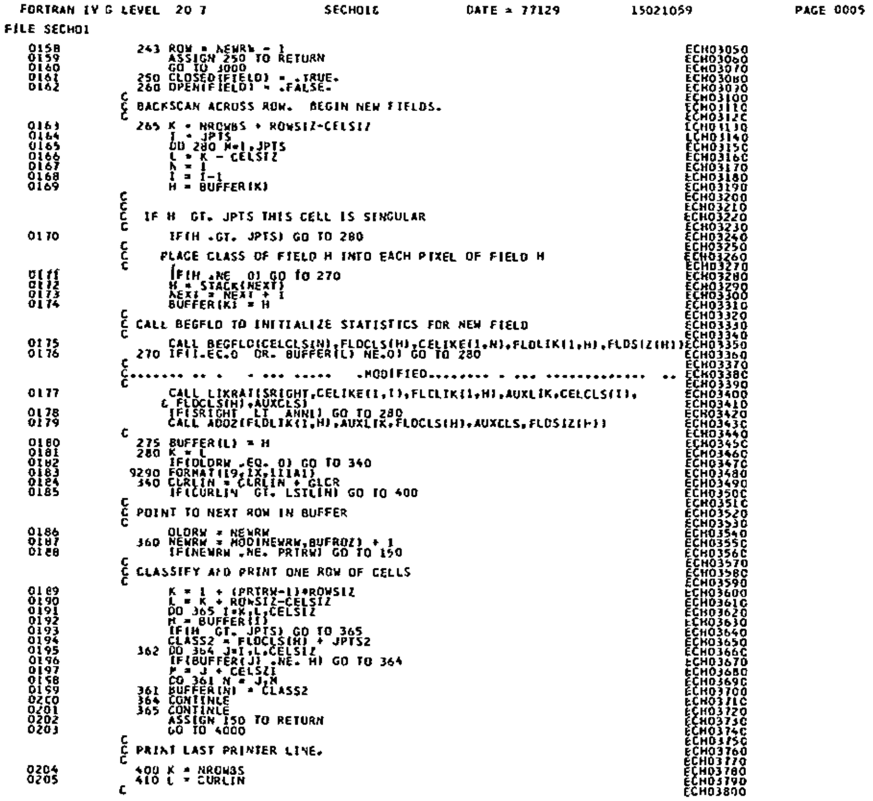

2
FORM PRINTER BUFFER
PRINT OUT CELL OR CLASS MAP
PRNTR
WORK ON OPEN FIELDS
WRITE OUT CLASSIFIED LINE TO TAPE
OUTPUT
FINISHED WITH AREA?
NO
YES
3
RETURN

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

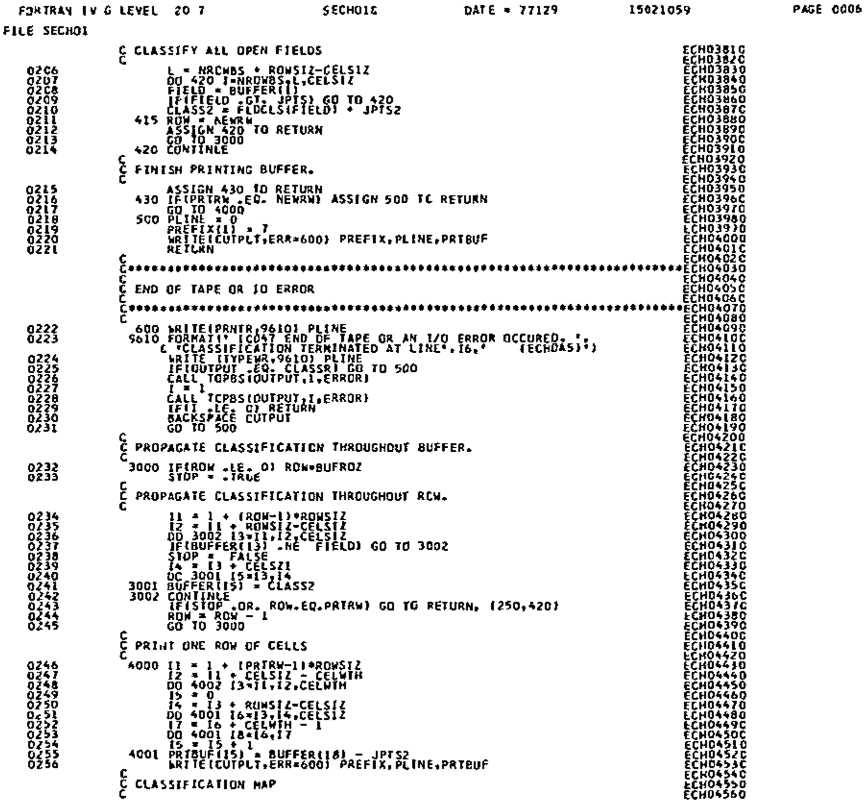
```
FORTRAN IV G LEVEL  20.7              SECH01            DATE = 77129         15021059            PAGE 0001
FILE SECH01

               C     SECH01     LARS  XXXX                                              ECH00010
               C                                                                        ECH00020
               C SPECIAL VERSION TO IMPLEMENT EXTRA SCANNING LOGIC.                     ECH00030
               C***********************************************************************ECH00040
               C                                                                        ECH00050
  0001               SUBROUTINE SECH01(ICLSCV,VECPRD,SCAPRD,DETCOV,CONST,              ECH00060
                    1 BDATA,RDATA,PIXCOR,PIXVAL,CELCOR,CELSUM,                          ECH00070
                    2 CELIKE,CELCLS,PIXCLS,STACK,OPEN,CLOSED,FLDSIZ,FLDCLS,             ECH00080
                    3 FLDLIK,PRTBUF,BUFFER)                                             ECH00090
  0002               IMPLICIT INTEGER * 4 (A-Z)                                         ECH00100
               C                                                                        ECH00110
  0003               COMMON /GLCCOM/ BLANK, CARD(20), CHKOUT, COPFIL, CLASSR, CLASSX,   ECH00120
                    1    CLUSTX, CONPUT, CPYOUT, CRDRDR, CRDSEG, DATAPE,                ECH00130
                    2    DUPLTP, DUPRUN, ERRMSG, FBPNT,                                 ECH00140
                    3    FILESV, FLDBND, HDATA, HEAD(88), ID(200), IMAGEX,              ECH00150
                    4    IMARK, KEYBD, MAPTAP, MAXCHA, MAXCLS,                          ECH00160
                    5    PAGSIZ, PNCH, POINT, PRESUX, PRNTR, READIN,                    ECH00170
                    6    RESTRT, RUNFIL, RUNTAB(10,3),                                  ECH00180
                    7    SDATA, SEPARX, SEPTPX, SPARE(10), TEMPAS(30),                  ECH00190
                    8    TPSTAT(6), TTFLDX ,TYPEWR,                                     ECH00200
                    9    TOP, ARRAY(2500)                                               ECH00210
               C                                                                        ECH00220
  0004               REAL * 8 ARRAY                                                     ECH00230
  0005               REAL * 4 FROCAL(5,30)                                              ECH00240
  0006               INTEGER * 4 COMENT(16), DATE(5), HED1(16), HED2(16), TIME(5)       ECH00250
  0007               INTEGER * 2 BLANK2                                                 ECH00260
  0008               LOGICAL * 4 CHKOUT                                                 ECH00270
  0009               LOGICAL * 1 BLANK1                                                 ECH00280
  0010               EQUIVALENCE (DATSAV,ID(1)), (CURRUN,ID(3)), (FRCCAL(1),ID(51)),    ECH00290
                    1          (HED1(1),HEAD(8)), (DATE(1),HEAD(26)), (HED2(1),HEAD(39)),ECH00300
                    2          (TIME(1),HEAD(58)), (COMENT(1),HEAD(72)),                ECH00310
                    3          (MAPSAV,TPSTAT(1)),                                      ECH00320
                    4          (SEPSCR,TPSTAT(2)), (DUPIN,TPSTAT(3)), (DASTAT,TPSTAT(4)),ECH00330
                    5          (CCPSER,TPSTAT(5)), (TRAOUT,TPSTAT(6)),                  ECH00340
                    6          (BLANK,BLANK2,BLANK1)                                    ECH00350
               C                                                                        ECH00360
               C                                                                        ECH00370
  0011               COMMON /SECCOM/ BUFADZ,CSELEC,CSET(3,30),CSET3(3,30),THRES,CELSIZ,ECH00380
                    1 CELWTH,INFO(17),INPUT, INTTAP, INTFIL, JPTS, LINES, NOCLS, NOFET,ECH00390
                    2 NOFET3,NOPOOL,NWORD,NWORD2, NVR, OUTPUT, POLNAM(2,60), PREFIX(2),ECH00400
                    3 PTS, ROTAPE, ROFILE, SYMCNT, VARSZ3, CSEL(30), CSEL3(30),         ECH00410
                    3 FETVEC(30), FETVC3(30), POLPTR(2,60), POLSTK(60),                 ECH00420
                    4 COFLAG, CLSMAP, CSET1(3,30), OBJMAP, PHASE1, PHASE2, POLNM1(60),  ECH00430
                    5 PRSTAT, SYM(60), SYMNTX(60)                                       ECH00440
               C                                                                        ECH00450
  0012               REAL * 4 CSELEC, CSET, CSET3, THRES, ANN1                          ECH00460
               C                                                                        ECH00470
  0013               INTEGER * 2 CSEL, CSEL3, FETVEC, FETVC3, POLPTR, POLSTK            ECH00480
               C                                                                        ECH00490
  0014               LOGICAL * 1 COFLAG, CLSMAP, CSET1, OBJMAP, PHASE1, PHASE2, POLNM1, ECH00500
                    1 PRSTAT                                                            ECH00510
               C                                                                        ECH00520
               C                                                                        ECH00530
               C                                                                        ECH00540
               C***********************************************************************ECH00550
               C                                                                        ECH00560
               C DEFINE LOCAL VARIABLES                                                 ECH00570
               C                                                                        ECH00580
               C***********************************************************************ECH00590
  0015               REAL * 4 RDATA(NVR,CELWTH),CELIKE(NWORD,JPTS),FLDLIK(NCPOOL,JPTS),ECH00600
                    1 ICLSCV(VARSZ3,NOPOOL),VECPRD(NOFET3,NOPOOL),SCAPRD(NOPOOL),       ECH00610
                    2DETCOV(NOPOOL),CONST(NOPOOL),CELSUM(NOFET3),CELCOR(VARSZ3),        ECH00620
                    3 PIXVAL(1),PIXCOR(1),AUXLIK(60,4),SUP,SRIGHT,SLEFT,SUPLUS,ANN1     ECH00630
               C                                                                        ECH00640
  0016               INTEGER * 4 FLDSIZ(JPTS)                                           ECH00650
               C                                                                        ECH00660
  0017               INTEGER * 2 CELCLS(JPTS),PIXCLS(NWORD2,JPTS),BDATA(1),STACK(JPTS),ECH00670
                    1 BUFFER(1), FLDCLS(JPTS), PRTBUF(PTS),BLOCK(4), MZ, CLASSZ,       ECH00680
                    2 JPTS2, SINGUL, VERTI, HORIZ, PLUS, L2, BLANK2, AUXCLS(4),HSYM(1) ECH00690
               C                                                                        ECH00700
  0018               LOGICAL * 1 OPEN(JPTS),CLOSED(JPTS),STOP                           ECH00710
               C                                                                        ECH00720
  0019               EQUIVALENCE (FSTLIN,INFO(4)), (LSTLIN,INFO(5)), (LININT,INFO(6))   ECH00730
  0020               EQUIVALENCE (CURLIN,INFO(4)),(SYM(1),HSYM(1))                      ECH00740
  0021               DATA SINGUL,VERTI,HORIZ,PLUS,L2/'O','I','-','/','L'/               ECH00750
               C                                                                        ECH00760
```

```
FORTRAN IV G LEVEL  20.7              SECH01            DATE = 77129         15021059            PAGE 0002
FILE SECH01

  0022               ASSIGN 228 TO ALPHA                                                ECH00770
  0023               ROWSIZ = JPTS * CELSIZ                                             ECH00780
               C                                                                        ECH00790
               C TURN OFF COLUMN HEADER IF NO PHASE2 PROCESSING                         ECH00800
               C                                                                        ECH00810
  0024               IF (.NOT. PHASE2) GO TO 90                                         ECH00820
               C                                                                        ECH00830
               C TRANSFORM ANNEXATION PARAMETER                                         ECH00840
               C                                                                        ECH00850
  0025               ANN1 = -ALOG(10.)*THRES                                            ECH00860
               C                                                                        ECH00870
               C                                                                        ECH00880
               C PRINT COLUMN NUMBERS ON OUTPUTV                                        ECH00890
               C                                                                        ECH00900
  0026               N = INFO(9)*CELWTH                                                 ECH00910
  0027               IF (.NOT. CLSMAP .AND. .NOT. OBJMAP) GO TO 90                      ECH00920
  0028               L = INFO(7)                                                        ECH00930
  0029               INTVL = (JPTS + 110) / 111                                         ECH00940
  0030               JPTS1 = JPTS / INTVL                                               ECH00950
  0031               M = L + (JPTS1-1)*N                                                ECH00960
  0032               I1 = 1                                                             ECH00970
  0033               IF(M .GT. 999) I1=10                                               ECH00980
  0034               I = L                                                              ECH00990
  0035               DO 30 K=1,JPTS1                                                    ECH01000
  0036               PRTBUF(K) = I/(100*I1)                                             ECH01010
  0037               PRTBUF(K+JPTS1) = MOD(I,100*I1)/(10*I1)                            ECH01020
  0038               PRTBUF(K+2*JPTS1) = MOD(I,10*I1)/I1                                ECH01030
  0039            30 I = I+N                                                            ECH01040
  0040               I1 = 2*JPTS1+1                                                     ECH01050
  0041               DO 40 I=1,I1,JPTS1                                                 ECH01060
  0042               J = I+JPTS1-1                                                      ECH01070
  0043            40 WRITE(PRNTR,9040) (PRTBUF(K),K=I,J)                                ECH01080
  0044          9040 FORMAT(' ',8X,111I1)                                               ECH01090
  0045               IF(M .LE. 999) GO TO 50                                            ECH01100
  0046               I = L                                                              ECH01110
  0047               DO 45 J=1,JPTS1                                                    ECH01120
  0048               PRTBUF(J) = MOD(I,10)                                              ECH01130
  0049            45 I = I+N                                                            ECH01140
  0050               WRITE(PRNTR,9040) (PRTBUF(I),I=1,JPTS1)                            ECH01150
  0051            50 WRITE(PRNTR,9120)                                                  ECH01160
  0052          9120 FORMAT('0')                                                        ECH01170
               C                                                                        ECH01180
               C***********************************************************************ECH01190
               C                                                                        ECH01200
               C INITIALIZE                                                             ECH01210
               C                                                                        ECH01220
               C***********************************************************************ECH01230
  0053            90 JPTS2 = JPTS                                                       ECH01240
               C                                                                        ECH01250
               C SKIP ZEROING OUT ARRAYS IF ONLY PHASE1 (THEY'RE NOT ALLOCATED---)      ECH01260
               C                                                                        ECH01270
  0054           100 IF (.NOT. PHASE2) GO TO 105                                        ECH01280
  0055               DO 110 I = 1,JPTS                                                  ECH01290
  0056               STACK(I) = I                                                       ECH01300
  0057               CLOSED(I) = .TRUE.                                                 ECH01310
  0058           110 OPEN(I) = .FALSE.                                                  ECH01320
  0059           105 GLCR = LININT*CELWTH                                               ECH01330
  0060               CELSZ1 = CELSIZ - 1                                                ECH01340
  0061               NOLINE = 0                                                         ECH01350
  0062               PLINE = FSTLIN                                                     ECH01360
  0063               NEXT = 1                                                           ECH01370
  0064               PRTRW = 1                                                          ECH01380
  0065               OLDRW = 0                                                          ECH01390
  0066               NEWRW = 1                                                          ECH01400
  0067               CURLIN = FSTLIN                                                    ECH01410
               C                                                                        ECH01420
               C***********************************************************************ECH01430
               C                                                                        ECH01440
               C ENTER MAIN LOOP                                                        ECH01450
               C                                                                        ECH01460
               C***********************************************************************ECH01470
               C                                                                        ECH01480
  0068           150 J = NOLINE*CELWTH                                                  ECH01490
  0069               IF(MOD(NOLINE,50) .EQ. 0) WRITE(TYPEWR,9150) J                     ECH01500
  0070          9150 FORMAT(I6,' LINES CLASSIFIED.')                                    ECH01510
  0071               NOLINE = NOLINE + 1                                                ECH01520
```

