DECUS
PROGRAM LIBRARY

DECUS NO.	8-132
TITLE	STRIP, A DATA DISPLAY AND ANALYSIS PROGRAM FOR THE PDP-8, 8/I
AUTHOR	John C. Alderman, Jr. Nuclear Research Center
COMPANY	Georgia Institute of Technology Atlanta, Georgia
DATE	July, 1968
SOURCE LANGUAGE	

STRIP, A DATA DISPLAY AND ANALYSIS PROGRAM FOR THE PDP-8, 8/I

DECUS Program Library Write-up DECUS No. 8-132

This program, using the PDP-8, high speed paper tape reader, and type 34 display, accepts paper tape data listings and displays the result on the display unit. Some elementary computations are made on the data and are also displayed. The program is deliberately designed to be open-ended, and most users will want to add features peculiar to their own problem. Almost all functions are carried out in subroutine form, and these subroutines can be called either from the keyboard or within another subroutine.

INTRODUCTION

At the Georgia Tech Nuclear Research Center there are in progress a number of small scale experiments, each involving several graduate students. All of these experiments use a data acquisition system which includes an on-line PDP-8. Our need is for a data processing system which will produce clearly interpretable results from the experiment in a relatively short period of time, since otherwise the apparatus may not be available for a repeat of the experiment.

Since most of the experiments take data as a function of some equal increments of an independent variable, a straightforward data display and reduction program has been devised for use with the type 34 display unit.

Two programming assumptions have been made:

(1) While computers are relatively good at doing computations, they are singularly unimaginative in making decisions; while graduate students may be capable of doing the computations, they are singularly unwilling to do so.

Consequently, the present version of STRIP depends on the computer for almost all of the calculations, and the user for all of the decisions.

(2) Any programming system which is to be used by several groups must be easily expanded in order to change and/or add functions to the original system. In the case of inexperienced programmers in particular, these changes and additions must be facilitated to the extent that the user can make the needed changes without spending a great deal of time learning the nuances of sophisticated assembly (PAL) language programming.

These considerations led to the development of STRIP, a PDP-8 program which produces a two-dimensional display with the independent (equal-increment) variable along the horizontal (X) axis, and the dependent variable along the vertical (Y) axis. Also included in the display is the result of some elementary numeric computations on the displayed data (i.e., the address of the maximum, its value, and the area under the displayed curve). These numbers can be used by the operator to determine parameters for later calculations.

In order to optimize data handling and display, two buffers are used. One contains the original data and the other data to be displayed. The display routine continuously circulates through the later, refreshing the display at a rate of about 20 times a second.

In the current STRIP version the operator/user manipulates the parameters of a calculated Guassian to fit his data. This is especially useful since many types of experimental data show such a Gaussian distribution, and the parametric form is desired for further data reduction. Since the fitting operation is accomplished by the user implicitly, the background does not have to be specified explicitly, simplifying the operation of obtaining the Gaussian parameters themselves.

Data Storage

The data for the program are stored in two buffers in the computer memory. The floating point data buffer contains each value of the original data stored in a 3 word floating decimal point format, as used by the standard Float Point Packages. These data are used as the basis for most of the computations, but are not disturbed by these computations (exceptions are the input routine "R" and the permanent Guassian subtraction routine "#"). The display buffer is stored in a 10 bit one-word integer format, suitable for deposition into the Y axis register of the type 34 display unit. The display routine cycles through this buffer displaying each point in turn while incrementing the horizontal axis by the appropriate horizontal step size.

A feature of the display routines is that as the display buffer is "built" by making computations on the data in the floating point buffer, the result is normalized before conversion to the 10 bit integer which is stored in the display buffer. Thus the display always occupies the maximum vertical displacement on the screen. The routine that calculates the data display also normalized the horizontal axis step size to make maximum use of the screen.

Keyboard Monitor

The keyboard monitor interprets the characters struck by the operator, and calls the corresponding subroutine from a table of starting addresses stored in page zero. The list of legal characters is expandable and terminated by a zero. The display routine is incorporated into the keyborad monitor flag test, such that the flag for the keyboard is tested after each loop through the display. The diplay is refreshed about 20 times a second (depending upon the number of points displayed). The most time-consuming operation of the display is the generation of the title, and a NOP can be inserted in the calling location for the titles subroutine, if desired.

The keyboard monitor presently recognizes a number of control characters which are listed as Table I. The functions are self-explanatory and the user will become familiar with them very quickly.

Usage

TABLE I

STRIP CONTROL KEYS

KEY	FUNCTION
L	Lower Boundary Marker
U	Upper Boundary Marker
C	Change to New Boundaries
F	Fetch Between Boundaries
D	Reset Boundaries
R	Read Input Data
S	Strip Trapezoid
J	Display Gaussian
G	Subtract Gaussian (display)
H	Get Gaussian Parameters
CTRL+ BELL	Permanent Upper Boundary
CTRL+ C	Return to Monitor (".")
#	Subtract Gaussian (data)

Let us assume that data has been entered into the data buffer (by using the R command), and that the shape of the observed peaks is a true Gaussian, obscured by noise. (See Figure 1). In order to begin with some reasonable values for the Gaussian parameters, let us narrow the limits by typing an:

L=+ 1 102

U=+ 160 150

C (See Figure 2)

Now we have narrowed the display to two peaks. Since the taller of the two peaks is the "MAX" on the display, and the endpoints of the display look as if they are on the flat portion of the background, we strike the "S" key. This causes the trapezoidal area between the zero reference and the value of the data at the absissa of the end points to be subtracted from the data. (See Figure 3). Notice that the display is renormalized to fill the screen. The new "AREA" and "MAX" are valid for the subtracted display. Notice that nothing has been done to the data in the "data buffer" (as you can discover by striking the "F" key, returning the display to its previous result by again hitting "S"). Now enter the subroutine that gets the Gaussian parameters by striking the "H" key. The program types out (in floating point E format) the current Full Width Half Maximum, and waits for a new value, or some non-numeric character. The standard deviation and the current value of the peak height are typed, and again the program waits for a new number. When the first non-numeric character is typed, the current value of the location of the peak (in units of channels, but not necessarily integer values of the channel number!) is typed and a new value accepted. When the next non-numeric character is typed, the area is computed and typed, and the program returns to the keyboard monitor. Note that there is no change in the display (See Figure 4).

In order to get some idea of the height of the fight hand peak, set the L limit to 127 temporarily, and expand the display with the C key (See Figure 4). Since the display is 11077 high, the right hand peak seems to be about 8000. The full width half maximum should be about 8.5, and the peak occurs at 133. Now strike the H key and enter those parameters:

```
H
FWHM=  +0.000000E+00 8.5  Sigma= +0.361162E+01
HEIGHT= +0.000000E+00  8000   AT  +0.000000E+00 133
AREA=  +0.724581E+05
```

In order to be able to observe the background, reset the L limit to 102. Now let's look at the Gaussian as it is generated in the program, by striking the J key. (See Figure 5). That seems to be pretty reasonable, so we subtract the curve in Figure 5 from that in Figure 2, and get Figure 6. The parameters entered seem to be good, but it might be possible to improve the "fit" if we moved the channel number .25 to the right.

```
H
FWHM=  +0.850000E+01   SIGMA=  +0.361162E+01
HEIGHT= +0.800000E+04 AT  +0.133000E+03  133.25
AREA=  +0.724581E05

F
G     (See Figure 7)
```

That doesn't look as good as the previous result. Maybe the width needs to be changed.

```
H
FWHM= +0.850000E+01 9  SIGMA=  +0.382407E+01
HEIGHT= +0.800000E+04  AT  +0.133250E+03  133
AREA= +0.767203E+05

F
G      (See Figure 8)
```

That looks better, let's make it even wider now.

```
H
FWHM=  +0.90000E+01  9.5  SIGMA= +0.403652E+01
HEIGHT= +0.800000E+04   AT  +0.133000E+03
AREA=  +0.809826E+05

F
G      (See Figure 9)
```

Much better. We are pretty close to the trees, so we can examine the forest better from a distance. To get the original full screen display, strike the D key.

D
G (See Figure 10)

From this viewpoint, it is obvious that the peak is a little too tall. Let's try 8500 for the HEIGHT parameter.

```
H
FWHM= +0.950000E+01 SIGMA= +0.403652E+01
HEIGHT= +0.800000E+04 8500 AT +0.133000E+03
AREA= +0.860440E+05
```

F
G (See Figure 11)

That's just a hair too much; try 8400.

```
H
FWHM= +0.950000E+01 SIGMA= +0.403652E+01
HEIGHT= +0.850000E+04 8400 AT +0.133000E+03
AREA= +0.850317E+05
```

F
G (See Figure 12)

That's pretty good. Perhaps you could better the "fit" by spending more time adjusting the parameters, but the improvement in the results would probably not warrant the effort. The differences in the last several moves are on the order of a few percent, and with data of this type, it probably isn't possible to do much better than that without using some sort of least squares technique.

Modification of STRIP

Let us suppose that a user has a requirement for a special routine to subtract a known background run from the current data field. Specifications for the subroutine might be:

Obtain a normalization factor from the operator/user and then read the data while point-by-point subtracting the product of the normalization factor times the input data from the resident spectrum and leaving the result in the resident spectrum.

The flow chart for this routine is Figure 13; the listing is Figure 14. The normalization factor is obtained by asking the operator for that number. The input routine is set up for reading from the high speed paper tape reader by depositing zero in location 56, then the DO pseudo-operation is used to call the initialization routine for the loop, after which the GET routine is used to get a number from the paper tape reader. The short computation in the floating point package substitutes the result of subtracting NORM times the just obtained number from the contents of the location pointed to by I1 (Location 105).

The CONT routine updates the pointers, and tests for the end of the loop. When the loop has been satisfied, the subroutine returns to the keyboard monitor for the next command (and restores location 56 to 7777 to enable keyboard input).

Notice that the program coding is relatively simple and that many functions are really calls to various subroutines, either in the Floating Point Package or the STRIP package.* One tricky point is that the user must be sure that the locations in the keyboard character and directory tables corresponds and do not interfere with other key-called functions active in the package. (See page 1 of the HULME routine* for additional keyboard called functions).

