TI PPC NOTES

NEWSLETTER OF THE TI PERSONAL PROGRAMMABLE CALCULATOR CLUB

P.O. Box 1421, Largo, FL 34649

Volume 12, Number 3 Third Quarter 1987

The "star" article in this issue is Maurice Swinnen's description of how to build a cable which will allow a TI-74 to be connected to the CC-40 peripherals. I have five sixteen inch long cables with a hex bus connector on one end which are available to members who want to build one of the interface cables. Send a dollar to cover shipping. First come, first served.

The emphasis in the remainder of the issue is on access to programs as opposed to the publication of programs in the newsletter. In pages 2 through 12 you will find information on the availability of fifty TI-95 programs from Hewlett Ladd, on the availability of seventy BASIC programs in books coauthored by former editor Maurice Swinnen, and on access to 140 additional PPX programs held by other members.

One of the questions frequently asked is "Should I buy a TI-74 or a TI-95? The answer probably depends upon the program library that you already have. If you have a large library of tried and true TI-59 programs then you will surely find it easier to convert to the TI-95. If instead, your primary experience is in BASIC then you will probably be happier with a TI-74. There is another aspect which may be of importance. The TI-95 memory mechanization makes it very easy to store a large number of programs either in user memory or in an 8K Constant Memory cartridge, and an efficient menu mechanization makes the programs easy to run. In contrast, the TI-74 memory mechanization permits only one program in user memory, and only one program in an 8K Constant Memory cartridge. The latest issue of Programmable Calculator News suggests the use of subprograms, but it is not so easy to combine independently written programs if dimension statements are involved.

TABLE OF CONTENTS

CCL-144 cleaning strips for the card reader of the TI-59 are no longer available.

Magnetic card service will continue to be available for TI-59 programs published in TI PPC Notes. One dollar per card and a SASE, please.

ERRATA

Supplemental Book Review - V12N2P23 noted that the book Engineering Statistics with a Programmable Calculator by William Volk had been favorably reviewed by W. J. Widmer in V7N7/8P15, but that some recent experience had shown that both the programs and the examples were suspect. The book received a second review in V9N6P3 where George Booth reported that the book contained 25 programs which were very useful. The book is listed on page 70 of the Educalc Catalog #38 as stock number E-110. The price for the hardbound book is now $31.95.

Aitken's Δ^2 Convergence Method - Hewlett Ladd writes: "It seems that Jorge Valencia's program in V12N2P13 will not find a root for every guess of a solution. Using Messer's benchmark equation from V11N4P16, as rearranged, it will indeed find the real root with guesses of -1, 0, 0.1, 0.9, 0.999, 0.99999, 1.2, 1.5, 1.9, and 2, 3, 4, 5, and 6, and perhaps with lower or higher guesses. However, it balks with a guess of 1, and with guesses very close to 1 such as 0.999999, 1.1, etc. I never got a convergence with those values. It seems to oscillate or else takes an unconscionable length of time to converge. Yet the root was quickly found with a guess of 1 using my versions of MU-15 and ML-08."

Editor's Note: If you modify the program on V12N2P13 by changing the sequences at steps 004-006 and 017-019 from SBR 072 to SBR 073, and insert a PRT command after the LBL E sequence at steps 069-070 you can monitor the convergence of the program. With a guess of zero the solution converges in four iterations. With a guess of one the "solution" settles at -13.188 after 19 iterations. Similarly, with a guess of 1.1 the "solution" settles at -12.685 after 22 iterations, and with a guess of 0.999999 the "solution" settles at -13.551 after 11 iterations. I have forwarded these results to both Hewlett and Jorge.

ANYONE NEED A BP-8? - I inherited this battery pack from a friend who didn't remember why he had it. It is the battery pack which is used with the old SR-40, TI-30 and Business Analyst I. Replacement units are currently listed in the Elek-Tek catalog for $8.50 . You can have this unit for two dollars to cover shipping.

