TI PPC NOTES

NEWSLETTER OF THE TI PERSONAL PROGRAMMABLE CALCULATOR CLUB

P. O. Box 1421, Largo, FL 34649

Volume 14, Number 4 — Extra Issue — April 1991

This "extra issue" is distributed to make up for the smaller than normal third issue in the current subscription year. I expect that the final issue for the current subscription year should be of normal size and be distributed in late summer. I also expect that the final issue for the current subscription year will be the last issue of our newsletter, at least with the present editor. The primary reason for that decision is that I have become a bit tired of it all after seven years as editor. Also, we have come upon "hard times" where we have only 139 members. That is by far the lowest number of members as we approach the final issue of a subscription year since I have been the editor. So, if you have hardware or software for sale, hardware or software you would like to buy, problems you would like to have solved, and the like then I suggest that you send the material to me as soon as possible and I will try to include the information in the final issue.

I also find that the source of TI-59's and PC-100's from the engineers at my former employer seems to be drying up. So, if you want to have a backup unit I suggest that you let me know. As in previous years, the costs are fifty dollars plus shipping. Send sixty-five dollars for each unit you want. I will return any excess shipping costs.

What will you find in this issue? The emphasis is on programs for the TI-81 and on translations from one machine to another. You will also find the first demonstration of the Printer 80 with the TI-74.

TABLE OF CONTENTS

SWAP ALGORITHMS 3
POLYNOMIAL CURVE FIT ON THE TI-81, P. Hanson 4
DISPLAYING COMBINATIONS, L. Leeds 8
"HANG-UP" OF THE TI-95, P. Messer 11
BATTERY LIFE IN THE TI-81, P. Hanson 11
THE WAFERTAPE DIGITAL TAPE DRIVE 12
RELIABILITY TEST ANALYSIS, T. Stewart, R. Cornwell, P. Hanson 14
MORE ON PI FINDING 17
PRINTED MATTER AVAILABILITY 2, 3
HARDWARE AVAILABILITY 2, 8, 11, 16

We continue to offer magnetic tape service for the TI-74 and the TI-95 for programs in the newsletter. Send five dollars for a tape with 10 programs.

<u>fx-7500G</u> <u>HINGES</u> - Gene Friel writes "One problem I had soon after buying a fx-7500G from ElekTek was the plastic hinge cracking. It's still useable but buyers should be careful not to open the calculator more than 180 degrees.

<u>ANOTHER</u> <u>NEWSLETTER</u> <u>OF</u> <u>RECREATIONAL</u> <u>COMPUTING</u> - V13N2P11 included an advertisement for the *Journal of Recreational Mathematics*. V14N1P15 reported the availability of the newsletter *Algorithm* edited by A. K. Dewdney, the long-time editor of the "Computer Recreations" column in *Scientific American*. There is yet another newsletter which emphasizes recreational computing, *Recreational and Educational Computing*. It is edited and published by Dr. Michael Ecker who also writes the "Easy Pieces" column for *Algorithm*. The price is $27.00 for a calendar year of eight issues. A three issue sample subscription is offered for $10.00 where that amount is creditable toward a full subscription. Write to

Dr. Michael W. Ecker, Editor/Publisher
Recreational & Educational Computing
909 Violet Terrace
Clarks Summit PA 18411

The *Journal of Recreational Mathematics* continues to offer a complimentary sample issue to prospective subscribers. Write to

Baywood Publishing Company, Inc.
26 Austin Avenue, Box 337
Amityville NY 11701

<u>MORE</u> <u>MEMBER</u> <u>LISTINGS</u> - V14N2P7-9, V14N2P13 and V14N3P18 listed the names, addresses, telephone numbers and interest areas for those members who were willing to be listed. Additional listings follow:

Peter K. Englund
503 Calumet Street
Lake Linden MI 49945
(806)-296-0494
TI-95, Navigation, Astronomy, Amateur Radio, Interfaces to Other Devices

Paul R. Braca
1121 Virginia Avenue
Bronx NY 10472
(212)-824-6120
Electronics puzzles and mathematics

Andreas Pack
Eickelerstr. 60
4690 Herne 2
Federal Republic of Germany

<u>MORE</u> <u>BENCHMARK</u> <u>RESULTS</u> - Dan Eicher ran the 1.0000001 squared 27 times benchmark on his CC-40 and TI-99/4A. HIs results were:

	squared 27 times	1.0000001^134217728
Exact Answer	674530.47074 10845 59...	674530.47074 10845 59...
TI-99/4A	674530.318	674530.9234
CC-40	674530.318	674621.4635

SWAP ALGORITHMS - Michael Ecker's column "Easy Pieces" in the March/April 1990 issue of *Algorithm* challenged users to develop a BASIC routine which perform the equivalent of the exchange (EXC) command which is available in many calculators. The routine was to:

> "... effect the swap by putting the value currently in A into B, without losing the value in B, and putting the original value of B in A. ... you cannot use any built-in swap command, nor can you use data values, nor can you use any foreknowledge of what values A and B might have."

One of the more obvious approaches is to combine the two values in one of the registers and then separate the values with appropriate subtractions, say with a sequence such as A=A+B:B=A-B:A=A-B . However, I recognized that that method will not work for some combinations of numbers; for example, with the TI-74 try A = 50000000 and B = 0.0000005 . After the swap A = 0 and B = 50000000 . But surprising results can be obtained with some machines. For example, my Radio Shack Color Computer, which uses radix 2 arithmetic, will appear to yield the correct result for an input such as A = 15.2 and B = 25.6, but if one saves the input, say with the statements AA = A and BB = B, then after the swap AA-B = 3.7252903E-09 but BB-A = 0 !

The "Easy Pieces" column in the November 1990 issue of *Algorithm* presented the same solution and noted that "...limitations in arithmetic precision and overflow might interfere with a swap of this form." Ecker then described solutions which use Boolean logic. Unfortunately, in many (maybe all) BASIC implementations the Boolean logic functions are limited to integers in the range from -32768 to +32767 . Furthermore, the BASIC implementations at my disposal convert non-integer values to integers by truncation (in the Model 100) or by rounding (in the CC-40 or TI-74) prior to performimg the Boolean functions. In conclusion, while there do not appear to be any general BASIC solutions to the swap problem the attempt to generate solutions and to explain the results will give the programmer a better feel for how a machine handles numbers.

RECENT PRICES - On March 30 Service Merchandise was offering the TI-81 for $89.97 and the fx-7000GA for $69.73 . TI-68's were available at Sears for less than 50 dollars.

PPX PROGRAM AVAILABILITY - Robert Prins donated several programs which provide an alternate source to those previously listed. Carl Rabe also donated several programs which perform multiprecision arithmetic, some of which are new listings.

1 398234 - 24 Digit Multiplication

14B 398239 - Triple Precision Arithmetic

1 398273 - Extended Precision Multiplication

134 398284 - Twenty Digit Arithmetic

19Q 908101 - Histogram, Bar Graph, Point Plotter

19Q 918178 - Starwars

1G 938041 - Position from Two Vortacs

POLYNOMIAL CURVE FIT ON THE TI-81 - P. Hanson

This program provides the capability to fit polynomials to a set of input data pairs. The highest degree is limited to five by the maximum dimensions of the matrices in the TI-81. The technique used is an extension of that for straight lines and parabolas in Chapter 13 of Spiegel's *Statistics* in the Schaum Outline Series in Mathematics. The fitted polynomials are of the form

$$Y = A0 + A1x + A2x^2 + A3x^3 + a4x^4 + A5x^5$$

User Instructions:

1. Run ClrStat to clear the statistical data as described on page 7-3 of the *TI-81 Graphics Calculator Guidebook*.

2. Select the Edit option from the STAT DATA menu and enter the data pairs to be fitted. See page 7-4 of the guidebook.

3. When data entry is complete leave the statistics menu and execute the program.

4. The calculator stops with the notation PAIRS(M)= and the number of pairs on the screen.

5. Press ENTER. The calculator selects the appropriate RANGE parameters and stops with the input data displayed in the xyLine format (page 7-15 of the guidebook). The data will have been sorted such that the X values are in ascending order. This step allows you to review the input data for wild points before proceeding with the curve fitting process.

