AgRISTARS

SR-L1-00308
JSC-17305
NASA-CR-161053

A Joint Program for Agriculture and Resources Inventory Surveys Through Aerospace Remote Sensing

Supporting Research

June 1981

"AS-BUILT" DESIGN SPECIFICATION FOR PARPLT

M. A. Tompkins

(E82-10101) AS-BUILT DESIGN SPECIFICATION FOR PARPLT (Lockheed Engineering and Management) 114 p HC A06/MF A01 CSCL 02C

N82-22543

Unclas
G3/43 00101

Lockheed Engineering and Management Services Company, Inc.
1830 NASA Road 1, Houston, Texas 77058

Lyndon B. Johnson Space Center
Houston, Texas 77058

SR-L1-00308
JSC-17305

"AS-BUILT" DESIGN SPECIFICATION
FOR
PARPLT

Job Order 71-308

Prepared By
M. A. Tompkins

Approved By

G. L. Clouette, Supervisor
Support Systems Software Section

R. Kent Lonnington, Supervisor
Techniques Development Section

R. A. McClane, Manager
Data Systems Department

T. C. Minter, Manager
Development and Evaluation Department

Prepared By

Lockheed Engineering and Management Services Company, Inc.

For

Earth Observations Division

Space and Life Sciences Directorate

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

June 1981

LEMSCO-16544

1. Report No. JSC-17305, SR-L1-00308	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle "As-Built" Design Specification for PARPLT		5. Report Date June 1981 6. Performing Organization Code SG2
7. Author(s) Mary Ann Tompkins, D. E. Cheng		8. Performing Organization Report No. LEMSCO-16544
9. Performing Organization Name and Address Lockheed Engineering and Management Services Company, Inc., Systems and Services Division Houston, Texas 77058		10. Work Unit No. 11. Contract or Grant No. NAS 9-15800
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Lyndon B. Johnson Space Center Houston, Texas 77058 Dr. G. Badhwar/SG3		13. Type of Report and Period Covered "As-Built" 14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

This document is the "As Built" Design Specification for the PARPLT program which is part of the CLASFYG package. The program produces scatter plots of the greenness profile derived parameters α, β t_0 computed by the CLASFYG program and statistical information concerning the parameters.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Classification Ground truth Universal format	

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	114	

*For sale by the National Technical Information Service, Springfield, Virginia 22161

JSC Form 1424 (Rev Nov 75)

NASA — JSC

CONTENTS

CONTENTS

FIGURES

PARPLT PROGRAM

1.0 SCOPE

This document contains the description of the implementation of the PARPLT program. The purposes of the program are as follows:

(1) To produce scatter plots of the greenness profile derived parameters α, β, and t_0 computed by the CLASFYG program. Alpha is the approximate greenness rise time, beta is the approximate greenness decay time, and t_0 is the spectral crop emergence date.

(2) To produce statistical information concerning the α, β, and t_0.

2.0 APPLICABLE DOCUMENTS

The following documents form a part of this specification:

AD 63-2457-3308-03 Transferring Badhwar Software.

AD NAS 9-15200 Technical Memorandum Format Specifications for LACIE (Phase III) and Accuracy Assessment Computer Data Products.

3.0 SYSTEM DESCRIPTION

3.1 SYSTEM FLOWCHART

The system level data flow diagram for the PARPLT Program is depicted in Figure 3.3.1. A program hierarchy is shown in Figure 3.1.2.

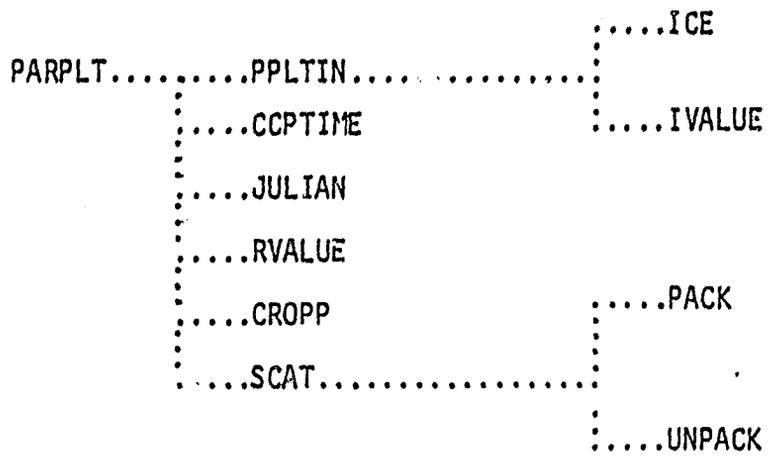


Figure 3.1.2 Hierarchy diagram for the PARPLT Program.

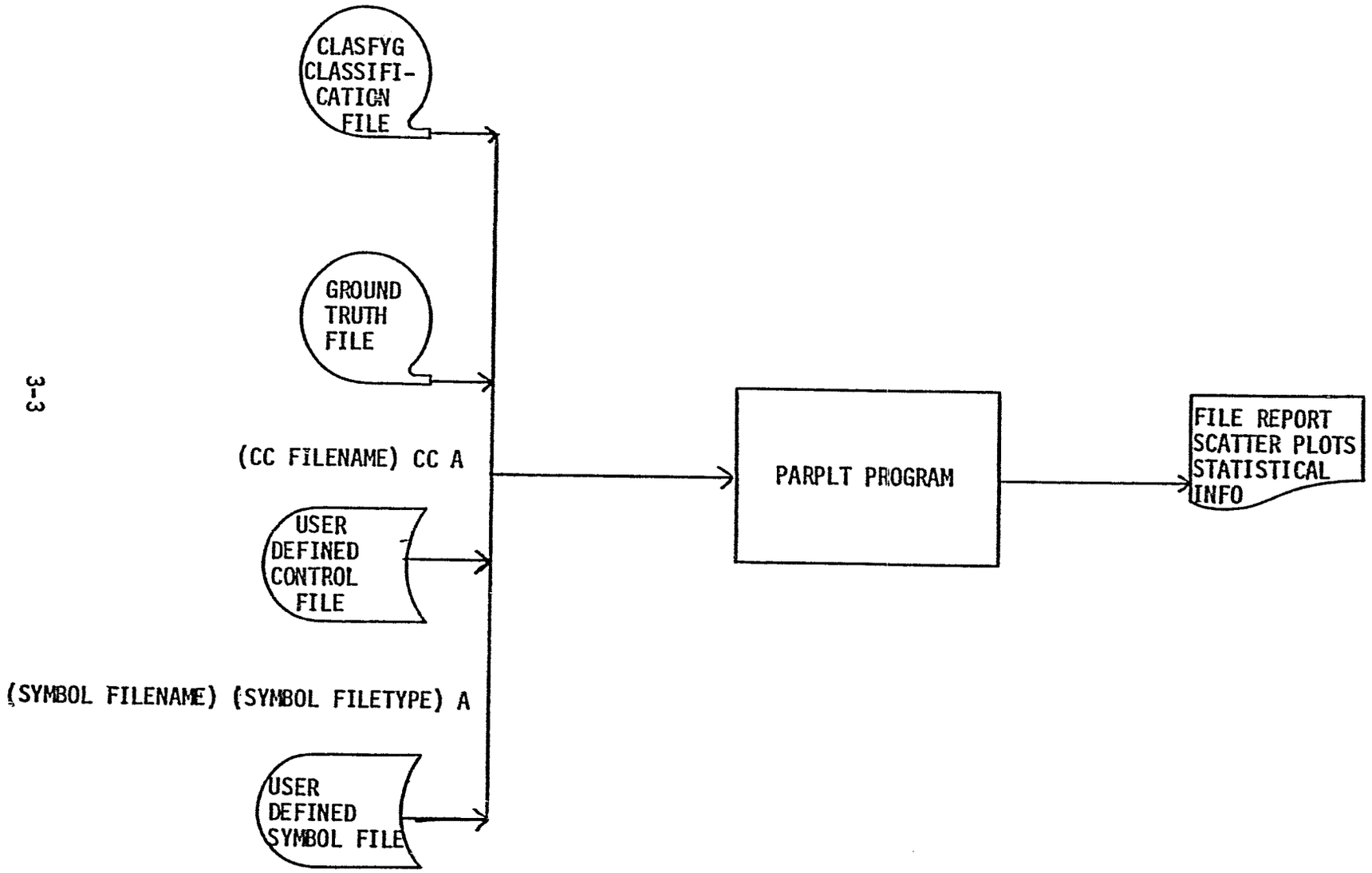


Figure 3.3.1 System level flow diagram for the PARPLT Program.

3.2 HARDWARE DESCRIPTION

The software for the PARPLT Program is operational on the IBM 3031 computer at PURDUE.

3.3 SOFTWARE DESCRIPTION

PARPLT was designed to produce scatter plots of the parameters α, β, and t_0 computed by the CLASFYG Program. Alpha is the approximate greenness rise time, beta is the approximate decay time, and t_0 is the spectral crop emergence date. Three types of plots are produced; namely (1) α vs. β, (2) α vs. t_0, and (3) β vs. t_0. For any given run, plots can be obtained for up to three categories. A category can be a single ground truth class like "spring wheat", or a combination of ground truth classes like "small grains". Ground Truth codes are mapped to specific categories through a "ground truth transformation" table. (See Section 3.4.2 for a complete description of the ground truth transformation table.)

Ground truth data is used in the process of determining if a specific pixel value is among the categories chosen to be plotted. The program processes each pixel in the following manner:

(1) Through the use of the ground truth transformation table a transformed numeric code is assigned to each of the six ground truth sub-pixels which correspond to the given pixel.

(2) The pixel is assigned to a majority category. This majority category is either the category of the first subpixel or is the category which occurs the most among the six sub-pixel categories.

(3) The number of occurance of the majority category is defined to be the purity of the given pixel. This purity is compared to a user specified pixel purity range, and if the number of occurances fall outside of this range then there is no further processing of this pixel.

(4) If the majority category is greater than the number of categories specified by the user then there is no further processing of this pixel.

(5) If the pixel value of the classification parameter α is zero then there is no further processing of this pixel.

(6) Each of the plots are udpated according to the classification parameter to reflect the occurance of an additional pixel.

(7) Statistical variables are updated.

When each pixel has been processed the program computes statistical information for all pixels, and the plots and statistical information are printed.

3.4 FILE DESCRIPTIONS

3.4.1 INPUT FILE(S)

Two input data files are required to execute the PARPLT program. The two files and their descriptions are as follows:

PARPLT accepts as input, a classification file output from the CLASFYG PROGRAM. This classification file consist of 118 records. The first record is a header record consisting of 3060 bytes. The header record follows universal format requirements. Records 2 - 118 contain data records. Each data record consists of 196 pixel vectors. Each pixel vector consists of four coefficients: α, β, t_o and χ^2. Each coefficient contains a signed integer value stored in a 32 -bit full word. (For a complete description of this file see "AS-BUILT" DESIGN SPECIFICATION FOR CLASFYG).

The Ground Truth file is in UNIVERSAL FORMAT with one channel per physical record. There are 351 records of length 540 8 bit-bytes. The contents of each byte have been biased with -128 and are stored in 8 bit twos-complement notation. (For a complete description of the file see Earth Resources Data Format Control Book, Volume 1, PHØ-TR543.)

3.4.2 USER DEFINED FILE (SYMBOL FILENAME) (SYMBOL FILETYPE) A

This file is used to specify (1) the pixel purity range and (2) the ground

cceptable pixel purity for any given
r a complete description of pixel purity
transformation allows the user to map
ssification Categories". These
cal order of the categories entered on
rd in red control card file (See Section 3.4.3
concerning the use of this card). Therefore, any
ent of crop code to a category greater than the total number of
ories enta signifies that the code is of no interest. If a crop
not assigned to a category the program assigns as a default a
y of 6. The last entry in the symbol file must be 0 0 0.

The first entry in the symbol file is as follows:

Column	1	2	3	
Format	I1	1X	I1	Purpose
	start		end	Defines the start and end of a range of pixel purity.

The remaining entries in the symbol file are as follows:

Columns	1-5	6-10	11-15	
Format	I5	I5	I5	Purpose
	start	end	category	Defines the start and end of a range of crop values assigned to a category.
	0	0	0	Signifies the end of Ground Truth transformation information.

The following is an example of a symbol file.

```
 1  6
       1     10     1
      11     20     2
      21     79     3
      80     80     6
      84     86     2
     127    127     3
     164    164     6
     165    165     3
       0      0     0
```

3.4.3 USER DEFINED FILE (FILENAME) CC A

This Control Card file is used to specify inputs to the PARPLT program.

The inputs are on card image records. Each record consists of (1) a keyword which is ten characters or less and begins in the first card column and (2) input parameters in columns 11 through 72. The following description lists the keywords and describes the corresponding inputs.

KEYWORD	ACCEPTABLE INPUTS	DESCRIPTION
ALPHARNG	Pair of postive numbers	This card defines the minimum and maximum values to be plotted for the Alpha parameter. The numbers are assumed to be positive real numbers and any integers will be converted. The numbers are separated by any delimiter except '.'.
BETARNG	Pair of positive numbers	This card defines the minimum and maximum values to be plotted for the beta parameter. The numbers are assumed to be positive real numbers and any integers will be converted. The numbers are separated by any delimiter except '.'.
TØRNG	Pair of positive numbers	This card defines the minimum and maximum values to be plotted for the t_0 parameter. The numbers are assumed to be positive real numbers and any integers will be converted. The numbers are separated by any delimiter except '.'.
CROP	1-3 crop names	This card gives names(s) to the categories to be plotted. The categories are separated by any non character deliminter except a blank. Embedded blanks are considered part of a categories' name.
AI	Analyst's name	This card specifies the analyst's name for report identification and is an optional input.

KEYWORD	ACCEPTABLE INPUTS	DESCRIPTION
DATE	Free form	This card gives the date for report identification and is an optional input.
SEGMENT	Segment number	This card specifies the segment number for report identification and is an optional input.
*END	Ignored	This card identifies the end of the user defined cards.

An example of a control card file follows:

```
AI              MARY ANN TOMPKINS
SEGMENT          0882
DATE             APRIL 1, 1981
ALPHARNG         0.0   4000.0
BETARNG          0.0   3600.0
TORNG            1000.0  2000.0
CROP             CORN, SOYBEAN, OTHERS
```

3.5 SOFTWARE DESCRIPTION

3.5.1 PARPLT

Purpose

(1) To produce scatter plots of the greenness profile derived parameters α, β, t_o from the CLASFYG program. Alpha is the approximate greenness rise time, Beta is the approximate greenness decay time and t_o is the spectral crop emergence date.

(2) To produce statistical information concerning the α, β, t_o parameters computed by the CLASFYG program.

Linkages

PARPLT calls subroutine PPLTIN, SCAT.

Interface

Calling sequence:

N/A (PARPLT EXEC which loads and executes PARPLT is described in Section 4.0).

Calling sequence parameters:

None applicable.

Function value:

None applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

Label	Variable Name	Element Position	Input/ Output	Description
/PLOT/	RXMX(10)	1	O	X min of X,Y plot coordinate.
	RXMN(10)	2	O	X max of X,Y plot coordinate.

Label	Variable Name	Element Position	Input/ Output	Description
	RYMX(10)	3	O	Y min of X,Y plot coordinate.
	RYMN(10)	4	O	Y max of X,Y plot coordinate.
/NSBIXL/	NSUBPX(6)	1	O	6 subpixel values that map to a pixel.
/HEADER/	RHEAD(18,20)	1	O	Heading printed on individual map.
/DEFGT/	GTRSEG(3)	1	O	Ground truth segment number.
	TYR	2	O	Year ground truth file created.
	GTDATE(3)	3	O	Julian date ground truth file created.
/DEFKL/	KLASES	1	O	Segment used to generate class file.
	KYR	2	O	Year class file created.
	KLDATE(3)	3	O	Julian date class file created.

Blank COMMON parameters:

None.

Inputs

Unit	Type	Description
9	Sequential data.	Ground truth file (See Section 3.4.1).
10	Sequential data.	Classification file (See Section 3.4.1).
19	Sequential data.	Symbolic mapping elements, pixel purity range (See Section 3.4.2).

ORIGINAL PAGE IS
OF POOR QUALITY

:ts

Description

Run ti e diagno .

sk Plot report.

age fi amer

: applicable.

Description

Ground truth data is used in the process of determining if a specific pixel value is among the categories chosen to be plotted. The program processes each pixel in the following manner:

(1) Through the use of the ground truth transformation table a transformed numeric code is assigned to each of the six ground truth sub-pixels which correspond to the given pixel.

(2) The pixel is assigned to a majority category. This majority category is either the category of the first subpixel or is the category which occurs the most among the six sub-pixels categories.

(3) The number of occurance of the majority category is defined to be the purity of the given pixel. This purity is compaired to a user specified pixel purity range, and if the number of occurances fall outside of this range then there is no further processing of this pixel.

(4) If the majority category is greater than the number of categories specified by the user then there is no further processing of this pixel.

(5) If the pixel value of the classification paramenter α is zero then there is no further processing of this pixel.

(6) Each of the plots are updated according to the classification parameter to reflect the occurance of an additional pixel.

(7) Statistical variables are updated.

the program outputs statistical
plots and statistical information are

m listing.

3.5.2 SUB

Purpose

CROPP acc s a pixel value representative of the su.

Linkages

CROPP is .

Interface

Calling sec

CALL C

Calling sec

Argumen		1
KROP		rop code.
IXLCNT		mber subpixels that match the de.

Function valu:

Not appl cable.

Labeled COMMON parameters:

Full descript blocks are contained in Appendix A.

Label		Input/ Output	Description
/NSBIXL/		I	Subpixels that map to a pixel.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

The first sub-pixel value upon entry is considered to be the majority pixel (KROP) value. The NSUBPX array is traversed and the number of codes equal to KROP counted. This proceuure continues until a count of each unique code is made. If at anytime a count becomes greater than the count of KROP this code becomes KROP's value.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.3 FUNCTION ICE

Purpose

To established the integer character equivalence of a byte.

Linkages

ICE is called by PPLTIN.

Interface

Calling sequence:

K = ICE (INT).

Calling sequence parameters:

Argument	Input/ Output	Description
INT	I	One byte in character form.

Function value:

Name	Description
K	One byte in computation form.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

ICE sets the input parameter to the function and returns.

Flowchart

Not applicable.

Listing

See Appendix B for function listing.

3.5.4 FUNCTION IVALUE

Purpose

To allow the in line storage and testing of integer quoted literals.

Linkages

IVALUE is called by PPLTIN.

Interface

Calling sequence:

K = IVALUE (INT).

Calling sequence parameters:

Argument	Input/ Output	Description
INT	I	Quoted literal declared integer.

Function value:

Name	Description
K	Quoted literal.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

IVALUE sets quoted literals to integer function and returns.

Flowchart

Not applicable.

Listing

See Appendix B for function listing.

3.5.5 SUBROUTINE JULIAN

Purpose

To convert a Gregorian Calendar date to a Julian calendar date.

Linkages

JULIAN is called by PARPLT.

Interface

Calling sequence:

```
CALL JULIAN (JDATE, INERR, INDAY, INMNTH, INYEAR).
```

Calling sequence parameters:

Argument	Input/ Output	Description
JDATE(3)	O	Array to return Julian date.
INERR	O	Error flag indicating input value out of range (zero returned if no error occurred).
INDAY	I	Day of month.
INMNTH	I	Month of the year.
INYEAR	I	Last two digits of the year.

Function value:

Not applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

None.

Blank COMMON parameters:

None.

Inputs

None

Outputs

None.

Storage requirement

Not applicable.

Description

If the input day is less than 1 or greater than 31, and input month is less than 1 or greater than 12 set INERR = 1 for error flag and return. Compute Julian date from month and day. If the year is a Leap year and the month is greater than two add one to the computed Julian date.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.6 SUBROUTINE PACK

Purpose

Packs the number of occurances that parameters map to a computed location in a byte.

Linkages

PACK is called by SCAT.

Interface

Calling sequence:

CALL PACK (LOTBLE, IX, IY, IPLOT).

Calling sequence parameters:

Argument	Input/ Output	Description
LOTBLE	I	Plot labels to count occurances of pixels.
IX	I	Column of LOTBLE (word).
IY	I	Row of LOTBLE.
IPLOT	I	Number of plot to update.

Function value:

Not applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

Compute the byte location to be updated. Store word from LOTBLE containing byte in temporary location (IWORD). Store computed byte in a temporary word. Add one to byte word and if greater than 255 return; else, change byte within IWORD to new value and store word back into LOTBLE.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.7 SUBROUTINE PPLTIN

Purpose

Reads, classifies, and analyzes cards describing the following:

AI,SEGMENT,DATE,FILE	DOCUMENTATION - READ AND WRITTEN.
ALPHARNG	MIN MAX RANGE FOR ALPHA COEFS.
BETARNG	MIN MAX RANGE FOR BETA COEFS.
TORNG	MIN MAX RANGE FOR TO COEFS.
CROP	1-3 CROPS TO MAP. ENBEDDED BLANKS ARE INCLUDED IN NAME.
*END	SPECIFIES THE END OF USER DEFINED CARDS.

Linkages

PPLTIN is called by PARPLT.

Interface

Calling sequence:

CALL PPLTIN (NCROP, RCROP, RANGE, IERR).

Calling sequence parameters:

Argument	Input/ Output	Description
NCROP	O	Total number of user input crops.
RCROP	O	Crops to be evaluated.
RANGE	O	Minimum, maximum of Alpha, Beta, and t_o.
IERR	O	Error flag 0 - OK 1 - error

Function value:

Not applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

None.

Blank COMMON parameters:

None.

Inputs

Unit	Type	Description
21	Seq. data	User defined control card file (See Section 3.4.3)

Ouputs

Unit	Type	Description
20	Printer	Control card input for report.
5	Terminal	Run time errors.

Storage requirement

Not applicable.

Description

PPLTIN processes each input card. After the *END card is reached PPLTIN test for the following conditions:

1. MORE/LESS THAN TWO NUMBERS ON THE ALPHA, BETA, TO RANGE CARDS.
2. MIN>=MAX ON THE ALPHA, BETA, TO RANGE CARDS.
3. ALPHA, BETA, TO RANGE CARD MISSING.
4. MORE THAN THREE CROPS ON A CROP CARD.
5. CROP CARD MISSING.

If any of the conditions exist a diagnostic message is issued and IERR is set equal to one. If a control card is not recognized a warning message is printed.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.8 SUBROUTINE SCAT

Purpose

SCAT updates the plots to reflect an occurances of a pixel and outputs the actual plots.

Linkages

SCAT is called by PARPLT.

Interface

Calling sequence:

```
CALL SCAT (IB, IPLOT, XV, YV).
```

Calling sequence parameters:

Argument	Input/ Output	Description
IB	I	Flag: 1 = Initilize plots to zero. 2 = Update plots. 3 = Output plots.
IPLOT	I	The number of the plot to be considered.
XV	I	Column of the plot.
YV	I	Row of the plot.

Function value:

Not applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

Label	Variable	Element Postion	Input/ Ouput	Description
/PLOT/	XMY(10)	1	I	X min of X,Y plot coordinate.
	XMN(10)		I	X max of X,Y plot coordinate.
	YMN(10)		I	Y max of X,Y plot coordinate.

Label	Variable	Element Position	Input/ Ouput	Description
/HEADER/	RHEAD(18,20)	1	I	Heading printed on individual map.
/DEFGT/	GTRSEG(3)	1	I	Ground truth segment number.
	TYR	2	I	Year ground truth file created.
	GTDATE(3)	3	I	Julian date ground truth file created.
/DEFKL/	KLASES	1	I	Segment used to generate class file.
	KYR	2	I	Year class file created.
	KLDATE(3)	3	I	Julian date class file created.

Blank COMMON parameters:

None.

Inputs

None.

Ouputs

Unit	Type	Description
6	Printer	Report file.

Storage requirement

Not applicable.

Description

SCAT initializes, updates, and outputs the scatter plots according to the value of IB. If IB=2 SCAT computes the address from the value of the parameters and then calls PACK to update the count of occurances to this address. If IB=3 SCAT calls UNPACK to retrieve the count of occurances and

uses the count to index into the IT array for a symbol to use in the actual plotting.

Flowchart

Not applicable.

Listing

See Appendix B for routine listing.

3.5.9 SUBROUTINE UNPACK

Purpose

Unpack a byte and store the contents into an integer word.

Linkages

UNPACK is called by SUBROUTINE SCAT.

Interface

Calling sequence:

CALL SUBROUTINE UNPACK (IRET, LOTBLE, IX, IY, IPLOT).

Calling sequence parameters:

Argument	Input/ Output	Description
IRET	O	Unpacked byte.
LOTBLE	I	Table containing the byte.
IX	I	Column of location of word containing byte.
IY	I	Row of location of word containing byte.
IPLOT	I	Plot within LOTBLE containing byte.

Function value:

Not applicable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

UNPACK computes the location within the word of the byte to UNPACK. The word containing the byte is stored in a temporary location and the byte to be UNPACKED is set equal to IRET.

Flowchart

Not applicable.

Listing

See Appendix B for routine listings.

3.5.10 FUNCTION RVALUE

Purpose

To allow inline testing and storage of quoted literals declared real.

Linkages

RVALUE is called by PARPLT.

Interface

Calling sequence:

R = RVALUE (REL).

Calling sequence parameters:

Argument	Input/ Output	Description
REL	I	Quoted literals declared real.

Function value:

Name	Description
R	Real variable.

Labeled COMMON parameters:

Full description of labeled COMMON blocks are contained in Appendix A.

None.

Blank COMMON parameters:

None.

Inputs

None.

Outputs

None.

Storage requirement

Not applicable.

Description

RVALUE sets quoted literals to a real function and returns.

Flowchart

Not applicable.

Listing

See Appendix B for function listing.

4.0 OPERATION

4.1 OPERATING DESCRIPTION

PARPLT is operational on the IBM 3031 computer at LARS, West Lafayette, Indiana.

The PARPLT program is one of the programs of the BADHWAR SYSTEM which includes the programs CLASFYT, CLASFYG, MISMAP, PARPLT, PARHIS, and PARCLA.

PARPLT requires the use of a D disk which is assigned as a temporary disk and an E disk which is used to load LARS routine onto. The user, therefore, must notassign a disk to his machine using either MODE E or MODE D. These disks, will be assigned as needed.

Prior to executing the PARPLT program the user must (1) establish on his A disk a SYMBOL file as described in Section 3.4.2 and (2) establish the CONTROL CARD file as described in Section 3.4.3, if he wishes to use this file.

4.2 COMMANDS DESCRIPTION

To execute PARPLT, the user enters a series of commands which invoke the JOB CONTROL SOFTWARE. These commands are divided into two classes namely (1) FUNCTION commands and (2) PROGRAM commands. The FUNCTION commands, which perform all the functions except executing the program are reusable; i.e., once they are invoked they remain in effect until reentered. The PROGRAM commands, which execute the program, must be reentered each time the program is to be executed.

The following list gives the commands required to execute the PARPLT program. They are all FUNCTION commands except the PROGRAM command PARPLT. These commands must be given in the listed order except that the order of the DEFGTRU command and the DEFCLAS command may be interchanged.

START

DEFGTRU.....

DEFCLAS.....

PARPLT.....

END

The following sections describe each of the commands in detail. Input fields are separated by blanks. If more than one word is required to describe an input field, the descriptive text is enclosed in pointed brackets <>. If an input is optional the field is enclosed in square brackets []. Do not include these explanatory characters <> [] when actually submitting input to the computer. To enter a command the user types one input per defined input field and separates each field with a blank.

4.2.1 START

The START command spools the user's console file. The use of this command, along with the END command, will provide a listing of all information appearing on the user's console file. (If running an interactive job, this is the terminal. If running a batch job this is a system defined device). The START command is invoked by the user typing the following:

START

4.2.2 DEFGTRU

The DEFGTRU command defines a Ground Truth file. The user can use this command to define Ground Truth files on tape, disk, or may request the use of a file from the LARS RT&E Data Base. If the data request is for the use of a file from the LARS Data Base a series of programs are invoked to provide interface with the data base. The following diagram illustrates his software flow.

```
                                  ..RTEERR (LARS ROUTINE)
DEFGTRU.............GTRUINF....: GTINFO (LARS ROUTINE)
```

For a detailed description of the above JOB CONTROL SOFTWARE See Appendix B.

The DEFGTRU command has the following forms and is invoked by typing one of the following, according to the user's requirement.

If the file is on tape -

DEFGTRU/TAPE# FILE# < TAPE DENSITY >

If the file is on disk -

DEFGTRU FILENAME FILETYPE FILEMODE

If the file is on the LARS Data Base -

DEFGTRU/SEGMENT# YEAR

(year-last two digits of data generation year)

This command remains in effect for the use of any of the BADHWAR SYSTEM PROGRAM commands and does not have to be reissued unless the user wishes to redefine the input Ground Truth File.

4.2.3 DEFCLAS

The DEFCLAS command defines the input Classification file. The user can use this command to define a Classification file on tape or disk. The DEFCLAS command has the following forms and is invoked by typing one of the following, according to the user's requirement.

If the file is on tape -

```
DEFCLAS TAPE# FILE# < TAPE DENSITY >
```

If the file is on disk -

```
DEFCLAS FILENAME FILETYPE FILEMODE
```

This command remains in effect for the use of any of the BADHWAR SYSTEM PROGRAM commands and does not have to be reissued unless the user wishes to redefine the input Classification File.

4.2.4 PARPLT

The PARPLT command is a PROGRAM command is used to invoke the execution of the PARPLT program. This command must not be used unless the DEFCLAS and the DEFGTRU FUNCTION commands have been previously issued. Also, as previously stated the user must have established a SYMBOL FILE and a CONTROL CARD FILE on his A disk before invoking this command. The PARPLT command is invoked by the user typing the following:

PARPLT < SYMBOL FILENAME > < SYMBOL FILETYPE > < CONTROL CARD FILENAME >

The output from the PARPLT program is spooled to the HOUSTON line printer. The output consists of a SCATTER PLOT and the USER INFORMATION file.

4.2.5 END

This command closes the user's console file and causes a spooled copy to be sent to the HOUSTON printer. This command has no effect if the START command was not previously issued. The END command is invoked by the user typing the following.

```
END
```

4.3 OPERATING EXAMPLE

For our example we will assume the following:

The symbol file is established on the user's A disk under the file description: PARPLT1 DATA A.