PROGRAMS FOR THE TI-95 - Hewlett Ladd has been by far the most active in converting programs from the TI-59 for the TI-95, and in writing new programs for the TI-95. Page 3 lists 50 of his programs. Hewlett has agreed to provide listings of these programs for one dollar per program plus a stamped and self- addressed envelope. Send cash or stamps; no small value personal checks, please. Hewlett is also willing to transfer programs to a cartridge. Write to make appropriate arrangements. His address is:

Hewlett F. Ladd
12 Ridgewood Avenue
Rowayton CT 06853

Programs for the TI-95 - (cont)

NAME	BYTES	DESCRIPTION
CAL	744	Calendar; Gregorian, Julian, JD #, Intervals
SOL	848	Solar: coordinates and sidereal time
SOT	240	Solar: transit time
SOR	408	Solar: sight reduction (LHA, h & azimuth)
LUN	936	Lunar: coordinates
LU1	1200	Lunar subroutines
LU2	1168	Lunar subroutines
LUT	280	Lunar: transit time
PHS	768	Lunar: new/full phases (dates & times)
SSN	384	Seasons: equinoxes & soltices (dates & times)
EST	392	Easter date: Julian and Gregorian calendars
LUR	200	Lunar: sight reduction (LHA, h and azimuth)
SRS	464	Solar: rise/set (UT & azimuth of local r/s)
CM1	784	Complex numbers: arithmetic (TI-59 ML-04)
CM2	552	Complex numbers: functions (TI-59 ML-05)
CM3	736	Complex numbers: trigonometry (TI-59 ML-06)
CQD	672	Complex quadratic (adapted from Hawes)
SQ1	400	Extended precision SQR (V10N4P23, 400 digits)
SQR	144	Mini SQR (20 digits)
e	160	Many digits of e (V11N4P4)
N!X	139	Exact factorial to 461! (V11N4P19)
N!>	88	Large factorials (Stirling's estimate)
DIV	568	Extended precision divide (V11N4P25)
MLT	272	Extended precision multiply (V11N4P21)
PRF	104	Prime factoring (but see SFF below)
LCM	72	LCM and GCD
DFR	312	Decimal to fraction conversion (V7N4/5P10)
exp	136	Large exponents (y^x and e^x)
CMB	128	Combinatorials (nPr, nCr, n!)
ODS	208	Odds against (at least K in N trials)
QRT	656	Quartic equations (V3N12P3 version)
LNQ	424	Linear equations, 16 unknowns (V11N4P8)
LOD	120	Data loader
LST	144	13 digit list
SRT	240	Shell sorter (V10N3P10)
R#s	216	Random numbers with sorting (calls SRT)
ANN	1176	Annuities & loans (TI-59 ML-19)
IRR	312	Internal rate of return (TI-59 RE-12)
CDS	96	Bank C/D's: Interest for n days; calls CAL
mlt	360	Extended precison multiply (faster, V12N1P16)
ZoX	464	Zeroes of functions (TI-59 ML-08, bisection)
ZoF	328	Root finder (TI-59 MU-15, Newton-Raphson)
SFF	360	Speedy factoring 13 digits (V8N5P23)
LON	544	Loan schedule & annual recap (V11N3P11)
LAG	272	LaGrange's (n-1) polynomial (V11N2P11)
CPR	1016	Calendar printer (from 1 AD onward)
PiX	296	Pi to 1573 digits (from V8N4P26)
ML2	952	Matrices, determinants & simultaneous equations, 10 x 10 max, (TI-59 ML-02)
IM2	632	Matrix inverses (TI-59 ML-02)
SDL	768	Vertical wall sundial (coordinates, etc.)

BOOK REVIEWS:

STATISTICS LIBRARY. Application Software for the Sharp EL-5500 and PC-1403 Scientific Computers. Maurice E. T. Swinnen and David Thomas. Systems Publications, Box 300488, Arlington, TX 76010. 1987. 105 pages. Paperback. $11.95.

The book contains 23 BASIC language programs for the solution of typical problems in statistics. The programs are:

- Cauchy Distribution Curve Fit
- Circle Best Fit
- Chi-Square Distribution
- Contingency Table
- #*Exponential Curve Fitting
- Gaussian Distribution
- Histogram
- *Hyperbolic Curve Fitting
- #*Logarithmic Curve Fitting
- Mann-Whitney Ranked-Sum Test
- Means and Moments
- Multiple Linear Regression with 2 Variables
- Multiple Linear Regression with 3 Variables
- One-Way ANOVA
- #*Parabolic Curve Fitting
- Poisson Distribution
- #*Power Curve Fitting
- *Reciprocal Curve Fitting
- #*Straight Line Curve Fitting
- Student's t-Distribution
- T-Test for Paired Observations
- T-Test for Unpaired Observations
- Two-Way ANOVA

The documentation for each program includes program listings, step-by-step user instructions, and sample problems. Although the programs were written for the Sharp machines they are easily convertible for use on the TI-74. An example conversion appears on pages 4 and 5 of this issue.