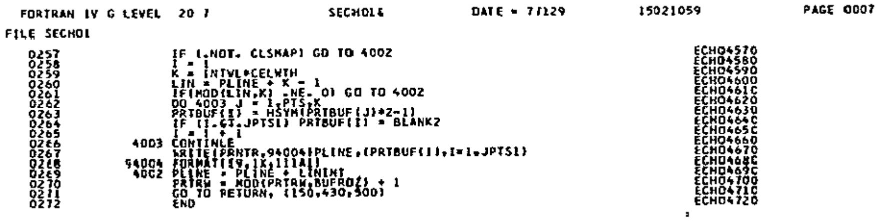
```
FORTRAN IV G LEVEL  20.7              SECH01C          DATE = 77129          15021059          , PAGE 0003
FILE SECH01
                ORONBS = 1 + (OLDRN-1)*ROWSIZ
                NRONBS = 1 + (NEWRN-1)*ROWSIZ
      C
      C FILL ONE ROW OF DATA
      C
                CALL FILBUF(ICLSCV,VECPRD,SCAPRD,LEICOV,CLNST,BDATA,RDATA,
              1 PIXCOR,PIXVAL,CELCOR,CELSUM,CELIKE,CELCLS,PIXCLS,CURLIN)
            155 IF ( NOT  CBJMAP) GO TO 157
                IIJJ = 1
                DO 156 I=1,JPTS,INTVL
                PRTBUF(IIJJ) = BLANK2
                IF (CELCLS(I) .NE. 0) GO TC 156
                PRTBUF(IIJJ) = SINGUL
            156 IIJJ = IIJJ + 1
                L = CURLIN
                WRITE(PRNTR,9290)L,(PRTBUF(I),I=1,JPTS1)
            157 IF (PHASE2) GO TO 160
                WRITE(CUTPLT) PREFIX,CURLIN,CELCLS,CELIKE
                CURLIN = CURLIN + GLCR
                IF (CURLIN  GE. INFO(5)) RETURN
                GO TO 150
      C
      C INSERT IN BUFFER CLASS NUMBERS FOR EACH PIXEL OF SINGULAR CELLS
      C
            160 J = NRCWBS
                DO 200 CELNUM=1,JPTS
                BUFFER(J) = 0
                IF(CELCLS(CELNUM) .NE. 0) GO TO 200
                K = J
                DC 165 I=1,CELSIZ
                BUFFER(K) = PIXCLS(I,CELNUM) + JPTS2
            165 K = K + 1
            200 J = J + CELSIZ
      C
      C
      C
      C
      C
      C
      99200 FORMAT(1X,28I4)
                IF(OLDRW .EQ. 0) GO TO 265
      C
      C COMPARE CELLS WITH FIELDS.
      C COMPARE CURRENT ROW WITH PREVIOUS ROW.
      C
            228 M = 1
                N = 1
                I3 = ORONBS
                J3 = NRONBS
      C
      C
      C
      C
      C
      C
                DO 240 CELNUM = 1,JPTS
                I2 = I3
                I3 = I3 + CELSIZ
                J2 = J3
                J3 = J3 + CELSIZ
                FLDUP = BUFFER(I2)
                IF(FLDUP .LE. JPTS) CLOSED(FLDUP) = .FALSE.
                IF(BUFFER(J2) .EQ  0) GO TO 229
                M = N
                GO TO 240
            229 SUP    = - 1E+70
                SRIGHT = - 1E+70
                SLEFT  = -.1E+70
      C
      C TRY TO COMPARE CELL WITH FIELD ABOVE.
      C
                IF(FLDSIZ(FLDUP) GE MAXSIZ .OR  FLDUP.GT JPTS) GO TO 230
                SUP = SUPLUS
      C
      C N .NE. M  IFF CELL 'CELNUM' HAS ALREADY BEEN COMPARED TO FIELD 'FLDUP'
      C

FORTRAN IV G LEVEL  20 7              SECH01C          DATE = 77129          15021059          PAGE 0004
FILE SECH01
      C IN WHICH CASE SETTING M=N IS ALL THAT IS NECESSARY.
      C
                IF(N .EQ  M) CALL LIKRAT(SUP,CELIKE(1,CELNUM),FLOLIK(1,FLDUP),
              &                AUXLIK(1,M),CELCLS(CELNUM),FLDCLS(FLDUP),AUXCLS(M))
            230 M = N
      C
      C TRY TO COMPARE CELL WITH FIELD ON LEFT.
      C
                FLDLFT = 0
                IF(CELNUM .EQ. 1) GO TO 232
                FLDLFT = BUFFER(J1)
                IF(FLDLFT.LE.JPTS  AND. FLDLFT.GT.0 .AND  FLDLFT NE.FLDUP  AND.
              &                                           FLDSIZ(FLDLFT).LT.MAXSIZ)
              & CALL LIKRAT(SLEFT,CELIKE(1,CELNUM),FLOLIK(1,FLDLFT),AUXLIK(1,2),
              & CELCLS(CELNUM),FLDCLS(FLDLFT),AUXCLS(2))
      C
      C TRY TO COMPARE CELL WITH FIELD ON RIGHT.
      C
            232 IF(CELNUM .EQ  JPTS) GO TO 234
                FLDRT = BUFFER(I3)
                IF(FLDRT.GT.JPTS .OR  FLDSIZ(FLDRT) GE.MAXSIZ .OR.
              &                                           FLDRT.EQ FLDUP) GO TO 234
                IF(BUFFER(J3).NE.0 .OR. FLDRT EQ FLDLFT) GO TO 234
                N = 4
                IF(M .EQ. 4) N=1
                CALL LIKRAT(SUPLUS,CELIKE(1,CELNUM+1),FLOLIK(1,FLDRT),AUXLIK(1,N),
              & CELCLS(CELNUM+1),FLDCLS(FLDRT),AUXCLS(N))
                IF(SUPLUS  GE  ANN1) CALL LIKRAT(SRIGHT,CELIKE(1,CELNUM),
              & AUXLIK(1,N),AUXLIK(1,3),CELCLS(CELNUM),AUXCLS(N),AUXCLS(3))
      C
      C FIND GREATEST LIKELIHOOD RATIO.
      C
            234 IF(SUP LT SLEFT .OR  SUP.LT.SRIGHT) GO TO 236
                IF(SUP .LT. ANN1) GO TO 240
                OPEN(FLDUP) = .TRUE.
                FIELD = FLDUP
                K = M
                GO TO 239
      C
            236 IF(SLEFT .GE. SRIGHT) GO TO 238
                IF(SRIGHT .LT  ANN1) GO TO 240
                OPEN(FLDRT) = .TRUE.
                FIELD = FLDRT
                K = 3
      C
      C ADD CELL TO FIELD.
      C
                BUFFER(J3) = FLDRT
                FLDSIZ(FLDRT) = FLDSIZ(FLDRT) + CELSIZ
                GO TO 239
      C
            238 IF(SLEFT .LT  ANN1) GO TO 240
                FIELD = FLDLFT
                K = 2
      C
      C ADD CURRENT CELL TO FIELD
      C
            239 BUFFER(J2) = FIELD
                CALL ADD2(FLOLIK(1,FIELD),AUXLIK(1,K),FLDCLS(FIELD),AUXCLS(K),
              & FLDSIZ(FIELD))
            240 J1 = J2
      C
      C**********************************************************************
      C
      C CLOSE FIELDS THAT ARE IN OLDRW BUT NOT IN NEWRW
      C
      C**********************************************************************
      C
                DO 260 FIELD=1,JPTS
                IF(CLOSED(FIELD) .OR. OPEN(FIELD)) GO TO 250
                NEXT = NEXT - 1
                STACK(NEXT) = FIELD
      C
      C CLASSIFY CLOSED FIELDS
      C
                CLASS2 = FLDCLS(FIELD) + JPTS2
```



```
  FORTRAN IV G LEVEL  20 7                SECHO1C           DATE = 77129          15021059                PAGE 0005
FILE SECHO1
   0158          243 ROW = NEWRW - 1                                                              ECH03050
   0159              ASSIGN 250 TO RETURN                                                         ECH03060
   0160              GO TO 3000                                                                   ECH03070
   0161          250 CLOSED(FIELD) = .TRUE.                                                       ECH03080
   0162          260 OPEN(FIELD) = .FALSE.                                                        ECH03090
               C                                                                                  ECH03100
               C BACKSCAN ACRUSS ROW.  BEGIN NEW FIELDS.                                          ECH03110
               C                                                                                  ECH03120
   0163          265 K = NROWBS + ROWSIZ-CELSIZ                                                   ECH03130
   0164              I = JPTS                                                                     ECH03140
   0165              DO 280 N=1,JPTS                                                              ECH03150
   0166              L = K - CELSIZ                                                               ECH03160
   0167              N = I                                                                        ECH03170
   0168              I = I-1                                                                      ECH03180
   0169              H = BUFFER(K)                                                                ECH03190
               C                                                                                  ECH03200
               C                                                                                  ECH03210
               C  IF H  GT. JPTS THIS CELL IS SENGULAR                                            ECH03220
               C                                                                                  ECH03230
   0170              IF(H .GT. JPTS) GO TO 280                                                    ECH03240
               C                                                                                  ECH03250
               C    PLACE CLASS OF FIELD H INTO EACH PIXEL OF FIELD H                             ECH03260
               C                                                                                  ECH03270
   0171              IF(H .NE  0) GO TO 270                                                       ECH03280
   0172              H = STACK(NEXT)                                                              ECH03290
   0173              NEXT = NEXT + 1                                                              ECH03300
   0174              BUFFER(K) = H                                                                ECH03310
               C                                                                                  ECH03320
               C CALL BEGFLD TO INITIALIZE STATISTICS FOR NEW FIELD                               ECH03330
               C                                                                                  ECH03340
   0175              CALL BEGFLD(CELCLS(N),FLDCLS(H),CELIKE(1,N),FLDLIK(1,H),FLDSIZ(H))ECH03350
   0176          270 IF(I.EC.0  OR. BUFFER(L) NE.0) GO TO 280                                     ECH03360
               C                                                                                  ECH03370
               C....... .. .    - ... .....    .MODIFIED........ - ... ............   .. ECH03380
               C                                                                                  ECH03390
   0177              CALL LIKRAT(SRIGHT,CELIKE(1,I),FLCLIK(1,H),AUXLIK,CELCLS(I),                 ECH03400
                   & FLDCLS(H),AUXCLS)                                                            ECH03410
   0178              IF(SRIGHT  LT  ANNL) GO TO 280                                               ECH03420
   0179              CALL ADDZ(FLDLIK(1,H),AUXLIK,FLDCLS(H),AUXCLS,FLDSIZ(H))                     ECH03430
               C                                                                                  ECH03440
   0180          275 BUFFER(L) = H                                                                ECH03450
   0181          280 K = L                                                                        ECH03460
   0182              IF(OLDRW .EQ. 0) GO TO 340                                                   ECH03470
   0183         9290 FORMAT(I9,1X,111A1)                                                          ECH03480
   0184          340 CURLIN = CURLIN + CLCR                                                       ECH03490
   0185              IF(CURLIN  GT. LSTLIN) GO TO 400                                             ECH03500
               C                                                                                  ECH03510
               C POINT TO NEXT ROW IN BUFFER                                                      ECH03520
               C                                                                                  ECH03530
   0186              OLDRW = NEWRW                                                                ECH03540
   0187          360 NEWRW = MOD(NEWRW,BUFROZ) + 1                                                ECH03550
   0188              IF(NEWRW .NE. PRTRW) GO TO 150                                               ECH03560
               C                                                                                  ECH03570
               C CLASSIFY AND PRINT ONE ROW OF CELLS                                              ECH03580
               C                                                                                  ECH03590
   0189              K = 1 + (PRTRW-1)*ROWSIZ                                                     ECH03600
   0190              L = K + ROWSIZ-CELSIZ                                                        ECH03610
   0191              DO 365 I=K,L,CELSIZ                                                          ECH03620
   0192              H = BUFFER(I)                                                                ECH03630
   0193              IF(H  GT. JPTS) GO TO 365                                                    ECH03640
   0194              CLASS2 = FLDCLS(H) + JPTS2                                                   ECH03650
   0195          362 DO 364 J=I,L,CELSIZ                                                          ECH03660
   0196              IF(BUFFER(J) .NE. H) GO TO 364                                               ECH03670
   0197              M = J + CELSZ1                                                               ECH03680
   0198              DO 361 N = J,M                                                               ECH03690
   0199          361 BUFFER(N) = CLASS2                                                           ECH03700
   0200          364 CONTINLE                                                                     ECH03710
   0201          365 CONTINLE                                                                     ECH03720
   0202              ASSIGN 150 TO RETURN                                                         ECH03730
   0203              GO TO 4000                                                                   ECH03740
               C                                                                                  ECH03750
               C PRINT LAST PRINTER LINE.                                                         ECH03760
               C                                                                                  ECH03770
   0204          400 K = NROWBS                                                                   ECH03780
   0205          410 L = CURLIN                                                                   ECH03790
               C                                                                                  ECH03800
```



```
  FORTRAN IV G LEVEL  20 7                SECHO1C           DATE = 77129          15021059                PAGE 0006
FILE SECHO1
               C CLASSIFY ALL OPEN FIELDS                                                         ECH03810
               C                                                                                  ECH03820
   0206              L = NROWBS + ROWSIZ-CELSIZ                                                   ECH03830
   0207              DO 420 I=NROWBS,L,CELSIZ                                                     ECH03840
   0208              FIELD = BUFFER(I)                                                            ECH03850
   0209              IF(FIELD .GT. JPTS) GO TO 420                                                ECH03860
   0210              CLASS2 = FLDCLS(FIELD) + JPTS2                                               ECH03870
   0211          415 ROW = NEWRW                                                                  ECH03880
   0212              ASSIGN 420 TO RETURN                                                         ECH03890
   0213              GO TO 3000                                                                   ECH03900
   0214          420 CONTINLE                                                                     ECH03910
               C                                                                                  ECH03920
               C FINISH PRINTING BUFFER.                                                          ECH03930
               C                                                                                  ECH03940
   0215              ASSIGN 430 TO RETURN                                                         ECH03950
   0216          430 IF(PRTRW .EQ. NEWRW) ASSIGN 500 TC RETURN                                    ECH03960
   0217              GO TO 4000                                                                   ECH03970
   0218          500 PLINE = 0                                                                    ECH03980
   0219              PREFIX(1) = 7                                                                ECH03990
   0220              WRITE(CUTPLT,ERR=600) PREFIX,PLINE,PRTBUF                                    ECH04000
   0221              RETURN                                                                       ECH04010
               C                                                                                  ECH04020
               C******************************************************************************ECH04030
               C                                                                                  ECH04040
               C END OF TAPE OR IO ERROR                                                          ECH04050
               C                                                                                  ECH04060
               C******************************************************************************ECH04070
               C                                                                                  ECH04080
   0222          600 WRITE(PRNTR,9610) PLINE                                                      ECH04090
   0223         9610 FORMAT(' EC047 END OF TAPE OR AN I/O ERROR OCCURED. ',                       ECH04100
                   & 'CLASSIFICATION TERMINATED AT LINE',I6,'    (ECHOA5)')                       ECH04110
   0224              WRITE (TYPEWR,9610) PLINE                                                    ECH04120
   0225              IF(OUTPUT .EQ. CLASSR) GO TO 500                                             ECH04130
   0226              CALL TOPBS(OUTPUT,1,ERROR)                                                   ECH04140
   0227              I = 1                                                                        ECH04150
   0228              CALL TCPBS(OUTPUT,I,ERROR)                                                   ECH04160
   0229              IF(I .LE. 0) RETURN                                                          ECH04170
   0230              BACKSPACE CUTPUT                                                             ECH04180
   0231              GO TO 500                                                                    ECH04190
               C                                                                                  ECH04200
               C PROPAGATE CLASSIFICATICN THROUGHOUT BUFFER.                                      ECH04210
               C                                                                                  ECH04220
   0232         3000 IF(ROW .LE. 0) ROW=BUFROZ                                                    ECH04230
   0233              STOP = .TRUE                                                                 ECH04240
               C                                                                                  ECH04250
               C PROPAGATE CLASSIFICATION THROUGHOUT RCW.                                         ECH04260
               C                                                                                  ECH04270
   0234              I1 = 1 + (ROW-1)*ROWSIZ                                                      ECH04280
   0235              I2 = I1 + ROWSIZ-CELSIZ                                                      ECH04290
   0236              DO 3002 I3=I1,I2,CELSIZ                                                      ECH04300
   0237              IF(BUFFER(I3) .NE  FIELD) GO TO 3002                                         ECH04310
   0238              STOP =  FALSE                                                                ECH04320
   0239              I4 = I3 + CELSZ1                                                             ECH04330
   0240              DC 3001 I5=I3,I4                                                             ECH04340
   0241         3001 BUFFER(I5) = CLASS2                                                          ECH04350
   0242         3002 CONTINLE                                                                     ECH04360
   0243              IF(STOP .OR. ROW.EQ.PRTRW) GO TG RETURN, (250,420)                           ECH04370
   0244              ROW = ROW - 1                                                                ECH04380
   0245              GO TO 3000                                                                   ECH04390
               C                                                                                  ECH04400
               C PRINT ONE ROW OF CELLS                                                           ECH04410
               C                                                                                  ECH04420
   0246         4000 I1 = 1 + (PRTRW-1)*ROWSIZ                                                    ECH04430
   0247              I2 = I1 + CELSIZ - CELWTH                                                    ECH04440
   0248              DO 4002 I3=I1,I2,CELWTH                                                      ECH04450
   0249              I5 = 0                                                                       ECH04460
   0250              I4 = I3 + ROWSIZ-CELSIZ                                                      ECH04470
   0251              DO 4001 I6=I3,I4,CELSIZ                                                      ECH04480
   0252              I7 = I6 + CELWTH - 1                                                         ECH04490
   0253              DO 4001 I8=I6,I7                                                             ECH04500
   0254              I5 = I5 + 1                                                                  ECH04510
   0255         4001 PRTBUF(I5) = BUFFER(I8) - JPTS2                                              ECH04520
   0256              WRITE(CUTPLT,ERR=600) PREFIX,PLINE,PRTBUF                                    ECH04530
               C                                                                                  ECH04540
               C CLASSIFICATION MAP                                                               ECH04550
               C                                                                                  ECH04560
```



```
 FORTRAN IV G LEVEL  20 7              SECHO1S           DATE = 77129          15021059              PAGE 0007
FILE SECHO1
                      IF (.NOT. CLSMAP) GO TO 4002                                     ECHO4570
                      I = 1                                                            ECHO4580
                      K = INTVL*CELWTH                                                 ECHO4590
                      LIN = PLINE + K - 1                                              ECHO4600
                      IF(MOD(LIN,K) .NE. 0) GO TO 4002                                 ECHO4610
                      DO 4003 J = 1,PTS,K                                              ECHO4620
                      PRTBUF(I) = HSYM(PRTBUF(J)*2-1)                                  ECHO4630
                      IF (I.GT.JPTS1) PRTBUF(I) = BLANK2                               ECHO4640
                      I = I + 1                                                        ECHO4650
                 4003 CONTINUE                                                         ECHO4660
                      WRITE(PRNTR,94004)PLINE,(PRTBUF(I),I=1,JPTS1)                    ECHO4670
                94004 FORMAT(I9,1X,111A1)                                              ECHO4680
                 4002 PLINE = PLINE + LININT                                           ECHO4690
                      PRTRW = MOD(PRTRW,BUFROW) + 1                                    ECHO4700
                      GO TO RETURN, (150,430,500)                                      ECHO4710
                      END                                                              ECHO4720
```