The control card file is established on the user's A disk under the file description: PARPLT1 CC A.

The user has two different Classification files which he wishes to input to the PARPLT PROGRAM. He will therefore, issue two PROGRAM commands in his command sequence.

Furthermore, the user has elected to use a ground truth file at LARS.

COMMAND	EXPLANATION OR ACTION TAKEN
START	Spools the console file.
DEFGTRU 882 79	Defines a ground truth file using data from the LARS Data Base.
DEFCLAS 088279 079 B	Defines a Class file on the users B disk. This data is on a disk which the user has previously attached to his disk using a B mode.
PARPLT PARPLT1 DATA PARPLT1	Executes the PARPLT PROGRAM.
DEFCLAS 2345 23 1600	Redefines the class file. This file is from a 1600 BPI tape.
PARPLT PARPLT1 DATA PARPLT1	Executes the PARPLT PROGRAM. The user has chosen to define his symbol file and CC file the same as in the previous execution of PARPLT.
END	Closes the user's console file and spools the files to the HOUSTON printer.

APPENDIX A
COMMON BLOCKS

COMMON BLOCKS

/NSBIXL/ NSUBPX(6)

NSUBPX Six subpixels that map to a pixel.

/DEFGT/ GTRSEG, TYR, GTDATE(3)

GTRSEG Ground truth segment number.
TYR Year of ground truth file creation.
GTDATE Julian date of ground truth file creation.

/DEFKL/ KLASEG, KYR, KLDATE

KLASEG Classification segment number.
KYR Year of classification file creation.
KLDATE Julian date of classification creation.

/PLOT/ RXMX(10), RXMN(10), RYMX(10), RYMN(10)

RXMX X maximum for X, Y plot coordinate.
RXMN X minimum for X, Y plot coordinate.
RYMX Y maximum for X, Y plot coordinate.
RYMN Y minimum for X, Y plot coordinate.

/HEADER/ RHEAD(18,20)

RHEAD Heading information for plots.

APPENDIX B
PROGRAM LISTINGS

REQUESTED OPTIONS: *OTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

```
      PROGRAM MAIN PARPLT FORTRAN                                       PAR00010
C     ---------------------------                                       PAR00020
C                                                                       PAR00030
C                                                                       PAR00040
C                                                                       PAR00050
C        PURPOSE                                                        PAR00060
C        -------                                                        PAR00070
C                                                                       PAR00080
C     THIS PROGRAM PRODUCES SCATTER PLOTS OF THE PARAMETERS             PAR00090
C     COMPUTED BY THE CLASYG PROGRAM                                    PAR00100
C                                                                       PAR00110
C                                                                       PAR00120
C        HISTORY                                                        PAR00130
C        -------                                                        PAR00140
C                                                                       PAR00150
C     THIS PROGRAM IS BASED ON THE PROGRAM AASGMAP. THEY ARE PART       PAR00160
C     OF THE ORIGINAL PROGRAM MISHAP                                    PAR00170
C                                                                       PAR00180
C        J.CARNES          LEMSCO      3/19/79      ORIGINAL CODE (PDP) PAR00190
C        MARY TOMPKINS     LEMSCO      2/4/81       TRANSFER DESIGN     PAR00200
C        D.CHENG           LEMSCO      2/6/81       MODIFIED FOR IBM    PAR00210
C                                                                       PAR00220
C                                                                       PAR00230
C         METHOD                                                        PAR00240
C         ------                                                        PAR00250
C                                                                       PAR00260
C                                                                       PAR00270
C     1. READ USER INFORMATION FILE FOR NO. OF CROPS, KIND              PAR00280
C        OF CROPS, SCALE FACTORS FOR ALPHA, BETA, TO AND                PAR00290
C        CHI-SQURE VALUE.                                               PAR00300
C                                                                       PAR00310
C     2. READ THE HEADER RECORDS OF BOTH GROUND TRUTH AND               PAR00320
C        CLASSIFICATION FILES AND PRINT OUT REPORT.                     PAR00330
C                                                                       PAR00340
C     3. READ GROUND TRUTH FILE AND ASSIGN A TRANSFORMED                PAR00350
C        CODE FOR EACH GROUND TRUTH SUBPIXEL. USE MAJORITY              PAR00360
C        RULE FOR THE PIXEL CODE ASSIGNMENT.                            PAR00370
C                                                                       PAR00380
C     4. READ CLASSIFICATION FILE FOR THE PARAMETERS                    PAR00390
C        (ALPHA, BETA, TO, CHI-SQURE ) AND UPDATE THE PARAMETERS        PAR00400
C        FOR PLOTTING TO REFLECT THE OCCURANCE OF PIXEL.                PAR00410
C                                                                       PAR00420
C                                                                       PAR00430
C        EXTERNAL REFERENCES                                            PAR00440
C        -------------------                                            PAR00450
C                                                                       PAR00460
C     CPTIME         CURRENT TIME AND DATE                              PAR00470
C     SCAT           SUBPROGRAM FOR SCATTER PLOTS                       PAR00480
C     JULIAN         COMPUTE JULIAN DATE                                PAR00490
C     CROPP          SUBPROGRAM FOR DETERMINATION OF MAJORITY RULE      PAR00500
C     PPLTIN         SUBPROGRAM FOR USER'S INPUT INFORMATION            PAR00510
C     RVALUE         FUNCTION FOR STORAGE OF QUOTED LITERALS            PAR00520
C                                                                       PAR00530
C                                                                       PAR00540
C        EXCEPTIONS                                                     PAR00550
C        ----------                                                     PAR00560
C                                                                       PAR00570
C     IF ANY OF THE FOLLOWING CONDITION EXISTS, SET IERR=1 .AND         PAR00580
C     SEND ERROR MESSAGE                                                PAR00590
C                                                                       PAR00600
C        1. PIXEL PRUITY INPUT ERRORS:                                  PAR00610
C           NPUPL>6 OR NPUPL<1< NPURU>6 OR NPURU<1                      PAR00620
C           NPURL >NPURU)                                               PAR00630
C        2. CLASSIFICATION HEADER ERROOR                                PAR00640
C                                                                       PAR00650
C        3. GROUND TRUTH HEADER ERROR                                   PAR00660
C                                                                       PAR00670
C        4.SEGMENT ERRORS                                               PAR00680
C           GTRSEG  .NE. KLASEG                                         PAR00690
C        5. GROUND TRUTH TRANSFORMATION ERRORS                          PAR00700
C              1>NB.NE.NO>255                                           PAR00710
C                                                                       PAR00720
C     IF IERROR=1 TERMINATE PROGRAM, BEFORE ENTERING DATA RECORDS       PAR00730
```

```
                C
                C
                C
                C     LOCAL DECLARATIONS
                C     ------------------
                C
ISN 0002        C     IMPLICIT INTEGER (A-Q,S-Z)                    SET R REAL. THE REST ARE INTEGER
ISN 0003        C     INTEGER GT(255)                               GROUND TRUTH TRANSFORMATION CODES (1-6)
ISN 0004        C     INTEGER NSUBPX(6)                             SUBPIXEL VALUES WHICH MAP TO A PIXEL
ISN 0005        C     INTEGER ABUF(196)                             ALPHA VALUES OF THE CLASSIFICATION FILE
ISN 0006        C     INTEGER BBUF(196)                             BETA VALUES OF THE CLASSIFICATION FILE
ISN 0007        C     INTEGER TBUF(196)                             TO VALUES OF THE CLASSIFICATION FILE
ISN 0008        C     INTEGER CBUF(196)                             CHI SQUARE VALUES OF THE CLASSIFICATION FILE
ISN 0009        C     INTEGER KLDATE(3)                             CLASSIFICATION JULIAN DATE
ISN 0010        C     INTEGER GTDATE(3)                             GROUND TRUTH JULIAN DATE
ISN 0011        C     INTEGER NB                                    BEGINNING VALUE OF TRANSFORMATION CODES
ISN 0012        C     INTEGER NE                                    ENDING VALUE OF TRANSFORMATION CODES
ISN 0013        C     INTEGER NO                                    TRANSFORMATION CODES(1 - 6)
ISN 0014        C     INTEGER NPURL                                 LOWER LIMIT ON PIXEL PURITY
ISN 0015        C     INTEGER NPURU                                 UPPER LIMIT ON PIXEL PURITY
ISN 0016        C     INTEGER NPLOTS                                NUMBER OF CROPS TO BE PROCESSED
ISN 0017        C     INTEGER KLASEG                                CLASSIFICATION SEGMENT NUMBER
ISN 0018        C     INTEGER KDAY                                  DATE OF THE MONTH IN THE CLASS HEADER RECORDS
ISN 0019        C     INTEGER KMON                                  MONTH OF THE YEAR IN THE CLASS HEADER RECORDS
ISN 0020        C     INTEGER KYR                                   YEAR IN THE CLASS HEADER RECORDS
ISN 0021        C     INTEGER GTRSEG                                GROUND TRUTH SEGMENT NUMBER
ISN 0022        C     INTEGER TDAY                                  DATE OF THE MONTH IN THE GROUND TRUTH HEADER
ISN 0023        C     INTEGER TMON                                  MONTH OF THE YEAR IN THE GROUND TRUTH HEADER
ISN 0024        C     INTEGER TYR                                   YEAR IN THE GROUND TRUTH HEADER
ISN 0025        C     INTEGER JERR                                  ERROR MSG FROM SUBROUTINE JULIAN
ISN 0026        C     INTEGER IERR                                  ERROR FLAG TO TERMINATE THE PROGRAM, IERR=1
ISN 0027        C     INTEGER IREC                                  GROUND TRUTH FILE DIMENSION COUNTER
ISN 0028        C     INTEGER LINE                                  117 ROWS OF PIXELS IN A FILE
ISN 0029        C     INTEGER SPIX                                  SUBPIXEL OF INTEREST
ISN 0030        C     INTEGER SAMP                                  196 PIXELS IN A LINE
ISN 0031        C     INTEGER GTCODE                                MAJORITY GROUND TRUTH CODE(TRANSFORMED)
ISN 0032        C     INTEGER NGROUP                                NUMBER OF PLOT
ISN 0033        C     INTEGER DATE(2)                               CPTIME DAY/MONTH/YEAR
ISN 0034        C     INTEGER*2 HMS(3)                              CPTIME FOR HOUR,MINUTE AND SECOND
ISN 0035        C     REAL RALPHA                                   ALPHA VALUE OF ABUF TO ENTER FOR PLOT
ISN 0036        C     REAL RTOT(6)                                  SUM OF PIXELS FOR ALL CATEGORIES
ISN 0037        C     REAL RBETA                                    BETA VALUE OF BBUF TO ENTER FOR PLOT
```

PAR00740 PAR00750 PAR00760 PAR00770 PAR00780 PAR00790 PAR00800 PAR00810 PAR00820 PAR00830 PAR00840 PAR00850 PAR00860 PAR00870 PAR00880 PAR00890 PAR00900 PAR00910 PAR00920 PAR00930 PAR00940 PAR00950 PAR00960 PAR00970 PAR00980 PAR00990 PAR01000 PAR01010 PAR01020 PAR01030 PAR01040 PAR01050 PAR01060 PAR01070 PAR01080 PAR01090 PAR01100 PAR01110 PAR01120 PAR01130 PAR01140 PAR01150 PAR01160 PAR01170 PAR01180 PAR01190 PAR01200 PAR01210 PAR01220 PAR01230 PAR01240 PAR01250 PAR01260 PAR01270 PAR01280 PAR01290 PAR01300 PAR01310 PAR01320 PAR01330 PAR01340 PAR01350 PAR01360 PAR01370 PAR01380 PAR01390 PAR01400 PAR01410 PAR01420 PAR01430 PAR01440 PAR01450 PAR01460 PAR01470 PAR01480 PAR01490 PAR01500 PAR01510

```
ISN 0038          REAL RXMX(10)
          C                            X MAXIMUM FOR X,Y COORDINATES
ISN 0039          REAL RXMN(10)
          C                            X MINIMUM FOR X,Y COORDINATES
ISN 0040          REAL RYMX(10)
          C                            Y MAXIMUM FOR X,Y COORDINATES
ISN 0041          REAL RYMN(10)
          C                            Y MINIMUM FOR X,Y COORDINATES
ISN 0042          REAL RHEAD(18,20)
          C                            CHARACTERS STORAGE FOR PLOTS
ISN 0043          REAL RTO
          C                            TO VALUE OF TBUF TO ENTER FOR PLOT
ISN 0044          REAL RX
          C                            X AXIS FOR PLOT
ISN 0045          REAL RY
          C                            Y AXIS FOR PLOT
ISN 0046          REAL RCROP(4,3)
          C                            ARRAY FOR INPUT CROP CHARACTERS
ISN 0047          REAL RARRY(6)
          C                            MINIMUMS, MAXIMUMS  FOR ALPHA,BETA,TO
ISN 0048          LOGICAL*1 GTBUF(3,540)
          C                            GROUND TRUTH PIXEL BUFFER (3X540)
ISN 0049          LOGICAL*1 GTBUF1(6,270)
          C                            GROUND TRUTH PIXEL BUFFER (6X270)
ISN 0050          LOGICAL*1 HDR(3060)
          C                            3060 BYTE HEADER RECORDS
          C
          C
ISN 0051          COMMON/PLOT/RXMX,RXMN,RYMX,RYMN
ISN 0052          COMMON/NSBIXL/NSUBPX
ISN 0053          COMMON/HEADER/RHEAD
ISN 0054          COMMON/DEFGT/GTRSEG,TYR,GTDATE
ISN 0055          COMMON/DEFCL/KLASEG,KYR,KLDATE
ISN 0056          EQUIVALENCE (GTBUF(1),GTBUF1(1,1))
          C
          C
          C PROCEDURE
          C ---------
          C
          C  CLEAR THE RHEAD ARRAY
          C
ISN 0057          DO 5 I=1,18
ISN 0058          DO 5 J=1,20
ISN 0059            RHEAD(I,J)=PVALUE('    ')
ISN 0060        5 CONTINUE
          C
          C  PRINT JOB DESCRIPTION, DATE AND TIME
          C
ISN 0061          CALL CPTIME(DATE,HMS,TT,VT)
          C
ISN 0062          WRITE(6,1000)(DATE(I),I=1,2),(HMS(J),J=1,3)
          C
          C  INITIALIZE THE ERROR FLAG
          C
ISN 0063          IERR=0
          C
          C  DETERMINE THE NUMBER OF CROPS, KINDS OF CROPS AND SCALE FACTORS
          C  FOR ALPHA, BETA AND TO FROM ARRAY
          C
ISN 0064          CALL PPLTIN(NPLOTS,RCROP, RARRY, IERR)
          C
          C
          C  READ IN RANGE OF PIXEL PURITY
          C  RANGE OF PIXEL PURITY - LOWER AND UPPER LIMITS ON PIXEL
          C  PURITIES WHICH WILL BE CONSIDERED IN AN (I1,1X,I1) FORMAT.
          C
ISN 0065          READ(19,1005) NPURL,NPURU
ISN 0066          IF((NPURL.GE.1.AND.NPURL.LE.6).AND.
               &     (NPURU.GE.1.AND.NPURU.LE.6).AND.
               &     (NPURL.LE.NPURU)) GO TO 10
          C
          C  WRITE ERROR MESSAGE, PIXEL PURITY ERROR
          C
ISN 0068          IERR=1
ISN 0069          WRITE(5,1008)
          C
          C  ENTER IN THE MINIMUMS, MAXIMUMS FOR ALPHA, BETA, AND TO.
          C  ENTER ALPHA AND PUT INTO X FOR PLOTS 1,2,4,5,7 & 8.
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
                C                                                                       PAR02300
ISN 0070          10 DO 20 J=1,7,3                                                      PAR02310
ISN 0071              RXMN(J)=  RARRY(1)                                                PAR02320
ISN 0072              RXMN(J+1)=  RARRY(1)                                              PAR02330
ISN 0073              RXMX(J)=  RARRY(2)                                                PAR02340
ISN 0074          20 RXMX(J+1)=  RARRY(2)                                               PAR02350
                C                                                                       PAR02360
                C  ENTER BETA AND PUT INTO Y FOR 1,4 & 7 AND X FOR 3, 6 & 9.            PAR02370
                C                                                                       PAR02380
ISN 0075             DO 30 J=1,7,3                                                      PAR02390
ISN 0076              RYMN(J)=  RARRY(3)                                                PAR02400
ISN 0077              RYMX(J)=  RARRY(4)                                                PAR02410
ISN 0078              RXMN(J+2)=  RARRY(3)                                              PAR02420
ISN 0079          30  RXMX(J+2)=  RARRY(4)                                              PAR02430
                C                                                                       PAR02440
                C  ENTER TO AND PUT INTO 2,3,5,6,8 & 9.                                 PAR02450
                C                                                                       PAR02460
ISN 0080             DO 40 J=2,8,3                                                      PAR02470
ISN 0081              RYMN(J)=  RARRY(5)                                                PAR02480
ISN 0082              RYMX(J)=  RARRY(6)                                                PAR02490
ISN 0083              RYMN(J+1)=  RARRY(5)                                              PAR02500
ISN 0084          40 RYMX(J+1)=  RARRY(6)                                               PAR02510
                C                                                                       PAR02520
                C  ENTER IN TITLES UP TO 16 CHARACTERS.                                 PAR02530
                C                                                                       PAR02540
ISN 0085             DO 50 M=1,NPLOTS                                                   PAR02550
ISN 0086              DO 60 I=1,3                                                       PAR02560
ISN 0087              J=((M-1)*3+I)*2                                                   PAR02570
ISN 0088              DO 60 K=1,4                                                       PAR02580
ISN 0089              L=K+10                                                            PAR02590
ISN 0090              RHEAD(L,J)=RCROP(K,M)                                             PAR02600
ISN 0091          60  CONTINUE                                                          PAR02610
ISN 0092          50  CONTINUE                                                          PAR02620
                C                                                                       PAR02630
                C  LOAD THE CONSTANT DATA INTO THE HEADING ARRAY.                       PAR02640
                C                                                                       PAR02650
ISN 0093             DO 70 I=1,NPLOTS                                                   PAR02660
ISN 0094             DO 70 J=1,3                                                        PAR02670
ISN 0095              K=((I-1)*3+J)*2                                                   PAR02680
ISN 0096          70 RHEAD(9,K)=RVALUE('CROP')                                          PAR02690
ISN 0097             DO 80 I=1,NPLOTS                                                   PAR02700
ISN 0098              J=((I-1)*3+1)*2                                                   PAR02710
ISN 0099              RHEAD(15,J)=RVALUE('X=AL')                                        PAR02720
ISN 0100              RHEAD(16,J)=RVALUE('PHA,')                                        PAR02730
ISN 0101              RHEAD(17,J)=RVALUE('Y=BE')                                        PAR02740
ISN 0102              RHEAD(18,J)=RVALUE('TA  ')                                        PAR02750
ISN 0103              RHEAD(15,J+2)=RVALUE('X=AL')                                      PAR02760
ISN 0104              RHEAD(16,J+2)=RVALUE('PHA,')                                      PAR02770
ISN 0105              RHEAD(17,J+2)=RVALUE('Y=TO')                                      PAR02780
ISN 0106              RHEAD(18,J+2)=RVALUE('    ')                                      PAR02790
ISN 0107              RHEAD(15,J+4)=RVALUE('X=BE')                                      PAR02800
ISN 0108              RHEAD(16,J+4)=RVALUE('TA,Y')                                      PAR02810
ISN 0109              RHEAD(17,J+4)=RVALUE('=TO ')                                      PAR02820
ISN 0110          80 RHEAD(18,J+4)=RVALUE('    ')                                       PAR02830
                C                                                                       PAR02840
                C  READ THE HEADER RECORD OF BOTH FILE,                                 PAR02850
                C  AND CALL  SUBPROGRAM FOR JULIAN DATE                                 PAR02860
                C                                                                       PAR02870
ISN 0111             READ(10,1120)(HDR(I),I=1,3060)                                     PAR02880
ISN 0112             KLASEG = HDR(67)*2**8 + HDR(68)                                    PAR02890
ISN 0113             KDAY = HDR(61)                                                     PAR02900
ISN 0114             KMON = HDR(62)                                                     PAR02910
ISN 0115             KYR  = HDR(63)                                                     PAR02920
ISN 0116             CALL JULIAN(KLDATE,JERR,   KDAY,KMON,KYR)                          PAR02930
                C                                                                       PAR02940
ISN 0117             IF(JERR.NE.0)IERR=1                                                PAR02950
                C                                                                       PAR02960
                C  ERROR MSG FROM CLASSIFICATION HEADER. IF ERROR OCCURS, OUTPUT MSG    PAR02970
                C                                                                       PAR02980
ISN 0119             IF(JERR.NE.0)WRITE(5,1010)                                         PAR02990
                C                                                                       PAR03000
ISN 0121             REWIND 10                                                          PAR03010
                C                                                                       PAR03020
ISN 0122             READ(9,1120) (HDR(I),I=1,3060)                                     PAR03030
ISN 0123             GTRSEG = HDR(67)*2**8 + HDR(68)                                    PAR03040
ISN 0124             TDAY = HDR(61)                                                     PAR03050
ISN 0125             TMON = HDR(62)                                                     PAR03060
ISN 0126             TYR  = HDR(63)                                                     PAR03070
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0127          CALL JULIAN(GTDATE,JERR,   TDAY,TMON,TYR)                    PAR03080
                C                                                              PAR03090
ISN 0128          IF(JERR.NE.0)IERR = 1                                        PAR03100
                C                                                              PAR03110
                CC  ERROR MSG FROM GROUND TRUTH HEADER, IF ERROR OCCURS, OUTPUT MSG  PAR03120
                CC                                                             PAR03130
ISN 0130          IF(JERR.NE.0)WRITE(5,1020)                                   PAR03140
                C                                                              PAR03150
ISN 0132          IF(GTRSEG.NE.KLASEG) IERR=1                                  PAR03160
                C                                                              PAR03170
                CC  ERROR MSG FROM BOTH SEGMENT NO.. IF ERROR OCCURS, OUTPUT MSG  PAR03180
                CC                                                             PAR03190
ISN 0134          IF(GTRSEG.NE.KLASEG) WRITE(5,1025)KLASEG,GTRSEG              PAR03200
                C                                                              PAR03210
                CC  PRINT OUT HEADER FOR SEGMENT NO. , JULIAN DATE AND PIXEL PURITY  PAR03220
                CC                                                             PAR03230
ISN 0136          WRITE(6,1043)GTRSEG,TYR ,(GTDATE(I),I=1,3)                   PAR03240
ISN 0137          WRITE(6,1047)KLASEG,KYR,(KLDATE(I),I=1,3)                    PAR03250
ISN 0138          WRITE(6,1050)NPURL,NPURU                                     PAR03260
                C                                                              PAR03270
                CC  SET GROUND TRUTH TRANSFORMATION ARRAY TO ALL 6'S.          PAR03280
                CC                                                             PAR03290
ISN 0139            DO 90 I=1,255                                              PAR03300
ISN 0140          90  GT(I)=6                                                  PAR03310
                C                                                              PAR03320
                CC  WRITE GROUND TRUTH TRANSFORMATION HEADER                   PAR03330
                CC                                                             PAR03340
ISN 0141          WRITE(6,1060)                                                PAR03350
                C                                                              PAR03360
                CC     LOAD GROUND TRUTH TRANSFORMATION DATA FROM "SYMBOL" FILE.  PAR03370
                CC     GROUND TRUTH TRANSFORMATIONS - 1 LINE FOR EACH          PAR03380
                CC     TRANSFORMATION IN THE FORM STARTING CODE, ENDING CODE,  PAR03390
                CC     AND NEW CODE IN A (3I5) FORMAT.                         PAR03400
                CC                                                             PAR03410
ISN 0142        100 READ(19,1030) NB,NE,NO                                     PAR03420
ISN 0143          IF(NB.EQ.0.AND.NE.EQ.0.AND.NO.EQ.0) GO TO 130                PAR03430
                C                                                              PAR03440
                CC  TEST FOR GT TRANSFORMATION CODE ERRORS, IF ERRORS OCCOR, OUTPUT MSG  PAR03450
                CC                                                             PAR03460
ISN 0145          IF(NB.GE.1.AND.NB.LE.255.AND.NE.GE.1.AND.NE.LE.255) GO TO 110  PAR03470
ISN 0147          IERR=1                                                       PAR03480
                C                                                              PAR03490
ISN 0148          WRITE(5,1033)NB,NE,NO                                        PAR03500
ISN 0149          GO TO 130                                                    PAR03510
ISN 0150        110 IF(NO.GT.6.OR.NO.LT.1) NO=6                                PAR03520
                C                                                              PAR03530
                CC  PRINT OUT GROUND TRUTH TRANSFORMATION                      PAR03540
                CC                                                             PAR03550
ISN 0152          WRITE(6,1070) NB,NE,NO                                       PAR03560
                C                                                              PAR03570
ISN 0153            DO 120 I=NB,NE                                             PAR03580
ISN 0154        120   GT(I)=NO                                                 PAR03590
ISN 0155            GO TO 100                                                  PAR03600
                C                                                              PAR03610
                CC                                                             PAR03620
                CC  PRINT OUT RANGE OF PIXEL PURITY.                           PAR03630
                CC                                                             PAR03640
ISN 0156        130 WRITE(6,1050) NPURL,NPURU                                  PAR03650
                C                                                              PAR03660
                CC  PRINT OUT RANGE FOR ALPHA, BETA, AND TO                    PAR03670
                CC                                                             PAR03680
ISN 0157          WRITE(6,1080) RXMN(1),RXMX(1),RYMN(1),RYMX(1),RYMN(2),RYMX(2)  PAR03690
                C                                                              PAR03700
                CC  PRINT OUT CROPS TO BE EVALUATED.                           PAR03710
                CC                                                             PAR03720
ISN 0158          WRITE(6,1090) NPLOTS,(I,(RCROP(J,I),J=1,4),I=1,NPLOTS)       PAR03730
                C                                                              PAR03740
                CC  INITIALIZE THE PLOTTING ROUTINES.                          PAR03750
                CC                                                             PAR03760
ISN 0159          N=3*NPLOTS                                                   PAR03770
ISN 0160          DO 140 I=1,N                                                 PAR03780
ISN 0161          CALL SCAT(1.,1,RX,RY)                                        PAR03790
ISN 0162        140 CONTINUE                                                   PAR03800
                C                                                              PAR03810
                CC                                                             PAR03820
                CC  PRINT OUT CONTENTS OF HEADER RECORD FOR GROUND TRUTH FILE  PAR03830
                CC                                                             PAR03840
ISN 0163          WRITE(6,1122)GTRSEG,TYR,(GTDATE(I),I=1,3)                    PAR03850
```

```
ISN 0164            WRITE(6,1130) (HDR(I),I=1,60)                                  PAR03860
ISN 0165            WRITE(6,1140) (HDR(I),I=2249,2285)                             PAR03870
ISN 0166            WRITE(6,1150) (HDR(I),I=2760,2789)                             PAR03880
                C                                                                  PAR03890
                C  READ THE CLASSIFICATION HEADER RECORD AGAIN                     PAR03900
                C  IN ORDER TO WRITE THE HEADER INFORMATION                        PAR03910
                C                                                                  PAR03920
ISN 0167            READ(10,1120)(HDR(I),I=1,3060)                                 PAR03930
                C                                                                  PAR03940
ISN 0168            WRITE(6,1125)KLASEG,KYR,(KLDATE(I),I=1,3)                      PAR03950
ISN 0169            WRITE(6,1130) (HDR(I),I=1,60)                                  PAR03960
ISN 0170            WRITE(6,1140) (HDR(I),I=2249,2285)                             PAR03970
ISN 0171            WRITE(6,1150) (HDR(I),I=2760,2789)                             PAR03980
ISN 0172            WRITE(6,1155)                                                  PAR03990
ISN 0173            WRITE(6,1160) (HDR(I),I=2941,3000)                             PAR04000
                C                                                                  PAR04010
                C  CLEAR SUMMING ARRAY.                                            PAR04020
                C                                                                  PAR04030
ISN 0174            DO 160 J=1,6                                                   PAR04040
ISN 0175          160 RTOT(J)=0.                                                   PAR04050
                C                                                                  PAR04060
                C  WRITE ERROR MSG AND TERMINATE THE PROGRAM                       PAR04070
                C  BECAUSE OF ERRORS IN INPUT DATA, IERR IS NONZERO                PAR04080
                C                                                                  PAR04090
ISN 0176            IF(IERR.EQ.1)WRITE(5,1065)                                     PAR04100
ISN 0178            IF(IERR.EQ.1)GO TO 900                                         PAR04110
                C                                                                  PAR04120
                C                                                                  PAR04130
                C  READ THROUGH IMAGES ONE LINE AT A TIME FOR CLASSIFICATION FILE  PAR04140
                C  THE VALUES OF ALPHA, BETA, TO AND THEDA                         PAR04150
                C     AND THREE LINES AT A TIME FOR THE GROUND TRUTH FILE.         PAR04160
                C                                                                  PAR04170
ISN 0180            DO 400 LINE=1,117                                              PAR04180
                C                                                                  PAR04190
                C                                                                  PAR04200
ISN 0181              READ(10,1170,END=500)(ABUF(I),BBUF(I),TBUF(I),CBUF(I),I=1,196) PAR04210
                C                                                                  PAR04220
                C  READ GROUND TRUTH FILE INTO A 3X540 ARRAY.                      PAR04230
                C  FOR I = 1,3  J = 1,72, GTBUF(I,J) CONTAINS ANCILLARY DATA.      PAR04240
                C  GROUND TRUTH FILE CONTAINS 6 TIMES AS MUCH DATA AS THE          PAR04250
                C  CLASSIFICATION FILE.  SO, CORRESPONDING TO EACH VIDEO BLOCK (196 PAR04260
                C  BYTES) OF THE CLASSIFICATION FILE, THE GROUND TRUTH FILE IS TWICE PAR04270
                C  AS LONG AND THREE ROWS DEEP.                                    PAR04280
                C  SUBPIXEL DATA REQUIRE 2 COLUMNS AND 3 ROWS.                     PAR04290
                C  READ 72 CONTROL CHARACTERS + 2*196 ADJUSTED TO A MULTIPLE       PAR04300
                C  OF 180.                                                         PAR04310
                C                                                                  PAR04320
ISN 0182              DO 170 IREC=1,3                                              PAR04330
ISN 0183          170 READ(9,1200) (GTBUF(IREC,I),I=1,540)                         PAR04340
                C                                                                  PAR04360
                C                                                                  PAR04370
                C  MAKE COMPARISON ON PIXEL BY PIXEL BASIS.                        PAR04380
                C                                                                  PAR04390
ISN 0184            DO 300 SAMP=1,196                                              PAR04400
                C                                                                  PAR04410
                C                                                                  PAR04420
                C  DETERMINE GROUND TRUTH CODE.                                    PAR04430
                C  GTBUF1 IS A 6X270 ARRAY EQUIVALENCED TO GTBUF.  THE CONTROL      PAR04440
                C  CHARACTERS ARE NOW STORED IN GTBUF1(I,J) FOR I = 1,6  J = 1,36  PAR04450
                C  AND SUBPIXEL DATA REQUIRE 1 COLUMN AND 6 ROWS.                  PAR04460
                C                                                                  PAR04470
ISN 0185            DO 180 SPIX=1,6                                                PAR04480
ISN 0186              M=GTBUF1(SPIX,SAMP+36)                                       PAR04490
                C                                                                  PAR04500
                C  ADD 128 TO COMPENSATE FOR ADJUSTMENT MADE IN CARTO LAB          PAR04510
                C                                                                  PAR04520
ISN 0187              M=M+128                                                      PAR04530
ISN 0188              IF(M.GT.255) M=M-256                                         PAR04540
ISN 0190          180 NSUBPX(SPIX)=GT(M)                                           PAR04550
                C                                                                  PAR04560
                C  DETERMINE MAJORITY RULE CODE FOR PIXEL.                         PAR04570
                C                                                                  PAR04580
ISN 0191            CALL CROPP(GTCODE,NC)                                          PAR04590
                C                                                                  PAR04600
ISN 0192            IF(NC.LT.NPURL.OR.NC.GT.NPURU)GO TO 300                        PAR04610
                C                                                                  PAR04620
ISN 0194            IF(GTCODE.GT.NPLOTS)GO TO 300                                  PAR04630
                C                                                                  PAR04640
```

```
                C  IF THE VALUE IS ZERO, THIS IS BAD DATA                                PAR04650
   ISN 0196           IF(ABUF(SAMP).EQ.0) GO TO 300                                      PAR04660
                C                                                                        PAR04670
                C  COUNT NUMBER OF PIXELS                                                PAR04680
                C                                                                        PAR04690
   ISN 0198           RTOT(GTCODE)=RTOT(GTCODE)+1.0                                      PAR04700
                C                                                                        PAR04710
                C  SEND DATA TO THE PROPER ROUTINE                                       PAR04720
                C  AND CALL FOR PLOT                                                     PAR04730
                C                                                                        PAR04740
   ISN 0199           RALPHA=ABUF(SAMP)                                                  PAR04750
   ISN 0200           RBETA=RBUF(SAMP)                                                   PAR04760
   ISN 0201           RTO=TBUF(SAMP)                                                     PAR04770
   ISN 0202           NGROUP=(GTCODE-1)*3                                                PAR04780
   ISN 0203           CALL SCAT(2,NGROUP+1,RALPHA,RBETA)                                 PAR04790
   ISN 0204           CALL SCAT(2,NGROUP+2,RALPHA,RTO)                                   PAR04800
   ISN 0205         CALL SCAT(2,NGROUP+3,RBETA,RTO)                                      PAR04810
   ISN 0206       300 CONTINUE                                                          PAR04820
   ISN 0207       400 CONTINUE                                                          PAR04830
                C                                                                        PAR04840
                C  PRINT OUT SUMMARY AND MAKE PLOTS.                                    PAR04850
                C                                                                        PAR04860
   ISN 0208       500 WRITE(6,1210) ((RCROP(J,I),J=1,4),RTOT(I),I=1,NPLOTS)             PAR04870
   ISN 0209           NPLOTS=NPLOTS*3                                                    PAR04880
   ISN 0210           DO 190 I=1,NPLOTS                                                  PAR04890
   ISN 0211       190 CALL SCAT(3,I,RX,RY)                                              PAR04900
                C                                                                        PAR04910
   ISN 0212       900 STOP                                                              PAR04920
                C                                                                        PAR04930
                C   THE FORMAT STATEMENTS START :                                       PAR04940
                C                                                                        PAR04950
   ISN 0213       1000 FORMAT(1H1,' JOB INITIATED ON ',2A4,' AT ',I2,':'                PAR04960
                     &  ,I2,':',I2,', PROGRAM PARPLT ')                                 PAR04970
   ISN 0214       1005 FORMAT(I1,1X,I1)                                                 PAR04980
   ISN 0215       1008 FORMAT(/1X,' PIXEL PURITY RANGE ',I3,I3,' - UNACCEPTABLE ')      PAR04990
   ISN 0216       1010 FORMAT(/1X,' ERRORS OCCUR IN CLASSIFICATION HEADER. ')           PAR05000
   ISN 0217       1020 FORMAT(/1X,' ERROPS OCCUR IN GROUND TRUTH HEADER. ')             PAR05010
   ISN 0218       1025 FORMAT(' CLASS FILE FOR SEG.',I5,' GROUND TRUTH FILE FOR SEG',I5) PAR05020
   ISN 0219       1030 FORMAT(3I5)                                                      PAR05030
   ISN 0220       1033 FORMAT(' GROUND TRUTH SYMBOL TABLE RANGE',I5,I5,1X,I1,           PAR05040
                     1     '  - UNACCEPTABLE')                                        PAR05050
   ISN 0221       1043 FORMAT(//5X,'GROUND TRUTH FILE -        ',11X,I5,I2,I1,I1,I1)  PAR05060
   ISN 0222       1047 FORMAT(/5X,'CLASSIFICATION FILE -       ',10X,I5,I2,I1,I1,I1)  PAR05070
   ISN 0223       1050 FORMAT(1H0,//,10X,'PIXEL PURITY RANGE - FROM ',I1,' TO ', I1,    PAR05080
                     *' SUBPIXELS'//)                                                   PAR05090
   ISN 0224       1060 FORMAT(/10X,'GROUND TRUTH TRANSFORMATIONS'/)                     PAR05100
   ISN 0225       1065 FORMAT(/1X,'THERE ARE ERRORS IN THE INPUT FILE, PROGRAM TERMINATESPAR05110
                     &.')                                                               PAR05120
   ISN 0226       1070 FORMAT(1H ,17X,I3,' TO ',I3,' = ',I1)                            PAR05130
   ISN 0227       1080 FORMAT(1H0,10X,'RANGES FOR ALPHA, BETA, AND TO '/3(11X,2F10.3/)) PAR05140
   ISN 0228       1090 FORMAT(1H0,10X,' NUMBER OF CROPS TO BE EVALUATED =',I3/          PAR05150
                     *3(11X,I2,4X,4A4/))                                                PAR05160
   ISN 0229       1120 FORMAT(17(180A1))                                                PAR05170
   ISN 0230       1122 FORMAT(//,11X,'HEADER INFORMATION FOR GROUND TRUTH FILE',11X     PAR05180
                     &      ,I4,I2,I1,I1,I1)                                           PAR05190
   ISN 0231       1125 FORMAT(//,11X,'HEADER INFORMATION FOR CLASS FILE',9X,I4,I2,I1,I1 PAR05200
                     &      ,I1)                                                       PAR05210
   ISN 0232       1130 FORMAT(1H0,10X,'COMPUTING SYSTEM ID = ',60A1)                    PAR05220
   ISN 0233       1140 FORMAT(1H0,10X,'ACQUISITIONS USED   = ',5(8A1,3X))               PAR05230
   ISN 0234       1150 FORMAT(1H0,10X,'PFC JOB IDENT.      = ',30A1)                    PAR05240
   ISN 0235       1155 FORMAT(33X,'(SAMPLE,LINE) ')                                     PAR05250
   ISN 0236       1160 FORMAT(11X,'USER SUPPLIED DATA  = ',60A1)                        PAR05260
   ISN 0237       1170 FORMAT(18(45A4))                                                 PAR05270
   ISN 0238       1200 FORMAT(3(180A1))                                                 PAR05280
   ISN 0239       1210 FORMAT(1H0,10X,'SUMMARY OF NUMBER OF PIXELS IN EACH CATEGORY'/   PAR05290
                     *3(11X,4A4,4X,F10.1/))                                             PAR05300
   ISN 0240           END                                                               PAR05310
```