I have started working my way through this book, including a cross check of the results against those from other published programs. I have verified the results within the limits of differences between machines for the programs identified with a # sign in the table above. There are some minor errors in the book:

a. Page 5 states that the Cauchy distribution has "... the rather peculiar characteristic of a mean but no standard deviation". References such as the Mathematical Dictionary by James and James, and Mathematical Methods by Cramer, indicate that the Cauchy distribution may have a mode and a median, but not a mean or a standard deviation since no moments of positive order are finite. William Volk made a similarly incorrect statement on page 76 of the second edition of his book Curve Fitting for Programmable Calculators which was reviewed in V8N2P20.

b. The equation for the logarithmic curve on page 35 should be y = a + b(LN(x)). The equation is correct in the program listings on page 37.

c. The programs which require multiple input such as the curve fitting programs accept the input as string values and convert the strings to numerics internally with VAL commands. This permits termination of the input by entering the letter E. A prompt "End Input by entering E" is provided to remind the user of this feature; however, for seven of the programs the address for an earlier GOTO is improper such that the prompt will not be seen. The programs with that discrepancy have an asterisk in the listing above. In each case the problem can be corrected by changing line 40 to IF PF=0 GOTO 60.

Old-timers will remember Maurice Swinnen as the editor of TI PPC Notes from 1980 through 1982. Maurice has arranged a twenty per cent discount to club members on this book and the two books which follow. Order from the address listed above, and mention TI PPC Notes when you order. There is a $2.00 dollar shipping and handling charge for an order of one book, or a $3.00 charge for two or more books.

Book Reviews - (cont)

ELECTRICAL ENGINEERING LIBRARY. Application Software for the Sharp EL-5500 and PC-1403 Scientific Computers. Maurice E. T. Swinnen and David Thomas. Systems Publications, Box 300488, Arlington, TX 76010. 1987. 119 pages. Paperback. $11.95.

This second book in the series of libraries of BASIC programs is available now. The included programs are:

Active Band-Pass Filter
Active High-Pass Filter
Active Low-Pass Filter
Biomedical Filtering Circuit
Bit-Error Probability
Bode/Nyquist Calculation
Gate-Bias FET Circuit
Logarithmic Conversions
Low Frequency Transistor Amplifier
Odd Resistance Value Synthesizer
Passive Filter Design
Phase Locked Loop Design
Power Supply Filter Design
Rating Unknown Power Transformers
Reactance Chart
Self-Bias FET Circuit
Serial-to-Parallel Conversion
Single-Layer Coil Design
T- and PI-Pad Attenuators
Temperature Conversions
Transistor Parameter Conversions
Twin-T Circuit Design
Voltage Feedback FET Circuit
Zener Diode Power Supply

MATHEMATICS LIBRARY. Application Software for the Sharp EL-5500 and PC-1403 Scientific Computers. Maurice E. T. Swinnen and David Thomas. Systems Publications, Box 300488, Arlington, TX 76010. 1988. 120 pages. Paperback. $11.95.

This third book in the series of libraries of BASIC programs will become available in early 1988. The included programs will cover subjects such as

Complex Functions: add, subtract, multiply, divide
Complex Functions: square, square root, reciprocal, log, exponent, polar to rectangular, and rectangular to polar.
Complex Trigonometric Functions: sine, cosine, tangent, arcsin, arccos and arctan.
Complex Functions: Y^X, Y^(1/X) and log to the base x of y.

Differential Equations, Runge-Kutta
Gamma Function
Gauss Quadrature INtegration
3D Coordinate Transformations
Complex Roots of a Complex Number
Derivatives
Multiprecision Division
Polynomial Function Evaluation
Extended Precison Factorials
Fourier Series
Three Point Interpolation
Polynomial Addition and Subtraction
Polynomial Multiplication
Polar to Rectangular
Rectangular to Polar
Quadratic Equations
Coordinate Translation
Rational Fractions
Simultaneous Equations

PPX PROGRAMS WANTED - Walter Bodenmuller wants the following PPX programs:

268032 Combination of Probabilities
268047 List and Plot Data Distribution
298001 Histogram
298082 Percentiles & Percentile Rank
908177 2 to 12 Cell Vertical Bar Graph
908233 Histogram & Bar Graph Plotter
908239 Histogram Plotter
928050 Travelin Man

Write to 8019 24th St. S.E., Calgary, Alberta Canada T2C 0Z4 if you can help.