LOADING AND DEBUGGING USER-WRITTEN SUBROUTINES

The disc resident version of STRIP has some coding at 3600 which tests the switch register at load time and halts if SR=0. The user may now use the Middle of Core Loader (MOCL) at 3777, and/or the version of ODT (DEC-08-COC1-PA) at 1000. If ODT is to be used, the contents of location 445 (BASE2*) must be changed, since the display buffer will over write ODT (1000-1577) otherwise. Debugging is not usually hampered by moving the display buffer up into the end of the floating point buffer area, since a limited display field is acceptable when debugging. The arrangement is intentionally designed to put the MOCL loader and ODT in data areas which will be overwritten by data during the normal operation of STRIP, since these programs would presumable need to be used only at load time of STRIP.

The non-disc-resident verions of STRIP can use the standard binary (SA 7777) loader and ODT (1000-1577) in a similar manner.

Applications

STRIP has proved useful in a wide variety of applications in spite of the fact that it has been available for only 3-1/2 months.

Since the data input routine for STRIP is via the Floating Point Package (FPP), the input format has the restrictions mentioned in the FPP writeup. Since the FPP output format is compatable with the input to STRIP, it can be used to plot data generated in FORTRAN, CALCULATOR, or FOCAL, or any other program using the FPP for output. (A minor modification to the input routines will allow the program to be used in installations without the high speed paper tape reader).

Spooner, et al.,[1,2,3] use the disc resident version of STRIP for an almost on-line plotter (as well as for initial data reduction) for data from a neutron diffractometer data acquisition system. The facility for rapid turn-around and the availability of Polaroid camera pictures of the display have made a significant improvement in the operation of their diffractometers. For example, the data used as the subject of the example in this paper was taken from such an experiment. The central peak (see Figure 1) of the data is the result of poor collimation of the incident beam, and the availability of the display allowed the experimenters to correct this situation before using up more beam time (each point on the plot represents 10 minutes of neutron beam time!).

*A listing of STRIP and the Gaussian routine HULME is available from the author.

In another application, a study of filatration of particles through sand beds by Champlain, et al.[4], has been made possible by STRIP. The volume of data acquired by the experimenter (about 500, 400 channel spectra) and the difficulties of dealing with the rather complicated background in this experiment were such that some mechanized data reduction scheme is required. Normal fitting techniques proved elusive because of the aforementioned difficult background situation.

The obvious use of STRIP is for reduction of data from Pulse Height Analysers. The saving in time of this method over hand methods of analysis has significantly improved the work done by a group doing neutron activation analysis. The "accuracy" of the results seems to compare favorably with tedious graphical methods usually involving centroid determination, and "block counting" integration methods. By use of the "#" key which permanently subtracts the currently defined Gaussian from the data buffer, it is possible to completely separate the peaks in a complicated spectrum from the background which may be quite complicated in shape also. In one case, the user was able to separate a small peak of 10% of the area of a large peak which was well up on the "skirt" of the large peak.

Conclusion

STRIP is a data display program that is easily used by the experimenter to examine and partially reduce his data. The reliance upon the judgement of the user in fitting operations make it very useful in situations where normal least squares techniques are unsatisfactory and the facility for expansion and change within the program make it possible for the program to "grow" toward solving the particular needs of a large number of widely different applications.

Bibliography

Spooner, S., and Wrege, D. E.: "Production of Polarized Neutrons Using Neutron Optics", 1968 Annual Meeting, Georgia Academy of Science, April 26, 1968.

Spooner, S., and Lynn, J. W.: "Neutron Diffraction Study of Ordering Phenomena in Magnetic Alloys", 1968 Annual Meeting, Georgia Academy of Science, April 26, 1968.

Spooner, S., and Young, R. A.: "Neutron Diffraction Study of Tooth Enamel", to be published.

Champlin, J.: "The Analysis of Wood by Neutron Activation", 1968 Annual Meeting, Georgia Academy of Science, April 26, 1968.