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

```
SYMBOL  INTERNAL STATEMENT NUMBERS
I       0057 0059 0062 0062 0086 0087 0093 0095 0097 0098 0111 0111 0111 0122 0122 0122 0136 0136
        0136 0137 0137 0137 0139 0140 0153 0154 0158 0158 0158 0158 0160 0161 0163 0163 0163 0164 0164
        0164 0165 0165 0166 0166 0166 0167 0167 0167 0168 0168 0168 0169 0169 0169 0170 0170 0170
        0171 0171 0171 0173 0173 0173 0181 0181 0181 0181 0181 0181 0183 0183 0183 0208 0208 0208 0208
        0210 0211
J       0058 0059 0062 0062 0062 0070 0071 0072 0073 0074 0075 0076 0077 0078 0079 0080 0081 0082 0083
        0084 0087 0090 0094 0095 0098 0099 0100 0101 0102 0103 0104 0105 0106 0107 0108 0109 0110 0158
```

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL STATEMENT NUMBERS
	0158 0158 0174 0175 0208 0208 0208
K	0088 0089 0090 0095 0096
L	0089 0090
M	0085 0087 0090 0186 0187 0187 0188 0188 0189 0190
N	0159 0160
GT	0003 0140 0154 0190
NB	0011 0142 0143 0145 0145 0148 0152 0153
NC	0191 0192 0192
NE	0012 0142 0143 0145 0145 0148 0152 0153
NO	0013 0142 0143 0148 0150 0150 0150 0152 0154
RX	0044 0161 0211
RY	0045 0161 0211
TT	0061
VT	0061
HDR	0050 0111 0112 0112 0113 0114 0115 0122 0123 0123 0124 0125 0126 0164 0165 0166 0167 0169 0170 0171 0173
HMS	0034 0061 0062
KYR	0020 0055 0115 0116 0137 0168
RTO	0201 0204 0205
RTO	0043
TYR	0024 0054 0126 0127 0136 0163
ABUF	0005 0181 0196 0199
BBUF	0006 0181 0200
CBUF	0008 0181
DATE	0033 0061 0062
IERR	0026 0063 0064 0068 0117 0128 0132 0147 0176 0178
IREC	0027 0182 0183
JERR	0025 0116 0117 0119 0127 0128 0130
KDAY	0018 0113 0116
KMON	0019 0114 0116
LINE	0028 0180
RTOT	0036 0175 0198 0198 0208
RXMN	0039 0051 0071 0072 0078 0157
RXMX	0038 0051 0073 0074 0079 0157
RYMN	0041 0051 0076 0081 0083 0157 0157
RYMX	0040 0051 0077 0082 0084 0157 0157
SAMP	0030 0184 0186 0196 0199 0200 0201
SCAT	0161 0203 0204 0205 0211
SPIX	0029 0185 0186 0190
TBUF	0007 0181 0201
TDAY	0022 0124 0127
TMON	0023 0125 0127
CROPP	0191
GTBUF	0048 0056 0183
NPURL	0014 0065 0066 0066 0066 0138 0156 0192
NPURU	0015 0065 0066 0066 0066 0138 0156 0192
RARRY	0047 0064 0071 0072 0073 0074 0076 0077 0078 0079 0081 0082 0083 0084
RBETA	0037 0200 0203 0205
RCROP	0046 0064 0090 0158 0208
RHEAD	0042 0053 0059 0090 0096 0099 0100 0101 0102 0103 0104 0105 0106 0107 0108 0109 0110
CPTIME	0061
GTBUF1	0049 0056 0186
GTCODE	0031 0191 0194 0198 0198 0202
GTDATE	0010 0054 0127 0136 0163
GTRSEG	0021 0054 0123 0132 0134 0134 0136 0163
JULIAN	0116 0127
KLASEG	0017 0055 0112 0132 0134 0134 0137 0168
KLDATE	0009 0055 0116 0137 0168
NGROUP	0032 0202 0203 0204 0205
NPLOTS	0016 0064 0085 0093 0097 0158 0158 0159 0194 0208 0209 0209 0210
NSUBPX	0004 0052 0190
PPLTIN	0064
RALPHA	0035 0199 0203 0204
RVALUE	0059 0096 0099 0100 0101 0102 0103 0104 0105 0106 0107 0108 0109 0110

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

LABEL	DEFINED	REFERENCES
5	0060	0057 0058
10	0070	0066
20	0074	0070
30	0079	0075
40	0084	0080
50	0092	0085
60	0091	0086 0088

*****FORTRAN CROSS REFERENCE LISTING*****

LABEL	DEFINED	REFERENCES
70	0096	0093 0094
80	0110	0097
90	0140	0139
100	0142	0155
110	0150	0145
120	0154	0153
130	0156	0143 0149
140	0162	0160
160	0175	0174
170	0183	0182
180	0190	0185
190	0211	0210
300	0206	0184 0192 0194 0196
400	0207	0180
500	0208	0181
900	0212	0178
1000	0213	0062
1005	0214	0065
1008	0215	0069
1010	0216	0119
1020	0217	0130
1025	0218	0134
1030	0219	0142
1033	0220	0148
1043	0221	0136
1047	0222	0137
1050	0223	0138 0156
1060	0224	0141
1065	0225	0176
1070	0226	0152
1080	0227	0157
1090	0228	0158
1120	0229	0111 0122 0167
1122	0230	0163
1125	0231	0168
1130	0232	0164 0169
1140	0233	0165 0170
1150	0234	0166 0171
1155	0235	0172
1160	0236	0173
1170	0237	0181
1200	0238	0183
1210	0239	0208

/ MAIN / SIZE OF PROGRAM 0038B2 HEXADECIMAL BYTES

NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
I	SFA	I*4	0006C4	J	SF	I*4	0006C8	K	SF	I*4	0006CC	L	SF	I*4	0006D0
M	SF	I*4	0006D4	N	SF	I*4	0006D8	GT	SF	I*4	001340	NB	SF	I*4	0006DC
NC	SFA	I*4	0006E0	NE	SF	I*4	0006E4	NO	SF	I*4	0006E8	RX	SFA	R*4	0006EC
RY	SFA	R*4	0006F0	IT	SFA	I*4	0006F4	VT	SFA	I*4	0006F8	HDR	SF	L*1	000744
HMS	SFA	I*2	001338	KYR	SFA C	I*4	000004	RTO	SFA	R*4	0006FC	RTO		R*4	NR
TYR	SFA C	I*4	000004	ABUF	SF	I*4	00173C	BBUF	SF	I*4	001A4C	CBUF	S	I*4	001C5C
DATE	SFA	I*4	00206C	IERR	SFA	I*4	000700	IREC	SF	I*4	000704	JERR	SFA	I*4	000708
KDAY	SFA	I*4	00070C	KMON	SFA	I*4	000710	LINE	SF	I*4	000714	RTOT	SF	R*4	002074
RXMN	SF C	R*4	000028	RXMX	SF C	R*4	000000	RYMN	SF C	R*4	000078	RYMX	SF C	R*4	000050
SAMP	SF	I*4	000718	SCAT	SF XF		000000	SPIX	SF	I*4	00071C	TBUF	SF	I*4	00208C
TDAY	SFA	I*4	000720	TMON	SFA	I*4	000724	CROPP	SF XF		000000	GTBUF	S E	L*1	0023F0
NPURL	SF	I*4	000728	NPURU	SF	I*4	00072C	RARRY	SFA	R*4	00239C	RBETA	SFA	R*4	000730
RCROP	SFA	R*4	0023B4	RHEAD	S C	R*4	000000	CPTIME	SF XF		000000	GTBUF1	F E	L*1	0023F0
GTCODE	SFA	I*4	000734	GTDATE	SFA C	I*4	000008	GTRSEG	SF C	I*4	000000	IBCOM#	F XF	I*4	000000
JULIAN	SF XF		000000	KLASEG	SF C	I*4	000000	KLDATE	SFA C	I*4	000008	NGROUP	SFA	I*4	000738
NPLOTS	SFA	I*4	00073C	NSUBPX	S C	I*4	000000	PPLTIN	SF XF		000000	RALPHA	SFA	R*4	000740
RVALUE	F XF	R*4	000000												

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK * PLOT* SIZE OF BLOCK 0000A0 HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
RXMX	P*4	000000	RXMN	R*4	000028	RYMX	P*4	000050	RYMN	R*4	000078

NAME OF COMMON BLOCK *NSRIXL* SIZE OF BLOCK 000018 HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
NSUBPX	I*4	000000									

NAME OF COMMON BLOCK *HEADER* SIZE OF BLOCK 0005A0 HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
RHEAD	R*4	000000									

NAME OF COMMON BLOCK * DEFGT* SIZE OF BLOCK 000014 HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
GIRSFG	I*4	000000	GIYR	I*4	000004	GIDATE	I*4	000008			

NAME OF COMMON BLOCK * DEFCL* SIZE OF BLOCK 000014 HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
KLASFG	I*4	000000	KYR	I*4	000004	KLDATE	I*4	000008			

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
5	60	002AD2	10	70	002BF4	20	74	002C22	30	79	002C6A
40	84	002CB2	60	91	002D2F	50	92	002D56	70	96	002D8C
80	110	002ED4	90	140	003124	100	142	00315C	110	150	003210
120	154	003264	130	156	00328C	140	162	00336E	160	175	00354C
170	183	003618	180	190	003696	300	206	0037CC	400	207	0037E0
500	208	0037F4	190	211	003864	900	212	003882			

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100000	1	002A90	100001	58	002AB0	100002	59	002AB4	100003	61	002ADE
100004	61	002AE2	100009	68	002BD8	100010	71	002BF8	100011	75	002C3C
100012	76	002C40	100013	80	002C84	100014	81	002C88	100015	85	002CCC
100016	86	002CD4	100017	87	002CDC	100018	89	002CFC	100019	92	002D42
100021	93	002D6A	100022	94	002D72	100023	95	002D76	100024	97	002DAA
100025	97	002DBE	100026	98	002DC6	100027	111	002F00	100030	118	002F76
100031	119	002F7E	100032	120	002F88	100033	121	002F9C	100036	129	00301E
100037	130	003026	100038	131	003030	100039	132	003044	100040	133	003050
100041	134	003058	100042	135	003064	100043	136	003088	100048	141	003148
200001	144	003192	200002	144	00319C	100049	145	0031A6	200003	146	003182
200004	146	0031BE	200005	146	0031CA	100050	147	0031D6	200006	151	00321C
100051	151	003228	100052	152	003230	100053	155	003288	100054	158	003314
100057	158	00334C	100058	161	003364	100059	163	003382	200007	171	0034E4
100080	176	003566	100081	177	003572	100082	178	003588	100083	180	003594
100084	181	00359C	100085	181	0035B0	100086	181	0035B8	100087	181	00360C
100090	184	003658	100091	185	003660	100092	186	003668	100093	189	00368E
100094	191	0036C6	200008	193	0036DC	100095	194	0036E8	100096	196	0036F4
100097	198	003706	100100	208	00380C	100103	208	00384C			

FORMAT STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
1000	213	000028	1005	214	000069	1008	215	000071	1010	216	0000A2
1020	217	0[illegible]001	1025	218	0000FF	1030	219	000136	1033	220	00013C
1043	221	00017A	1047	222	0001A9	1050	223	0001D8	1060	224	000215
1065	225	000239	1070	226	000277	1080	227	0002AF	1090	228	0002C3
1120	229	0002FF	1122	230	000308	1125	231	000344	1130	232	000379
1140	233	00039C	1150	234	0003C4	1155	235	0003E7	1160	236	0003FA
1170	237	000418	1200	238	000424	1210	239	00042D			

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

STATISTICS SOURCE STATEMENTS = 239, PROGRAM SIZE = 14514, SUBPROGRAM NAME = MAIN

STATISTICS NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ****** 240K BYTES OF CORE NOT USED

```
*LEVEL 2.3.0 (JUNE 78)                  OS/360  FORTRAN H EXTENDED          DATE 81.139/13.20.39      PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

    ISN 0002            SUBROUTINE CROPP(KROP,IXLCNT)                                          CR000010
                C       ----------------------------                                           CR000020
                C                                                                              CR000030
                C                                                                              CR000040
                C       INTEGER KROP      CROP CODE - MAJORITY RULE OR FIRST IN                CR000050
                C       INTEGER IXLCNT    COUNT OF CODE USED AS CROP CODE                      CR000060
                C                                                                              CR000070
                C                                                                              CR000080
                C     HISTORY                                                                  CR000090
                C     -------                                                                  CR000100
                C                                                                              CR000110
                C       CARL AKLERS    LEC                        ORIGINAL CODE (PDP)      CR000120
                C       MARY TOMPKINS  LFMSCO     11/21/80        REDO (IBM)               CR000130
                C                                                                              CR000140
                C     PURPOSE                                                                  CR000150
                C     -------                                                                  CR000160
                C                                                                              CR000170
                C       THIS ROUTINE IS CALLED TO MAKE THE MAJORITY RULE DECISION             CR000180
                C       FOR THE 6 SUB-PIXELS .                                               CR000190
                C                                                                              CR000200
                C     EXTERNAL REFERENCES                                                      CR000210
                C     -------------------                                                      CR000220
                C                                                                              CR000230
                C       NONE.                                                                  CR000240
                C                                                                              CR000250
                C                                                                              CR000260
                C     EXCEPTIONS                                                               CR000270
                C     ----------                                                               CR000280
                C                                                                              CR000290
                C       NONE.                                                                  CR000300
                C                                                                              CR000310
                C                                                                              CR000320
                C     LOCAL DECLARATIONS                                                       CR000330
                C     ------------------                                                       CR000340
                C                                                                              CR000350
                C       INTEGER KOUNT       COUNT OF CURRENT TYPE CROP CODE                    CR000360
                C                                                                              CR000370
                C                                                                              CR000380
                C     PROCEDURE                                                                CR000390
                C     ---------                                                                CR000400
                C                                                                              CR000410
                C                                                                              CR000420
    ISN 0003            COMMON/NSBIXL/NSUBPX(6)                                                CR000430
                C                                                                              CR000440
    ISN 0004            IXLCNT = 0                                                             CR000450
                C                                                                              CR000460
    ISN 0005            DO 20 I = 1,6                                                          CR000470
    ISN 0006               KOUNT = 0                                                           CR000480
    ISN 0007               DO 10 J = 1,6                                                       CR000490
    ISN 0008                  IF(NSUBPX(I).EQ.NSUBPX(J))KOUNT = KOUNT + 1                      CR000500
    ISN 0010            10    CONTINUE                                                        CR000510
    ISN 0011               IF(KOUNT.LE.IXLCNT) GO TO 20                                        CR000520
    ISN 0013               IXLCNT = KOUNT                                                      CR000530
    ISN 0014               KROP = NSUBPX(I)                                                    CR000540
    ISN 0015               IF(IXLCNT.GE.3)RETURN                                               CR000550
    ISN 0017            20 CONTINUE                                                            CR000560
    ISN 0018               RETURN                                                              CR000570
    ISN 0019               END                                                                 CR000580

                        *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL   INTERNAL  STATEMENT NUMBERS
I        0005  0008  0014
J        0007  0008
KROP     0002  0014
CROPP    0002
KOUNT    0006  0008  0008  0011  0013
IXLCNT   0002  0004  0011  0013  0015
NSUBPX   0003  0005  0008  0014

                        *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
 LABEL   DEFINED   REFERENCES
```

```
                    *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
 LABEL   DEFINED   REFERENCES
    10      0010      0007
    20      0017      0005  0011

                                                    /   CROPP /     SIZE OF PROGRAM 000194 HEXADECIMAL BYTES

      NAME     TAG   TYPE  ADD.       NAME     TAG   TYPE  ADD.       NAME     TAG   TYPE  ADD.       NAME     TAG   TYPE  ADD.
         I  SF        I*4   000090          J  SF        I*4   000094    KROP  S        I*4   000098    CROPP            R*4   00009C
    KOUNT  SF        I*4   0000A0    IXICNT  S        I*4   0000A4    NSUBPX  F   C    I*4   000000

                                        *****  COMMON INFORMATION  *****

       NAME OF COMMON BLOCK  *NSBIXL*   SIZE OF BLOCK     000018 HEXADECIMAL BYTES

      VAR. NAME  TYPE  REL. ADDR.   VAR. NAME  TYPE  REL. ADDR.   VAR. NAME  TYPE  REL. ADDR.   VAR. NAME  TYPE  REL. ADDR.
        NSUBPX   I*4    000000

  SOURCE STATEMENT LABELS

     LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR
        10   10  0000FE            20   17  00012C

  COMPILER GENERATED LABE

     LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR
    100001    2  0000AC       100002    6  0000D0       100003    8  0000D4       100004    9  0000EC
    100005   11  0000F6       100006   13  0000FC       100007   16  000114       100008   18  000134

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*      SOURCE STATEMENTS =       18, PROGRAM SIZE =      404, SUBPROGRAM NAME = CROPP
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                  296K BYTES OF CORE NOT USED
```

```
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
  ISN 0002             INTEGER FUNCTION ICE(INT)
              C                                  INTEGER CHARACTER EQUIVALENCE
              C        -----------------------------------------------------
              C
              C
              C HISTORY
              C -------
              C
              C        M A TOMPKINS     LEMSCO      01/27/81     ORIGINAL CODE
              C
              C
              C METHOD
              C ------
              C
              C        SET INPUT TO FUNCTION. RETURN.
              C
              C
              C EXCEPTIONS
              C ----------
              C
              C
              C        IF THE FUNCTION IS APPLIED TO A WORD THE VALUE RETURNED IS OF
              C        THE LEFT MOST BYTE
              C LOCAL DECLARATION
              C -----------------
              C
              C
  ISN 0003             LOGICAL*1 INT
              C
              C
              C PROCEDURE
              C ---------
              C
              C
  ISN 0004             ICE = INT
  ISN 0005             RETURN
  ISN 0006             END

                    *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL   INTERNAL STATEMENT NUMBERS
ICE      0002  0004
INT      0002  0003  0004

                                               /     ICE /     SIZE OF PROGRAM 0000C2 HEXADECIMAL BYTES
     NAME   TAG   TYPE  ADD.      NAME   TAG   TYPE  ADD.      NAME   TAG   TYPE  ADD.      NAME   TAG   TYPE  ADD.
      ICE S       I*4   000084     INT F       L*1   000080

  COMPILER GENERATED LABELS

   LABEL  ISN   ADDR         LABEL  ISN   ADDR         LABEL  ISN   ADDR         LABEL  ISN   ADDR
  100001    2  000088
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*     SOURCE STATEMENTS =       5, PROGRAM SIZE =      194, SUBPROGRAM NAME =   ICE
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                      296K BYTES OF CORE NOT USED
```

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED           DATE 81.140/12.31.14        PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
  ISN 0002             INTEGER FUNCTION IVALUE(INT)                                              IVA00010
                 C                                 ALLOWS QUOTED LITERALS                        IVA00020
                 C     ----------------------------------------------------                      IVA00030
                 C                                                                               IVA00040
                 C                                                                               IVA00050
                 C HISTORY                                                                       IVA00060
                 C -------                                                                       IVA00070
                 C                                                                               IVA00080
                 C     M A TOMPKINS     LEMSCO      01/27/81     ORIGINAL CODE                   IVA00090
                 C                                                                               IVA00100
                 C                                                                               IVA00110
                 C METHOD                                                                        IVA00120
                 C ------                                                                        IVA00130
                 C                                                                               IVA00140
                 C     SET INPUT TO FUNCTION. RETURN.                                            IVA00150
                 C                                                                               IVA00160
                 C PROCEDURE                                                                     IVA00170
                 C ---------                                                                     IVA00180
                 C                                                                               IVA00190
                 C                                                                               IVA00200
  ISN 0003             IVALUE = INT                                                              IVA00210
  ISN 0004             RETURN                                                                    IVA00220
  ISN 0005             END                                                                       IVA00230

                     *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL  INTERNAL STATEMENT NUMBERS
INT     0002  0003
IVALUE  0002  0003

                                        /  IVALUE /    SIZE OF PROGRAM 0000C2 HEXADECIMAL BYTES
      NAME   TAG   TYPE  ADD.       NAME    TAG   TYPE  ADD.      NAME    TAG   TYPE  ADD.      NAME    TAG   TYPE  ADD.
       INT  F      I*4   000080    IVALUE S       I*4   000084

  COMPILER GENERATED LABELS
    LABEL  ISN   ADDR           LABEL  ISN   ADDR           LABEL  ISN   ADDR           LABEL  ISN   ADDR
   100001    2  000088
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFFCT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*    SOURCE STATEMENTS =       4, PROGRAM SIZE =      194, SUBPROGRAM NAME =IVALUE
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                  296K BYTES OF CORE NOT USED
```

```
*LEVEL 2.3.0 (JUNE 78)                  OS/360  FORTRAN H EXTENDED          DATE 81.139/13.25.27       PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