LARS Program Abstract ______

MODULE IDENTIFICATION

Module Name: GATHER Function Name: ECHO FUNCTIONAL SUPPORT

Purpose: Gather cell statistics information for processing.

System/Language: CMS/FORTRAN

Author: P. D. Alenduff Date: 4/12/77

Latest Revisor: ______ Date: -

MODULE ABSTRACT

GATHER computes information about cell statistics for actual classification by SECHO.

PURDUE UNIVERSITY
Laboratory for Applications of Remote Sensing
1220 Potter Drive
West Lafayette, Indiana 47906

1. Module Usage

GATHER

```
CALL GATHER (RDATA, PIXVAL, PIXCOR, CELSUM, CELCOR, CELWTH,
             VECSIZ, MTXSIZ, NSR, NVR)
```

Input Arguments:

RDATA (NSR, NVR)	R*4 Calibrated data values from data tape with NSR samples and NVR channels.
CELWTH	I*4 Number of data points in width of cell.
VECSIZ	I*4 Number of channels to be used in classification.
MTXSIZ	I*4 Number of entries in the triangular covariance matrix. Equal to (channels * (channels + 1))/2.
NSR	I*4 Number of data samples per line in RDATA.
NVR	I*4 Number of channels of data in RDATA array.

Output Arguments:

PIXVAL (NOFET3, CELWTH*CELWTH)	R*4 An array of actual data values in the cell dimensioned channels by (cell width)^2.
PIXCOR (MTXSIZ, CELWTH*CELWTH)	R*4 Cross product terms for all data points in a cell, dimensioned (cell width)^2 by MTXSIZ.
CELSUM (VECSIZ)	R*4 Sum of data values in the cell for each channel of data (dimensioned channels).

CELCOR (MTXSIZ)   R*4 Correlation matrix for data values in cell consisting of a sum of products of data values in cell.

GATHER is called to return information concerning the data characteristics of one cell of data.

2. Internal Description

GATHER loops for each channel of data and each pixel within the cell to produce the data array returned.

3. Input Description

Not applicable

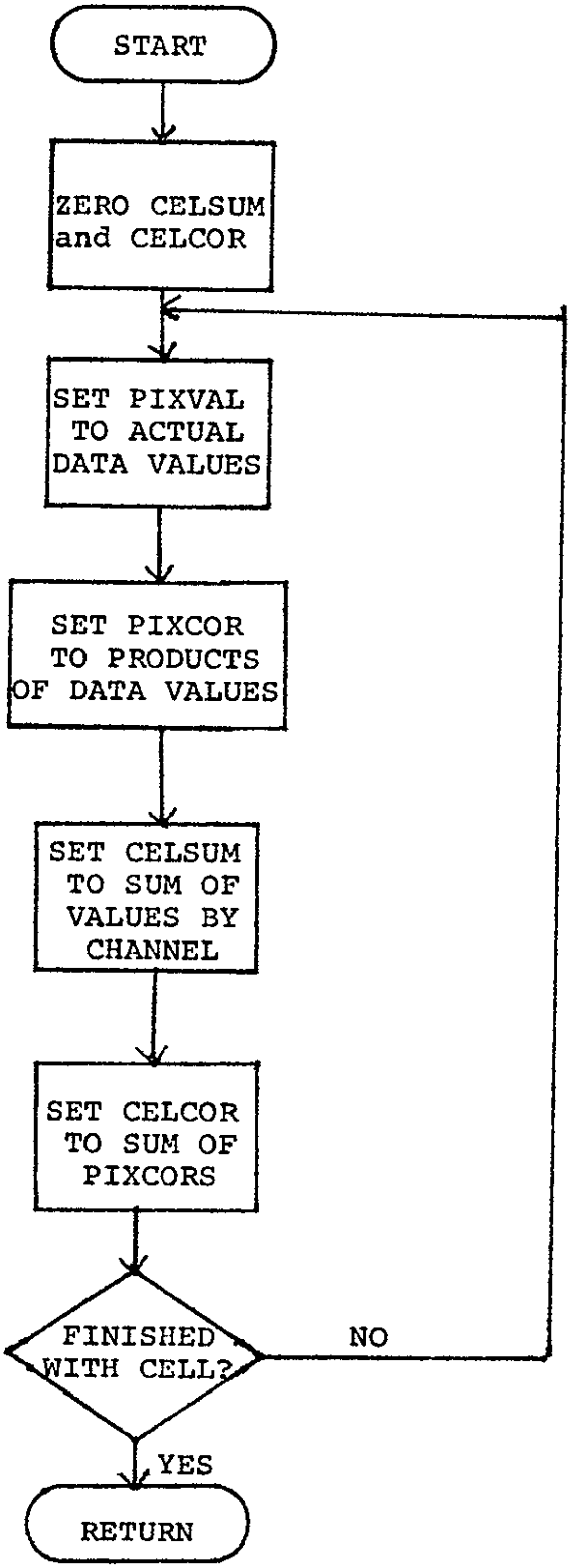
4. Output Description

Not applicable

5. Supplemental Information

Not applicable

6. Flowchart


START
ZERO CELSUM and CELCOR
SET PIXVAL TO ACTUAL DATA VALUES
SET PIXCOR TO PRODUCTS OF DATA VALUES
SET CELSUM TO SUM OF VALUES BY CHANNEL
SET CELCOR TO SUM OF PIXCORS
FINISHED WITH CELL?
NO
YES
RETURN

```
C    GATHER        LARS  XXXX
C                  GATHER STATISTICS ON A GIVEN CELL
C
C***********************************************************************
C=GATHER              REVISED 04/11/74  R.L.KETTIG
C COMPUTE STATISTICS FOR THE CELL WHOSE FIRST DATA VALUE IS RDATA(1).
      SUBROUTINE GATHER(RDATA,PIXVAL,PIXCOR,CELSUM,CELCOR,CELWTH,VECSIZ,
     & MTXSIZ,NSR,NVR)
      IMPLICIT INTEGER * 4 (A-Z)
      REAL*4 RDATA(1),PIXVAL(1),PIXCOR(1),CELSUM(VECSIZ),CELCOR(MTXSIZ)
      DO 10 L=1,VECSIZ
   10 CELSUM(L) = 0.
      DO 11 L=1,MTXSIZ
   11 CELCOR(L) = 0.
      N = CELWTH*MTXSIZ
      J1 = 0
      J2 = 0
      L2 = 0
      DO 4 J=1,CELWTH
      K1 = J1
      K2 = J2
C J1 = (J-1)*NVR
C J2 = (J-1)*N
      DO 3 K=1,CELWTH
      K1 = K1 + 1
      M2 = 0
      L1 = K1
      K3 = L2
C K1 = J1 + K
C K2 = J2 + (K-1)*MTXSIZ
C K3 = (K-1)*VECSIZ + (J-1)*VECSIZ*CELWTH
      DO 2 L=1,VECSIZ
      M1 = K3
      L2 = L2 + 1
C L1 = K + (L-1)*NSR + (J-1)*NVR
C L2 = L + (K-1)*VECSIZ + (J-1)*VECSIZ*CELWTH
      PIXVAL(L2) = RDATA(L1)
      CELSUM(L) = CELSUM(L) + PIXVAL(L2)
      DO 1 M=1,L
      M1 = M1 + 1
      M2 = M2 + 1
      M3 = K2 + M2
C M1 = M + (K-1)*VECSIZ + (J-1)*VECSIZ*CELWTH
C M2 = M + L*(L-1)/2
C M3 = M + L*(L-1)/2 + (K-1)*MTXSIZ + (J-1)*N
      PIXCOR(M3) = PIXVAL(M1)*PIXVAL(L2)
    1 CELCOR(M2) = CELCOR(M2) + PIXCOR(M3)
    2 L1 = L1 + NSR
    3 K2 = K2 + MTXSIZ
      J1 = J1 + NVR
    4 J2 = J2 + N
      RETURN
      END
```

LARS Program Abstract ________

MODULE IDENTIFICATION

Module Name: FILBUF Function Name: SECSUP

Purpose: Retrieve cell statistics

System/Language: CMS/FORTRAN

Author: P. D. Alenduff Date: 4/18/77

Latest Revisor: ________ Date: ________

MODULE ABSTRACT

FILBUF returns a set of buffers with cell information from either direct processing from the data tape or reading from the intermediate tape.

