Data Transfer from the HP3000 to the HP2647A Intelligent Graphics Terminal for Plotting Under Terminal Basic AGL

V. Zitko

Biological Station
St. Andrews, N. B. EOG 2XO

June 1984

Canadian Manuscript Report of Fisheries and Aquatic Sciences No. 1763

Government of Canada
Fisheries and Oceans

Gouvernement du Canada
Pêches et Océans

Canadian Manuscript Report of Fisheries and Aquatic Sciences

These reports contain scientific and technical information that represents an important contribution to existing knowledge but which for some reason may not be appropriate for primary scientific (i.e. *Journal*) publication. They differ from Technical Reports in terms of subject scope and potential audience: Manuscript Reports deal primarily with national or regional problems and distribution is generally restricted to institutions or individuals located in particular regions of Canada. No restriction is placed on subject matter and the series reflects the broad interests and policies of the Department of Fisheries and Oceans, namely, fisheries management, technology and development, ocean sciences, and aquatic environments relevant to Canada.

Manuscript Reports may be cited as full publications. The correct citation appears above the abstract of each report. Each report will be abstracted by *Aquatic Sciences and Fisheries Abstracts* and will be indexed annually in the Department's index to scientific and technical publications.

Numbers 1-900 in this series were issued as Manuscript Reports (Biological Series) of the Biological Board of Canada, and subsequent to 1937 when the name of the Board was changed by Act of Parliament, as Manuscript Reports (Biological Series) of the Fisheries Research Board of Canada. Numbers 901-1425 were issued as Manuscript Reports of the Fisheries Research Board of Canada. Numbers 1426-1550 were issued as Department of Fisheries and the Environment, Fisheries and Marine Service Manuscript Reports. The current series name was changed with report number 1551.

Details on the availability of Manuscript Reports in hard copy may be obtained from the issuing establishment indicated on the front cover.

Rapport manuscrit canadien des sciences halieutiques et aquatiques

Ces rapports contiennent des renseignements scientifiques et techniques qui constituent une contribution importante aux connaissances actuelles mais qui, pour une raison ou pour une autre, ne semblent pas appropriés pour la publication dans un journal scientifique. Ils se distinguent des Rapports techniques par la portée du sujet et le lecteur visé; en effet, ils s'attachent principalement à des problèmes d'ordre national ou régional et la distribution en est généralement limitée aux organismes et aux personnes de régions particulières du Canada. Il n'y a aucune restriction quant au sujet; de fait, la série reflète la vaste gamme des intérêts et des politiques du Ministère des Pêches et des Océans, notamment gestion des pêches; techniques et développement, sciences océaniques et environnements aquatiques, au Canada.

Les Manuscrits peuvent être considérés comme des publications complètes. Le titre exact paraît au haut du résumé de chaque rapport, qui sera publié dans la revue *Aquatic Sciences and Fisheries Abstracts* et qui figuera dans l'index annuel des publications scientifiques et techniques du Ministère.

Les numéros de 1 à 900 de cette série ont été publiés à titre de manuscrits (Série biologique) de l'Office de biologie du Canada, et après le changement de la désignation de cet organisme par décret du Parlement, en 1937, ont été classés en tant que manuscrits (Série biologique) de l'Office des recherches sur les pêcheries du Canada. Les numéros allant de 901 à 1425 ont été publiés à titre de manuscrits de l'Office des recherches sur les pêcheries du Canada. Les numéros 1426 à 1550 ont été publiés à titre de Rapport manuscrits du Service des pêches et de la mer, Ministère des Pêches et de l'Environnement. Le nom de la série a été changé à partir du rapport numéro 1551.

La page couverture porte le nom de l'établissement auteur où l'on peut se procurer les rapports sous couverture cartonnée.

Canadian Manuscript Report of
Fisheries and Aquatic Sciences 1763

June 1984

DATA TRANSFER FROM THE HP3000 TO THE HP2647A INTELLIGENT GRAPHICS TERMINAL FOR PLOTTING UNDER TERMINAL BASIC AGL

by

V. Zitko

Fisheries and Environmental Sciences
Fisheries Research Branch
Department of Fisheries and Oceans
Biological Station
St. Andrews, N. B. EOG 2XO
Canada

Cat. No. FS 97-4/1763E ISSN 0706-6473

Correct citation for this publication:

Zitko, V. 1984. Data transfer from the HP3000 to the HP2647A Intelligent Graphics Terminal for plotting under Terminal BASIC AGL. Can. MS Rep. Fish. Aquat. Sci. 1763: iii + 9 p.