   ISN 0002            SUBROUTINE JULIAN (JDATE,INERR,INDAY,INMNTH,INYEAR)                    JUL00010
                     C                                                                        JUL00020
                     C                                                                        JUL00030
                     C   PURPOSE:  TO CONVERT STANDARD DAY AND MONTH TO JULIAN DATE           JUL00040
                     C                                                                        JUL00050
                     C                                                                        JUL00060
                     C   HISTORY:                                                            JUL00070
                     C     J C CRISP         EMSCO     02/09/81     ORIGINAL CODE           JUL00080
                     C                                                                        JUL00090
                     C                                                                        JUL00100
                     C   DESCRIPTION OF ARGUMENTS:                                          JUL00110
                     C     JDATE--ARRAY TO RETURN JULIAN DATE (1 DIGIT PER ARRAY ELEMENT)     JUL00120
                     C     INERR--ERROR FLAG INDICATING INPUT VALUE OUT OF RANGE (ZERO      JUL00130
                     C            RETURNED IF NO ERROR OCCURS)                              JUL00140
                     C     INDAY--DAY OF MONTH                                              JUL00150
                     C     INMNTH--MONTH OF YEAR AS INTEGER                                 JUL00160
                     C     INYEAR--LAST TWO DIGITS OF YEAR                                  JUL00170
                     C                                                                        JUL00180
                     C                                                                        JUL00190
                     C   DESCRIPTION OF VARIABLES:                                          JUL00200
                     C     JTEMP--TEMPORARY STORAGE FOR JULIAN DATE                         JUL00210
                     C                                                                        JUL00220
                     C                                                                        JUL00230
   ISN 0003            DIMENSION JDATE (3)                                                    JUL00240
                     C                                                                        JUL00250
                     C CHECK FOR INPUT VALUE OUT OF RANGE                                     JUL00260
                     C                                                                        JUL00270
   ISN 0004            IF(((INDAY.LT.1).OR.(INDAY.GT.31)) GO TO 800                           JUL00280
   ISN 0006            IF (((INMNTH.LT.1).OR.(INMNTH.GT.12)) GO TO 800                        JUL00290
                     C                                                                        JUL00300
                     C DETERMINE JULIAN DATE FOR INPUT MONTH                                  JUL00310
                     C                                                                        JUL00320
   ISN 0008            IF (INMNTH.EQ.1) JTEMP=0                                               JUL00330
   ISN 0010            IF (INMNTH.EQ.2) JTEMP=31                                              JUL00340
   ISN 0012            IF (INMNTH.EQ.3) JTEMP=59                                              JUL00350
   ISN 0014            IF (INMNTH.EQ.4) JTEMP=90                                              JUL00360
   ISN 0016            IF (INMNTH.EQ.5) JTEMP=120                                             JUL00370
   ISN 0018            IF (INMNTH.EQ.6) JTEMP=151                                             JUL00380
   ISN 0020            IF (INMNTH.EQ.7) JTEMP=181                                             JUL00390
   ISN 0022            IF (INMNTH.EQ.8) JTEMP=212                                             JUL00400
   ISN 0024            IF (INMNTH.EQ.9) JTEMP=243                                             JUL00410
   ISN 0026            IF (INMNTH.EQ.10) JTEMP=273                                            JUL00420
   ISN 0028            IF (INMNTH.EQ.11) JTEMP=304                                            JUL00430
   ISN 0030            IF (INMNTH.EQ.12) JTEMP=334                                            JUL00440
   ISN 0032            JTEMP=JTEMP+INDAY                                                      JUL00450
                     C                                                                        JUL00460
                     C CHECK FOR LEAP YEAR AND ADD ONE DAY IF MONTH IS 3 OR GREATER           JUL00470
                     C                                                                        JUL00480
   ISN 0033            IF ((INMNTH.GE.3).AND.(MOD(INYEAR,4).EQ.0)) JTEMP=JTEMP+1              JUL00490
                     C                                                                        JUL00500
                     C STORE DATE IN ARRAY, ONE DIGIT PER ARRAY ELEMENT                       JUL00510
                     C                                                                        JUL00520
   ISN 0035            JDATE(1)=JTEMP/100                                                     JUL00530
   ISN 0036            JDATE(2)=(JTEMP-((JTEMP/100)*100))/10                                  JUL00540
   ISN 0037            JDATE(3)=JTEMP-((JTEMP/10)*10)                                         JUL00550
   ISN 0038            INERR=0                                                                JUL00560
   ISN 0039            GO TO 900                                                              JUL00570
                     C                                                                        JUL00580
                     C INPUT ERROR                                                            JUL00590
                     C                                                                        JUL00600
   ISN 0040          800 INERR=1                                                              JUL00610
                     C                                                                        JUL00620
                     C                                                                        JUL00630
   ISN 0041          900 RETURN                                                               JUL00640
   ISN 0042            END                                                                    JUL00650
                                                                                              JUL00660
                                                                                              JUL00670
                                                                                              JUL00680
                                                                                              JUL00690
                                                                                              JUL00700
                                                                                              JUL00710
```

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

```
SYMBOL   INTERNAL STATEMENT NUMBERS
MOD      0033
INDAY    0002  0004  0004  0032
INERR    0002  0038  0040
JDATE    0002  0003  0035  0036  0037
JTEMP    0008  0010  0012  0014  0016  0018  0020  0022  0024  0026  0028  0030  0032  0032  0033  0033  0035  0036  0036
         0037  0037
INMNTH   0002  0006  0006  0008  0010  0012  0014  0016  0018  0020  0022  0024  0026  0028  0030  0033
INYEAR   0002  0033
JULIAN   0002
```

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

```
LABEL   DEFINED   REFERENCES
  800     0040    0004  0006
  900     0041    0039
```

/ JULIAN / SIZE OF PROGRAM 000300 HEXADECIMAL BYTES

```
   NAME     TAG   TYPE  ADD.      NAME     TAG   TYPE  ADD.      NAME     TAG   TYPE  ADD.      NAME     TAG   TYPE  ADD.
  INDAY     F     I*4   0000E4    INERR    S     I*4   0000E8    JDATE    S     XR    000000    JTEMP    SF    I*4   0000FC
  INMNTH          I*4   0000F0    INYEAR   A     I*4   0000F4    JULIAN          I*4   0000F8
```

SOURCE STATEMENT LABELS

```
LABEL  ISN  ADDR      LABEL  ISN  ADDR      LABEL  ISN  ADDR      LABEL  ISN  ADDR
  800   40  00027F      900   41  000282
```

COMPILER GENERATED LABELS

```
LABEL   ISN  ADDR      LABEL   ISN  ADDR      LABEL   ISN  ADDR      LABEL   ISN  ADDR
100001    2  000110    100002    6  000148    100003    8  00016A    100004    9  000170
100005   10  000174    100006   11  00017C    100007   12  000180    100008   13  000188
100009   14  00018C    100010   15  000194    100011   16  000198    100012   17  0001A0
100013   18  0001A4    100014   19  0001AC    100015   20  000180    100016   21  0001B8
100017   22  0001BC    100018   23  0001C4    100019   24  0001C8    100020   25  0001D0
100021   26  0001D4    100022   27  0001DC    100023   28  0001E0    100024   29  0001E8
100025   30  0001EC    100026   31  0001F4    100027   32  0001F8    100028   34  00022C
100029   35  00022F
```

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

STATISTICS SOURCE STATEMENTS = 41, PROGRAM SIZE = 768, SUBPROGRAM NAME =JULIAN

STATISTICS NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ****** 292K BYTES OF CORE NOT USED

```
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

ISN 0002        SUBROUTINE PACK(
           C                       UPDATE BYTE COUNTER
               I LOTBLE,
           C                      PLOT TABLES USED TO COUNT OCCURANCE OF LABLES
               ( IX,IY,
           C                      X,Y AXIS FOR PLOT.
               (IPLOT)
           C                      NUMBER OF PLOT WHICH IS TO BE UPDATE
           C    ------------------------------------------------------------
           C
           C
           C  HISTORY
           C  -------
           C
           C    MARY TOMPKINS    LEMSCO      12/23/80      REQUIREMENTS
           C    MARY TOMPKINS    LEMSCO      12/23/80      ALGORITHM DESIGN
           C    DONALD CHENG     LEMSCO      12/24/80      ALGORITHM CODING
           C
           C
           C  METHOD
           C  ------
           C
           C    PACK INTO THE APPROPRIATE BYTE OF LOTABLE WORD THE COUNT OF
           C    LABLES GRAPH TO A PARTICULAR POINT.
           C
           C  MACHINE-DEPENDENT CODE
           C  ----------------------
           C
           C    ASSUMES IBM 32 BIT, 4BYTE WORD SIZE
           C
           C
           C  EXTERAL REFERENCES
           C  ------------------
           C
           C    NONE.
           C
           C
           C  EXCEPTIONS
           C  ----------
           C
           C    THE MAXIMUN COUNT OF LABLES IS 255.
           C
           C
           C  GLOBAL DECLARATIONS
           C  -------------------
           C
           C    NONE.
           C
           C
           C  LOCAL DECLARATIONS
           C  ------------------
           C
ISN 0003        INTEGER LOTBLE(13,50,10)
           C                       ARGUMENT.
ISN 0004        LOGICAL*1 ITEWRD( )
           C                       BYTE ADRESSABLE WORD.
ISN 0005        INTEGER IWORD
           C                       ALLOWS EQUIVALENCING OF WORD TO BYTE
           C                       ADDRESSABLE WORD.
ISN 0006        INTEGER ITEMP
           C                       TEMP LOCATION FOR TESTING LABLE COUNT.
           C
ISN 0007        INTEGER ICRWD
           C                       ICRWD IS THE WORD CONTAINING BYTE
           C                       TO BE UPDATED
           C
ISN 0008        EQUIVALENCE(IWORD,ITEWRD(1))
           C
           C
           C  PROCEDURE
           C  ---------
```

```
                 C  FIGURE OUT THE NUMBER OF WORD TO USE FOR STORING BYTE                     PAC00740
                 C                                                                            PAC00750
    ISN 0009          ICRNWD=(IX-1)/4+1                                                       PAC00760
                 C                                                                            PAC00770
    ISN 0010          IWORD=LOTBLE(ICRNWD,IY,IPLOT)                                           PAC00780
                 C                                                                            PAC00790
                 C  IBYTE RETURN FOR 1,2,3,0 AS IX=1,2,3,4                                    PAC00800
                 C  MAKE IBYTE=4 INSTEAD OF ZERO                                              PAC00810
                 C                                                                            PAC00820
    ISN 0011          IBYTE=MOD(IX,4)                                                         PAC00830
    ISN 0012          IF(IBYTE.EQ.0) IBYTE=4                                                  PAC00840
                 C                                                                            PAC00850
                 C  LOCATE THE BYTE AND MOVE INTO ITEMP WORD                                  PAC00860
                 C                                                                            PAC00870
    ISN 0014          ITEMP=ITEWRD(IBYTE)                                                     PAC00880
                 C                                                                            PAC00890
                 C  INCREMENT ITEMP BY ONE, IF ITEMP<255 STORE BACK, ELSE RETURN              PAC00900
                 C                                                                            PAC00910
    ISN 0015          ITEMP=ITEMP+1                                                           PAC00920
    ISN 0016          IF(ITEMP.GT.255) GO TO 900                                              PAC00930
    ISN 0018          ITEWRD(IBYTE)=ITEMP                                                     PAC00940
    ISN 0019          LOTBLE(ICRNWD,IY,IPLOT)=IWORD                                           PAC00950
    ISN 0020      900 RETURN                                                                  PAC00960
    ISN 0021          END                                                                     PAC00970
```

```
                    *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL   INTERNAL STATEMENT NUMBERS
IX       0002  0009  0011
IY       0002  0010  0019
MOD      0011
PACK     0002
IBYTE    0011  0012  0012  0014  0018
IPLOT    0002  0010  0019
ITEMP    0006  0014  0015, 0015  0016  0018
IWORD    0005  0008  0010  0019
ICRNWD   0007  0009  0010  0019
ITEWRD   0004  0008  0014  0018
LOTBLE   0002  0003  0010  0019

                    *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
 LABEL   DEFINED   REFERENCES
  900     0020      0016

                              /   PACK /   SIZE OF PROGRAM 0001E2 HEXADECIMAL BYTES

    NAME   TAG  TYPE  ADD.       NAME   TAG  TYPE  ADD.      NAME   TAG  TYPE  ADD.      NAME   TAG  TYPE  ADD.
      IX   FA   I*4   000098       IY   F    I*4   00009C    PACK        R*4   0000A0    IBYTE SF    I*4   0000A4
   IPLOT   F    I*4   0000A8    ITEMP   SF   I*4   0000AC    IWORD SF E  I*4   0000B8   ICRNWD SF    I*4   0000B0
  ITEWRD   SF E L*1   0000B8   LOTBLE   SF XR I*4  000000

 SOURCE STATEMENT LABELS

   LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR
    900    20  000170

 COMPILER GENERATED LABELS

   LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR
  100001    2  0000C8        100002   13  000126        100003   14  00012E        100004   18  000144

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*     SOURCE STATEMENTS =      20, PROGRAM SIZE =      482, SUBPROGRAM NAME =  PACK
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                              288K BYTES OF CORE NOT USED
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

ISN 0002              SUBROUTINE PPLTIN(
                C                          PARPLT INPUT FROM CARDS
                     O NCROP,
                C                 TOTAL NUMBER OF CROPS
                     O RCROP,
                C                 CROPS TO BE EVALUATED
                     O RANGE,
                C                 MIN AND MAX OF ALPHA, BETA, TO
                     O IERR)
                C                 ERROR FLAG
                C                  0 - OK
                C                  1 - ERROR INCOUNTERED
                C ---------------------------------------------------
                C
                C
                C HISTORY
                C -------
                C
                C     MARY TOMPKINS     LEMSCO     02/09/80     ORIGINAL CODE
                C
                C METHOD
                C ------
                C
                C     READS,CLASSIFIES, AND ANALYZES CARDS DESCRIBING THE FOLLOWING:
                C        AI,SEGMENT,DATE,FILE    DOCUMENTATION _- READ AND WRITTEN
                C        ALPHARNG                MIN MAX RANGE FOR ALPHA COEFS.
                C        BETARNG                 MIN MAX RANGE FOR BETA COEFS.
                C        TORNG                   MIN MAX RANGE FOR TO COEFS.
                C        CROP                    1-3 CROPS TO MAP. ENBEDDED BLANKS
                C                                ARE INCLUDED IN NAME.
                C        *END                    SPECIFIES THE END OF USER DEFINED CARDS
                C
                C
                C EXTERNAL REFERENCE
                C ------------------
                C
                C     ICE    INTEGER CHARACTER EQUIVALENT
                C     IVALUE ALLOWS END LINE TESTING / STORING OF QUOTED LITERALS.
                C
                C
                C EXCEPTIONS
                C ----------
                C
                C     IF ANY OF THE FOLLOWING CONDITIONS EXIST IERR IS SET EQUAL
                C     TO 1. A DIAGNOSTIC MESSAGE IS ISSUED AND EXCUTION CONTINUES.
                C        1.MORE/LESS THAN 2 NUMBERS ON THE ALPHA,BETA,TO RANGE CARDS.
                C        2.MIN>=MAX ON THE ALPHA,BETA,TO RANGE CARDS.
                C        3.ALPHA,BETA,TO RANGE CARD MISSING.
                C        4.MORE THAN 3 CROPS ON A CROP CARD.
                C        5.CROP CARD MISSING.
                C     IF A CONTROL CARD IS NOT RECOGNIZED AS ONE OF THE DEFINED
                C     TYPE A WARNING MESSAGE IS PRINTED.
                C
                C
                C LOCAL VARIABLES
                C ---------------
                C
ISN 0003              LOGICAL*1 KHAR(80)
                C                         1 CARD IMAGE. PACKED
ISN 0004              INTEGER ICOFF(3)
                C                         ARRAY CONTAINING NAMES OF COEFS.
ISN 0005              INTEGER NNKARD
                C                         COUNT OF NUM. ON CARD (2 REQ.)
ISN 0006              INTEGER NUMCNT
                C                         SET ACCORDING TO SPECIFIC COFF
ISN 0007              INTEGER KOLTMP
                C                         COUNTER TO ADJUST FOR BYTES CONTAINING
                C                         A DIGIT.
ISN 0008              INTEGER KOLCHA
                C                         COLUMNS OF CHARS ON CARD
ISN 0009              INTEGER IUNIT
                C                         UNIT OF DIGIT
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0010          INTEGER NDIV
               C                                       NUMBER OF DIGITS AFTER DECIMAL POINT
ISN 0011          INTEGER KARTMP
               C                                       COUNTER TO ADJUST FOR BYTES CONTAINING
               C                                       CHARACTERS
ISN 0012          REAL RANGE(6)
               C                                       ARGUMENT
ISN 0013          REAL RCROP(4,3)
               C                                       ARGUMENT
ISN 0014          INTEGER KEYWRD
               C                                       KEY WORD IN CARD (BYTES 1-4)
ISN 0015          REAL RCRPWD(20)
               C                                       ONE CROP PACKED IN WORDS
ISN 0016          LOGICAL*1 RCRPBY(80)
               C                                       ONE CROP PACKED IN BYTES
ISN 0017          EQUIVALENCE(KHAR(1),KEYWRD)
ISN 0018          EQUIVALENCE(RCRPWD(1),RCRPBY(1))
ISN 0019          INTEGER INVEC(9)/'AI','SEGM','DATE','FILE','ALPH','BETA','TORN',
                &  'CROP','*END'/
               C
               C
               C PROCEDURE
               C ---------
               C
               C
               C INITIALIZE VARIABLES
               C
ISN 0020          RCROP(1,1) = IVALUE('    ')
ISN 0021          IERR = 0
ISN 0022          DO 10 I = 1,6
ISN 0023             RANGE(I) = 0
ISN 0024       10 CONTINUE
ISN 0025          ICOEF(1) = IVALUE('ALPH')
ISN 0026          ICOEF(2) = IVALUE('BETA')
ISN 0027          ICOEF(3) = IVALUE('TO  ')
               C
               C
               C WRITE HEADING FOR INPUT SUMMARY
               C
ISN 0028          WRITE(20,200)
ISN 0029      200 FORMAT(//' INPUT SUMMARY'//)
               C
               C
               C READ CARD, PRINT CARD ON SUMMARY
               C
ISN 0030      210 READ(21,220,END=890)(KHAR(I),I = 1,80)
ISN 0031      220 FORMAT(80A1)
ISN 0032          WRITE(20,230)(KHAR(I),I = 1,80)
ISN 0033      230 FORMAT(1X,80A1)
               C
               C
               C ANALYZE CARD
               C -- DETERMINE CARD TYPE
               C
ISN 0034          DO 240 I = 1,9
ISN 0035             IF(KEYWRD.EQ.INVEC(I)) GO TO (300,300,300,300,310,310,310,
                &    500,800),I
ISN 0037      240 CONTINUE
               C
               C
               C INVALID CARD TYPE
               C
ISN 0038          WRITE(20,250)
ISN 0039      250 FORMAT(' INVALID INPUT CARD -- IGNORED')
ISN 0040          GO TO 210
               C
               C
               C DOCUMENTARY CARDS: AI,SEGMENT #,DATE,FILE
               C
ISN 0041      300 GO TO 210
               C
               C
               C MIN AND MAX RANGE FOR COEFS -- ALPHA BETA TO
               C          RANGE(1) MIN-ALPHA      RANGE(2) MAX-ALPHA
               C          RANGE(3) MIN-BETA       RANGE(4) MAX-BETA
               C          RANGE(5) MIN-TO         RANGE(6) MAX-TO
               C
ISN 0042      310 IF(KEYWRD.EQ.IVALUE('ALPH'))NUMCNT = 0
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
ISN 0044            IF(KEYWRD.EQ.IVALUE('BETA'))NUMCNT = 2                          PPL01520
ISN 0046            IF(KEYWRD.EQ.IVALUE('TORN'))NUMCNT = 4                          PPL01530
ISN 0048            NNKARD = 0                                                      PPL01540
                C                                                                   PPL01550
                C                                                                   PPL01560
                C VALID INPUTS START IN COLUMN 11, ENDS IN COLUMN 72.               PPL01570
                C INPUT MUST BE NUMERIC -- TWO NUMBERS PER CARD.                    PPL01580
                C                                                                   PPL01590
ISN 0049            KOLCHA = 11                                                     PPL01600
ISN 0050          320 DO 330 I = KOLCHA,72                                          PPL01610
ISN 0051            KOL = I                                                         PPL01620
ISN 0052            IF((ICE(KHAR(I)).GE.ICE('0').AND.                               PPL01630
                &       ICE(KHAR(I)).LE.ICE('9')).OR.                               PPL01640
                &      ICE(KHAR(I)).EQ.ICE('.')) GO TO 340                          PPL01650
ISN 0054          330 CONTINUE                                                      PPL01660
ISN 0055            GO TO 210                                                       PPL01670
                C                                                                   PPL01680
                C                                                                   PPL01690
                C CHARACTER IS NUMERIC -- DECODE                                    PPL01700
                C                                                                   PPL01710
ISN 0056          340 NNKARD = NNKARD + 1                                           PPL01720
ISN 0057            NUMCNT = NUMCNT + 1                                             PPL01730
ISN 0058            IUNIT = 1                                                       PPL01740
ISN 0059            NDIV = 1                                                        PPL01750
ISN 0060            IREAL = 0                                                       PPL01760
                C                                                                   PPL01770
ISN 0061            DO 380 NUMCOL = KOL,72                                          PPL01780
ISN 0062              KOLTMP = NUMCOL                                               PPL01790
ISN 0063              IF(ICE(KHAR(NUMCOL)).EQ.ICE('.')) IREAL = 1                   PPL01800
ISN 0065              IF(ICE(KHAR(NUMCOL)).EQ.ICE('.')) GO TO 380                   PPL01810
ISN 0067              IF(ICE(KHAR(NUMCOL)).LT.ICE('0').OR.                          PPL01820
                &        ICE(KHAR(NUMCOL)).GT.ICE('9')) GO TO 390                   PPL01830
ISN 0069              IF(NNKARD.GT.2)GO TO 400                                      PPL01840
ISN 0071              RANGE(NUMCNT) = RANGE(NUMCNT)*IUNIT+(ICE(KHAR(NUMCOL))-ICE('0'))PPL01850
ISN 0072              IF(IREAL.EQ.1)NDIV = NDIV * 10                                PPL01860
ISN 0074              IUNIT = 10                                                    PPL01870
ISN 0075          380 CONTINUE                                                      PPL01880
                C                                                                   PPL01890
ISN 0076          390 IF(IREAL.NE.0.AND.RANGE(NUMCNT).GT.0)                         PPL01900
                &    RANGE(NUMCNT) = RANGE(NUMCNT)/NDIV                           PPL01910
                C                                                                   PPL01920
ISN 0078            KOLCHA = KOLTMP + 1                                             PPL01930
ISN 0079            IF(KOLCHA.LT.72) GO TO 320                                      PPL01940
ISN 0081            GO TO 210                                                       PPL01950
                C                                                                   PPL01960
                C                                                                   PPL01970
                C MORE THAN TWO NUMBERS ON A CARD                                   PPL01980
                C                                                                   PPL01990
ISN 0082          400 WRITE(5,410)                                                  PPL02000
ISN 0083          410 FORMAT(' TOO MANY NUMBERS ON CONTROL CARD.')                   PPL02010
ISN 0084            IERR = 1                                                        PPL02020
ISN 0085            GO TO 210                                                       PPL02030
                C                                                                   PPL02040
                C                                                                   PPL02050
                C CROP CARD VALID INPUT STARTS IN COLUMN 11, ENDS IN COLUMN 72      PPL02060
                C INPUT IS CHARACTER, MAXIMUN OF 3 CROPS, ENBEDED BLANKS ARE        PPL02070
                C IGNORED.                                                          PPL02080
                C                                                                   PPL02090
ISN 0086          500 KAR = 11                                                      PPL02100
ISN 0087            NCROP = 0                                                       PPL02110
ISN 0088          505 DO 510 KARCOL = KAR,72                                        PPL02120
ISN 0089            KARSTR = KARCOL                                                 PPL02130
ISN 0090            IF(ICE(KHAR(KARCOL)).NE.ICE(' ')) GO TO 530                     PPL02140
ISN 0092          510 CONTINUE                                                      PPL02150
ISN 0093            GO TO 210                                                       PPL02160
                C                                                                   PPL02170
                C                                                                   PPL02180
                C CHARACTER IS FIRST IN CROP (NON BLANK)                            PPL02190
                C                                                                   PPL02200
ISN 0094          530 NCROP = NCROP + 1                                             PPL02210
ISN 0095            KROCHA = 1                                                      PPL02220
ISN 0096            DO 540 I = 1,16                                                 PPL02230
ISN 0097              CROPBY(I) = ICE(' ')                                          PPL02240
ISN 0098          540 CONTINUE                                                      PPL02250
                C                                                                   PPL02260
ISN 0099            DO 560 I = KARSTR,72                                            PPL02270
ISN 0100              KARTMP = I                                                    PPL02280
ISN 0101              IF(ICE(KHAR(I)).EQ.ICE(',')) GO TO 600                        PPL02290
```

```
ISN 0103                RCRPRY(KPOCHA) = KHAR(I)                                        PPL02300
ISN 0104                KPOCHA = KPOCHA + 1                                             PPL02310
ISN 0105          560 CONTINUE                                                          PPL02320
                  C                                                                     PPL02330
                  C                                                                     PPL02340
                  C STORE 16 BYTES IN RCROP ARRAY                                       PPL02350
                  C                                                                     PPL02360
ISN 0106          600 IF(NCROP.LE.3)GO TO 630                                           PPL02370
ISN 0108              WRITE(5,610)                                                      PPL02380
ISN 0109          610 FORMAT(' TOO MANY CROPS SPECIFIED ON CONTROL CARD.')               PPL02390
ISN 0110              NCROP = NCROP - 1                                                 PPL02400
ISN 0111              IERR = 1                                                          PPL02410
ISN 0112              GO TO 710                                                         PPL02420
ISN 0113          630 DO 650 I = 1,4                                                    PPL02430
ISN 0114              RCROP(I,NCROP) = RCPPWD(I)                                        PPL02440
ISN 0115          650 CONTINUE                                                          PPL02450
                  C                                                                     PPL02460
                  C                                                                     PPL02470
                  C                                                                     PPL02480
                  C CHECK CURRENT BYTE                                                  PPL02490
                  C                                                                     PPL02500
ISN 0116              KAR = KARTMP + 1                                                  PPL02510
ISN 0117              IF(KARCOL.LT.72) GO TO 505                                        PPL02520
ISN 0119              GO TO 210                                                         PPL02530
                  C                                                                     PPL02540
                  C *END CHECK FOR ERRORS                                               PPL02550
                  C                                                                     PPL02560
                  C --                                                                  PPL02570
                  C    RANGE MIN >= MAX                                                 PPL02580
                  C                                                                     PPL02590
ISN 0120          800 DO 840 I = 1,6,2                                                  PPL02600
ISN 0121              IF(RANGE(I).GE.RANGE(I+1))IERR = 2                                PPL02610
ISN 0123              IF(RANGE(I+1).EQ.0)WRITE(5,820)ICOEF((I+1)/2)                     PPL02620
ISN 0125          820    FORMAT(1X,A4,' CARD MISSING OR INCORRECTLY ENTERED.')          PPL02630
ISN 0126              IF(IERR.EQ.2)WRITE(5,830)RANGE(I),RANGE(I+1)                      PPL02640
ISN 0128          830    FORMAT(' RANGE ',F8.2,2X,F8.2,' UNACCEPTABLE')                 PPL02650
ISN 0129              IF(IERR.EQ.2)IERR = 1                                             PPL02660
ISN 0131          840 CONTINUE                                                          PPL02670
                  C                                                                     PPL02680
                  C --                                                                  PPL02690
                  C    CROP CARD MISSING??                                              PPL02700
                  C                                                                     PPL02710
                  C                                                                     PPL02720
ISN 0132              IF(RCROP(1,1).NE.IVALUE('    ')) GO TO 890                      PPL02730
ISN 0134              WRITE(5,850)                                                      PPL02740
ISN 0135          850 FORMAT(' CROP CARD MISSING.')                                     PPL02750
ISN 0136              IERR = 1                                                          PPL02760
                  C                                                                     PPL02770
                  C                                                                     PPL02780
                  C                                                                     PPL02790
                  C CARDS ARE COMMENT CARDS                                             PPL02800
                  C                                                                     PPL02810
ISN 0137          890 READ(21,220,END=900)(KHAR(I),I = 1,80)                            PPL02820
ISN 0138              WRITE(20,230)(KHAR(I),I = 1,80)                                   PPL02830
ISN 0139              GO TO 890                                                         PPL02840
ISN 0140          900 RETURN                                                            PPL02850
ISN 0141              END                                                               PPL02860
```


*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL STATEMENT NUMBERS																		
I	0022	0023	0030	0030	0030	0032	0032	0032	0034	0035	0035	0050	0051	0052	0052	0052	0096	0097	0099
	0100	0101	0103	0113	0114	0114	0120	0121	0121	0123	0123	0126	0126	0137	0137	0137	0138	0138	0138
ICE	0052	0052	0052	0052	0052	0052	0063	0063	0065	0065	0067	0067	0067	0067	0071	0071	0090	0090	0097
	0101	0101																	
KAR	0086	0088	0116																
KOL	0051	0061																	
IERR	0002	0021	0084	0111	0121	0126	0129	0129	0136										
KHAR	0003	0017	0030	0032	0052	0052	0052	0063	0065	0067	0067	0071	0090	0101	0103	0137	0138		
NDIV	0010	0059	0072	0072	0076														
ICOEF	0004	0025	0026	0027	0123														
INVEC	0019	0019	0035																
IREAL	0060	0063	0072	0076															
IUNIT	0009	0058	0071	0074															
NCROP	0002	0087	0094	0094	0106	0110	0110	0114											
RANGE	0002	0012	0023	0071	0071	0076	0076	0076	0121	0121	0123	0126	0126						
RCROP	0002	0013	0020	0114	0132														
IVALUE	0020	0025	0026	0027	0043	0044	0046	0132											

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL STATEMENT NUMBERS										
KARCOL	0088	0089	0090	0117							
KARSTR	0089	0099									
KARTMP	0011	0100	0116								
KEYWRD	0014	0017	0035	0042	0044	0046					
KOLCHA	0008	0049	0050	0078	0079						
KOLTMP	0007	0062	0078								
KROCHA	0095	0103	0104	0104							
NNKARD	0005	0048	0056	0056	0069						
NUMCNT	0006	0042	0044	0046	0057	0057	0071	0071	0076	0076	0076
NUMCOL	0061	0062	0063	0065	0067	0067	0071				
PPLTIN	0002										
RCRPBY	0016	0018	0097	0103							
RCRPWD	0015	0018	0114								

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

LABEL	DEFINED	REFERENCES						
10	0024	0022						
200	0029	0028						
210	0030	0040	0041	0055	0081	0085	0093	0112	0119
220	0031	0030	0137						
230	0033	0032	0138						
240	0037	0034							
250	0039	0038							
300	0041	0035	0035	0035	0035				
310	0042	0035	0035	0035					
320	0050	0079							
330	0054	0050							
340	0056	0052							
380	0075	0061	0065						
390	0076	0067							
400	0082	0069							
410	0083	0082							
500	0086	0035							
505	0089	0117							
510	0092	0088							
530	0094	0090							
540	0098	0096							
560	0105	0099							
600	0106	0101							
610	0109	0108							
630	0113	0106							
650	0115	0113							
800	0120	0035							
820	0125	0123							
830	0129	0126							
840	0131	0120							
850	0135	0134							
890	0137	0030	0132	0139					
900	0140	0137							

/ PPLTIN / SIZE OF PROGRAM 000C24 HEXADECIMAL BYTES

NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
I	SFA	I*4	0002D4	ICE	XF	I*4	000000	KAR	SF	I*4	0002D8	KOL	SF	I*4	0002DC
IERR	S	I*4	0002E0	KHAR	SFA E	L*1	000358	NDIV	SF	I*4	0002E4	ICOEF	SF	I*4	000310
INVEC		I*4	000328	IREAL	S	I*4	0002E8	IUNIT	SF	I*4	0002EC	NCROP	SF	I*4	0002F0
RANGE	SF XR	R*4	000000	RCROP	S XR	R*4	000000	IBCOM#	F XF	I*4	000000	IVALUE	F XF	I*4	000000
KARCOL	SFA	I*4	0002F4	KARSTR	SF	I*4	0002F8	KARTMP	SF	I*4	0002FC	KEYWRD	F E	I*4	000358
KOLCHA	SF	I*4	000300	KOLTMP	SF	I*4	000304	KROCHA	SF	I*4	000308	NNKARD	SF	I*4	00030C
NUMCNT	SF	I*4	000310	NUMCOL	SFA	I*4	000314	PPLTIN		R*4	000318	RCRPBY	S E	L*1	000380
RCRPWD	F E	R*4	000380												

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
10	24	0004A0	210	30	0004E0	240	37	000574	300	41	0005A0
310	42	0005A4	320	50	0005E6	330	54	0006B2	340	56	0006C8
380	75	000836	390	76	000840	400	82	000894	500	86	0008B6
505	89	0008C2	510	92	000902	530	94	000918	540	98	000934
560	105	0009A2	600	106	0009B4	630	113	0009D8	650	115	000A00
800	120	000A2C	820	131	000A9E	830	137	000B3A	900	140	000B94

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100001	2	000438	100002	23	000480	100003	25	000482	100004	30	000504
100009	35	000548	100010	36	000550	100012	38	000586	100013	43	0005B6
100014	44	0005BA	100015	45	0005CC	100016	46	0005D0	100017	47	0005E2
100018	48	0005E6	100019	51	0005FF	100020	55	0006C4	100021	62	0006E8
100022	64	000718	100023	65	00071C	100024	67	000748	200001	68	000774
100025	69	0007A0	100026	71	0007AC	100027	73	000822	100028	74	00082E
200002	77	000848	100030	77	000854	100031	78	000884	100032	81	000896
100033	83	0008CA	100034	93	000914	100035	97	000922	100036	99	000946
100037	100	00094E	100038	103	000986	100040	108	0009BC	100041	114	0009DC
100042	116	000A12	100043	119	000A28	100044	121	000A30	100045	122	000A4C
100046	123	000A50	100047	124	000A62	100048	126	000A96	100049	127	000A9E
100050	129	000AD4	100051	130	000ADC	100052	132	000AF2	100053	134	000B22
100054	137	000B50									

FORMAT STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
200	29	000028	220	31	00003E	230	33	000044	250	39	00004C
410	83	000C6E	610	109	000094	820	125	0000C2	830	128	0000EF
850	135	000111									

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

STATISTICS SOURCE STATEMENTS = 140, PROGRAM SIZE = 3108, SUBPROGRAM NAME =PPLTIN

STATISTICS NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ****** 260K BYTES OF CORE NOT USED

ORIGINAL PAGE IS
OF POOR QUALITY