1. Module Usage

FILBUF

```
CALL FILBUF (ICLSCV, VECPRD, SCAPRD, SETCOV, CONST, BDATA,
             RDATA, PIXCOR, PIXVAL, CELCOR, CELSUM, CELIKE,
             CELCLS, PIXCLS, CURLIN, *)
```

Input Arguments:

ICLSCV (VARSZ3, NOPOOL) — R*4 Class covariance matrices from statistics dimensioned classes by (channels * (channels + 1))/2.

VECPRD (NOFET3, NOPOOL) — R*4 Vector product matrix that is inverse of class mean matrix transposed times the covariance matrix for the class, dimensioned channels by classes.

SCAPRD (NOPOOL) — R*4 Scalar product of the classes for all statistics classes.

DETCOV (NOPOOL) — R*4 Determinant of the covariance matrix for each class.

CONST (NOPOOL) — R*4 Constant array computed from ln (2*PI*class covariance matrix) for each class.

BDATA (NVR, ID(6)) — L*1 Buffer array used by GADLIN for reading data tape dimensioned channels on tape by samples per line on tape.

RDATA (NVR, CELWTH) — R*4 Array used to store calibrated data values from tape for one row of cells (multiple data lines).

PIXCOR (VARSZ3, CELWTH^2) R*4 Array used to store cross product data values in one cell for a correlation type indication.

PIXVAL (NOFET3, CELWTH^2) R*4 Array used to store pixel data values for a cell.

CELCOR (VARSZ3) R*4 Sum of the PIXCOR values for a cell being processed.

Output Arguments:

CELSUM (NOFET3) R*4 The sum of all data points in a cell for each channel.

CELIKE (NWORD, JPTS) R*4 Likelihood values for each class for each cell in a row of cells.

CELCLS (JPTS) I*2 Class numbers for each cell in a row of cells.

PIXCLS (NWORDS, JPTS) I*2 Class numbers of pixels for all cells in a row of cells.

CURLIN I*4 Current line number requested.

FILBUF is called to fill up CELCLS and CELIKE for annexation or writing to the intermediate tape. By either reading from the intermediate tape for annexation processing, or reading from the data tape and carrying out processing for the area, cell information can be produced.

2. Internal Description

FILBUF produces arrays CELCLS and CELIKE from either reading the intermediate tape that is in a classification format or from reading the input data tape via GADLIN. The raw information is then processed by producing cell statistics with GATHER, establishing cell classes with SAMCLS, testing for cell homogeneity, and outputing a row of cell information. This routine uses common blocks GLOCOM and SECCOM.

3. Input Description

The only input is possible input from the intermediate tape which stores the same format used by classification tapes and stores cell statistics for later annexation processing.

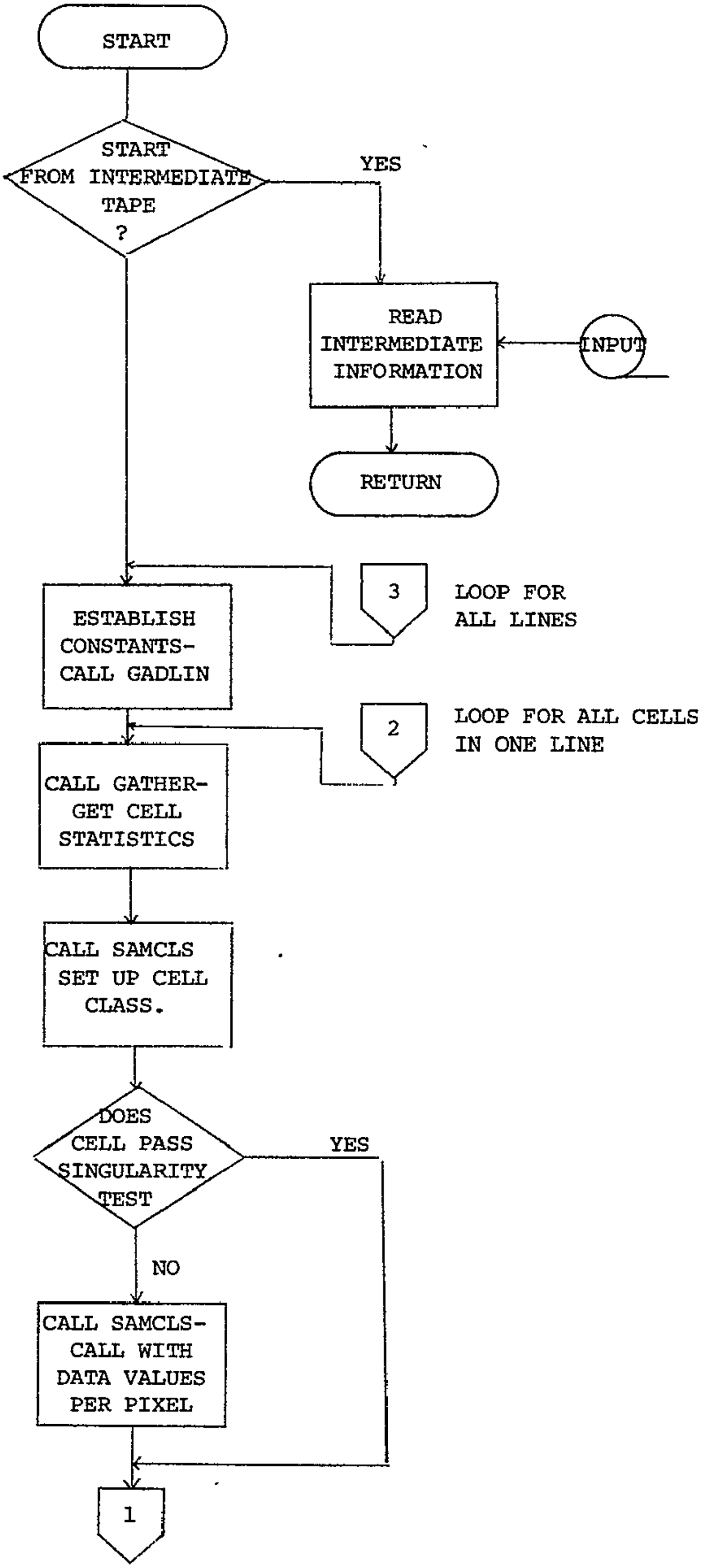
4. Output Description

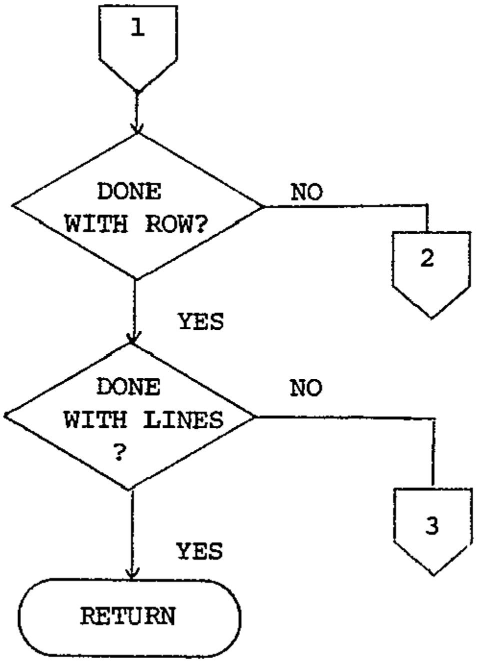
Not applicable.

5. Supplemental Information

Not applicable.

6. Flowchart


START
START FROM INTERMEDIATE TAPE ?
YES
READ INTERMEDIATE INFORMATION
INPUT
RETURN
3
LOOP FOR ALL LINES
ESTABLISH CONSTANTS- CALL GADLIN
2
LOOP FOR ALL CELLS IN ONE LINE
CALL GATHER- GET CELL STATISTICS
CALL SAMCLS SET UP CELL CLASS.
DOES CELL PASS SINGULARITY TEST
YES
NO
CALL SAMCLS- CALL WITH DATA VALUES PER PIXEL
1


1
DONE WITH ROW?
NO
2
YES
DONE WITH LINES ?
NO
3
YES
RETURN

```
C   FILBUF   LARS   XXXX
C
C*********************************************************************
C
C RETURNS CELCLS AND CELIKE VALUES FROM EITHER DATA TAPE OR INTER
C        MEDIATE TAPE
C
C*********************************************************************
C
      SUBROUTINE FILBUF (ICLSCV, VECPRD, SCAPRD, DETCOV, CONST,
     & BDATA, RDATA, PIXCOR, PIXVAL, CELCOR, CELSUM, CELIKE, CELCLS,
     & PIXCLS,CURLIN)
C
      IMPLICIT INTEGER * 4 (A-Z)
C
      COMMON /GLOCOM/ BLANK, CARD(20), CHKOUT, COPFIL, CLASSR, CLASSX,
     1    CLUSTX, CONPUT, CPYOUT, CRDRDR, CRDSEQ, DATAPE,
     2    DUPLTP, DUPRUN, ERRMSG, FBPNT,
     3    FILESV, FLDBND, HDATA, HEAD(88), ID(200), IMAGEX,
     4    IMARK, KEYBD, MAPTAP, MAXCHA, MAXCLS,
     5    PAGSIZ, PNCH, POINT, PRESUX, PRNTR, READIN,
     6    RESTRT, RUNFIL, RUNTAB(10,3),
     7    SDATA, SEPARX, SEPTPX, SPARE(10), TEMPAS(30),
     8    TPSTAT(6), TTFLDX ,TYPEWR,
     9    TOP, ARRAY(12500)
C
      REAL * 8 ARRAY
      REAL * 4 FRQCAL(5,30)
      INTEGER * 4 COMENT(16), DATE(5), HED1(16), HED2(16), TIME(5)
      INTEGER * 2 BLANK2
      LOGICAL * 4 CHKOUT
      LOGICAL * 1 BLANK1
      EQUIVALENCE (DATSAV,ID(1)), (CURRUN,ID(3)), (FRQCAL(1),ID(51)),
     1        (HED1(1),HEAD(8)), (DATE(1),HEAD(26)), (HED2(1),HEAD(39)),
     2        (TIME(1),HEAD(58)), (COMENT(1),HEAD(72)),
     3        (MAPSAV,TPSTAT(1)),
     4        (SEPSCR,TPSTAT(2)), (DUPIN,TPSTAT(3)), (DASTAT,TPSTAT(4)),
     5        (CCPSER,TPSTAT(5)), (TRAOUT,TPSTAT(6)),
     6        (BLANK,BLANK2,BLANK1)
C
      COMMON /SECCOM/ BUFROZ,CSELEC,CSET(3,30),CSET3(3,30),THRES,CELSIZ,
     1 CELWTH,INFO(17),INPUT, INTTAP, INTFIL, JPTS, LINES, NOCLS, NOFET,
     2 NOFET3,NOPOOL,NWCRD,NWORD2, NVR, OUTPUT, POLNAM(2,60), PREFIX(2),
     3 PTS, ROTAPE, ROFILE, SYMCNT, VARSZ3, CSEL(30), CSEL3(30),
     3 FETVEC(30), FETVC3(30), POLPTR(2,60), POLSTK(60),
     4 CDFLAG, CLSMAP, CSET1(3,30), OBJMAP, PHASE1, PHASE2, POLNM1(60),
     5 PRSTAT, SYM(60), SYMMTX(60)
C
      REAL * 4 CSELEC, CSET, CSET3, THRES
C
      INTEGER * 2 CSEL, CSEL3, FETVEC, FETVC3, POLPTR, POLSTK
C
      LOGICAL * 1 CDFLAG, CLSMAP, CSET1, OBJMAP, PHASE1, PHASE2, POLNM1,
     1 PRSTAT
C
C
C
C
C
C  LOCAL VARIABLES
C
      INTEGER*2 BLOCK(4),CELCLS(JPTS),PIXCLS(NWCRD2,JPTS),
     1 BDATA(1)
      REAL * 4 ICLSCV(VARSZ3,NOPOOL), VECPRD(NOFET3,NOPOOL),
     1 SCAPRD(NOPOOL),DETCOV(NOPOOL),CONST(NOPOOL),CELSUM(NOFET3),
     2 CELCOR(VARSZ3),PIXVAL(1),PIXCOR(1),CELIKE(NWORD,JPTS),
     3 RDATA(NVR,CELWTH),CHISQR,AI
C
C
C INITIALIZE
C
      ERRFG = 0
      DO 100 I=2,4
  100 BLOCK(I) = INFO(I+5)
C
C GLCR = NUMBER OF DATA LINES TO SKIP TO GET NEXT CELL DATA LINE
```

```
C
      NSR = NVR/NOFET3
C
C***********************************************************************
C
C ENTER MAIN LOOP
C
C***********************************************************************
C
C GET ONE ROW OF DATA FOR CELLS (MAY BE MORE THAN ONE LINE)
C
      IF (PHASE2 .AND. .NOT. PHASE1) GO TO 400
      I = CURLIN
      DO 155 J=1,CELWTH
      BLOCK(1) = I
      CALL GAOLIN(BLOCK,CSEL3,CSET3,ID,DATAPE,NOFET3,NSR,BDATA,
     & RDATA(1,J),ROLL,ERROR)
      IF(ERROR.GT.0) CALL LINERR(ERROR,I,&810,&155,&810)
  155 I = I + INFO(6)
      H  = 1
C
C GET CELL DATA ARRAYS UPDATED FOR EACH CELL IN LINE
C
      DO 200 CELNUM=1,JPTS
      CALL GATHER(RDATA(H,1),PIXVAL,PIXCOR,CELSUM,CELCOR,CELWTH,
     & NOFET3,VARSZ3,NSR,NVR)
C
C CLASSIFY EACH CELL WITH MAXIMUM LIKELIHOOD AND RETURN CHI-SQUARE
C          VALLE FOR THRESHOLDING HOMOGENEOUS DEFINITION
C
      CALL SAMCLS(CLASS,CHISQR,CELIKE(1,CELNUM),ICLSCV,VECPRD,
     &SCAPRD,CONST,CELCOR,CELSUM,CELSIZ,NOFET3,VARSZ3,NOPOOL,.TRUE.)
      CELCLS(CELNUM) = CLASS
      IF (CHISQR .LE. CSELEC) GO TO 200
C
C NON HOMOGENEOUS CELL.  CLASSIFY VECTORS SEPARATELY.
C
      L = 1
      M = 1
      CELCLS(CELNUM) = 0
      DO 196 PIXNUM=1,CELSIZ
C
C TRY AGAIN
C
      CALL SAMCLS(CLASS,CHISQR,A1,ICLSCV,VECPRD,SCAPRD,CONST,
     & PIXCOR(M),PIXVAL(L),1,NOFET3,VARSZ3,NOPOOL,.FALSE.)
      PIXCLS(PIXNUM,CELNUM) = CLASS
      L = L + NOFET3
  196 M = M + VARSZ3
  200 H = H + CELWTH
      GO TO 210
C***********************************************************************
C
C READ FROM INPUT INTERMEDIATE TAPE
C
C***********************************************************************
  400 READ(INPUT,ERR=810)PREFIX,I,CELCLS,CELIKE
      IF (I .LT. CURLIN) GO TO 400
  210 RETURN
C
C***********************************************************************
C
C ERROR PROCESSING
C
C***********************************************************************
C
  810 WRITE(PRNTR,9810) CURLIN
 9810 FORMAT('0READ ERROR FROM GAOLIN.  CURRENT LINE NO. IS ',I4//
     & ' PROCESSING TERMINATED.')
      ERRFG = 1
      GO TO 210
      END
```

LARS Program Abstract ____________

MODULE IDENTIFICATION

Module Name: STATS2 Function Name: SECSUP

Purpose: Carry out annexation tests and processing

System/Language: CMS/FORTRAN

Author: P. D. Alenduff Date: 4/20/77

Latest Revisor: ____________ Date: ____________

MODULE ABSTRACT

STATS2 contains entry points LIKRAT, ADD2, and BEGFLD. LIKRAT produces a likelihood test by subtracting log - likelihoods and returns a resultant likelihood value. ADD2 adds a cell to the statistics of a field. BEGFLD initializes field statistics from a fields statistics.

