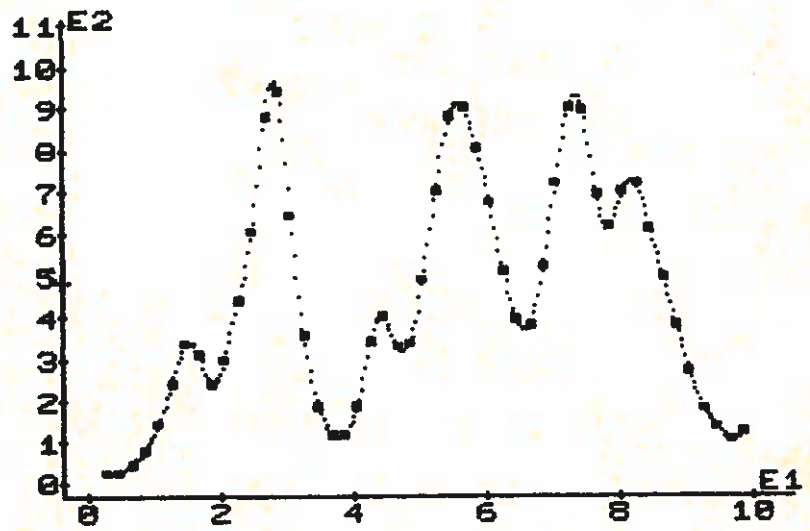
SINGLETON

Interactive Microware, Inc.
P.O. Box 771
State College, Pa 16801
(814) 238-8294

CURFIT
CURFIT2

CURVE FITTER

By Paul K. Warme



Interactive Microware, Inc.
P.O. Box 771
State College, Pa 16801
(814) 238-8294

C U R V E F I T T E R

By Paul K. Warme

INTERACTIVE MICROWARE, INC.

Interactive Microware, Inc.
P.O. Box 771
State College, Pa 16801
(814) 238-8294

NOTES ON PROGRAM LOADING

All Interactive Microware programs are supplied either in DOS 3.2 format or in a special "double boot" format which can be booted on either a DOS 3.2 (13-sector) or DOS 3.3 (16-sector) disk system. The label on the disk indicates whether the disk is DOS 3.2 or DOS 3.2/3.3 (double boot) format. If you are using the DOS 3.3 system, the programs can be converted from the DOS 3.2 format, as supplied, to a DOS 3.3 format disk by using the MUFFIN program (on your System Master disk). For your convenience, the programs are not copy protected; however, please note that the programs are copyrighted, so it is illegal to make copies, except for your own use.

DOS 3.2 USERS:

You should be able to boot, load, or copy programs from any of our disks labelled DOS 3.2 or DOS 3.2/3.3.

DOS 3.3 USERS:

If the disk is labelled DOS 3.2, you must boot your system with the BASICS diskette or mount the System Master disk and type BRUN BOOT13. Then, when the prompting message on the screen tells you to mount the DOS 3.2 (13-sector) disk, insert our program disk and press RETURN.

If the disk is labelled DOS 3.2/3.3, turn off the computer and insert our program disk and turn the computer on. The DOS 3.2 operating system is first loaded from our disk, and then the HELLO program is run on a DOS 3.2 system.

To convert our programs to DOS 3.3 form, boot the DOS 3.3 System Master disk, type BRUN MUFFIN, and follow the instructions to copy the desired programs to a disk that has been INITialized in DOS 3.3 format.

FEATURES OF CURVE FITTER

Curve Fitter makes it easy for scientists to fit a wide variety of curves to their experimental data and select the curve that best fits their results. Data may be input from the keyboard, from a disk file or from a special sensor, such as a spectrophotometer or pH meter. After scaling the raw data or adding a constant offset or converting to log form, N-point averaging may be carried out to reduce the number of points or N-point smoothing may be used to average noise. Then, a curve may be interpolated between the data points, using the polynomial, cubic spline or Stineman interpolation method. Alternatively, least squares fitting may be used to produce the standard curve, using a polynomial, geometric or exponential least squares method. Any or all of these methods may be applied, in turn, to the data in order to select the best fitting method. Finally, unknown samples may be evaluated by interpolating values from the fitted curve.

Other features of Curve Fitter that make it especially useful are:

* Demonstration files on disk make it easy to explore the capabilities of this system.
* Data may be entered as X,Y pairs or as Y values at fixed X intervals.
* All working files may be saved on disk, thus making it possible to repeat the run later or transfer results to other programs, such as Scientific Plotter.
* The known points and fitted curves are nicely plotted using high resolution graphics (280 by 192 points).
* The axes are automatically scaled and numeric labels are printed along the axes.
* Four different plotting symbols may be used to distinguish multiple curves on the same graph.
* A single key stroke switches between text and graphics modes, so that you can alternate between viewing the questions or the graph.
* The switch from text to graphics mode and back is done automatically whenever the graph is being updated, so that you can see what is going on.

The best feature of Curve Fitter is its finely-tuned user interface. Many convenient features have been built into this program from the ground up. The allowable range of input values is indicated and a warning message is printed if you mistakenly enter an improper answer. At any time, you may save the working files on disk and repeat the fitting procedure. All previously selected options automatically become the defaults, so that you can quickly step through the program up to the point where a change is required. All of the options selected during the fitting procedure may be saved on the disk as a format file, which may be used for future runs on similar data. With all of these features, you will not be surprised to learn that the

author of this program is a scientist who understands your needs for curve fitting tools.

HOW TO PRODUCE A BACKUP DISK

Although this program is copyrighted, you have permission to make copies for your own use on a single machine. Thus, your first action should be to copy the master disk to another disk as follows:

1. Mount the master disk and type RUN CURFIT, and then press the RETURN key (from now on, you should always press RETURN after entering your command or response). After a short time, a copyright notice will appear and then the screen will be erased and the question, "READ FORMAT FILE NAME()? <NONE>" will appear. Now, type control Q (hold down the CONTROL key and press the Q key) to stop the program.

2. Mount the slave disk and type SAVE CURFIT. After the program has been saved on disk, type BSAVE FITCHAR,A$3F00,L$F2 in order to save the machine language part of the program.

3. The demonstration plot files may be saved on your slave disk later as described below.

METHODS OF USER INTERACTION

Before you actually use this program, there are a few things that you should know about. The program will guide you through the fitting procedure by printing a series of prompting questions. Each question is followed by another clause within parentheses (). This clause specifies the range of permissible responses, based on previous information. If you type a response outside this range, the program will ring the bell and print "INVALID ENTRY; CHECK (RANGE)". Then, the previous prompting question will be printed once more and you will be allowed to type a different response. If nothing is printed within the parentheses, your response is unrestricted. Each prompting question also contains a third clause, enclosed in <> brackets. This indicates the default value, which is initially blank, but later, it will contain the most recent response to this question. Default values may also be read from a file stored on disk. If you wish to accept the default value, just press the RETURN key. Otherwise, type a different response, followed by RETURN. This method of saving the last response as the default value will save you a great deal of typing. Of course, if nothing is printed within the <> brackets, no default value is available and you must type some valid response.

CONTROL CHARACTERS FOR EDITING AND PROGRAM CONTROL

As summarized in flowchart J, a number of control characters cause special actions by the program.

The back arrow key (Control H) deletes the last character that you typed and Control X deletes the entire input line.

Control G switches to Graphics mode so that you can examine the graph and Control T switches back to Text mode.

Control Y causes the program to call a user-supplied subroutine at lines 2900-2999 in order to read a value from a special sensor, such as a spectrophotometer or pH meter or anything else that you can rig up. The sensor value must be returned in variable V0; your subroutine may scale V0 as appropriate. Currently, subroutine 2900 always returns V0 = 0. When your subroutine returns, the value of V0 is printed on the input line, just as if you had typed it. If you don't like that value, you may backspace over it and type Control Y again to input a different sensor value. When the sensor value is satisfactory, press RETURN to enter it.

Control P allows you to select a Print device for subsequent output. After the Control P, type the single-digit number denoting the slot occupied by the printer controller or type 0 to restore normal screen output.

Control A switches the program into the Automatic mode of operation. In Automatic mode, the default values are used instead of user input. This mode ends when any error occurs or when a default value is outside the range of permissible values or when you type any key. The program also switches out of Automatic mode when the SAVE FILES segment of the program is reached.

Control Z may be typed at any time to escape from the normal sequence of prompting questions and jump to the SAVE FILES section of the program.

Control Q is the signal to Quit the program entirely and return to BASIC.

All other Control keys are ignored during user input; however, Control S may be used to stop screen output and Control C will stop the program (unless the program is waiting for user input; in this case, use Control Q to stop the program).

HOW TO RUN THE DEMONSTRATIONS

The master disk contains five demonstrations that will introduce you to some of the features of the Curve Fitter program. It is suggested that you try at least one of these demonstrations before attempting to fit your own data. Most of us learn faster by example than by reading a manual. It's also

more fun to learn by example.

To start the program, mount the master disk and type RUN CURFIT or, if the program has already been loaded, just type RUN. After the copyright notice is printed and the screen has been cleared, the program prints:

READ FORMAT FILE NAME() ? <NONE>

Since nothing is printed in the range field within (), your input is unrestricted. If you simply press RETURN, the default value <NONE> will be entered as your response and this will mean that you do not want to load a format file from disk. However, for this demonstration, you should type HTWTFORM, so that the format file for the first demo will be read from the disk. In other words, the default values within <> for all of the other questions will be read from the disk file called HTWTFORM. Thus, you may just press RETURN after each question and the program will automatically fit the curve. Since our objective is to learn how to use the program by example, you should read the default values printed after each question and try to understand how these responses affect the results. If you don't understand something, you may look up that question in the table of contents and read the explanation given later in this manual. Alternatively, you may just press RETURN and see what happens. At any time, you may type Control G to see the graph or type Control T to see the questions. If you want to ignore the questions and just watch the action, type Control A to enter Automatic mode. Remember that Automatic mode can be stopped by typing any key.

The data file HTWTDATA contains the heights and weights of twelve men. This first demonstration performs linear least squares fitting and plots a fitted straight line of 100 points. Then, the unknowns file called HTWTUNKN is read in and the interpolated weights are printed for men of heights ranging from 61 to 73 inches.

When the program prints ********** SAVE FILES, it is time to stop pushing buttons and pay attention to what we are doing. The program next prints:

WRITE STANDARDS FILE NAME()? <NONE>

If you want to save this demonstration, mount your slave disk and type HTWTDATA. Note that if you try to write a file on a write-protected disk, the program will halt because of a WRITE PROTECTED error.

The next question is:

WRITE FITTED FILE NAME()? <NONE>

You should just press RETURN here, since the fitted file can be regenerated easily by the program.

Now, the program asks:

WRITE UNKNOWNS FILE NAME()? <NONE>

Since the format file for this demo expects to evaluate unknowns stored in a disk file called HTWTUNKN, type this name if you want to save these data.

At this point, the program will print:
WRITE FORMAT FILE NAME()? <NONE>
Type HTWTFORM here, so that all of the responses to questions will be saved on your slave disk. Although the naming of files is completely arbitrary, you might consider adopting our convention of using the FORM suffix for format files, DATA for standards, FIT for fitted curves and the UNKN suffix for unknowns.

The other four demonstrations may be run in a similar fashion. PEAKFORM shows the results of averaging and smoothing some data (PEAKDATA) collected from an analog to digital converter. A cubic spline interpolation is then used to draw a smooth curve through the scattered points.

STINEFORM uses Stineman interpolation to fit a curve (STINEDATA) having a sharp change of slope and a broad plateau at its peak. You might want to try fitting a cubic spline curve to this data in order to observe how polynomials tend to overshoot in such cases. A polynomial of degree four does quite a respectable job of fitting the data, though. Go ahead and experiment a little.