CONVERSION OF SHARP BASIC TO TI-74 AND CC-40 BASIC - P. Hanson. The books reviewed on pages 4 and 5 include program listings in BASIC; however; the listings contain commands which are unique to Sharp BASIC. Maurice Swinnen has given me permission to reproduce one of the programs from the Statistics Library to illustrate the conversion process. The program selected was that for Power Curve Fitting from pages 73-75. The listing in the left-hand column on the facing page is from page 75 of the book. The listing in the center column is a conversion for the TI-74 and CC-40. The mathematics portions of the program can be used as is by the TI-74 or CC-40. Most of the input/output routines require some change. You should try to understand each change in order to be able to translate other programs. Representative changes are:

In line 40 the GOTO has been changed to THEN to accommodate the differences between the two BASICs. A similar change is required at lines 70, 100, 110, 260, 270, and 290. The THEN address is changed to 60 to permit access to the prompt on ending input of data.

The BEEP 2 commands in lines 60, 70, 260, 280, 290 and 380 are deleted since the TI-74 does not have that capability. CC-40 users should replace BEEP 2 with DISPLAY BEEP.

The PAUSE ALL command inserted at line 210 stops the program when any subsequent sequence completes a line in the display.

The IMAGE command at line 230 provides format control for the PRINT USING commands in lines 240 and 250.

Lines 320-360 are changed to convert the "Use Printer" selection to TI BASIC.

The listing in the right-hand column on page 7 is an enhancement to provide output of residual errors. The changes relative to the program in the center column are:

Line 15 provides the dimension statement to set up the arrays to hold the input data pairs for use in the residual calculations. If you are going to use more than twenty data pairs you should change to dimensions accordingly.

The statement N=0 is added to line 20 to zero the counter of data pairs.

Line 145 increments the counter and stores the input data pairs in the arrays.

Line 255 permits the user to elect output of residuals.

Lines 500-580 are the subroutine which calculates the residuals, and the mean and RMS of the residuals.

Parentheses which were not needed have been deleted in lines 160 through 200.

Prinntouts for the sample problem from the book appears at the right. Note that the mean of the residuals is not very close to zero. This occurs because of the transformation used to linearize the power equation for the least squares calculation. The least squares calculation is in the linearized variables, LN(X) and LN(Y), not in the variables X and Y. The mean of the residuals from the equation LN(Y) = LN(a) + b*LN(X) is -1.8E-14, near zero as expected.

```
Power  y = ax^b

X=  1
Y=  3.2
X=  2
Y=  7.4
X=  3
Y=  12
X=  4
Y=  16.8
X=  5
Y=  22
a=  3.211745293
b=  1.196427406
RR=          .99996805
Corr.RR=     .99995741
X=  3
Y=  11.9558936
```

```
Power  y = ax^b.

X=  1
Y=  3.2
X=  2
Y=  7.4
X=  3
Y=  12
X=  4
Y=  16.8
X=  5
Y=  22
a=  3.211745293
b=  1.196427406
RR=          .99996805
Corr.RR=     .99995741
d1=-.011745293
d2= .0395963302
d3= .044106398
d4=-.067944759
d5=-.0296686814
d ave = -.0051312011
d RMS =  .0427736287
```

