Release Notes for STAT2 Version 1.7: An Addendum to NBS Special Publication 400-75

R. L. Mattis

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
National Engineering Laboratory
Center for Electronics and Electrical Engineering
Semiconductor Electronics Division
Gaithersburg, Maryland 20899

March 1986

U.S. DEPARTMENT OF COMMERCE

NATIONAL BUREAU OF STANDARDS

NBSIR 86-3333

RELEASE NOTES FOR STAT2 VERSION 1.7: AN ADDENDUM TO NBS SPECIAL PUBLICATION 400-75

R. L. Mattis

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
National Engineering Laboratory
Center for Electronics and Electrical Engineering
Semiconductor Electronics Division
Gaithersburg, Maryland 20899

March 1986

U.S. DEPARTMENT OF COMMERCE, Malcolm Baldrige, *Secretary*
NATIONAL BUREAU OF STANDARDS, Ernest Ambler, *Director*

Table of Contents

List of Table and Figures

Release Notes for STAT2 Version 1.7:
An Addendum to NBS Special Publication 400-75

Richard L. Mattis
Semiconductor Electronics Division
National Bureau of Standards
Gaithersburg, MD 20899

ABSTRACT

This document describes the changes which have been made in the STAT2 computer program since its documentation in NBS Special Publication 400-75, Semiconductor Measurement Technology: A FORTRAN Program for Analysis of Data from Microelectronic Test Structures, and NBS Internal Report 83-2779, Release Notes for STAT2 Version 1.31. It is assumed that the reader has these documents, and no attempt is made to review STAT2 features or operation. The changes extend the functionality and versatility of the program. More specifically, the new features added in Version 1.7 include data base extension, an input data format suitable for test sites not in a periodic array, an outlier exclusion algorithm suitable for small numbers of sites, common site exclusions for related data sets, a vector map, a scatter plot, a trend chart, extended macro command file capability, and other changes. Following the description of the changes is an annotated listing of new error messages. This document and the two previous publications cited constitute the documentation of Version 1.7 of STAT2.

Key words: ATE; computer program; contour map; data base; outlier; process validation wafer; scatter plot; trend chart; vector map; wafer map.

WARNING

The structure of the data base has been changed in Version 1.7 of STAT2. Do not try to use any of the data base commands of Version 1.7 on a data base created under Version 1.31 until the data base has been updated as described in section 2.1.

DISCLAIMER

Certain commercial equipment, instruments, or materials are identified in this paper in order to adequately specify the experimental procedure. Such identification does not imply recommendation or endorsement by the National Bureau of Standards, nor does it imply that the materials or equipment identified are necessarily the best available for the purpose.

1. INTRODUCTION AND SUMMARY OF CHANGES

The STAT2 program is intended for analysis, mapping, and correlation of data from microelectronic test structures. The program is written in FORTRAN to run on a Digital Equipment Corporation VAX-11/780 under the VMS operating system. The program can be made to run on other systems through suitable modifications. The original Version 1.01 is documented in NBS Special Publication 400-75, Semiconductor Measurement Technology: A FORTRAN Program for Analysis of Data from Microelectronic Test Structures [1]. The previous update, Version 1.31, is documented in NBS Internal Report 83-2779, Release Notes for STAT2 Version 1.31 [2]. This report documents the current released version, Version 1.7.

The data base feature of STAT2 was originally intended to allow correlations among various data sets to be made. It still serves that purpose, but has been extended to contain additional information such as sample mean and sample standard deviation which may be included in tabulations or summaries produced by user-written programs which access the data base. A new sampling plan has also been added which provides for data bases having only a label file.

A new data input format removes a constraint on test data by allowing test sites to be at any location on a wafer. Previously, test sites had to be located either within a contiguous region on the wafer or at random sites within a periodic array of sites.

Three commands have been added to permit a user to define or determine a set of excluded sites which are to apply to several related data sets. This is useful when a single device has multiple parameters associated with it. The three commands are CXR, CXO, and CXC.

Three new data displays are available: a vector map, a scatter plot, and a trend chart. In the vector map, two data sets are analyzed separately, stored in a data base, then recalled to produce a vector map display. In the scatter plot, one data set is plotted against another, both of which were previously stored in a data base. In the trend chart, a display can be made of the variation of up to 256 data sets, all of which are recalled from a data base.

Two new commands allow a user to multiply all data values by a constant or take the common logarithm of all data values. The MAC command has been enhanced to allow some nesting of macro command files and to provide for a default .MAC extension. A PAG command advances the log file to a new page. A LOG command provides for logging of interactive command input. A control-C interrupt capability has been added to five commands so that unwanted long listings can be aborted without exiting the program. The HELP command now makes use of the VMS Help facility. Informational messages have been added to the site exclusion commands. Three programming errors have been corrected. Finally, CRDB and CORTABLE have been modified to accommodate the extended data base and new sampling plan.

The following index will allow the user to go directly to the feature of interest.

2. DATA BASE CHANGES

2.1. Additional Information Stored in the Label File

Additional fields have been added to the data base label file. These fields contain the sample mean, sample standard deviation, number of sites tested, and number of included sites. Included sites are those sites which were not determined to be outliers and so were not excluded from the population of data values for the particular data set. The location and size of the fields in the data base label record are given in Table 1. The record length of the label file has been increased from 80 bytes to 132 bytes. The CRDB program has been changed to reflect the change in size. No change has been made to the data base data file.

When information is extracted from the data base, such as by the LDB command, the information can no longer fit in the 80-character width of many terminal screens, resulting in wrap-around. However, the STAT2.LOG file prints each entry on a separate line. The displays produced by the data base commands contain the new column headings NTOTAL, NGOOD, MEAN, and STDDEV over the number of sites tested, number of included sites, sample mean, and sample standard deviation, respectively.

Because the new information written to the data base must be known before it can be written, it is now necessary to execute PRS with no intervening site exclusions or reinsertions or change in N value prior to executing WDB. N is a user-defined multiple of the sample standard deviation which is set by the ENN command.

Table 1. Identification of Fields in the Data Base Label File

Bits	Name	Description	Format
1		Not used.	1X
2-6	ENT	Entry number (also record number within file).	I5
7	DLT	Delete flag.	A1
8-13	PTR	Pointer to header record in data base data file.	I6
14-17	NDAT	Number of data points in the sample.	I4
18-20		Not used.	3X
21-23	PAT	Pattern number.	I3
24-26	LOT	Lot number.	I3
27-29	WAF	Wafer number.	I3
30-33	DEV	Device number.	I4
34-37	PCODE	Parameter code.	I4
38-39		Not used.	2X
40-48	DATE	Date written in dd-mmm-yy format.	9A1
49		Not used.	1X
50-57	TIME	Time written in hh:mm:ss format.	8A1
58		Not used.	1X
59-62	NTOTAL	Total number of sites in data set.	I4
63-66	NGOOD	Number of included sites in data set.	I4
67-79	DMEAN	Sample mean of included sites.	E13.6
80-92	DSIGMA	Sample standard deviation of included sites.	E13.6
93-132		Not used.	

The intended application of the extended data base is to enable tabulations or summaries to be produced by user-written programs which access the data base. For example, a user might have an application in which the same 50 parameters are measured at each of 20 sites on a lot of wafers. A file containing descriptive text relating to each of the 50 parameters could be prepared. A user-written program could then combine this descriptive text with the sample mean, sample standard deviation, and yield information now contained in the data base to produce a tabulation for each wafer and a summary for the entire lot.

To convert data bases created under STAT2 Version 1.31 so that they can be read and written to by Version 1.7, a program, command file, and help file are given in Appendices I, II, and III. This updating must be done once to all data bases created using the CRDB program supplied with STAT2 Version 1.31. The conversion program supplies zero values for the NTOTAL, NGOOD, MEAN, and STDDEV fields of the data base label file. Note that the TRC command (see section 8) cannot process entries containing these zero values.

2.2. Sampling Plan With No Data File

There may be applications where the sample mean and sample standard deviation which are now contained in the label file are all that are of interest to the user and no correlations among data sets are needed. In this case, the data base data file becomes superfluous. Sampling plan 5 has been added to STAT2 to deal with such situations. CRDB has been altered such that when a data base is created with sampling plan 5, only a label file is created.

When STAT2 tries to access a data base having sampling plan 5, certain operations are modified or become illegal. The PTR and NDAT fields of such a data base are always 0. A WDB command writes only to the label file. A DEL command operates in the same manner as it did in previous versions. A GET command is illegal, and since SDB must be preceded by GET, SDB is implicitly illegal. When LDB or LEN are such that data values are requested, the command is altered so as not to request data values and a warning message is printed.

3. INPUT DATA FORMAT FOR NONPERIODIC SITE PLACEMENT

A new data format is now readable using the REA command. It is denoted format 6. Format 6 enables data to be read when the data sites are not located within a periodic grid. Site locations can be arbitrary, and coordinate positions are expressed in terms of micrometers from some user-defined origin rather than in unit cell distances from a reference cell location.

As with formats 4 and 5, the data are read from a .LGA file produced on an Accutest 3000 test system, and the second parameter of the REA command specifies the serial order of the data set within the data from each die. An example of a .LGA file which can be read using format 6 is given in figure 1.