6. When you are satisfied with the input data press ENTER. The calculator will stop with the notation DEGREE(<6,<M-2)= on the screen followed by a question mark on the next line. You are limited to a degree less than six by the maximum size of the matrices, and to a degree less than (M-2) by the curve fitting process. Enter the degree of the polynomial you wish to use to fit the data and press ENTER.

7. The calculator will run for a short period of time and then:

   a. Plot the calculated polynomial using a dotted line format.

   b. Superimpose the input data in the xyLine format.

Compare the polynomial curve with the input data for wild points and to determine if a different degree polynomial should be used. You cannot use the cursor or the graphing keys at this point.

8. Press ENTER and the calculator stops with the statistics of the fit. The number on the top line is the mean of the residual errors. It would be zero if the calculations were exact. It will actually be a small number due to truncation and rounding effects. The value S.E. is the unbiased standard error defined by the equation

$$SE = \left(\frac{\sum_{i=1}^{M} (Y_i - P(x_i))^2}{M - (n+1)} \right)^{1/2}$$

where M is the number of data pairs and n is the degree of the polynomial. The values of the extremes of the residuals (DYmax and DYmin) are also displayed.

9. Press ENTER. The calculator will stop with a plot of the residuals in an xyLine format. You can use the cursor to examine the size of individual residuals.

Polynomial Curve Fit on the TI-81 - (cont)

10. Press MATRIX ▶ 3 to see the coefficients of the polynomial where [C](1,1) is A0, [C](2,1) is A1, etc.

11. The data pairs in the statistics data were sorted so that the x values are in ascending order. Otherwise, the input data has not been changed. Thus, it is not necessary to re-enter the input data to try a polynomial with a different degree.

Sample Problem:

Enter the set of five data pairs (-7, -81), (-4, 27), (0, 3), (2, 27), and (5, 243) which are points on the polynomial

$$y = 3 - 2x + 5x^2 + x^3$$

The following tables show the displayed results for degrees 1, 2 and 3.

Linear	Quadratic	Cubic
2e-12	4e-12	-3.72e-10
SE=	SE=	SE=
76.57905995	74.02670139	1.051665346e-9
DYmin=	DYmin=	DYmin=
-75.48898678	-54.15634047	-7.4e-10
DYmax=	DYmax=	DYmax=
77.62995595	74.40243196	-1e-10

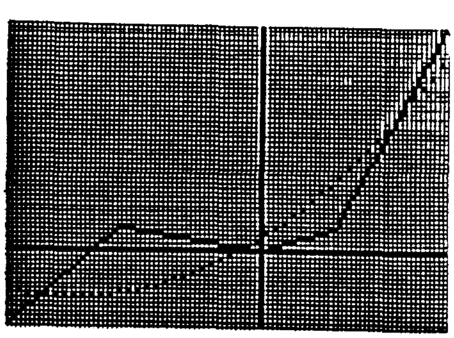
Linear	Quadratic	Cubic
[C] 2x1	[C] 3x1	[C] 4x1
1,1=60.568281938	1,1=18.618413434	1,1=3.0000000007
2,1=20.960352423	2,1=26.347712797	2,1=-2
	3,1=2.4606253619	3,1=5
		4,1=1

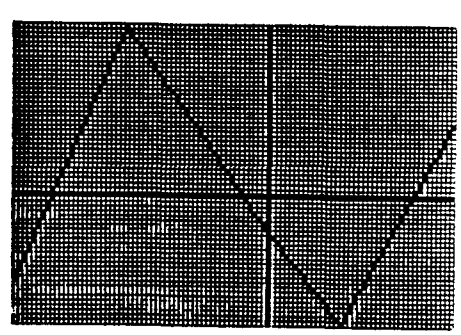
The calculated coefficients for the cubic have been rounded for the display. The actual coefficients for the cubic are:

A0 = 3.0000000007
A1 = -1.999999999926
A2 = 4.999999999977
A3 = 0.999999999997

where, for example, the exact value for A2 is obtained from the evaluation of the expression [C](3,1) - 5 as -2.3e-11 .

Sample plots for the polynomial and for the residuals for the second degree solution follow:





Polynomial Curve Fit for the TI-81 - (cont)

Comments on the Program:

Lines 1 through 3 provide some initialization which is needed later to provide a dot graph for the solution from the least squares fit.

Lines 4 through 7 copy the number of data pairs from Dim{x} into variable M and then display that number with annotation.

Lines 8 through 15 set the RANGE variables to accommodate the input data pairs. The data pairs are sorted with respect to the y values and the first and last y values are used as Ymin and Ymax. Then, the data pairs are sorted with respect to the x values and the first and last x values are used as Xmin and Xmax.

Lines 16 and 17 display the data pairs graphically and stop the calculator to permit review of the input data from the plot.

Lines 18 through 25 provide for user entry of the degree of the polynomial to be used for the solution. The input is rejected if it would cause either the allowable matrix dimensions of the calculator to be exceeded or the least squares algorithm to fail.

Lines 26 through 36 set the matrix dimensions to be consistent with the order (degree + 1) of the desired solution and clear the matrix elements. The use of 6 for Crow (see line 31) clears the unused elements of the C matrix. The resulting zeroes will be used later in the program.

Lines 37 through 56 generate the matrix [A] and the vector [B] which are needed to solve for the best fit polynomial. The steps are a straightforward translation of the C-40 BASIC routine from V8N5P17:

```
190 FOR L=1 TO K
200 H(1)=1
210 FOR I=2 TO N
220 H(I)=H(I-1)*X(L):NEXT I
230 FOR I=1 TO N:FOR J=1 TO N
240 A(I,J)=A(I,J)+H(I)*H(J):NEXT J
250 B(I)=B(I)+H(I)*Y(L):NEXT I
260 NEXT L
```

where the C matrix of the TI-81 implementation is equivalent to the H matrix of the BASIC routine. In the BASIC routine K is the number of data pairs and N is the order of the matrix. Note that the IS function provides a particularly easy translation of BASIC FOR-NEXT routines to the TI-81 language.

```
   Prgm1:POLYFIT
 1 :All-Off
   :Dot
   :2→Xres
   :Dim{x}→M
 5 :Disp "PAIRS(M)=
   "
   :Disp M
   :Pause
   :ySort
   :{y}(1)→Ymin
10 :{y}(M)→Ymax
   :xSort
   :{x}(1)→Xmin
   :{x}(M)→Xmax
   :0→Xscl
15 :0→Yscl
   :xyLine
   :Pause
   :Lbl A
   :Disp "DEGREE(<6
   ,<M-2)="
20 :Input N
   :N+1→N
   :If N<M
   :Goto B
   :Disp "TOO LARGE
   "
25 :Goto A
   :Lbl B
   :N→Arow
   :N→Acol
   :N→Brow
30 :1→Bcol
   :6→Crow
   :1→Ccol
   :0→[A]
   :0→[B]
35 :0→[C]
   :N→Crow
   :1→L
   :Lbl C
   :1→[C](1,1)
40 :1→K
   :Lbl D
   :[C](K,1){x}(L)→
   [C](K+1,1)
   :IS>(K,N-1)
   :Goto D
45 :1→I
   :Lbl E
   :1→J
   :Lbl F
   :[A](I,J)+[C](I,
   1)[C](J,1)→[A](I
   ,J)
50 :IS>(J,N)
   :Goto F
```

Polynomial Curve Fit for the TI-81 - (cont)

Line 57 solves the set of linear equations generated in lines 37 through 56 and place the result in the first column of matrix [C].

Line 58 sets the Crow dimension to 6 for use in the calculations to follow.