ABSTRACT

Zitko, V. 1984. Data transfer from the HP3000 to the HP2647A Intelligent Graphics Terminal for plotting under Terminal BASIC AGL. Can. MS. Rep. Fish. Aquat. Sci. 1763: iii + 9 p.

A program running on the HP-3000 displays data on the HP2647A. Terminal BASIC reads the data from the screen and manipulates them as desired. The procedure is convenient for graphic displays of relatively small sets of data that are usually encountered in laboratory work.

RÉSUMÉ

Zitko, V. 1984. Data transfer from the HP3000 to the HP2647A Intelligent Graphics Terminal for plotting under Terminal BASIC AGL. Can. MS. Rep. Fish. Aquat. Sci. 1763: iii + 9 p.

Un programme exécuté sur HP-3000 affiche les données sur HP2647A. Le terminal BASIC lit les données qui apparaissent sur l'écran et les manipule tel que désiré. Le processus convient bien à l'affichage graphique de petites séries de données recueillies dans le cours de travaux de laboratoire ordinaires.

INTRODUCTION

This report describes a simple technique for the transfer of data from the HP3000 to the Intelligent Graphics Terminal HP2647A. The data can then be plotted by the HP Terminal BASIC (TBASIC). TBASIC contains "A Graphics Language" (AGL) extension which makes graphic operations very easy.

Graphic displays are an essential tool for data evaluation. They help to put data in proper perspective and aid in their comprehension, paraphrasing a well-known proverb, "a picture is better than a kilogram of printout." Yet, the various software packages used on the HP3000 have very limited graphic capabilities. In addition, the requirements in respect to graphic displays are extremely diverse. Even very complex (and expensive) packages are not likely to handle all situations.

The graphic capabilities of the HP2647A can be controlled from the HP3000 by using "escape sequences" (see Appendices in Reference [1] or [2]). This is quite awkward for the average user. In comparison, programming with the TBASIC AGL is simple.

The cooperation between the HP3000 and the TBASIC interpreter is not described very explicitly in the manuals and this lack of information hinders the full utilization of the graphic capabilities of the HP2647A.

Since versatility and customized applications are the important point, this report describes the general technique. It does not attempt to develop a program. It tries to outline the approach. One example of a program handling an actual plotting task is given.

DISPLAY PREPARATION

The numbers to be plotted are displayed by the HP3000 on the HP2647A. Up to 263 lines of 80 characters each may be displayed. This should be sufficient for most applications. TBASIC then takes over, reads the display and plots the data in the desired format.

A number of approaches may be used. The output of a program may be used directly or it may require some editing. This may be done manually on the HP2647A or the program output may be directed to an ASCII file which is then edited in QEDIT. To do this for a BASIC program, build a file:

```
:BUILD filename;REC=-80,,F,ASCII
```

Then run the program, with the output going to the file:

```
>RUN, OUT=filename
```

Get in QEDIT, load the file (/T filename), edit it, list quietly (/LQ), and activate the reading and plotting TBASIC program (see later).

Alternatively, the edited file may be read by another program and only subsets of the data may be used. This will be illustrated in an example.

It is a good practice to mark the beginning of data distinctively, for example by "---".

READING THE DISPLAY

The TBASIC print functions "move cursor absolute", MOVCA(R,C) and "move cursor relative" MOVCR(R,C) are needed to position the cursor. In the simplest case the cursor is moved to the beginning of the display (the cursor is "homed"; PRINT MOVCA(1,1)).

The data are then read by the TBASIC input/output function DSPIN$(L,X), starting at the current cursor position. The absolute value of L specifies the number of characters to read. Negative values of L are used to ignore "nonprintable" ASCII characters 128-255. These should not be normally present in the display but the program should ignore them, just in case.