Conversion of Sharp BASIC to TI BASIC - (cont)

```
10:"C":WAIT 150:PRINT "
      Power  y=ax^b"
20:T1=0:T8=0:T9=0:U0=0:
   U1=0:U2=0
30:GOSUB 310:USING
40:IF PF=0 GOTO 70
50:PRINT "    Power  y=
   ax^b"
60:BEEP 2:INPUT "End in
   put by entering E";Z
   Z$
70:BEEP 2:INPUT "X=? ";
   XX$:IF XX$="E" GOTO
   160
80:X=VAL (XX$):IF PF=0
   GOTO 100
90:PRINT "X= ";X
100:BEEP 2:INPUT "Y=? ";
   YY$:IF YY$="E" GOTO
   160
110:Y=VAL (YY$):IF PF=0
   GOTO 130
120:PRINT "Y= ";Y
130:T1=T1+1:T8=T8+LN (X)
   :T9=T9+LN (X)^2:U0=U
   0+LN (Y):U1=U1+LN (Y
   )^2
140:U2=U2+(LN (X)*LN (Y)
   )
150:GOTO 70
160:R5=(T1*T9)-(T8*T8)
170:S1=((T9*U0)-(T8*U2))
   /R5:A=EXP (S1)
180:S2=((T1*U2)-(T8*U0))
   /R5:B=S2
190:RR=((S1*U0)+(S2*U2)-
   ((U0*U0)/T1))/(U1-((
   U0*U0)/T1))
200:RS=1-((1-RR)*(T1-1)/
   (T1-2))
210:PRINT "a= ";A
220:PRINT "b= ";B
230:USING "#####.########
   #"
240:PRINT "RR=     ";RR
250:PRINT "Corr.RR=";RS:
   USING
260:BEEP 2:INPUT "Predic
   t Y? Y/N ";Z$:IF Z$=
   "N" OR Z$="n" GOTO 2
   80
270:GOSUB 370:BEEP 2:
   INPUT "Y again? Y/N
   ";Z$:IF Z$="Y" OR Z$
   ="y" GOTO 270
280:BEEP 2:INPUT "Add mo
   re data? Y/N ";Z$:IF
   Z$="Y" OR Z$="y"
   GOTO 70
290:BEEP 2:INPUT "New ca
   lculation? Y/N ";Z$:
   IF Z$="Y" OR Z$="y"
   GOTO 20
300:END
310:REM  ***Printer?***
320:PF=0:WAIT :BEEP 2:
   INPUT "Printer? Y/N
   ";N$
330:IF N$="N" OR N$="n"
   GOTO 350
340:PF=1:PRINT = LPRINT
   :PRINT "------------
   ------------":GOTO 3
   60
350:PRINT = PRINT
360:RETURN
370:REM ** Predict Y **
380:BEEP 2:INPUT "X=? ";
   XX:IF PF=0 GOTO 400
390:PRINT "X= ";XX
400:YY=A*XX^B
410:PRINT "Y= ";YY
420:RETURN
```

```
10 A$="Power  y = ax^b":
PRINT A$:PAUSE 2
20 T1=0:T8=0:T9=0:U0=0:U
1=0:U2=0
30 GOSUB 310
40 IF PF=0 THEN 60
50 PRINT #1,A$:PRINT #1
60 INPUT "End input by e
nterinq E ";ZZ$
70 INPUT "X = ";XX$:IF X
X$="E"OR XX$="e"THEN 160
80 X=VAL(XX$):IF PF=0 TH
EN 100
90 PRINT #1,"X= ";X
100 INPUT "Y=? ";YY$:IF
YY$="E"OR YY$="e"THEN 16
0
110 Y=VAL(YY$):IF PF=0 T
HEN 130
120 PRINT #1,"Y= ";Y
130 T1=T1+1:T8=T8+LN(X):
T9=T9+LN(X)^2:U0=U0+LN(Y
):U1=U1+LN(Y)^2
140 U2=U2+LN(X)*LN(Y)
150 GOTO 70
160 R5=(T1*T9)-(T8*T8)
170 S1=((T9*U0)-(T8*U2))
/R5:A=EXP(S1)
180 S2=((T1*U2)-(T8*U0))
/R5:B=S2
190 RR=((S1*U0)+(S2*U2)-
((U0*U0)/T1))/(U1-((U0*U
0)/T1))
200 RS=1-((1-RR)*(T1-1)/
(T1-2))
210 PAUSE ALL:PRINT #PF,
"a= ";A
220 PRINT #PF,"b= ";B
230 IMAGE #####.########