Lines 59 defines the polynomial solution as the Y1 expression. In the polynomial $x^{2\,2}$ yields x^4. All six elements of matrix [C] are included so that a single Y1 expression can be used. Those elements for which the first subscript is greater than the order of the solution will be zero due to the use of the dimension Crow = 6 during the matrix clearing in lines 26 through 35. If Crow were not set to 6 in line 58 an error would occur. The polynomial is plotted as a dotted line due to the initialization in lines 2 and 3.

Line 60 superimposes the input data on the polynomial plot.

Line 61 stops the calculator with the polynomial and the input data on the screen.

Lines 62 through 79 calculate the residuals between the polynomial and the input y values, accumulate the sums needed for statistical evaluation of the fit, and set RANGE variables Ymin and Ymax to be compatible with the range of the residual errors. In lines 65, 66 and 71 the value of the polynomial at a given value of the independent variable is obtained by storing that value in the X register and calling Y1.

Lines 80 through 89 display the results of the statistical evaluation of the polynomial fit including the mean of the residuals, the unbiased standard error and the maximum and minimum residuals. The annotation "DYmin=" in line 85 is assembled with the key sequence ALPHA " ALPHA D VARS ◀ 4 2nd TEST 1 ALPHA " .

Lines 90 through 99 generate a plot of the residuals using the Line capability of the TI-81. An alternative method would have been to store the residuals in the statistical data, replacing the input y values and then using the xyLine command. However, that method would have made it necessary to reenter the y values if the user wanted to try a polynomial of a different degree. Again, note that the expression defined as Y1 can be evaluated by storing the desired independent variable value in X and calling Y1.

Line 99 sets Crow equal to the order of the solution. This prevents unwanted zeroes from being displayed when the coefficients are recalled using the EDIT option of the MATRIX function.

```
52 :[B](I,1)+[C](I,
   1){y}(L)→[B](I,1
   )
   :IS>(I,N)
   :Goto E
55 :IS>(L,M)
   :Goto C
   :[A]^-1[B]→[C]
   :6→Crow
   :"[C](1,1)+[C](2
   ,1)X+[C](3,1)X^2+
   [C](4,1)X^3+[C](5
   ,1)X^2^2+[C](6,1)X
   ^2X^3"→Y1
60 :xyLine
   :Pause
   :Y1→Off
   :1→I
   :{x}(1)→X
65 :{y}(1)-Y1→Ymin
   :{y}(1)-Y1→Ymax
   :0→R
   :0→S
   :Lbl G
70 :{x}(I)→X
   :{y}(I)-Y1→D
   :R+D→R
   :S+D^2→S
   :If D<Ymin
75 :D→Ymin
   :If D>Ymax
   :D→Ymax
   :IS>(I,M)
   :Goto G
80 :R/M→R
   :Disp R
   :Disp "SE="
   :√(S/(M-N))→S
   :Disp S
85 :Disp "DYmin="
   :Disp Ymin
   :Disp "DYmax="
   :Disp Ymax
   :Pause
90 :1→I
   :Lbl H
   :{x}(I)→X
   :{y}(I)-Y1→P
   :{x}(I+1)→X
95 :{y}(I+1)-Y1→Q
   :Line({x}(I),P,{
   x}(I+1),Q)
   :IS>(I,M-1)
   :Goto H
99 :N→Crow
   :Stop
```

604 Bytes

DISPLAYING COMBINATIONS - Larry Leeds wrote that he had a problem which required the display (or printout) of all available combinations of n things taken r at time. That is different from the standard calculation of the number of combinations (nCr). In response I noted that the solution for the case where r = 3 was implicit in the smallest circle program (see lines 345-355 and 735-760 on V13N1P29). I used those steps as the starting point for the short TI=74 program at the right. A sample output appears below the program.

```
100 INPUT "Use PC-324? Y
/N ";Z$
110 IF Z$="Y"OR Z$="y"TH
EN PN=1 ELSE 130
120 OPEN #1,"12",OUTPUT
130 PRINT #PN
140 PRINT #PN,"nC3 Combi
nations":PAUSE 1
150 INPUT "n = ";N
160 PRINT #PN,"n = ";N:P
AUSE 1
200 FOR I=N TO 3 STEP -1
210 FOR J=(I-1)TO 2 STEP
 -1
220 FOR K=(J-1)TO 1 STEP
 -1
230 PRINT #PN,I;J;K
240 IF PN=0 THEN PAUSE
250 NEXT K
260 NEXT J
270 NEXT I
999 END
```

```
nC3 Combinations
n =  5
 5  4  3
 5  4  2
 5  4  1
 5  3  2
 5  3  1
 5  2  1
 4  3  2
 4  3  1
 4  2  1
 3  2  1
```

In a few days Larry responded with a BASIC program which will handle values of r from 2 through 9. I converted his program for the TI-74. I also incorporated the prompt for the use of the Printer 80 (see V14N3P14) in line 50. The program listing as obtained from the Printer 80 appears on page 9.

Sample printouts with various printers which can be interfaced with the TI-74 appear on page 10. For those printouts the PC-324 was connected to the TI-74 through the built-in cable, the HX-1000 was connected to the PC-324 through the adapter cable defined by Maurice Swinnen in V12N3p13, and the Printer 80 was connected to the HX-1000 through a standard hex bus cable. An appropriate power supply was connected to each printer yielding a veritable rat's nest of cabling on my kitchen table.

On the surface Larry's program appears to be a "brute force" implementation which could be substantially shortened with the use of subscripts, subroutines, ON-GOTO's and the like. So far I haven't had much success in optimization, primarily because the BASIC implementations at my disposal (Microsoft BASIC in the Model 100 and TI BASIC in the CC-40 and TI-74) do not support the use of subscripted variables as the index in FOR-NEXT statements. Members are invited to submit improvements on Larry's program where the desired result (at least in my view) would be a program which could handle any r within the range of a DIM statement.

Displaying Combinations - (cont)

Program Listing on the Printer 80

```
10 Y$="nCr Combinations":PRINT Y$:PAUSE 1
15 REM 1 April 1991
25 INPUT "Use printer? Y/N ";Z$
30 IF Z$="Y"OR Z$="y"THEN PN=1 ELSE 100
35 PRINT "Device Numbers:":PAUSE 1
40 PRINT "For the HX-1000 enter 10":PAUSE 1
45 PRINT "For the PC-324 enter 12":PAUSE 1
50 PRINT "For the Printer 80 enter 16":PAUSE 1
55 INPUT "Enter device number ";D$
60 OPEN #1,D$,OUTPUT
65 IF D$="10"THEN PRINT #1,CHR$(18)
75 PRINT #1:PRINT #1,Y$:PRINT #1
100 INPUT "n = ";N
105 INPUT "r (r<n; 1<r<10) = ";R
110 IF R<2 OR R>9 OR R>N-1 THEN PRINT "r is out of range" ELSE 120
115 PAUSE:GOTO 100
120 P$=STR$(N)&"C"&STR$(R):PRINT #PN,P$
125 IF PN=0 THEN PAUSE ELSE PRINT #1
130 M=0:PAUSE ALL
135 FOR A=N TO R STEP -1
140 FOR B=A-1 TO R-1 STEP -1
145 IF R=2 THEN M=M+1:PRINT #PN,A;B:GOTO 290
150 FOR C=B-1 TO R-2 STEP -1
155 IF R=3 THEN M=M+1:PRINT #PN,A;B;C:GOTO 280
160 FOR D=C-1 TO R-3 STEP -1
165 IF R=4 THEN M=M+1:PRINT #PN,A;B;C;D:GOTO 270
170 FOR E=D-1 TO R-4 STEP -1
175 IF R=5 THEN M=M+1:PRINT #PN,A;B;C;D;E:GOTO 260
180 FOR F=E-1 TO R-5 STEP -1
185 IF R=6 THEN M=M+1:PRINT #PN,A;B;C;D;E;F:GOTO 250
190 FOR G=F-1 TO R-6 STEP -1
195 IF R=7 THEN M=M+1:PRINT #PN,A;B;C;D;E;F;G:GOTO 240
200 FOR H=G-1 TO R-7 STEP -1
205 IF R=8 THEN M=M+1:PRINT #PN,A;B;C;D;E;F;G;H:GOTO 230
210 FOR I=H-1 TO R-8 STEP -1
215 PRINT #PN,A;B;C;D;E;F;G;H;I:M=M+1
220 NEXT I
230 NEXT H
240 NEXT G
250 NEXT F
260 NEXT E
270 NEXT D
280 NEXT C
290 NEXT B
300 NEXT A
310 PAUSE 0:PRINT #PN
320 PRINT #PN,P$&"= ";M:IF PN=0 THEN PAUSE
330 PRINT #PN:GOTO 100
999 END
```