If the beginning of the data is marked, the search for the mark must be incorporated in the program. For example, the program below finds the mark ("---"), then reads the number of data pairs and, finally, the data pairs themselves, placing them into arrays X() and Y(). Note that the data are read as string and then converted into numeric variables (statements 70 and 120).

```
10 DIM X(100),Y(100),C$(3),I$(3),X$(15),Y$(15)
20 PRINT MOVCA(1,1) !Home cursor
30 C$=DSPIN$(-3,X1)          <----------------------------------
40 PRINT                        Look for data mark            |
50 IF C$<>"---" THEN 30      <----------------------------------
60 I$=DSPIN$(-3,X1)             Number of data pairs
70 I1=VAL(I$)
80 FOR I=1 TO I1              <----------------------------------
90 PRINT                                                      |
100 X$=DSPIN$(-15,X1)                Read data                |
110 Y$=DSPIN$(-15,X1)                                         |
120 X(I)=VAL(X$)\Y(I)=VAL(Y$)                                 |
130 NEXT I                    <----------------------------------
140 PRINT "Reading completed "
150 PRINT
160 FOR I=1 TO I1\ PRINT X(I),Y(I)\ NEXT I
170 END
```

The program will read a section of the display marked by "---":

```
3
1              2
2              4
3              6
```

It is important to be certain about the positions of the data on the screen. To check whether the cursor is positioned as expected,its movements can be slowed down by incorporating "dummy" loops (such as FOR I=1 TO 1000 STEP .2\NEXT I) temporarily in the program. For example as statement 105 in the above program, this loop will stop the cursor for a while before reading Y$.

INTERACTION BETWEEN THE HP3000 AND THE HP2647A

To communicate with the HP3000, the HP2647A must be in the "Remote" mode. On the other hand, for reading the screen and plotting data, the HP2647A must be in the "Local" mode. Otherwise, sooner or later something would be sent to the HP3000, generating an error message and disturbing the display.

There is no BASIC statement to control the HP2647A mode and the "escape sequence" must be used. The code for "Remote ON" is ESC&k1R, that for "Remote OFF" ESC&k0R. ESC (escape) is ASCII 27. In BASIC these two escape sequences are CHR$(27)&"&k1R" and CHR$(27)&"&k0R", respectively, and may be executed by PRINT statements.

In addition, the TBASIC interpreter must be suspended during the preparation of the screen by the HP3000. This is achieved by the SLEEP statement.

When needed, the interpreter is brought into action by the "ON KEY #.. GOTO .." statement. Theoretically, almost any key could be used. In practice, some keys may wake up the interpreter unintentionally. The "]" key (#93) is a safe choice.

The program DEM01 includes these features.

```
DEM01
10 DIM X(100),Y(100),C$(3),I$(3),X$(15),Y$(15)  <--------------------
20 Rmote$=CHR$(27)&"&k1R"\Locl$=CHR$(27)&"&k0R"                      |
40 PLOTR                                                             |
50 ON KEY #93 GOTO 70                                <---------------
60 SLEEP
70 PRINT Locl$;
75 PRINT CHR$(141) !Clear screen
80 LOCATE (100,190,8,98)                             <---------------
90 SCALE (0,6.28,-1,1)                                Locate the graph, |
100 FRAME                                             draw the axes     |
110 LAXES (-.2,.1,0,0,5,10,2)                        <---------------
120 PRINT MOVCA(1,1) !Home cursor
130 C$=DSPIN$(-3,X1)                                 <---------------
140 PRINT                                                            |
150 IF C$<>"---" THEN 130                                            |
160 I$=DSPIN$(-3,X1)                                                 |
170 I1=VAL(I$)                                        Read data from |
180 FOR I=1 TO I1                                     the screen     |
190 PRINT                                                            |
200 X$=DSPIN$(-14,X1)                                                |
210 Y$=DSPIN$(-14,X1)                                                |
220 X(I)=VAL(X$)\Y(I)=VAL(Y$)                                        |
230 NEXT I                                           <---------------
240 FOR I=1 TO I1                                    <---------------
250 X=X(I)\Y=Y(I)                                                    |
260 MOVE (X-.1,Y)\DRAW (X+.1,Y)                       Plot           |
270 MOVE (X,Y+.05)\DRAW (X,Y-.05)                                    |
280 NEXT I                                           <---------------
285 PRINT Rmote$
290 SLEEP
```

The data on the screen may be generated for example by a simple BASIC program:

```
TEMP2
    1 PRINT "---"
    2 PRINT 33
   10 FOR I=0 TO 6.2 STEP .2
   20   PRINT I,SIN(3*I)
   30 NEXT I
```