GEOMFORM uses geometric least squares to fit the data in file GEOMDATA. You could also try an exponential least squares fit or polynomials of degree two or higher to fit these data. Normally, the standard error of estimate is a good indicator of the quality of the fit. However, since the geometric and exponential least squares methods use LOG transformations, their standard error values cannot be compared directly to those for polynomial fitting.

CUBICFORM uses cubic polynomial least squares to fit some data (CUBICDATA) for the specific heat of water (Y values) as a function of temperature (X values). This problem is discussed in reference 2, pp. 326-330. In trying polynomials of degree 2 through 6, I found that the standard error of estimate decreased from 9.55E-4 to 2.67E-5 and the apparent fit improved considerably.

DETAILED EXPLANATIONS OF PROGRAM OPTIONS

The Table of Contents lists each of the prompting questions asked by the Curve Fitter program, in the same order that they are encountered in the programs. On the page number listed after each question, you will find a detailed description of the effects of your responses on the outcome. Note that some of the questions are skipped over, depending on previous options that you have selected.

USING THE FLOWCHARTS

The easiest way to grasp the overall flow of the program is to look at flow chart A, which shows the major parts of the program and will direct you to one of the more detailed flowcharts B through H for further information. Flowcharts I and J pertain to user input; you should bear in mind that they are relevant whenever you are typing any response. Also, note that the control characters in flowchart J take effect immediately after you type them and they may be typed at any point in the input line. However, these control characters are not included as part of your actual response to the question.

READ FORMAT FILE NAME

If you want to load a format file previously stored on the disk, type its file name. Optionally, you may append a volume, slot or drive number after the file name (this is also true for all other file names used by this program). The default file name for this question is always <NONE>, so if you only type RETURN, no format file will be read and the default values will be whatever was previously answered for each question. If you have just started the program by typing RUN, all of the default values are blank, so you will have to type some valid response to each question.

INPUT STANDARDS (FLOWCHART B)

All of the data for the standards, fitted curve and the unknowns are stored in a single array called D(1000). This approach minimizes storage requirements. If necessary, you may change the dimension of this array in program line 10, but you must also make the same change in the value of MX at the end of line 10.

Although the original input data for the standards may consist of 1, 2 or 3 values per point (see below), the storage format is changed in the SCALE STANDARDS segment, so that separate X and Y values are stored for each standard point. Also, the error bar values are discarded, so that after scaling, each standard point uses two elements of the D array. The X values are stored at element 0 and other even-numbered elements, while the Y values are stored at odd-numbered elements starting with D(1).

The fitted data points are stored immediately following the scaled standards. Only the Y value is stored for each fitted point, because the X interval is always a constant.

The data for the unknowns is stored right after the fitted data, and either one or two elements in array D are required for each unknown, depending on whether a different multiplier value is required for each unknown.

In summary, the total size required for array D may be calculated by doubling the number of standards, adding the number of fitted points and then adding the number of unknowns (times two if multipliers are needed).

SAME STANDARDS (Y:N)? <N>

If you have previously entered some data for standards, this question will be asked. The default is always set to No. You should type Y here if you wish to retain the last set of standards and you don't want to make any deletions or changes of scale. In this case, the program skips to the LIST STANDARDS question.

X,Y PAIRS (Y:N)

Type Y here if both X and Y values are to be entered for each point. The next two questions are asked only if you type N.

FIRST X VALUE

If you answered No to the last question, it means that only Y values will be entered, with constant intervals between X values. In this case, you should now type the X value of the first point.

X INTERVAL

Here, you should give the value of the constant interval between X values.

ERROR BARS (Y:N)

Although error bars are not used by this program, this option is included in order to maintain compatibility with data files used with the Scientific Plotter program sold by Interactive Microware. If you want to use this program to enter raw data which will later be plotted by Scientific Plotter, you may type Y to this question. In this case, you will be asked to enter a "+/- ERROR" value after each Y value.

DISK, KEYBOARD OR SENSOR INPUT

There are three ways to input data for standards or unknowns:

You may type the values on the keyboard, you may read the data from a disk file, or you may input values from a special sensor, such as a spectrophotometer or pH meter. As explained above (see CONTROL CHARACTERS FOR EDITING AND PROGRAM CONTROL), sensor input is a special case of keyboard input. You merely type Control Y to enter the sensor value on the input line.

READ FILE NAME

If you wish to input data from a disk file, type the file name here. Otherwise, type NONE in order to skip over this part of the program.

It is easy to create data files that are compatible with Curve Fitter. Just be sure that the data are stored as a text file in the format described above under the headingINPUT STANDARDS. Data stored on the disk by the APPLAB Data Acquisition System, Scientific Plotter and other programs available from Interactive Microware, Inc. are directly compatible with Curve Fitter.

The following general method may be used to store data on disk for later fitting by this program:

```
10 DIM D(1000)
1000 CD$=CHR$(4):F$="FITDATA"
1010 PRINT CD$;"OPEN ";F$:PRINT CD$; "WRITE ";F$
1015 PRINT NUM: REM NUMBER OF DATA VALUES
1020 FORI=0 TO D(0): PRINT D(I):NEXT
1030 PRINT CD$; "CLOSE ";F$
```

Between statements 10 and 1000, you could insert any program that stores data values in array D.

FIRST POINT TO BE USED

This question is asked only if you are reading data from a disk file. In some cases, you will only want to plot some of the points from a larger data file. If so, enter the number of the first point to be read. Enter 1 (one) to start reading at the first value. Notice that since you have already indicated whether X, Y pairs or error values are stored in the data file, the program can automatically calculate the position of the first requested data point in the file.

INTERVAL BETWEEN POINTS

If you are reading data from a disk file, this question allows you to skip over some of the points and read only selected points. Type 1 if you want to read every point, type 2 to read every second point and so on.

X VALUE, Y VALUE, +/- ERROR

If you have selected keyboard or sensor input, the program prints POINT 1, and then asks for X, Y and +/- error values, as appropriate. If you are inputting unknowns, the +/- ERROR message will be replaced by MULTIPLIER. The default value will be the value previously stored at that position in the D array. The purpose of this feature is to make it convenient for you to edit your data. If the default value is correct, just press RETURN.

If you type Control Y, the sensor value calculated by subroutine 2900 will be printed as your response and you may simply press RETURN. If you type 10000, that point will be deleted later. When you type 9999, this signals the end of the input data. If you really want to include the value 9999 or 10000 in your data, type 9999.001 or 10000.001 instead.

It is possible to use this keyboard input routine to edit data for standards or unknowns entered previously. To do this, type Control Z at any point and skip through the program until you come to the INPUT STANDARDS or EVALUATE UNKNOWNS segment. Then, answer the questions as though you want to type the data on the keyboard. Since the previous values are now the defaults, you can scan through the data by pressing RETURN after each question (or type Control A), until you come to a value that needs to be edited. After all corrections have been entered, type Control A to scan to the end of the data.

SCALE STANDARDS (FLOWCHART C)

In this segment of Curve Fitter, the standards data will be rearranged in the D array, sorted by ascending X value and scaled according to your instructions. Also, deleted values and error bar values will be discarded. If you want to save your original input data on disk, this is your last chance, as announced by the message, TYPE CONTROL Z NOW TO SAVE RAW DATA. The program goes on to print the next question, but you can type control Z anytime before pressing RETURN.

INTERCHANGE X & Y DATA (Y:N)

The reason for this question is that the least squares fitting routines always assume that the major errors are in the Y values and that the errors in the X values are negligible. Thus, if the X values in your input data are most likely to be in error, you should answer Y.

From this point on, your standards data are always stored in the D array as X, Y pairs, even if you entered only Y values at

constant X intervals. This makes it possible to delete bad points (outliers) in the latter case.

X SCALE FACTOR

You may now enter a scale factor, which will multiply each X value. If no scaling is required, type 1 (one).

X OFFSET

This allows you to add a constant offset to each X value. For instance, if your X values range from 10000 to 10001, you could enter -10000 here so that the plotted values will range from 0 to 1. If no offset is desired, type 0. The plotted X value is now calculated by the formula:

X=XSCALE*(XINPUT+XOFFSET)

X LOG SCALE

If you type LOG or LN in response to this question, the base 10 or base e logarithm of X will replace the value of X calculated above. If no logarithmic conversion is desired, type N as your answer.

Y SCALE FACTOR

Y OFFSET

Y LOG SCALE

These conversions on the Y values are carried out in exactly the same way as those of the X values (see above).

LIST AND PLOT STANDARDS (FLOWCHART D)

LIST STANDARDS (Y:N)

At this point, you may get a listing of the transformed data by answering Y. To direct the output to a printer, type Control P, followed by the slot number of the printer controller, before pressing RETURN.

PLOT STANDARDS (Y:N)

Unless the standards have already been plotted, you should

answer Y. The program automatically calculates suitable numeric labels for the X and Y axis. At the top end of the Y axis and at the right end of the X axis, the letter E is printed, followed by a number. This stands for the Exponent of ten for each of the labels on the axes. For example, the label 12 with exponent E2 would have the value 1200. Likewise, a label of 9 with exponent E-1 means 0.9 and 8E0 means 8.

Next, the program plots the standards, always using the small box symbol. Bear in mind that you can view the Graph by typing Control G and return to Text mode by typing Control T.

AVERAGING AND SMOOTHING (FLOWCHART E)

POINTS AVERAGING

The usual reason for averaging your data is to decrease the number of standards, while retaining the "flavor" of all of the data points. If you don't want to reduce the number of points and your primary purpose is to smooth random errors, you should use the smoothing option, instead. Of course, you could also reduce the number of standards by just skipping some of the values during data input, but this ignores all information about intermediate values.

If you type 0 or 1 as your response, no averaging will be done. If you type 2, the first and second X and Y values will be averaged, so will the third and fourth values, and so on. The number of points will be divided by 2. Similarly, if you answer 3, groups of 3 points will be averaged and the number of points is divided by 3. The original values are not lost, since the averaged values are stored above the standards in the D array. Each averaged point requires two elements in the D array and if insufficient space is available, the message OUT OF ROOM is printed and the averaging process terminates prematurely.

After the averaging process is completed, the averaged values are plotted with + symbols to distinguish them from the original points, which are denoted by square symbols. You should type Control G to see the graph.

MAKE PERMANENT (Y:N)

You now have the choice of discarding the averaged points (by typing N) or accepting them as the new standards (by typing Y). If you type N, the averaged points are erased and the program returns to the # POINTS AVERAGING question, so you can make a different choice. If you type Y, the old standards are erased and the averaged points take their place in all subsequent calculations. You will note that #/2 points at each end of the X

axis disappear after averaging. This is a natural consequence of the fact that both the X and Y values for each point are averaged.

POINTS SMOOTHING

The purpose of smoothing is to reduce random variations in the data values by averaging two or more successive values. You will find that small peaks in the graph tend to disappear after averaging. It is up to you to decide whether the small peaks are just "noise" or whether they are real effects.

If you respond by typing Ø or 1, no smoothing will be done and the program will skip to the Curve Fitting segment. If you type 2, the first and second X and Y values are averaged, and so are the second and third, third and fourth, etc. Note that this reduces the number of points by #-1, where # is your response to the # POINTS SMOOTHING question. Here's another example: if you type 3, the first, second and third X and Y values are averaged, as are the second, third and fourth, etc. The original standards are not lost, because the smoothed points are stored above the standards in the D array. Each smoothed point takes up two elements unless there is no more room in the D array; in this event, the OUT OF ROOM warning is printed and the smoothing process ends immediately.

After smoothing, the smoothed values are plotted using the + symbol, which is easy to spot among the box symbols used for the standards. Type Control G to see the graph.