240 PRINT #PF,"RR=     "
;:PRINT #PF;USING 230;RR
250 PRINT #PF,"Corr.RR="
;:PRINT #PF;USING 230;RS
260 INPUT "Predict Y? Y/
N ";Z$:IF Z$="N"OR Z$="n
"THEN 280
270 GOSUB 370:INPUT "Y a
gain? Y/N ";Z$:IF Z$="Y"
OR Z$="y"THEN 270
280 INPUT "Add more data
? Y/N ";Z$:IF Z$="Y"OR Z
$="y"THEN 70
290 INPUT "New calculati
on? Y/N ";Z$:IF Z$="Y"OR
 Z$="y"THEN 20
300 END
310 REM ***Printer?***
320 IF PF=1 THEN CLOSE #
1:PF=0
330 INPUT "Use Printer?
Y/N ";N$
340 IF N$="Y"OR N$="y"TH
EN PF=1 ELSE 360
350 OPEN #1,"12",OUTPUT
360 PRINT #PF:RETURN
370 REM ** Predict Y **
380 INPUT "X= ";XX
390 PRINT #PF,"X= ";XX
400 YY=A*XX^B
410 PRINT #PF,"Y= ";YY
420 RETURN
```

```
10 A$="Power  y = ax^b":
PRINT A$:PAUSE 2
15 DIM U(20),V(20)
20 T1=0:T8=0:T9=0:U0=0:U
1=0:U2=0:N=0
30 GOSUB 310
40 IF PF=0 THEN 60
50 PRINT #1,A$:PRINT #1
60 INPUT "End input by e
nterinq E ";ZZ$
70 INPUT "X = ";XX$:IF X
X$="E"OR XX$="e"THEN 160
80 X=VAL(XX$):IF PF=0 TH
EN 100
90 PRINT #1,"X= ";X
100 INPUT "Y=? ";YY$:IF
YY$="E"OR YY$="e"THEN 16
0
110 Y=VAL(YY$):IF PF=0 T
HEN 130
120 PRINT #1,"Y= ";Y
130 T1=T1+1:T8=T8+LN(X):
T9=T9+LN(X)^2:U0=U0+LN(Y
):U1=U1+LN(Y)^2
140 U2=U2+LN(X)*LN(Y)
145 N=N+1:U(N)=X:V(N)=Y
150 GOTO 70
160 R5=T1*T9-T8*T8
170 S1=(T9*U0-T8*U2)/R5:
A=EXP(S1)
180 S2=(T1*U2-T8*U0)/R5:
B=S2
190 RR=(S1*U0+S2*U2-U0*U
0/T1)/(U1-U0*U0/T1)
200 RS=1-(1-RR)*(T1-1)/(
T1-2)
210 PAUSE ALL:PRINT #PF,
"a= ";A
220 PRINT #PF,"b= ";B
230 IMAGE #####.########
240 PRINT #PF,"RR=     "
;:PRINT #PF;USING 230;RR
250 PRINT #PF,"Corr.RR="
;:PRINT #PF;USING 230;RS
255 INPUT "Display resid
uals? Y/N ";Z$:IF Z$="Y"
OR Z$="y"THEN GOSUB 500
260 INPUT "Predict Y? Y/
N ";Z$:IF Z$="N"OR Z$="n
"THEN 280
270 GOSUB 370:INPUT "Y a
gain? Y/N ";Z$:IF Z$="Y"
OR Z$="y"THEN 270
280 INPUT "Add more data
? Y/N ";Z$:IF Z$="Y"OR Z
$="y"THEN 70
290 INPUT "New calculati
on? Y/N ";Z$:IF Z$="Y"OR
 Z$="y"THEN 20
300 END
310 REM ***Printer?***
320 IF PF=1 THEN CLOSE #
1:PF=0
330 INPUT "Use Printer?
Y/N ";N$
340 IF N$="Y"OR N$="y"TH
EN PF=1 ELSE 360
350 OPEN #1,"12",OUTPUT
360 PRINT #PF:RETURN
370 REM ** Predict Y **
380 INPUT "X= ";XX
390 PRINT #PF,"X= ";XX
400 YY=A*XX^B
410 PRINT #PF,"Y= ";YY
420 RETURN
500 REM ** Residuals **
510 S1=0:S2=0
520 FOR I=1 TO N
530 D=V(I)-A*U(I)^B:S1=S
1+D:S2=S2+D*D
540 PRINT #PF,"d"&STR$(I
)&"=";D
550 NEXT I
560 PRINT #PF,"d ave = "
;S1/N
570 PRINT #PF,"d RMS = "
;SQR(S2/N)
580 RETURN
```