This is the result on the screen:

```
 ---
 33
 0              0
 .2             .564643
 .4             .932039
 .6             .973848
 .8             .675463
 1              .14112
 1.2           -.442521
 1.4           -.871576
 1.6           -.996165
 1.8           -.772764
 2             -.279415
 2.2            .311541
 2.4            .793667
 2.6            .998543
 2.8            .8546
 3.             .41212
 3.2           -.174324
 3.4           -.699873
 3.6           -.980936
 3.8           -.91933
 4.            -.536577
 4.2           3.36183E-02
 4.4            .592069
 4.6            .943694
 4.8            .96566
 5.             .650293
 5.2            .107762
 5.4           -.472413
 5.6           -.887564
 5.8           -.99266
 6.            -.750994
 6.2           -.246985
```

To try it, sign on the HP3000, get in BASIC and type in the program TEMP2. Then go manually off "Remote", (in "Local"), load the TBASIC interpreter and run the program DEMO1. Go in "Remote" (off "Local") and prepare the data on the screen. In this case, run TEMP2. Finally, activate DEMO1 by the "]" key.

The program DEMO1 may be also loaded from its QEDIT file by the /LQ command. The TBASIC interpreter must be active during this time. An error message may appear on the screen, but the program should load properly.

The result of PLOT2 and DEMO1 is reproduced in Fig. 1.

SCALING THE AXES

In the example above, a trigonometric function was selected on purpose, to make the scaling of the axes simple (Y-axis range -1,1; X-axis range 0,6.2, which is roughly 0,360 degrees). In other cases the scaling may be more difficult. It is a good practice to let the HP3000 find the range of the values (it is faster than TBASIC). The rest may be done by TBASIC.

A simple scaling routine has been described by Schifreen [3] and later expanded by Reischel [4] to include negative values. The step size in both routines is limited to an integer power of 10. The program given below selects a step size as 1, 2, or 5 times an integer power of 10, to cover the desired range in 8-10 steps. It also shifts the scale to provide convenient step values.

The program requires the input of Y1 and Y0, the maximum and the minimum value, respectively. It calculates the base 10 logarithm of their difference. When the logarithm is negative (the difference is <1), the logarithm is transformed to have negative characteristic (for example log(0.2)=-0.6990=0.3010-1). The step size is selected next (log(2)=0.30103, log(5)=0.69897). This is followed by shifting the scale so that it starts on an integer step multiple. Finally, the scale is printed. The printout format is computer-dependent and should be modified as required. Similarly, some computers have the base 10 LOG function and do not need the conversion coefficient (0.434295).

```
 10 F1=0
 20 INPUT "MAXIMUM VALUE ",Y1
 30 INPUT "MINIMUM VALUE ",Y0
 40 Y3=.434295*LOG(ABS(Y1-Y0))
 50 IF Y3>0 THEN 160
 60 REM ----------------------------------------
 70 REM DETERMINE NEGATIVE CHARACTERISTIC
 80 FOR I=1 TO 30
 90   Y3=Y3+1
100   IF Y3>0 THEN 120
110 NEXT I
120 Y4=-I
130 Y5=Y3
140 GOTO 200
150 REM ----------------------------------------
160 Y4=INT(Y3)
170 Y5=Y3-Y4
180 REM ----------------------------------------
190 REM SELECT STEP SIZE
200 IF Y5<=.30103 THEN Y5=.30103
210 IF Y5>.30103 AND Y5<=.69897 THEN Y5=.69897
```

```
220 IF Y5>.69897 THEN Y5=1
230 Y2=10**(Y4+Y5-1)
240 PRINT "STEP ";Y2
250 REM ---------------------------------------
260 REM ADJUST SCALE TO CONVENIENT UNITS
270 X1=Y0+Y2
280 X2=INT(X1/Y2)
290 X3=(X2-1)*Y2
300 PRINT "NEW START ";X3
310 Y0=X3
320 REM ---------------------------------------
330 REM PRINT THE RESULT
340 FOR I=1 TO 10
350   Y9=Y0+(I-1)*Y2
360   IF Y9>Y1 THEN F1=1
370   PRINT USING "#,MD.2DE,2X";Y9
380   IF I MOD 5=0 THEN PRINT
390   IF F1=1 THEN 410
400 NEXT I
410 PRINT
420 END
```

Example

```
MAXIMUM VALUE 1234
MINIMUM VALUE 32
STEP  200.
NEW START  0
 0.00E+00   2.00E+02   4.00E+02   6.00E+02   8.00E+02
 1.00E+03   1.20E+03   1.40E+03
```

"REAL-LIFE" EXAMPLE

Measurements of a number of parameters in two tissues of control and of exposed animals were presented as an ASCII file. Each parameter was measured at 16 times, except for two parameters, which have been measured only at 14 times. The task was to plot the means of the parameters against time. One graph should be prepared for each parameter and should contain data for both control and exposed animals. Standard errors of parameter values should be indicated by vertical lines, pointing up or down depending on the relative position of the control and exposure lines.