MAKE PERMANENT (Y:N)

If you type N here, the smoothed points are erased and the program returns to the # POINTS SMOOTHING question so that you can make a second choice. The answer Y causes erasure of the previous standards and the smoothed points become the new standards. As a consequence of the fact that both X and Y values of each point are averaged, (#-1)/2 points disappear at each end of the X axis. If you wish to keep the range of X values the same as before, you could skip to the INPUT STANDARDS part of the program and append the original standards at the extremes of the X axis. Of course, these points no longer carry the same weight as the smoothed points.

CURVE FITTING (FLOWCHART F)

NEXT SYMBOL

This allows you to select one of three different symbols for

plotting the next fitted curve. Symbol 2 is a point (.), symbol 3 is a plus sign (+) and symbol 4 is a cross mark (x). The standards are always plotted using the square, symbol 1, so that they can be distinguished from the fitted curve. Generally, symbol 2 is the best choice, but you may want to choose a different symbol so that you can see the differences among several alternative fitted curves.

INTERPOLATION OR LEAST SQUARES FITTING

Curve Fitter allows you to produce a curve through your points in two conceptually different ways: interpolation and least squares fitting. An interpolated curve always exactly passes through each standard point, whereas a least squares fitted curve minimizes the sum of the squares of the distances between the fitted curve and the standard points. In other words, a least squares fitted curve passes as close as possible to the standard points, subject to the limitations of the equation selected for fitting. If you are confident that your standards are accurate, interpolation is the reasonable choice (type I). On the other hand, if you assume that there are random errors in the Y values, least squares fitting is the logical choice (type L). Other considerations affecting this decision are discussed below under the heading, THEORETICAL CONSIDERATIONS OF CURVE FITTING.

If you type N to this question, the program skips to the SAVE FILES segment of the program. If you type I for Interpolation, the program prints AVERAGING DUPLICATE VALUES because the interpolation methods do not allow more than one point with the same X value. However, the least squares methods do allow duplicate X values and normally, you will want to include the influence of duplicates in the fitting. Thus, if you plan to try both interpolation and least squares fitting, it is best to do least squares fitting first, before duplicates are averaged.

OF POINTS ON CURVE

Your response to this question determines how many points will be plotted for the fitted curve. It is a good idea to request relatively few points at first, while you are searching for the best fitting method, because the calculation time increases for more points, especially in the case of polynomial fitting. However, the evaluation of unknowns becomes more accurate with increasing number of points (see EVALUATE UNKNOWNS). Thus, after you select a fitting method, you should repeat the calculation using a larger number of points, consistent with the desired degree of accuracy.

POLYNOMIAL, CUBIC SPLINE OR STINEMAN INTERPOLATION

If you have chosen I for Interpolation above, you now have a choice of three methods:

Polynomial interpolation includes linear, quadratic, cubic and higher order polynomial fitting up to degree 6. A polynomial of degree N is calculated for N+1 standards spanning each fitted point. This is the slowest method of the three, particularly for higher degrees.

The Cubic spline method fits a cubic polynomial through each four successive points and saves three coefficients for each pair of standards. These coefficients are stored in arrays A, B, and C, which are dimensioned in line 12. If the number of standards exceeds the value of ABC (set equal to 99, the dimensions of arrays A, B and C at the end of line 12) plus 1, the message TOO MANY STANDARDS is printed and the program returns to the beginning of the CURVE FITTING segment. Although the results obtained with the cubic spline method are very similar to those obtained with the polynomial method using degree 3, the cubic spline method is significantly faster.

The Stineman method of interpolation is described in Creative Computing magazine, July 1980, pp. 54-57. It is especially well suited for cases where an abrupt change of slope occurs in the fitted curve. In these circumstances, a polynomial curve will often oscillate above and below the standard curve in an undesireable fashion. This method is well protected against wild departures from the standard points. In fact, in many cases, the Stineman curve is not very different from linear interpolation. My experience is that a Stineman curve seems to fall somewhere between a linear (straight-line) interpolation and a cubic interpolation. The maximum number of standards to be fitted is again limited to the value of ABC (line 12) plus 1. The calculation time is about the same as that for a cubic spline fit.

In the course of developing this program, Lagrange interpolation and Neville's iteration were also investigated. However, the excessively long calculation times and large storage requirements for these methods limits their usefulness. In addition, the fitted curves were either worse or not significantly better than results obtained using methods included in this program.

DEGREE OF POLYNOMIAL

If you have selected polynomial interpolation, you must now select the degree of the equatin to be used. A degree of 1 yields linear (straight-line) interpolation, as expressed by the formula Y=A+B*X. Here, A is a constant and B is the slope of the line. A degree of 2 produces a quadratic curve, Y=A+B*X+C*X↑2. For each successively higher degree, an additional term is added to the equation, consisting of a new constant multiplied by the

next power of X. The largest degree allowed by the present program is 6, because numerical errors increase rapidly at higher degrees and calculation time becomes excessive. If the number of standards is fewer than 7, the maximum degree allowed is one less than the number of standards.

LINEAR, GEOMETRIC, EXPONENTIAL OR POLYNOMIAL LEAST SQUARES

If you have chosen least squares fitting, you will now select one of these four methods:

Linear least squares fits the best straight line through a set of standards, using the equation Y=A+B*X. Although this is a special case of polynomial least squares with degree 1, it is computationally convenient to treat it separately.

Geometric least squares applies an equation of the form Y=A*X↑B to the standards. This is accomplished by using a log transformation of the former equation, log Y = log A + B*log X. This transformed equation is then fit by linear least squares.

Exponential least squares assumes that the standards conform to an equation of the form, Y = A*EXP(BX). This equation can be linearized by taking the log, yielding log Y=log A + B*X. This transformed equation is then fit by linear least squares.

Polynomial least squares attempts to fit the standards by an equation of the form Y=A+B*X+C*X↑2..., where the number of terms is N+1 and N is the selected degree of the equation. This is done by solving a system of simultaneous equations involving derivatives of the error equation with respect to the N+1 coefficients. The solution is found by the method of Gaussian elimination, as described in reference 2, pp. 155-175 and pp. 324-330.

After completion of the least squares fitting, the coefficients of the equation and several measures of the reliability of the fit are printed. A good discussion of these measures is given in reference 1, pp. 260-275. The coefficient of determination is the proportion of variance of Y that is "explained" by linear regression on X. Its value ranges from 0 to 1; a value close to 1 indicates a strong linear relationship between X and Y, while a value of 0 means there is no linear relationship. The coefficient of correlation is the square root of the coefficient of determination. Again, its value ranges from 0 to 1 and may be interpreted similarly. If its value is 1, there is a perfect linear relationship between X and Y. The standard error of estimate is a measure of how much the values estimated from the fitted curve deviate from those observed. In other words, multiple experimental observations at this value of X are likely to vary this much from the calculated value.

DEGREE OF POLYNOMIAL

If you have selected polynomial least squares, you must now type the degree of the polynomial. In general, you should choose the lowest degree that gives a decent fit. Curve Fitter makes it easy for you to try several possibilities. If you have a good theoretical reason to expect an equation of a particular degree, don't yield to the temptation to choose a higher degree. Remember that with a polynomial of degree N-1, you can exactly fit a group of N points. Thus, the program only allows an N-2 degree polynomial. Also, if the number of standards is fewer than 4, the program prints TOO FEW STANDARDS and forces a different choice of method. The maximum degree allowed is 6, because accuracy is limited at higher degrees and the calculation time becomes excessive.

SATISFACTORY (Y:N)

This question appears after any of the fitting methods has completed its calculations and the fitted curve has been plotted (type Control G to see it). You are the final judge of whether the fit is satisfactory. If you type Y, the present curve will be used for evaluating unknowns.

ERASE FITTED CURVE (Y:N)

If you answered N to the last question, you have the option of erasing the last fitted curve. The program then recycles to the beginning of the Curve Fitting segment so that you may choose a different fitting method.

EVALUATE UNKNOWNS (FLOWCHART G)

The equations used to calculate the fitted curve are suited for calculating Y values for a given X. However, it is often desireable to calculate X values for a given Y and therefore, this program interpolates unknowns directly from the fitted curve, rather than using the exact equations. In other words, unknowns are evaluated by linear interpolation between the closest pair of points on the fitted curve. In regions where the fitted curve is linear (e.g., for the case of linear interpolation or linear least squares), this approach gives exactly the same value as you would obtain from the fitting equation. However, in curved regions of the fitted curve, there will be a small discrepancy. As you increase the number of points on the fitted curve, the potential error is decreased. With 100 or more points on the fitted curve, the error is generally negligible. If greater accuracy is required, you may write a small program to evaluate unknowns by using the

coefficients printed by the program and the appropriate fitting equation. This will also be necessary if you want to extrapolate values outside the range of the standards.

INPUT UNKNOWN VALUES (Y:N)

If you answer N to this question, the program skips to the SAVE FILES segment.

ENTER X OR Y VALUES

Now, you may select the axis of interpolation. Either X or Y values may be entered and the complementary Y or X value will be interpolated from the fitted curve.

MULTIPLIER; TYPE Ø IF VARIABLE

In many cases, the samples for the unknowns are prescaled in some way so that the measured values fall within the range of the standards. If the prescaling multiplier is the same for all samples, type its value. Each interpolated X or Y value will be multiplied by this factor before it is printed (in the VALUE column). If no scaling is required, type 1. If the prescale value varies for different samples, type Ø and then you will be asked to enter a separate multiplier for each unknown.

X VALUE, Y VALUE, MULTIPLIER

Input of unknowns is the same as inputting standards (see DISK, KEYBOARD OR SENSOR INPUT). Again, the data may be entered from disk, keyboard or a special sensor (type Control Y) and the previous values in the D array become the defaults for subsequent entries. As before, you may delete an entry by typing 10000; the signal that you have entered the last value is 9999. If the D array is full, the OUT OF ROOM message will be printed and no more input will be allowed. However, you will have a chance later to enter more unknowns.

READY FOR LISTING

You may want to turn on your printer at this point by typing Control P and the slot number. The only acceptable response to this question is Y, so type Y when you are ready for a listing of the results. The number of the unknown is printed in column 1, followed by the input value, the calculated value and finally, in column 4, the multiplier. Each input value is converted by the scale factor and offset previously specified for the standard and a log conversion is performed if the standards were so transformed. Next, the scaled input value is compared with the

fitted curve. If it is outside the range of the fitted curve, the OUT OF RANGE message is printed. Otherwise, the corresponding value is interpolated on the other axis. This result is multiplied by the specified multiplier and the final result is printed in the VALUE column.

Let's go through a simple example which will help to clarify all of this. Suppose that you have entered the following standards:

POINT	X=	Y=	SCALED X	SCALED Y
1	1	1	2	2
2	2	2	3	4

The last 2 columns show the values actually plotted if we assume that you specified an X offset of 1 and a Y scale of 2. For simplicity, we'll select interpolation with a polynomial of degree one; that is, a straight line from X,Y=(2,2) to (3,4). Next, we'll enter some X values with variable multipliers and calculate the unknown Y values.

POINT	INPUT X	MULTIPLIER	SCALED X	Y=	VALUE
1	1.5	1	2.5	3	3
2	2.0	2	3.0	4	8
3	2.5	3	3.5	OUT OF RANGE	

The printed report for this session would be as follows:

POINT	X=	VALUE	MULTIPLIER
1	2.5	3	1
2	3.0	8	2
3	3.5	OUT OF RANGE	

Note that the X value that is printed is the scaled value, so that you can refer it to the graph.

MORE UNKNOWNS (Y:N)

Now, you have the opportunity to enter more unknowns or edit the previous ones. If you type Y, the program returns to the ENTER X OR Y VALUES question.