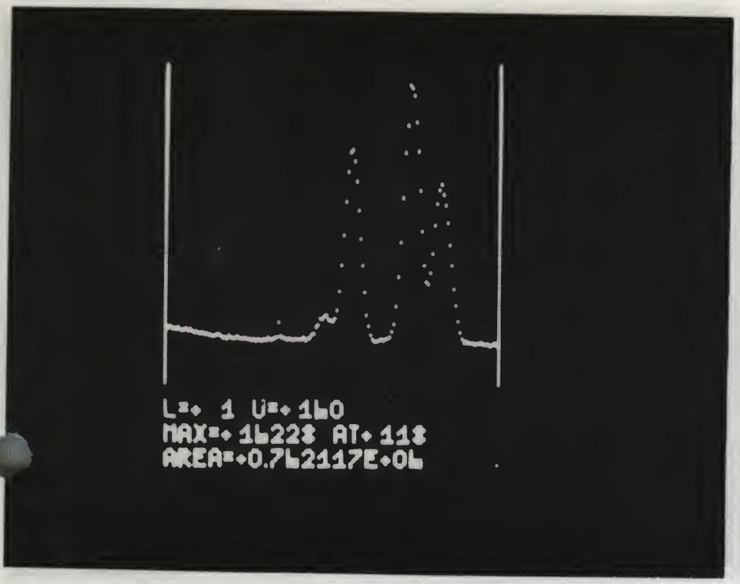

Figure 1

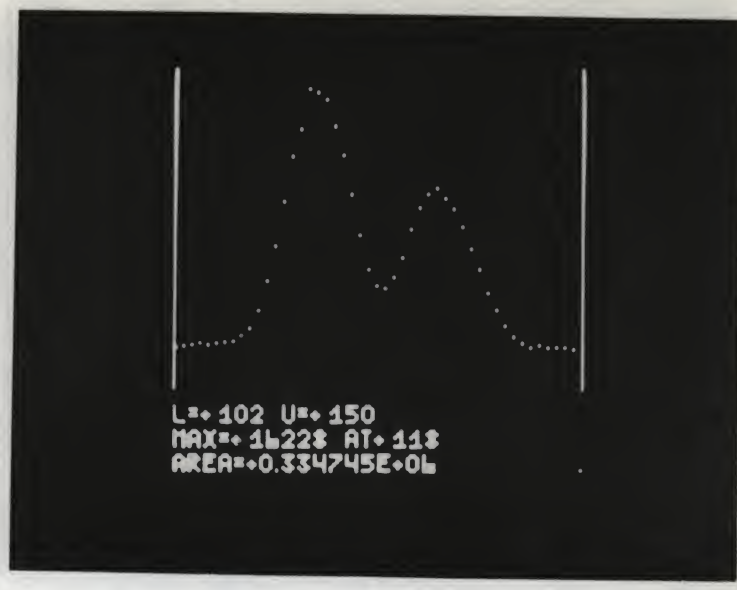

Figure 2

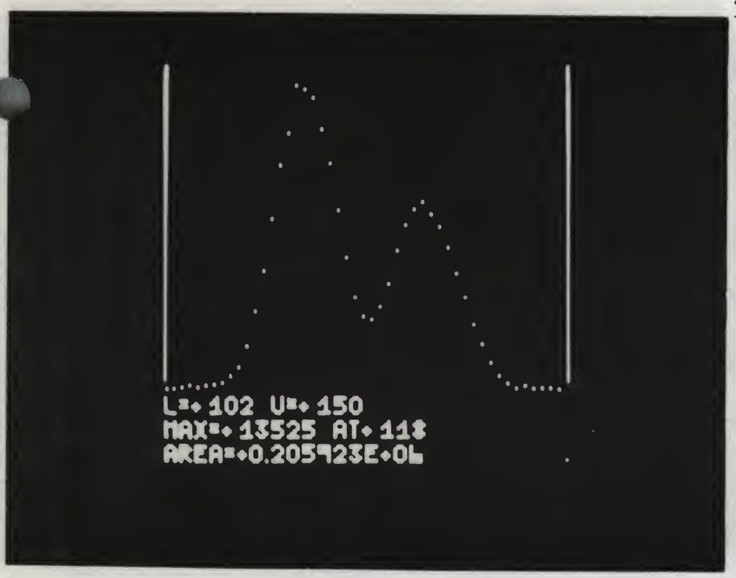

Figure 3

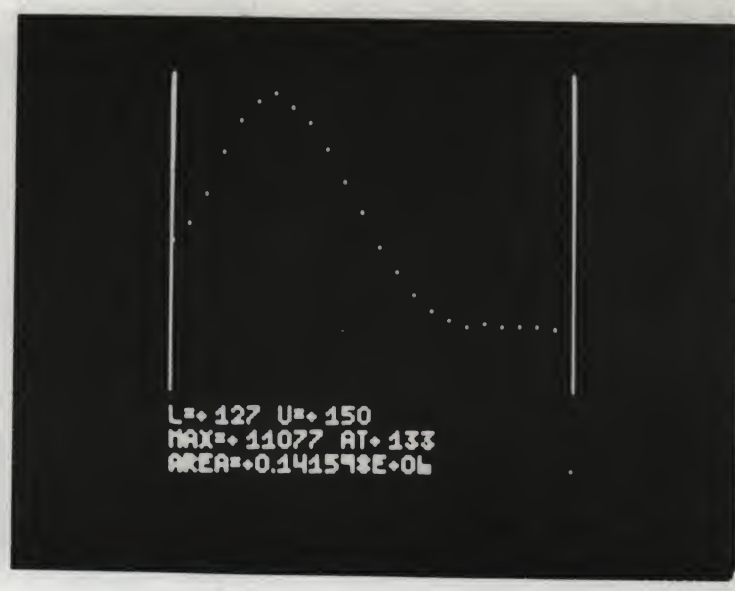

Figure 4

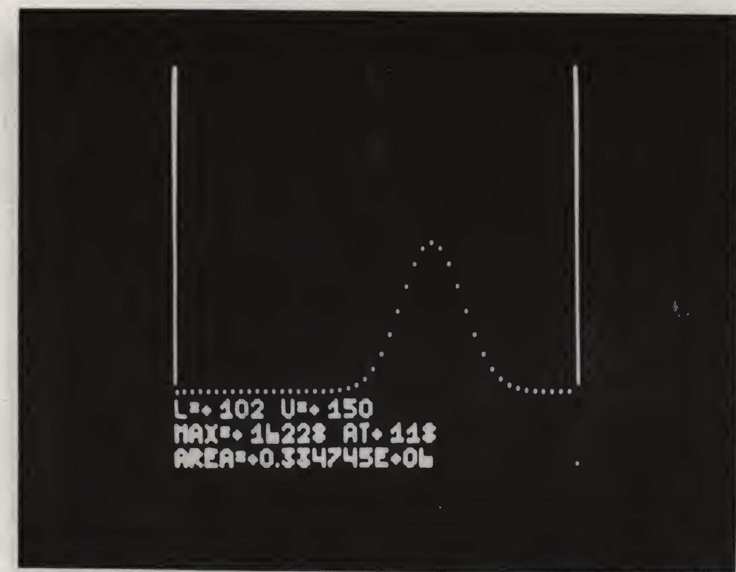

Figure 5

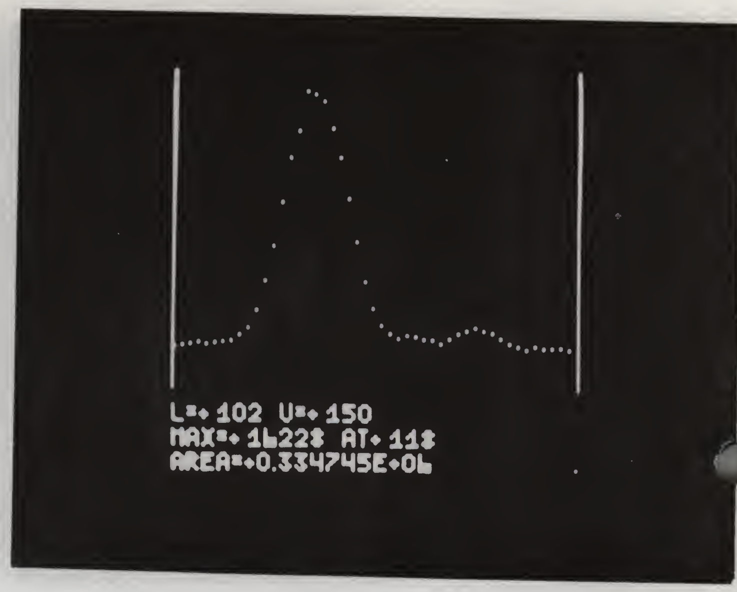

Figure 6

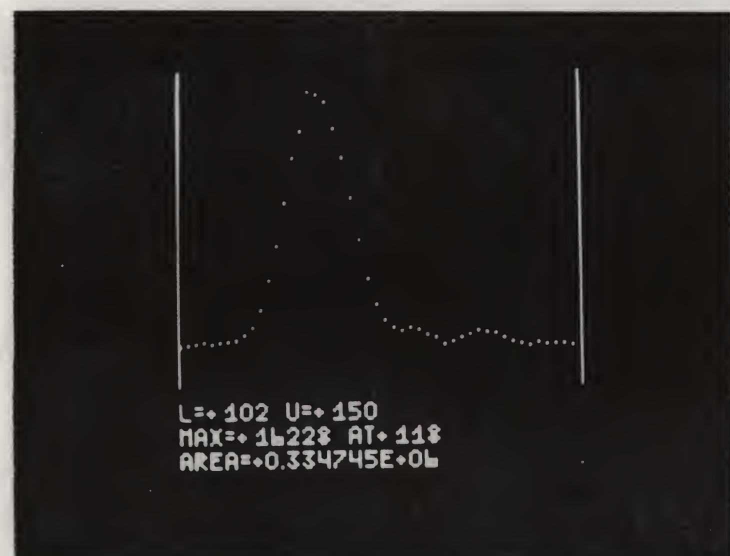

Figure 7

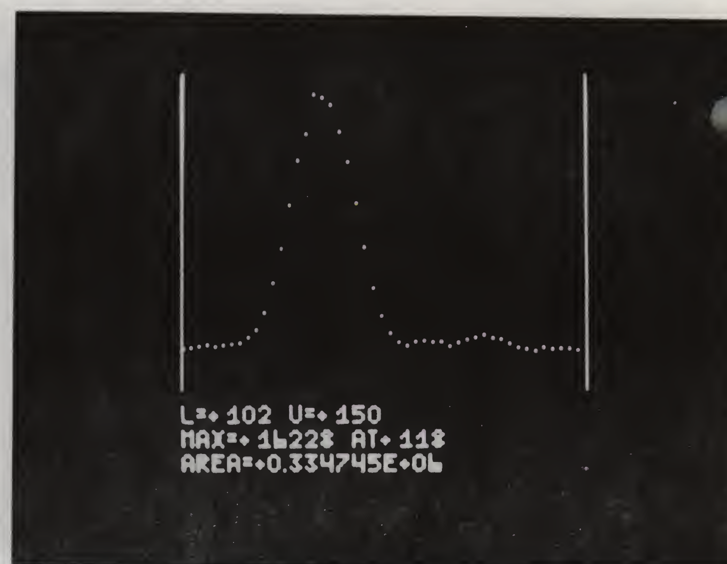

Figure 8

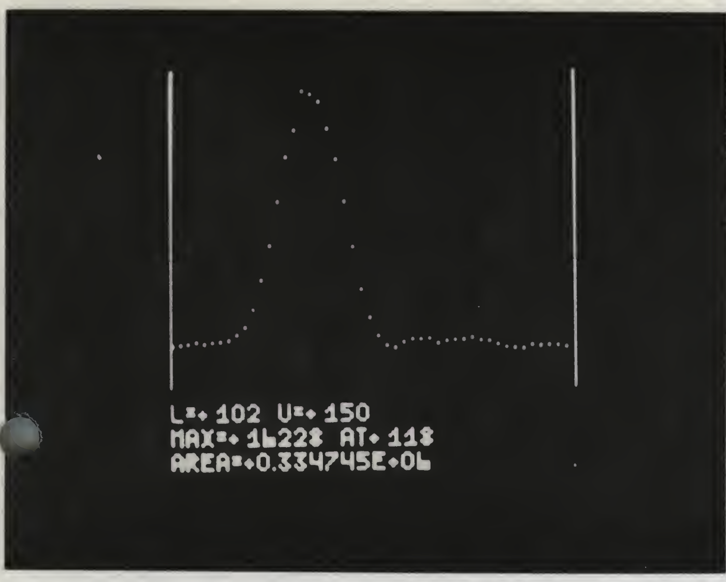

Figure 9

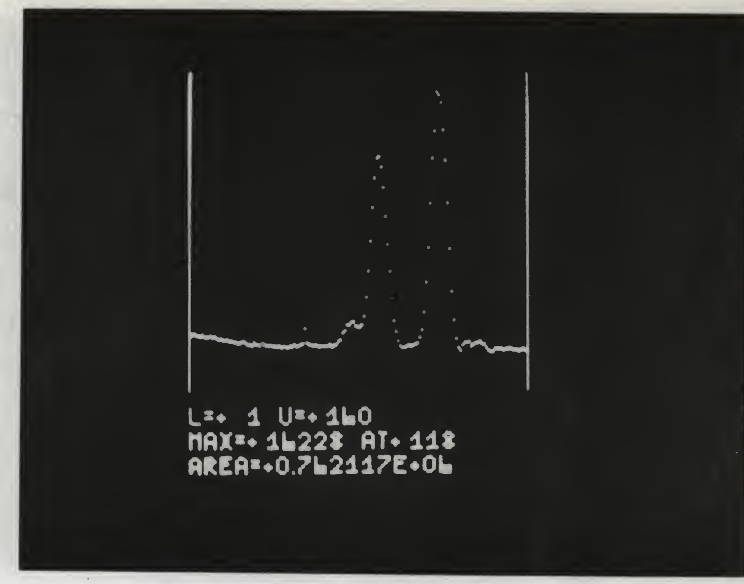

Figure 10

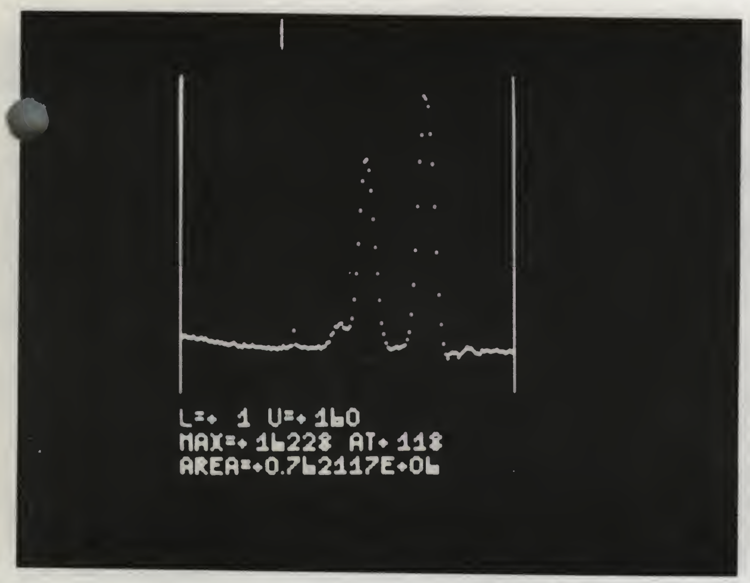

Figure 11

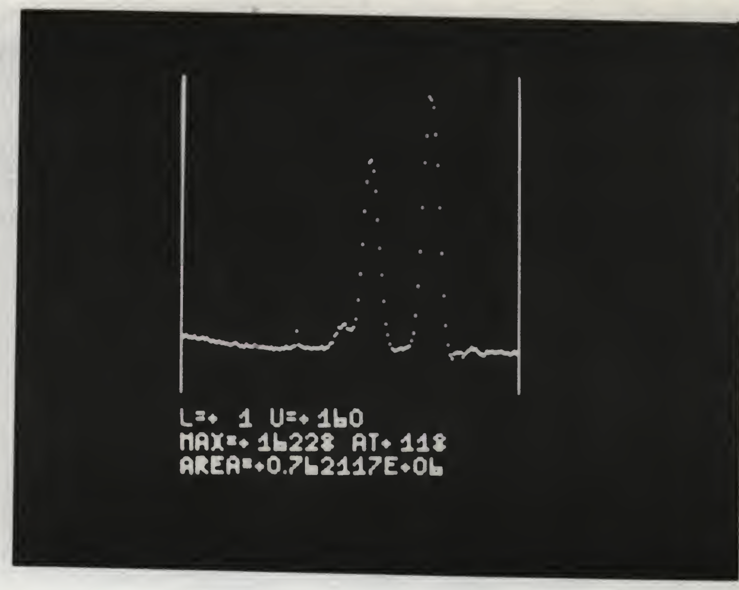

Figure 12

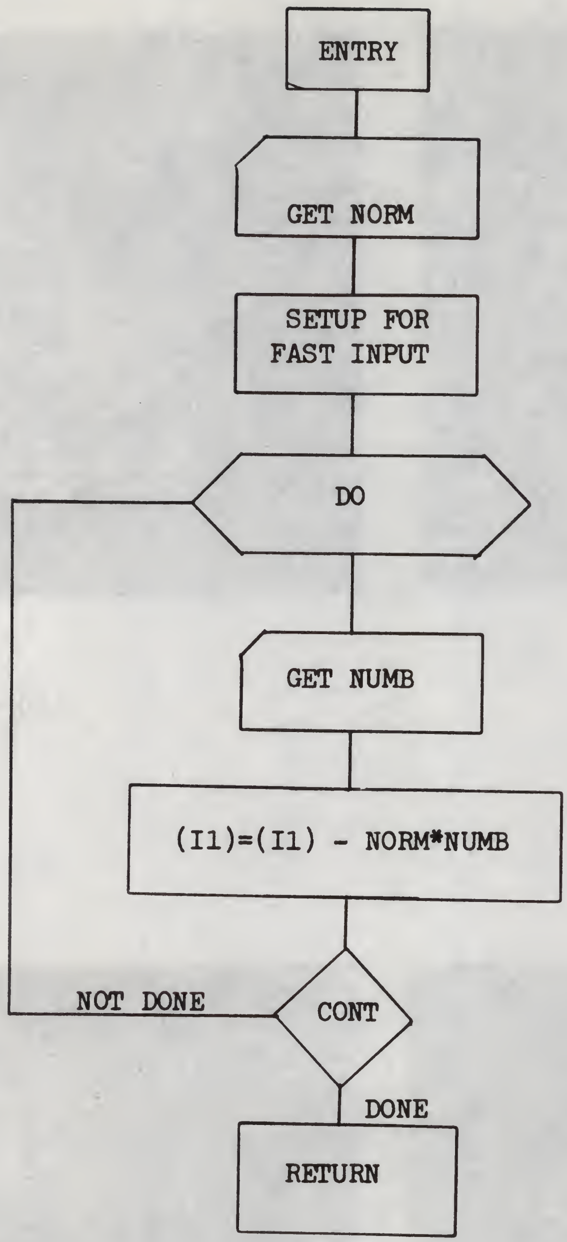
ENTRY
GET NORM
SETUP FOR
FAST INPUT
DO
GET NUMB
(I1)=(I1) - NORM*NUMB
NOT DONE
CONT
DONE
RETURN

FIGURE 13

Figure 14

```
                 /SUBROUTINE TO SUBTRACT BACKGROUND SPECTRUM TIMES
                 /OPERATOR-SUPPLIED NORMALIZATION FACTOR, FROM THE RESIDENT
                 /SPECTRUM.
                 /CALL WITH "W" KEY, AND SUPPLY NORMALIZATION FACTOR
                 /AS ASKED FOR. FAST TAPE READER WILL THEN READ
                 /BACKGROUND SPECTRUM WHILE SUBTRACTING NORMALIZED SP-
                 /ECTRUM FROM EACH POINT

                  DO=JMS I 111
                  CONT=JMS I 112
                  FNTR=JMS I 7
                  FNEG=1Ø
                  I1=1Ø5
                 /SETUP TNIZE W KEY
                 *267
Ø267   7451                -327
Ø27Ø   ØØØØ                Ø
                 *154
Ø154   41ØØ                W              /ENTRY IN DIRECTORY TABLE
                 *41ØØ
41ØØ   ØØØØ      W,        Ø              /BACKGROUND SUBTRACTION ROUTINE
41Ø1   4726                JMS I CRLFP    /
41Ø2   4727                JMS I MESSAG   /MESSAGE PRINTOUT ROUTINE
41Ø3   1617                1617           /NO
41Ø4   2215                2215           /RM
41Ø5   754Ø                754Ø           /= SP
41Ø6   ØØØØ          Ø
41Ø7   4531                JMS I 131      /INPUT NORM
411Ø   44Ø7                FNTR           /ENTER FLOATING POINT
4111   633Ø                FPUT NORM      /STASH FACTOR
4112   ØØØØ                FEXT
4113   3Ø56                DCA 56         /56=Ø IS FAST READER CONDITION
4114   4511                DO
4115   4531                JMS I 131      /INPUT A NUMBER
4116   44Ø7                FNTR
4117   333Ø                FMPY NORM
412Ø   ØØ1Ø                FNEG
4121   15Ø5                FADD I I 1
4122   65Ø5                FPUT I I 1     /(I1)=(I1)-NORM*NUMBER
4123   ØØØØ                FEXT
4124   4512                CONT
4125   57ØØ                JMP I W
4126   417Ø      CRLFP,    417Ø           /SEE LISTING
4127   4274      MESSAG,   4274
413Ø   ØØØØ      NORM,     Ø
4131   ØØØØ                Ø
4132   ØØØØ                Ø
```

CONT	4512
CRLFP	4126
DO	4511
FNEG	ØØ1Ø
FNTR	44Ø7
I1	Ø1Ø5
MESSAG	4127
NORM	413Ø
W	41ØØ

LOADING STRIP ONTO THE DISC

There are 4 binary files on the distributed version of STRIP.

```
.LOAD
*IN-R:,R:,R:,R:
*
*
*
*
OPT-2
ST=
↑↑↑↑↑↑↑↑↑↑↑
zSAVE STRI!:0-777,1600-2177,3600,4200-7577;3600
```

The present version of STRIP callable from the disc has a small patch at 36ØØ to test the switches before execution, and halt if they are ØØØØ. This can be used in conjunction with a binary loader in the page starting at 36ØØ, to load user written subroutines over STRIP. ODT (1ØØØ) will fit in the "hole" at 1ØØØ-1577, but the user should change the contents of 445 (pointer to the beginning of the display buffer) before beginning execution of STRIP, since the display buffer will overwrite ODT (low) if this is not done.

Calling STRIP from the disc:

1. With disc monitor resident 76ØØ-7777, set SR to 76ØØ, press LOAD ADD, START
2. If halt before execution is desired, set SR to Ø.
3. Type STRIP
4. Teleprinter will type a (totally meaningless) "?", and the display should show some data.

LOADING STRIP USING A NON-DISC SYSTEM

1. With Binary Loader in core, set SR to 7777, Press LOAD ADD
2. (Set switches for fast reader option if applicable).
3. Load paper tape into reader, turn it on.
4. Press START.
5. When tape stops, press CONT
6. When tape stops, press CONT
7. When tape stops, press CONT
8. When tape stops, set SR to Ø2ØØ, press LOAD ADD, then START.