Displaying Combinations - (cont)

Sample Printouts for 10C8

Printer 80

```
nCr Combinations

10C8

 10   9   8   7   6   5   4   3
 10   9   8   7   6   5   4   2
 10   9   8   7   6   5   4   1
 10   9   8   7   6   5   3   2
 10   9   8   7   6   5   3   1
 10   9   8   7   6   5   2   1
 10   9   8   7   6   4   3   2
 10   9   8   7   6   4   3   1
 10   9   8   7   6   4   2   1
 10   9   8   7   6   3   2   1
 10   9   8   7   5   4   3   2
 10   9   8   7   5   4   3   1
 10   9   8   7   5   4   2   1
 10   9   8   7   5   3   2   1
 10   9   8   7   4   3   2   1
 10   9   8   6   5   4   3   2
 10   9   8   6   5   4   3   1
 10   9   8   6   5   4   2   1
 10   9   8   6   5   3   2   1
 10   9   8   6   4   3   2   1
 10   9   8   5   4   3   2   1
 10   9   7   6   5   4   3   2
 10   9   7   6   5   4   3   1
 10   9   7   6   5   4   2   1
 10   9   7   6   5   3   2   1
 10   9   7   6   4   3   2   1
 10   9   7   5   4   3   2   1
 10   9   6   5   4   3   2   1
 10   8   7   6   5   4   3   2
 10   8   7   6   5   4   3   1
 10   8   7   6   5   4   2   1
 10   8   7   6   5   3   2   1
 10   8   7   6   4   3   2   1
 10   8   7   5   4   3   2   1
 10   8   6   5   4   3   2   1
 10   7   6   5   4   3   2   1
 9   8   7   6   5   4   3   2
 9   8   7   6   5   4   3   1
 9   8   7   6   5   4   2   1
 9   8   7   6   5   3   2   1
 9   8   7   6   4   3   2   1
 9   8   7   5   4   3   2   1
 9   8   6   5   4   3   2   1
 9   7   6   5   4   3   2   1
 8   7   6   5   4   3   2   1

10C8=   45
```

HX-1000

```
nCr Combinations

10C8

10  9  8  7  6  5  4  3
10  9  8  7  6  5  4  2
10  9  8  7  6  5  4  1
10  9  8  7  6  5  3  2
10  9  8  7  6  5  3  1
10  9  8  7  6  5  2  1
10  9  8  7  6  4  3  2
10  9  8  7  6  4  3  1
10  9  8  7  6  4  2  1
10  9  8  7  6  3  2  1
10  9  8  7  5  4  3  2
10  9  8  7  5  4  3  1
10  9  8  7  5  4  2  1
10  9  8  7  5  3  2  1
10  9  8  7  4  3  2  1
10  9  8  6  5  4  3  2
10  9  8  6  5  4  3  1
10  9  8  6  5  4  2  1
10  9  8  6  5  3  2  1
10  9  8  6  4  3  2  1
10  9  8  5  4  3  2  1
10  9  7  6  5  4  3  2
10  9  7  6  5  4  3  1
10  9  7  6  5  4  2  1
10  9  7  6  5  3  2  1
10  9  7  6  4  3  2  1
10  9  7  5  4  3  2  1
10  9  6  5  4  3  2  1
10  8  7  6  5  4  3  2
10  8  7  6  5  4  3  1
10  8  7  6  5  4  2  1
10  8  7  6  5  3  2  1
10  8  7  6  4  3  2  1
10  8  7  5  4  3  2  1
10  8  6  5  4  3  2  1
10  7  6  5  4  3  2  1
9  8  7  6  5  4  3  2
9  8  7  6  5  4  3  1
9  8  7  6  5  4  2  1
9  8  7  6  5  3  2  1
9  8  7  6  4  3  2  1
9  8  7  5  4  3  2  1
9  8  6  5  4  3  2  1
9  7  6  5  4  3  2  1
8  7  6  5  4  3  2  1

10C8=  45
```

PC-324

```
nCr Combinations

10C8

 10   9   8   7   6   5   4   3
 10   9   8   7   6   5   4   2
 10   9   8   7   6   5   4   1
 10   9   8   7   6   5   3   2
 10   9   8   7   6   5   3   1
 10   9   8   7   6   5   2   1
 10   9   8   7   6   4   3   2
 10   9   8   7   6   4   3   1
 10   9   8   7   6   4   2   1
 10   9   8   7   6   3   2   1
 10   9   8   7   5   4   3   2
 10   9   8   7   5   4   3   1
 10   9   8   7   5   4   2   1
 10   9   8   7   5   3   2   1
 10   9   8   7   4   3   2   1
 10   9   8   6   5   4   3   2
 10   9   8   6   5   4   3   1
 10   9   8   6   5   4   2   1
 10   9   8   6   5   3   2   1
 10   9   8   6   4   3   2   1
 10   9   8   5   4   3   2   1
 10   9   7   6   5   4   3   2
 10   9   7   6   5   4   3   1
 10   9   7   6   5   4   2   1
 10   9   7   6   5   3   2   1
 10   9   7   6   4   3   2   1
 10   9   7   5   4   3   2   1
 10   9   6   5   4   3   2   1
 10   8   7   6   5   4   3   2
 10   8   7   6   5   4   3   1
 10   8   7   6   5   4   2   1
 10   8   7   6   5   3   2   1
 10   8   7   6   4   3   2   1
 10   8   7   5   4   3   2   1
 10   8   6   5   4   3   2   1
 10   7   6   5   4   3   2   1
 9   8   7   6   5   4   3   2
 9   8   7   6   5   4   3   1
 9   8   7   6   5   4   2   1
 9   8   7   6   5   3   2   1
 9   8   7   6   4   3   2   1
 9   8   7   5   4   3   2   1
 9   8   6   5   4   3   2   1
 9   7   6   5   4   3   2   1
 8   7   6   5   4   3   2   1

10C8=  45
```

"HANG-UP" OF THE TI-95 - Peter Messer called to ask if I had ever experienced a "hang-up" condition when using the TI-95 with the PC-324 printer. I replied that I remembered having the problem once when operating with a PC-324 with dead batteries but had difficulty duplicating the condition. Peter had similar difficulty in duplicating the condition, and seemed to get different results with different calculators.

The "hang-up" can be cleared by pressing the RESET button. Fortunately, as indicated on page A-2 of the *TI-95 User's Guide* the use of the RESET button does NOT clear data registers or program registers.

BATTERY LIFE IN THE TI-81 - A few weeks after the purchase of my first TI-81 I found that I needed to change the batteries. I thought that might be a result of some unusually heavy useage as I became familiar with the device. I loaned that unit to my daughter in college for the spring semester. Less than two months into the semester she called to report that the TI-81 was "acting up". I coached her in the use of the display contrast indicator as an indicator of battery status from page 1-5 of the *TI-81 Graphics Calculator Guidebook*. The indication was that a battery change was required. I also needed a battery change on my second TI-81 after only a few weeks of use. The TI-81 uses four AAA batteries and I have always used the alkaline versions. My fx-7000G's which use three lithium CR2032C batteries have operated for a year or more on a set. Have others experienced unexpectedly short battery life with the TI-81?

BATTERY USEAGE IN THE PC-324 - My PC-324 is also "battery eater" even though I seldom use it without an AC9201 attached. Has anyone else had this experience, or do I have a defective unit?