SAVE FILES (FLOWCHART H)

By typing Control Z, you may exit to this part of the program at any time in order to save your standards, fitted curve, unknowns or a format file on disk. In all cases below, the file name defaults to NONE, so you may simply press RETURN if you don't want to save a file. This provides an extra measure of protection against accidentally overwriting a file that you wanted to keep. Also note that the automatic mode of operation terminates at this point so that you may save your files as necessary. Typing Control D sends the entire preceding input line to the disk operating system. For example, to list the catalog of drive 2 in slot 6, you should type CATALOG,S6,D2 followed by Control D. Thereafter, all disk I/O will use that same drive.

WRITE STANDARDS FILE NAME

If you enter a valid file name here, your standards will be saved on disk. The first value in the text file is the number of points times 2, followed by the X and Y values for each standard. Remember that if you have scaled or offset the data or converted it to log form, the data will be stored in the converted form. To circumvent this, type Control Z immediately after entering the data.

WRITE FITTED FILE NAME

You may also save the fitted data. This option is useful for transferring the results to the Scientific Plotter program for producing a professional-looking graph. The data is saved as a text file containing first the number of points and then the Y value for each point. You will recall that fitted data is always produced at a fixed X interval, starting with the X value of the standard having the smallest X value. For future reference, the values of FIRST X VALUE and X INTERVAL are printed. You should remember these values for subsequent entry in the Scientific Plotter program.

WRITE UNKNOWN FILE NAME

This option permits you to save the data for the unknowns in the same form as you entered it (that is, unscaled). In this form, you may read the unknowns from disk instead of reyping them at a later time. The text file will contain the number of unknowns as its first value and then the input values. If you have entered multipliers for each unknown, the first value is the number of unknowns times 2, followed by the input value and then the multiplier for each point.

WRITE FORMAT FILE NAME

Now, you may save the current list of default responses to all questions as a format file. Later, if you want to carry out a similar fitting operation, just enter the name of this format file after the READ FORMAT FILE NAME question and from there on, you may simply press RETURN after each question where the default value is satisfactory.

ERASE GRAPH (Y:N)

If you answer Y to this question, the screen will be erased and the program will start over. If you type N, the program will start over without erasing the screen. This option is useful if you want to edit the standards (delete or add a point) and then compare the fitted curve to the original curve obtained before

editing. If the range of the X and Y values is not changed by editing, the axes plotted when you answer Y to the PLOT STANDARDS question will coincide with the original axes. If the range is changed by editing, you may still compare the fitted curves if you answer N to the PLOT STANDARDS question. However, in this case, the edited standards and out of range values will not appear on the graph.

USAGE WITH SCIENTIFIC PLOTTER

The formats of the standards file and fitted file are compatible with the Scientific Plotter program, also sold by Interactive Microware. Thus, you may use Curve Fitter to enter data which will later be plotted in final form by Scientific Plotter. The advantage of entering data in Curve Fitter is that you may produce a fitted curve and then plot the fitted curve, together with the standards by reading the saved disk files into Scientific Plotter.

If you enter error bar values in Curve Fitter for later use with Scientific Plotter, you must save the standards at the very beginning of the Scale Standards segment, because after that, the error values are deleted. Also, if you have entered only Y values, the data are expanded to include both X and Y values in the D array. Thus, the format of the standards file saved on disk will be different after the Scale Standards segment.

THEORETICAL CONSIDERATIONS OF CURVE FITTING

Although it is not possible to cover this large subject in depth in the small space available here, I will include a few guidelines and rules of thumb. A much more detailed treatment will be found in the books listed under REFERENCES, which I have found particularly useful.

The usual reasons for fitting curves to data are to obtain interpolation formulas or calibration curves or to confirm a theoretical relationship or to select an appropriate theoretical model. Another reason comes to mind; a graph usually looks better when the observed points are connected by a curve of some sort. However, we must be careful not to be misled into thinking that we know more than we do about the values lying between the observations. Curves drawn through a small number of observed points without a sound basis in theory may represent only wishful thinking.

AVERAGING AND SMOOTHING

All experimentally measured values contain a certain amount of error, including systematic errors, which affect all measurements in the same way, and random errors, which may either increase or decrease the observed value. It is up to the experimenter to detect and correct systematic errors, but this program can help you to deal with random errors. If we make a large number of "identical" measurements, the average of all measured values will be closer to the "true" value than are most of the individual measured values, because some of the random errors are in the positive direction and others are in the negative direction. In the absence of contrary evidence, we shall assume that the random errors follow a normal distribution, or bell-shaped curve, since this is true of most natural processes.

If the conditions of the experiment are continuously changing there is less justification for averaging, even if we are sure that random errors are present. In this case, we must compare the magnitude of the random errors to the magnitude of the underlying signal. If the signal is changing linearly over the range of averaging, then the average value will be a more faithful representation of the underlying signal than the individual noisy measurements. However, if the signal change is nonlinear, the averaged curve will be distorted to some extent by averaging and thus, averaging should be considered only when the magnitude of the error is greater than the change in the value of the underlying signal. Similar considerations apply to smoothing, which is just an averaging technique that does not reduce the number of data points.

INTERPOLATION

Interpolation methods seek to exactly fit a set of standard points and altogether ignore the possibility of inherent errors in the data. A polynomial equation of degree N can fit N+1 points exactly; moreover, the solution is unique. However, if the data set includes more than N+1 points, the shape of the curve will be affected by the choice of which set of N+1 points are used to define the interpolating curve for a given interval. Insofar as possible, this program uses the same number of points on either side of the two adjacent points defining the interval of interpolation. The first few points and the last few points in the set of standards are a necessary exception to this criterion, so the interpolation may be less accurate at both ends of the fitted curve.

A polynomial curve of degree N can have, at most, N-2 inflection points. Therefore, if it is clear that an inflection point is needed to give a reasonable fit to your standards, you should use a polynomial of degree 3 or higher.

LEAST SQUARES FITTING

Least squares methods are useful in cases where interpolation is inappropriate, due to the presence of relatively large random errors in the measured values. The following important assumptions are common to all of these methods:

1. The X values are free of sampling and measurement errors.
2. The errors in Y values are normally distributed at each X value.
3. The population variance of Y is the same at each value of X.
4. The population mean of Y is a linear function of X.
5. The Y values are randomly sampled at each X and independent of measurements at other values of X.

Under these conditions, minimizing the sum of the squares of the deviations of the observed Y values from the fitted line produces the most probable fit, according to statistics. Regarding the first assumption, a rule of thumb is that "least squares analysis can be used safely if the variance of X is less than a tenth of the average scatter of the X's from their mean" (Ref. 3, p. 32). Assumption 2 is softened by the knowledge that "if the statistical errors in the Y values follow <u>any</u> distribution with constant variance, then [least squares] produces the most probable solution" (Ref. 1, p. 314). Also relevant to assumption 2, if the errors in Y are randomly distributed, a histogram of the frequencies of the magnitudes of the deviations should produce a bell shaped curve. A defect in assumption 3 or 4 may show up in the fitted curve; points toward the extremes of the X axis will tend to deviate more from the fitted curve than points near the center. In other cases, a plot of the magnitudes of the deviations from the fitted curve may show curvature. A breakdown in assumption 5 may manifest itself by a periodic distribution of the standard points about the fitted line; that is, a series of points will all be above the fitted curve and another series of points will be below the fitted curve.

In the cases of geometric and exponential least squares fitting, we actually minimize the sum of the squares of the deviations of <u>LOG</u> Y from the curve, rather than using the deviations of <u>the</u> Y values themselves. Thus, the statistical arguments in favor of least squares fitting are weakened for these cases. Nevertheless, these methods do appear to give good results for some curves. Exponential least squares is suggested when a plot of log Y versus X appears to be linear, whereas geometric least squares works well when a plot of log Y versus log X is linear.

A useful guideline for choosing the form of equation to be used for least squares fitting is that it is wise to choose an equation with the least number of constants. If two different equations give about the same standard error of estimate, choose the simpler one; that is, choose linear over exponential or choose a degree two rather than a degree three polynomial. Note

that the standard error for geometric and exponential fits is affected by the log transformation, so it cannot be compared to the standard errors for other methods.

Curve Fitter provides facilities for editing your data and removing "bad" points (outliers). But how can you decide whether a point is bad? The most immediate indication is that the distance of that point from the fitted curve is much greater than that of the other points. A large deviation can have a significant effect on the fitted curve, because we are minimizing the squares of the deviations. Bad points at the extremes of X will significantly affect the slope of the curve, while bad points near the center of the curve will have less effect on the slope than on the position (height) of the curve. After reviewing the experimental conditions when the bad point was measured, you may be able to justify throwing it out. Physically impossible values or samples measured prior to attainment of equilibrium are clear candidates for elimination. More sophisticated tests for detecting outliers are mentioned in reference 3, p. 127.

ADVANCED FITTING TECHNIQUES

The methods included in Curve Fitter are the most widely used methods and work well for most cases, but there are also many cases where special techniques are required. Sometimes, additional transformations are needed before least squares fitting, such as taking the reciprocal of X or Y or using trigonometric functions of X or Y (see discussions in ref. 2, pp. 330-335 and ref. 3, pp. 19-24). In principle, linear least squares can be used whenever the underlying equation can be transformed so that it is linear in the coefficients. When this is impossible, nonlinear least squares methods are required (see ref. 3, pp. 267-337).

Another limitation of the methods presented here is that they do not work for equations containing more than one independent variable. In such cases, one can often design experiments such that only one variable changes while all others remain constant. Under these conditions, our methods are applicable. However, you can produce a curve with any possible shape by altering two or more independent variables at the same time. In such cases, multiple regression methods are required (see ref. 3, pp. 50-82).