1. Module Usage

LIKRAT

```
CALL LIKRAT (STAT, CELIKE, FLDLIK, AUXLIK, CELCLS, FLDCLS,
             AUXCLS)
```

Input Arguments:

CELIKE (NOPOOL) R*4 Log likelihood values for this cell for each class in the statistics.

FLDLIK (NOPOOL) R*4 Log likelihood values of all cells annexed to the field for each class in the statistics.

CELCLS I*2 Class number of cell being considered.

FLDCLS I*2 Class number of field being tested.

Output Arguments:

AUXLIK (NOPOOL) R*4 Resultant log likelihood of annexation of cell to given field for each class.

AUXCLS I*2 Resultant class of annexation test for this cell.

STAT R*4 Resulting statistic concerning annexation performance for this cell to the field being considered.

LIKRAT produces a resulting likelihood, class and statistic if the given cell is annexed to the supplied field.

ADD2

```
CALL ADD2 (FLDLIK, AUXLIK, FLDCLS, AUXCLS, FLDSIZ)
```

Input Arguments:

AUXLIK (NOPOOL) R*4 Likelihood values formed with temporary annexation of cell to field.

AUXCLS I*2 Class number of cell when annexation of cell to field was successful.

FLDSIZ I*4 Number of pixels included in the given field.

Output Arguments:

FLDLIK (NOPOOL) R*4 Likelihood values of new field formed.

FLDCLS I*2 Class of field formed.

FLDSIZ I*4 (see above).

ADD2 adds the cell supplied to the field information supplied. Temporary likelihood values are assigned as new likelihoods for field.

BEGFLD

CALL BEGFLD (CELCLS, FLDCLS, CELIKE, FLDLIK, FLDSIZ)

Input Arguments:

CELCLS I*2 See above.

CELIKE (NOPOOL) R*4 See above.

Output Arguments:

FLDCLS I*2 See above.

FLDLIK (NOPOOL) R*4 See above.

FLDSIZ I*4 See above.

BEGFLD initialize FLDCLS, FLDLIK, and FLDSIZ to be equal to CELCLS, CELIKE, and CELSIZ.

2. Internal Description

See above.

3. Input Description

Not applicable

4. Output Description

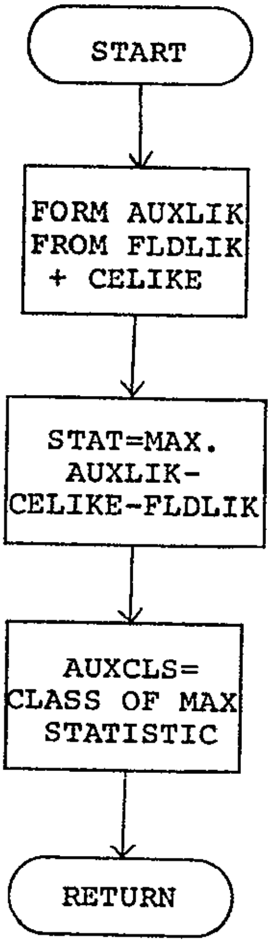
Not applicable

5. Supplemental Information

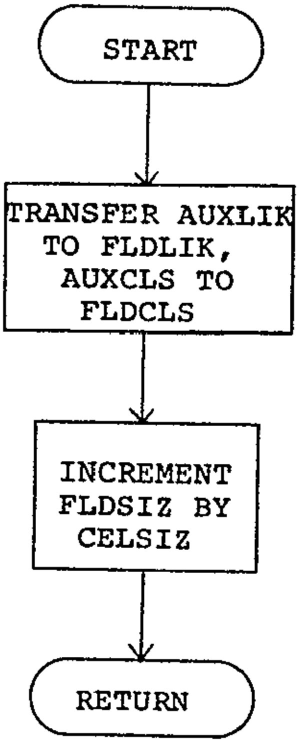
Not applicable

6. Flowchart

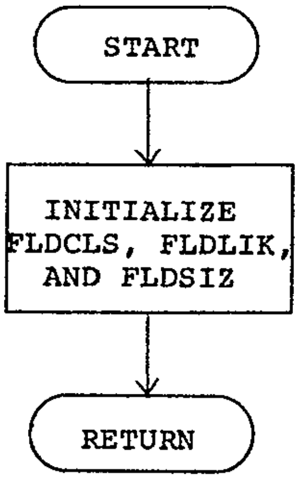
LIKRAT


START
FORM AUXLIK FROM FLDLIK + CELIKE
STAT=MAX. AUXLIK-CELIKE-FLDLIK
AUXCLS= CLASS OF MAX STATISTIC
RETURN

ADD2


START
TRANSFER AUXLIK TO FLDLIK, AUXCLS TO FLDCLS
INCREMENT FLDSIZ BY CELSIZ
RETURN

BEGFLD



```
  FORTRAN IV G LEVEL  20.7              STATS2            DATE = 77133          21008002            PAGE 0001
FILE STATS2
                 C       STATS2      LARS  XXXX                                                    ECH00010
                 C                                                                                  ECH00020
                 C*************************************************************************ECH00030
                 C                                                                                  ECH00040
                 C THIS SUBROUTINE WAS WRITTEN TO HELP IMPLEMENT A MORE COMPLEX SCANNING   ECH00050
                 C LOOP THAN THE ONE IN THE ORIGINAL ECHO.  ESSENTIALLY IT DOES WHAT       ECH00060
                 C ANNEX DOES, BUT IT OPERATES ONLY IN SUPERVISED MODE OF ANNEXATION.       ECH00070
                 C                                                                                  ECH00080
                 C*************************************************************************ECH00090
                 C                                                                                  ECH00100
  0001                 SUBROUTINE STATS2                                                   ECH00110
  0002                 IMPLICIT INTEGER * 4 (A-Z)                                          ECH00120
                 C                                                                                  ECH00130
  0003                 COMMON /SECCOM/ BUFROZ,CSELEC,CSET(3,30),CSET3(3,30),THRES,CELSIZ,ECH00140
                      1 CELWTH,INFO(17),INPUT, INTTAP, INTFIL, JPTS, LINES, NOCLS, NOFET,ECH00150
                      2 NOFET3,NOPOOL,NWORD,NWORD2, NVR, OUTPUT, POLNAM(2,60), PREFIX(2),ECH00160
                      3 PTS, ROTAPE, ROFILE, SYMCNT, VARSZ3, CSEL(30), CSEL3(30),         ECH00170
                      3 FETVEC(30), FETVC3(30), POLPTR(2,60), POLSTK(60),                  ECH00180
                      4 CDFLAG, CLSMAP, CSETI(3,30), OBJMAP, PHASE1, PHASE2, POLNM1(60),  ECH00190
                      5 PRSTAT, SYM(60), SYMMTX(60)                                        ECH00200
                 C                                                                                  ECH00210
  0004                 REAL * 4 CSELEC, CSET, CSET3, THRES                                 ECH00220
                 C                                                                                  ECH00230
  0005                 INTEGER * 2 CSEL, CSEL3, FETVEC, FETVC3, POLPTR, POLSTK             ECH00240
                 C                                                                                  ECH00250
  0006                 LOGICAL * 1 CDFLAG, CLSMAP, CSETI, OBJMAP, PHASE1, PHASE2, POLNM1,ECH00260
                      1 PRSTAT                                                             ECH00270
                 C                                                                                  ECH00280
  0007                 REAL * 4 ANN1                                                       ECH00290
                 C                                                                                  ECH00300
  0008                 REAL * 4 STAT,CELIKE(NOPOOL),FLDLIK(NOPOOL),AUXLIK(NOPOOL)          ECH00310
  0009                 INTEGER*2 CELCLS,FLDCLS,AUXCLS                                      ECH00320
                 C                                                                                  ECH00330
  0010                 ENTRY LIKRAT(STAT,CELIKE,FLDLIK,AUXLIK,CELCLS,FLDCLS,AUXCLS)        ECH00340
  0011                 DO 20 I=1,NOPOOL                                                    ECH00350
  0012                 AUXLIK(I) = CELIKE(I) + FLDLIK(I)                                   ECH00360
  0013                 IF(I .EQ. 1) GO TO 10                                               ECH00370
  0014                 IF(AUXLIK(I) .LE. STAT) GO TO 20                                    ECH00380
  0015              10 STAT = AUXLIK(I)                                                    ECH00390
  0016                 AUXCLS = I                                                          ECH00400
  0017              20 CONTINUE                                                            ECH00410
  0018                 STAT = STAT - CELIKE(CELCLS) - FLDLIK(FLDCLS)                       ECH00420
  0019                 RETURN                                                              ECH00430
                 C                                                                                  ECH00440
  0020                 ENTRY ADD2(FLDLIK,AUXLIK,FLDCLS,AUXCLS,FLDSIZ)                      ECH00450
  0021              50 DO 60 I=1,NOPOOL                                                    ECH00460
  0022              60 FLDLIK(I) = AUXLIK(I)                                               ECH00470
  0023                 FLDCLS = AUXCLS                                                     ECH00480
  0024                 FLDSIZ = FLDSIZ + CELSIZ                                            ECH00490
  0025                 RETURN                                                              ECH00500
                 C                                                                                  ECH00510
                 C-ENTRY FOR FIELD INITIALIZATION.                                                  ECH00520
                 C                                                                                  ECH00530
  0026                 ENTRY BEGFLD(CELCLS,FLDCLS,CELIKE,FLDLIK,FLDSIZ)                    ECH00540
  0027                 FLDCLS = CELCLS                                                     ECH00550
  0028                 FLDSIZ = CELSIZ                                                     ECH00560
  0029                 DO 40 I=1,NOPOOL                                                    ECH00570
  0030              40 FLDLIK(I) = CELIKE(I)                                               ECH00580
  0031                 RETURN                                                              ECH00590
  0032                 END                                                                 ECH00600
```

REPRODUCIBILITY OF THE ORIGINAL PAGE IS POOR

LARS Program Abstract ______

MODULE IDENTIFICATION

Module Name: SAMCLS Function Name: SECSUP

Purpose: Classify cell using maximum likelihood

System/Language: CMS/FORTRAN

Author: P. D. Alenduff Date: 4/14/77

Latest Revisor: ______ Date: ______

MODULE ABSTRACT

SAMCLS performs a maximum likelihood classification by calling LOGLIK and finding the maximum likelihood class for a cell.

1. Module Usage

SAMCLS

```
CALL SAMCLS (CLASS, CHISQR, LIKELY, COVMTX, VECPRD, SCAPRD,
             CONST, SAMCOR, SAMSUM, SAMPTS, NOFET3, MTXSIZ,
             NOPOOL, RETLIK)
```

Input Arguments:

COVMTX (VARSZ3, NOPOOL)	R*4 An array dimensioned (channels * (channels + 1)/2) * number of classes, containing the upper half of the triangular covariance matrices for each of the classes used for classification.
VECPRD (NOFET3, NOPOOL)	R*4 An array dimensioned channels * classes containing the vector product of the classes for each channel from the inverse covariance matrix and the determinant of the covariance matrix.
SCAPRD (NOPOOL)	R*4 An array dimensioned classes for the scalar product of the mean and covariance matrices for each class.
CONST (NOPOOL)	R*4 An array dimensioned "classes" words containing the constant term used consisting of an expression.
SAMSUM (NOFET3)	R*4 Sum of data values for all points in cell for given channel.
SAMPTS	I*4 Number of points per cell.
NOFET3	I*4 Number of channels.
MTXSIZ	I*4 Number of words needed to store

half of triangular covariance matrix.

NOPOOL I*4 Number of classification pools.

RETLIK L*4 Indication of whether or not likelihoods are to be returned in LIKELY.

Output Arguments:

CLASS I*4 Class number of cell.

CHISQR R*4 Resultant Chi-square value from LOGLIK and classification.

LIKELY (NOPOOL) R*4 Log likelihood of cell belonging to pools.

2. Internal Description

SAMCLS calls LOGLIK once for each class and is returned a log likelihood for the cell for each class. If the likelihood is greater than the previously stored, the new likelihood is saved with the class number and, when done, this class and value are returned.

3. Input Description

Not applicable.

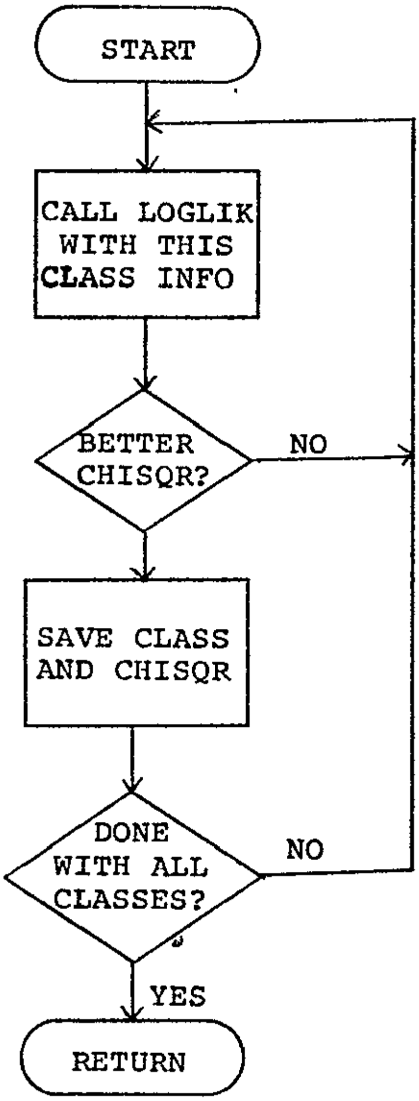
4. Output Description

Not applicable.