STRIP FOR ASR-33 PAPER TAPE READER

STRIP can be overlaid with the patch available from the library for the purpose, and will read data from the low speed paper tape reader when the "R" key is struck, until the number of data points is satisfied, or the <CTRL>P key is struck by the operator, who must be quick to turn the reader off, or data will be interpreted as commands, generating many question marks.

```
                /PAGE 1

                /DEFINITIONS

                MONTR=7577
                DO=JMS I DU
                CONT=JMS I CONTNU
                FIXX=JMS I FIXP
                FLOTE=JMS I FLOTEP
                INPUT=JMS I GETP
                PRINT=JMS I OUTP
                FNTR=JMS I 7
                FACC=44
                FAST=DCA 56
                OUTPUT=JMS I 6
                SYMGEN=5000
                CHAR=57


                /FLOATING POINT PACKAGE SETUP
                *5
0005    7400    7400
0006    7200    7200
0007    5600    5600
                        *SYMGEN+116
5116    5200            SYMGEN+200
                        *SYMGEN+126
5126    5127            .+1
                        *7143
7143    5744            JMP I .+1
7144    0530            SPEEDS
                        *62
0062    0006            6
                *56
0056    0000    0

                /FLOATING PAGE ZERO CONSTANTS
                        *20
0020    0000    MAX,    0
0021    0000            0
0022    0000            0
0023    0012    F1024,  12
0024    3000            3000     /1400(8)
0025    0000            0
0026    0000    A,      0
0027    0000            0
0030    0000            0
0031    0000    B,      0
0032    0000            0
0033    0000            0
0034    0000    FAREA,  0        /FLOATING AREA UNDER CURVE
0035    0000            0
0036    0000            0
```

```
                /PAGE 2

                /FIXED PAGE ZERO CONSTANTS
                     *100
0100   2000   P2000,  2000
0101   0400   P400,   400
0102   0004   P4,     4
0103   0077   P77,    77
0104   0006   P6,     6
0105   0000   I1,     0
0106   0000   I2,     0
0107   0000   I3,     0
0110   0000   MAXADD, 0
0111   0400   DU,     DOIT
0112   0433   CONTNU, LOOP+1
0113   0475   FIXP,   FIX
0114   0001   L,      1          /LOWER LIMIT
0115   0400   U,      400        /UPPER LIMIT
0116   0447   FLOTEP, FLOAT
0117   0275   EQ,     275
0120   0240   SP,     240
0121   7344   OUTP,   7344
0122   0004   X,      4
0123   2054   SHIFT,  CSHIFT
0124   2072   LBFR,   LBFRA
0125   0000   IMAX,   0
0126   0001   L1,     1          /LOWER VERTICAL LINE CHANNEL
0127   0400   U1,     400
0130   7774   M4,     -4
0131   0557   GETP,   GET
0132   0732   MP,     MAXIM
0133   0755   AP,     AREA
0134   1707   NP,     NORM
0135   2000   CP,     T
0136   0636   RP,     R
0137   0600   ADDRS,  ELL
0140   0605           EWE
0141   0701           DATAIN
0142   0714           D
0143   0724           E
0144   1600           S
0145   0660           EFF
0146   0612           UMAX
0147   7577           MONTR
                      *172
0172   4542           JMS I ADDRS+3   /EXECUTE "D"
0173   1177   QUEST,  TAD Q
0174   4521           JMS I OUTP
0175   5576           JMP I .+1
0176   0204           LOOK
0177   0277   Q,      277
```

```
                  /PAGE 3

                        *200
0200   6044   READIN, 6044     /INITIALIZE SOME FLAGS
0201   6014           6014
0202   6032           6032
0203   7000           NOP      /SPARES

              /LOOK TO KEYBOARD FOR NEXT INSTRUCTION
0204   4532   LOOK,   JMS I MP
0205   4533           JMS I AP
0206   4535           JMS I CP
0207   4534           JMS I NP
0210   1246           TAD CR
0211   4521           JMS I OUTP
0212   1247           TAD LF
0213   4521           JMS I OUTP
0214   4301   LOOKY,  JMS AGAIN
0215   7240           CLA CMA /SET 56
0216   3056           DCA 56
0217   4405           JMS I 5
0220   1060           TAD 60
0221   7640           SZA CLA
0222   5173           JMP QUEST        /CHANGE THIS IF NUMERICAL ARGUMENTS
                      /ARE VALID
              /SEARCH A TABLE OF CHARACTERS , AND GO TO LOCATION INDICATED
              /BY TABLE STARTING AT TP, FOR A ROUTINE TO DO WHAT THE
              /CHARACTER IMPLIED.      ENTER HERE WITH "CHAR" ALREADY SET

0223   3105           DCA I1  /CLEAR INDEX
0224   1251           TAD TP  /SET POINTER
0225   3017           DCA 17  /SETUP INDEX REGISTER
0226   1417   LOOP1,  TAD I 17         /GET TEST CHARACTER
0227   7450           SNA
0230   5173           JMP QUEST        /NOT IN TABLE
0231   1057           TAD CHAR         /ADD CHARACTER
0232   7650           SNA CLA
0233   5236           JMP .+3 /GOT IT!
0234   2105           ISZ I1  /STEP POINTER
0235   5226           JMP LOOP1        /CONTINUE
0236   1105           TAD I1  /ADD INDEX
0237   1250           TAD BASE3        /TO ROUTINES TABLE BASE
0240   3105           DCA I1
0241   1505           TAD I I1         /GET POINTER TO PROGRAM
0242   3105           DCA I1  /DOUBLE INDIRECT
0243   4505           JMS I I1         /EXECUTE CALLED SUBROUTINE
0244   7300           CLA CLL
0245   5204           JMP LOOK
0246   0215   CR,     215
0247   0212   LF,     212
0250   0137   BASE3,  ADDRS
0251   0251   TP,     .
0252   7464           -314     /L
0253   7453           -325     /U
0254   7456           -322     /R
0255   7474           -304     /D
0256   7475           -303     /C
0257   7455           -323     /S
0260   7472           -306     /F
0261   7571           -207     /BELL
0262   7575           -203     /↑C
0263   0000           0        /TABLE TERMINATOR!
```