In order to read data using format 6, use the ASG command to enter the file specifier of the .LGA file containing the data to be analyzed, and issue the REA command as REA,6,i,

```
TPN NM4V202
STA     2
DAT 14-OCT-84
BTI 15:22:30
ETI
OPE
LOT
DEV
WFT M48U11
TSN
DSN   431
WFN    11
DIE     1
POS  2.945600E+04 8.848400E+04  <--- X and Y coordinates of first site
VAL     1 -2.914521E+00         <--- Value of first parameter at first site
VAL     2  3.450001E-07         <--- Value of second parameter at first site
BIN     0
DSN   432
WFN    11
DIE     2
POS  3.910800E+04 8.848400E+04  <--- X and Y coordinates of second site
VAL     1 -3.008414E+00         <--- Value of first parameter at second site
VAL     2  3.570000E-07         <--- Value of second parameter at second site
BIN     0
DSN   433
WFN    11
DIE     3
POS  4.876000E+04 8.848400E+04
VAL     1 -3.058542E+00
VAL     2  3.600002E-07
BIN     0
DSN   434
WFN    11
DIE     4
POS  6.069800E+04 8.848400E+04
VAL     1 -3.056347E+00
VAL     2  3.860001E-07
BIN     0
DSN   435
WFN    11
DIE     5
POS  2.463000E+04 8.137199E+04  <--- X and Y coordinates of last site
VAL     1 -2.843635E+00         <--- Value of first parameter at last site
VAL     2  3.780001E-07         <--- Value of second parameter at last site
BIN     0
EOF
***
```

Figure 1. Example of an input data file suitable for reading by format 6.

where i is the data set number to be read. There is no .SAD file or equivalent in format 6. Commands such as LAP, LNS, and LXP which list information about particular sites give the coordinate locations in micrometers from a user-defined origin rather than in cell coordinates.

Internally, when reading data in format 6, STAT2 stores data values in DATA(1,1), DATA(1,2), ..., DATA(1,32), DATA(2,1), ... , etc. until all values (up to 1024 of them) are read in. The STEND array in this case contains no information about the site locations but only defines the elements of the DATA array which contain actual data. The actual site locations are contained in two arrays, XCOL and YROW, each dimensioned (32,32). The LSITES array still contains a record of excluded sites. The topological type has no meaning in format 6, and the TYPE array elements of all sites are set equal to 10.

The user must be careful when designing a sampling plan to write a data base using format 6. Instructions for adding a sampling plan to STAT2 are contained in comment statements in subroutine DB02; however, these instructions address the case in which the subscripts of the DATA array give the actual site locations. A sampling plan can still be designed, knowing the internal use of the DATA array as explained in the previous paragraph and knowing the order of actual site locations as contained in POS statements in the .LGA data file. The whole matter of site locations can be circumvented by using sampling plan 4, which includes all sites measured in the sampling plan. Additionally, because the sites are not contiguous in format 6, the following commands are not allowed and produce an error message: AIP, AXP, FPL, FQD, MP1, MP2, MP3, MP4, PLT, SPL, SQD, VMP, and XPP.

Additional information about format 6 is contained in the introductory comments to subroutine FMT6SR.

4. OUTLIER EXCLUSION ALGORITHM FOR SMALL NUMBER OF SITES

To meet the need to analyze small data samples, a new command, XSS (for exclude small sample), has been added to STAT2 which can be used to identify and exclude outliers when the number of included sites is in the range 3 to 25, inclusive. The format of the command is XSS,P1, where P1 represents the probability that one or more good sites might be excluded in a situation where there are no true outliers. P1 may be given values of 0.6, 0.4, 0.2, 0.1, 0.04, 0.02, or 0.01. Other values produce an error message.

The algorithm employs the Dixon test [3]. For this test, data values are first sorted, then the tests are made on the extreme values. If one of the extreme values is excluded, a second examination is made based on the new extreme values. This process continues until no more values are excluded. Conceivably, with a sample size of only 3 or 4, XSS may exclude one or two sites leaving fewer than the minimum three sites required. It is known that consecutive applications of the Dixon test may fail to detect outliers when two or more outliers are present. A good discussion of this problem, and several procedures for overcoming it, are given in [4].

Following execution of XSS, the user probably would want to calculate and print statistics using PRS. A typical command sequence might be as follows:

```
ASG FILE.DAT    ! Assign input data file.
REA 3 15        ! Read in the desired data set.
XGT 1E10        ! Exclude values which are
XLT 0           !    known to be bad data or
XIP 1 6         !    to have physical defects.
XSS 0.2         ! Invoke XSS.
PRS             ! Calculate and print statistics.
END             ! End of this example.
```

Additional information concerning the XSS command is contained in the introductory comments to subroutine XSS.

5. CUMULATIVE SITES EXCLUSIONS

Three new commands have been added to STAT2 to allow the respective sets of excluded sites for several related data sets to be accumulated or pooled and that pool of excluded sites to be applied to each of the related data sets. These commands are intended for use in situations where a single device has more than one measured parameter and where these parameters have values which are not independent of one another. In such a situation, if any parameter value associated with the device is found to be an outlier, then all measurements related to that device which might depend on that parameter should also be declared outliers. The three new commands which provide for this are CXR, CXO, and CXC. They operate by moving data between the existing LSITES array and the new LCUM array, both dimensioned (32,32). The LSITES array contains a 0 at the coordinate positions corresponding to an included site, and a 1 at the coordinate positions corresponding to an excluded site.

CXR resets or initializes the LCUM array by writing a 0 into all elements of the LCUM array which represent test sites, that is, within the region defined by the STEND array. The STEND array is discussed on pages 3 and 4 of [1].

CXO performs a merging operation of the form

LSITES(I,J)   OR LCUM(I,J) -> LCUM(I,J)

where all (I,J) elements within the region defined by the STEND array are processed. That is, the current LSITES array and the LCUM array are OR'ed together and the result put in the LCUM array. The sets of excluded sites of several data sets may be merged in this way.

When all related data sets have been processed, the CXC command can be used to copy the LCUM array back into the LSITES array, thereby excluding all sites represented by the accumulated set of exclusions. The following example illustrates how this works. The example assumes that data sets 14, 15, and 16 in a particular file are related and that a set of excluded sites common to all of them is to be applied to all of them.

```
ASG FILE.DAT    ! Specify the data file.
CXR             ! Clear the LCUM array.
REA 3 14        ! Read data set 14.
XOL 0.2         ! Identify and exclude outliers.
CXO             ! OR these exclusions with the (empty) LCUM array.
REA 3 15        ! Read data set 15.
XOL 0.2         ! Identify and exclude outliers.
CXO             ! OR these exclusions with those already in LCUM.
REA 3 16        ! Read in data set 16.
XOL 0.2         ! Identify and exclude outliers.
CXO             ! OR these exclusions with those already in LCUM. The
*               !    LCUM array now contains an accumulation of excluded
*               !    sites from data sets 14, 15, and 16.
CXC             ! Apply the combined exclusions to the current data set,
*               !    which is data set 16.
PRS             ! Calculate statistics based on the cumulative site
*               !    exclusions.
REA 3 15        ! Read data set 15 back in.
CXC             ! Apply combined exclusions.
PRS             ! Calculate statistics.
REA 3 14        ! Read data set 14 back in.
CXC             ! Apply combined exclusions.
PRS             ! Calculate statistics.
END             ! End of this example.
```

Note that, once the three individual sets of excluded sites have been OR'ed into LCUM, it is necessary to go back and read in all but the last of the data sets processed (data set 16 in this example) in order to apply the accumulated exclusions to those data sets. Note also that other operations which might follow PRS (such as the making of maps or histogram operations) are omitted here in the interest of simplicity. CXO and CXC cannot be executed unless a data set has been first input using the REA command.

A word of caution is in order with respect to these three commands. STAT2 does not verify that all STEND arrays corresponding to these data sets are the same. If all STEND arrays are not the same, or if there is not a sound statistical reason for accumulating excluded sites, then results produced using these commands will be meaningless.

6. VECTOR MAP

The ability to make vector maps has been added to the STAT2 program. Vector maps are useful for displaying two sets of data simultaneously and have particular application in analysis of photomask misalignment as measured by an electrical alignment test structure [5]. Using this application to illustrate how the new command works, a user would have in a disc file or files two data sets, one of which contains misalignments in the x-direction, the other of which contains misalignments in the y-direction. Each of these data sets must be analyzed separately using STAT2 commands for reading data, identifying and excluding outliers, and writing to a data base.

The format of the VMP command is

```
VMP P1,P2,P3,P4
```

where P1 is the plot diameter in inches (from 3 to 8, inclusive, is allowed), P2 is the entry number of the data base entry containing the x-misalignment values, P3 is the entry number of the data base entry containing the y-misalignment values, and P4 is a label flag, set nonzero if a map label is to be prompted for and entered on the following command line, or set zero if no label is to be entered. The lengths of the vectors are automatically scaled so that the longest vector has a length equal to the grid spacing of the vector origins on the map. Figure 2 gives an example of a vector map of photomask misalignment measured using an electrical alignment test structure is displayed.

In order for VMP to be used, six conditions must be met. (1) The data base must have been written using sampling plan 4. The code was written around the assumption that the user would want a map of all sites rather than a sampling of sites. (2) REA formats 5 and 6 are not permitted. (3) The entry numbers (P2 and P3) must be legal and not marked deleted. (4) PAT, LOT, WAF, and DEV for the two entries must match. This is to prevent the user from accidentally making a map using two data sets which bear no relation to one another. (5) The STEND array pertinent to the data must be supplied by issuing an REA command prior to VMP. The data base itself contains no information on site locations, so a valid STEND array must be present so that site locations can be reconstructed. (6) NUM must be equal to NDAT, that is, the number of sites calculated from the STEND array must equal the number of data records per entry in the data base. This is a necessary but not sufficient condition that the STEND array which was read by the REA command is in fact the correct one. Sites which are marked as excluded in either of the entries result in a zero-length vector (blank) at the location on the map. It is not necessary to use CXR, CXO, and CXC to create a set of included sites common to both data sets; this is handled automatically by VMP.