The format of the edited ASCII file is shown below.

The program LPROG reads data from columns 1-2, 5-7, 9-10, 11-18, and 28-34, for specified parameter and tissue.

The first two letters are the "tissue code" (MC=muscle control, ML=muscle low pH, and similarly LC and LL for liver). The next 2-3 characters are the "parameter code". The following two digits give time in days. The next number is the mean and the last number is its standard error (we are not interested in the middle number). The data are arranged by tissues, then by parameters and by time.

LPROG searches through the file until it finds the tissue control. Then it finds the parameter, retrieves the data sequentially, then finds the same parameter in the "exposed" tissue. Finally, LPROG determines the range of the values and which is larger (control or low pH) and indicates this by printing 1 or 2 in the fifth field of the 2647A screen.

```
123456789012345678901234567890123456789 <--Column numbers
MC  ATP 01  6.3890   1.2481  .5095
MC  ATP 03  7.7632   2.8840 1.1774
MC  ATP 05  6.7787    .6104  .2492
MC  ATP 07  7.0815   1.1865  .4844
MC  ATP 09  5.8852   1.0896  .4448
123456789012345678901234567890123456789 <-- Column numbers
```

```
LPROG
  10 DIM M[16,2],E[16,3],T$[2],P$[3]
  11 DIM G$[2],M$[8],E$[7],T1$[2],P1$[2]
  15 DIM B$[72],N$[8]
  20 FILES *,*
  30 N$="VLOUT"
  35 MAT E=ZER
  36 MAT M=ZER
  40 ASSIGN N$,1,F1
  50 IF F1=0 THEN 100
  60 IF F1=3 THEN PRINT "File does not exist"
  70 IF F1<>3 THEN PRINT "Another file problem"
  80 STOP
 100 INPUT "Group (MC or LC ) ",T$
 101 PRINT "The parameters are:"
 102 PRINT "    ATP,ADP,AMP,CP,TOT,AEC,GLU,GLY,LTH,WT"
 110 INPUT "Parameter ",P$
 115 J=1
 116 I2=16
 117 IF P$="GLU" OR P$="GLY" THEN I2=14
 120 ON END #1 THEN 1000
 130 LINPUT #1;B$
 140 IF B$[1;2]=T$ THEN 160      <== Find the first "tissue" line
 150 GOTO 130
 155 LINPUT #1;B$
 160 IF B$[5;3]=P$ THEN 180
 161 IF LEN(P$)=3 THEN 170
 162 IF B$[5;2]=P$ THEN 180   <== For 2-character parameter codes
 170 GOTO 155
 180 M$=B$[11;8]          <== Start the reading
 185 E$=B$[28;7]                 "   "
 190 CONVERT M$ TO M[1,J]
 195 CONVERT E$ TO E[1,J]
 200 FOR I=2 TO I2
 210   LINPUT #1;B$
 220   M$=B$[11;8]              continue until all I2 value pairs
 225   E$=B$[28;7]                   are read
 230   CONVERT M$ TO M[I,J]
 235   CONVERT E$ TO E[I,J]
 240 NEXT I
 245 IF J=2 THEN 300
 250 IF T$="MC" THEN T$="ML"   <== Switch to "low pH"
 260 IF T$="LC" THEN T$="LL"
 270 J=2
 280 GOTO 130
 300 A=1E6
 301 B=0
 310 FOR I=1 TO I2
 320   FOR J=1 TO 2
 330     IF M[I,J]-E[I,J]>A THEN 360
 340     A=M[I,J]-E[I,J]                <== Do the scaling
 360     IF M[I,J]+E[I,J]<B THEN 380
 370     B=M[I,J]+E[I,J]
 380   NEXT J
 390   E[I,3]=1
 400   IF M[I,1]<M[I,2] THEN E[I,3]=2
 410 NEXT I

 415 PRINT T$;" ";P$
 420 FOR I=1 TO 16
 430   PRINT USING "3D.4D,2X";M[I,1],E[I,1],M[I,2],E[I,2],E[I,3]
 440 NEXT I
 450 PRINT "Minimum ";A,"Maximum ";B
1000 END
```