/PAGE 4

```
                          *TP+30
0301  0000  AGAIN,  0           /DISPLAY DATA, TITLES,LINES
0302  6031          6031        /TEST KEYBOARD FLAG
0303  7410          SKP         /REVERSE SENSE OF TEST
0304  5701          JMP I AGAIN       /EXIT IF KEYBOARD STRUCK
0305  6077          6077        /SET INTENSITY REGISTER
0306  1100          TAD P2000
0307  3330          DCA XAXIS         /ADD ZERO OFFSET
0310  4511          DO
0311  1506          TAD I I2          /GET VALUE FROM DISPLAY REGISTER
0312  1177          TAD Q
0313  6063          6063
0314  7200          CLA
0315  1330          TAD XAXIS
0316  6057          6057     /DISPLAY THE POINT
0317  1122          TAD X    /ADD X
0320  3330          DCA XAXIS
0321  4512          CONT
0322  1126          TAD L1
0323  4331          JMS LINES
0324  1127          TAD U1
0325  4331          JMS LINES
0326  4360          JMS TITLES
0327  5302          JMP AGAIN+1
0330  0000  XAXIS,  0

0331  0000  LINES,  0
0332  7041          CIA
0333  1114          TAD L
0334  7041          CIA
0335  3360          DCA TITLES
0336  1122          TAD X
0337  7041          CIA
0340  3330          DCA XAXIS         /SETUP COUNTER
0341  1360          TAD TITLES
0342  2330          ISZ XAXIS         /TEST COUNTER
0343  5341          JMP .-2
0344  1100          TAD P2000         /ADD OFFSET
0345  6053          6053
0346  7300          CLA CLL
0347  1252          TAD TP+1
0350  3105          DCA I1
0351  1100          TAD P2000
0352  6067          6067     /DISPLAY
0353  1130          TAD M4
0354  2105          ISZ I1  /TEST
0355  5352          JMP .-3
0356  7300          CLL CLA /CLEAR
0357  5731          JMP I LINES
0360  0000  TITLES, 0       /DISPLAY TEXT
0361  1370          TAD ORDI
0362  3767          DCA I VALUP
0363  1124          TAD LBFR
0364  4766          JMS I GIANTS
0365  5760          JMP I TITLES
0366  5000  GIANTS, SYMGEN
0367  5176  VALUP,  SYMGEN+176
0370  0200  ORDI,   200
```

```
                    /PAGE 5

                    /THIS SUBROUTINE ACTS LIKE A DO LOOP, AND CONTINUE STATEMENT
                    /IN FORTRAN, EXCEPT THAT ARGUMENTS ARE NOT VARIABLE.    THE
                    /CALL TO THE SUBROUTINE IS SIMPLY "DO" AND      "CONT",WHICH
                    /THEN EXECUTES IBTRUCTIONS AFTER DO DOWN TO CONT, UNTIL THE
                    /END OF THE LOOP, WHERE THE NEXT INSTRUCTION FOLLOWS CONT.

                            *READIN+200
0400    0000        DOIT,   0       /INITIAL ENTRY
0401    1114                TAD L
0402    7041                CIA
0403    7500                SMA     /L .LE.0 IS ILLEGAL
0404    5172                JMP QUEST-1
0405    1115                TAD U   /U-L
0406    7550                SPA SNA /
0407    5172                JMP QUEST-1     /TOO SMALL
0410    3233                DCA LOOP+1      /SAVE IT
0411    1233                TAD LOOP+1
0412    7041                CIA
0413    3246                DCA CNTR        /SETUP COUNTER
0414    1246                TAD CNTR
0415    1101                TAD P400
0416    7710                SPA CLA
0417    5172                JMP QUEST-1     /TOO BIG!
0420    1114                TAD L   /L*3
0421    7104                CLL RAL
0422    1114                TAD L
0423    1244                TAD BASE1       /3L+BASE1=FIRST ADDRESS FLOATING DAA
0424    3105                DCA I1
0425    1245                TAD BASE2       /L+BASE2=FIRST ADDRESS DISPLAY DATA
0426    1114                TAD L
0427    3106                DCA I2
0430    1114                TAD L
0431    3107                DCA I3  /SETUP POSITIVE INDEX COUNTER
0432    5600        LOOP,   JMP I DOIT
0433    0000                0
0434    2105                ISZ I1  /STEP INDEXES
0435    2105                ISZ I1  /I1 IS A FLOATING POINT INDEX
0436    2105                ISZ I1
0437    2106                ISZ I2
0440    2107                ISZ I3
0441    2246                ISZ CNTR        /TEST COUNTER
0442    5600                JMP I DOIT      /CONTINUE
0443    5633                JMP I LOOP+1    /EXIT
0444    2200        BASE1,  2200
0445    1000        BASE2,  1000
0446    0000        CNTR,   0
```

```
/PAGE 6


/SUBROUTINE TO FLOAT THE NUMBER IN THE ACC AND PUT IT IN
/LOCATION DESIGNATED BY ADDRESS FOLLOWING CALL
/CALL BY        JMS FLOAT       /NUMBER IN ACC
/               ADDRESS
/               RETURN HERE     /ACC CLEAR
0447  0000  FLOAT,  0
0450  3045          DCA 45
0451  1647          TAD I FLOAT
0452  3327          DCA TEMP
0453  3046          DCA 46
0454  1325          TAD C13
0455  3044          DCA 44
0456  1327          TAD TEMP
0457  1266          TAD M44
0460  7640          SZA CLA /TEST IF ADDRESS WAS FACC
0461  5267          JMP FINE
0462  4407          FNTR
0463  7000          FNOR
0464  0000          FEXT
0465  5273          JMP FINIS
0466  7734  M44,    -44
0467  4407  FINE,   FNTR
0470  7000          FNOR
0471  6727          FPUT I TEMP
0472  0000          FEXT
0473  2247  FINIS,  ISZ FLOAT
0474  5647          JMP I FLOAT

            /SUBROUTINE TO FIX NUMBER IN FLOATING ACCUMULATOR AND
            /EXIT WITH IT IN ACC.   FACC DESTROYED. IF NUMBER
            / GREATER THAN 2047 OR LESS THAN -2047, RETURN IS TO
            /CALL+1.            IF CONVERSION SUCCESSFUL RETURN TO CALL+2
0475  0000  FIX,    0       /FIX F(AC) AS 11-BIT SIGNED INTEGER
0476  7200          CLA
0477  1044          TAD 44  /FETCH EXPONENT
0500  7540          SZA SMA /IS THE EXPONENT<1?
0501  5304          JMP .+3 /NO:
0502  7200          CLA     /YES: FIX IT TO 0
0503  5323          JMP DONE+1
0504  1326          TAD M13 /NO: SET BINARY POINT AT
0505  7450          SNA     /11 (10) PLACES TO RIGHT OF CURRENT PT.
0506  5322          JMP DONE        /IT IS ALREADY THERE: ALL DONE
0507  7500          SMA     /TEST TO SEE IF IT IS TOO LARGE
0510  5675          JMP I FIX       /YES: NUMBER>2**11
0511  3044          DCA 44  /NO: SET SCALE COUNT
0512  7100  GO,     CLL     /0 TO C(L)
0513  1045          TAD 45  /FETCH MANTISSA
0514  7510          SPA     /IS IT <0?
0515  7020          CML     /YES: PUT A 1 IN LEFT BIT
0516  7010          RAR     /SCALE RIGHT
0517  3045          DCA 45  /RESTORE IT
0520  2044          ISZ 44  /TEST IF SHIFTED ENOUGH
0521  5312          JMP GO  /NO: CONTINUE
0522  1045  DONE,   TAD 45  /ANSWER IN C(AC)
0523  2275          ISZ FIX
0524  5675          JMP I FIX       /RETURN
0525  0013  C13,    13
0526  7765  M13,    -13
0527  0000  TEMP,   0
```

```
                /PAGE 7

0530  1056  SPEEDS, TAD 56  /TEST FLAG
0531  7640          SZA CLA /IS ZERO, FAST INPUT
0532  5346          JMP SLOWS
0533  2352          ISZ TEMP1
0534  7410          SKP
0535  5751          JMP I PEFM2     /READER OUT OF TAPE, EXIT
0536  6011          6011
0537  5333          JMP .-4
0540  3352          DCA TEMP1
0541  6016          6016    /GET THE DATA
0542  0354          AND P177
0543  1353          TAD P200
0544  3057          DCA 57
0545  5756          JMP I P7152
0546  4750  SLOWS,  JMS I AGAINP
0547  5755          JMP I P7146
0550  0301  AGAINP, AGAIN
0551  0656  PEFM2,  EFF-2
0552  0000  TEMP1,  0
0553  0200  P200,   200
0554  0177  P177,   177
0555  7146  P7146,  7146
0556  7152  P7152,  7152
0557  0000  GET,    0
0560  4405          JMS I 5 /GET A NUMBER
0561  1060          TAD 60  /VALID?
0562  7650          SNA CLA
0563  5360          JMP .-3 /IGNORE IF NOT
0564  5757          JMP I GET
                    *READIN+400
0600  0000  ELL,    0
0601  1126          TAD L1
0602  4217          JMS FETCH
0603  3126          DCA L1
0604  5600          JMP I ELL
0605  0000  EWE,    0
0606  1127          TAD U1
0607  4217          JMS FETCH
0610  3127          DCA U1
0611  5605          JMP I EWE
0612  0000  UMAX,   0
0613  1101          TAD P400        /MAXIMUM LIMIT OF U
0614  4217          JMS FETCH
0615  3101          DCA P400
0616  5612          JMP I UMAX
0617  0000  FETCH,  0
0620  4516          FLOTE
0621  0044          FACC
0622  1117          TAD EQ
0623  4521          JMS I OUTP
0624  4406          OUTPUT
0625  1120          TAD SP
0626  4521          JMS I OUTP
0627  4405          JMS I 5
0630  1060          TAD 60
0631  7650          SNA CLA
0632  2217          ISZ FETCH
0633  4513          FIXX
0634  5173          JMP QUEST
0635  5617          JMP I FETCH
```

```
                /PAGE 8

0636    0000    R,      0
0637    3044            DCA 44  /CLEAR FACC
0640    3045            DCA 45
0641    3046            DCA 46
0642    4511            DO
0643    4407            FNTR
0644    6505            FPUT I I1       /CLEAR DATA BUFFER
0645    0000            FEXT    /BEFORE READING IN MORE
0646    4512            CONT
0647    3056            FAST
0650    1636            TAD I R
0651    3125            DCA IMAX
0652    2236            ISZ R
0653    4511            DO
0654    4525            JMS I IMAX      /EXECUTE PROGRAM POINTED TO
                        /IN LOCATION FOLLOWING CALL
0655    4512            CONT
0656    4260            JMS EFF
0657    5636            JMP I R
0660    0000    EFF,    0
0661    4532            JMS I MP        /FIND MAXIMUM
0662    4407            FNTR
0663    5020            FGET MAX
0664    4023            FDIV F1024      /NORMALIZE
0665    6020            FPUT MAX
0666    0000            FEXT
0667    4511            DO
0670    4407            FNTR
0671    5505            FGET I I1
0672    4020            FDIV MAX
0673    0000            FEXT
0674    4513            FIXX
0675    7300            CLA CLL
0676    3506            DCA I I2
0677    4512            CONT
0700    5660            JMP I EFF
0701    0000    DATAIN, 0       /READ DATA INTO BUFFER
0702    4314            JMS D   /RESET LIMITS BEFORE INPUTTING
0703    4536            JMS I RP
0704    0706            REED