MORE PPX PROGRAM AVAILABILITY - Earlier issues set up an informal program exchange to provide access to programs which were formerly available from the PPX Exchange. The following list shows 140 additional programs which have been made available by member Morris Karp:

Q 088013 - Buying vs Renting a House
Q 128002 - Constant Reduction Loan
Q 138001 - Auto Finance Contract
Q 148002 - Household Budgeting
1Q 148007 - Checkbook Maintenance
1Q 148013 - Checking Account Management
Q 178003 - Federal Income Tax 1978
1Q 188004 - Call Option Ratio Writer
1Q 188009 - Call OPtion Spreading
1Q 188010 - Screen Stocks: Quality and Quantity
1Q 188011 - Value of Call Option
Q 188012 - Universal Rate of Return
Q 208013 - Data Fit to Eight Curves
Q 208035 - Non-linear Regression
Q 208039 - Logistic Curve Fit
1Q 208040 - Eight Curve Fit
Q 208041 - Multiple Curve Fit
Q 208050 - Automatic Curve Fit (Nine equations)
Q 208051 - Evolutionary Curve Fitting
Q 388001 - Linear Programming
1Q 388003 - PERT/CPM: Operations Research (Formerly 098006)
1Q 388004 - Linear Programming with Mixed Constraints
Q 398096 - Polar Graphing Program
Q 398098 - Spherical, Cylindrical, Rectangular Coordinates
Q 398122 - Addition of Fractions
1Q 408015 - Specific Gravity Computation
1Q 418001 - The Perfect Gas Law
Q 418005 - Dilution
Q 418007 - First Order Chemical Kinetics
Q 418008 - Second Order Chemical Kinetics - Type I
Q 418009 - Second Order Chemical KInetics - Type II
Q 428011 - Guggenheim Method of 1st Order Chemical Kinetics
Q 418037 - Periodic Table of the Elements
Q 418039 - Conversions of Concentration of Solutions
Q 418042 - Temperature Correction for Equilibrium Solution
Q 418048 - Mole, Weight & Volume Conversions for Gases
Q 418049 - Concentration Conversion
Q 418067 - Formula and Composition (Replaces 418006)
Q 418068 - Concentration Exchange
1Q 508001 - Calorie Determined by Weight Program
1Q 618001 - Flash Calculation - L/V Ratio to be Determined
1Q 618002 - Redlich-Kwong Equation
1Q 618003 - Bubble and Dew Point Temperatures
Q 618004 - Hydraulic Loop Calculations (Flow)
Q 618005 - Beattie-Bridgman Equation of State
1Q 618008 - Solution to Pipe Problems (Flow)
1Q 618009 - Binary Distillation by McCable-Thiele Method
1Q 618010 - Steam Tables - Vapor
1Q 618011 - Gas or Steam Piping Flow Pressure Drop
1Q 618012 - Heat Transfer Coefficient/Pressure Drop
1Q 618013 - Incompressible Fluid Pipeline Pressure Drop
1Q 618014 - Heat Exchanger Thermodynamics
1Q 618015 - Heat Exchanger Log Mean Delta Temperature

More PPX Program Availability - (cont)

1Q 618017 - Orifice Calculations for Liquids
Q 618018 - Binary Distillation
Q 618020 - Valve Sizing for Perfect Gas Flow
Q 618021 - Valve Sizing for Liquid Flow
Q 618022 - Valve Sizing for Saturated Steam Flow
Q 618024 - Gas Orifice Calculations (Flange Taps)
1Q 618029 - Fluid Flow One-Phase
Q 618031 - Mean Temperature Difference Exchangers in Series
Q 618032 - Multicomponent Enthalpy Calculations
Q 618034 - Average Particle Size
Q 618035 - Heat of Reaction and Rate Constant
1Q 618036 - Two PHase Flow Resistance
Q 618037 - Relative Humidity/Dew POint
Q 618038 - Pipeline Gas Velocity
Q 618040 - Heat Capacity and Enthalpy
Q 618042 - Slurry (Specific Gravity/Dry Solid Rate)
Q 618044 - Steam Flow Rate throgh Orifice Plate
Q 618047 - Pipeline Pressure Drop - Compressible
1Q 628001 - Trapezoidal Channel Depth and Velocity
1Q 628022 - Hazen-Williams Formula (Pipes)
1Q 628024 - Equivalent Pipe Method
Q 628028 - Pipe Flow Calculations
Q 628038 - Head Loss over a Submerged Weir
Q 628040 - Flow through a Circular Pipe
1Q 628042 - Hydraulic Pressure - Head Loss
Q 628046 - Flow in Trapezoidal, Rectangular or Triangular Channel
Q 628047 - Drainage Area Computations (Conversions)
Q 628058 - Junction Pressure Loss in PIpes
Q 628060 - Backwater in Pipe Conduit
Q 628061 - Channel Flow for Storm Drain and Flood Control
1Q 628071 - Pump Data Analysis (Flow)
1Q 628079 - Pipe Network Calculator
Q 628092 - Sewer Capacity/Design
Q 628093 - Hazen-Williams Formula and Hardy Cross Method
Q 628111 - Gravity Drain Flow
GQ 638001 - ALC/1 Emulator
GQ 638002 - ALU Byte Operations
1Q 638003 - EBCDIC Code Converter
1GQ 638004 - ASCII Code Converter
1GQ 638005 - ASCII and EBCDIC Encoder
1Q 638006 - TI Programmable Simulator
1Q 638007 - TMS 9900 Disassembler
1GQ 638008 - Intel 8080 Disassembler
Q 638009 - Boolean Truth Table
Q 638018 - Boolean Simplification
Q 658111 - Resistor Value Decoder/Encoder
Q 668001 - Pipe Calculations
1Q 668003 - Fan/Blower Ratings
Q 668009 - Psychrometric Calculator (Air Properties)
1Q 668011 - Weymouth Gas Pipeline Pressure Drop
1Q 668013 - Centrifugal Compressor
1Q 668015 - Gas or LIquid Pipe Size
1Q 668017 - Flue Gaqs Enthalpy
1Q 668018 - Steam Flow Through Pipe
1Q 668030 - PV-01 Pressure Vessels
Q 668033 - PV-02 Pressure Vessel Sizing
Q 668041 - PV-03 Pressure Vessels
Q 668042 - PV-06 Pressure Vessels