```
*LEVEL 2.3.0 (JUNE 78)                    OS/360  FORTRAN H EXTENDED           DATE 81.140/12.30.38        PAGE   1
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
   ISN 0002               REAL FUNCTION RVALUE(REL)                                   RVA00010
                C                             ALLOW IN LINE STORAGE AND TESTING       RVA00020
                C                             OF QUOTED LITERALS.                     RVA00030
                C         ----------------------------------------------------------  RVA00040
                C                                                                     RVA00050
                C                                                                     RVA00060
                C HISTORY                                                             RVA00070
                C -------                                                             RVA00080
                C                                                                     RVA00090
                C         M A TOMPKINS     LEMSCO      02/27/81     ORIGINAL CODE     RVA00100
                C                                                                     RVA00110
                C                                                                     RVA00120
                C METHOD                                                              RVA00130
                C ------                                                              RVA00140
                C                                                                     RVA00150
                C         SET INPUT TO FUNCTION. RETURN.                              RVA00160
                C                                                                     RVA00170
                C PROCEDURE                                                           RVA00180
                C ---------                                                           RVA00190
                C                                                                     RVA00200
                C                                                                     RVA00210
   ISN 0003               RVALUE = REL                                                RVA00220
   ISN 0004               RETURN                                                      RVA00230
   ISN 0005               END                                                         RVA00240

                    *****F O R T R A N   C R O S S   R E F E R E N C E   L I S T I N G*****
SYMBOL  INTERNAL STATEMENT NUMBERS
REL     0002  0003
RVALUE  0002  0003

                                              /  RVALUE /     SIZE OF PROGRAM 0000C2 HEXADECIMAL BYTES

     NAME   TAG   TYPE  ADD.       NAME    TAG   TYPE  ADD.       NAME   TAG   TYPE  ADD.       NAME   TAG   TYPE  ADD.
      REL  F      R*4   000080    RVALUE S       R*4   000084

  COMPILER GENERATED LABELS

    LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR          LABEL  ISN   ADDR
   100001    2  000088
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*    SOURCE STATEMENTS =       4, PROGRAM SIZE =      194, SUBPROGRAM NAME =RVALUE
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                      296K BYTES OF CORE NOT USED
```

REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

```
ISN 0002            SUBROUTINE SCAT (IR,IPLOT,XV,YV)                                    SCA00010
ISN 0003            COMMON/PLOT/XMX(10),XMN(10),YMX(10),YMN(10)                         SCA00020
ISN 0004            COMMON/DEFGT/GTRSEG,TYP,GTDATE                                      SCA00030
ISN 0005            COMMON/DEFCL/KLASEG,KYP,KLDATE                                      SCA00040
ISN 0006            COMMON/HEADER/HEAD(18,20)                                           SCA00050
ISN 0007            INTEGER KLDATE(3)                                                   SCA00060
ISN 0008            INTEGER GTDATE(3)                                                   SCA00070
ISN 0009            DIMENSION AAL(3),BUFF(50),DUM(1)                                    SCA00080
ISN 0010            DIMENSION ID(13,50,10),IV(100)                                      SCA00090
ISN 0011            DIMENSION IS(4),G(6),IT(36)                                         SCA00100
ISN 0012            DIMENSION CCC(10),XVA(10),XVS(10),YVA(10),YVS(10),XYVS(10)          SCA00110
ISN 0013            DIMENSION XUN(10),XOV(10),YUN(10),YOV(10)                           SCA00120
ISN 0014            DIMENSION AXLA(3)                                                   SCA00130
ISN 0015            INTEGER*2 HMS(3)                                                    SCA00140
ISN 0016            INTEGER*4 DATE(2)                                                   SCA00150
ISN 0017            DATA IS/1H ,1H-,1HI,1H./                                            SCA00160
ISN 0018            DATA AXLA/4HX AX,4HIS L,4HABEL/                                     SCA00170
ISN 0019            DATA BLANK/4H    /                                                  SCA00180
ISN 0020            DATA IT/1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1HA,1HB,1HC,1HD,1HE,1HSCA00190
                   1F,1HG,1HH,1HI,1HJ,1HK,1HL,1HM,1HN,1HO,1HP,1HQ,1HR,1HS,1HT,1HU,1HV,SCA00200
                   2 1HW,1HX,1HY,1HZ,1H*/                                              SCA00210
ISN 0021            IC=IR                                                               SCA00220
ISN 0022            IJKL=1                                                              SCA00230
ISN 0023            GO TO (1,2,3), IC                                                   SCA00240
ISN 0024          1 XUN(IPLOT)=0.                                                       SCA00250
ISN 0025            XVA(IPLOT)=0.                                                       SCA00260
ISN 0026            XVS(IPLOT)=0.                                                       SCA00270
ISN 0027            YVS(IPLOT)=0.                                                       SCA00280
ISN 0028            YVA(IPLOT)=0.                                                       SCA00290
ISN 0029            XYVS(IPLOT)=0.                                                      SCA00300
ISN 0030            CCC(IPLOT)=0.                                                       SCA00310
ISN 0031            XOV(IPLOT)=0.                                                       SCA00320
ISN 0032            YUN(IPLOT)=0.                                                       SCA00330
ISN 0033            YOV(IPLOT)=0.                                                       SCA00340
ISN 0034            DO 4 I=1,50                                                         SCA00350
ISN 0035            DO 4 J=1,13                                                         SCA00360
ISN 0036          4 ID(J,I,IPLOT)=0                                                     SCA00370
ISN 0037            RETURN                                                              SCA00380
ISN 0038          2 CONTINUE                                                            SCA00390
ISN 0039            DX=(XMX(IPLOT)-XMN(IPLOT))/50                                       SCA00400
ISN 0040            DY=(YMX(IPLOT)-YMN(IPLOT))/50                                       SCA00410
ISN 0041            X=XV-XMN(IPLOT)                                                     SCA00420
ISN 0042            Y=YV-YMN(IPLOT)                                                     SCA00430
ISN 0043            IF(X.LT.0.) XUN(IPLOT)=XUN(IPLOT)+1.                                SCA00440
ISN 0045            IF(Y.LT.0.) YUN(IPLOT)=YUN(IPLOT)+1.                                SCA00450
ISN 0047            XIX=X/DX+1.                                                         SCA00460
ISN 0048            YIY=Y/DY+1.                                                         SCA00470
ISN 0049            IXOV=0                                                              SCA00480
ISN 0050            IYOV=0                                                              SCA00490
ISN 0051            IF(XIX.LT.51) GO TO 500                                             SCA00500
ISN 0053            IXOV=1                                                              SCA00510
ISN 0054          500 IF(YIY.LT.51) GO TO 501                                           SCA00520
ISN 0056            IYOV=1                                                              SCA00530
ISN 0057          501 IF(IXOV.EQ.1) XOV(IPLOT)=XOV(IPLOT)+1.                            SCA00540
ISN 0059            IF(IYOV.EQ.1) YOV(IPLOT)=YOV(IPLOT)+1.                              SCA00550
ISN 0061            IF(X.LT.0.) RETURN                                                  SCA00560
ISN 0063            IF(Y.LT.0.) RETURN                                                  SCA00570
ISN 0065            IF(IXOV.EQ.1.) RETURN                                               SCA00580
ISN 0067            IF(IYOV.EQ.1.) RETURN                                               SCA00590
ISN 0069            IX=XIX                                                              SCA00600
ISN 0070            IY=YIY                                                              SCA00610
ISN 0071            CALL PACK(ID,IX,IY,IPLOT)                                           SCA00620
ISN 0072            XVA(IPLOT)=XVA(IPLOT)+XV                                            SCA00630
ISN 0073            YVA(IPLOT)=YVA(IPLOT)+YV                                            SCA00640
ISN 0074            XVS(IPLOT)=XVS(IPLOT)+XV*XV                                         SCA00650
ISN 0075            YVS(IPLOT)=YVS(IPLOT)+YV*YV                                         SCA00660
ISN 0076            XYVS(IPLOT)=XYVS(IPLOT)+XV*YV                                       SCA00670
ISN 0077            CCC(IPLOT)=CCC(IPLOT)+1.                                            SCA00680
ISN 0078            RETURN                                                              SCA00690
ISN 0079          3 CONTINUE                                                            SCA00700
ISN 0080            IF(CCC(IPLOT).EQ.0.) RETURN                                         SCA00710
ISN 0082            KO=1                                                                SCA00720
ISN 0083            WRITE(6,97) IPLOT                                                   SCA00730
```

```
ISN 0084        97 FORMAT(1H1,' PLOT OUTPUT NO ',I2)                                    SCA00740
ISN 0085           CALL CPTIME(DATE,HMS,TI,VT)                                          SCA00750
ISN 0086           WRITE(6,123)(DATE(I),I=1,2),(HMS(J),J=1,3)                           SCA00760
ISN 0087       123 FORMAT(//,1X,' DATE ',2A4,'  TIME ',I2,':',I2,':',I2)               SCA00770
ISN 0088           WRITE(6,124)GTRSEG,TYR,(GTDATE(I),I=1,3),KLASEG,KYR,                 SCA00780
                  1(KLDATE(I),I=1,3)                                                   SCA00790
ISN 0089       124 FORMAT(/2X,'GROUND TRUTH FILE- ',I5,I2,I1,I1,I1,30X,                SCA00800
                  1'CLASSIFICATION FILE- ',I5,I2,I1,I1,I1,//)                          SCA00810
ISN 0090           IYLAB=IPLOT*2                                                        SCA00820
ISN 0091           IXLAB=IYLAB-1                                                        SCA00830
ISN 0092       100 FORMAT(20X,'Y AXIS LABEL- ',18A4)                                    SCA00840
ISN 0093        21 CONTINUE                                                             SCA00850
ISN 0094           DO 5 I=1,6                                                           SCA00860
ISN 0095           A=I-1                                                                SCA00870
ISN 0096           DY=(YMX(IPLOT)-YMN(IPLOT))/50.                                       SCA00880
ISN 0097         5 G(I)=YMN(IPLOT)+A*10.*DY                                             SCA00890
ISN 0098           WRITE(6,95) (G(I),I=1,6)                                             SCA00900
ISN 0099           IF(KQ.EQ.1) GO TO 20                                                 SCA00910
ISN 0101           WRITE(6,100) (HEAD(I,IYLAB),I=1,18)                                  SCA00920
ISN 0102           WRITE(6,992)XUN(IPLOT),XOV(IPLOT),YUN(IPLOT),YOV(IPLOT),CCC(IPLOT)   SCA00930
ISN 0103           IF(CCC(IPLOT).LT.3.) GO TO 995                                       SCA00940
ISN 0105           X=XVA(IPLOT)/CCC(IPLOT)                                              SCA00950
ISN 0106           Y=YVA(IPLOT)/CCC(IPLOT)                                              SCA00960
ISN 0107           XX=1./CCC(IPLOT)/(CCC(IPLOT)-1.)*                                    SCA00970
                  1 (CCC(IPLOT)*XVS(IPLOT)-XVA(IPLOT)*XVA(IPLOT))                      SCA00980
ISN 0108           YY=1./CCC(IPLOT)/(CCC(IPLOT)-1.)*                                    SCA00990
                  1 (CCC(IPLOT)*YVS(IPLOT)-YVA(IPLOT)*YVA(IPLOT))                      SCA01000
ISN 0109           XX=SQRT(XX)                                                          SCA01010
ISN 0110           YY=SQRT(YY)                                                          SCA01020
ISN 0111           XXX=XVS(IPLOT)/CCC(IPLOT)                                            SCA01030
ISN 0112           IF(CCC(IPLOT).GT.1.) GO TO 600                                       SCA01040
ISN 0114           A1=0.0                                                               SCA01050
ISN 0115           A0=0.0                                                               SCA01060
ISN 0116           R2=0.0                                                               SCA01070
ISN 0117           SYX=0.0                                                              SCA01080
ISN 0118           S0=0.0                                                               SCA01090
ISN 0119           GO TO 996                                                            SCA01100
ISN 0120       600 XSUM=YVS(IPLOT)-YVA(IPLOT)*YVA(IPLOT)/CCC(IPLOT)                     SCA01110
ISN 0121           YSUM=XVS(IPLOT)-XVA(IPLOT)*XVA(IPLOT)/CCC(IPLOT)                     SCA01120
ISN 0122           A1=(XYVS(IPLOT)-YVA(IPLOT)*XVA(IPLOT)/CCC(IPLOT))/XSUM               SCA01130
ISN 0123           A0=XVA(IPLOT)/CCC(IPLOT)-(A1*YVA(IPLOT)/CCC(IPLOT))                  SCA01140
ISN 0124           R2=(XYVS(IPLOT)-YVA(IPLOT)*XVA(IPLOT)/CCC(IPLOT))                    SCA01150
                  *  **(XYVS(IPLOT)-YVA(IPLOT)*XVA(IPLOT)/CCC(IPLOT))/(XSUM*YSUM)      SCA01160
ISN 0125           SYX=SQRT((XVS(IPLOT)-A0*XVA(IPLOT)-A1*XYVS(IPLOT))                   SCA01170
                  *   /(CCC(IPLOT)-2.))                                                SCA01180
ISN 0126           S0=SYX*SQRT(YVS(IPLOT)/XSUM)                                         SCA01190
ISN 0127           S1=SYX/SQRT(XSUM)                                                    SCA01200
ISN 0128           GO TO 996                                                            SCA01210
ISN 0129       995 X=0                                                                  SCA01220
ISN 0130           Y=0                                                                  SCA01230
ISN 0131           XX=0                                                                 SCA01240
ISN 0132           YY=0                                                                 SCA01250
ISN 0133           XXX=0                                                                SCA01260
ISN 0134       996 WRITE(6,994) X,XX,Y,YY,XXX                                           SCA01270
ISN 0135       994 FORMAT('       MEAN  X=',F10.4,'+/-',F10.4,'    MEAN  Y=',F10.4,    SCA01280
                  1'+/-',F10.4,'   MSE= ',F10.4)                                       SCA01290
ISN 0136           WRITE(6,900)A0,S0,A1,S1,R2,SYX                                       SCA01300
ISN 0137       900 FORMAT(1H ,'X = (',F8.3,'+-',F8.3,') + (',F8.3,'+-',F8.3,') * Y',   SCA01310
                  *'  COEFF. OF DETER. = ',F7.5,'  STD. ERROR(Y ON X) = ',F8.3)        SCA01320
ISN 0138           NUND=XUN(IPLOT)                                                      SCA01330
ISN 0139           NOVR=XOV(IPLOT)                                                      SCA01340
ISN 0140           DX=(XMX(IPLOT)-XMN(IPLOT))/50.                                       SCA01350
ISN 0141           NUND=YUN(IPLOT)                                                      SCA01360
ISN 0142           NOVR=YOV(IPLOT)                                                      SCA01370
ISN 0143           DO 750 IBB=1,50                                                      SCA01380
ISN 0144       750 BUFF(IBB)=0.                                                         SCA01390
ISN 0145           RETURN                                                               SCA01400
ISN 0146        20 CONTINUE                                                             SCA01410
ISN 0147        95 FORMAT(F5.4,5F20.4)                                                  SCA01420
ISN 0148           DO 6 I=1,100                                                         SCA01430
ISN 0149         6 IV(I)=IS(2)                                                          SCA01440
ISN 0150           DO 7 I=10,100,10                                                     SCA01450
ISN 0151         7 IV(I) =IT(34)                                                        SCA01460
ISN 0152           WRITE(6,94)IT(18),(IV(I),I=1,100),IT(18)                             SCA01470
                C  PRINT 94,V                                                          SCA01480
ISN 0153           IF(KQ.EQ.2) GO TO 21                                                 SCA01490
ISN 0155           LPT=0                                                                SCA01500
ISN 0156        94 FORMAT(10X,102A1)                                                    SCA01510
```

```
ISN 0157          73 FORMAT(F10.3,102A1,1X,3A4)                                  SCA01520
ISN 0158             DX=(XMX(IPLOT)-XMN(IPLOT))/50.                              SCA01530
ISN 0159             DY=(YMX(IPLOT)-YMN(IPLOT))/50.                              SCA01540
ISN 0160             DO 8 I=1,50                                                 SCA01550
ISN 0161             DO 11 J=1,100                                               SCA01560
ISN 0162          11 IV(J)=IS(1)                                                 SCA01570
ISN 0163             JJ=1                                                        SCA01580
ISN 0164             DO 9 J=1,100,2                                              SCA01590
ISN 0165             CALL UNPACK(II,ID,L,JJ,IPLOT)                               SCA01600
ISN 0166             JJJ=JJ+1                                                    SCA01610
ISN 0167             IF(II.EQ.0) GO TO 10                                        SCA01620
ISN 0169             IV(J) = II(36)                                              SCA01630
ISN 0170             IF(II.LT.36) IV(J)=IT(II)                                   SCA01640
ISN 0172          10 CONTINUE                                                    SCA01650
ISN 0173           9 CONTINUE                                                    SCA01660
ISN 0174             X=1.                                                        SCA01670
ISN 0175             X=XMN(IPLOT)+(X-.5)*DX                                      SCA01680
ISN 0176             DO 802 IP=1,3                                               SCA01690
ISN 0177          802 AAL(IP)=BLANK                                              SCA01700
ISN 0178             IF((L.LT.22).OR.(L.GT.28)) GO TO 800                        SCA01710
ISN 0180             IF(LPT.EQ.0) GO TO 801                                      SCA01720
ISN 0182             LSTRT=3*LPT-2                                               SCA01730
ISN 0183             LEND=LSTPT+2                                                SCA01740
ISN 0184             IK=1                                                        SCA01750
ISN 0185             DO 803 IL=LSTRT,LEND                                        SCA01760
ISN 0186             AAL(IK)=HEAD(IL,IXLAB)                                      SCA01770
ISN 0187          803 IK=IK+1                                                    SCA01780
ISN 0188             LPT=LPT+1                                                   SCA01790
ISN 0189             GO TO 800                                                   SCA01800
ISN 0190          801 AAL(1)=AXLA(1)                                             SCA01810
ISN 0191              AAL(2)=AXLA(2)                                             SCA01820
ISN 0192              AAL(3)=AXLA(3)                                             SCA01830
ISN 0193              LPT=LPT+1                                                  SCA01840
ISN 0194          800 CONTINUE                                                   SCA01850
ISN 0195             WRITE(6,93)X,IS(3),(IV(JJ),JJ=1,100)  ,IS(3),AAL            SCA01860
ISN 0196           8 CONTINUE                                                    SCA01870
ISN 0197          992 FORMAT('          X UNDER=',F6.0,'      X OVER=',F6.0,'      Y USCA01880
                    1NDER=',F6.0,'      Y OVER=',F6.0,'    PLOTTED=',F6.0)      SCA01890
ISN 0198             KQ=2                                                        SCA01900
ISN 0199             GO TO 20                                                    SCA01910
ISN 0200             END                                                         SCA01920
```

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL STATEMENT NUMBERS																		
A	0095	0097																	
G	0011	0097	0098																
I	0034	0036	0086	0086	0086	0088	0088	0088	0088	0088	0098	0094	0095	0097	0098	0098	0098	0101	0101
	0101	0148	0149	0150	0151	0152	0152	0152											
J	0035	0036	0086	0086	0086	0161	0162	0164	0169	0170									
L	0160	0165	0174	0178	0178														
X	0041	0043	0047	0061	0105	0129	0134	0174	0175	0175	0195								
Y	0042	0045	0048	0063	0106	0130	0134												
A0	0116	0123	0125	0136															
A1	0114	0122	0123	0125	0136														
DX	0039	0047	0140	0158	0175														
DY	0040	0048	0096	0097	0159														
IB	0002	0021																	
IC	0021	0023																	
ID	0010	0036	0071	0165															
II	0165	0167	0170	0170															
IK	0184	0186	0187	0187															
IL	0185	0186																	
IP	0176	0177																	
IS	0011	0017	0149	0162	0195	0195													
IT	0011	0020	0151	0152	0152	0169	0170												
IV	0010	0149	0151	0152	0162	0169	0170	0195											
IX	0069	0071																	
IY	0070	0071																	
JJ	0195	0195	0195																
KQ	0082	0099	0153	0198															
R2	0116	0124	0135																
S0	0118	0126	0136																
S1	0127	0136																	
TT	0085																		
VT	0085																		
XV	0002	0041	0072	0074	0074	0076													
XX	0107	0109	0109	0131	0134														

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL	STATEMENT NUMBERS																
YV	0002	0042	0073	0075	0075	0076												
YY	0108	0110	0110	0132	0134													
AAL	0009	0177	0186	0190	0191	0192	0195											
CCC	0012	0030	0077	0077	0080	0102	0103	0105	0106	0107	0107	0107	0108	0108	0108	0111	0112	0120	0121
	0122	0123	0123	0124	0124	0125												
DUM	0009																	
HMS	0015	0085	0086															
IBB	0143	0144																
JJJ	0163	0165	0166	0166														
KYR	0005	0088																
LPT	0155	0180	0182	0188	0189	0193	0193											
SYX	0117	0125	0126	0127	0136													
TYR	0004	0088																
XIX	0047	0051	0069															
XMN	0003	0039	0041	0140	0158	0175												
XMX	0003	0039	0140	0158														
XOV	0013	0031	0057	0057	0102	0139												
XUN	0013	0024	0043	0043	0102	0138												
XVA	0012	0025	0072	0072	0105	0107	0107	0121	0121	0122	0123	0124	0124	0125				
XVS	0012	0026	0074	0074	0107	0111	0121	0125										
XXX	0111	0133	0134															
YIY	0048	0054	0070															
YMN	0003	0040	0042	0096	0097	0159												
YMX	0003	0040	0096	0159														
YOV	0013	0033	0059	0059	0102	0142												
YUN	0013	0032	0045	0045	0102	0141												
YVA	0012	0028	0073	0073	0106	0108	0108	0120	0120	0122	0123	0124	0124					
YVS	0012	0027	0075	0075	0108	0120	0126											
AXLA	0014	0018	0190	0191	0192													
BUFF	0009	0144																
DATE	0016	0085	0086															
HEAD	0006	0101	0186															
IJKL	0022																	
IXOV	0049	0053	0057	0065														
IYOV	0050	0056	0059	0067														
LEND	0183	0185																
NOVR	0139	0142																
NUND	0138	0141																
PACK	0071																	
SCAT	0002																	
SQRT	0109	0110	0125	0126	0127													
XSUM	0120	0122	0124	0126	0127													
XYVS	0012	0029	0076	0076	0122	0124	0124	0125										
YSUM	0121	0124																
BLANK	0019	0177																
IPLOT	0002	0024	0025	0026	0027	0028	0029	0030	0031	0032	0033	0036	0039	0039	0040	0040	0041	0042	0043
	0043	0045	0045	0057	0057	0059	0059	0071	0072	0072	0073	0073	0074	0074	0075	0075	0076	0076	0077
	0077	0080	0083	0090	0096	0096	0097	0102	0102	0102	0102	0102	0103	0105	0105	0106	0106	0107	0107
	0107	0107	0107	0107	0108	0108	0108	0108	0108	0108	0108	0111	0111	0112	0120	0120	0120	0120	0121	0121
	0121	0121	0122	0122	0122	0122	0123	0123	0123	0123	0123	0124	0124	0124	0124	0124	0124	0124	0124	0125
	0125	0125	0125	0126	0138	0139	0140	0140	0141	0142	0158	0158	0159	0159	0165	0175				
IXLAB	0091	0186																		
IYLAB	0090	0091	0101																	
LSTRT	0182	0183	0185																	
CPTIME	0085																			
GTDATE	0004	0088	0088																	
GTRSEG	0004	0088																		
KLASEG	0005	0088																		
KLDATE	0005	0087	0088																	
UNPACK	0165																			

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

LABEL	DEFINED	REFERENCES	
1	0024	0023	
2	0038	0023	
3	0079	0023	
4	0036	0034	0035
5	0097	0094	
6	0149	0148	
7	0151	0150	
8	0196	0160	
9	0173	0164	
10	0172	0167	
11	0162	0161	

***** F O R T R A N C R O S S R E F E R E N C E L I S T I N G *****

LABEL	DEFINED	REFERENCES
20	0146	0099 0194
21	0093	0153
93	0157	0195
94	0154	0152
95	0147	0098
97	0084	0093
100	0092	0101
123	0087	0086
124	0089	0088
500	0054	0051
501	0057	0054
600	0120	0112
750	0144	0143
800	0194	0178 0189
801	0190	0180
802	0177	0176
803	0187	0185
900	0137	0136
992	0197	0102
994	0135	0134
995	0129	0103
996	0134	0119 0128

/ SCAT / SIZE OF PROGRAM 007C96 HEXADECIMAL BYTES

NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
A	SF	R*4	000330	G	SF	R*4	000400	I	SF	I*4	000334	J	SF	I*4	000338
L	SFA	I*4	00033C	X	SF	R*4	000340	Y	SF	R*4	000344	AO	SFA	R*4	000348
AI	SFA	R*4	00034C	UX	SF	R*4	000350	DY	SF	R*4	000354	IB	F	I*4	000358
IC	SF	I*4	00035C	ID	SFA	I*4	0008C0	II	SFA	I*4	000360	IK	SF	I*4	000364
IL	SF	I*4	000368	IP	SF	I*4	00036C	IS	F	I*4	000418	IT	F	I*4	000428
IV	SF	I*4	0004B8	IX	SFA	I*4	000370	IY	SFA	I*4	000374	JJ	F	I*4	000378
KQ	S	I*4	00037C	R2	SF	R*4	000380	S0	SF	R*4	000384	SI	SF	R*4	000388
TT	SFA	R*4	00038C	VT	SFA	R*4	000390	XV	F	R*4	000394	XX	SFA	R*4	000398
YV	F	R*4	00039C	YY	SFA	R*4	0003A0	AAI	SF	R*4	000648	CCC	SFA	R*4	000654
DUM		R*4	NR	HMS	SFA	I*2	0003F8	IBM	SF	I*4	0003A4	JJJ	SFA	I*4	0003AB
KYR	F C	I*4	000004	LPT	SF	I*4	0003AC	SYA	SF	R*4	0003B0	TYR	F C	R*4	000004
XIX	SF	R*4	0003B4	XMN	F C	R*4	000028	XMX	F C	R*4	000000	XOV	SF	R*4	00067C
XUN	SF	R*4	0006A4	XVA	SFA	R*4	0006CC	XVS	SFA	R*4	0006F4	XXX	SF	R*4	000388
YIY	SF	R*4	0003BC	YMN	F C	R*4	000078	YMX	F C	R*4	000050	YOV	SF	R*4	00071C
YUN	SF	R*4	000744	YVA	SF	R*4	00076C	YVS	SFA	R*4	000794	AXLA	F	R*4	00078C
BUFF	S	R*4	0007C8	DATE	SFA	I*4	000890	HEAD	F C	R*4	000000	IJKL	S	I*4	0003C0
IXOV	S	I*4	0003C4	IYOV	S	I*4	0003C8	LEND	SF	I*4	0003CC	NOVR	S	I*4	0003D0
NUND	S	I*4	0003D4	PACK	SF XF		000000	SCAT		R*4	0003D8	SQRT	F XF	R*4	000000
XSUM	SFA	R*4	0003DC	XYVS	SFA	R*4	000898	YSUM	SF	R*4	0003E0	BLANK	F	R*4	0003E4
IPLOT	SFA	I*4	0003E8	IXLAB	SF	I*4	0003EC	IYLAB	SF	I*4	0003F0	LSTRT	SF	I*4	0003F4
CPTIME	SF XF		000000	GTDATE	F C	I*4	000008	GTRSEG	F C	R*4	000000	IBCOM#	F XF	I*4	000000
KLASEG	F C	I*4	000000	KLDATE	F C	I*4	000008	UNPACK	SF XF		000000				

***** COMMON INFORMATION *****

NAME OF COMMON BLOCK * PLOT* SIZE OF BLOCK 0000A0 HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
XMX	R*4	000000	XMN	R*4	000028	YMX	R*4	000050	YMN	R*4	000078

NAME OF COMMON BLOCK * DEFGT* SIZE OF BLOCK 000014 HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
GTRSEG	R*4	000000	TYR	R*4	000004	GTDATE	I*4	000008			

NAME OF COMMON BLOCK * DEFCL* SIZE OF BLOCK 000014 HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
KLASEG	I*4	000000	KYR	I*4	000004	KLDATE	I*4	000008			

NAME OF COMMON BLOCK *HEADR** SIZE OF BLOCK 0005A0 HEXADECIMAL BYTES

VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.	VAR. NAME	TYPE	REL. ADDR.
HEAD	I*4	000000									

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
1	24	006F82	4	36	006F56	2	38	006FA0	500	54	007088
501	57	007098	3	79	007268	21	93	00736A	5	97	0073A6
600	120	007600	995	129	00780A	996	134	007822	750	144	007940
20	146	007972	6	149	007974	7	151	007990	11	162	007A3E
10	172	007AAA	9	173	007AA8	802	177	007AFC	803	187	007B8E
801	190	007B88	800	194	007BDA	8	196	007C28			

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100001	2	006F80	100003	35	006F54	100004	37	006F86	100005	37	006F90
100006	44	007000	100007	45	00702A	100008	46	007034	100009	47	007050
100010	53	007084	100011	56	007094	100012	58	0070A0	100013	59	0070BC
100014	60	0070C4	100015	61	0070E0	100016	62	0070FA	100017	63	0070FA
100018	64	007104	100019	65	007114	100020	66	007130	100021	67	007140
100022	68	00715C	100023	69	00716C	100024	81	00727A	100025	82	00728A
100034	95	00736C	100035	98	0073CE	100038	101	007400	100041	105	0074AC
100042	114	0075E4	200001	126	0077C6	100043	145	007962	100044	150	00798C
100045	152	0073AA	100048	155	0079F0	100049	161	007A3C	100050	163	007A56
100051	165	007A5C	100052	169	007A7A	100053	171	007A94	100054	174	007AB6
100055	178	007B0E	100056	180	007B3A	100057	182	007B44	100058	186	007B68
100059	188	007BAA	100062	197	007C3A						

FORMAT STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
97	84	000028	123	87	000041	124	89	00006A	100	92	0000B3
994	135	0000C9	900	137	00010F	95	147	000173	94	156	000170
93	157	000185	992	197	000194						

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

STATISTICS SOURCE STATEMENTS = 199, PROGRAM SIZE = 31894, SUBPROGRAM NAME = SCAT

STATISTICS NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ****** 244K BYTES OF CORE NOT USED