CC-40 HARDWARE AND SOFTWARE WANTED - write to Kelvin Cane,
3111 Artaban Place
Baltimore MD 21216

CC-40 HARDWARE WANTED - If you have an Editor/Assembler cartridge, a video interface or a Wafertape drive in working order write to M. J. M. Wright, 45 Centerville Drive, Salem, NH 03079.

TI-74 EQUIPMENT FOR SALE

TI-74	BASICCALC with manuals	$ 80.00
PC-324	Portable Printer with Paper-holder bracket	70.00
AC-9201	Adapter	10.00
PA-201	AC Interface	5.00
CI-7	Cassette Interface	20.00
CTR-73	Tape Recorder - Realistic	35.00
	TI-74 Learn BASIC Guidebook	5.00

A substantial reduction in price will be offered on a package deal. This equipment was all purchased new in October 1989 and used in one project and not used since. Write to Carl Rabe, P.O. Box 2941, Santa Maria, CA 93457-2941.

A HISTORICAL NOTE - from BITS, BYTES & PIXELS, Volume 6, Number 9

NEVER RELEASED OFFICIAL TI PERIPHERALS: THE WAFERTAPE DIGITAL TAPE DRIVE

a hands on description by Charles Good
Lima Ohio User Group

The WAFERTAPE DIGITAL TAPE DRIVE was supposed to be a step up from cassette data storage. The device is totally under the control of the computer (no manual rewinding or keeping track of cassette counter numbers). A directory on the tape allows the computer to automatically advance the tape to the beginning of any desired file. Recording data digitally (as 1 or 0, on or off) rather than as part of a continuous spectrum of sound frequencies as is done on a regular sound cassette recorder such as T.I.'s Data Recorder, was supposed to be a more reliable way of recording and retrieving data. Had this device been released in 1983 after its debut in January of that year, it and the also never released HexBus interface would have formed an inexpensive mass storage upgrade (compared to using a cassette recorder) for the 99/4A at a time when a new full PE box (SSSD drive, 32K, disk controller) cost anywhere from $550-$1200. The last 99/4A catalog published by TI in the fall of 1983 lists the Wafertape drive for $139.95 and the HexBus interface for $59.95.

The wafertape drive was shown at the Consumer Electronics Show in January 1983 together with the first showing of the CC40. It was to be the major mass storage device for the CC40, and is described and pictured in the user guides that come with most CC40 software and peripherales. Unfortunately these user guides now also come with an addendum sheet that states "The Wafertape Digital Tape Drive is not available." The non release of the Wafertape Drive left the CC40 totally without a mass storage device until the 1986/1987 introduction of the Mechatronic HexBus Quickdisk drive. This lack of a mass storage device probably killed most consumer interest in the CC40.

I recently purchased for $100 a working Wafertape Drive, serial number 0000007, ATA3883. It is my understanding that the ATA number is a date code, indicating in this case manufacture in the 38th week of 1983. Most 99/4A hardware and software modules have an ATA number. If my understanding of ATA numbers is correct, my Wafertape Drive was not one of those shown at the January 1983 exhibit. I personally know of one other working Wafertape drive, serial number 0000095. I have been told that there are about 10 or 11 working Wafertape Drives in private hands and others that don't work.

The Wafertape Drive is comparable in size to other TI HexBus peripherals, measuring about 11.5cm wide, 14.5cm deep and 3.5cm tall. It is designed to be stacked with the HexBus RS232, modem, and Printer/Plotter. It is battery powered with 4 AA cells. You can also use an AC adapter. The front has an on/off switch and a slot for inserting the wafertape. On the back are two HexBus ports, an AC adapter jack, and a rotary switch for setting the device number. The switch on mine has positions 0-9. However, only positions 0-7 work, corresponding to devices 1-8. By accessing each HexBus peripheral individually by number, a single CC40 could control up to 8 Wafertape Drives, if one could somehow gather that many working drives together in one place.

Wafertapes come in a cartridge measuring 68x40x5 mm, about as big as the miniature cassettes sold these days for small tape recorders. The top is clear plastic and the bottom is black plastic. T.I. calls these cartridges "wafers". They fit easily and snugly into the slot in the front of the Wafertape Drive. Inside the wafer a dark colored magnetic tape 1.7mm wide (very thin) is wound in a continuous loop, in the same manner as the tape of an "8 track" music cartridge. The tape moves only in one direction and its ends are attached to each other with a piece of reflective silver tape. T.I.'s last 99/4A catalog lists 50 foot ($7.95), 25 foot ($6.95), and 10 foot ($5.95) wafertapes. Five foot wafertapes are also mentioned in the Wafertape User's Guide. Long tapes store more, but it takes longer to find the beginning of a specific file. Official T.I. wafers have a little sliding panel that covers the exposed part of the tape at the edge of the wafer. This slides open as the wafer is inserted into the slot on the Wafertape Drive. I have one such T.I. wafer and I also have some wafers without a T.I. label and without the sliding panel. I wonder who made these "generic" wafers? I do know that other wafertape drives were planned or actually sold for other devices. A wafertape drive for the Tandy 100 laptop computer is described in the March 1984 issue of Creative Computing. Also, see the comments below from Tony McGovern about another wafertape device. Maybe my "generic" wafertapes were not made specifically for the TI Wafertape Drive.

The method of write protecting wafers is unusual. When I first got my Wafertape Drive I thought it was defective because I always got a "write protected" error message whenever I tried to initialize a wafer. I could find nothing resembling a write protect tab on my wafers, and looking inside the Wafertape Drive slot revealed no evidence of a mechanical pin associated with the "remove the tab and it is write protected" system of protection typically found on most audio and video tape cartridges. It turns out that you have to put a "write enable" paper sticker at a specific location on the top of the wafer in order to write to the wafer. Anything white will do. This "write enable" piece of paper is OPTICALLY sensed from above by the Wafertape Drive. In the absence of the sticker, the optical sensor sees through the transparent wafer upper surface and does not get a reflection off of the black wafer bottom. A second optical sensor detects the silver end/beginning of tape marker. This marker is the only reference point the wafertape drive has to tell the relative position of everything else on the tape. On "8 track" continuous loop music cartridges, the end/beginning of tape marker is detected electronically. This silver marker on a wafertape is detected through a

A Historical Note - (cont)

window in the center of the wafer's transparent upper surface. If the "write enable" sticker you are using is too big and covers this window, then the wafer will not be usable as I finally figured out after lots of frustration.

When you initialize a wafer with the FORMAT command of the CC40, the tape is advanced until the marker is optically detected, and then the Wafertape Drive prepares a new tape directory area at the beginning of the tape. The entire tape is not magnetically encoded with FORMAT, just the directory area. Software designed to directly read the contents of a wafertape directory reveals that if a previously used wafer is reFORMATted, the old file names are retained in the directory area, but all file lengths are set to zero. The time required for a FORMAT depends on how close the tape is to its beginning when FORMAT begins. Even a 50 foot tape formats very rapidly if it is already almost at its beginning.

A directory has room for 16 files irrespective of tape length. Of course short tapes may not have enough room for 16 files if they are of significant length. Files are written to the tape sequentially, with the directory keeping track of the file name, number (0-15), and length. Apparently the Wafertape Drive locates specific files by file number, counting the End-Of-File indicators that pass by the read/write head as the tape is advanced to the start of the desired file. Only the last file can be overwritten by another file of the same name. If you write a file with the same name as a file already on the wafer that is not the last file on the wafer, the old file's directory reference is deleted and the new file of the same name is written to fresh space after all the other files currently on the tape. (I hope you understood that.) File types supported include PROGRAM, and INTERNAL or DISPLAY data files. RELATIVE files are not supported, and you can't open data files as APPEND. Although this is not made clear in available documentation, I think that when data files are read from wafertape, the entire file is read into computer memory for manipulation by the controlling program and then later if necessary written back to wafertape. This limits the size of data files. Only one wafertape file can be opened at a time.