0705    5701            JMP I DATAIN
0706    0000    REED,   0       /INPUT AND STASH SUBR
0707    4531            INPUT
0710    4407            FNTR
0711    6505            FPUT I I1
0712    0000            FEXT
0713    5706            JMP I REED
0714    0000    D,      0       /RESTORE L=0,U=UMAX LIMITS
0715    7301            CLA CLL IAC /SET ACC=1
0716    3126            DCA L1  /SETUP FOR LINES
0717    1101            TAD P400
0720    3127            DCA U1
0721    4324            JMS E
0722    4260            JMS EFF
0723    5714            JMP I D
```

```
                    /PAGE 8A

0724  0000  E,      0         /SET DOLOOP FOR NEW LIMITS
0725  1126          TAD L1
0726  3114          DCA L
0727  1127          TAD U1
0730  3115          DCA U
0731  5724          JMP I E
0732  0000  MAXIM,  0          /SUBROUTINE TO FIND MAXIMUM
0733  7240          CLA CMA /SET EXPONENT TO 7777
0734  3020          DCA MAX
0735  4511          DO
0736  4407          FNTR
0737  5505          FGET I I1
0740  2020          FSUB MAX
0741  0000          FEXT
0742  1045          TAD FACC+1
0743  7710          SPA CLA
0744  5353          JMP ENDM
0745  4407          FNTR
0746  5505          FGET I I1         /RESET MAX
0747  6020          FPUT MAX
0750  0000          FEXT
0751  1107          TAD I3
0752  3110          DCA MAXADD
0753  4512  ENDM,   CONT
0754  5732          JMP I MAXIM
0755  0000  AREA,   0
0756  7300          CLA CLL
0757  3044          DCA 44  /CLEAR FACC
0760  3045          DCA 45
0761  3046          DCA 46
0762  4511          DO
0763  4407          FNTR
0764  1505          FADD I I1
0765  6034          FPUT FAREA
0766  0000          FEXT
0767  4512          CONT
0770  5755          JMP I AREA
```

```
                /PAGE 9
                        *1600
1600  0000      S,      0
1601  4532              JMS I MP          /FIND MAXIMUM
1602  4511              DO        /USING "DO" FOR TESTING!
1603  1512              TAD I CONTNU
1604  4516              FLOTE
1605  1704              C
1606  4407              FNTR
1607  5020              FGET MAX
1610  4023              FDIV F1024
1611  6301              FPUT SCALE
1612  0000              FEXT
1613  3044              DCA 44  /CLR FACC
1614  3045              DCA 45
1615  3046              DCA 46
1616  4407              FNTR
1617  6020              FPUT MAX
1620  6034              FPUT FAREA        /CLEAR FLOATING AREA
1621  5505              FGET I I1
1622  6026              FPUT A
1623  0000              FEXT
1624  4511              DO
1625  4407              FNTR
1626  5505              FGET I I1
1627  6031              FPUT B
1630  0000              FEXT
1631  4512              CONT
1632  4407              FNTR
1633  5031              FGET B
1634  2026              FSUB A
1635  4304              FDIV C
1636  6031              FPUT B  /B=SLOPE=(F[U]-F[L])/U-L
1637  0000              FEXT
                        SLOPE=B
```

```
                /PAGE 10

                /SETUP COMPLETE, NOW DO STRIPPING
1640   4511             DO
1641   4407             FNTR
1642   5505             FGET I I1       /GET F(I)
1643   2026             FSUB A  /F(I)-A
1644   6304             FPUT C  /TEMPORARY SSTORAGE
1645   5026             FGET A
1646   1031             FADD SLOPE      /A+SLOPE
1647   6026             FPUT A
1650   5304             FGET C
1651   1034             FADD FAREA      /COMPUT AREA UNDER NEW CURVE
1652   6034             FPUT FAREA
1653   5304             FGET C
1654   2020             FSUB MAX        /NOW TEST MAX
1655   0000             FEXT
1656   1045             TAD 45
1657   7710             SPA CLA /SKIP IF NEW MAXIMUM FOUND
1660   5267             JMP OK
1661   4407             FNTR
1662   5304             FGET C  /RESET MAX
1663   6020             FPUT MAX
1664   0000             FEXT
1665   1107             TAD I3
1666   3110             DCA MAXADD
1667   4407     OK,     FNTR
1670   5304             FGET C  /SCALE RESULT
1671   4301             FDIV SCALE
1672   0000             FEXT
1673   4513             FIXX
1674   7200             CLA
1675   3506             DCA I I2        /IDATA(I)=FIXF(F(I)-A)
1676   4512             CONT
1677   5700             JMP I .+1
1700   0206             LOOK+2  /DONT RECOMPUTE MAXIMUM AND AREA!
1701   0000     SCALE,  0
1702   0000             0
1703   0000             0
1704   0000     C,      0       /TEMPORARY STORAGE
1705   0000             0
1706   0000             0
```

```
                /PAGE 11

1707  0000  NORM,   0         /NORMALIZE DISPLAY BUFFER AXES
1710  3125          DCA IMAX
1711  4511          DO
1712  1506          TAD I I2          /GET IDATA(I)
1713  7041          CIA
1714  1125          TAD IMAX          /IMAX-IDATA(I)
1715  7700          SMA CLA
1716  5323          JMP ENDN          /GREATER THAN -1
1717  1506          TAD I I2          /GET IT AGAIN
1720  3125          DCA IMAX          /MAKE IT THE MAXIMUM
1721  1107          TAD I3  /GET THE CHANNEL THAT DID IT
1722  3110          DCA MAXADD
1723  4512  ENDN,   CONT
1724  1125          TAD IMAX
1725  4516          FLOTE
1726  0020          MAX     /FLOTE MAXIMUM NUMBER
1727  4407          FNTR
1730  5020          FGET MAX
1731  4023          FDIV F1024
1732  6020          FPUT MAX
1733  0000          FEXT
1734  4511          DO
1735  1506          TAD I I2          /GET DISPLAY BUFFER WORD
1736  4516          FLOTE
1737  0044          FACC
1740  4407          FNTR
1741  4020          FDIV MAX          /NORMALIZE VERTICAL
1742  0000          FEXT
1743  4513          FIXX
1744  7200          CLA
1745  3506          DCA I I2          /PUT NORMALIZED RESULT BACK IN BUFFR
1746  4512          CONT
1747  4511          DO
1750  1512          TAD I CONTNU      /GET U-L
1751  4516          FLOTE
1752  0020          MAX
1753  4407          FNTR
1754  5023          FGET F1024
1755  4020          FDIV MAX          /NORMALIZE HORIZONTAL
1756  0000          FEXT
1757  4513          FIXX
1760  7326          CLA CLL CML RTL /SET TO 2 IF ILLEGAL
1761  3122          DCA X
1762  5707          JMP I NORM
```

```
        /PAGE 12

                    *S+200
2000  0000  T,      0       /PREPARE TITLE
2001  1102          TAD P4
2002  3062          DCA 62  /SETUP FPP FOR 4 DIGIT OUTPUT
2003  1263          TAD PSHIFT
2004  3662          DCA I P7345
2005  1124          TAD LBFR
2006  3016          DCA 16  /SET AUTOINDEX TO DBUFFER
2007  1126          TAD L1
2010  4516          FLOTE
2011  0044          FACC
2012  4406          OUTPUT  /THIS DOESN'T PRINT!
2013  1266          TAD UBFR
2014  3016          DCA 16
2015  1127          TAD U1
2016  4516          FLOTE
2017  0044          FACC
2020  4406          OUTPUT
2021  1270          TAD ADDBFR
2022  3016          DCA 16
2023  1110          TAD MAXADD
2024  4516          FLOTE
2025  0044          FACC
2026  4406          OUTPUT
2027  1104          TAD P6
2030  3062          DCA 62
2031  1267          TAD MBFR
2032  3016          DCA 16
2033  4407          FNTR
2034  5020          FGET MAX
2035  0000          FEXT
2036  4406          OUTPUT
2037  3353          DCA LAST-2      /CLEAR FINAL DIGIT OF BUFFER
2040  3062          DCA 62  /SETUP FOR E FORMAT OUTPUT
2041  1271          TAD ABFR
2042  3016          DCA 16
2043  4407          FNTR
2044  5034          FGET FAREA
2045  0000          FEXT
2046  4406          OUTPUT
2047  1104          TAD P6
2050  3062          DCA 62
2051  1265          TAD P6041
2052  3662          DCA I P7345
2053  5600          JMP I T
2054  0103  CSHIFT, AND P77
2055  7106          CLL RTL
2056  7006          RTL
2057  7006          RTL
2060  3416          DCA I 16
2061  5664          JMP I P7351
2062  7345  P7345,  7345
2063  5523  PSHIFT, JMP I SHIFT
2064  7351  P7351,  7351
2065  6041  P6041,  TSF
2066  2101  UBFR,   UBFRA
2067  2112  MBFR,   MBFRA
2070  2123  ADDBFR, ADBFRA
2071  2135  ABFR,   ABFRA
```

```
                  /PAGE 13
                  /HERE IS THE MESSAGE AREA

2072    1475    LBFRA,1475         /L=
2073    0000          0            /VALUE
2074    0000          0            /OF
2075    0000          0
2076    0000          0            /L HERE
2077    0000          0            /SP
2100    0000          0
2101    2575    UBFRA,  2575       /U=
2102    0000          0
2103    0000          0
2104    0000          0
2105    0000          0
2106    0000          0
2107    0000          0            /SP
2110    0027          27           /CR-LF
2111    1501          1501         /MA
2112    3075    MBFRA,  3075       /X=
2113    0000    0
2114    0000    0
2115    0000    0
2116    0000    0
2117    0000    0
2120    0000    0
2121    0000    0
2122    0000    0
2123    0124    ADBFRA, 0124       /AT
2124    0000          0
2125    0000          0
2126    0000          0
2127    0000          0
2130    0000          0
2131    0000          0
2132    0027          27           /CR-LF
2133    0122          0122
2134    0501          0501         /EA
2135    7500    ABFRA,  7500       /=
2136    0000    0
2137    0000    0
2140    0000    0
2141    0000    0
2142    0000    0
2143    0000    0
2144    0000    0
2145    0000    0
2146    0000    0
2147    0000    0
2150    0000    0
2151    0000    0
2152    0000    0
2153    0000    0
2154    0015          15           /CR
2155    0001    LAST,   1          /END OF MESSAGE
```

```
PAUSE


/HULME 17 JAN 68
/ROUTINES TO SUBTRACT A GAUSSIAN FROM DATA IN THE FP BUFFER
/AND DISPLAY THE RESULT IN THE DISPLAY BUFFER
/GHILL GETS THE PARAMETERS OF THE GAUSSIAN, AND COMPUTES
/SIGMA AND EXACT AREA.  CALL IT WITH "H" KEY
/AMON DOES THE ACTUAL SUBTRACTION USING THE PARAMETERS.
/CALL IT WITH THE "G" KEY.
/SRTEES IS A ROUTINE TO DISPLAY THE GAUSSIAN ALONE
/CALL IT WITH THE "J" KEY
/RINDT IS A ROUTINE TO SUBTRACT THE GUASSIAN FROM THE