```
REQUESTED OPTIONS: NOTFRM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
   ISN 0002          SUBROUTINE UNPACK(
                C                           RETREIVE INFORMATION FROM A BYTE
                    I IPET,
                C
                C                    WORD RETURN FOR CONTAINING BYTE
                    ( LOTBLE,
                C                    PLOT TABLES USED TO COUNT OCCURANCE OF LABLES
                    ( IX,IY,
                C                    X,Y AXIS FOR PLOT.
                    (IPLOT)
                C                    NUMBER OF PLOT WHICH IS TO BE UPDATE
                C   ----------------------------------------------------------------
                C
                C
                C  HISTORY
                C  -------
                C
                C   MARY TOMPKINS      LEMSCO        12/23/80        REQUIREMENTS
                C   MARY TOMPKINS      LEMSCO        12/23/80        ALGORITHM DESIGN
                C   DONALD CHENG       LEMSCO        12/24/80        ALGORITHM CODING
                C
                C
                C  METHOD
                C  -------
                C
                C   UNPACK THE LOTABLE WORD INTO APPROPRIATE BYTE
                C
                C
                C  MACHINE-DEPENDENT CODE
                C  ----------------------
                C
                C   ASSUMES IBM 32 BIT, 4BYTE WORD SIZE
                C
                C
                C  EXTERAL REFERENCES
                C  ------------------
                C
                C   NONE.
                C
                C
                C  EXCEPTIONS
                C  ----------
                C
                C   THE MAXIMUN COUNT OF LABLES IS 255.
                C
                C
                C  GLOBAL DECLARATIONS
                C  -------------------
                C
                C   NONE.
                C
                C
                C  LOCAL DECLARATIONS
                C  ------------------
                C
   ISN 0003         INTEGER LOTBLF(13,50,10)
                C                              ARGUMENT.
   ISN 0004         LOGICAL*1 ITEWRD(4)
                C                              BYTE ADRESSABLE WORD.
   ISN 0005         INTEGER IWORD
                C                              ALLOWS EQUIVALENCING OF WORD TO BYTE
                C                              ADDRESSABLE WORD.
                C
   ISN 0006         INTEGER ICRWD
                C                              WORD CONTAINING BYTE TO BE UPDATE
   ISN 0007         EQUIVALENCE(IWORD,ITEWRD(1))
                C
                C
                C  PROCEDURE
                C  ---------
                C
   ISN 0008         ICRWD=(IX-1)/4+1
                C
```

ORIGINAL PAGE IS OF POOR QUALITY

```
   ISN 0009            IWORD=LOTBLE(ICRNWD,IY,IPLOT)                          UNP00740
                     C                                                        UNP00750
                     C                                                        UNP00760
                     C  IBYTE RETURN FOR 1,2,3,0 AS IX=1,2,3,4                UNP00770
                     C  MAKE IBYTE=4 INSTEAD OF 0                             UNP00780
                     C                                                        UNP00790
   ISN 0010            IBYTE=MOD(IX,4)                                        UNP00800
   ISN 0011            IF(IBYTE.EQ.0) IBYTE=4                                 UNP00810
                     C                                                        UNP00820
                     C  RETURN FOR THE WORD                                   UNP00830
                     C                                                        UNP00840
   ISN 0013            IRET=ITEWRD(IBYTE)                                     UNP00850
                     C                                                        UNP00860
   ISN 0014            RETURN                                                 UNP00870
   ISN 0015            END                                                    UNP00880
```

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL	STATEMENT NUMBERS		
IX	0002	0008	0010	
IY	0002	0009		
MOD	0010			
IRET	0002	0013		
IBYTE	0010	0011	0011	0013
IPLOT	0002	0009		
IWORD	0005	0007	0009	
ICRNWD	0006	0008	0009	
ITEWRD	0004	0007	0013	
LOTBLE	0002	0003	0009	
UNPACK	0002			

/ UNPACK / SIZE OF PROGRAM 0001BA HEXADECIMAL BYTES

NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
IX	FA	I*4	000094	IY	F	I*4	000098	IRET	S	I*4	00009C	IBYTE	SF	I*4	0000A0
IPLOT	F	I*4	0000A4	IWORD	S E	I*4	0000B8	ICRNWD	SF	I*4	0000AB	ITEWRD	F E	L*1	000BB
LOTBLE	F XR	I*4	000000	UNPACK		R*4	0000AC								

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100001	2	0000C8	100002	12	000122	100003	13	000126			

```
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*     SOURCE STATEMENTS =      14, PROGRAM SIZE =      442, SUBPROGRAM NAME =UNPACK
*STATISTICS*  NO DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                          288K BYTES OF CORE NOT USED
```

APPENDIX C
JOB CONTROL SOFTWARE

```
    &CONTROL OFF
*
*
* PARPLT EXEC
* -----------
*
*
* HISTORY
* -------
*
*    MARY TOMPKINS      LEMSCO      02/09/81      ORIGINAL CODE
*
*
* PURPOSE
* -------
*
*
*    THIS EXEC EXCUTES THE PARPLT PROGRAM AND REISSUES FILEDEFS FOR
*    GROUND TRUTH AND CLASS BY FILES EXECUTING THE FILRCALL PROGRAM.
*
*    ARGUMENTS TO THIS EXEC ARE:
*    SYMBOL FILE NAME    SYMBOL FILE TYPE
*
*    FILE DEFINITION DESCRIPTION FOR ALL FILES USED IN BADHWAR
*    PROGRAMS AND EXECS ARE AS FOLLOWS:
*         UNIT              DESCRIPTION
*          2-4              BADHWAR SYSTEM
*          5                TERMINAL - WRITE
*          6                PRINTER FILE STORED IN FILE OUT LISTING
*          7-8              BADHWAR SYSTEM
*          9                GROUND TRUTH (MUST BE DEFINED PREVIOUSLY)
*         10                CLASS FILE  (MUST BE DEFINED PREVIOUSLY)
*         11                BADHWAR SYSTEM
*         12-18             BADHWAR SYSTEM
*         19                SYMBOL FILE STORED IN &1 &2 A
*         20                DOCUMENTATION -- STORED IN PPLT FILE D
*         21                USER DEFINED FILE STORED IN &3 CC A
*         22                RECALL FILE FOR GROUND TRUTH FILE
*         23                RECALL FILE FOR CLASS FILE
*         24-28             BADHWAR SYSTEM
*         30                REREAD UNIT
*
*    NOTE THOSE FILES USED BY BADHWAR SYSTEM CAN BE USED IN THIS
*    PROGRAM. THIS IS JUST A WARNING THAT ONE SHOULD BE CAREFUL BEFORE
*    DOING SO IF HE INTENDS TO MAKE A BADHWAR SYSTEM RUN.
*
*
* EXCEPTION
* ---------
*
*    THE FOLLOWING ERRORS CAUSE PROGRAM TERMINATION:
*
*        1. NO TEMPORARY DISK ASSIGNED.
*        2. INSUFFICIENT PARAMETERS INPUT TO PROGRAM
*        3. GROUND TRUTH/CLASS FILE NOT DEFINED
*
*
* PROCEDURE
* ---------
*
*
* ASSIGN PRINTER. SPECIFY LIBRARIES
*
     &IERR = 0
     &SPACE 1
     &TYPE PARPLT &1 &2 &3
     TAG DEV PRINTER HOUSTON
     SPOOL PRINTER CONT NOH TO RSCS
*
*
* CHECK TO SEE IF TEMPORARY DISK IS ASSIGNED
*
     CP QUERY VIRTUAL 192
     &IF &RETCODE EQ 0 &GOTO -TRUE
     &TYPE GROUND TRUTH / CLASS FILE NOT YET DEFINED.
     &IERR = 1
-TRUE
*
*
```

```
* CHECK FOR ACCEPTABLE PARAMETERS
*
     &IF &INDEX EQ 3 &GOTO -CONT
     &TYPE TOO MANY-TOO FEW INPUTS
     &IERR = 1
-CONT &IF &IERR EQ 1 &EXIT 1
*
**
* ISSUE FILEDEFS
*
     FILEDEF FT02F001 DISK GTCLINE EXEC D (LRECL 80 BLKSIZE 80 PERM
     FILEDEF 5 TERMINAL (PERM
     FILEDEF FT06F001 DISK OUT LISTING D(PERM
     FILEDEF FT19F001 DISK &1 &2 A(PERM LRECL 80 BLKSIZE 80
     FILEDEF FT20F001 DISK PPLT FILE D (LRECL 132 BLKSIZE 132 PERM
     FILEDEF FT21F001 DISK &3 CC A(PERM LRECL 80 BLKSIZE 80
     FILEDEF FT22F001 DISK FLGTRU FILE D (LRECL 80 BLKSIZE 80 PERM
     FILEDEF FT23F001 DISK FLCLAS FILE D (LRECL 80 BLKSIZE 80 PERM
*
**
* RECALL INFO FOR FILDEF
*
     LOAD FILRCALL (CLEAR NOMAP START
*
**
* LOAD EXEC TO FILEDEF GT AND CLASS FILES
*
     EXEC GTCLINE D
     &READ VARS &IERROR
     &IF &IERROR EQ 1 &EXIT 2
*
**
* LOAD MAIN PROGRAM
*
     LOAD PARPLT (NOMAP CLEAR START
*
**
* CLOSE FILE PRINT FILES
*
     PRINT PPLT FILE D
     PRINT OUT LISTING D
     SPOOL PRINTER CLOSE
     &EXIT
&END
```

```
    &CONTROL OFF
*
* DEFCLAS
* -------
*
* HISTORY
* -------
*
*    M A TOMPKINS     LEMSCO     02/04/81     ORIGINAL CODE
*
* PURPOSE
* -------
*
*    THIS EXEC IS USED TO DEFINE CLASSIFICATION/CLUSTER FILES.
*
*    FILENAME FILETYPE FILEMODE OF CLASS FILE OR WRITTEN
*    ON A RECALL FILE (UNIT 23) BY FORTRAN ROUTINE FILWRT.
*    ARGUMENTS TO THE EXEC ARE AS FOLLOWS:
*
*    FOR SEGMENT ON DATA FILE:
*       FILENAME FILETYPE FILEMODE
*    FOR SEGMENT ON TAPE:
*       TAPE# FILE# TAPEDENSITY
*
*    FILE DEFINITION DESCRIPTION FOR ALL FILES USED IN THESE PROGRAMS
*    AND EXECS ARE AS FOLLOWS:
*          UNIT                  DESCRIPTION
*           2                    GTRUINFO
*           3                    TERMINAL: READ
*           4                    LARS GTRUINF
*           5                    TERMINAL: WRITE LARS ERROR MSG ROUTINE
*           6                    BADHWAR SYSTEM
*           7                    BADHWAR SYSTEM
*           8                    BADHWAR SYSTEM
*           9                    BADHWAR SYSTEM
*          10                    CLASSIFICATION/CLUSTER FILE
*          11 - 19               BADHWAR SYSTEM
*          21                    BADHWAR SYSTEM
*          22                    BADHWAR SYSTEM
*          23                    RECALL FILE FOR CLASS FILE
*          24-28                 BADHWAR SYSTEM
*          30                    REREAD UNIT
*
*    NOTE: THOSE FILES USED BY BADHWAR SYSTEM CAN BE USED IN THIS
*    PROGRAM THIS IS JUST A WARNING THAT ONE SHOULD BE CAREFUL BEFORE
*    DOING SO.
*
* EXCEPTION
* ---------
*
*    THE FOLLOWING ERRORS CAUSE PROGRAM TERMINATION:
*
*       1. NO TEMPORARY DISK AVAILABLE.
*       2. INSUFFICIENT PARAMETERS INPUT TO PROGRAM
*       3. ERROR IN ACCESSING LARS DATA BASE
*
* PROCEDURE
* ---------
*
* ASSIGN A TEMP DISK, SPECIFY LIBRARIES
*
    &SPACE 3
    &TYPE DEFCLAS &1 &2 &3
    GLOBAL TXTLIB CMSLIB FORTMOD2
    CP QUERY VIRTUAL 192
    &IF &RETCODE NE 0 GETDISK TEMP  2M CLEAR
    &IF &RETCODE NE 0 &TYPE NO TEMP DISK ACCESSED.
    &IF &RETCODE NE 0 &EXIT 1
*
*
*    STACK UNIT NUMBER AND FILEDEF RECALL UNIT
```

```
*
     &STACK 10
     FILEDEF 3 TERM(PERM
     FILEDEF FT23F001 DISK DLCLAS FILE D(LRECL 80 BLKSIZE 80 PERM
*
*
* CHECK FOR ACCEPTABLE PARAMETER COUNT AND DETERMINE INPUT
* OPTION
*
     &IF &INDEX EQ 3 &GOTO -TRUE
     &TYPE TOO MANY-TOO FEW INPUTS
     &EXIT 2
*
*
-TRUE
     &IF &3 EQ 1600 &GOTO -TAPE
     &IF &3 EQ  800 &GOTO -TAPE
     &TEST = &DATATYPE &3
     &IF &TEST EQ NUM &TYPE INPUTS NOT CORRECT
     &IF &TEST EQ NUM &EXIT 3
*
*
* DATA IS ON DISK
*
 FILEDEF FT10F001 DISK &1 &2 &3( LRECL 3168 BLOCK 3168 PERM RECFM U
     &STACK &1
     &STACK &2
     &STACK &3
     LOAD FILWRT (CLEAR NOMAP START
     &EXIT 4
*
*
* DATA IS ON TAPE
*
-TAPE &NAME = &CONCAT &1 &2
     TAPMOUNT &1 TAP1 RO &3
     TAPE REW (TAP1
     &SK = &2 - 1
     &IF &SK EQ 0 &SKIP 1
     TAPE FSF &SK
     FILEDEF INMOVE TAP1(LRECL 3168 BLOCK 3168 RECFM U PERM DEN &3
     FILEDEF OUTMOVE DISK &NAME CLA D(LRECL 3168 BLOCK 3168 RECFM U PERN
     MOVEFILE
     DETACH 18)
     FILEDEF FT10F001 DISK &NAME CLA D(LRECL 3168 BLOCK 3168 RECFM U PERM
     &STACK &NAME
     &STACK CLA
     &STACK D
     LOAD FILWRT(CLEAR NOMAP START
     &EXIT
&END
```

```
     &CONTROL OFF
*
* DEFGTRU EXEC
* ------------
*
* HISTORY
* -------
*
*    M A TOMPKINS      LEMSCO      02/04/81      ORIGINAL CODE
*
* PURPOSE
* -------
*
*    THIS EXEC EXECUTES A FORTRAN PROGRAN (GTRUINF) WHICH ACCESSES
*    THE LARS RIKE DATA BASE FOR INFO ON REQUESTED GROUND TRUTH TAPES
*    GTRUINF WRITES AN EXEC (GTRUINFO) WHICH TRANSMITS TO THIS EXEC
*    THE TAPE# FILE# OF TAPE THAT CONTAINS THE REQUESTED SEGMENT.
*    IN ADDITION THE FILENAME,FILETYPE,FILEMODE, AND UNIT 9 ARE
*    ARE PASSED TO PROGRAM FILWRT WHICH WRITES THIS INFO TO A FILE
*    DEFINED TO UNIT 22.
*
*    ARGUMENTS TO THE EXEC ARE AS FOLLOWS:
*
*    FOR SEGMENT ON DATA FILE:
*       FILENAME FILETYPE FILEMODE
*    FOR SEGMENT ON TAPE:
*       TAPE# FILE# TAPEDENSITY
*    FOR SEGMENT AT LARS:
*       SEGMENT# YEAR  (YEAR IS THE LAST 2 DIGITS OF THE YEAR OF SEGMENT)
*
*    FILE DEFINITION DESCRIPTION FOR ALL FILES USED IN THESE PROGRAMS
*    AND EXECS ARE AS FOLLOWS:
*          UNIT                DESCRIPTION
*           2                  GTRUINFO
*           3                  TERMINAL: READ
*           4                  LARS GTRUINF
*           5                  TERMINAL: WRITE LARS ERROR MSG ROUTINE
*           6                  BADHWAR SYSTEM
*           7                  BADHWAR SYSTEM
*           8                  BADHWAR SYSTEM
*           9                  GROUND TRUTH FILES
*          10                  BADHWAR SYSTEM
*          11 - 19             BADHWAR SYSTEM
*          21                  BADHWAR SYSTEM
*          22                  GROUND TRUTH FILE INFO
*          23                  BADHWAR SYSTEM
*          24-28               BADHWAR SYSTEM
*          30                  REREAD UNIT
*
*    NOTE: THOSE FILES USED BY BADHWAR SYSTEM CAN BE USED IN THIS
*    PROGRAM THIS IS JUST A WARNING THAT ONE SHOULD BE CAREFUL BEFORE
*    DOING SO.
*
* EXCEPTION
* ---------
*
*    THE FOLLOWING ERRORS CAUSE PROGRAM TERMINATION:
*
*       1. NO TEMPORARY DISK AVAILABLE.
*       2. INSUFFICIENT PARAMETERS INPUT TO PROGRAM
*       3. 3RD INPUT NOT AS EXPECTED
*       4. ERROR IN ACCESSING LARS DATA BASE
*
* PROCEDURE
* ---------
*
* ASSIGN A TEMP DISK, SPECIFY LIBRARIES
*
     &SPACE 3
     &TYPE DEFGTRU &1 &2 &3
     GLOBAL TXTLIB CMSLIB FORTMOD2
     CP QUERY VIRTUAL 192
```

```
     &IF &RETCOD| NE 0 GETDISK TEMP  2M CLEAR
     &IF &RETCOD| NE 0 &TYPE NO TEMP DISK ACCESSED.
     &IF &RETCOD| NE 0 &EXIT 1
*
*
* ISSUE FILEDEFS FOR FILWRT FORTRAN PROGRAM
* CHECK FOR ACCEPTABLE PARAMETER COUNT AND DETERMINE INPUT
* OPTION
*
     FILEDEF 3 TERM(PERM
     FILEDEF FT22F001 DISK FLGTRU FILE D(LRECL 80 BLKSIZE 80 PERM
     &IF &INDEX LT 2 &GOTO -NO
     &IF &INDEX LE 3 &GOTO -TRUE
-NO  &TYPE TOO MANY-TOO FEW INPUTS
     &EXIT 2
*
*
-TRUE &UNIT = 09
     &IF &INDEX EQ 2 &GOTO -LARS
     &IF &3 EQ 1600 &GOTO -TAPE
     &IF &3 EQ  800 &GOTO -TAPE
     &TEST = &DATATYPE &3
     &IF &TEST EQ NUM &TYPE INPUTS NOT CORRECT
     &IF &TEST EQ NUM &EXIT 3
*
*
* DATA IS ON DISK
*
 FILEDEF FT09F001 DISK &1 &2 &3( LRECL 3060 BLOCK 3060 PERM RECFM U
     &STACK &UNIT
     &STACK &1
     &STACK &2
     &STACK &3
     LOAD FILWRT (CLEAR·NOMAP START
     &EXIT 4
*
*
* DATA IS ON TAPE
*
-TAPE &NAME = &CONCAT &1 &2
     TAPMOUNT &1 TAP1 RO &3
     TAPE REW (TAP1
     &SK = &2 - 1
     &IF &SK EQ 0 &SKIP 1
     TAPE FSF &SK
     FILEDEF INMOVE TAP1(LRECL 3060 BLOCK 3060 RECFM U PERM DEN &3
     FILEDEF OUTMOVE DISK &NAME GTO D(LRECL 3060 BLOCK 3060 RECFM U PERM
     &STACK &UNIT
     &STACK &NAME
     &STACK GTO
     &STACK D
     LOAD FILWRT (CLEAR NOMAP START
     MOVEFILE
     DETACH 181
     FILEDEF FT09F001 DISK &NAME GTO D(LRECL 3060 BLOCK 3060 RECFM U PERM
     &EXIT 5
*
*
* GET TAPE AND FILE NUMBER FROM LARS
*
-LARS
*
*
* CHECK FOR PROPER ARGUMENTS
*
     &TEST = &DATATYPE &1
     &IF &TEST EQ CHAR &TYPE INPUTS NOT CORRECT FOR LARS DATA BASE
     &IF &TEST EQ CHAR &EXIT 6
*
     &TEST = &DATATYPE &2
     &IF &TEST EQ CHAR &TYPE INPUTS NOT CORRECT FOR LARS DATA BASE
     &IF &TEST EQ CHAR &EXIT 7
*
*
* GET LARS DISK WHICH CONTAINS RTAE DATA BASE
*
     GETDISK JSCDISK 19A E
*
*
```

```
* FILEDEF TERMINAL AND EXEC FILE WRITTEN DURNING RUN.
*
     FILEDEF 3 TERM(PERM
     FILEDEF 5 TERM(PERM
     FILEDEF FT02F001 DISK GROINFO EXEC D( LRECL 80 BLKSIZE 80 PERM
*
*
* IF LENGTH OF &1 (SEGMENT NUMBER) < 4 CONCATENATE 0
*
-LOOP &SEGLNG = &LENGTH &1
     &IF &SEGLNG EQ 4 &SKIP 2
     &1 = &CONCAT 0 &1
     &GOTO -LOOP
*
*
* STACK INPUTS TO ALLOW GTRUINF TO ACCESS LARS RT&E DATA BASE.
* GTRUINF WILL WRITE GTRUINFO EXEC TO ALLOW THE PASS THROUGH OF
* TAPE# FILE#.
*
     &STACK &1
     &STACK &2
*
     LOAD GTRUINF GTINFOHX RTEERRHX (CLEAR NOMAP START
*
*
*
* LOAD EXEC WRITTEN BY PROGRAM.
*
     EXEC GROINFO D
     REL E(DET
     &READ VARS &TAPE
     &READ VARS &FILE
     &1 = &TAPE
     &2 = &FILE
     &3 = 800
     &IF &1 NE 0 &GOTO -TAPE
     &EXIT
&END
```

```
     &CONTROL OFF
*
* END EXEC
* --------
*
*
* PURPOSE
* -------
*    THIS EXEC WILL CLOSE CONSOLE FILE AND PRINT THE FILE
*
*
* PROCEDURE
*
     SPOOL CONSOLE STOP CLOSE
     &EXIT
&END
```

```
&CONTROL OFF
*
* START EXEC
* ----------
*
* PURPOSE
* -------
*    THIS EXEC WILL ALLOW THE USER TO SPOOL ALL RESPUNCES TO THE
*    CONSOLE.THIS IS TO BE USED WITH END EXEC WHICH WILL PRINT THE FILE.
*
* PROCEDURE
* ---------
*
     TAG DEV CONS HOUSTON
     SPOOL CONS START NOHOLD TO RSCS
     &EXIT
&END
```

REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM I8M FLAG(I)

```
                      C     PROGRAM FILRCALL                                                        FIL00010
                      C                                                                             FIL00020
                      C                     READ FILE INFO FOR GT FILE CLASS FILE.                  FIL00030
                      C     ---------------------------------------------------                     FIL00040
                      C                                                                             FIL00050
                      C                                                                             FIL00060
                      C     HISTORY                                                                 FIL00070
                      C     -------                                                                 FIL00080
                      C                                                                             FIL00090
                      C         MARY TOMPKINS     LEMSCO     03/12/81     ORIGINAL CODE           FIL00100
                      C                                                                             FIL00110
                      C                                                                             FIL00120
                      C     METHOD                                                                  FIL00130
                      C     ------                                                                  FIL00140
                      C                                                                             FIL00150
                      C         READ FROM RECALL FILE 22 FOR GT FILE INFO AND FILE 23 FOR           FIL00160
                      C         CLASSIFICATION FILE INFO. IF BOTH OR EITHER ARE EMPTY WRITE         FIL00170
                      C         EXEC WITH ERROR FLAG SET TO 1.IF BOTH FILES ARE AVAILABLE           FIL00180
                      C         WRITE EXEC TO FILEDEF GT AND CLASS FILE. THIS IS NECESSARY          FIL00190
                      C         ONLY BECAUSE OF THE POSSIBILITY OF THE FILE DEFNITION BEING         FIL00200
                      C         LOST ON A SYSTEM ERROR ON ONE OF THE PROGRAM EXECUTIONS.            FIL00210
                      C                                                                             FIL00220
                      C                                                                             FIL00230
                      C     EXTERNAL REFERENCES                                                     FIL00240
                      C     -------------------                                                     FIL00250
                      C                                                                             FIL00260
                      C         NONE.                                                               FIL00270
                      C                                                                             FIL00280
                      C                                                                             FIL00290
                      C     EXCEPTIONS                                                              FIL00300
                      C     ----------                                                              FIL00310
                      C                                                                             FIL00320
                      C         1.IF RECALL FILE IS EMPTY ISSUE DIAGNOSTIC MSG                      FIL00330
                      C           AND WRITE ERROR EXEC.                                             FIL00340
                      C                                                                             FIL00350
                      C                                                                             FIL00360
                      C     LOCAL DECLARATIONS                                                      FIL00370
                      C     ------------------                                                      FIL00380
                      C                                                                             FIL00390
ISN 0002              C         INTEGER NAMEGT(2)                                                   FIL00400
                      C                           GROUND TRUTH FILE NAME                            FIL00410
ISN 0003              C         INTEGER NAMGTY(2)                                                   FIL00420
                      C                           GROUND TRUTH FILE TYPE                            FIL00430
ISN 0004              C         INTEGER MODEGT                                                      FIL00440
                      C                           GROUND TRUTH FILE MODE                            FIL00450
ISN 0005              C         INTEGER IUNIT                                                       FIL00460
                      C                           UNIT FOR GROUND TRUTH -- 9                        FIL00470
ISN 0006              C         INTEGER NAMFCL(2)                                                   FIL00480
                      C                           CLASS FILE NAME                                   FIL00490
ISN 0007              C         INTEGER NAMCTY(2)                                                   FIL00500
                      C                           CLASS FILE TYPE                                   FIL00510
ISN 0008              C         INTEGER MODECL                                                      FIL00520
                      C                           CLASS FILE MODE                                   FIL00530
ISN 0009              C         INTEGER JUNIT                                                       FIL00540
                      C                           UNIT FOR GROUND TRUTH -- 10                       FIL00550
                      C                                                                             FIL00560
                      C                                                                             FIL00570
                      C     PROCEDURE                                                               FIL00580
                      C     ---------                                                               FIL00590
                      C                                                                             FIL00600
                      C                                                                             FIL00610
                      C     START WRITING RECALL EXEC                                               FIL00620
                      C                                                                             FIL00630
ISN 0010              C         WRITE(2,20)                                                         FIL00640
ISN 0011              C      20 FORMAT(' &CONTROL OFF')                                             FIL00650
                      C                                                                             FIL00660
                      C                                                                             FIL00670
                      C     READ FROM RECALL FILE FOR GROUND TRUTH DATA                             FIL00680
                      C                                                                             FIL00690
ISN 0012              C         IUNIT = 0                                                           FIL00700
ISN 0013              C         READ(22,100,END=200)IUNIT,(NAMEGT(I),I = 1,2),                      FIL00710
                      C        & (NAMGTY(I),I = 1,2),MODEGT                                         FIL00720
ISN 0014              C      100 FORMAT(I2,2A4,2A4,A1)                                              FIL00730
```