The output looks like this:

```
Group (MC or LC ) MC
The parameters are:
    ATP,ADP,AMP,CP,TOT,AEC,GLU,GLY,LTH,WT
Parameter TOT
ML TOT
  7.3443     .5048    7.3652     .4144    2.0000
  8.6747    1.1894    7.6494     .2870    1.0000
  7.5797     .2303    6.7590     .5468    1.0000
  7.8753     .4871    7.8643     .4741    1.0000
  6.8095     .4406    7.6070     .1073    2.0000
  6.4982     .5707    6.5840     .6881    2.0000
  6.7582     .7078    6.4335     .9609    1.0000
  7.6288     .2778    6.6358     .5703    1.0000
  7.2010     .7070    6.2108     .3801    1.0000
  7.2990     .4302    7.3610     .6398    2.0000
  8.0370     .3294    7.0418     .2645    1.0000
  7.2090     .4659    7.8485     .8040    2.0000
  7.1305     .2572    7.5815     .9983    2.0000
  6.8075     .2044    6.2195     .5783    1.0000
  7.6967     .1309    6.0948     .2591    1.0000
  7.4445     .1678    5.7422     .5818    1.0000
Minimum  5.1604                Maximum  9.8641
```

The Minimum and Maximum are values on the Y-axis. The user is later asked for the desired range for the Y-axis. The plotting program LOUPLT tries to accommodate the range in steps of 1,2 or 5 times 10**x units, starting at a multiple of the step.

The plotting program, LOUPLT is in TBASIC and must be loaded in the TBASIC interpreter as described above.