5. Supplemental Information

Not applicable.

6. Flowchart


START
CALL LOGLIK WITH THIS CLASS INFO
BETTER CHISQR?
NO
SAVE CLASS AND CHISQR
DONE WITH ALL CLASSES?
NO
YES
RETURN

```
  FORTRAN IV G LEVEL  20.7                SAMCLS             DATE = 77133         21008018           PAGE 0001
FILE SAMCLS
                C   SAMCLS      LARS   XXXX                                                    SAM00010
                C               PROVIDES LIKELIHCOD CLASSIFICATION OF CELLS                    SAMCC020
                C                                                                              SAMCC030
                C*******************************************************************SAM0C040
                C                                                                              SAM0C050
                C               REVISED 05/01/77 BY P.D. ALENDUFF                              SAMCC060
                C                                                                              SAMCC070
                C                                                                              SAM0C080
                C*******************************************************************SAM00C90
 0001                 SUBROUTINE SAMCLS(CLASS,CHISQR,LIKELY,ICLSCV,VECPRD,SCAPRD,CONST,  SAMC010C
                     & SAMCOR, SAMSUM, SAMPTS, VECSIZ, MTXSIZ, NUMCLS, RETLIK)           SAMCC11C
                C                                                                        SAM00120
 0002                 IMPLICIT INTEGER * 4 (A-Z)                                         SAMC0130
                C                                                                        SAM00140
 0003                 REAL * 4 CHISQR,ICLSCV(MTXSIZ,NUMCLS),VECPRD(VECSIZ,NUMCLS),        SAM00150
                     & SCAPRD(NUMCLS),CONST(NUMCLS),SAMCOR(MTXSIZ),SAMSUM(VECSIZ),A,B,C,SAM00160
                     & LIKELY(NUMCLS)                                                    SAM0017G
                C                                                                        SAM00180
 C004                 LOGICAL * 4 RETLIK                                                 SAM00190
                C                                                                        SAM00200
 0005                 DO 2 I=1,NUMCLS                                                    SAM00210
 00C6                 CALL LOGLIK(B,C,ICLSCV(1,I),VECPRD(1,I),SCAPRD(I),CONST(I),        SAM00220
                     & VECSIZ,SAMCOR,SAMSUM,SAMPTS)                                      SAMC0230
 0007                 IF(RETLIK) LIKELY(I)=B                                             SAM00240
 0008                 IF (I.EQ.1) GO TO 1                                                SAM0C250
 0009                 IF (B.LE.A) GO TO 2                                                SAM0C260
 0010               1 A=B                                                                SAMC0270
 0011                 CHISQR = C                                                         SAMC0280
 0012                 CLASS = I                                                          SAM00290
 0013               2 CONTINUE                                                           SAM00300
 0014                 RETURN                                                             SAM00310
 0015                 END                                                                SAM00320
```

LARS Program Abstract ____

MODULE IDENTIFICATION

Module Name: LOGLIK Function Name: SECSUP

Purpose: Produce Log likelihood of a cell

System/Language: CMS/FORTRAN

Author: P. D. Alenduff Date: 4/13/77

Latest Revisor: ____ Date: ____

MODULE ABSTRACT

LOGLIK produces a log likelihood for a class in the input statistics.

1. Module Usage

LOGLIK

```
CALL LOGLIK (LIKELY, CHISQR, ICLSCV, VECPRD, SCAPRD, CONST,
             DIMEN, SAMCOR, SAMSUM, SAMPTS)
```

Input Arguments:

ICLSCV (MTXSIZ) R*4 Class covariance matrix from statistics, dimensioned channels * (channels + 1)/2.

VECPRD (DIMEN) R*4 Vector product of class covariance matrix determinant and means dimensioned "channels" words.

SCAPRD R*4 The scalar product term for the class being considered.

CONST R*4 The constant term derived in SECPRT.

DIMEN I*4 Number of channels of data in cell information.

SAMCOR (MTXSIZ) R*4 The cross product terms from the cells data values dimensioned as ICLSCV.

SAMSUM (DIMEN) R*4 The sum of the data values for each channel for the cell.

SAMPTS I*4 Number of data points in a cell, actually (cell width)^2.

Output Arguments:

LIKELY R*4 Log likelihood value for the cell.

CHISQR R*4 Chi-square value returned for the cell.

LOGLIK is called to return a log likelihood and a Chi-square value for the cell being considered.

2. Internal Description

The Chi-square value is calculated by the equation:

$$\text{CHISQR} = \text{tr}(C_j^{-1} \overset{m}{\Sigma}\, Y_i Y_i') - 2M_j'\, C_j^{-1} \overset{m}{\Sigma}\, Y_i + mM_j'\, C_j^{-1}\, M_j$$

and the log likelihood is calculated by:

$$\text{LIKELY} = -.5\ (m \cdot \ln|2\pi\ C_j| + \text{CHISQR})$$

Where:

C_j is the class covariance matrix for class j.

M_j is the mean vector for class j.

Y_i is the channel response vector for pixel i of the cell.

m is the number of pixels in the cell.

3. Input Description

Not applicable.

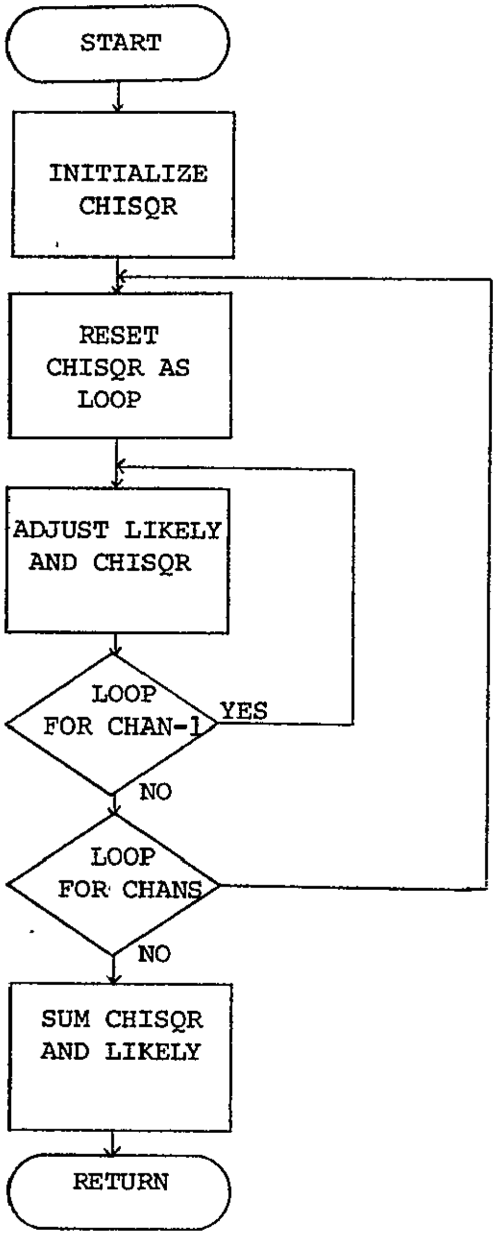
4. Output Description

Not applicable.

5. Supplemental Information

Not applicable.

6. Flowchart


START
INITIALIZE CHISQR
RESET CHISQR AS LOOP
ADJUST LIKELY AND CHISQR
LOOP FOR CHAN-1
YES
NO
LOOP FOR CHANS
NO
SUM CHISQR AND LIKELY
RETURN

```
 FORTRAN IV G LEVEL  20.7               LOGLIK             DATE = 77129          15025029                PAGE 0001
FILE LOGLIK
            C LOGLIK    LARS   XXXX                                                   LOG00010
            C                                                                         LOG00020
            C***********************************************************************LOG00030
            C                                                                         LOG00040
 0001             SUBROUTINE LOGLIK(LIKELY,CHISQR,ICLSCV,VECPRD,SCAPRD,CONST,DIMEN,  LOG00050
                 & SAMCOR, SAMSUM, SAMPTS)                                            LOG00060
            C                                                                         LOG00070
            C                                                                         LOG00080
            C     LOCAL VARIABLES                                                     LOG00090
            C                                                                         LOG00100
            C DIMEN = NUMBER OF CHANNELS USED                                          LOG00110
            C SAMPTS = NUMBER OF PTS PER CELL                                          LOG00120
            C CONST = LN(2*PI*C(N)) WHERE C(N) IS COVARIANCE MATRIX                    LOG00130
            C         FOR CLASS N (N IS DETERMINED BY SAMCLS)                          LOG00140
            C SAMSUM = SUM OF Y(I)                                                     LOG00150
            C SAMCOR = SUM OF Y(I)*Y(I) TRANSPOSED                                     LOG00160
            C ICLSCV = C(J) ** -1                                                      LOG00170
            C SCAPRD = M(J)*C(J) TRANSPOSED *M(J) WHERE C(J) = CLASS                   LOG00180
            C          COVARIANCE MATRIX AND M(J) = CLASS MEAN MATRIX                  LOG00190
            C VECPRD = M(J) TRANSPOSE * C(J) ** -1                                     LOG00200
            C                                                                         LOG00210
            C***********************************************************************LOG00220
            C                                                                         LOG00230
 0002             IMPLICIT INTEGER * 4 (A-Z)                                          LOG00240
 0003             REAL * 4 LIKELY,CHISQR,ICLSCV(1),VECPRD(DIMEN),SCAPRD,CONST,        LOG00250
                 &         SAMCOR(1), SAMSUM(DIMEN)                                   LOG00260
 0004             K = 1                                                               LOG00270
 0005             LIKELY = 0                                                          LOG00280
            C                                                                         LOG00290
            C SET CHISQR = .5 * NO. OF POINTS/CELL * LN(2*PI*DET(CLS COVAR MTX))      LOG00300
            C                                                                         LOG00310
 0006             CHISQR = .5 * SAMPTS * SCAPRD-VECPRD(1)*SAMSUM(1)+.5*ICLSCV(1)*SAMLOG00320
                 1COR(1)                                                              LOG00330
            C                                                                         LOG00340
 0007             DO 2 I = 2,DIMEN                                                    LOG00350
 0008             CHISQR = CHISQR - VECPRD(I)*SAMSUM(I)                               LOG00360
 0009             L = I - 1                                                           LOG00370
 0010             DO 1 J=1,L                                                          LOG00380
 0011             K = K+1                                                             LOG00390
 0012           1 CHISQR = CHISQR + ICLSCV(K)*SAMCOR(K)                               LOG00400
 0013             K = K + 1                                                           LOG00410
 0014           2 LIKELY = LIKELY + ICLSCV(K) * SAMCOR(K)                             LOG00420
 0015             CHISQR = CHISQR + .5*LIKELY                                         LOG00430
            C                                                                         LOG00440
            C AT THIS POINT, CHISQR = .5 * SAMPTS * SCAPRD + .5 * TRACE (ICLSCV*       LOG00450
            C                         SAMCOR) - VECPRD*SAMSUM                          LOG00460
            C                                                                         LOG00470
 0016             LIKELY = -SAMPTS*CONST - CHISQR                                     LOG00480
 0017             CHISQR = CHISQR + CHISQR                                            LOG00490
 0018             RETURN                                                              LOG00500
 0019             END                                                                 LOG00510
```

LARS Program Abstract ______

MODULE IDENTIFICATION

Module Name: SECRD Function Name: SECSUP

Purpose: Read data arrays from intermediate tape.

System/Language: CMS/Fortran

Author: P. D. Alenduff Date: 4/12/77

Latest Revisor: ______ Date: ______

MODULE ABSTRACT

SECRD reads data arrays from the intermediate tape for annexation processing by SECHO.

1. Module Usage

SECRD

CALL SECRD (COVMTX, AVEMTX)

Input Arguments:

COVMTX (VARSZ3, NOPOOL) R*4 An array dimensioned (channels * (channels + 1))/2 fullwords for storing the covariance matrices of classes used in classification.

AVEMTX (NOPET3, NOPOOL) R*4 An array dimensioned channels by classes words for storing the class means of classes used in the classification.

Output Arguments:

Same as above.

SECRD is called to read one reocrd from the intermediate tape. COVMTX and AVEMTX are read in an unformatted fashion from the unit INPUT. This program uses common blocks GLOCOM and SECCOM.

2. Internal Description

See above.

3. Input Description

SECRD reads from unit INPUT a record type 4 from the classification format file.

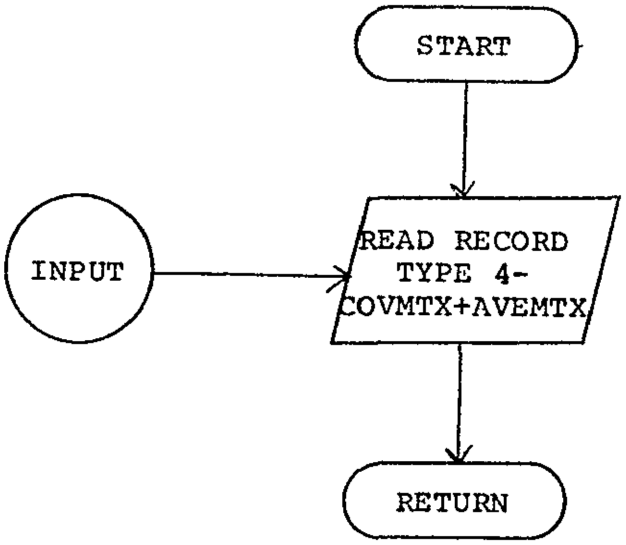
4. Output Description

Not applicable.

5. Supplemental Information

See LARSYS Systems Manual for format of classification results file.