A typical command sequence is given as follows:

```
ASG DATAFILE.EXT        ! Assign the file containing wafer test data.
REA 3,10                ! Read the x-misalignment data.
XOL 0.2                 ! Identify and exclude outliers.
WDB 34,2,17,105,1       ! Write data to a data base.
REA 3,11                ! Read the y-misalignment data.
XOL 0.2                 ! Identify and exclude outliers.
WDB 34,2,17,105,2       ! Write data to the data base.
VMP 8 2 3 1             ! Make the vector map.
SAMPLE VECTOR MAP       ! Entered map label.
```

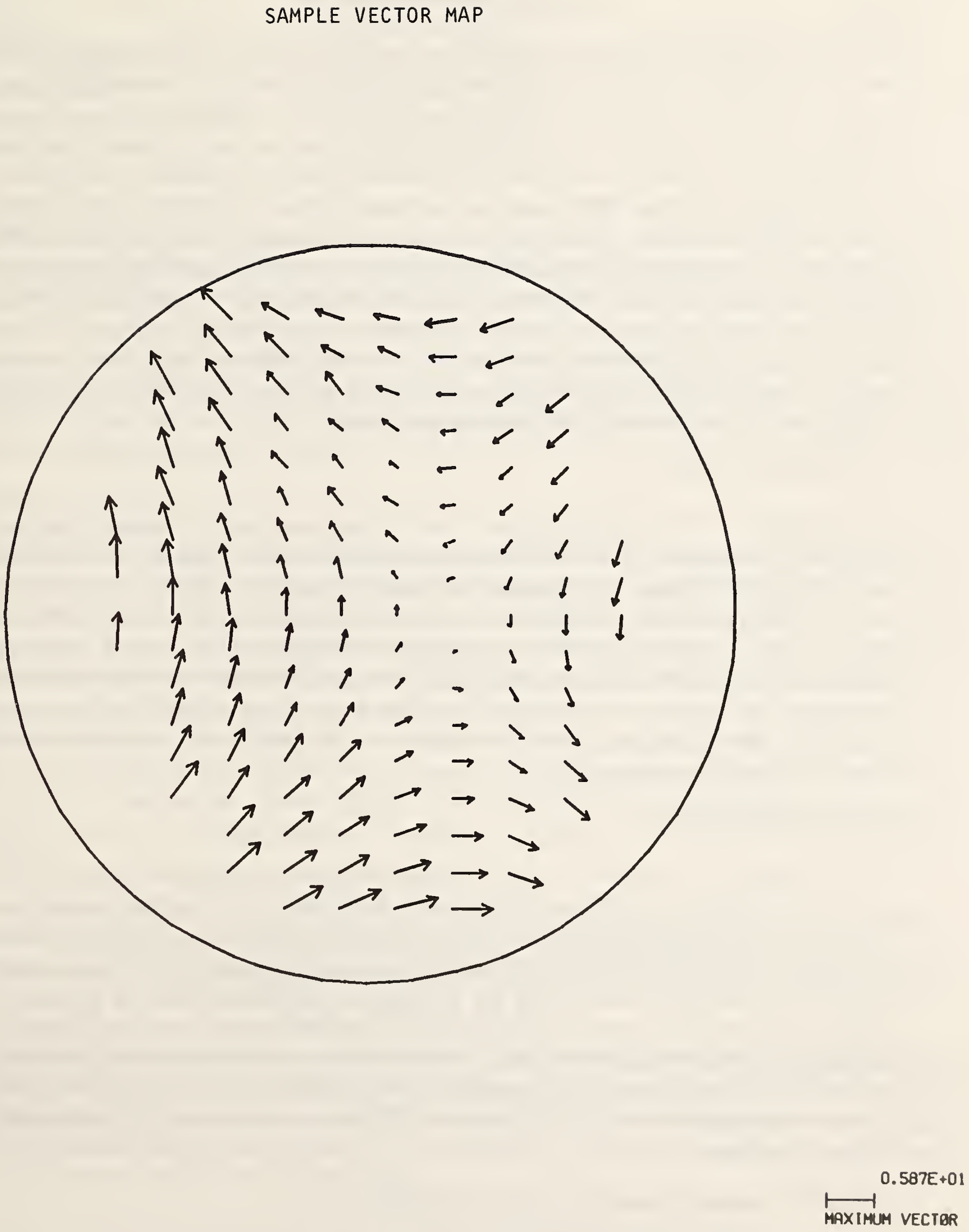

Figure 2. Sample vector map.

```
END                           ! End of this example.
```

Not shown is the fact that the data base uses sampling plan 4. This would be specified when running CRDB to create the data base. Note that in the VMP command it is necessary to know the entry numbers which are pertinent. The entry numbers are reported after WDB is issued. Should the user want to make a vector map at some time after which the STEND array has been overwritten by other STAT2 commands, it would be necessary to reissue the ASG and REA commands above prior to the VMP so that a valid STEND array would be present. When VMP is executed, the map does not go to STAT2.LOG; a metacode file named IOP020.DAT is produced. A metacode translator named MCVAX must be invoked to produce the map from the metacode file after STAT2 has terminated. This is also the case for the contour map produced by the MP3 command. The translator is part of the NCAR plotting package [6]. Because NBS does not have a commitment to support the NCAR plotting package, the level of support available for VMP may not be as great as for the rest of STAT2 in the long term.

7. SCATTER PLOT

A new scatter plot command allows a user to plot one data set against another. Such a plot shows the extent to which the two data sets correlate, but shows more information about the nature of the correlation than does the sample correlation coefficient calculated by the SDB command. A linear fit to the point pairs is calculated by the method of least squares. As with the vector map, data which go into the scatter plot must first be written to a data base. In the case of SCA, however, it is not necessary that the data base have sampling plan 4, and data from random site locations (REA formats 5 and 6) are permitted. Sampling plan 5, which contains no data file, is not permitted, however. The format of the SCA command is:

```
SCA,P1,P2,P3,P4,P5
```

where P1 and P2 are the plot width and height in inches, respectively (from 3 to 8 inclusive are allowed), P3 is the entry number of data base entry to be plotted on the x-axis, P4 is the entry number of data base entry to be plotted on the y-axis, and P5 is a label request flag which is set nonzero if a label for the map is to be prompted for and entered on the next line or zero if no label is to be entered. In order for SCA to be used, the entry numbers (P3 and P4) must be legal and not marked deleted, and the number of points (NDAT) in both sets must be equal. Sites which are marked as excluded in either of the entries do not appear on the scatter plot or in the calculations. Therefore, as with VMP, it is not necessary to use CXR, CXO, and CXC to create a set of included sites common to both data sets.

A typical command sequence is given as follows:

```
ASG DATAFILE.EXT              ! Assign the file containing the test data.
REA 3,1                       ! Read in first data set.
XOL 0.2                       ! Exclude outliers.
```

```
WDB 17,3,8,74,1             ! Write to data base.
REA 3,2                     ! Read in second data set.
XOL 0.2                     ! Exclude outliers.
WDB 17,3,8,52,7             ! Write to data base.
SCA 6,8,17,18,1             ! Make scatter plot.
SAMPLE SCATTER PLOT         ! Supply plot label.
END                         ! End of this example.
```

The output produced by SCA includes the point pairs plotted as plus signs, the line which represents the linear least squares fit to the points, and some statistics. The statistics are the equation of the fitting line, the correlation coefficient, the percent standard deviation of the slope of the fitting line and the percent standard deviation of the y-axis intercept of the fitting line. As with the vector map, the scatter plot does not go to STAT2.LOG; rather it goes to a metacode file which must be processed by the MCVAX translator supplied by NCAR. An example of a scatter plot produced by SCA is shown in figure 3 where the threshold voltages of two MOSFETs having different channel widths are displayed. The statistics relating to the fit are printed in STAT2.LOG as well as on the plot itself.

8. TREND CHART

A trend chart is a display which shows the manner in which a parameter has varied over a period of time. It may be used to display, for example, instrument calibration done on a daily basis, or transistor threshold voltage on a wafer from each of a number of process lots produced over a period of time. An example of a trend chart produced by STAT2 using simulated data is shown in figure 4. The horizontal axis is a time scale normalized to the percent of the time interval between the earliest and latest times of the entries plotted. The time and date of the first and last sets of data plotted are given under the plot. Each vertical bar represents the distribution of included data values measured on a particular parameter at a particular point in time. The bars extend from the value represented by sample mean plus sample standard deviation to sample mean minus sample standard deviation.

The vertical scale is automatically determined to provide a display which approximately fills the plot area. The time interval between parameter ranges may be as short as a minute or may be as long as several years. Scaling of both axes is done automatically by STAT2. Up to 256 data sets may appear in the plot.