LOUPLT is activated by the "]" key. It asks for the range on the Y-axis. The program will do the plotting, then ask for Y-axis label, its placement, and similarly, for a master label (Fig. 2). You can regain control by <control A> over the Terminal BASIC, and by <control Y> over the HP3000 BASIC.

```
LOUPLT
10 DIM X(16),M1$(10),M2$(10),E1$(10),E2$(10),M(16,2),E(16,3) <------
20 DIM T$(2),P$(3),Y$(50)                                          |
30 DATA 1,3,5,7,9,11,13,20,27,34,41,48,55,62,69,76                 |
40 DATA 3,5,7,9,13,20,27,34,41,48,55,62,69,76                      |
45 L1$="CONTROL"\L2$="LOW PH"                                      |
50 Rmote$=CHR$(27)&"&k1R"                              Initialize  |
60 Locl$=CHR$(27)&"&k0R"                                           |
70 PRINT CHR$(141)                                                 |
80 PLOTR                                                           |
85 FXD (0,2)                                                       |
90 ON KEY #93 GOTO 110                    <------------------------
95 REM ------------------------------------------
100 SLEEP
110 PRINT Locl$;
120 LOCATE (20,195,6,98)                  <------------------------|
130 INPUT "Y start ",Y0                                            |
140 INPUT "Y maximum ",Y1                 Draw & scale axes        |
150 SCALE (0,80,Y0,Y1)                    <------------------------
160 PRINT MOVCR(-22,-1)
170 T$=DSPIN$(-2,X1)                      <------------------------
180 P$=DSPIN$(-4,X1)                                               |
185 P$=TRIM$(P$)                                                   |
190 I2=16                                                          |
200 IF P$="GLU" OR P$="GLY" THEN GOTO 730                          |
210 FOR I=1 TO I2\ READ X(I)\ NEXT I                               |
220 PRINT MOVCR(0,-6)                                              |
230 FOR I=1 TO I2                                Read display      |
240 V1$=DSPIN$(-9,X1)                                              |
250 E1$=DSPIN$(-10,X1)                                             |
260 V2$=DSPIN$(-10,X1)                                             |
270 E2$=DSPIN$(-10,X1)                                             |
280 E3$=DSPIN$(-9,X1)                                              |
290 M(I,1)=VAL(V1$)\E(I,1)=VAL(E1$)\M(I,2)=VAL(V2$)\E(I,2)=VAL(E2$)|
300 E(I,3)=VAL(E3$)                                                |
310 PRINT MOVCR(0,-46)                                             |
320 NEXT I                                <------------------------
```

```
325 REM ------------------------------------------
330 PRINT CHR$(27)&"J";
340 PRINT CHR$(143)   !A DISP OFF
350 FRAME
355 GOSUB 800         !Y AXIS SCALING
360 LAXES (-2,Y2,0,Y0,2,1,2)
365 REM ------------------------------------------
370 FOR J=1 TO 2                              <----------------------
380 J1=(J-1)*6                                                       |
390 LINE (J1)                                                        |
400 MOVE (X(1),M(1,J))                                               |
410 I=1                                                              |
420 GOSUB 650        !STD ERROR DRAWING                              |
430 FOR I=2 TO I2                                    Plotting        |
440 DRAW (X(I),M(I,J))                                               |
450 GOSUB 650        !STD ERROR DRAWING                              |
460 NEXT I                                                           |
470 NEXT J                                    <----------------------
475 REM ------------------------------------------
480 PRINT CHR$(142)\ PRINT CHR$(143)  !G DISP OFF, A DISP ON
490 INPUT "Y-axis label ",Y$                  <----------------------
500 PRINT "Indicate starting position of Y-axis label by graphics curso
r"                                                                   |
510 PRINT CHR$(142)\ PRINT CHR$(136)                                 |
520 INPUT "Type / <RETURN> ",W$                                      |
530 LDIR (90)                                                        |
535 CURSOR (X,Y)\MOVE (X,Y)                                          |
540 PRINT #0;Y$                                                      |
550 INPUT "Master label ",Y$                         Labels          |
560 PRINT "Place the graphics cursor"                                |
565 LDIR (0)                                                         |
570 CSIZE (6)                                                        |
590 INPUT "Type / <RETURN> ",W$                                      |
595 CURSOR (X,Y)\MOVE (X,Y)                                          |
600 PRINT #0;Y$                                                      |
610 CSIZE (4)                                                        |
615 INPUT "Place cursor for CONTROL label, / <RETURN> ",W$           |
616 CURSOR (X,Y)\MOVE (X,Y)\ PRINT #0;L1$                            |
617 INPUT "Place cursor for LOW PH label, /<RETURN> ",W$             |
618 CURSOR (X,Y)\MOVE (X,Y)\ PRINT #0;L2$ <----------------------
620 PRINT Rmote$
630 PRINT CHR$(143)\ PRINT CHR$(136)
640 END
645 REM ------------------------------------------
650 LINE (1)                                          <-----------|
660 IF E(I,3)=1 AND J=1 THEN DRAW (X(I),M(I,J)+E(I,J))            |
670 IF E(I,3)=2 AND J=1 THEN DRAW (X(I),M(I,J)-E(I,J))  Standard  |
680 IF E(I,3)=1 AND J=2 THEN DRAW (X(I),M(I,J)-E(I,J))   error    |
690 IF E(I,3)=2 AND J=2 THEN DRAW (X(I),M(I,J)+E(I,J))   drawing  |
700 MOVE (X(I),M(I,J))                                            |
710 LINE (J1)                                                     |
720 RETURN                                            <-----------
725 REM ------------------------------------------
730 RESTORE 40                                <----------------------
740 I2=14                                                            |
750 FOR I=1 TO I2\ READ X(I)\ NEXT I         Changing times          |
760 RESTORE 30                                                       |
770 GOTO 220                                  <----------------------
775 REM ------------------------------------------
800 Y2=Y1-Y0                                  <----------------------
810 Y3=.434295*LOG(Y2)                                               |
815 IF Y3<0 THEN 900                                                 |
820 Y4=INT(Y3)    !EXPONENT                                          |
830 Y5=Y3-Y4      !MANTISSA                                          |
840 Y6=0                                                 Step        |
850 IF Y5<=.30103 THEN Y6=.30103                         calculation |
860 IF Y5>.30103 AND Y5<=.69897 THEN Y6=.69897                       |
870 IF Y5>.69897 THEN Y6=1                                           |
880 Y2=10^(Y4+Y6-1)                                                  |
881 Y3=INT((Y0+Y2)/Y2)                                               |
882 Y0=(Y3-1)*Y2                                                     |
890 RETURN                                    <----------------------
895 REM ------------------------------------------
```

```
900 FOR I=1 TO 30                    <--------------------------
910 Y3=Y3+1                                                    |
920 IF Y3>0 THEN 940                   Negative characteristic |
930 NEXT I                             format conversion       |
940 Y4=-I\Y5=Y3                                                |
950 GOTO 840                         <--------------------------
955 REM ------------------------------------------
```

ACKNOWLEDGMENTS

Dr. K. Haya and Mr. L. E. Burridge commented on the manuscript. Mrs. Jeanine Hurley formatted the manuscript report and Mrs. Ruth Garnett looked after the editorial aspects.