6. Flowchart



FILE SECRD

```
              C    SECRD    LARS    XXXX                                            SEC00010
              C                                                                     SEC00020
              C*********************************************************************SEC00030
              C                                                                     SEC00040
              C SECPRD  ROUTINE TO READ DATA FROM INPUT INTERMEDIATE TAPE             SEC00050
              C                                                                     SEC00060
              C*********************************************************************SEC00070
              C                                                                     SEC00080
 0001               SUBROUTINE SECRD (COVMTX, AVEMTX)                               SEC00090
 0002               IMPLICIT INTEGER * 4 (A-Z)                                      SEC00100
              C                                                                     SEC00110
 0003               COMMON /GLOCOM/ BLANK, CARD(20), CHKOUT, COPFIL, CLASSR, CLASSX, SEC00120
                  1     CLUSTX, CONPUT, CPYOUT, CRDRDR, CRDSEQ, DATAPE,             SEC00130
                  2     DUPLTP, DUPRUN, ERRMSG, FBPNT,                              SEC00140
                  3     FILESV, FLDBND, HDATA, HEAD(88), ID(200), IMAGEX,           SEC00150
                  4     IMARK, KEYBD, MAPTAP, MAXCHA, MAXCLS,                       SEC00160
                  5     PAGSIZ, PNCH, POINT, PRESUX, PRNTR, READIN,                 SEC00170
                  6     RESTRT, RUNFIL, RUNTAB(10,3),                               SEC00180
                  7     SDATA, SEPARX, SEPTPX, SPARE(10), TEMPAS(30),               SEC00190
                  8     TPSTAT(6), TTFLDX ,TYPEWR,                                  SEC00200
                  9     TOP, ARRAY(12500)                                           SEC00210
              C                                                                     SEC00220
 0004               REAL * 8 ARRAY                                                  SEC00230
 0005               REAL * 4 FRQCAL(5,30)                                           SEC00240
 0006               INTEGER * 4 COMENT(16), DATE(5), HED1(16), HED2(16), TIME(5)    SEC00250
 0007               INTEGER * 2 BLANK2                                              SEC00260
 0008               LOGICAL * 4 CHKOUT                                              SEC00270
 0009               LOGICAL * 1 BLANK1                                              SEC00280
 0010               EQUIVALENCE (DATSAV,ID(1)), (CURRUN,ID(3)), (FRQCAL(1),ID(51)), SEC00290
                  1         (HED1(1),HEAD(8)), (DATE(1),HEAD(26)), (HED2(1),HEAD(39)),SEC00300
                  2         (TIME(1),HEAD(58)), (COMENT(1),HEAD(72)),               SEC00310
                  3         (MAPSAV,TPSTAT(1)),                                     SEC00320
                  4         (SEPSCR,TPSTAT(2)), (DUPIN,TPSTAT(3)), (DASTAT,TPSTAT(4)),SEC00330
                  5         (COPSER,TPSTAT(5)), (TRAOUT,TPSTAT(6)),                 SEC00340
                  6         (BLANK,BLANK2,BLANK1)                                   SEC00350
              C                                                                     SEC00360
              C                                                                     SEC00370
 0011               COMMON /SECCOM/ BUFROZ,CSELEC,CSET(3,30),CSET3(3,30),THRES,CELSIZ,SEC00010
                  1 CELWTH,INFO(17),INPUT, INTTAP, INTFIL, JPTS, LINES, NOCLS, NOFET,SEC00020
                  2 NOFET3,NOPOOL,NWORD,NWORD2, NVR, OUTPUT, POLNAM(2,60), PREFIX(2),SEC00030
                  3 PTS, RQTAPE, RQFILE, SYMCNT, VARSZ3, CSEL(30), CSEL3(30),       SEC00040
                  3 FETVEC(30), FETVC3(30), POLPTR(2,60), POLSTK(60),               SEC0005C
                  4 CDFLAG, CLSMAP, CSET1(3,30), OBJMAP, PHASE1, PHASE2, POLNM1(60), SEC00060
                  5 PRSTAT, SYM(60), SYMMTX(60)                                     SEC00070
              C                                                                     SEC00080
 0012               REAL * 4 CSELEC, CSET, CSET3, THRES                             SEC00090
              C                                                                     SEC00100
 0013               INTEGER * 2 CSEL, CSEL3, FETVEC, FETVC3, POLPTR, POLSTK         SEC00110
              C                                                                     SEC00120
 0014               LOGICAL * 1 CDFLAG, CLSMAP, CSET1, OBJMAP, PHASE1, PHASE2, POLNM1,SEC00130
                  1 PRSTAT                                                          SEC00140
              C                                                                     SEC00150
              C                                                                     SEC00150
              C                                                                     SEC00530
              C                                                                     SEC00540
              C                                                                     SEC00550
 0015               REAL * 4 COVMTX(VARSZ3,NOPOOL), AVEMTX(NOFET3,NOPOOL)           SEC00560
              C                                                                     SEC00570
 0016               READ (INPUT) PREFIX, COVMTX, AVEMTX                             SEC00580
 0017               RETURN                                                          SEC00590
 0018               END                                                             SEC00600
```

REPRODUCIBILITY OF THE
ORIGINAL PAGE IS POOR

LARS Program Abstract

MODULE IDENTIFICATION

Module Name: SECPRT Function Name: SECSUP

Purpose: Statistics printing and output routine for SECHO

System/Language: CMS/FORTRAN

Author: P. D. Alenduff Date: 4/12/77

Latest Revisor: Date:

MODULE ABSTRACT

SECPRT produces tabular statistics for user information as well as carrying out minor data array processing.

PURDUE UNIVERSITY
Laboratory for Applications of Remote Sensing
1220 Potter Drive
West Lafayette, Indiana 47906

each channel. Then the information concerning classes used is printed and WRTMTX is called to print the correlation matrix. The determinant of the covariance matrices is formed by a call to SAMINV, and a check is made to determine if any determinants were zero or negative. Then SMMULT is called to produce the array SCAPRD, the scalar product, and a calculation is performed to produce the constant array. Execution is terminated if any of the determinants were zero or negative. This routine uses common blocks GLOCOM and SECCOM.

3. Input Description

Not applicable.

4. Output Description

A record type 4 of the classification format output file is written to unit OUTPUT.

Also, statistics information is produced on unit PRNTR.

5. Supplemental Information

Not applicable.

1. Module Usage

SECPRT

CALL SECPRT (COVMTX, AVEMTX, SCAPRD, DETCOV, CONST, WORK)

Input Arguments:

COVMTX (VARSZ3, NOPOOL)	R*4 Covariance matrix for each class.
AVEMTX (NOFET3, NOPOOL)	R*4 Mean matrix for each class.

Output Arguments:

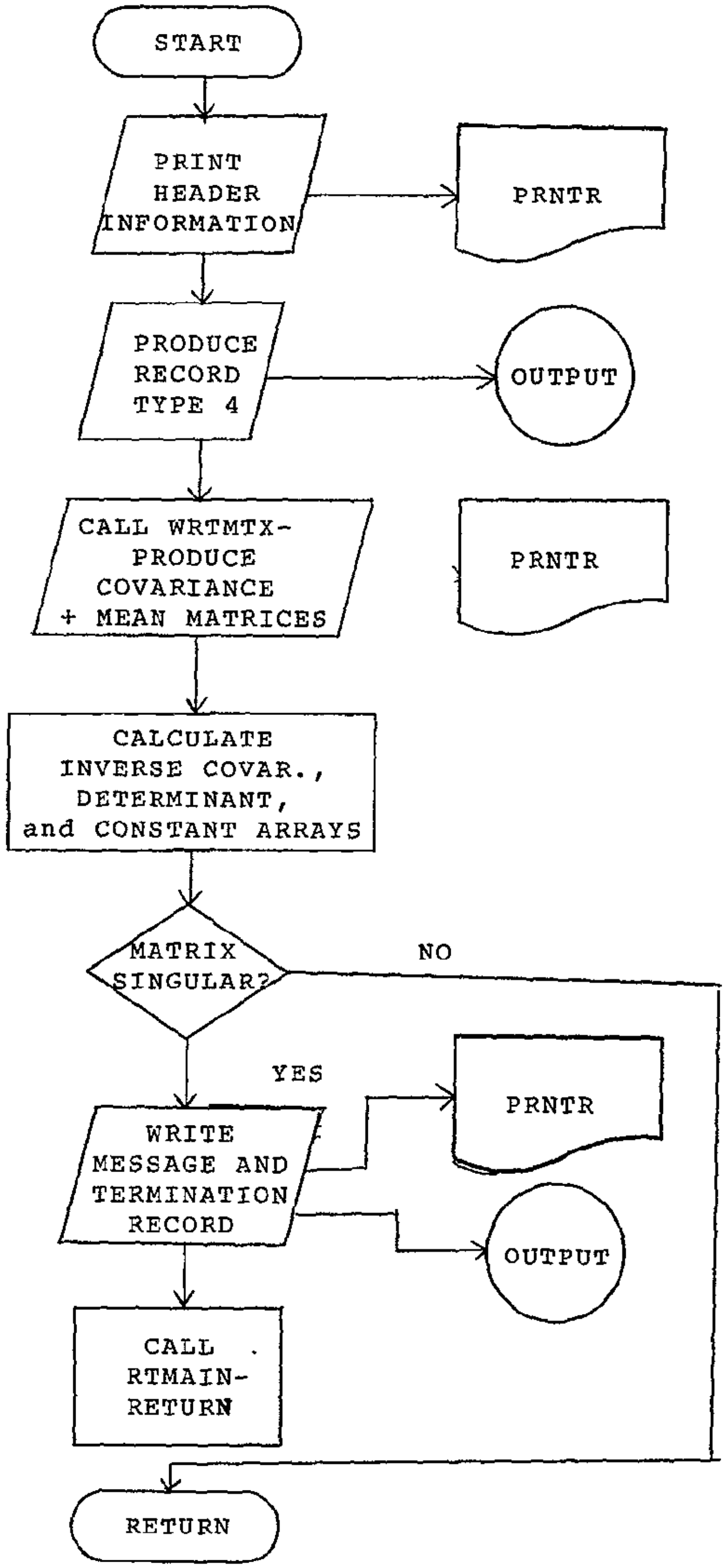
SCAPRD (NOPOOL)	R*4 Scalar product for each class.
DETCOV (NOPOOL)	R*4 Determinant of the covariance matrix for each class.
CONST (NOPOOL)	R*4 Constant formed for computation in classification for each class.
WORK (NOFET3 * (NOFET3 + 2))	R*4 Buffer array for inverting covariance matrix at least channels by channels + 2 words.

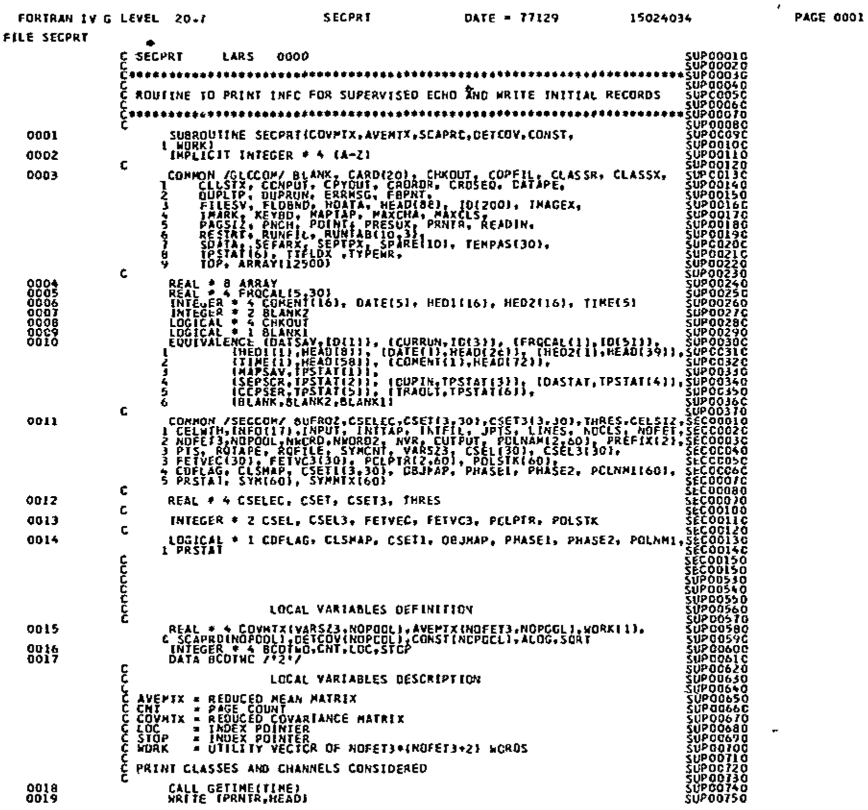
SECPRT is called to print the statistics information to be used for classification, if requested, and produce the record type 4 of a classification results format file for SECHO. Also the determinant, scalar product, and constant arrays, DETCOV, SCAPRD, and CONST, are produced for later usage.

2. Internal Description

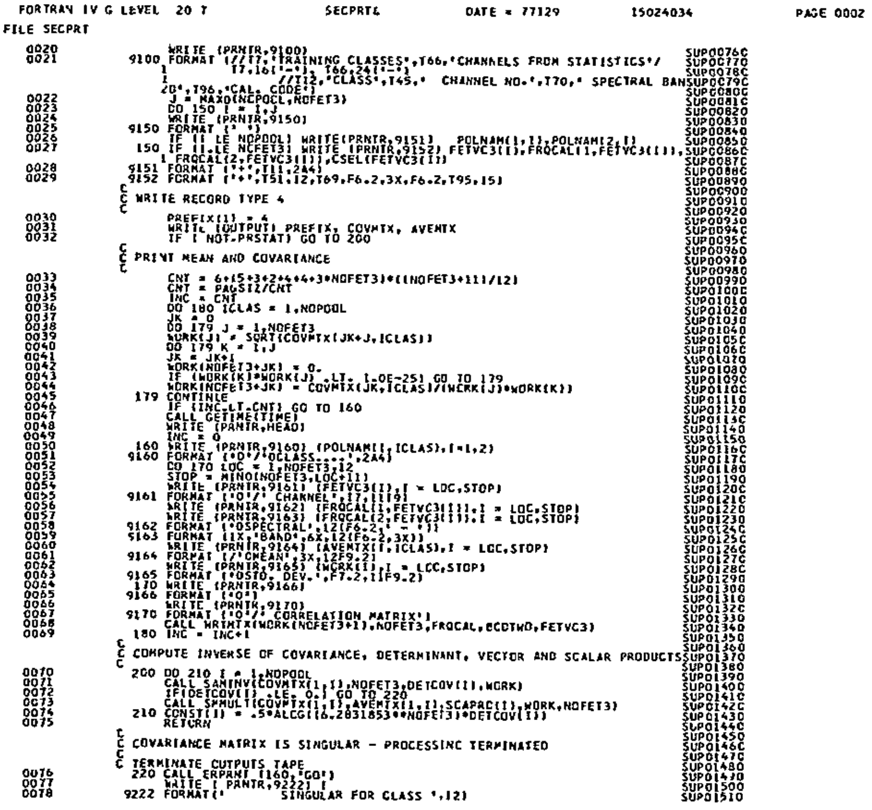
Header information is printed on the Unit PRNTR followed by the writing of record type 4 on the unit OUTPUT. This classification format record contains COVMTX and AVEMTX, the covariance and mean matrices used in the classification for