The format of a command to produce a trend chart is the same as the format used in the LDB and SDB commands as explained on pages 11 through 13 of [2]. The keywords VALUES and COEF are illegal in the TRC command. As an example, to produce a trend chart of measurements made repeatedly on a particular device on a particular wafer, over a specified period of time, one might enter:

```
TRC PAT 34 LOT 5 WAF 16 DEV 23 PCODE 2 DATE 2/4/85-6/22/85.
```

All data base entries which match the specification of the TRC command are included in the trend chart. After the TRC command is entered, the user is prompted for a title to

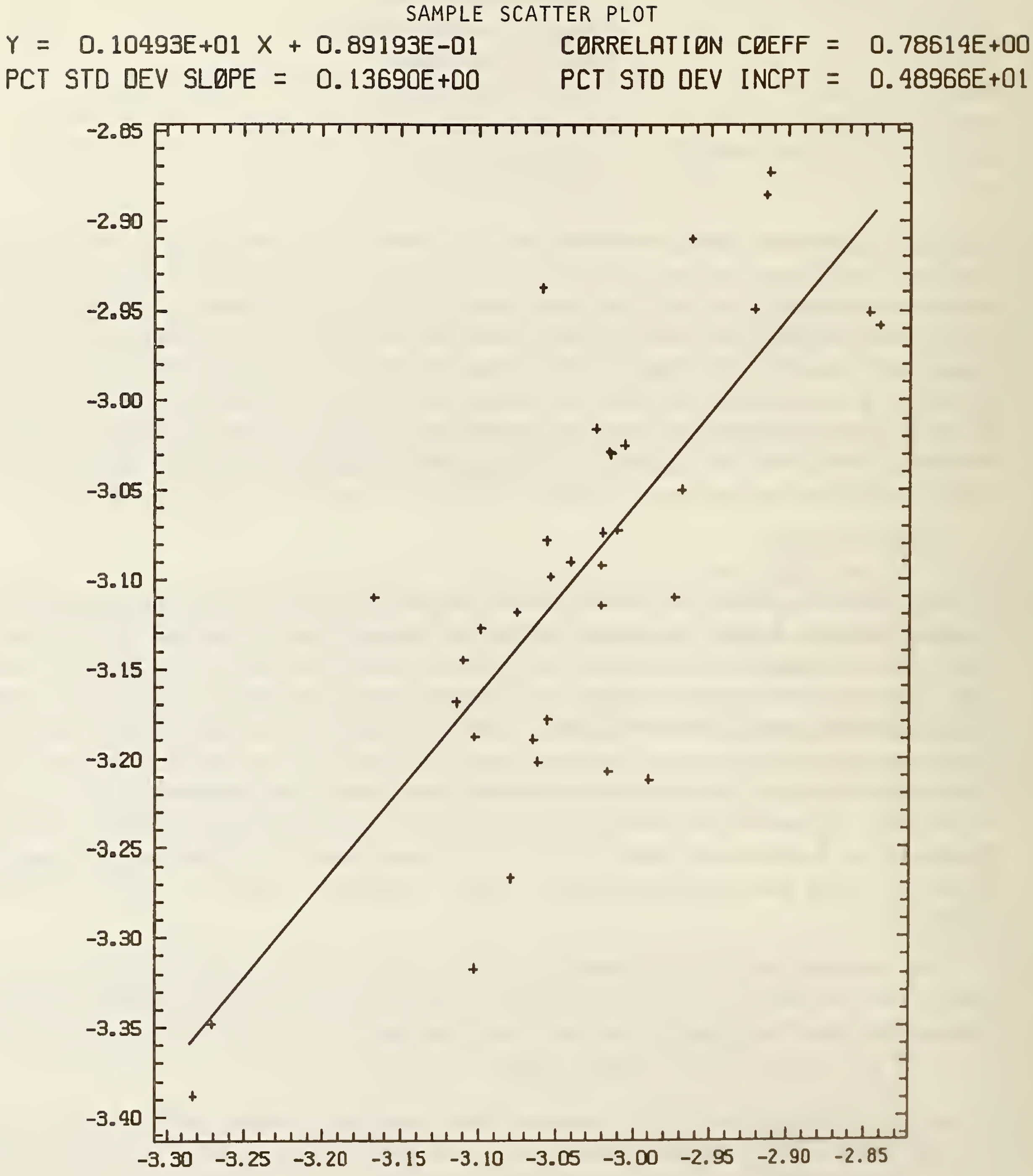

Figure 3. Sample scatter plot.

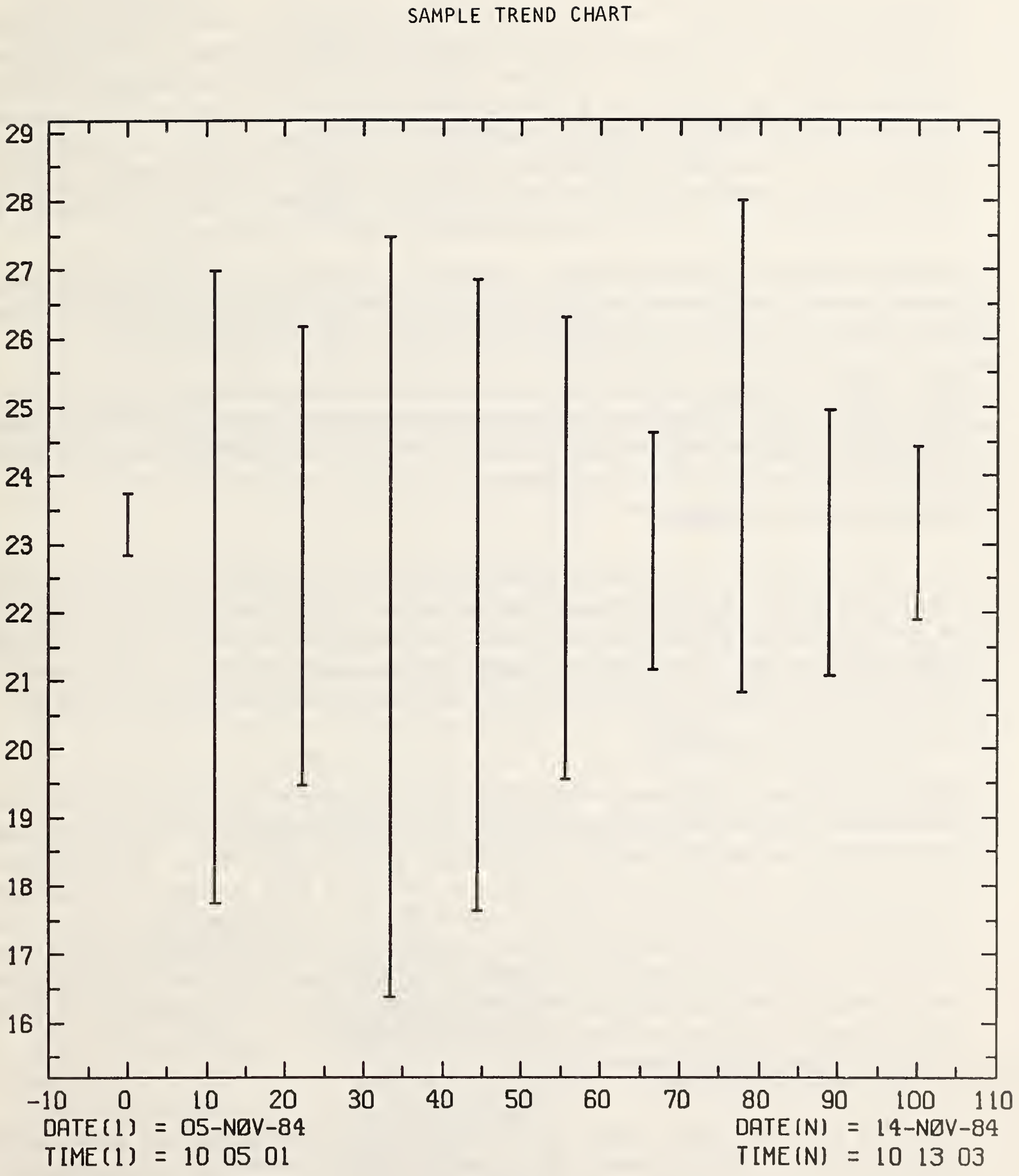

Figure 4. Sample trend chart.

be placed above the plot. The user may enter a title or simply a carriage return if no title is desired. Note that the prompt for the title always appears, whereas in the case of other maps or plots it appears only on request. Note also that TRC offers the user no control over the size of the display. An 8 × 8 inch plot is produced in all cases. It is important that test data be written to the data base in a timely manner, so that the date and time associated with each entry are close to the date and time of the actual data acquisition. It is also important that outlier exclusion be done in a consistent manner through all data sets. This is to ensure that the bars on the plot all have the same meaning.

As with VMP and SCA, when TRC is executed, the map goes to a metacode file which must be processed by the metacode translator MCVAX supplied by NCAR [6]. Some information about each entry plotted is also written to the STAT2.LOG file.

The TRC command cannot properly process entries having all zeroes in the four new fields of the extended data base. Such entries are created when the UPDATE3 procedure described in Appendices I through III is used. The TRC command specification must be such that any such entries are passed over.

9. THE XBN COMMAND

The XBN command (for exclude between) allows a user to exclude all sites having data values between two specified values. The form of the command is XBN,P1,P2, where P1 is a lower bound and P2 is an upper bound. Sites having data values out to and including these bounds are excluded. The case in which P1 is equal to P2 allows a user to exclude sites having a particular data value. If P2 is less than P1, an error message is printed.

10. THE MUL COMMAND

The MUL command allows a user to multiply the data values of all sites by a constant. The form of the command is MUL,P1, where P1 is a constant expressed in I, F, or E format. It is useful for multiplying data by a scale factor such as a power of ten. MUL cannot be executed unless a data set has first been input using the REA command. Replacement values should be calculated for outliers using AXP prior to invoking MUL to reduce the likelihood of floating-point overflow. This command alters the DATA array. It also resets STAFLG, meaning that after executing MUL, any statistics which have been calculated are no longer current and must be recalculated using PRS. If it is necessary to undo the MUL operation, the data set should be re-read using REA.