To compare the speeds of the Quickdisk and Wafertape drives, I timed the SAVE and OLD of a 15300 byte text file from Memo Processor using a newly FORMATted wafer and disk. Wafertape: SAVE, 4 min 25 sec; OLD; 3 min, 10 sec. Quickdisk: SAVE, 2 min 15 sec; OLD, only 38 seconds.

Why wasn't the Wafertape Digital Tape Drive ever officially released? It just did not meet T.I.'s standards for reliability. It does not work well on battery power. Even with four newly installed, fresh alkaline AA batteries, you almost always get an I/O error 25 (low batteries in peripheral) when you try to load something, and you often get the same error when you try to SAVE while on battery power. The Wafertape Drive only works with much reliability when operated with the AC adapter. Apparently the speed at which the tape crosses the Wafertape Drive's read/write head is critical, and variations in this speed are not tolerated. With any battery, continuous power drain results in a voltage decrease compared to the initial voltage put out by the battery. Such a voltage decrease slows down the drive motor. Also, as the Wafertape Drive operates it turns itself on and off several times as it loads or stores data. Starting an electric motor requires an immediate surge of extra current compared to the current needed to keep the motor operating at constant speed once it has started. It is possible that the Wafertape Drive's AA batteries are not able to maintain constant voltage with all the required on/off cycles. In addition to the battery problem, it is sometimes possible to write data beyond the end of the wafertape and wipe out the directory on the other side of the reflective end/beginning of tape marker. This renders all files on the tape useless. I have managed to overwrite the end of a wafertape on two occasions. At other times, when I deliberately tried to do this, the CC40 would not let me write past the end of a wafertape. I sometimes get I/O error 6 (device error, try again) with the Wafertape drive for no reason I can determine. Sometimes trying again doesn't work. I never get these error messages when using my Quickdisk drive. Finally, I suspect that wafertapes are not as durable as disks. The tape is very tiny and is subject to a lot of physical movement and twisting as it moves within the wafer. I suspect that with time the tape may break. I know for example that I have had a higher percentage of my "8 track" music tapes break compared to my reel to reel cassette music tapes. Let me quote from a letter received recently from Tony McGovern of Australia, senior author of FUNNELWEB. "Wafer-tapes were always a disaster area! I think they appeared in one of Sinclair's UK machines. The other place they appeared, also in the UK, was in an abomination produced by ICL sold here, a computer phone -a combination of low end PC with modem/phone all built in- but no disk drive, only the wretched unreliable wafertape. Telecom Australia probably has a warehouse full of these things that they would rather not be reminded of. They had a great marketing campaign to sell these several years ago and no one wanted the turkeys."

I do use my Waertape drive. It isn't that unreliable. I have a little briefcase in which I can keep my CC40, the Wafertape Drive, my HexBus printer/plotter, a power strip, and all the necessary AC adapters all pluged in and ready to go. I can open the briefcase and plug in the power strip and use the peripherals as they sit in the briefcase. I can also use the printer/plotter and CC40 while still in the briefcase using battery power, but I have given up trying to use the Wafertape Drive with battery power. I keep my HexBus RS232 and Quickdisk drive pluged in (these two devices both REQUIRE AC power) next to my 99/4A. If possible, anything saved to wafertape ALSO eventually gets save to disk with the Quickdisk drive, which I find to be quite reliable.

RELIABILITY TEST ANALYSIS - MORE ON TRANSLATION OF TI-59 PROGRAMS FOR THE TI-95

Thomas Stewart asked for help in translating a TI-59 reliability test analysis program from the January 1981 issue of Quality for use with the TI-95. We had difficulty in reading the listing in the magazine. Thomas contacted the author R. C. Cornwell and received a listing in the standard TI-59 format. That listing appears on page 14 together with some comments which will facilitate the translation. Mr. Cornwell's description of the program states:

"Solves the defining equation for the chi-square probability distribution, permitting the calculation of lower one sided confidence limits on MTBF, the calculation of the probability of any given number of failures, and the number of failures that will fail a test to demonstrate an MTBF at a specified confidence level."

User instructions for the TI-59:

A. Enter the program. Note: When the program is first entered, press 2nd B' to calculate and store factorials 1 through 59 in data registers 11 through 69. Restore the standard partitioning (6 Op 17) and save all four banks on magnetic cards.

B. Press A to initialize. See a zero in the display.

C. Data entry (enter zero for the unknown)
1. Enter the test time (T) and press R/S. See a zero in the display.
2. Enter the number of failures (r) and press R/S. See a zero in the display.
3. Enter the confidence level (1-) and press R/S. See a zero in the display.
4. Enter the MTBF or failure rate ( ) and press R/S. See a zero in the display.

D. Solve for the unknown:
1. Press B to return the MTBF given T, 1- , and r.
2. Press C to return the number of failures given T, 1- , and MTBF.
3. Press D to return the number of spares given T, 1- , and .
4. Press E to return the probability of r or less failures given T, r, and MTBF.

For a sample problem consider the case where T = 668, r = 3,and 1- = 0.9 . The TI-59 obtains the MTBF as 9.9988488 01 in about 39 seconds. The TI-95 obtains the MTBF as 9.998849 01 in about 5 seconds.

Comments on the translation for the TI-95 on page 16:

1. Steps 0000 through 0036 emulate the User Defined Keys of the TI-59 with a function key menu on the TI-95. The concept was discussed in V12N4P16.

2. Ease of translation is attained by conversion from the absolute addressing of the TI-59 program for label addressing on the TI-95. The added notations on TI-59 listing on page 15 illustrates one technique. One simply goes through the program, identifies an absolute address call (as in steps 016 through 018), replaces the absolute address with a GTL with an arbitrarily selected label, and inserts the corresponding label at the appropriate point (at step 304 for the example).

3. The A' subroutine calls of the TI-59 program at steps 116, 246 and 302 must be replaced with SBL calls on the TI-95. This translation replaces A' with AB.