```
   ISN 0015        200 IF(IUNIT.NE.0)GO TO 400                                    FIL00740
   ISN 0017            WRITE(5,210)                                               FIL00750
   ISN 0018        210 FORMAT(' GROUND TRUTH FILE IS UNDEFINED')                  FIL00760
   ISN 0019            WRITE(2,220)                                               FIL00770
   ISN 0020        220 FORMAT(' &STACK 1')                                        FIL00780
   ISN 0021            GO TO 400                                                  FIL00790
                   C                                                              FIL00800
                   C                                                              FIL00810
                   C READ FROM RECALL FILE FOR CLASSIFICATION DATA                FIL00820
                   C                                                              FIL00830
   ISN 0022        400 JUNIT = 0                                                  FIL00840
   ISN 0023            READ(23,100,END=405)JUNIT,(NAMECL(I),I = 1,2),             FIL00850
                      & (NAMCTY(I),I = 1,2),MODECL                                FIL00860
   ISN 0024        405 IF(JUNIT.EQ.0)WRITE(5,410)                                 FIL00870
   ISN 0026        410 FORMAT(' CLASSIFICATION FILE IS UNDEFINED')                FIL00880
   ISN 0027            IF(JUNIT.EQ.0.AND.IUNIT.NE.0)WRITE(2,220)                  FIL00890
   ISN 0029            IF(JUNIT.EQ.0.OR.IUNIT.EQ.0)WRITE(2,420)                   FIL00900
   ISN 0031        420 FORMAT(' &EXIT')                                           FIL00910
   ISN 0032            IF(JUNIT.EQ.0.OR.IUNIT.EQ.0)GO TO 900                      FIL00920
                   C                                                              FIL00930
                   C                                                              FIL00940
                   C WRITE EXEC                                                   FIL00950
                   C                                                              FIL00960
   ISN 0034            WRITE(2,430)                                               FIL00970
   ISN 0035        430 FORMAT(' &STACK 0')                                        FIL00980
   ISN 0036            WRITE(2,440)(NAMEGT(I),I = 1,2),(NAMGTY(I),I = 1,2),MODEGT FIL00990
   ISN 0037        440 FORMAT('FI FT09F001 DISK ',2A4,' ',2A4,' ',A1              FIL01000
                      & ,'(LRECL 3060 BLOCK 3060 PERM RECFM U')                   FIL01010
                   C                                                              FIL01020
   ISN 0038            WRITE(2,450)(NAMECL(I),I = 1,2),(NAMCTY(I),I = 1,2),MODECL FIL01030
   ISN 0039        450 FORMAT('FI FT10F001 DISK ',2A4,' ',2A4,' ',A1              FIL01040
                      & ,'(LRECL 3168 BLOCK 3168 PERM RECFM U')                   FIL01050
   ISN 0040            WRITE(2,420)                                               FIL01060
                   C                                                              FIL01070
   ISN 0041        900 STOP                                                       FIL01080
   ISN 0042            END                                                        FIL01090
```


II-C

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL STATEMENT NUMBERS																		
I	0013	0013	0013	0013	0013	0013	0023	0023	0023	0023	0023	0023	0036	0036	0036	0036	0036	0036	0038
	0038	0038	0038	0038	0038														
IUNIT	0005	0012	0013	0015	0027	0029	0032												
JUNIT	0009	0022	0023	0024	0027	0029	0032												
MODECL	0008	0023	0038																
MODEGT	0004	0013	0036																
NAMCTY	0007	0023	0038																
NAMECL	0006	0023	0038																
NAMEGT	0002	0013	0036																
NAMGTY	0003	0013	0036																

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

LABEL	DEFINED	REFERENCES
20	0011	0010
160	0014	0013 0023
200	0015	0013
210	0018	0017
220	0020	0019 0027
400	0022	0015 0021
405	0024	0023
410	0026	0024
420	0031	0029 0040
430	0035	0034
440	0037	0036
450	0039	0038
900	0041	0032

/ MAIN / SIZE OF PROGRAM 00044C HEXADECIMAL BYTES

NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
I	F	I*4	0001AC	IUNIT	S	I*4	000180	JUNIT	S	I*4	000184	IBCOM#	F XF	I*4	000000
MODECL	SF	I*4	000198	MODEGT	SF	I*4	0001BC	NAMCTY	SF	I*4	0001C0	NAMECL	SF	I*4	0001C8
NAMEGT	SF	I*4	0001D0	NAMGTY	SF	I*4	0001D8								

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
200	15	000260	400	22	000298	405	24	0002EC	900	41	000420

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100000	1	0001F4	100001	13	000210	100002	13	000224	100007	17	00026A
100008	23	00029C	100009	23	000280	100014	25	0002F6	100015	27	00030C
200001	28	000316	100016	28	000320	100017	29	000334	200002	30	00033E
100018	30	000348	100019	32	00035C	200003	33	000366	100020	34	000370

FORMAT STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
20	11	000028	100	14	000039	210	18	000047	220	20	00006A
410	26	000077	420	31	00009C	430	35	0000A6	440	37	000083
450	39	0000FD									

*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(00) SIZE(MAX) AUTODBL(NONE)

*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

STATISTICS SOURCE STATEMENTS = 41, PROGRAM SIZE = 1100, SUBPROGRAM NAME = MAIN

STATISTICS NO DIAGNOSTICS GENERATED

****** END OF COMPILATION ****** 292K BYTES OF CORE NOT USED

REQUESTED OPTIONS: NOTERM

OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

```
                      C     PROGRAM FILWRT                                                    FIL00010
                      C                 WRITE INPUTS TO DEFGTRU/DEFCLAS EXECS TO RECALL FILE.FIL00020
                      C     ------------------------------------------------------------------FIL00030
                      C                                                                       FIL00040
                      C                                                                       FIL00050
                      C     HISTORY                                                           FIL00060
                      C     -------                                                           FIL00070
                      C                                                                       FIL00080
                      C         M A TOMPKINS     LEMSCO     03/12/81     ORIGINAL CODE       FIL00090
                      C                                                                       FIL00100
                      C                                                                       FIL00110
                      C     METHOD                                                            FIL00120
                      C     ------                                                            FIL00130
                      C                                                                       FIL00140
                      C         READ FROM INPUT OF DEFGTRU/DEFCLAS STACK AND OUTPUT TO FILE.  FIL00150
                      C                                                                       FIL00160
                      C                                                                       FIL00170
                      C     EXTERNAL REFERENCES                                               FIL00180
                      C     -------------------                                               FIL00190
                      C                                                                       FIL00200
                      C         NONE.                                                         FIL00210
                      C                                                                       FIL00220
                      C                                                                       FIL00230
                      C     EXTERNAL REFERENCES                                               FIL00240
                      C     -------------------                                               FIL00250
                      C                                                                       FIL00260
                      C         NONE.                                                         FIL00270
                      C                                                                       FIL00280
                      C                                                                       FIL00290
                      C     EXCEPTIONS                                                        FIL00300
                      C     ----------                                                        FIL00310
                      C                                                                       FIL00320
                      C         NONE.                                                         FIL00330
                      C                                                                       FIL00340
                      C     LOCAL DECLARATIONS                                                FIL00350
                      C     ------------------                                                FIL00360
                      C                                                                       FIL00370
ISN 0002              C         INTEGER IUNIT                                                 FIL00380
                      C                     LOGICAL UNIT NUMBER OF FILE 9-G TRUTH 10-CLASS  FIL00390
ISN 0003              C         INTEGER NAME(2)                                               FIL00400
                      C                     FILENAME                                        FIL00410
ISN 0004              C         INTEGER NAMETY(2)                                             FIL00420
                      C                     FILENAME TYPE                                   FIL00430
ISN 0005              C         INTEGER MODE                                                  FIL00440
                      C                     FILE MODE                                       FIL00450
                      C                                                                       FIL00460
                      C                                                                       FIL00470
                      C     PROCEDURE                                                         FIL00480
                      C     ---------                                                         FIL00490
                      C                                                                       FIL00500
                      C                                                                       FIL00510
                      C     READ FROM EXEC STACK                                              FIL00520
                      C                                                                       FIL00530
ISN 0006              C         READ(3,100)IUNIT                                              FIL00540
ISN 0007                100 FORMAT(I2)                                                        FIL00550
                      C                                                                       FIL00560
ISN 0008              C         READ(3,150)(NAME(I),I = 1,2)                                  FIL00570
ISN 0009                150 FORMAT(2A4)                                                       FIL00580
                      C                                                                       FIL00590
ISN 0010              C         READ(3,150)(NAMETY(I),I = 1,2)                                FIL00600
                      C                                                                       FIL00610
ISN 0011              C         READ(3,200)MODE                                               FIL00620
ISN 0012                200 FORMAT(A1)                                                        FIL00630
                      C                                                                       FIL00640
                      C                                                                       FIL00650
                      C     WRITE FILES                                                       FIL00660
                      C                                                                       FIL00670
ISN 0013              C         IF(IUNIT.EQ.9)WRITE(22,300)IUNIT,(NAME(I),I = 1,2),           FIL00680
                         &  (NAMETY(I),I = 1,2),MODE                                          FIL00690
ISN 0015                300 FORMAT(I2,2A4,2A4,A1)                                           FIL00700
ISN 0016              C         IF(IUNIT.EQ.10)WRITE(23,300)IUNIT,(NAME(I),I = 1,2),          FIL00710
                         &  (NAMETY(I),I = 1,2),MODE                                          FIL00720
                      C                                                                       FIL00730
```

```
*LEVEL 2.3.0 (JUNE 78)           MAIN           OS/360  FORTRAN H EXTENDED           DATE 81.140/12.33.53       PAGE   2
                    C                                                                      FIL00740
   ISN 0018                 STOP                                                           FIL00750
   ISN 0019                 END                                                            FIL00760

                         *****F O R T R A N    C R O S S    R E F E R E N C E    L I S T I N G*****
SYMBOL    INTERNAL STATEMENT NUMBERS
I         0008  0008  0008  0010  0010  0010  0013  0013  0013  0013  0013  0013  0016  0016  0016  0016  0016  0016
MODE      0005  0011  0013  0016
NAME      0003  0008  0013  0016
IUNIT     0002  0006  0013  0013  0016  0016
NAMETY    0004  0010  0013  0016

                         *****F O R T R A N    C R O S S    R E F E R E N C E    L I S T I N G*****
 LABEL     DEFINED     REFERENCES
  100      0007        0006
  150      0009        0008  0010
  200      0012        0011
  300      0015        0013  0016

                                               /     MAIN /     SIZE OF PROGRAM 000238 HEXADECIMAL BYTES

      NAME    TAG    TYPE  ADD.         NAME   TAG    TYPE  ADD.         NAME   TAG   TYPE  ADD.         NAME   TAG    TYPE  ADD.
         I    F      I*4   0000H0       MODE SF       I*4   0000B4       NAME SF      I*4   0000BC       IUNIT SF      I*4   0000B8
    IBCOM#    F  XF  I*4   000000     NAMETY SF       I*4   0000C4

  COMPILER GENERATED LABELS

    LABEL   ISN   ADDR          LABEL   ISN   ADDR          LABEL   ISN   ADDR          LABEL   ISN   ADDR
   100000     1   0000D4       100005    14   00016C       100010    16   0001B8       100011    17   0001C0
   100016    1A   00020C

  FORMAT STATEMENT LABELS

    LABEL   ISN   ADDR          LABEL   ISN   ADDR          LABEL   ISN   ADDR          LABEL   ISN   ADDR
     100     7   000028          150     9   00002C          200    12   000032          300    15   000036
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*     SOURCE STATEMENTS =      18, PROGRAM SIZE =      568, SUBPROGRAM NAME =  MAIN
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                     296K BYTES OF CORE NOT USED
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
REQUESTED OPTIONS: NOTERM
OPTIONS IN EFFECT: NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
                   SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)

                C     PROGRAM GTRUINF
                C                 GET GROUND TRUTH INFO FROM LARS DATA BASE
                C     ------------------------------------------------------
                C
                C
                C HISTORY
                C -------
                C     M A TOMPKINS    LEMSCO     02/04/81      ORIGINAL CODE
                C
                C
                C METHOD
                C ------
                C
                C     READ NUMBER OF ACOS.,SEGMENT NUMBER AND LAST TWO DIGITS OF SEG
                C     YEAR. ACCESS THE LARS RT&E DATA BASE. IF SUCESSFUL WRITE GTRUINFO
                C     EXEC TO TRANSMIT THE TAPE# FILE# TO THE DEFGTRU EXEC.
                C
                C
                C EXTERNAL REFERENCES
                C -------------------
                C
                C     GTINFO      LARS ROUTINE TO ACQUIRE INFO FROM LARS RT&E DATA BASE
                C     RTEERR      LARS ERROR MESSAGE ROUTINE
                C
                C
                C EXCEPTIONS
                C ----------
                C     IF IERR <> 0 OR 4 WRITE ERROR MESSAGE AND WRITE EXEC
                C     TO TERMINATE PROGRAM.
                C
                C
                C LOCAL DECLARATIONS
                C ------------------
                C
                C
ISN 0002              INTEGER INDEX(9,64)
                C                          INFO ON GROUND TRUTH TAPES
ISN 0003              INTEGER IYR
                C                          LAST 2 DIGITS OF YEAR OF GROUND TRUTH
ISN 0004              INTEGER IERR
                C                          LARS ERROR FLAG
ISN 0005              INTEGER ISEGNO
                C                          SEGMENT NUMBER
ISN 0006              INTEGER IDUMMY(64)
                C                          ARG THAT DOESN'T PERTAIN TO THIS APPLICA-
                C                          TION OF LARS STANDARD  ROUTINE PARAMETERS
                C
                C
                C PROCEDURE
                C ---------
                C
                C
                C READ FROM CONSOLE STACK USER INPUTS. START WRITING EXEC
                C
ISN 0007              WRITE(2,100)
ISN 0008          100 FORMAT(' &CONTROL OFF')
                C
ISN 0009              READ(3,110)ISEGNO
ISN 0010          110 FORMAT(I4)
                C
ISN 0011              READ(3,120)IYR
ISN 0012          120 FORMAT(I2)
                C
                C
                C CALL LARS ROUTINE FOR INFO.
                C
ISN 0013              CALL GTINFO(ISEGNO,IYR,IDUMMY,INDEX,IERR,4,'E')
                C
                C
                C CHECK FOR ERROR
                C
ISN 0014              IF(IERR.EQ.0.OR.IERR.EQ.4)GO TO 160
ISN 0016              CALL RTEERR(IERR,5)
```

ORIGINAL PAGE IS
OF POOR QUALITY

```
   ISN 0017          WRITE(2,150)                                          GTR00740
   ISN 0018      150 FORMAT(' &STACK 0',/,' &EXIT')                        GTR00750
   ISN 0019          GO TO 900                                             GTR00760
   ISN 0020      160 WRITE(2,180)INDEX(1,1),INDEX(2,1)                     GTR00770
   ISN 0021      180 FORMAT(' &STACK',I5,/,' &STACK',I5,/,' &EXIT')        GTR00780
   ISN 0022      900 STOP                                                  GTR00790
   ISN 0023          END                                                   GTR00800
```

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

SYMBOL	INTERNAL STATEMENT NUMBERS				
IYR	0003	0011	0013		
IERR	0004	0013	0014	0014	0016
INDEX	0002	0013	0020	0020	
GTINFO	0013				
IDUMMY	0006	0013			
ISEGNO	0005	0009	0013		
RTEERR	0016				

*****F O R T R A N C R O S S R E F E R E N C E L I S T I N G*****

LABEL	DEFINED	REFERENCES
100	0008	0007
110	0010	0009
120	0012	0011
150	0018	0017
160	0020	0014
180	0021	0020
900	0022	0019

/ MAIN / SIZE OF PROGRAM 000C14 HEXADECIMAL BYTES

NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.	NAME	TAG	TYPE	ADD.
IYP	SFA	I*4	000118	IERR	SFA	I*4	00011C	INDEX	SFA	I*4	000124	GTINFO	SF XF		000000
IBCOMP	F XF	I*4	000000	IDUMMY	SFA	I*4	000A24	ISEGNO	SFA	I*4	000120	RTEERR	SF XF		000000

SOURCE STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
160	20	000AC4	900	22	000BE8						

COMPILER GENERATED LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100000	1	000334	200001	15	000994	100001	16	000BA0			

FORMAT STATEMENT LABELS

LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR	LABEL	ISN	ADDR
100	R	000028	110	10	000039	120	12	00003D	150	18	000041
180	21	000057									

```
*OPTIONS IN EFFECT*NAME(MAIN) OPTIMIZE(1) LINECOUNT(80) SIZE(MAX) AUTODBL(NONE)
*OPTIONS IN EFFECT*SOURCE EBCDIC NOLIST NODECK OBJECT MAP NOFORMAT NOGOSTMT XREF ALC NOANSF NOTERM IBM FLAG(I)
*STATISTICS*      SOURCE STATEMENTS =      22, PROGRAM SIZE =     3092, SUBPROGRAM NAME =  MAIN
*STATISTICS*  NO  DIAGNOSTICS GENERATED
****** END OF COMPILATION ******                                  292K BYTES OF CORE NOT USED
```

APPENDIX D

PROGRAM RUN EXAMPLES

INPUT SUMMARY

```
AI          MARY ANN TOMPKINS
DATE        APRIL 22,1981
SEGMENT     0123
ALPHARNG    0.0 4000.00
BETARNG     0.0 3600.0
TORNG       1000.0 2000.0
CROP        CORN, SOYBEAN, OTHERS
*END
THIS IS AN EXAMPLE OF A PARPLT RUN USING DATA FROM THE IBM VERSION.
```

JOB INITIATED ON 05/19/81 AT 14:43:12 PROGRAM PAPPLT

GROUND TRUTH FILE - 12379365
CLASSIFICATION FILE - 12381064

PIXEL PURITY RANGE - FROM 1 TO 6 SUBPIXELS

GROUND TRUTH TRANSFORMATIONS

```
  1 TO  10 = 1
 11 TO  20 = 2
 21 TO  79 = 3
 80 TO  80 = 6
 81 TO  91 = 3
 92 TO  92 = 1
 93 TO  96 = 3
 97 TO  97 = 2
 98 TO 116 = 3
117 TO 117 = 1
118 TO 121 = 3
122 TO 122 = 2
123 TO 163 = 3
164 TO 164 = 6
165 TO 255 = 3
```

PIXEL PURITY RANGE - FROM 1 TO 6 SUBPIXELS

RANGES FOR ALPHA, BETA, AND TO

```
     0.0    4000.000
     0.0    3600.000
  1000.000  2000.000
```

NUMBER OF CROPS TO BE EVALUATED = 3

1 CORN
2 SOYBEAN
3 OTHERS

HEADER INFORMATION FOR GROUND TRUTH FILE 12379365

COMPUTING SYSTEM ID = PDP-11/45 TAPEOUT PROGRAM

ACQUISITIONS USED =

PFC JOB IDENT. =

HEADER INFORMATION FOR CLASS FILE 12381064

COMPUTING SYSTEM ID = SR MULT TEMPR CLASSIFIER

ACQUISITIONS USED = 78107 78161 78197 78233 78269

PFC JOB IDENT. = SR CLASS FOR

USER SUPPLIED DATA = (SAMPLE,LINE)
TR FLD (58., 3.),(64., 3.),(66., 8.),(60., 8.),

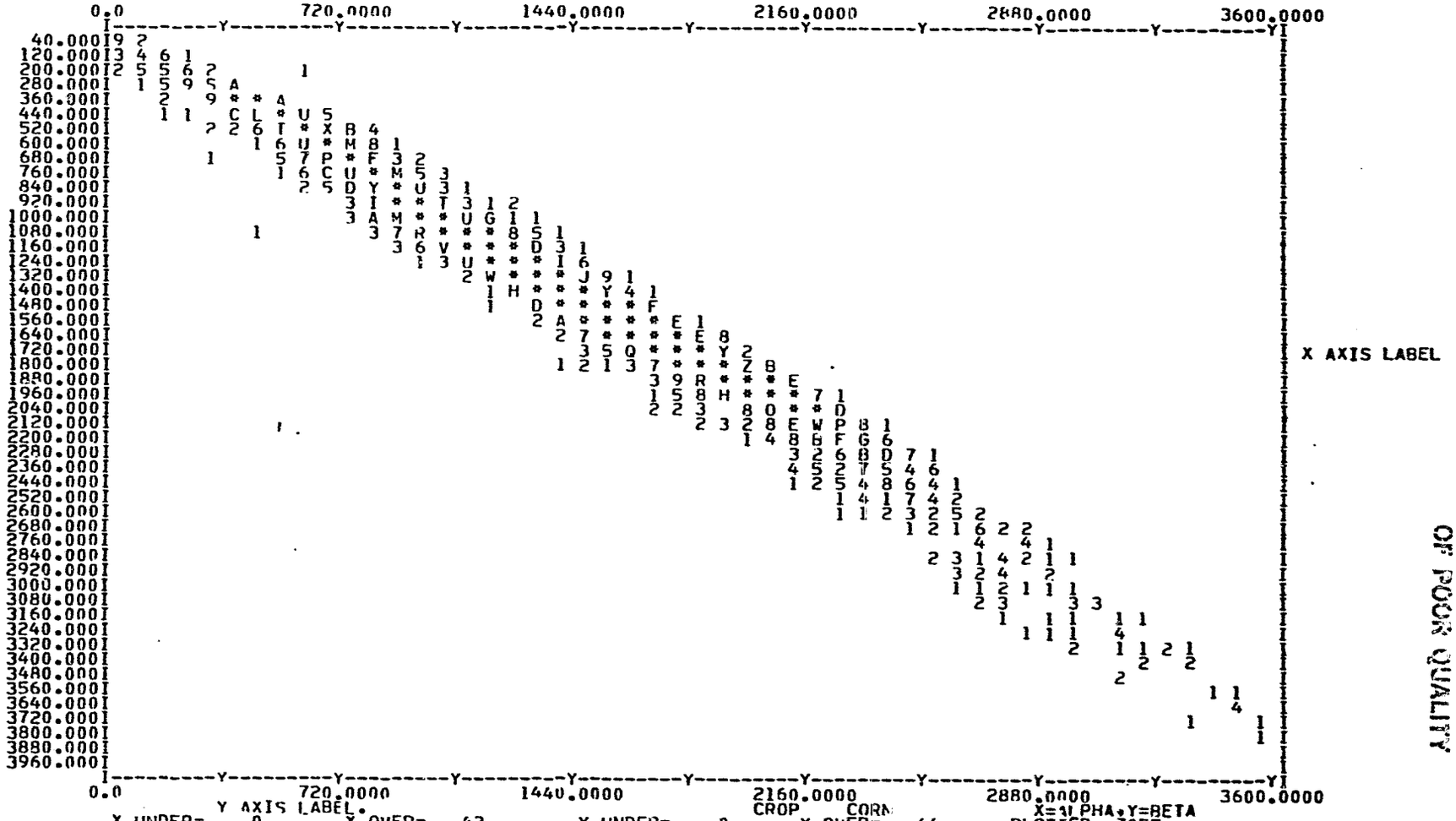
SUMMARY OF NUMBER OF PIXELS IN EACH CATEGORY