11. THE LGT COMMAND

The LGT command allows a user to take the common logarithm (log to base 10) of the absolute value of all data values. By convention, LGT operating on a data value of 0 converts it to −40. The form of the command is simply LGT. It takes no parameters. It is useful for producing logarithmic plots or histograms. For example, to produce a map of a data set having values which vary over a range from 1.0E-12 to 1.0E-04, the following command sequence could be used.

```
ASG FILE.DAT          ! Specify the data file.
REA 3 20              ! Read in a data set.
XLT 0                 ! Identify and
XGT 1E-03             !    exclude outliers.
AXP                   ! Calculate replacement values for excluded sites.
LGT                   ! Take log to base ten of all data values.
MP4 6 0 0 0 −12 −4    ! Make map with scale from 1E-12 to 1E-4.
END                   ! End of this example.
```

As with the MUL command, LGT cannot be executed unless a data set has first been input using the REA command, and LGT should be invoked after replacement values have been calculated by AXP to reduce the likelihood of floating-point overflow. Note that AXP, which does linear interpolation and extrapolation, does not give very good replacement values when data values vary over several orders of magnitude. This command alters the DATA array. Unlike MUL, LGT does not reset STAFLG. Any statistical values printed on map keys or written in the data base relate to the data prior to invoking LGT. Calculating statistics (using PRS) after invoking LGT would not usually be meaningful. If it is necessary to undo the LGT operation, the data set should be re-read using REA.

12. EXTENDED MAC COMMAND CAPABILITY

Under Version 1.31 of STAT2, it was illegal for a macro command file to contain a MAC command which would invoke another macro command file. This restriction has been removed in Version 1.7. Macro command files may call other such files to a nesting depth of five. Logical units 21 through 25 are used for the five levels of macro files. Those commands which have one or two lines of input which follow the command may have the command in the called macro file and the line of input in the calling macro file. The commands which have additional lines of input are the maps VMP, SCA, TRC, and formats 4 and 5 of REA. An example of how this might be used is illustrated in the two files below, where MAC1.MAC calls MAC2.MAC. Together, they process three data sets from a single input file.

```
* This is file MAC1.MAC
ASG FILE.EXT          ! Specify the data file.
REA 3 1               ! Read first data set.
MAC MAC2.MAC          ! Call MAC2.MAC.
MAP OF DATA SET 1     ! Map label for MP4 command in MAC2.MAC.
REA 3 2               ! Read second data set.
MAC MAC2.MAC          ! Call MAC2.MAC.
MAP OF DATA SET 2     ! Map label.
REA 3 3               ! Read third data set.
MAC MAC2.MAC          ! Call MAC2.MAC.
MAP OF DATA SET 3     ! Map label.
END                   ! End of this example.
```

```
* This is file MAC2.MAC
XGT 1E20                    ! This file contains STAT2 commands
PRS                         !    to be applied to each of the
XOL 0.2                     !    three data sets.
AXP                         !
MP4 6 1 1 0 0 0             ! The map label is supplied in MAC1.MAC.
```

It is no longer necessary to include the extension in the file specifier of the MAC command if the extension is '.MAC'. For example, 'MAC FILE' is interpreted as 'MAC FILE.MAC', 'MAC [DIR.SUB]FILE' is interpreted as 'MAC [DIR.SUB]FILE.MAC', and 'MAC FILE;14' is interpreted as 'MAC FILE.MAC;14'.

13. The PAG COMMAND

A PAG command has been added under Version 1.7 which when invoked issues a form feed to the STAT2.LOG file, such that subsequent printing appears at the top of a new page. The format of the command is simply PAG with no parameters.

14. The LOG COMMAND

A LOG command has been added under Version 1.7 which permits the user to log interactive commands to a file. The intent is to allow the user to later repeat the command sequence simply by calling the command file using the MAC command. The format of the command is LOG,P1. When P1 is nonzero, the file ST2COM.MAC is opened in the default directory on logical unit 15. All interactive input is logged including those commands which produce errors, commands entered following the OPN command, comments, and the second or third line of input for those commands which require it. (Because commands which produce errors are also logged, it may be necessary to edit this file before invoking it.) When P1 is zero, any log file which is in effect is closed and logging of interactive commands is terminated. Commands executed from within a macro command file are not logged since they are not entered interactively.

The operation of the LOG command is illustrated by the following command sequence:

```
* Begin example. These two lines and the LOG
* command which follows them are not logged.
LOG 1
* These six lines, including the 'LOG 0'
* command are logged to ST2COM.MAC.
LDB ALL
GET 14
SDB WAF 7
LOG 0
* This line and the next are not logged.
END
```

15. CONTROL-C INTERRUPT

The capability has been added for the user to type control-C to interrupt execution of some commands and return to the STAT2> prompt. The commands for which this feature has been implemented are LAP, LDB, LEN, SDB, and TRC; that is, the five commands which potentially may produce long listings which the user may want to abort without terminating the program. This feature utilizes system service calls which are peculiar to VAX/VMS and would have to be modified for other computing environments.

When a control-C is typed, a flag is set which is read periodically by each of the five subroutines mentioned above. On the next occasion that the flag is read, control transfers back to the command processor and the STAT2> prompt is issued. This may not happen immediately if LDB or LEN have asked that data values also be printed, but it will happen following the reading of the next label record. For commands other than the five mentioned, a "∧C" is echoed to the terminal and execution continues with possible loss of the remainder of the current line of output.

16. HELP FACILITY REVISIONS

The HELP command now makes use of the VMS help facility. There is now a single help library rather than 50 or so individual files. This makes installation easier and makes it easier for a user to peruse the help messages. The disadvantage is that this is a feature not directly portable to computers not running VMS. The help library must have the name STAT2.HLB and be stored in the directory SYS$HELP.

17. OTHER CHANGES

17.1. Informational Messages For Outlier Exclusions

Under previous versions, when one of the outlier exclusion commands was entered, the user was not told whether any sites were excluded as a result of issuing the command. Under Version 1.7, a message is printed every time an outlier exclusion command is entered telling how many sites were excluded. The XIP command is an exception to this. When this command produces no message, it means that the specified site was excluded. A warning message is produced by XIP if the site was already excluded or if no such site existed. These messages may be disabled and enabled using the ERM command.

17.2. Error Fixed in ASG Command

An error was corrected in subroutine ASG so that execution of the ASG command now makes it necessary to issue REA before performing any operations on input data. Before the correction, data from the previous input file were still available even after the new file had been assigned.

17.3. Error Fixed in Mapping Commands

Corrections were made in subroutine GRSCAL to prevent overflow when making a map in instances where some excluded sites had not had replacement values substituted for the original data values.

17.4. Overflow Sensed in Subroutine PRS

Under previous versions, the accumulation of sums and sums of squares of data values which occurs in response to the PRS command would cause STAT2 to terminate abnormally when one or more included sites had data values of sufficiently large magnitude. Under Version 1.7, an included site having a data value greater in magnitude than 1.0E+15 causes an OVERFLOW ERROR message to be printed. The user should then either exclude offending sites using the XGT and XLT commands or scale down all data values using the MUL command. If there are a large number of included sites having data values near the 1.0E+15 limit, overflow may still occur and cause abnormal program termination.

17.5. Number of Lines Per Page of STAT2.LOG Changed

The number of lines printed per page of the STAT2.LOG file was reduced from 62 to 60 to reduce crowding. This may affect maps made with the MP1 command in situations where the map occupies the entire vertical page distance. Also, under VAX/VMS 4.0 and higher, the STAT2.LOG file should be printed using the /NOFEED qualifier, that is, PRINT/NOFEED STAT2.LOG.

18. MODIFIED CRDB PROGRAM

The CRDB program, used to create a data base, has been changed. When the program is run, the user is asked to enter the filename of the data base file or files being created. Two files are created for all sampling plans except sampling plan 5 for which only one is created. The data base file or files are then created in the current default directory with .LAB and .DAT extensions for the label and data files, respectively. As mentioned previously, the record length of the label file is now 132 rather than the former 80 bytes. The user no longer needs to rename the files after CRDB is run. The files must still be assigned prior to running STAT2, however. A sample run of the modified CRDB program is shown in figure 5.

19. MODIFIED CORTABLE PROGRAM

The CORTABLE program has been modified to allow let it to read and print the information in the new label file fields. It also reports an error if an attempt is made to process a file having sampling plan 5.

20. DATA BASE EDITOR

There are times when it is desirable to edit one of the data base files, such as to undelete an entry or to change one of the key fields (PAT, LOT, WAF, DEV, and PCODE) associated with an entry. Because the data base files are direct access, many text editors cannot manipulate them properly.

DBEDIT is a VMS command procedure which provides for editing of these files. Essentially, it takes a direct access file, creates an associated sequential access file, edits the

```
$ RUN CRDB                               ! Invoke the program.
ENTER FILENAME ONLY OF DATA BASE FILES   ! Program prompts
    (9 CHARACTERS OR LESS)               !   for filename.
DEMO                                     ! Enter filename.
ENTER THE DATA BASE TITLE (9A4)          ! Program prompts for title.
DATA BASE TO DEMONSTRATE CRDB RUN        ! Enter title.
ENTER THE SAMPLING PLAN CODE IN FREE FORMAT  ! Prompting message.
4                                        ! Enter sampling plan code.
!
! The first record of the label file and the first two
!   records of the data file are then printed out.
!
          DATA BASE TO DEMONSTRATE CRDB RUN      4     1      2
T  DATA BASE TO DEMONST
T    RATE CRDB RUN
FORTRAN STOP
$ DIR DEMO.*      ! This is not part of the CRDB run procedure but only
                  !   shows the names of the two files which are created.
Directory DRB1:[MATTIS]

DEMO.DAT;1          DEMO.LAB;1

Total of 2 files.
$
```

Figure 5. Sample run of the CRDB program. (Text which follows an exclamation point and lines beginning with an exclamation point are explanatory and are not a part of the procedure.)

sequential access file, then creates a new direct access file from the new sequential access file. A data base label file to be edited must have an .LAB extension, and a data base data file must have a .DAT extension.