4. T register tests are translated using system register 2079 as explained in V13N2P8-10 and V13N3P2.

(continued on page 16)

```
GTL GG { 000  61  GTO
         001  00  00
         002  76  76
LBL AB   003  76  LBL
         004  16  A'
         005  86  STF
         006  02  02
         007  00  0
         008  42  STO
         009  09  09
         010  42  STO
         011  06  06
         012  43  RCL
         013  03  03
         014  42  STO
         015  00  00
GTL AC   016  67  EQ
         017  03  03
         018  04  04
LBL AE → 019  43  RCL
         020  08  08
         021  45  Y^x
         022  43  RCL
         023  00  00
         024  55  ÷
         025  53  (
         026  43  RCL
         027  00  00
x!     { 028  85  +
         029  01  1
         030  00  0
         031  54  )
         032  42  STO
         033  04  04
         034  73  RC*
         035  04  04
         036  95  =
         037  44  SUM
         038  06  06
         039  87  IFF
         040  02  02
GTL AD   041  03  03
         042  10  10
LBL AN → 043  97  DSZ
         044  00  00
         045  00  00
GTL AE   046  19  19
         047  01  1
LBL AM → 048  44  SUM
         049  06  06
         050  43  RCL
         051  08  08
         052  94  +/-
         053  22  INV
         054  23  LNX
         055  49  PRD
         056  09  09
         057  65  ×
         058  43  RCL
         059  06  06
         060  95  =
         061  42  STO
         062  07  07
         063  92  RTN
LBL AA   064  76  LBL
         065  11  A
         066  25  CLR
         067  01  1
         068  69  OP
         069  17  17
         070  47  CMS
         071  07  7
         072  69  OP
         073  17  17
         074  00  0
         075  81  RST
LBL GG → 076  86  STF
         077  08  08
         078  91  R/S
         079  48  EXC
         080  02  02
         081  91  R/S
         082  48  EXC
         083  03  03
         084  91  R/S
         085  94  +/-
         086  85  +
         087  01  1
         088  95  =
         089  48  EXC
```

```
         090  05  05
         091  91  R/S
LBL BB   092  76  LBL
         093  12  B
         094  43  RCL
         095  03  03
         096  65  ×
         097  53  (
         098  02  2
         099  93  .
         100  05  5
         101  75  -
         102  43  RCL
         103  05  05
         104  54  )
         105  55  ÷
         106  02  2
         107  93  .
         108  00  0
         109  03  3
         110  85  +
         111  93  .
         112  07  7
         113  95  =
         114  42  STO
         115  08  08
LBL AH → 116  16  A'
SBL AB   117  55  ÷
         118  43  RCL
         119  05  05
         120  75  -
         121  01  1
         122  95  =
         123  50  I×I
         124  75  -
         125  01  1
         126  52  EE
         127  94  +/-
         128  03  3
         129  95  =
         130  22  INV
         131  77  GE
GTL AG   132  03  03
         133  18  18
         134  43  RCL
         135  07  07
         136  75  -
         137  43  RCL
         138  05  05
         139  95  =
         140  55  ÷
         141  43  RCL
         142  09  09
         143  95  =
         144  44  SUM
         145  08  08
         146  43  RCL
         147  08  08
GTL AH { 148  61  GTO
         149  01  01
         150  16  16
LBL CC   151  76  LBL
         152  13  C
         153  35  1/X
GTL AI { 154  61  GTO
         155  01  01
         156  74  74
LBL DD   157  76  LBL
         158  14  D
         159  87  IFF
         160  00  00
GTL AI   161  01  01
         162  74  74
         163  32  X:T
         164  01  1
         165  48  EXC
         166  05  05
         167  94  +/-
         168  44  SUM
         169  05  05
         170  86  STF
         171  00  00
         172  00  0
         173  32  X:T
LBL AI → 174  65  ×
         175  43  RCL
         176  02  02
         177  95  =
         178  42  STO
         179  08  08
```

```
         180  55  ÷
         181  01  1
         182  93  .
         183  06  6
         184  55  ÷
         185  53  (
         186  01  1
         187  93  .
         188  02  2
         189  75  -
         190  43  RCL
         191  05  05
         192  54  )
         193  65  ×
         194  53  (
         195  24  CE
         196  55  ÷
         197  05  5
         198  54  )
         199  45  Y^x
         200  53  (
         201  93  .
         202  03  3
         203  04  4
         204  75  -
         205  43  RCL
         206  05  05
         207  55  ÷
         208  02  2
         209  54  )
         210  65  ×
         211  53  (
         212  01  1
         213  75  -
         214  01  1
         215  55  ÷
         216  09  9
         217  55  ÷
         218  43  RCL
         219  05  05
         220  22  INV
         221  45  Y^x
         222  04  4
         223  54  )
         224  65  ×
         225  53  (
         226  01  1
         227  75  -
         228  93  .
         229  02  2
         230  02  2
         231  65  ×
         232  43  RCL
         233  05  05
         234  33  X²
         235  54  )
         236  65  ×
         237  01  1
         238  93  .
         239  01  1
         240  03  3
         241  05  5
         242  95  =
         243  59  INT
         244  42  STO
         245  03  03
LBL AL → 246  16  A'
SBL AB   247  75  -
         248  43  RCL
         249  05  05
         250  95  =
         251  77  GE
GTL AJ   252  02  02
         253  68  68
         254  01  1
         255  44  SUM
         256  00  00
         257  43  RCL
         258  03  03
         259  87  IFF
         260  03  03
GTL AK   261  02  02
         262  86  86
         263  86  STF
         264  01  01
GTL AL { 265  61  GTO
         266  02  02
         267  46  46
LBL AJ → 268  43  RCL
         269  03  03
```

```
         270  87  IFF
         271  01  01
GTL AK   272  02  02
         273  86  86
         274  67  EQ
GTL AK   275  02  02
         276  86  86
         277  01  1
         278  22  INV
         279  44  SUM
         280  03  03
         281  86  STF
         282  03  03
         283  61  GTO
GTL AL   284  02  02
         285  46  46
       → 286  22  INV
LBL AK   287  86  STF
         288  01  01
         289  22  INV
         290  86  STF
         291  02  02
         292  91  R/S
LBL EE   293  76  LBL
         294  15  E
         295  35  1/X
         296  65  ×
         297  43  RCL
         298  02  02
         299  95  =
         300  42  STO
         301  08  08
SBL AB   302  16  A'
         303  91  R/S
LBL AC → 304  01  1
         305  42  STO
         306  09  09
GTL AM { 307  61  GTO
         308  00  00
         309  48  48
LBL AD → 310  42  STO
         311  09  09
         312  22  INV
         313  86  STF
         314  02  02
GTL AN { 315  61  GTO
         316  00  00
         317  43  43
LBL AG → 318  43  RCL
         319  02  02
         320  55  ÷
         321  43  RCL
         322  08  08
         323  95  =
         324  91  R/S
         325  76  LBL
         326  17  B'
         327  86  STF
         328  08  08
         329  07  7
         330  69  OP
         331  17  17
         332  01  1
         333  01  1
         334  42  STO
         335  10  10
         336  01  1
         337  42  STO
         338  69  69
LBL FF → 339  72  ST*
         340  10  10
         341  65  ×
         342  01  1
         343  44  SUM
         344  10  10
         345  44  SUM
         346  69  69
         347  43  RCL
         348  69  69
         349  95  =
GTL FF { 350  61  GTO
         351  03  03
         352  39  39
         353  00  0
         354  00  0
         355  00  0
         356  00  0
         357  00  0
         358  00  0
         359  00  0
```

Reliability Test Analysis - (cont)

5. The Lbl B' routine of the TI-59 (steps 325 through 352) calculates factorials 1 through 59 and stores the values in data registers 11 through 69. The values are recalled as needed by steps 025 through 035 of the program. The translation uses the built-in factorial function of the TI-95 in place of steps 028 through 035. This also eliminates the need to translate the Lbl B' routine of the TI-59 program.

6. The Lbl A (initialize) routine of the TI-59 program clears data registers 00 through 09 with a 1 Op 17 Cms sequence. The TI-95 translation uses a Dsz loop to clear those registers (steps 0128 through 0146).

7. The initialize routine of the TI-59 clears the system flags with the RST at step 075 and later sets flag 8 so that program execution is suspended when an error occurs (see page V-67 of *Personal Programming*). The translation clears the system flags with the CFG command and sets flag 15 so that program execution is suspended when an error occurs (see page 5-13 of the *TI-95 Programming Guide*). Steps 000 through 002 of the TI-59 program are not needed in the translation.

Why do we need to stop program execution when an error occurs? One case would be a calculation which asks for the factorial of a value higher than 69! which results in an overflow.

The user instructions for the TI-95 are the same as those in steps B through D of the TI-59 instructions except that the user presses GO (F1) instead of R/S in steps C.1. through C.4.