/FLOATING DATA BUFFER.  BE SURE YOU WANT TO DO THIS BEFORE
/CALLING IT WITH THE "#" KEY!
/NOTE THAT THE GUASSIAN IS ACTUALLY COMPUTED IN A SEPARATE
/ROUTINE CALLED PHILL.

/PAGE 1

SQUARE=1
FNEG=10
IN=11
OUT=12
MIF=6544

/SETUP KEYS AND FPP
*TP+12
0263  7471          -307    /G
0264  7470          -310    /H
0265  7466          -312    /J
0266  7535          -243    /#
0267  0000          0       /TABLE TERMINATOR!
*ADDRS+11
0150  4400          AMON
0151  4200          GHILL
0152  4430          SRTEES
0153  4342          RINDT
*MIF+FNEG
6554  6000          6000
6555  7400          7400
6556  7200          7200
```

```
                /PAGE 2

                *4200
                        LINE=JMS I CRLFP
                        TYPE=JMS FOYT
4200    0000    GHILL,  0       /GET SOME DATA FROM THE OPERATOR
4201    3062            DCA 62  /SETUP FOR FLOATING OUTPUT FORMAT
4202    4673            LINE
4203    4274            TYPE
4204    0627            0627    /FW
4205    1015            1015    /HM
4206    7540            7540    /= SP
4207    0000            0
4210    1356            TAD FWHMP
4211    4322            JMS FETCHF
4212    4274            TYPE
4213    4023            4023    /SP S
4214    1107            1107    /IG
4215    1501            1501    /MA
4216    7540            7540    /= SP
4217    0000            0
4220    4407            FNTR
4221    5756            FGET I FWHMP
4222    4362            FDIV KONST1     /BY 2.354(10)
4223    6031            FPUT 31
4224    0012            OUT
4225    5031            FGET 31 /OUT DESTROYS FACC
4226    0001            SQUARE
4227    3755            FMPY I TWOP     /DOUBLE IT
4230    6757            FPUT I SIGP     /2*SIGMA*SIGMA
4231    0000            FEXT
4232    4673            LINE
4233    4274            TYPE
4234    1005            1005    /HE
4235    1107            1107    /IG
4236    1024            1024    /HT
4237    7540            7540    /= SP
4240    0000            0
4241    1360            TAD HITEP
4242    4322            JMS FETCHF
4243    4274            TYPE
4244    4001            4001    /SP A
4245    2440            2440    /T SP
4246    0000            0
4247    1361            TAD AP1
4250    4322            JMS FETCHF
4251    4673            LINE
4252    4274            TYPE
4253    0122            0122    /AR
4254    0501            0501    /EA
4255    7540            7540    /= SP
4256    0000            0
4257    4407            FNTR
4260    5760            FGET I HITEP
4261    3031            FMPY 31 /BY SIGMA
4262    3365            FMPY KONST2     /BY SQT(2*PI)
4263    6031            FPUT 31
4264    0012            OUT
4265    5031            FGET 31
4266    0000            FEXT
4267    4673            LINE
4270    1104            TAD 104
4271    3062            DCA 62
4272    5600            JMP I GHILL
4273    4170    CRLFP,  CRLF
```

```
                /PAGE 3
                *4170
4170  0000  CRLF,   0
4171  1376          TAD CR1
4172  4521          PRINT
4173  1377          TAD LF1
4174  4521          PRINT
4175  5770          JMP I CRLF
4176  0215  CR1,    215
4177  0212  LF1,    212
                    *CRLFP+1
4274  0000  FOYT,   0       /MESSAGE PRINTOUT ROUTINE.
4275  1674          TAD I FOYT
4276  7112          CLL RTR
4277  7012          RTR
4300  7012          RTR
4301  4310          JMS GURNEY
4302  1674          TAD I FOYT      /GET WORD AGAIN
4303  4310          JMS GURNEY      /TEST AND OUTPUT
4304  2274          ISZ FOYT
4305  5275          JMP FOYT+1
4306  2274  CLARK,  ISZ FOYT        /STEP POINTER
4307  5674          JMP I FOYT
4310  0000  GURNEY, 0       /TEST 6 BIT CHARACTER AND PRINT
4311  0103          AND P77 /SAVE RIGHT BITS
4312  7450          SNA     /TERMINATE IF ZERO HALF-WORD FOUND
4313  5306          JMP CLARK       /EXIT
4314  1370          TAD M40
4315  7510          SPA     /.LT. 40?
4316  1371          TAD P100        /YES, ADD 100
4317  1120          TAD SP  /ADD 240
4320  4521          PRINT
4321  5710          JMP I GURNEY
            /ENTER WITH ADDRESS OF FLOATING NUMBER IN ACC
            /ROUTINE TYPES PRESENT CONTENTS, AND ACCEPTS A NEW ONE,UNLES
            /ILLEGAL CHARACTER TYPE WHEREUPON OLD CONTENTS RETAINED
4322  0000  FETCHF, 0
4323  3342          DCA TEMP2
4324  4407          FNTR
4325  5742          FGET I TEMP2
4326  0012          OUT
4327  0000          FEXT
4330  1120          TAD SP
4331  4521          PRINT
4332  4405          JMS I 5
4333  1060          TAD 60
4334  7650          SNA CLA /SKIP IF NUMBER CONVERTED
4335  5722          JMP I FETCHF
4336  4407          FNTR
4337  6742          FPUT I TEMP2
4340  0000          FEXT
4341  5722          JMP I FETCHF
            TEMP2,  /TEMPORARY STORAGE
4342  0000  RINDT,  0
4343  4511          DO
4344  4754          JMS I PHILLP
4345  4407          FNTR
4346  0010          FNEG
4347  1505          FADD I I1
4350  6505          FPUT I I1
4351  0000          FEXT
4352  4512          CONT
4353  5742          JMP I RINDT
4354  4441  PHILLP, PHILL
```

```
                /PAGE 4

                /POINTERS & CONSTANTS
4355   4545     TWOP,     TWO
4356   4562     FWHMP,    FWHM
4357   4565     SIGP,     SIGMA2
4360   4570     HITEP,    HITE
4361   4573     AP1,      AT
4362   0002     KONST1,   2
4363   2265               2265
4364   0000               0
4365   0002     KONST2,   2          /SQT(2*PI) = 2.50713(10)
4366   2404               2404
4367   0000               0
4370   7740     M40,      -40
4371   0100     P100,     100


                /SUBROUTINE TO SUBTRACT GUASSIAN, USING PARAMETERS
                /OBTAINED IN GHILL.

                *GHILL+200
4400   0000     AMON,     0          /SUBTRACT GUASSIAN
4401   1237               TAD RUBY
4402   3216               DCA JONES
4403   1240               TAD RUBY+1
4404   3217               DCA JONES+1
4405   4540 [4532] GRANT, JMS I 140 [132]   /JMS MAXIM
4406   4407               FNTR
4407   5020               FGET 20 /GET MAX
4410   4023              FDIV 23 //1024
4411   6026               FPUT 26 /SCALING FACTOR
4412   0000               FEXT
4413   4511               DO
4414   4241               JMS     PHILL   /EVALUATE THE GUASSIAN AT THE POINT
4415   4407               FNTR
4416   0010     JONES,    FNEG
4417   1505               FADD I I1       /SUBTRACT FROM VALUE
4420   4026               FDIV 26 /SCALE FOR DISPLAY
4421   0000               FEXT
4422   4513               FIXX
4423   7300               CLL CLA /PUT IN ZERO IF ERROR
4424   3506               DCA I I2
4425   4512               CONT
4426   5627               JMP I .+1
4427   0210               210
4430   0000     SRTEES,   0          /DISPLAY GUASSIAN ONLY
4431   1236               TAD P7
4432   3216               DCA JONES
4433   1236               TAD P7  /P7 IS FLOATING NOP
4434   3217               DCA JONES+1
4435   5205               JMP GRANT
4436   0007     P7,       7
4437   0010     RUBY,     FNEG
4440   1505               FADD I I1
```

```
                   /PAGE 5

4441  0000  PHILL,  0        /COMPUTE VALUE OF GUASSIAN AT I3,
                    /LEAVE RESULT IN FACC
4442  1107          TAD I3
4443  4516          FLOTE
4444  0044          FACC
4445  4407          FNTR
4446  2373          FSUB    AT
4447  0001          SQUARE
4450  4365          FDIV    SIGMA2  /(X-A)↑2/2*SIGMA
4451  3337          FMPY LG2E       /STARTING EXP ROUTINE
4452  6351          FPUT X1
4453  0000          FEXT
4454  4513          FIXX
4455  7300          CLA CLL /PUT IN ZERO, IF ERROR
4456  3350          DCA FLAG2
4457  1350          TAD FLAG2
4460  4516          FLOTE
4461  0044          FACC
4462  4407          FNTR
4463  6354          FPUT XSQR
4464  5351          FGET X1
4465  2354          FSUB XSQR
4466  6351          FPUT X1
4467  3351          FMPY X1
4470  6354          FPUT XSQR
4471  1334          FADD D1
4472  6357          FPUT TEMPE
4473  5331          FGET C1
4474  4357          FDIV TEMPE
4475  2351          FSUB X1
4476  1323          FADD A1
4477  6357          FPUT TEMPE
4500  5326          FGET B1
4501  3354          FMPY XSQR
4502  1357          FADD TEMPE
4503  6357          FPUT TEMPE
4504  5351          FGET X1
4505  4357          FDIV TEMPE
4506  3345          FMPY TWO
4507  1342          FADD ONE
4510  0000          FEXT
4511  1350          TAD FLAG2
4512  1044          TAD 44
4513  3044          DCA 44
4514  4407          FNTR
4515  6351          FPUT X1
4516  5342          FGET ONE
4517  4351          FDIV X1
4520  3370          FMPY HITE       /HITE*EXP(-((X-A)↑2)/2*SIGMA)
4521  0000          FEXT
4522  5641          JMP I PHILL
```

```
                /PAGE 6

                /CONSTANTS FOR EXP ROUTINE
4523  0004  A1,     4
4524  2372          2372
4525  1402          1402
4526  7774  B1,     -4
4527  2157          2157
4530  5157          5157
4531  0012  C1,     12
4532  5454          5454
4533  0343          343
4534  0007  D1,     7
4535  2566          2566
4536  5341          5341
4537  0001  LG2E,   1
4540  2705          2705
4541  2435          2435
4542  0001  ONE,    1
4543  2000          2000
4544  0000          0
4545  0002  TWO,    2
4546  2000          2000
4547  0000          0
4550  0000  FLAG2,  0

                /TEMPORARY STORAGE FOR EXP ROUTINE
4551  0000  X1,     0
4552  0000          0
4553  0000          0
4554  0000  XSQR,   0
4555  0000          0
4556  0000          0
4557  0000  TEMPE,  0
4560  0000          0
4561  0000          0
                /PARAMETERS FOR GUASSIAN
4562  0000  FWHM,   0
4563  0000          0
4564  0000          0
4565  0000  SIGMA2, 0
4566  0000          0
4567  0000          0
4570  0000  HITE,   0
4571  0000          0
4572  0000          0
4573  0000  AT,     0
4574  0000          0
4575  0000          0
```