The data base editor consists of DBEDIT.COM and four FORTRAN programs. Instructions for using DBEDIT are contained in comment statements at the top of DBEDIT.COM which is listed in Appendix IV. The four FORTRAN programs are listed in Appendices V through VIII.

21. MESSAGES

An alphabetical list of messages which are present in Version 1.7 but were not present in earlier versions follows. As in previous versions, messages beginning with three asterisks are error messages. Messages beginning with three exclamation points may be either warning messages or simply informational messages. Most warning and informational messages may be disabled or enabled by the ERM command. Lower case letters in the messages represent alphanumeric characters which would appear in the actual messages. Messages which have an altered form when presenting coordinate information for nonperiodic sites but which do not otherwise differ from the messages in previous versions are not included.

! ! ! AMEAN = 0, PSIGMA SET TO 0

In PRS, the sample mean has been found to be 0. To avoid floating point overflow, the percent standard deviation has been set to 0.

*** CANNOT ADD .MAC EXTENSION TO MACRO FILE

When a .MAC extension is added to the file specifier of the MAC command, the file specifier will be more than the 66-character limit on length.

*** COEF KEYWORD ILLEGAL

A COEF keyword has been used in an LDB or TRC command.

*** DENOM IS 0.0 IN SLOPE CALCULATION

In attempting to produce a scatter plot, the slope of the fitting line was found to be infinite.

! ! ! EXITING filename

An informational message to indicate that a macro command file is being exited.

*** GET AND SCA ILLEGAL WITH SAMPLING PLAN 5

Because a data base made using sampling plan 5 has no data file, the GET and SCA commands have no meaning and are illegal.

! ! ! HELP LIBRARY FILE COULD NOT BE ACCESSED

The help library was not found in SYS$HELP:STAT2.HLB or could not be accessed such as due to file protection.

! ! ! LOGGING IS ALREADY {ENABLED, DISABLED}

A LOG command has been issued to enable (disable) interactive command logging, but such logging had already been enabled (disabled).

*** MATCHES <= 1, THEREFORE NO TREND CHART

In processing the TRC command, fewer than two data base entries satisfy the requirements of the command. No trend chart can be produced.

*** MAXIMUM 256 MATCHING ENTRIES EXCEEDED

In processing the TRC command, more than the allowed 256 data base entries were found which matched the TRC command specification. All matching entries may be listed using the LDB command with the same command specification as that of the TRC command.

*** NDATX AND NDATY DO NOT AGREE

When making a scatter plot, the NDAT fields of the two entries being plotted have been found to be unequal. The NDAT field contains the number of data points in the sample stored in the data base.

*** NESTING DEPTH LIMIT OF MAC COMMAND EXCEEDED

An attempt has been made to call a macro command file beyond the maximum allowed nesting depth of five.

! ! ! NO DATA VALUES IN SAMPLING PLAN 5

When an LDB or LEN command is entered such as to request printing of data values in a data base made using sampling plan 5, the command is changed so that data values are not requested and this warning message is produced.

*** NUMBER OF INCLUDED SITES EXCEEDS 25

When using the XSS command, the number of included sites must be in the range 3 to 25, inclusive.

*** NUMBER OF INCLUDED SITES IS LESS THAN 3

When using the XSS command, the number of included sites must be in the range 3 to 25, inclusive.

*** NUMBER OF POINTS LESS THAN 3

In attempting to produce a scatter plot, the number of included sites common to the two data sets being plotted is less than 3.

*** OVERFLOW ERROR

An included site has a data value with a magnitude greater than 1.0E+15. Either exclude all such sites using the XGT and XLT commands or scale the data down using the MUL command.

*** P1 MUST BE .6, .4, .2, .1, .04, .02, OR .01

When using the XSS command, the parameter P1 may have only the specified values.

*** PAT, LOT, WAF OR DEV ARE NOT IN AGREEMENT

When making a vector map, the respective PAT, LOT, WAF, and DEV fields of the two entries being plotted must be equal.

*** POS STATEMENT NOT FOUND AFTER SITE i

When using the REA command with format 6, a POS statement was not found at its expected record location. One or more records have been added or removed between POS statements. This is a check on the integrity of the .LGA file.

! ! ! PROGRAM EXECUTION INTERRUPTED

An informational message which appears when control-C is hit while LAP, LDB, LEN, SDB, or TRC is executing. In these five subroutines, control-C aborts the listing and returns the STAT2> prompt.

*** SAMPLING PLAN CODE MUST EQUAL 4

When making a vector map, the sampling plan code of the data base being used must equal 4.

! ! ! i SITE(S) EXCLUDED BY {XNS, XPP, XLT, XGT, XBN, XOL}

When any of the outlier exclusion commands contained in the braces is entered, this informational message indicates how many sites were excluded as a result of that particular command.

*** TOO MANY DATA VALUES

A format 6 data file contains data from more than 1024 sites.

*** UNEXPECTED EOF IN .LGA FILE

When reading a data file using format 6, an end-of-file was encountered before reading a data value associated with the last POS statement. The data file is probably incomplete.

*** UPPER BOUND < LOWER BOUND

In an XBN command, the upper bound, P2, must not be less than the lower bound, P1.

*** VERINT CALLED IN FORMAT 6

This message should not appear. Its occurrence indicates a program logic error.

*** WIDTH AND HEIGHT MUST BE BETWEEN 3 AND 8

When specifying the size of a scatter plot, the width and height must each be in the range 3 to 8 inches.

ACKNOWLEDGMENTS

The author wishes to thank Ruth Zucker for coding and testing of the vector map, scatter plot, and trend chart commands, Charles P. Reeve for helpful suggestions regarding the identification of outliers in the case of small sample sizes, Thomas J. Russell for supplying the data used in the vector map example, Vance Tyree of Information Sciences Institute for supplying the data used in the scatter plot example, Colleen H. Ellenwood for reviewing this document and the changes in the program, Loren W. Linholm and John S. Suehle for helpful suggestions regarding this document, and E. Jane Walters for manuscript preparation.

Portions of this work were supported by the Defense Advanced Research Projects Agency (ARPA Order No. 3882).

REFERENCES

1. Mattis, R. L., Semiconductor Measurement Technology: A FORTRAN Program for Analysis of Data from Microelectronic Test Structures, NBS Special Publication 400-75 (July 1983).

2. Mattis, R. L., and Zucker, R., Release Notes for STAT2 Version 1.31: An Addendum to NBS Special Publication 400-75, NBSIR 83-2779 (November 1983).

3. Natrella, M. G., Experimental Statistics, NBS Handbook 91, pp. 17-3 and T-27 (Reprinted October 1966).

4. Beckman, R. J., and Cook, R. D., Outliers, Technometrics 25 (2), 119-163 (May 1983).

5. Russell, T. J., Leedy, T. F., and Mattis, R. L., A Comparison of Electrical and Visual Alignment Test Structures for Evaluating Photomask Alignment in Integrated Circuit Manufacturing, Technical Digest, International Electron Devices Meeting, Washington, D.C., December 5-7, 1977, pp. 7A-7F.

6. The SCD Graphics Utilities, Technical Note NCAR-TN/166+IA, G. R. McArthur, ed., Scientific Computing Division, National Center for Atmospheric Research, Boulder, CO. For information on availability, contact Sue Long, Software Distribution, NCAR/SCD, P. O. Box 3000, Boulder, CO 80307, telephone (303) 497-1245.