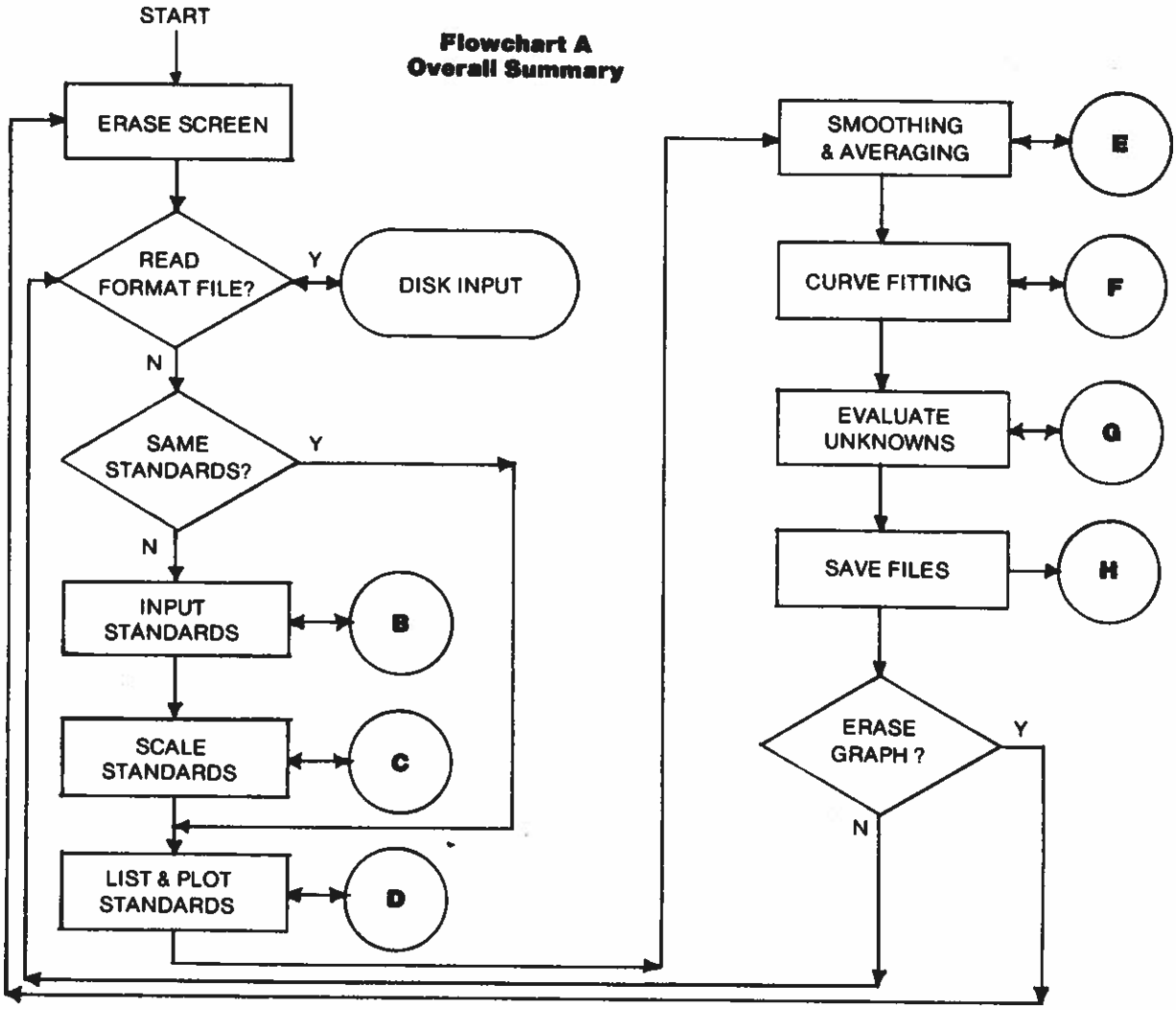
REFERENCES

1. R. D. Remington and M. A. Schork, "Statistics with Applications to the Biological and Health Sciences" (Prentice-Hall, Inc., Englewood Cliffs, NJ) 1970.

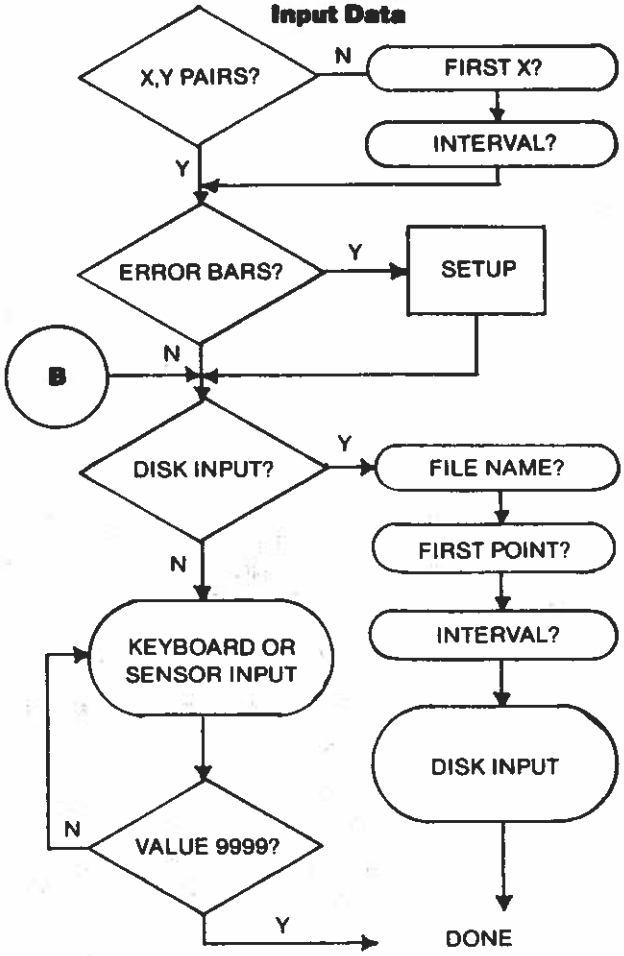
2. W. S. Dorn and D. D. McCracken, "Numerical Methods with FORTRAN IV Case Studies" (John Wiley and Sons, Inc., New York) 1972.