A	0026
ABFR	2071
ABFRA	2135
ADBFRA	2123
ADDBFR	2070
ADDRS	0137
AGAIN	0301
AGAINP	0550
AMON	4400
AP	0133
AP1	4361
AREA	0755
AT	4573
A1	4523
B	0031
BASE1	0444
BASE2	0445
BASE3	0250
B1	4526
C	1704
CHAR	0057
CLARK	4306
CNTR	0446
CONT	4512
CONTNU	0112
CP	0135
CR	0246
CRLF	4170
CRLFP	4273
CR1	4176
CSHIFT	2054
C1	4531
C13	0525
D	0714
DATAIN	0701
DO	4511
DOIT	0400
DONE	0522
DU	0111
D1	4534
E	0724
EFF	0660
ELL	0600
ENDM	0753
ENDN	1723
EQ	0117
EWE	0605
FACC	0044
FAREA	0034
FAST	3056
FETCH	0617
FETCHF	4322
FINE	0467
FINIS	0473
FIX	0475
FIXP	0113
FIXX	4513
FLAG2	4550
FLOAT	0447
FLOTE	4516
FLOTEP	0116
FNEG	0010
FNTR	4407
FOYT	4274
FWHM	4562
FWHMP	4356
F1024	0023
GET	0557
GETP	0131
GHILL	4200
GIANTS	0366
GO	0512
GRANT	4405
GURNEY	4310
HITE	4570
HITEP	4360
IMAX	0125
IN	0011
INPUT	4531
I1	0105
I2	0106
I3	0107
JONES	4416
KONST1	4362
KONST2	4365
L	0114
LAST	2155
LBFR	0124
LBFRA	2072
LF	0247
LF1	4177
LG2E	4537
LINE	4673
LINES	0331
LOOK	0204
LOOKY	0214
LOOP	0432
LOOP1	0226
L1	0126
MAX	0020
MAXADD	0110
MAXIM	0732
MBFR	2067
MBFRA	2112
MIF	6544
MONTR	7577
MP	0132
M13	0526
M4	0130
M40	4370
M44	0466
NORM	1707
NP	0134
OK	1667
ONE	4542
ORDI	0370
OUT	0012
OUTP	0121
OUTPUT	4406
PEFM2	0551
PHILL	4441
PHILLP	4354
PRINT	4521
PSHIFT	2063
P100	4371
P177	0554
P200	0553
P2000	0100
P4	0102
P400	0101
P6	0104
P6041	2065
P7	4436
P7146	0555
P7152	0556
P7345	2062
P7351	2064
P77	0103
Q	0177
QUEST	0173
R	0636
READIN	0200
REED	0706
RINDT	4342
RP	0136
RUBY	4437
S	1600
SCALE	1701
SHIFT	0123
SIGMA2	4565
SIGP	4357
SLOPE	0031
SLOWS	0546
SP	0120
SPEEDS	0530
SQUARE	0001
SRTEES	4430
SYMGEN	5000
T	2000
TEMP	0527
TEMPE	4557
TEMP1	0552
TEMP2	4342
TITLES	0360
TP	0251
TWO	4545
TWOP	4355
TYPE	4274
U	0115
UBFR	2066
UBFRA	2101
UMAX	0612
U1	0127
VALUP	0367
X	0122
XAXIS	0330
XSQR	4554
X1	4551