Appendix I. UPDATE3.FOR

```
C
C     PROGRAM UPDATE3
C
C     UPDATES A STAT2 DATA BASE LABEL FILE FROM STAT2 REVISION 1.31 TO
C     REVISION 1.70.  THE DIFFERENCE IS THAT THE RECORDSIZE OF THE LABEL
C     FILE HAS BEEN INCREASED FROM 80 TO 132 BYTES TO MAKE ROOM FOR
C     THE NTOTAL, NGOOD, MEAN AND STDDEV FIELDS.  PREVIOUSLY EXISTING
C     FIELDS ARE NOT ALTERED.
C
      BYTE DATE(9),TIME(8)
      INTEGER TITLE(9)
      INTEGER ENT,DLT,PTR,PAT,WAF,DEV,PCODE
C
C     OPEN OLD LABEL FILE.
C
      OPEN (UNIT=27,STATUS='OLD',ERR=10,ACCESS='DIRECT',
     * FORM='FORMATTED',ORGANIZATION='RELATIVE',SHARED,
     * RECL=80,RECORDTYPE='FIXED')
C
C     OPEN NEW LABEL FILE.
C
      OPEN (UNIT=37,STATUS='NEW',ERR=20,ACCESS='DIRECT',
     * FORM='FORMATTED',ORGANIZATION='RELATIVE',SHARED,
     * RECL=132,RECORDTYPE='FIXED',MAXREC=99999)
C
C     DEFINE DEFAULT VALUES FOR NEW FIELDS.
C
      NTOTAL=0
      NGOOD=0
      DMEAN=0.0
      STDDEV=0.0
C
C     COPY FIRST RECORD.
C
      READ (27'1,1) TITLE,ISPC,LASTLR,LASTDR
      WRITE (37'1,1) TITLE,ISPC,LASTLR,LASTDR
1     FORMAT (10X,9A4,2X,I3,1X,I5,1X,I6)
C
C     COPY LABEL RECORDS UNTIL END-OF-FILE ENCOUNTERED.
C
      LREC=2
2     READ (27'LREC,3,ERR=50) ENT,DLT,PTR,NDAT,PAT,LOT,WAF,DEV,PCODE,
     * DATE,TIME
3     FORMAT (1X,I5,A1,I6,I4,3X,I3,I3,I3,I4,I4,2X,9A1,1X,8A1)
      WRITE (37'LREC,4) ENT,DLT,PTR,NDAT,PAT,LOT,WAF,DEV,PCODE,
     * DATE,TIME,NTOTAL,NGOOD,DMEAN,STDDEV
4     FORMAT (1X,I5,A1,I6,I4,3X,I3,I3,I3,I4,I4,2X,9A1,1X,8A1,
     * 1X,I4,I4,E13.6,E13.6)
      LREC=LREC+1
      GOTO 2
C
C     PRINT MESSAGE FOR TERMINATION.
C
```

```
50    LREC=LREC-1
      WRITE (6,51) LREC
51    FORMAT (1X,'PROGRAM TERMINATION -- ',I5,' RECORDS CONVERTED')
      GOTO 60
C
C     IF ERROR OPENING FILES.
C
10    WRITE (6,11)
11    FORMAT (1X,'*** ERROR OPENING OLD LABEL FILE')
      GOTO 60
20    WRITE (6,21)
21    FORMAT (1X,'*** ERROR OPENING NEW LABEL FILE')
60    STOP
      END
```

Appendix II. UPDATE3.COM

```
$ !
$ ! File UPDATE3.COM
$ !
$ ASSIGN 'P1' FOR027
$ ASSIGN 'P1' FOR037
$ RUN UPDATE3
```

Appendix III. UPDATE3.HLP

File UPDATE3.HLP

UPDATE3 is a Fortran program and VMS command file to update a STAT2 data base label file from STAT2 version 1.31 to STAT2 version 1.70. The difference is that the recordsize of the data base label file has been increased from 80 to 132 to accommodate the NTOTAL, NGOOD, MEAN and STDDEV which were included in the data base label file beginning with version 1.70. The data base data file is not affected by going to the higher version.

To execute the program, type:

```
@UPDATE3 file-specifier
```

where 'file-specifier' is the file specifier (including directory if not the default directory, and including the extension) of the data base label file. The program reads the old label file and writes a new label file having the same file specifier but a version number one greater than the old label file. When the program is finished, it prints a message giving the number of entries converted and exits. The user should verify that this number is correct by comparing it with the LASTLR field of the first label record. Once this verification is done, the old label file may be deleted by purging.

If an error occurs in trying to open either of the label files, a message is printed as

```
*** ERROR OPENING OLD LABEL FILE    or
*** ERROR OPENING NEW LABEL FILE    .
```

```
$ !
$ ! File DBEDIT.COM
$ !
$ ! Modified for version 1.7000 of STAT2.
$ !
$ ! A command file to edit a data base label file or data base data file.
$ ! The command file copies the data base file into a sequential file,
$ ! the sequential file is edited using EDT, then the sequential file is
$ ! converted into a new direct access data base file.
$ !
$ ! This command procedure assumes that LD2S.EXE, LS2D.EXE, DD2S.EXE
$ ! and DS2D.EXE are in a directory named [USERLIB].  To install
$ ! this procedure:
$ !
$ ! - Change all instances of [USERLIB] to an appropriate [directory] to
$ !   contain DBEDIT.COM and the four .EXE files given above.
$ !
$ ! - Copy DBEDIT.COM and the four executables into [directory].
$ !
$ ! - Define the symbol DBEDIT:==@[directory]DBEDIT  .
$ !
$ ! To use DBEDIT to edit a data base label or data file:
$ !
$ ! - $ DBEDIT filename.LAB   or
$ ! - $ DBEDIT filename.DAT
$ !
$ ON CONTROL_Y THEN GOTO DONE
$ ON ERROR THEN GOTO DONE
$ !
$ FSPEC == P1
$ !
$ ! Locate and identify the extension of the file.  The extension follows
$ ! a period, but it must first be determined that any periods within
$ ! brackets are eliminated from consideration.
$ !
$ LENGTH = 'F$LENGTH(FSPEC)'
$ BRACKET = 'F$LOCATE("]",FSPEC)'
$ !
$ ! If BRACKET and LENGTH are equal, then there are no brackets in the
$ ! FSPEC string.  Then the next section which separates the directory
$ ! from the rest of the file specifier can be skipped.
$ !
$ IF BRACKET .EQ. LENGTH THEN GOTO NODIRECT
$ !
$ ! Extract the substring following the bracket.
$ !
$ LENSUB = LENGTH - BRACKET
$ START = BRACKET + 1
$ SUBSTR := 'F$EXTRACT(START,LENSUB,FSPEC)'
$ DOT = 'F$LOCATE(".",SUBSTR)'
$ DOTP1 = DOT + 1
$ EXT := 'F$EXTRACT(DOTP1,3,SUBSTR)'
$ GOTO CHKEXT
$ !
$ NODIRECT:
$ DOT = ’F$LOCATE(".",FSPEC)’
$ DOTP1 = DOT + 1
$ EXT := ’F$EXTRACT(DOTP1,3,FSPEC)’
$ !
$ CHKEXT:
$ !
$ ! If the extension is LAB then we are editing a label file.  If it is
$ ! DAT then we are editing a data file.  All other extensions are illegal.
$ !
$ FLAG := "ERROR"
$ IF EXT .EQS. "LAB" THEN FLAG := "LABEL"
$ IF EXT .EQS. "DAT" THEN FLAG := "DATA"
$ IF FLAG .NES. "ERROR" THEN GOTO MAKSEQ
$ WRITE SYS$OUTPUT "File to be edited must have .LAB or .DAT extension."
$ EXIT
$ !
$ MAKSEQ:
$ !
$ ! Create a sequential access file from the direct access file.
$ !
$ ASSIGN ’FSPEC’ FOR011
$ IF FLAG .EQS. "LABEL" THEN RUN [USERLIB]LD2S
$ IF FLAG .EQS. "DATA"  THEN RUN [USERLIB]DD2S
$ !
$ ! Use EDT to edit the sequential file.
$ !
$ ASSIGN/USER_MODE SYS$COMMAND: SYS$INPUT:
$ EDIT/EDT FOR021.DAT
$ !
$ ! Now make a direct access file from the sequential file.
$ !
$ IF FLAG .EQS. "LABEL" THEN RUN [USERLIB]LS2D
$ IF FLAG .EQS. "DATA"  THEN RUN [USERLIB]DS2D
$ RENAME FOR031.DAT ’FSPEC’
$ !
$ ! Purge old data base file on request and delete FOR021.DAT files.
$ !
$ INQUIRE PURG "Do you want to purge old versions of "’FSPEC’"?"
$ ANSWER := ’F$EXTRACT(0,1,PURG)’
$ IF (ANSWER .EQS. "Y") .OR. (ANSWER .EQS. "y") THEN PURGE ’FSPEC’
$ DELETE FOR021.DAT;*
$ !
$ DONE:
$ EXIT
```

Appendix V. LD2S.FOR

```
C
C     PROGRAM LD2S
C
C     A PROGRAM TO CONVERT A LABEL FILE FROM DIRECT TO SEQUENTIAL ACCESS.
C
      BYTE LINE(132)
      OPEN (UNIT=11,STATUS=’OLD’,ACCESS=’DIRECT’,FORM=’FORMATTED’,
     * ORGANIZATION=’RELATIVE’,RECL=132,RECORDTYPE=’FIXED’)
      OPEN (UNIT=21,STATUS=’NEW’)
      IREC=1
10    READ (11’IREC,20,ERR=60) LINE
20    FORMAT (132A1)
      WRITE (21,20) LINE
      IREC=IREC+1
      GOTO 10
C
60    STOP
      END
```

Appendix VI. LS2D.FOR

```
C
C     PROGRAM LS2D
C
C     A PROGRAM TO CONVERT A LABEL FILE FROM SEQUENTIAL TO DIRECT ACCESS.
C
      BYTE LINE(132)
      OPEN (UNIT=21,STATUS='OLD')
      OPEN (UNIT=31,STATUS='NEW',ACCESS='DIRECT',FORM='FORMATTED',
     * MAXREC=99999,
     * ORGANIZATION='RELATIVE',RECL=132,RECORDTYPE='FIXED')
      IREC=1
10    READ (21,20,ERR=60) LINE
20    FORMAT (132A1)
      WRITE (31'IREC,20) LINE
      IREC=IREC+1
      GOTO 10
C
60    STOP
      END
```

Appendix VII. DD2S.FOR

```
C
C     PROGRAM DD2S
C
C     A PROGRAM TO CONVERT A DATA FILE FROM DIRECT TO SEQUENTIAL ACCESS.
C
      BYTE LINE(24)
      OPEN (UNIT=11,STATUS='OLD',ACCESS='DIRECT',FORM='FORMATTED',
     * ORGANIZATION='RELATIVE',RECL=24,RECORDTYPE='FIXED')
      OPEN (UNIT=21,STATUS='NEW')
      IREC=1
10    READ (11'IREC,20,ERR=60) LINE
20    FORMAT (24A1)
      WRITE (21,20) LINE
      IREC=IREC+1
      GOTO 10
C
60    STOP
      END
```