REFERENCES

1. Hewlett Packard 1979. Intelligent Graphics 2647A User's Manual. Part Number 02647-90001.

2. Hewlett Packard 1978. Intelligent Graphics 2647A Quick Reference Guide. Part Number 02647-90006.

3. Schifreen, R.S. 1983. A simple BASIC program for automatic scaling of Cartesian graphics. Am. Lab. 13(9), 140.

4. Reischel, M.T. 1984. A technique for scaling Cartesian graphics. Am. Lab. 14(2), 136.

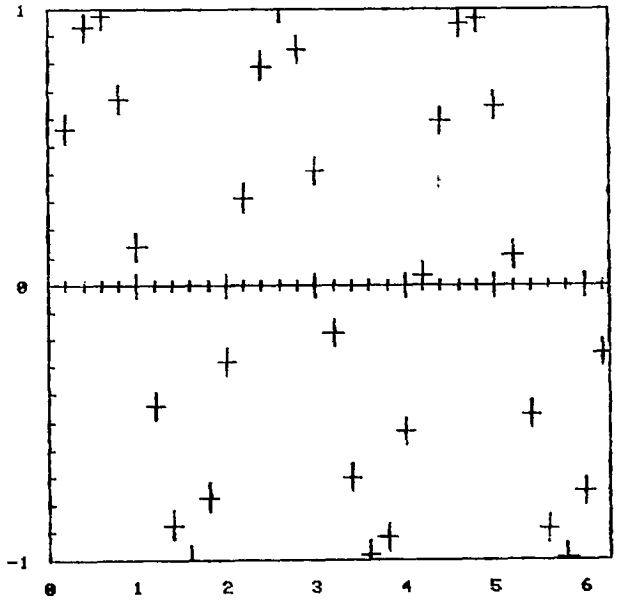

Fig. 1. Plot of Y=sin(3x) produced by DEMO1 from data provided by TEMP2.

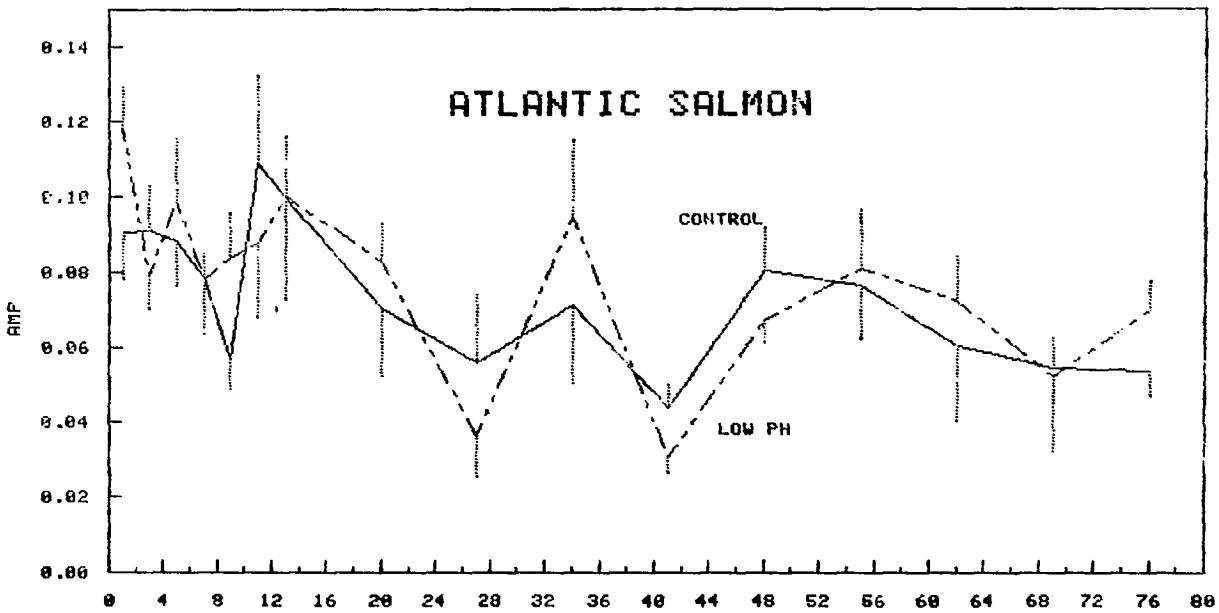

Fig. 2. Plot produced by LOUPLT of data retrieved by LPROG from an ASCII file.