```
CORN      7101.0
SOYBEAN   7038.0
OTHERS    6216.0
```

PLOT OUTPUT NO 1

DATE. 05/19/81 TIME. 14:45:12

GROUND TRUTH FILE- 12379365 CLASSIFICATION FILE- 12381064



Y AXIS LABEL.

CROP CORN X=ALPHA,Y=BETA

X UNDER= 0. X OVER= 43. Y UNDER= 0. Y OVER= 44. PLOTTED= 7055.

MEAN X= 1446.3999+/- 489.8298 MEAN Y= 1478.7021+/- 491.4246 MSE= **********

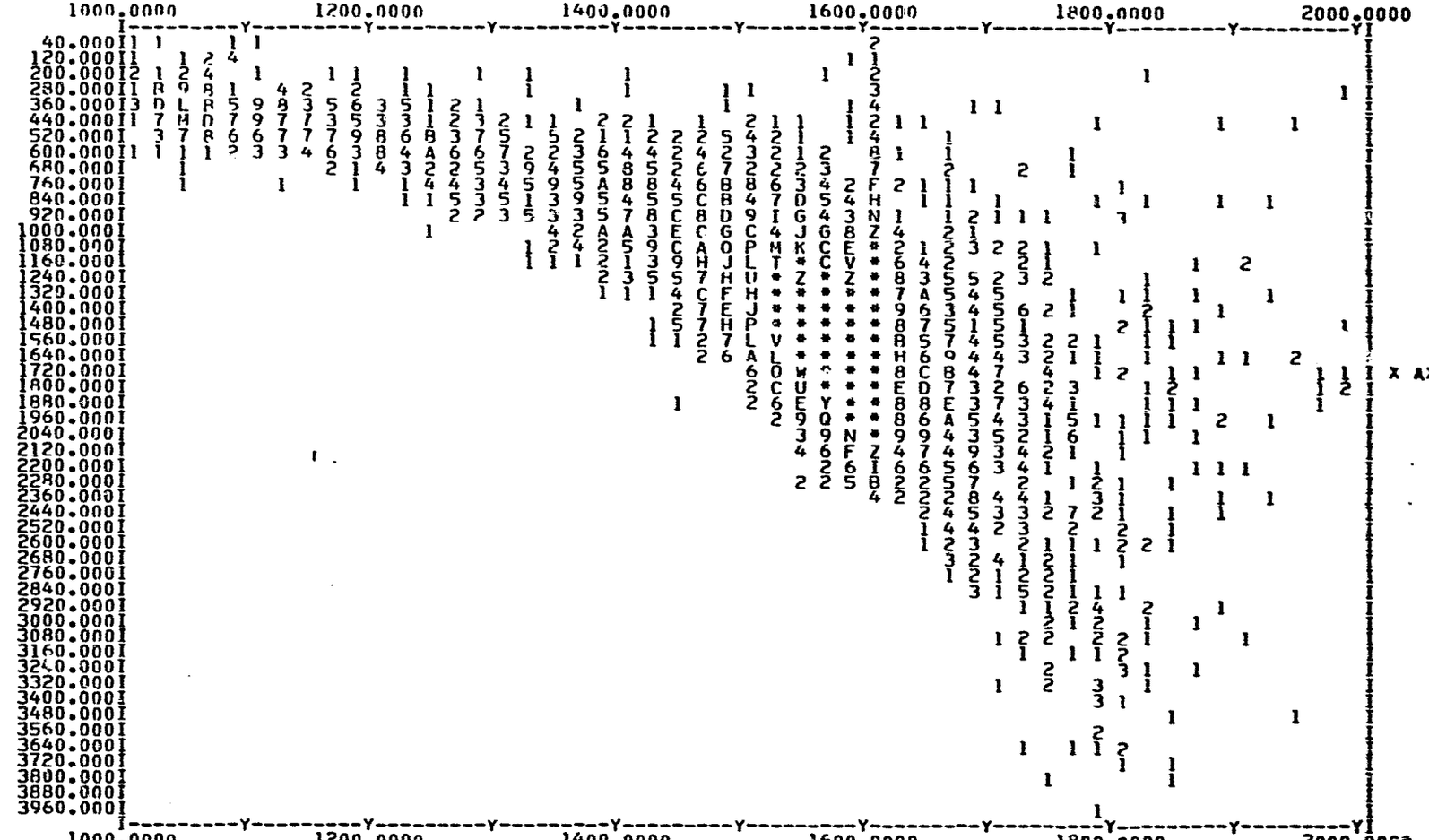
X = (-5.263+- 268.833) + (0.982+- 0.002) * Y COEFF. OF DETER. = 0.97005 STD. ERROR(Y ON X) = 84.778

D-3

PLOT OUTPUT NO 2

DATE. 05/19/81 TIME. 14:45:15

GROUND TRUTH FILE- 12379365 CLASSIFICATION FILE- 12381064



Y AXIS LABEL.

CROP CORN X=ALPHA,Y=TO

X UNDER= 0. X OVER= 43. Y UNDER= 16. Y OVER= 55. PLOTTED= 7028.

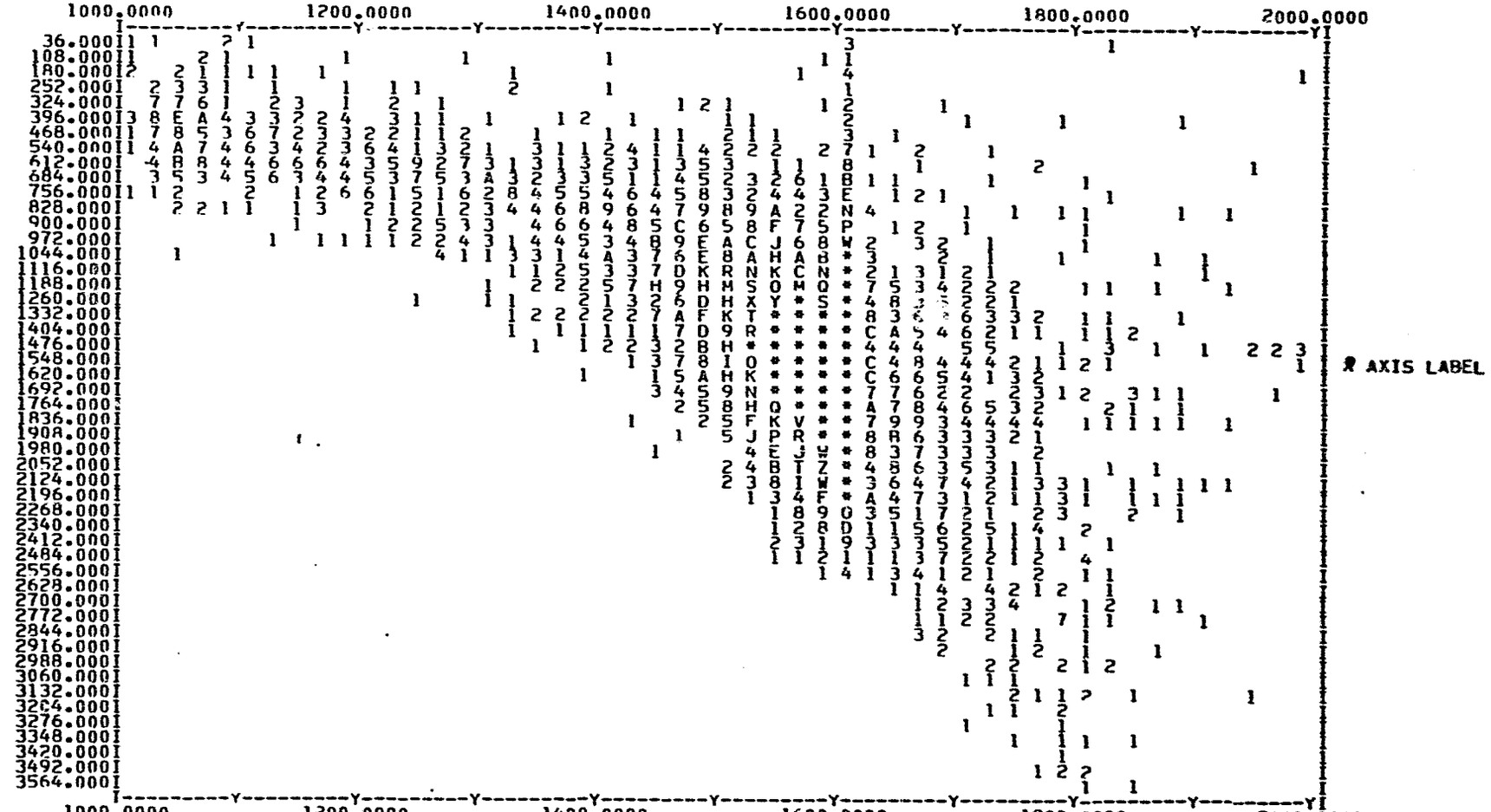
MEAN X= 1451.5876+/- 488.0088 MEAN Y= 1557.0784+/- 128.7195 MSE= **********

X = (**********+-4083.459) + (2.747+- 0.031) * Y COEFF. OF DETER. = 0.52489 STD. ERROR(Y ON X) = 336.398

PLOT OUTPUT NO 3

DATE. 05/19/81 TIME. 14:45:18

GROUND TRUTH FILE- 12379365 CLASSIFICATION FILE- 12381064



Y AXIS LABEL.

CROP CORN X=BETA,Y=TO

X UNDER= 0. X OVER= 44. Y UNDER= 16. Y OVER= 55. PLOTTED= 7025.

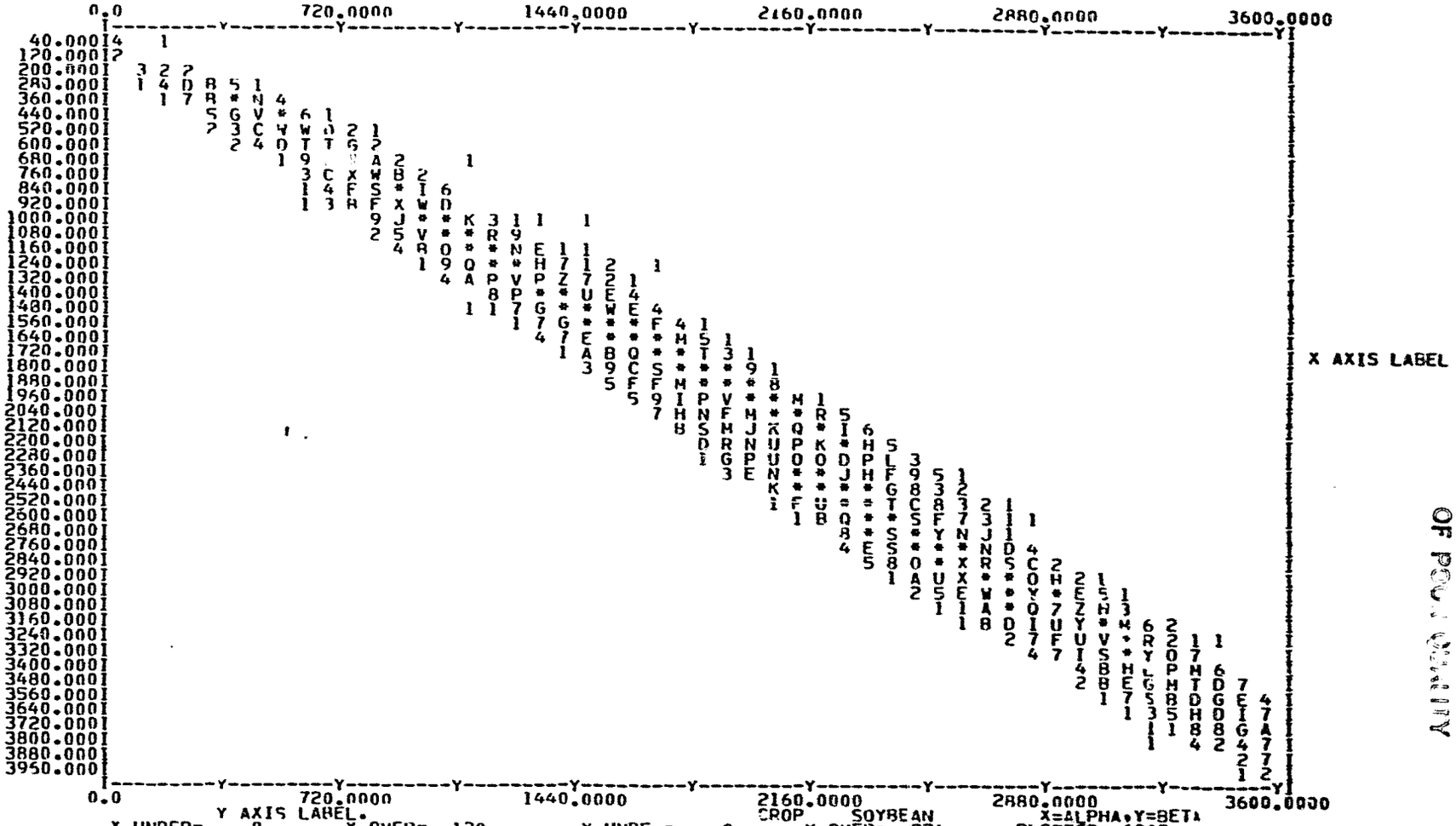
MEAN X= 1483.1072+/- 487.0791 MEAN Y= 1556.9912+/- 128.6820 MSE= **********

X = (*********+-4480.684) + (2.471+- 0.034) * Y COEFF. OF DETER. = 0.42604 STD. ERROR(Y ON X) = 369.040

PLOT OUTPUT NO 4

DATE. 05/19/81 TIME. 14:45:21

GROUND TRUTH FILE- 12379365 CLASSIFICATION FILE- 12381064



CROP SOYBEAN X=ALPHA,Y=BETA

X UNDFR= 0. X OVER= 120. Y UNDF.= 0. Y OVER= 234. PLOTTED= 6803.

MEAN X= 2063.6726+/- 874.0210 MEAN Y= 1972.902+/- 796.0867 MSE= **********

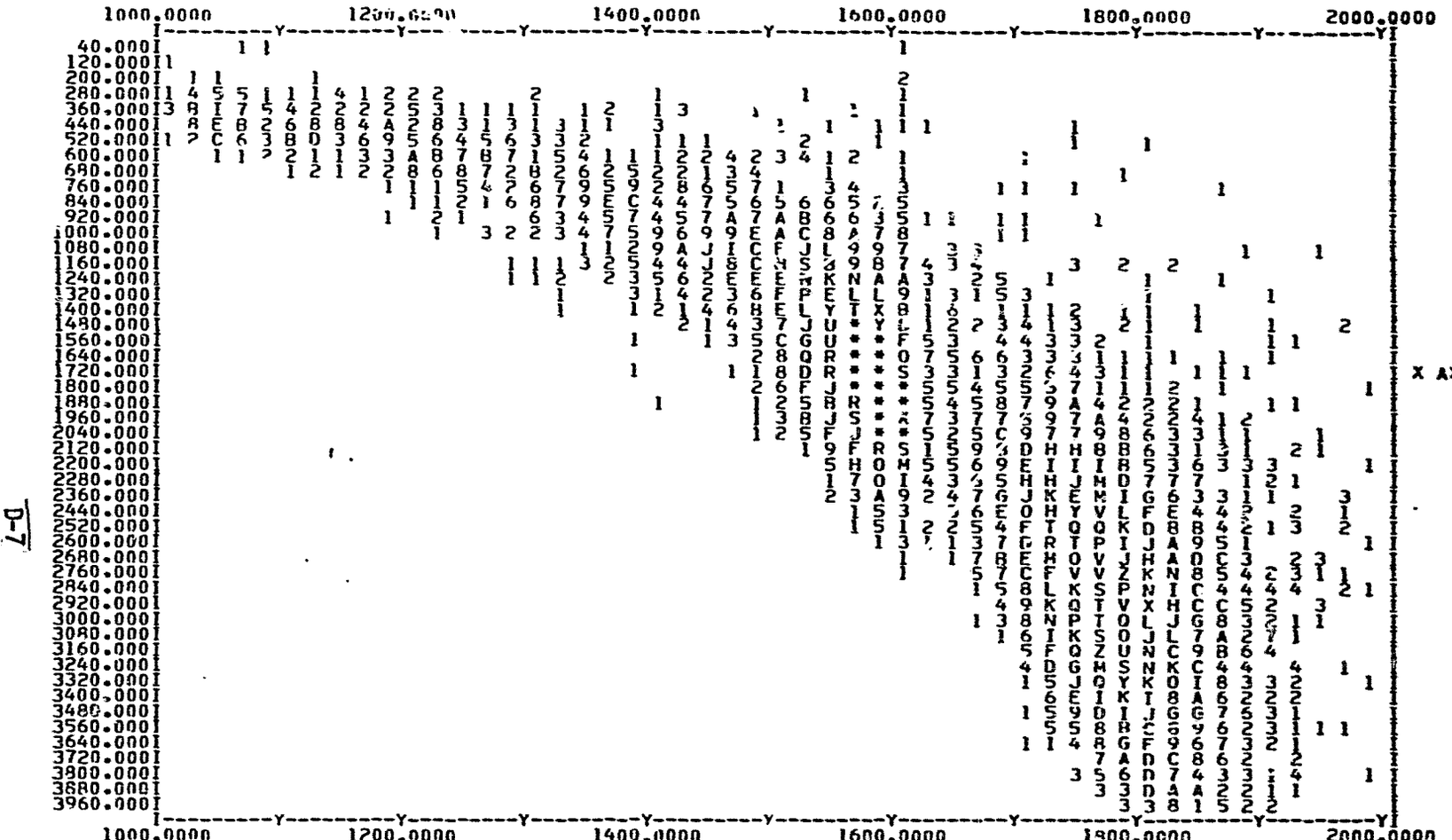
X = (-77.248+- 354.793) + (1.085+- 0.002) * Y COEFF. OF DETER. = 0.97693 STD. ERROR(Y ON X) = 132.753

D-6

PLOT OUTPUT NO 5

DATE. 05/19/81 TIME. 14:45:24

GROUND TRUTH FILE- 12379365 CLASSIFICATION FILE- 12381064



Y AXIS LABEL.

CROP SOYBEAN X=ALPHA,Y=TO

X UNDER= 0. X OVER= 120. Y UNDER= 10. Y OVER= 49. PLOTTED= 6899.

MEAN X= 2095.8833+/- 893.1562 MEAN Y= 1642.2107+/- 177.3153 MSE= **********

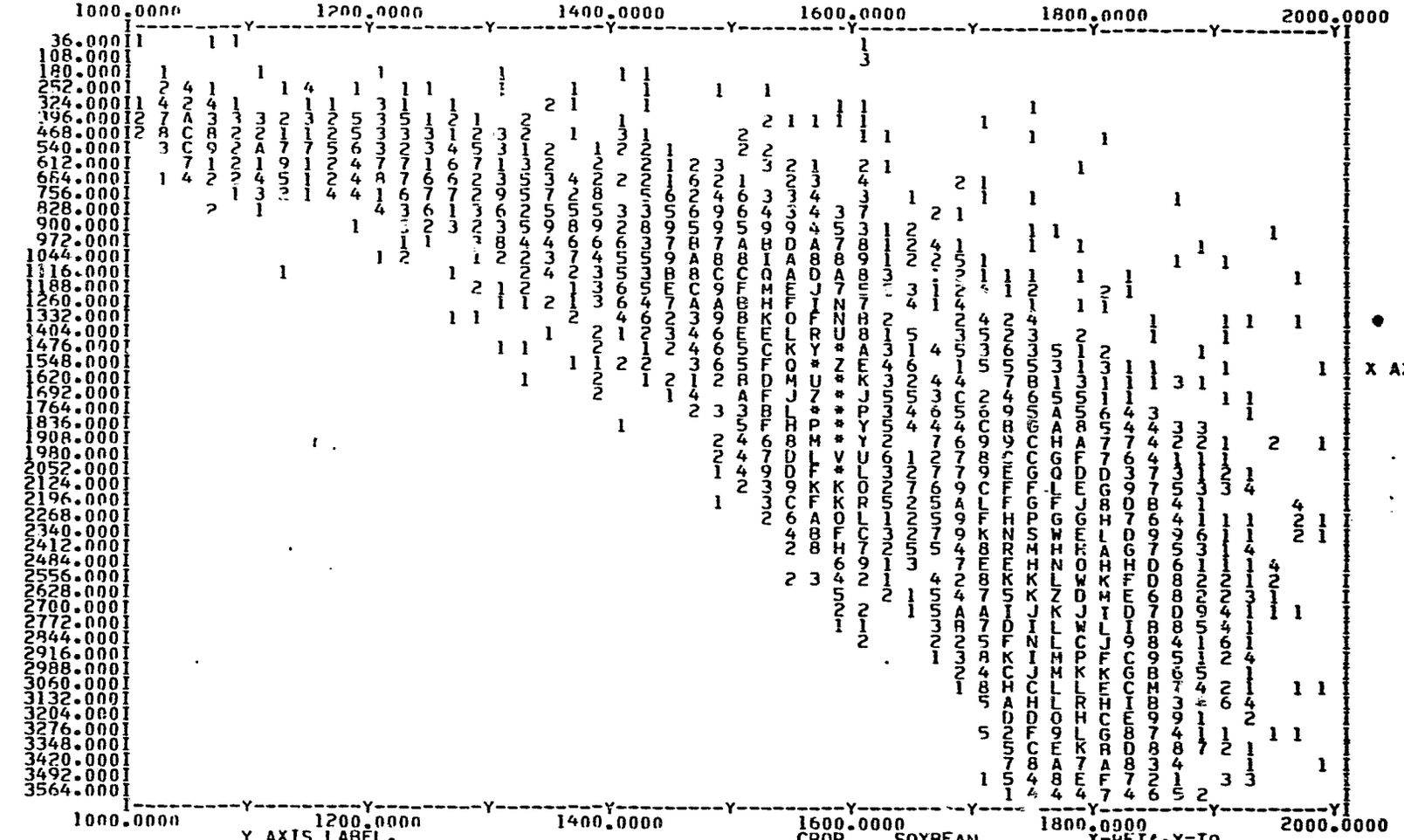
X = (********+-4301.996) + (4.312+- 0.031) * Y COEFF. OF DETER. = 0.73272 STD. ERROR(Y ON X) = 461.786

D-7

PLOT OUTPUT NO 6

DATE. 05/19/81 TIME. 14:45:25

GROUND TRUTH FILE- 12379365 CLASSIFICATION FILE- 12381064



ORIGINAL PAGE IS
OF POOR QUALITY

Y AXIS LABEL.

CROP SOYBEAN X=BETA,Y=TO

X UNDER= 0. X OVER= 234. Y UNDER= 10. Y OVER= 49. PLOTTED= 6785.

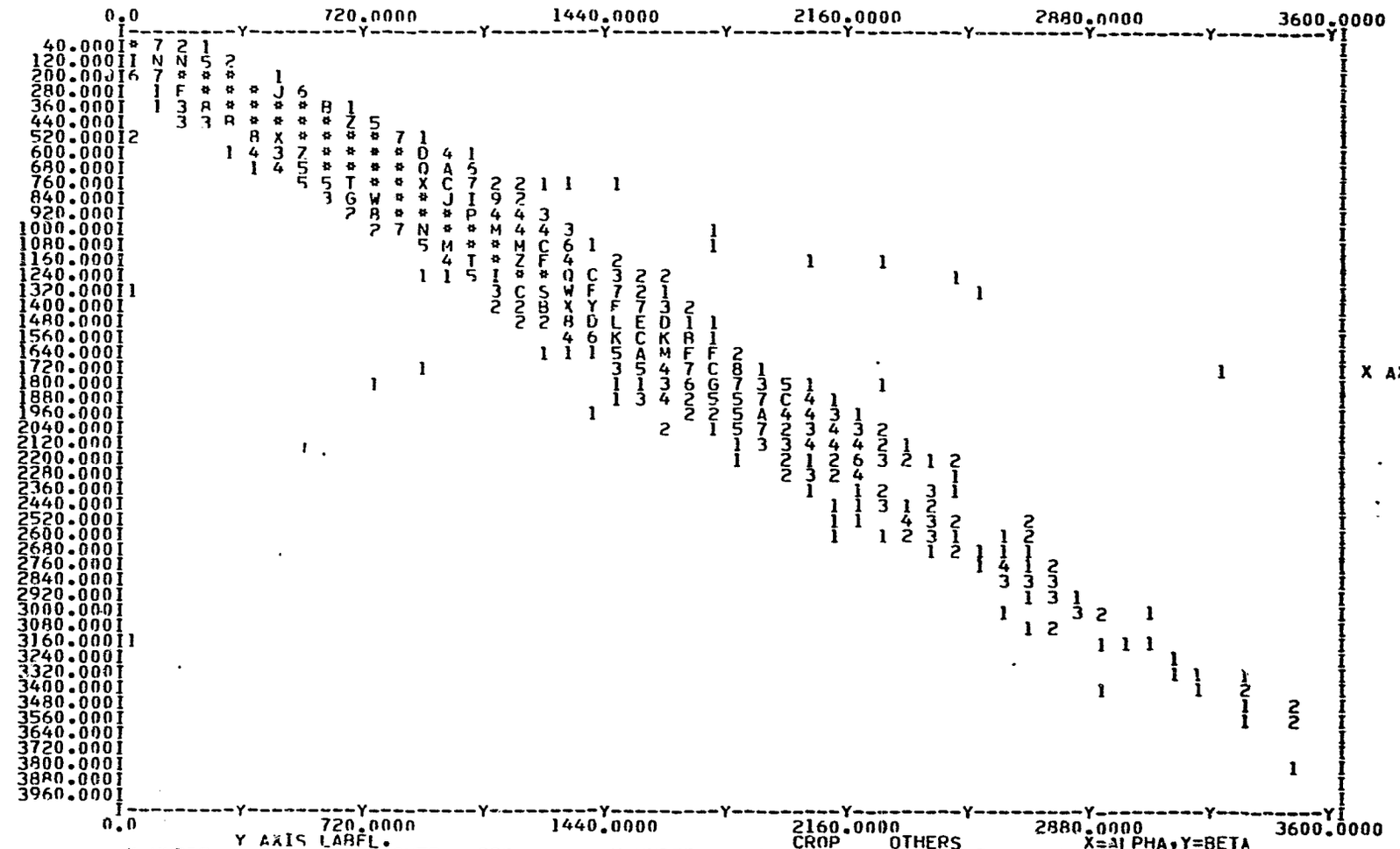
MEAN X= 1975.8848+/- 793.6958 MEAN Y= 1639.4124+/- 177.5128 MSE= *********

X = (*********+-4358.922) + (3.606+- 0.032) * Y COEFF. OF DETER. = 0.65057 STD. ERROR(Y ON X) = 469.206

PLOT OUTPUT NO 7

DATE. 05/19/81 TIME. 14:45:26

GROUND TRUTH FILE- 12379365 CLASSIFICATION FILE- 12381064



Y AXIS LABEL.

CROP OTHERS X=ALPHA,Y=BETA

X UNDER= 0. X OVER= 299. Y UNDER= 0. Y OVER= 287. PLOTTED= 5913.

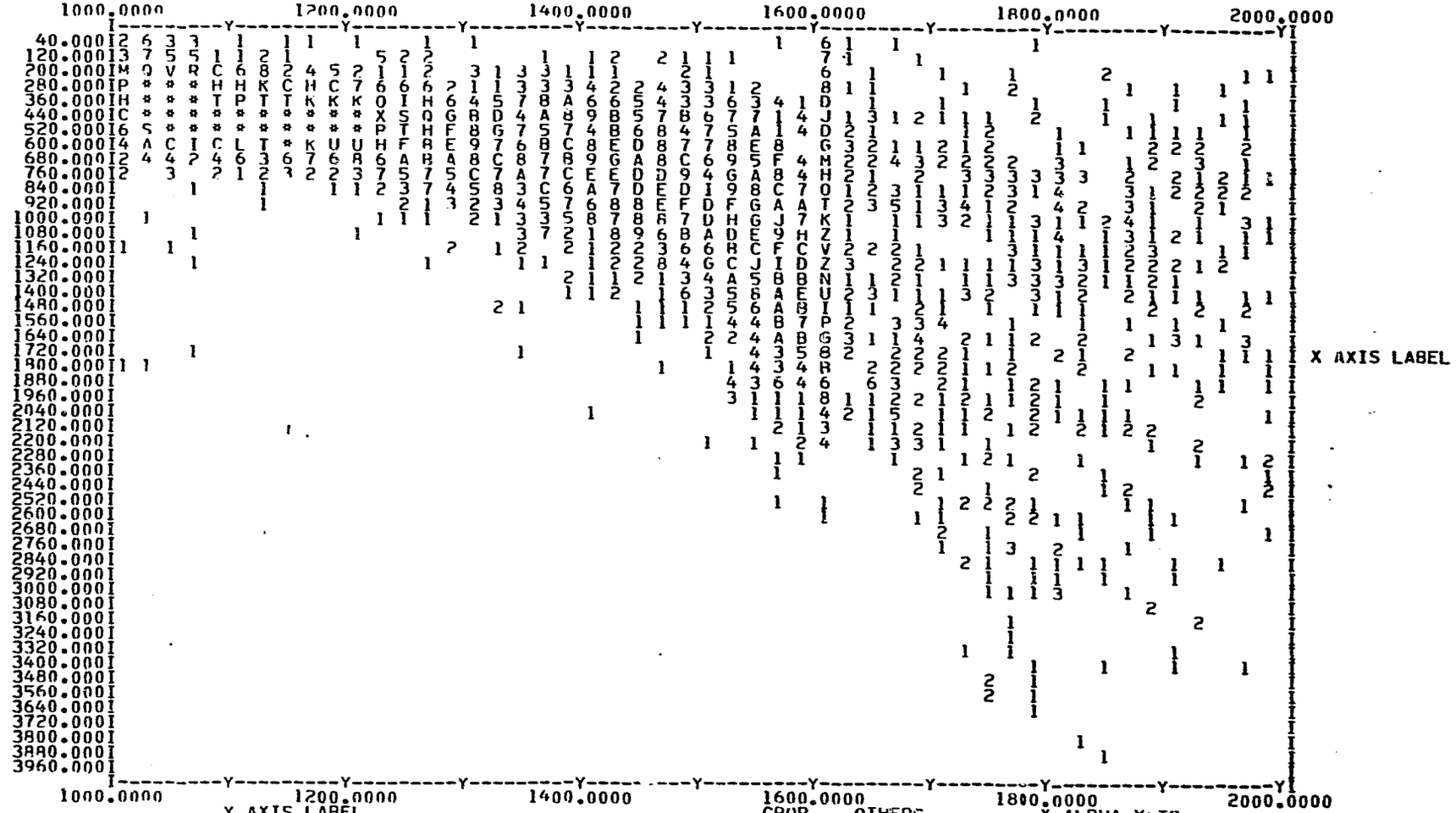
MEAN X= 731.7307+/- 510.7788 MEAN Y= 784.7390+/- 483.0894 MSE= *********

X = (-76.259+- 221.548) + (1.030+- 0.003) * Y COEFF. OF DETER. = 0.94831 STD. ERROR(Y ON X) = 116.136

PLOT OUTPUT NO 8

DATE. 05/19/81 TIME. 14:45:28

GROUND TRUTH FILE- 12379365 CLASSIFICATION FILE- 12381064



Y AXIS LABEL.

CROP OTHERS X=ALPHA,Y=TO

X UNDER= 0. X OVER= 299. Y UNDER= 138. Y OVER= 371. PLOTTED= 5705.

MEAN X= 743.1558+/- 511.6936 MEAN Y= 1342.0459+/- 250.0280 MSE= ********

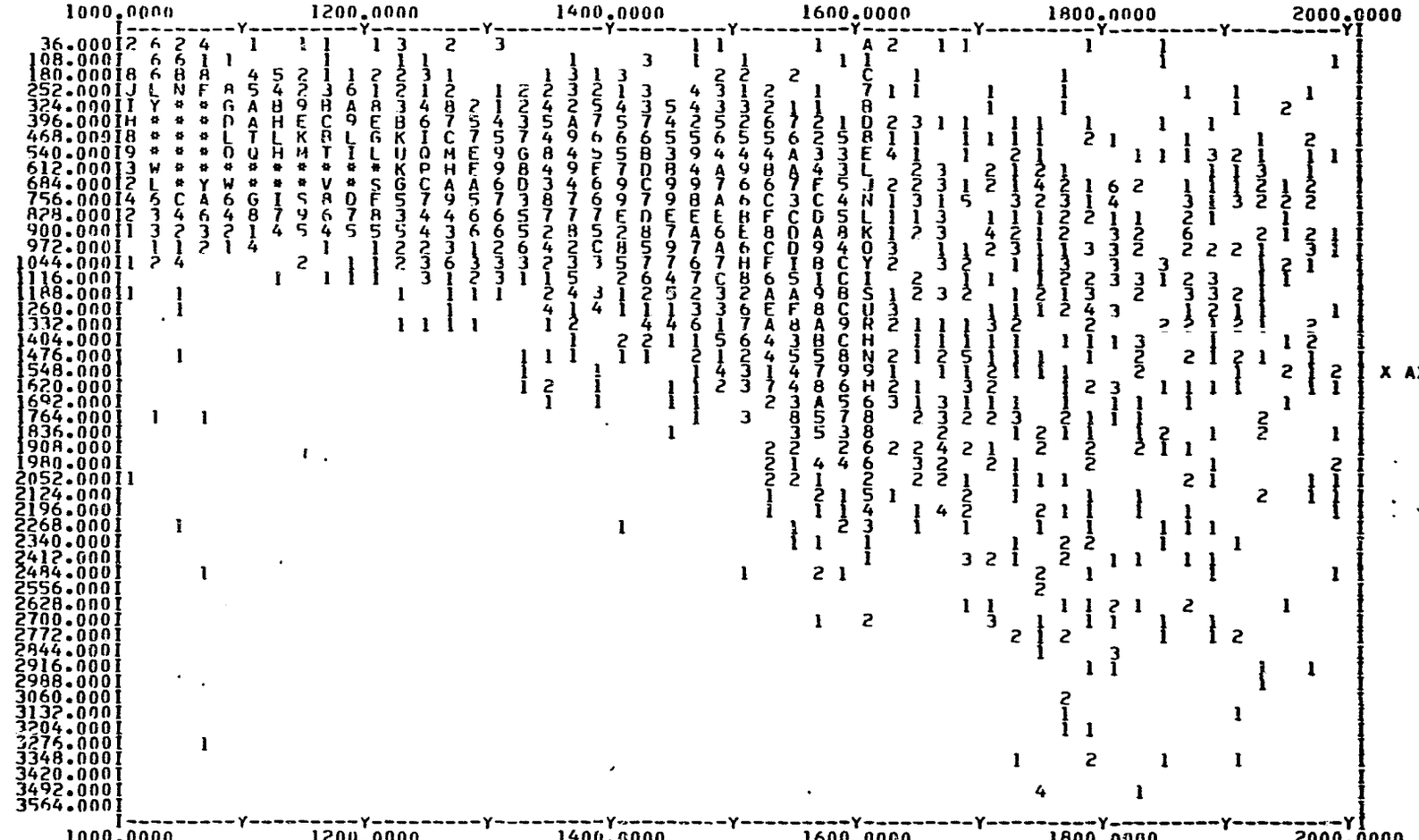
X = (********+-2009.581) + (1.422+- 0.019) * Y COFFF. OF DETER. = 0.48279 STD. ERROR(Y ON X) = 368.029

D-10

PLOT OUTPUT NO 6

DATE. 05/19/81 TIME. 14:45:29

GROUND TRUTH FILE- 12379365 CLASSIFICATION FILE- 12381064



Y AXIS LABEL.

CROP OTHERS X=TA,Y=TO

X UNDER= 0. X OVER= 287. Y UNDER= 138. Y OVER= 371. PLOTTED= 5701.

MEAN X= 796.7039+/- 480.5161 MEAN Y= 1341.9944+/- 249.8854 MSE= *********

X = (-772.188+-2084.473) + (1.169+- 0.020) * Y COEFF. OF DETER. = 0.36962 STD. ERROR(Y ON X) = 381.549

NASA-JSC

ORIGINAL PAGE IS
OF POOR QUALITY