Appendix VIII. DS2D.FOR

```
C
C     PROGRAM DS2D
C
C     A PROGRAM TO CONVERT A DATA FILE FROM SEQUENTIAL TO DIRECT ACCESS.
C
      BYTE LINE(24)
      OPEN (UNIT=31,STATUS='NEW',ACCESS='DIRECT',FORM='FORMATTED',
     * ORGANIZATION='RELATIVE',RECL=24,RECORDTYPE='FIXED')
      OPEN (UNIT=21,STATUS='OLD')
      IREC=1
10    READ (21,20,ERR=60) LINE
20    FORMAT (24A1)
      WRITE (31'IREC,20) LINE
      IREC=IREC+1
      GOTO 10
C
60    STOP
      END
```

NBS-114A (REV. 2-8C)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET *(See instructions)*	1. PUBLICATION OR REPORT NO. NBSIR-86/3333	2. Performing Organ. Report No.	3. Publication Date March 1986

4. TITLE AND SUBTITLE

Release Notes for STAT2 Version 1.7: An Addendum to NBS Special Publication 400-75

5. AUTHOR(S)

R. L. Mattis

6. PERFORMING ORGANIZATION *(If joint or other than NBS, see instructions)*

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

7. Contract/Grant No.

8. Type of Report & Period Covered

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS *(Street, City, State, ZIP)*

National Bureau of Standards
Department of Commerce
Gaithersburg, MD 20899

Defense Advanced Research Projects Agency
Arlington, VA 22209

10. SUPPLEMENTARY NOTES

□ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

11. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)*

This document describes the changes which have been made in the STAT2 computer program since its documentation in NBS Special Publication 400-75, Semiconductor Measurement Technology: A FORTRAN Program for Analysis of Data from Microelectronic Test Structures, and NBS Internal Report 83-2779, Release Notes for STAT2 Version 1.31. It is assumed that the reader has these documents, and no attempt is made to review STAT2 features or operation. The changes extend the functionality and versatility of the program. More specifically, the new features added in version 1.7 include data base extension, an input data format suitable for test sites not in a periodic array, an outlier exclusion algorithm suitable for small numbers of sites, common site exclusions for related data sets, a vector map, a scatter plot, a trend chart, extended macro command file capability, and other changes. Following the description of the changes is an annotated listing of new error messages. This document and the two previous publications cited constitute the documentation of version 1.7 of STAT2.

12. KEY WORDS *(Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)*

ATE; computer program; contour map; data base; outlier; process validation wafer; scatter plot; trend chart; vector map; wafer map

13. AVAILABILITY

☒ Unlimited
□ For Official Distribution. Do Not Release to NTIS
□ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.
☒ Order From National Technical Information Service (NTIS), Springfield, VA. 22161

14. NO. OF PRINTED PAGES

41

15. Price

$9.95

FEDERAL INFORMATION PROCESSING STANDARD SOFTWARE SUMMARY

01. Summary date

Yr.	Mo.	Day
86	03	18

02. Summary prepared by (*Name and Phone*)
Richard L. Mattis (301) 921-3801

03. Summary action
New □ Replacement ☒ Deletion □
Previous Internal Software ID

04. Software date

Yr.	Mo.	Day
86	03	18

05. Software title
Release Notes for STAT2 Version 1.7: An Addendum to NBS Special Publication 400-75

06. Short title STAT2 Version 1.7

07. Internal Software ID

08. Software type
□ Automated Data System
☒ Computer Program
□ Subroutine/Module

09. Processing mode
☒ Interactive
□ Batch
□ Combination

10. Application area

General
□ Computer Systems Support/Utility
☒ Scientific/Engineering
□ Bibliographic/Textual
□ Management/Business
□ Process Control
□ Other

Specific
Analysis of Integrated Circuit Process Validation Wafer Test Data

11. Submitting organization and address

Semiconductor Electronics Division
National Bureau of Standards
Gaithersburg, MD 20899

12. Technical contact(s) and phone

Richard L. Mattis
(301) 921-3801

13. Narrative

The program performs the following functions: read data as a two-dimensional array in several formats; calculate mean, standard deviation, median; identify outliers; calculate replacement values for outliers; make gray tone, numerical, and contour data maps: make a histogram, vector map, scatter plot, and trend chart; construct a data base for making correlations among various data sets; search said data base using several selective keys. The package includes a help library callable from the program, sample input data, and sample output.

14. Keywords
ATE; computer program; contour map; data base; outlier; process validation wafer; scatter plot; trend chart; vector map; wafer map.

15. Computer manuf'r and model DEC VAX-11/785	16. Computer operating system VMS 4.2	17. Programing language(s) FORTRAN	18. Number of source program statements 16827
19. Computer memory requirements 832 KB	20. Tape drives None	21. Disk/Drum units Disk required	22. Terminals 1 video or hard copy

23. Other operational requirements

24. Software availability
Available ☒ Limited □ In-house only □

25. Documentation availability
Available ☒ Inadequate □ In-house only □

The user's manual is available and consists of NBS Spec. Publ. 400-75, NBSIR 83-2779, and NBSIR 86-3333.

26. FOR SUBMITTING ORGANIZATION USE

INSTRUCTIONS

01. Summary Date. Enter date summary prepared. Use Year, Month, Day format: YYMMDD.

02. Summary Prepared By. Enter name and phone number (including area code) of individual who prepared this summary.

03. Summary Action. Mark the appropriate box for new summary, replacement summary or deletion of summary. If this software summary is a replacement, enter under "Previous Internal Software ID" the internal software identification as reported in item 07 of the original summary, and enter the new internal software identification in item 07 of this form; complete all other items as for a new summary. If a software summary is to be deleted, enter under "Previous Internal Software ID" the internal software identification as reported in item 07 of the original summary; complete only items 01, 02, 03 and 11 on this form.

04. Software Date. Enter date software was completed or last updated. Use Year, Month, Day format: YYMMDD.

05. Software Title. Make title as descriptive as possible.

06. Short Title. (Optional) Enter commonly used abbreviation or acronym which identifies the software.

07. Internal Software ID. Enter a unique identification number or code.

08. Software Type. Mark the appropriate box for an **Automated Data System** (set of computer programs), **Computer Program**, or **Subroutine/Module**, whichever best describes the software.

09. Processing Mode. Mark the appropriate box for an **Interactive**, **Batch**, or **Combination** mode, whichever best describes the software.

10. Application Area.
General: Mark the appropriate box which best describes the general area of application from among:
Computer Systems Support/Utility
Management/Business
Scientific/Engineering
Process Control
Bibliographic/Textual
Other
Specific: Specify the sub-area of application; e.g.: "COBOL optimizer" if the general area is "Computer Systems Support/Utility"; "Payroll" if the general area is "Management/Business"; etc. Elaborate here if the general area is "Other."

11. Submitting Organization and Address. Identify the organization responsible for the software as completely as possible, to the Branch or Division level, but including Agency, Department (Bureau/Administration), Service, Corporation, Commission, or Council. Fill in complete mailing address, including mail code, street address, city, state, and ZIP code.

12. Technical Contact(s) and Phone: Enter person(s) or office(s) to be contacted for technical information on subject matter and/or operational aspects of software. Include telephone area code. Provide organization name and mailing address, if different from that in item 11.

13. Narrative. Describe concisely the problem addressed and methods of solution. Include significant factors such as special operating system modifications, security concerns, relationships to other software, input and output media, virtual memory requirements, and unique hardware features. Cite references, if appropriate.

14. Keywords. List significant words or phrases which reflect the functions, applications and features of the software. Separate entries with semicolons.

15. Computer Manufacturer and Model. Identify mainframe computer(s) on which software is operational.

16. Computer Operating System. Enter name, number, and release under which software is operating. Identify enhancements in the Narrative (item 13).

17. Programing Language(s). Identify the language(s) in which the software is written, including version; e.g., ANSI COBOL, FORTRAN V, SIMSCRIPT II.5, SLEUTH II.

18. Number of Source Program Statements. Include statements in this software, separate macros, called subroutines, etc.

19. Computer Memory Requirements. Enter minimum internal memory necessary to execute software, exclusive of memory required for the operating system. Specify words, bytes, characters, etc., and number of bits per unit. Identify virtual memory requirements in the Narrative (item 13).

20. Tape Drives. Identify number needed to operate software. Specify, if critical, manufacturer, model, tracks, recording density, etc.

21. Disk/Drum Units. Identify number and size (in same units as "Memory"—item 19) needed to operate software. Specify, if critical, manufacturer, model, etc.

22. Terminals. Identify number of terminals required. Specify, if critical, type, speed, character set, screen/line size, etc.

23 Other Operational Requirements. Identify peripheral devices, support software, or related equipment not indicated above, e.g., optical character devices, facsimile, computer-output microfilm, graphic plotters.

24. Software Availability. Mark the appropriate box which best describes the software availability from among: **Available to the Public**, **Limited Availability** (e.g.: for government use only), and **For-In-house Use Only**. If the software is "Available", include a mail or phone contact point, as well as the price and form in which the software is available, if possible.

25. Documentation Availability. Mark the appropriate box which best describes the documentation availability from among: **Available to the Public**, **Inadequate for Distribution**, and **For In-house Use Only**. If documentation is "Available", include a mail or phone contact point, as well as the price and form in which the documentation is available, if possible. If documentation is presently "Inadequate", show the expected availability date.

26. For Submitting Organization Use. This area is provided for the use of the organization submitting this summary. It may contain any information deemed useful for internal operation.

GPO : 1974 O - 544-655