3. C. Daniel and F. S. Wood, "Fitting Equations to Data" (John Wiley and Sons, Inc., New York) 2nd Edition, 1980.

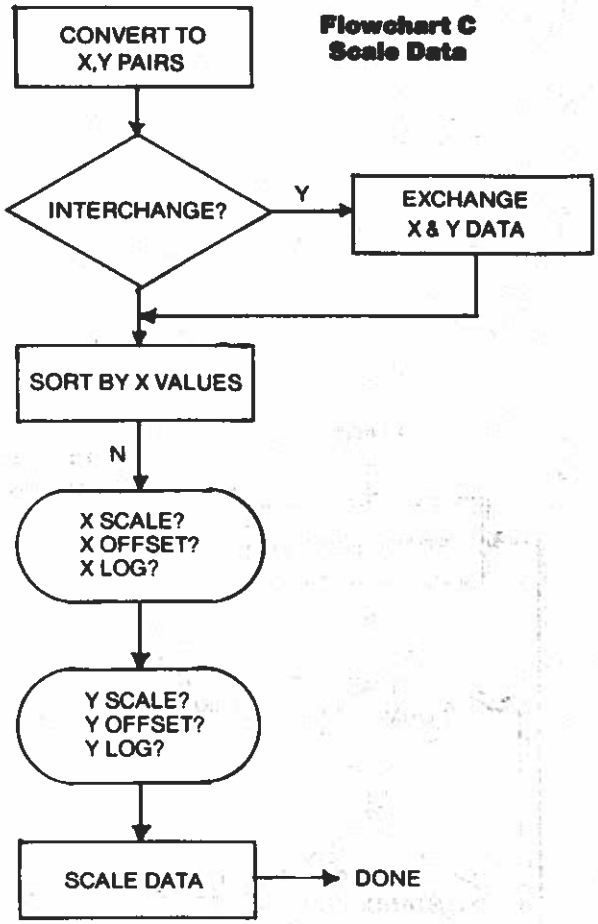
Flowchart A
Overall Summary



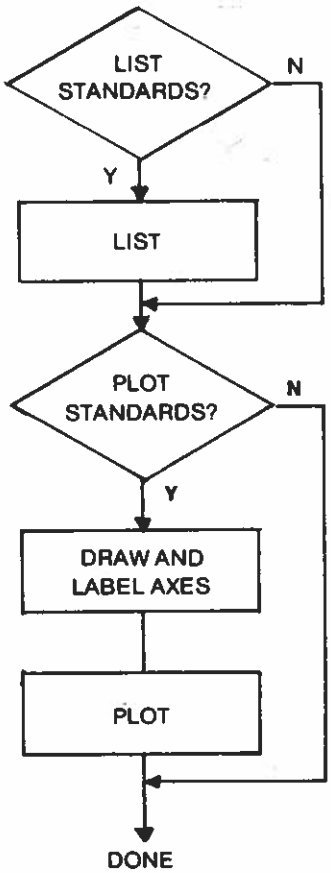
Flowchart B
Input Data



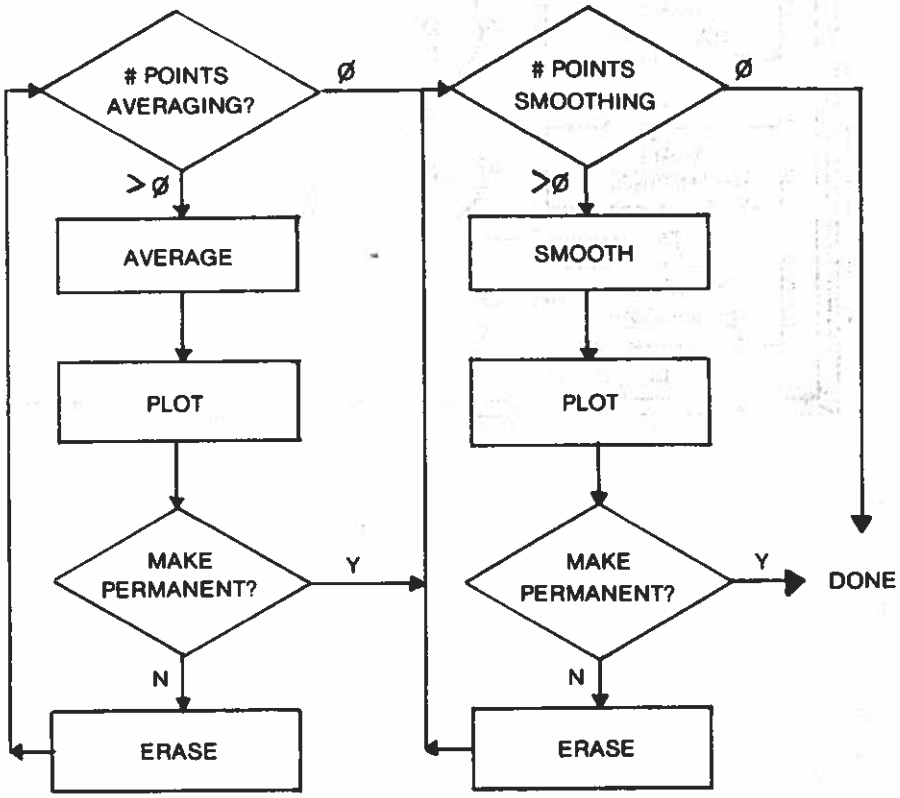
Flowchart C
Scale Data



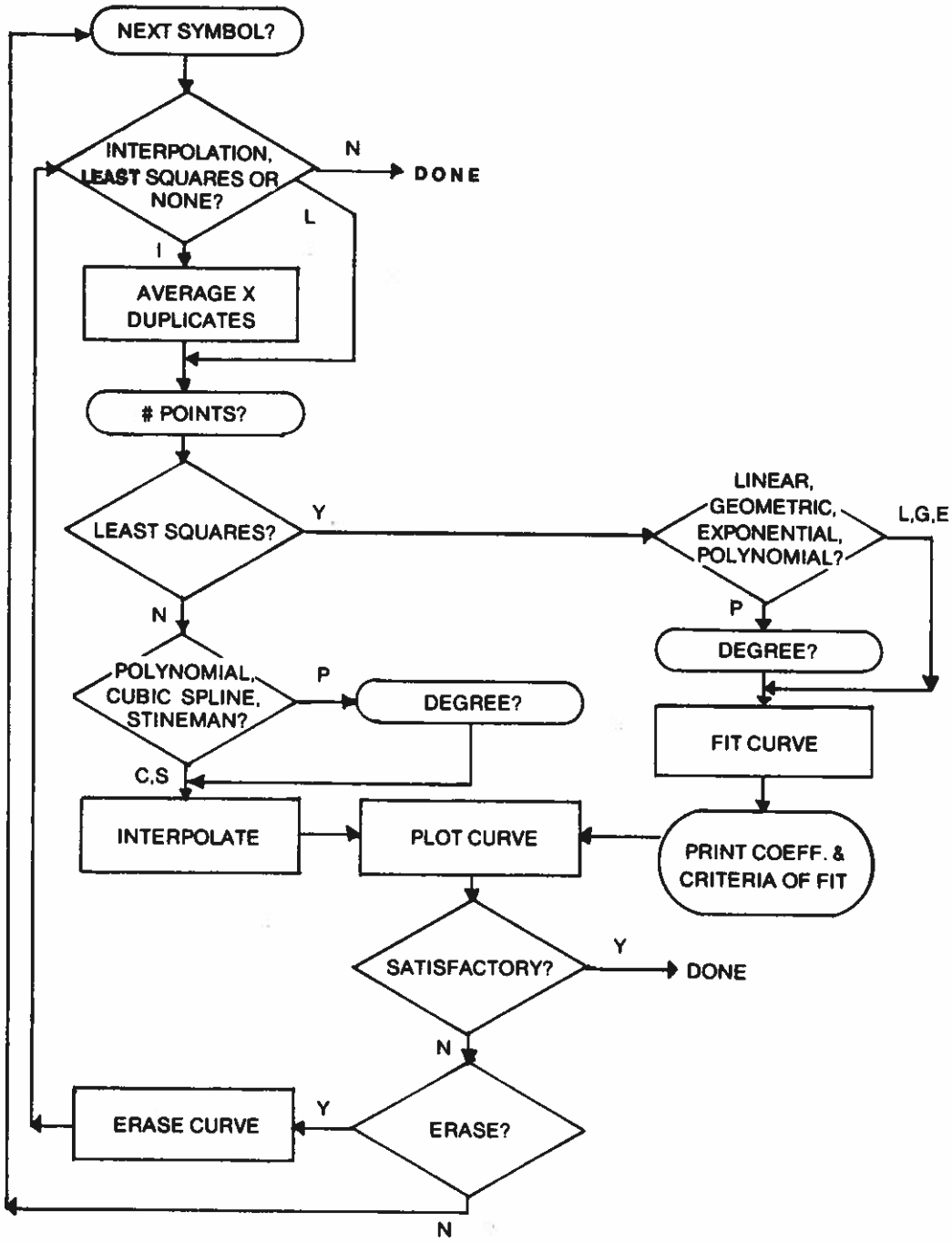
Flowchart D
List and Plot Standards



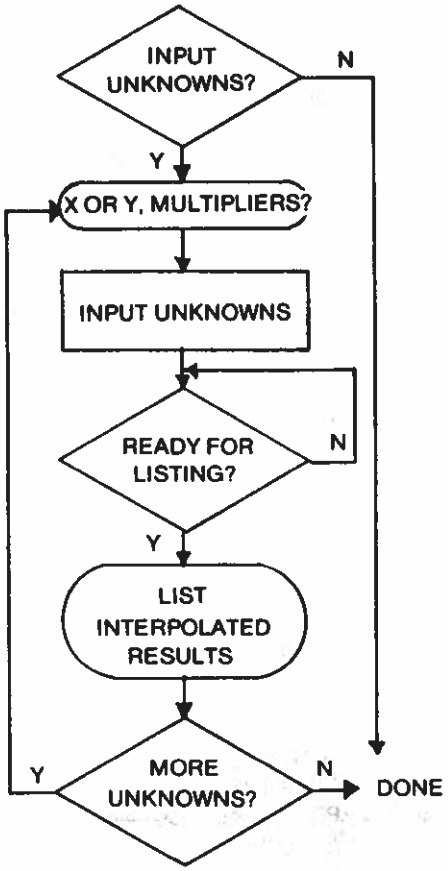
Flowchart E
Smoothing and Averaging



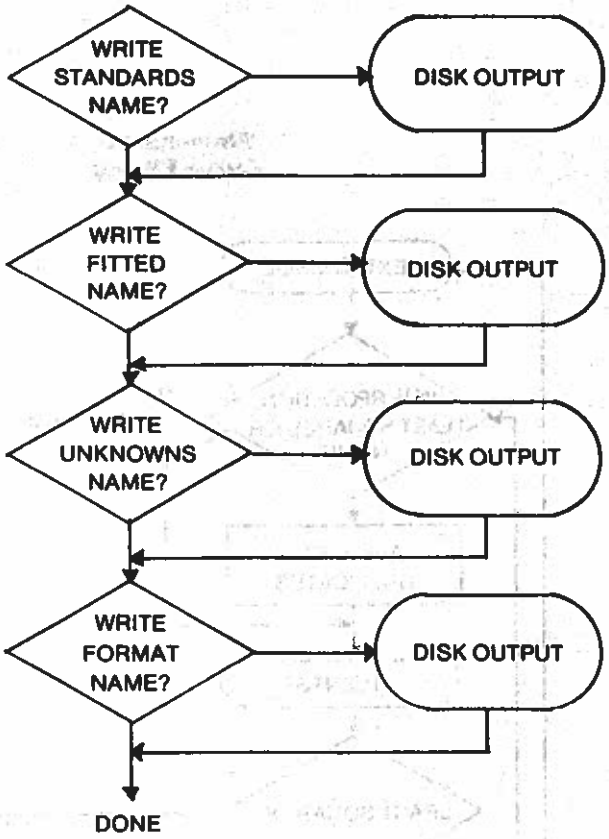
Flowchart F
Curve Fitting



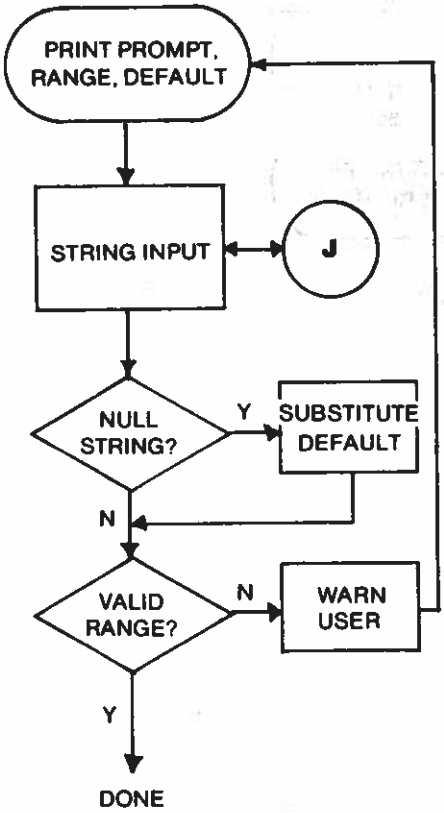
Flowchart G
Evaluate Unknowns



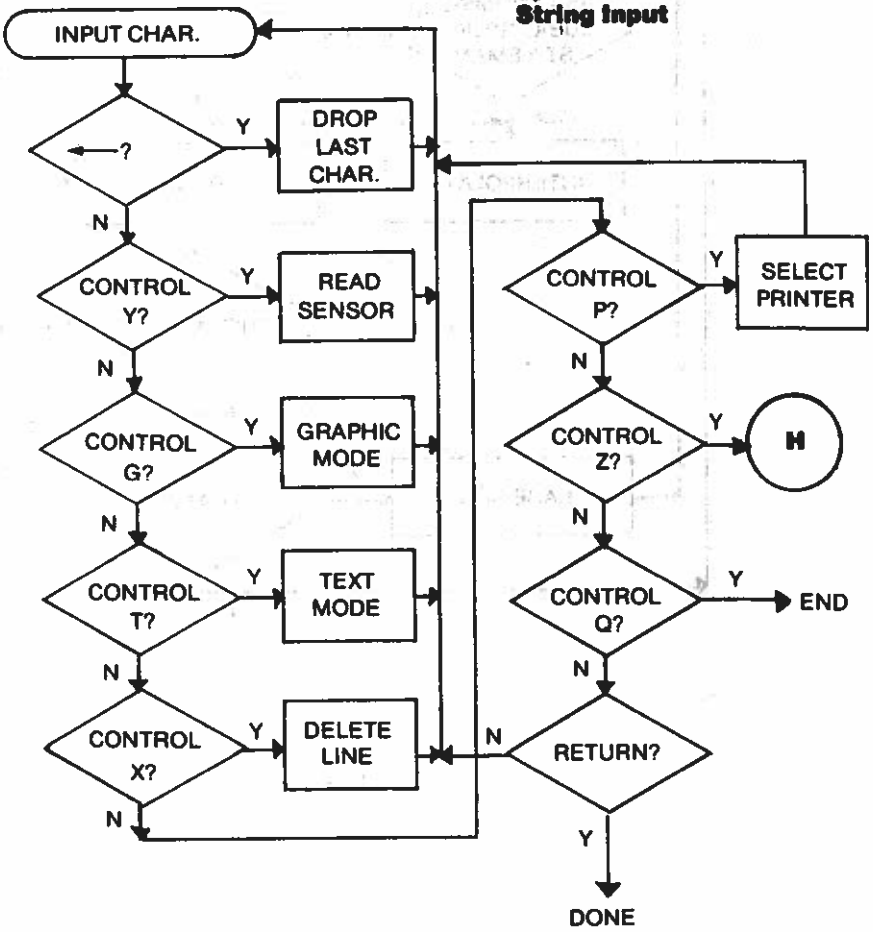
Flowchart H
Save Files



Flowchart I
User Input



Flowchart J
String Input



Interactive Microware, Inc.
P.O. Box 771
State College, Pa 16801
(814) 238-8294

SCIENTIFIC PLOTTER AND CURVE FITTER

CHANGES FOR USE WITH SILENTYPE PRINTER

If you own a Silentype printer, the changes described below will enable you to print your graphs at the touch of a key. Also, you will be able to print the series of questions and answers that produced the graph, tables of results and other information that you wish to save.

First, LOAD SCIPLOT or CURFIT and type the following changes exactly:

```
2852  IF C = 16 THEN  GET C$: PRINT : PRINT CD$ + "PR#" + C$: PRINT  CHR$
    (20): POKE  - 12529,0: GOTO 2810

2856  IF C = 17 THEN  POKE  - 12529,255: POKE  - 12528,7: POKE  - 12525,64
    : POKE  - 12524,0: PRINT  CHR$ (17): GOTO 2810
2858  IF C = 18 THEN  TEXT : END
```

Now, SAVE this modified version on your own disk (do not remove the write protect tab from the original disk). When you RUN the modified program, you may type CTRL P, followed by a single digit number that tells what slot your Silentype interface card is in. For example, CTRL P1 would select slot 1. From this point on, the questions and answers should appear both on the screen and on the printer. If the screen display (echo) is off, type CTRL P (#1-7) again; each time you type CTRL P (#1-7), the screen echo switch is toggled, just as though you had typed CTRL T. You will note that when the screen echo is off, some of the tables printed by CURVE FITTER do not TAB correctly; in this event, repeat the CTRL P command.

After the printer is activated (by CTRL P), you may print your graph by merely typing CTRL Q. To stop the program, use CTRL R instead of CTRL Q.

Interactive Microware, Inc.
P.O. Box 771
State College, Pa 16801
(814) 238-8294

CURVE FITTER UPDATE #1

1. You may increase the number of data values stored in CURFIT by changing the DIMension of array D(1000) in line 10 to D(2000). At the end of line 10, you must also change the value of MX to the same maximum dimension (MX=2000).

2. You may increase the maximum number of standard points for the cubic spline and Stineman interpolation methods by increasing the DIMensions of arrays A(99), B(99), and C(99) in line 12. In this case, you must also increase the value of ABC at the end of line 12 by a corresponding amount. If you get an OUT OF MEMORY error, you will have to decrease the size of these arrays or else use a smaller dimension for the D(2000) array.

Interactive Microware, Inc.
P.O. Box 771
State College, Pa 16801
(814) 238-8294

CURVE FITTER UPDATE #2

1. The original version of Curve Fitter does not allow extrapolation of unknowns outside the range of the standards data. However, it is sometimes valid to extrapolate a least squares curve, with some caution. Theoretically, there is no guarantee that extrapolating the fitted curve will give a reasonable prediction of results. At your own risk, you may make these changes to permit extrapolation:

```
470  INPUT "EXTRAPOLATION FACTOR (0:1) ?";EX

480 X2 = (X1 - X0) * EX:X0 = X0 - X2:X1 = X1 + X2:X2 = X1 - X0

585 Y2 = (Y1 - Y0) * EX:Y0 = Y0 - Y2:Y1 = Y1 + Y2:Y2 =
     Y1 - Y0:YY = (YT - YB) / Y2

860 XY = 2:DD = 2:DD(1,1) = DD(0,1) + IV:DD(2,1) = 1:
     DD(3,1) = 0:DD(4,1) = X0:DD(5,1) = (X1 - X0) /
     IV*.9999
```

The extrapolation factor widens the margins around the standards in the plotted curve. For example, an extrapolation factor of 0.5 adds 50% of the range of the standards on the left, right, top and bottom margins. You must always enter an extrapolation factor of 0 when using an interpolation method instead of a least squares fitting method.