TI-95 Program Listing

```
0000 CLR DFN F1: A @AA
0008 DFN F2: B @BB
0015 DFN F3: C @CC
0022 DFN F4: D @DD
0029 DFN F5: E @EE HLT
0037 LBL AB SF 02 0
0043 STO 009 STO 006
0049 RCL 003 STO 000
0055 IF= 2079 GTL AC
0061 LBL AE RCL 008 y^x
0068 RCL 000 /( RCL 000
0076 x! = ST+ 006 TF 02
0083 GTL AD
0086 LBL AN DSZ 000
0092 GTL AE 1
0096 LBL AM ST+ 006
0102 RCL 008 +/- INV LN
0108 ST* 009 * RCL 006 =
0116 STO 007 RTN
0120 LBL AA CLR x~t CFG
0126 SF 15 10 STO 000
0133 LBL AF 0
0137 STO IND 000 DSZ 000
0144 GTL AF BRK EXC 002
0151 BRK EXC 003 BRK +/-
0157 +1= EXC 005 HLT
0164 LBL BB RCL 003 *(2.
0174 5- RCL 005 )/2.03+.
0187 7= STO 008
0192 LBL AH SBL AB /
0199 RCL 005 -1= ABS -1
0208 EE +/- 3= IF< 2079
0215 GTL AG RCL 007 -
0222 RCL 005 =/ RCL 009
0230 = ST+ 008 RCL 008
0237 GTL AH
0240 LBL CC 1/x GTL AI
0247 LBL DD TF 00 GTL AI
0255 x~t 1 EXC 005 +/-
0261 ST+ 005 SF 00 0 x~t
0268 LBL AI * RCL 002 =
0276 STO 008 /1.6/(1.2-
0289 RCL 005 )*( CE /5)
0299 y^x (.34- RCL 005 /
0309 2)*(1-1/9/ RCL 005
0322 INV y^x 4)*(1-.22*
0334 RCL 005 x^2 )*1.135
0345 = INT STO 003
0350 LBL AL SBL AB -
0357 RCL 005 = INV
0362 IF< 2079 GTL AJ 1
0369 ST+ 003 RCL 003
0375 TF 03 GTL AK SF 01
0382 GTL AL
0385 LBL AJ RCL 003
0391 TF 01 GTL AK
0396 IF= 2079 GTL AK 1
0403 ST- 003 SF 03
0408 GTL AL
0411 LBL AK RF 01 RF 03
0418 HLT
0419 LBL EE 1/x *
0424 RCL 002 = STO 008
0431 SBL AB HLT
0435 LBL AC 1 STO 009
0442 GTL AM
0445 LBL AD STO 009
0451 RF 02 GTL AN
0456 LBL AG RCL 002 /
0463 RCL 008 = HLT
```

A MINT CONDITION TI-59 - In late 1990 Jorge Valencia wrote: "Please note that the following store, Empire Shops, Inc., 318 Lincoln Road, Miami Beach, FL 33139 has a brand-new TI-59 for sale. Not having my card library or PC-100C with me I could not run our diagnostics on it, but the calculator passes the Diagnostic Card that comes with it and otherwise seems as good as new. The man initially tagged it a $149 ... " Empire Shops telephone number is (305)-532-5053.

<u>MORE</u> <u>ON</u> <u>PI</u> <u>FINDING</u> - Michael Ecker's column "Easy Pieces" in the January 1991 issue of *Algorithm* notes that the ratio of the area of circle inscribed in a square is pi/4. He then continues on:

> "... This means that if we threw a random dart at the square, the probability that it would land inside the circle is exactly pi/4. Suppose we reverse this conclusion and treat pi itself as the unknown. We an use a computer program to simulate thousands of dart-tosses in a very short time. The program will record the number of hits; i.e., tosses that land in the circle. Dividing this by the total number of tosses and multiplying by 4 should produce an approximation of pi. The method is called Monte Carlo simulation because each toss uses a random number, like rolling a die. ..."

Ecker's column also provides an algorithm which:

1. Allows the user to select the number of tosses, and then for each toss,

2. Picks two random numbers x and y, each of which are between 0 and 1,

3. Calculates the value (x-.5)^2 + (y-.5)^2 which is the square of the distance from the common center of the square and circle to the impact point,

4. Scores a hit if the value is less than or equal to 0.25.

5. Calculates pi as 4 times the number of hits divided by the number of tosses.

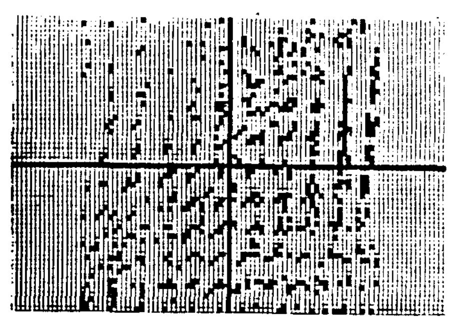
For graphing calculators such as the fx-7000G or the TI-81 the user can also display the impacts on the screen as they occur. Three programs which implement the Ecker algorithm and plot the impacts on a 60 x 60 grid appear on the next page. The left hand column is a BASIC program for the Radio Shack Model 100 which plots the tosses on a square in the center of the screen. The center column is an fx-7000G program. The right hand column is a TI-81 program.

Now, if the random number algorithm of a given machine yields uniformly distributed numbers then the screen should approach being completely filled as the number of tosses is increased. That is the case for the Model 100 and the TI-81. For example:

* For 24,000 tosses on the Model 100 only 19 of the 3600 points on the 60 x 60 grid are missed and pi is calculated as 3.137333...

* For 24,000 tosses on the TI-81 only 35 of the 3600 points on the 60 x 60 grid are missed and pi is calculated as 3.1351666...

However, for 24,000 tosses on the fx-7000G less than half of the points on the 60 x 60 grid have been hit and pi is calculated as 3.278833... Even when 100,000 tosses are used the screen is less than half full as shown in the figure at the right, and pi is calculated as 3.44812 . This seems to suggest even greater problems with the random number generator of the fx-7000G than previously reported in V11N1P8/9.



Old-timers will remember that we addressed this method of pi-finding problem in V10N1P14-15 in response to a discussion in A. K. Dewdney's column "Computer Recreations" in the April 1985 issue of *Scientific American*. The algorithm in Dewdney's column and the programs on V10N1P14 recognized that a solution for pi could be obtained by comparing x^2 + y^2 to unity.

More on Pi Finding - (cont)

Model 100

```
10 INPUT "N=";N
20 CLS
30 LINE (100,32)-(163,32)
40 LINE (132,0)-(132,63)
50 M = 0
60 M1=0
100 FOR I = 1 TO N
110 X = RND(1)
120 Y = RND(2)
123 IF X = 1 THEN PRINT "X=1":STOP
127 IF Y = 1 THEN PRINT "Y=1":STOP
130 R2 = (X-.5)^2+(Y-.5)^2
140 IF R2<=0.25 THEN M=M+1
150 IF R2=0.25 THEN M1=M1+1
160 PSET (100+63*X,63*Y)
170 NEXT I
200 PRINT 4*M/N
220 PRINT "M1=";M1
230 END
```

fx-7000G

```
Range -47,47,50,
-31,31,35
"N="?→N
Cls:0→M:N→L
Lbl 0:Ran#→X:Ran
#→Y
Pol(X-.5,Y-.5)<.
5⇒Isz M
Plot -31+63X,-31
+63Y
Dsz L:Goto 0
Graph Y=0
"π=":4M÷N ◢
```

TI-81

```
Prgm1:PI
:ClrDraw
:-47→Xmin
:47→Xmax
:50→Xscl
:-31→Ymin
:31→Ymax
:35→Yscl
:Disp "N ="
:Input N
:0→M
:0→I
:Lbl A
:Rand→X
:Rand→Y
:If R▶P(X-.5,Y-.
5)<.5
:M+1→M
:PT-On(-31+63X,-
31+63Y)
:IS>(I,N)
:Goto A
:4M/N→P
:Disp "  ="
:Disp P
:End
```

131 bytes

The three programs provide examples of conversion of programs between BASIC, the fx-7000G and the TI-81. Specific comments on the programs follow:

1. The Model 100 program calculates the square of the distance from the common center to the impact point and compares the result to the value 0.25 . The fx-7000G and TI-81 programs use the rectangular to polar function to obtain the distance from the common center to the impact point and compare the result to 0.5 .

2. The fx-7000G program appears much shorter, but actually isn't. For example, the fx-7000G only two lines to set the range variables while the TI-81 requires six lines; however, the fx-7000G uses 19 bytes to set the range variables and the TI-81 uses 26 bytes. In addition the fx-7000G permits several statements on one line while the TI-81 allows only one statement per line.

3. The M1 variable in the Model 100 is used to check for occurrence of the unlikely event that the square of the distance from the common center to the impact point is exactly equal to 0.25. No such cases were found during many runs with the program.

4. In his column "Computer Adventures" in the September 1990 issue of *Algorithm* A. K. Dewdney stated "... I have yet to obtain a 1 from any random number generator." Lines 123 and 127 of the Model 100 program would stop the program if a random number were exactly one. No stops occurred during our testing.