6. Flowchart


START
PRINT HEADER INFORMATION
PRNTR
PRODUCE RECORD TYPE 4
OUTPUT
CALL WRTMTX- PRODUCE COVARIANCE + MEAN MATRICES
PRNTR
CALCULATE INVERSE COVAR., DETERMINANT, and CONSTANT ARRAYS
MATRIX SINGULAR?
NO
YES
WRITE MESSAGE AND TERMINATION RECORD
PRNTR
OUTPUT
CALL RTMAIN- RETURN
RETURN



```
FORTRAN IV G LEVEL  20.7                SECPRT              DATE = 77129          15024034              PAGE 0001
FILE SECPRT
                    C SECPRT     LARS   0000                                                            SUP00010
                    C                                                                                   SUP00020
                    C*******************************************************************************SUP0003C
                    C                                                                                   SUP00040
                    C ROUTINE TO PRINT INFC FOR SUPERVISED ECHO AND WRITE INITIAL RECORDS               SUPC005C
                    C                                                                                   SUP0006C
                    C*******************************************************************************SUP00070
                    C                                                                                   SUP0008C
  0001                    SUBROUTINE SECPRT(COVPIX,AVENTX,SCAPRC,DETCOV,CONST,                          SUPOC09C
                        1 WORK)                                                                         SUP0010C
  0002                    IMPLICIT INTEGER * 4 (A-Z)                                                    SUP00110
                    C                                                                                   SUP00120
  0003                    COMMON /GLCCOM/ BLANK, CARD(20), CHKOUT, COPFIL, CLASSR, CLASSX,              SUPC013C
                         1    CLLSTX, CCNPUT, CPYOUT, CRORDR, CROSEQ, DATAPE,                           SUP00140
                         2    DUPLTP, DUPRUN, ERRMSG, FBPNT,                                            SUP00150
                         3    FILESV, FLDBND, HDATA, HEAD(88), ID(200), IMAGEX,                         SUP0016C
                         4    IMARK, KEYBD, MAPTAP, MAXCHA, MAXCLS,                                     SUP0017C
                         5    PAGSIZ, PNCH, POINT, PRESUX, PRNTR, READIN,                               SUP0018C
                         6    RESTRT, RUNFIL, RUNTAB(10,3),                                             SUP0019C
                         7    SDATA, SEFARX, SEPTPX, SPARE(10), TEMPAS(30),                             SUPC020C
                         8    TPSTAT(6), TIFLDX ,TYPEWR,                                                SUP0021C
                         9    TOP, ARRAY(12500)                                                         SUP00220
                    C                                                                                   SUP00230
  0004                    REAL * 8 ARRAY                                                                SUP00240
  0005                    REAL * 4 FRQCAL(5,30)                                                         SUP0025C
  0006                    INTEGER * 4 COMENT(16), DATE(5), HED1(16), HED2(16), TIME(5)                  SUP00260
  0007                    INTEGER * 2 BLANK2                                                            SUP0027C
  0008                    LOGICAL * 4 CHKOUT                                                            SUP0028C
  0009                    LOGICAL * 1 BLANK1                                                            SUP00290
  0010                    EQUIVALENCE (DATSAY,ID(1)), (CURRUN,ID(3)), (FRQCAL(1),ID(5)),                SUP0030C
                         1          (HED1(1),HEAD(8)), (DATE(1),HEAD(26)), (HED2(1),HEAD(39)),SUPC031C
                         2          (TIME(1),HEAD(58)), (COMENT(1),HEAD(72)),                           SUPC032C
                         3          (MAPSAV,TPSTAT(1)),                                                 SUP0033C
                         4          (SEPSCR,TPSTAT(2)), (DUPIN,TPSTAT(3)), (DASTAT,TPSTAT(4)),SUP00340
                         5          (CCPSER,TPSTAT(5)), (TRAOLT,TPSTAT(6)),                             SUPOC350
                         6          (BLANK,BLANK2,BLANK1)                                               SUP0036C
                    C                                                                                   SUP00370
  0011                    COMMON /SECCOM/ BUFROZ,CSELEC,CSET(3,30),CSET3(3,30),THRES,CELSIZ,SEC00010
                        1 CELWTH,INFO(17),INPUT, INITAP, INIFIL, JPTS, LINES, NOCLS, NOFET,SECC002C
                        2 NOFET3,NOPOOL,NWCRD,NWORD2, NVR, CUTPUT, POLNAM(2,60), PREFIX(2),SEC0003C
                        3 PTS, ROTAPE, ROFILE, SYMCNT, VARSZ3, CSEL(30), CSEL3(30),                     SECOC040
                        3 FETVEC(30), FETVC3(30), PCLPTR(2,60), POLSTK(60),                             SECCC05C
                        4 CDFLAG, CLSMAP, CSET1(3,30), OBJMAP, PHASE1, PHASE2, PCLNM1(60),              SECOCC6C
                        5 PRSTAT, SYM(60), SYMMTX(60)                                                   SEC0007C
                    C                                                                                   SEC00080
  0012                    REAL * 4 CSELEC, CSET, CSET3, THRES                                           SEC00070
                    C                                                                                   SEC00100
  0013                    INTEGER * 2 CSEL, CSEL3, FETVEC, FETVC3, PCLPTR, POLSTK                       SEC0011C
                    C                                                                                   SEC00120
  0014                    LOGICAL * 1 CDFLAG, CLSMAP, CSET1, OBJMAP, PHASE1, PHASE2, POLNM1,SEC0013C
                        1 PRSTAT                                                                        SEC0014C
                    C                                                                                   SEC00150
                    C                                                                                   SEC00150
                    C                                                                                   SUP00530
                    C                                                                                   SUP00540
                    C                                                                                   SUP00550
                    C                        LOCAL VARIABLES DEFINITION                                 SUP00560
                    C                                                                                   SUP00570
  0015                    REAL * 4 COVMTX(VARSZ3,NOPOOL),AVEMTX(NOFET3,NOPOOL),WORK(1),                 SUP00580
                        C SCAPRD(NOPOOL),DETCOV(NOPCOL),CONST(NCPOCL),ALOG,SORT                         SUP0059C
  0016                    INTEGER * 4 BCDTWO,CNT,LOC,STOP                                               SUP0060C
  0017                    DATA BCDTWC /'2'/                                                             SUP0061C
                    C                                                                                   SUP00620
                    C                        LOCAL VARIABLES DESCRIPTION                                SUP00630
                    C                                                                                   SUP00640
                    C AVEMTX = REDUCED MEAN MATRIX                                                      SUP00650
                    C CNT    = PAGE COUNT                                                               SUP0066C
                    C COVMTX = REDUCED COVARIANCE MATRIX                                                SUP00670
                    C LOC    = INDEX POINTER                                                            SUP00680
                    C STOP   = INDEX POINTER                                                            SUP00690
                    C WORK   = UTILITY VECTCR OF NOFET3*(NOFET3+2) WORDS                                SUP00700
                    C                                                                                   SUP00710
                    C PRINT CLASSES AND CHANNELS CONSIDERED                                             SUPOC720
                    C                                                                                   SUP00730
  0018                    CALL GETIME(TIME)                                                             SUP00740
  0019                    WRITE (PRNTR,HEAD)                                                            SUP00750
```



```
FORTRAN IV G LEVEL  20 7                SECPRT&             DATE = 77129          15024034              PAGE 0002
FILE SECPRT
  0020                    WRITE (PRNTR,9100)                                                            SUP0076C
  0021              9100 FORMAT (//T7,'TRAINING CLASSES',T66,'CHANNELS FROM STATISTICS'/                SUPOC770
                        1           T7,16('-'), T66,24('-')                                             SUP0078C
                        1                 //T12,'CLASS',T45,'  CHANNEL NO.',T70,' SPECTRAL BANSUPOC79C
                        2D',T96,'CAL. CODE')                                                            SUPC080C
  0022                    J = MAXO(NCPOCL,NOFET3)                                                       SUP0081C
  0023                    DO 150 I = 1,J                                                                SUP00820
  0024                    WRITE (PRNTR,9150)                                                            SUP00830
  0025              9150 FORMAT (' ')                                                                   SUP00840
  0026                    IF (I LE NOPOOL) WRITE(PRNTR,9151)   POLNAM(1,I),POLNAM(2,I)                  SUP00850
  0027               150 IF (I.LE NCFET3) WRITE (PRNTR,9152) FETVC3(I),FRQCAL(1,FETVC3(I)),SUPC086C
                        1 FRQCAL(2,FETVC3(I)),CSEL(FETVC3(I))                                           SUP0087C
  0028              9151 FORMAT ('+',T11,2A4)                                                           SUP0088C
  0029              9152 FCRMAT ('+',T51,I2,T69,F6.2,3X,F6.2,T95,I5)                                    SUP00890
                    C                                                                                   SUPOC900
                    C WRITE RECORD TYPE 4                                                               SUP00910
                    C                                                                                   SUP00920
  0030                    PREFIX(1) = 4                                                                 SUP00930
  0031                    WRITE (OUTPUT) PREFIX, COVMTX, AVEMTX                                         SUP0094C
  0032                    IF ( NOT.PRSTAT) GO TO 200                                                    SUP0095C
                    C                                                                                   SUP00960
                    C PRINT MEAN AND COVARIANCE                                                         SUP00970
                    C                                                                                   SUP00980
  0033                    CNT = 6+(5+3+2+4+4+3*NOFET3)*((NOFET3+11)/12)                                 SUP00990
  0034                    CNT = PAGSIZ/CNT                                                              SUP0100C
  0035                    INC = CNT                                                                     SUP01010
  0036                    DO 180 ICLAS = 1,NOPOOL                                                       SUP01020
  0037                    JK = 0                                                                        SUP01030
  0038                    DO 179 J = 1,NOFET3                                                           SUP01040
  0039                    WORK(J) = SQRT(COVMTX(JK+J,ICLAS))                                            SUP0105C
  0040                    DO 179 K = 1,J                                                                SUP0106C
  0041                    JK = JK+1                                                                     SUP01070
  0042                    WORK(NOFET3+JK) = 0.                                                          SUP01080
  0043                    IF (WORK(K)*WORK(J) .LT. 1.0E-25) GO TO 179                                   SUP0109C
  0044                    WORK(NCFET3+JK) = COVMTX(JK,ICLAS)/(WCRK(J)*WORK(K))                          SUP0110C
  0045               179 CONTINUE                                                                       SUP01110
  0046                    IF (INC.LT.CNT) GO TO 160                                                     SUP01120
  0047                    CALL GETIME(TIME)                                                             SUP0113C
  0048                    WRITE (PANTR,HEAD)                                                            SUP01140
  0049                    INC = 0                                                                       SUP01150
  0050               160 WRITE (PRNTR,9160) (POLNAM(I,ICLAS),I=1,2)                                     SUP0116C
  0051              9160 FORMAT ('0'/'0CLASS....',2A4)                                                  SUP0117C
  0052                    DO 170 LOC = 1,NOFET3,12                                                      SUP01180
  0053                    STOP = MINO(NOFET3,LOC+11)                                                    SUP01190
  0054                    WRITE (PRNTR,9161) (FETVC3(I),I = LOC,STOP)                                   SUP0120C
  0055              9161 FORMAT ('0'/' CHANNEL',I7,11I9)                                                SUP0121C
  0056                    WRITE (PRNTR,9162) (FRQCAL(1,FETVC3(I)),I = LOC,STOP)                         SUP01220
  0057                    WRITE (PRNTR,9163) (FRQCAL(2,FETVC3(I)),I = LOC,STOP)                         SUP01230
  0058              9162 FORMAT ('OSPECTRAL',12(F6.2,' - '))                                            SUP0124C
  0059              9163 FORMAT (1X,'BAND',6X,12(F6.2,3X))                                              SUP0125C
  0060                    WRITE (PRNTR,9164) (AVEMTX(I,ICLAS),I = LOC,STOP)                             SUP0126C
  0061              9164 FORMAT (/'OMEAN',3X,12F9.2)                                                    SUP0127C
  0062                    WRITE (PRNTR,9165) (WCRK(I),I = LCC,STOP)                                     SUP0128C
  0063              9165 FORMAT ('OSTD. DEV.',F7.2,11F9.2)                                              SUP01290
  0064               170 WRITE (PRNTR,9166)                                                             SUP01300
  0065              9166 FORMAT ('0')                                                                   SUP01310
  0066                    WRITE (PRNTR,9170)                                                            SUP0132C
  0067              9170 FORMAT ('0'/' CORRELATION MATRIX')                                             SUP01330
  0068                    CALL WRTMTX(WORK(NOFET3+1),NOFET3,FRQCAL,BCDTWO,FETVC3)                       SUP01340
  0069               180 INC = INC+1                                                                    SUP01350
                    C                                                                                   SUP01360
                    C COMPUTE INVERSE OF COVARIANCE, DETERMINANT, VECTOR AND SCALAR PRODUCTSSUP01370
                    C                                                                                   SUP01380
  0070               200 DO 210 I = 1,NOPOOL                                                            SUP01390
  0071                    CALL SANINV(COVMTX(1,I),NOFET3,DETCOV(I),WORK)                                SUP01400
  0072                    IF(DETCOV(I) .LE. 0.) GO TO 220                                               SUP0141C
  0073                    CALL SPMULT(COVPTX(1,I),AVEMTX(1,I),SCAPAC(I),WORK,NOFET3)                    SUP0142C
  0074               210 CONST(I) = .5*ALCG((6.2831853**NOFET3)*DETCOV(I))                              SUP01430
  0075                    RETCRN                                                                        SUP0144C
                    C                                                                                   SUP01450
                    C COVARIANCE MATRIX IS SINGULAR - PROCESSINC TERMINATED                             SUP0146C
                    C                                                                                   SUP0147C
                    C TERMINATE CUTPUTS TAPE                                                            SUP01480
  0076               220 CALL ERPRNT (160,'GO')                                                         SUP01490
  0077                    WRITE ( PRNTR,9222) I                                                         SUP01500
  0078              9222 FORMAT('         SINGULAR FOR CLASS ',I2)                                      SUP01510
  FORTRAN IV G LEVEL  20.7              SECPRT6          DATE = 77129          15024034              PAGE 0003
FILE SECPRT
   0079                IF (OUTPUT .EQ. CLASSR) CALL RTMAIN                                      SUP01520
   0080                CALL TCPAF (OUTPUT)                                                      SUP01530
   0081                I3 = 3                                                                   SUP0154C
   0082                K = -1                                                                   SUP01550
   0083                IO = 0                                                                   SUP01560
   0084                NO = FILESV+1                                                            SUP01570
              C                                                                                 SUP01580
   0085                WRITE (OUTPUT) PREFIX, RQTAPE, NO, I3, K, SERIAL, (IO,I = 1,7)           SUP01590
   0086                CALL TOPEF (OUTPUT)                                                      SUP01600
   0087                CALL TOPRF (OUTPUT)                                                      SUP0161C
   0088                CALL TCPEF (OUTPUT)                                                      SUP0162C
   0089                CALL TOPBF (OUTPUT)                                                      SUP01630
   0090                CALL RTMAIN                                                              SUP01640
   0091                RETURN                                                                   SUP01650
   0092                END                                                                      SUP01660
```