2. If you perform logarithmic conversion on your X or Y data for your standards, the original version of Curve Fitter reports the interpolated results for the unknowns in logarithmic form. The changes below will convert the results for the unknowns back to the antilog form.

```
1890 Z2 = D * (P0 - U + (Z1 - D(P0)) / (D(P2) - D(P0)))
     + W: IF XG THEN Z2 =  EXP (Z2 / XG)
1895  GOTO 1945

1920 Z2 = (Z0 -  INT (Z0)) * (D(P0 + 1 + EB) - D(P0)) +
     D(P0): IF YG THEN Z2 = EXP (Z2 / YG)
1925  GOTO 1945
```

Interactive Microware, Inc.
P.O. Box 771
State College, Pa 16801
(814) 238-8294

CHANGES IN SCIENTIFIC PLOTTER, CURVE FITTER AND VIDICHART

FOR USE WITH EPSON MX-7Ø AND EPSON MX-8Ø PRINTERS AND FOR THE PKASO* PRINTER INTERFACE SERIES

The changes below assume that you are using either the EP-12 parallel interface card (made by Interactive Structures, Inc.) or the GRAPPLER(tm) interface card (made by Orange Micro, Inc.). Both of these cards feature ROM routines that enable you to print the high resolution screen by typing just a few control characters. If you do not already own one of these printer interface cards, we highly recommend the new EP-12 card because it has special commands to print low resolution graphics, half-tone graphics, super high resolution graphics (96Ø x 792 points) and special characters and symbols defined by you. The EP-12 also works with Pascal and CP/M systems, and it includes a DEMO DISK showing examples of all features. The price of this powerful interface card is only $175, and it is now available from Interactive Microware.

For both Scientific Plotter and Curve Fitter, make the following changes for use with the EP-12 interface:

```
2852 IF C=16 THEN GET C$: PRINT: PRINT CD$+"PR#"+C$:
     PRINT CHR$(9)"8ØP": GOTO 281Ø

2858 IF C=17 THEN PRINT CHR$(9)"15H": GOTO 281Ø

2859 IF C=18 THEN TEXT: END
```

For the GRAPPLER interface, omit the PRINT CHR$(9)"8ØP": in line 2852 and change "15H" in line 2858 to "G2". After making these changes, you can turn on the printer at any time by typing CTRL P, followed by a single digit number denoting the slot that your interface card is plugged into. If the printer is on, you can print the hi-res screen by merely typing CTRL Q. To stop the program, type CTRL P instead of CTRL Q (as in the manual).

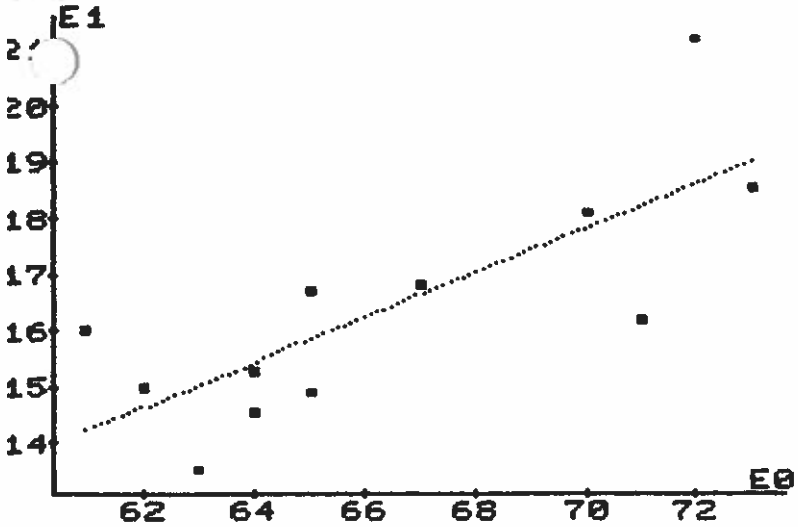
* For the PKASO series of interface cards (by Interactive Structures, Inc.), change "80P" to "I" and change "15H" to "33H".

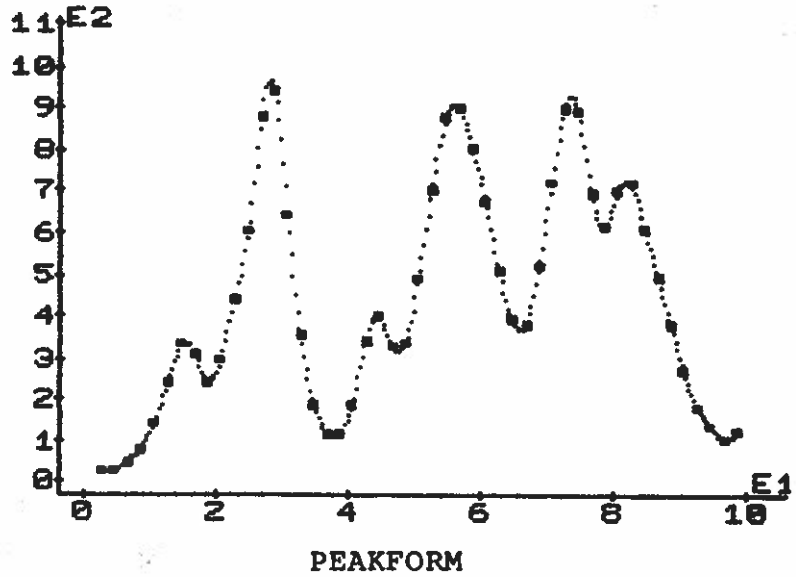
The modifications for VIDICHART closely follow the changes described in the manual for use with the Silentype printer. Thus, make the same changes in lines 32, 974, 975, 978 and 979, and alter line 976 and 977 as follows:

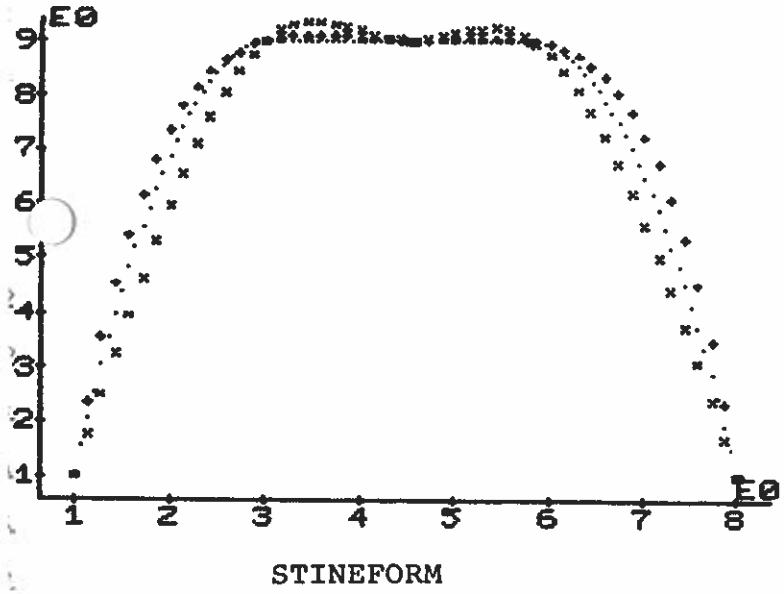
```
976 IF C=17 THEN PRINT CHR$(9)"15H" (or "G2" for the GRAPPLER).

977 IF C=2Ø THEN PRINT CHR$(9)"8ØP"
```

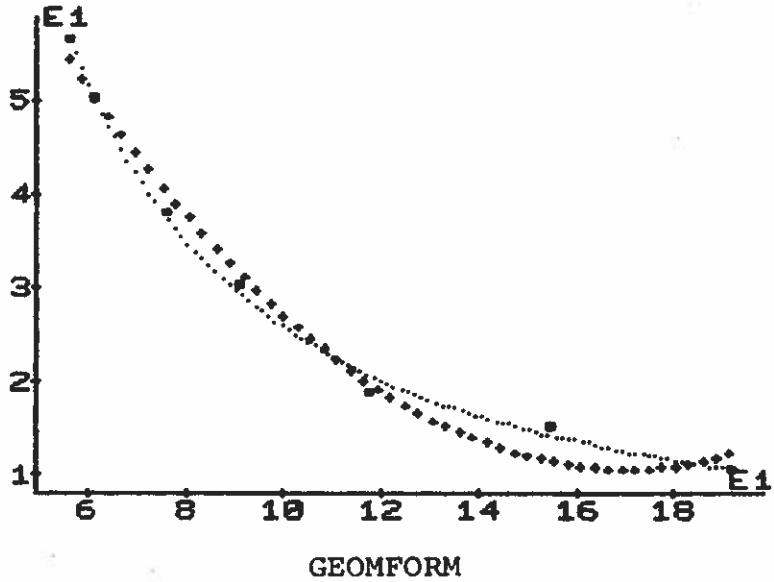
Bear in mind that you must turn off the "cueing" mode (type CTRL U) before activating the printer by typing CTRL P, followed by the slot number. When the printer is on, type CTRL Q to print the screen image and type CTRL R to print the report at the bottom of the screen. To stop the program, type CTRL S. In text mode, typing CTRL T when the printer is on will turn on the screen echo so that characters are printed on both the screen and the printer.

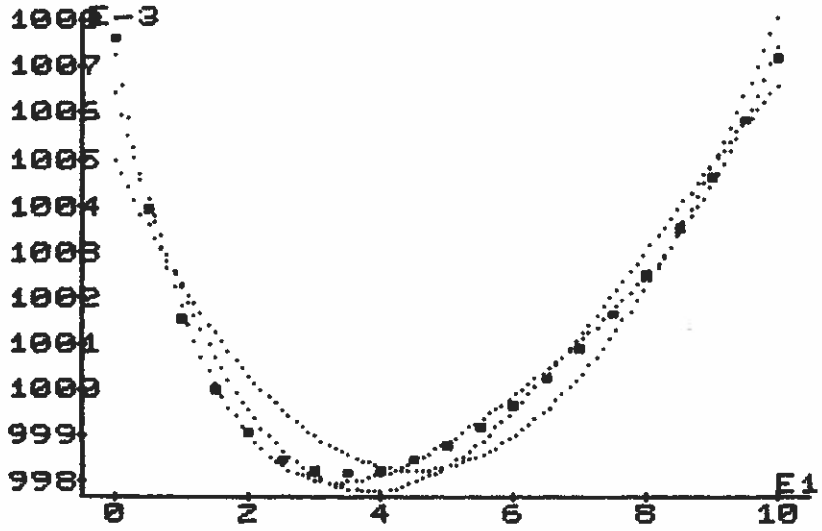
CURVE FITTER EXAMPLES

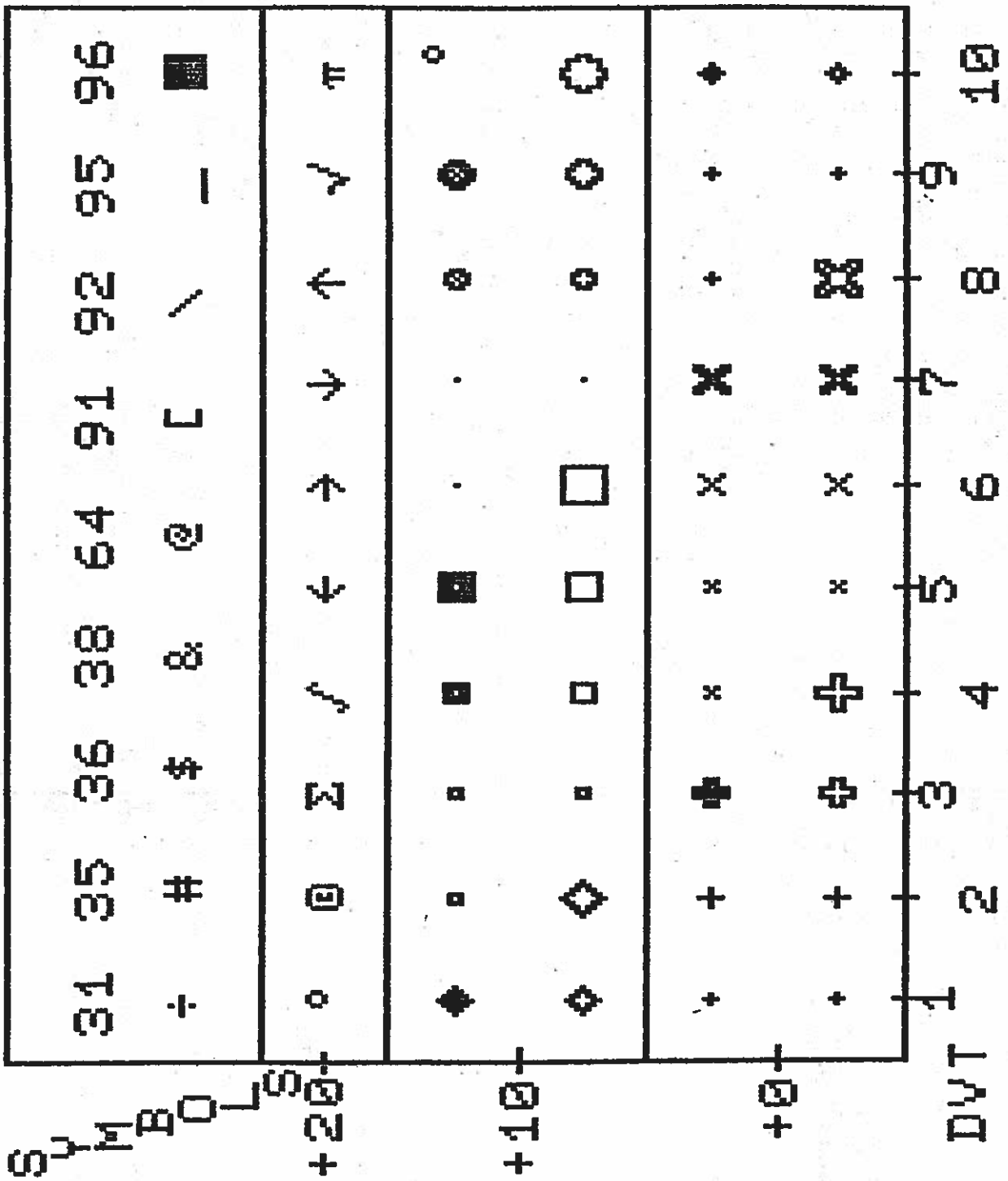




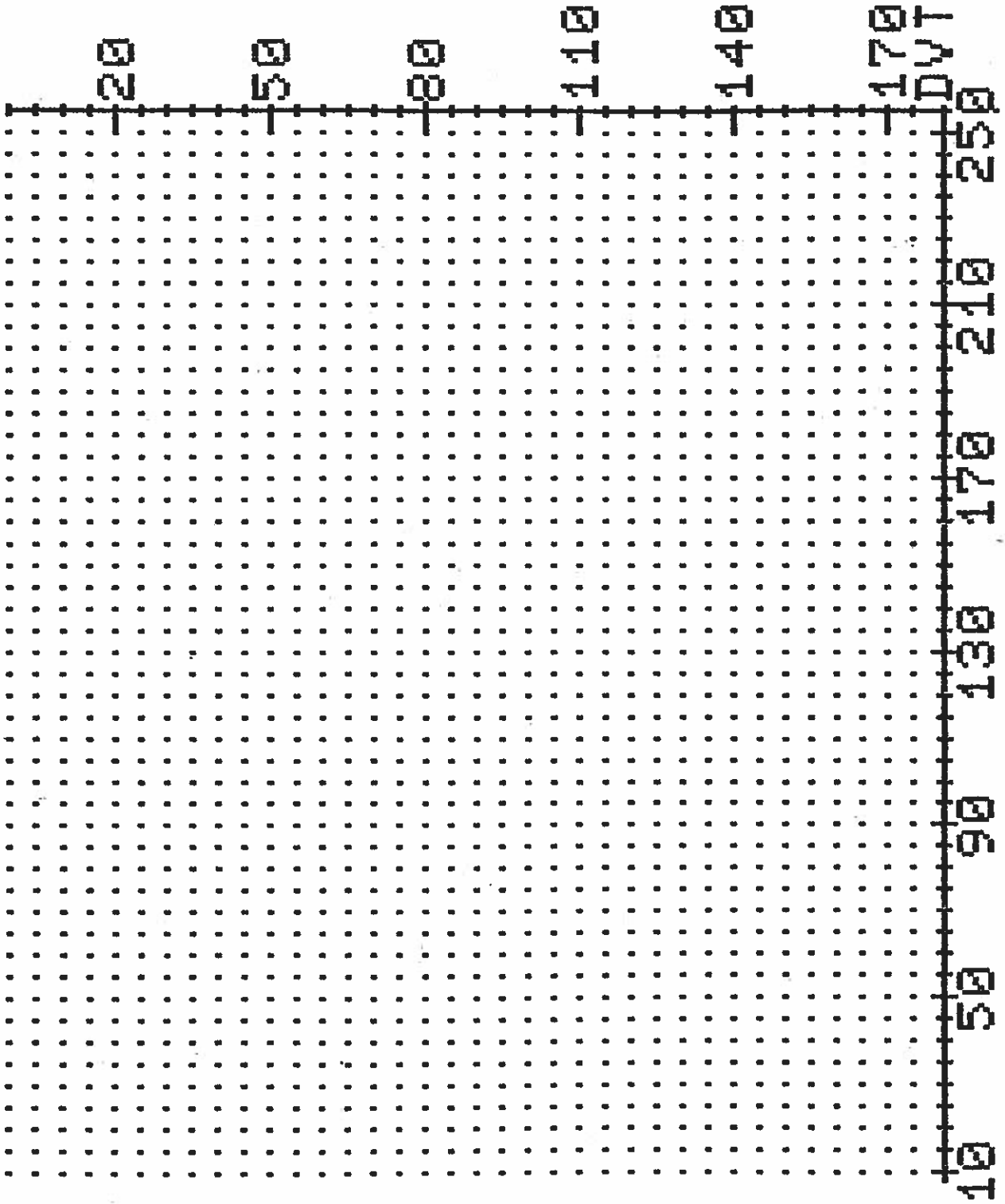








Used with the permission of Donald V. Tappan, 187 Rockwell Street, Norwich, CT 06360.

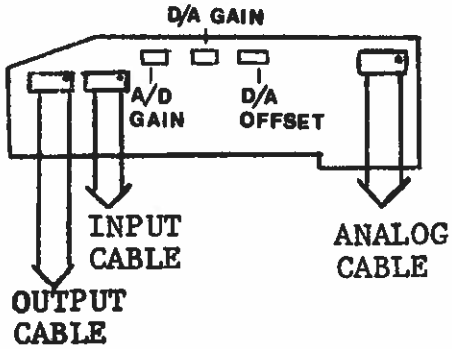


Used with the permission of Donald V. Tappan, 187 Rockwell Street, Norwich, CT 06360.

Interactive Microware, Inc.
P.O. Box 771
State College, Pa 16801
(814) 238-8294

Attention: ADALAB Users

ADALAB is shipped with jumpers set for installation in SLOT 2 and a range of ±4 volts. The cables of the self-test adapter should be plugged into the sockets of the ADALAB card as shown, with pin 1 at the upper right.



The self-test adapter connects the parallel outputs to the parallel inputs. It also connects D/A to A/D+ and A/D- to GND. You may connect your instruments to the D/A or A/D converter after removing the two plastic jumpers. The jumpers may also be turned at right angles so that a 22 or 24 gauge wire can be inserted into the vacant hole from below. The jumpers should be returned to normal before running the self-test program.

Calibration Procedure Changes: A zero offset potentiometer adjustment has been added for the D/A converter, as shown in the figure above. After calibrating the D/A and A/D converters as described in the ADALAB Hardware Manual, set the D/A output voltage to zero and adjust the D/A offset pot to give an A/D reading of zero volts. This may necessitate a slight readjustment of the D/A and A/D gain pots.

If you have purchased both the ADALAB hardware and the VIDICHART software at the same time, the QUICKI/O disk contains an updated version called VIDICHARTAD. This program is described in the note in the back of the ADALAB manual entitled "Using the ADALAB A/D Converter with Curve Fitter VIDICHART and other

BASIC Programs." VIDICHARTAD requires an additional file called VISIOBJ, which is on the original disk for VIDICHART. Thus, to run VIDICHARTAD, you should first LOAD it from the QUICKI/O disk and then SAVE it on a copy of the VIDICHART disk before RUNning the program. The VIDICHARTAD update also requires the ADOBJ file from the QUICKI/O disk. Therefore, you should also read ADOBJ from the QUICKI/O disk (type BLOAD ADOBJ,A$320) and write it on the copy disk (type BSAVE ADOBJ,A4320,L$50).

Interactive Microware, Inc.
P.O. Box 771
State College, Pa 16801
(814) 238-8294

ADALAB WARRANTY INFORMATION

Interactive Microware, Inc. (IMI) warrants each new ADALAB(tm) Interface Board against defects in material or workmanship for a period of 90 days from the date of purchase and agrees to repair or replace any defective unit without charge.

IMPORTANT: This warranty does not cover damage resulting from accident, misuse, or abuse, lack of reasonable care, modifications other than those done in accordance with instructions published by Interactive Microware, loss of parts or subjecting the product to service or repairs not performed by IMI or registered repair centers. This warranty applies only to the ADALAB Interface Board; IMI will assume no responsibility for damage to the Apple Computer or for damage to other peripherals or instruments to which ADALAB is attached.

Inasmuch as IMI does not have sufficient information to evaluate the suitability of ADALAB for any particular application, the purchaser is expected to understand the operational characteristics of this interface, as described in the documentation supplied by IMI, and to assess the suitability of ADALAB for his application.

This warranty is in lieu of all other warranties, expressed or implied, including warranties of merchantability and fitness for use. In no event will IMI be liable for incidental and consequential damages arising from or in any way connected with the use of its products.

The purchaser of this product is hereby notified that the probability of failure of this device is not zero, and that this product is NOT RECOMMENDED for any use in dangerous or hazardous situations in which its failure has a chance of causing injury, damage, death, or a human decision resulting in injury, damage or death.

This warranty extends only to the original purchaser. Proof of purchase is necessary for products returned for warranty service. To obtain service, return the product (postage paid) to:

IMI Repair Service Department
P.O. Box 771
State College, PA 16801

A detailed description of the problem must be included. After the warranty period expires, IMI will repair or replace the card for a charge that will vary from $25 to $150, depending on the parts and labor required. In such cased, the product will be returned by UPS C.O.D. unless a company purchase order or payment in full has been received.