More PPX Program Availability - (cont)

Q 668049 - PV-04 Pressure Vessels
Q 668053 - PV-07 Pressure Vessels
Q 668056 - Nozzle Calculation
1Q 668061 - Heat Transfer Through Insulated Pipe
Q 668129 - Relief Valve Sizing
1Q 668130 - Safety-Pressure Relief Valve Sizing
Q 668131 - Pressure Relief Valve Sizing
Q 698001 - Perspective Drawing
1Q 698004 - Perspective - "In Depth"
9GQ 698006 - Axonometric Projection
Q 698007 - Isometric Projection
Q 698008 - Oblique Projection
Q 738001 - Heating Load Caclulation
1Q 738002 - Heating/Cooling CFM Distribution
1Q 738006 - Duct Design (Pressure Drop)
Q 758004 - I'll Do It Myself (Programming)
Q 758005 - Basic Language Simulation
Q 868001 - Well Log Interpretation
Q 868002 - Rapid Gas Reserve Estimate
1Q 868003 - Compressibility Factors for Sweet Natural Gases
Q 868004 - Production Schedule for Exponentially Declining Wells
Q 868005 - Gas Well Deliverability
Q 868006 - Oil Reserve Estimate Constant Percentage Decline
Q 868011 - Gas Flow Measurement through an Orifice Meter
Q 868012 - Accumulator Sizing
1Q 908004 - Alpha Printing Clock
1Q 908005 - Flag Tester
1Q 908006 - RPN Simulator
1Q 908012 - Function PLotter for TI-59/PC-100
1Q 908013 - Bar Graph Plotter
1Q 908015 - TI-59 Banner Program
DQ 908018 - Universal PLotter
DQ 908019 - Multiple Plot
Q 908022 - Utility Routines II: Register Manipulations
Q 908023 - Utility Programs I: Indicators
1Q 908024 - Utility Routines III: Display
Q 908025 - Print Conversion Integer Form
1Q 908026 - Conversion Routine: Numerals to Alpha Code
Q 908027 - Store/Sort Utility Program
Q 908031 - Data LIst
Q 908033 - Plotter/Printer
Q 908036 - Extended Data Plotting
DQ 908038 - Functions on Memory
Q 908039 - File Management
KQ 908045 - Large Aalphanumerica
Q 908050 - Index Numbeers - Store and Recall
1Q 908051 - Print Conversion (with Decimal and Sign)
Q 908055 - Op 07 Extended
1DQ 908056 - Utility Routines IV: Flags
DQ 908057 - Utility Routines V: Miscellaneous
DQ 908059 - Utility Routines VII: Register Consolidation
DQ 908060 - Utility Routines VIII: Registers
Q 908062 - Files
1Q 908063 - Hierarchy Register Functions
DQ 908064 - Bar Graph Printer with Labels
Q 908065 - Mial LIst/Message Composer
Q 908066 - Constant Significant Figure Engineering Format
Q 908067 - Right Margin Numerical Printer

More PPX Program Availability - (cont)

Q 908068 - SR-52 Program Listing
1Q 908069 - Alphanumeric Register Listing with Tags
DQ 908082 - TI-59 Memory Structure
DQ 908090 - Multiple Strip Printing (Plotting)
Q 908092 - Print Format Routine
DQ 908095 - Data REgister Manipulations
Q 908097 - ReversePolish Notation using Hierarchy
1Q 908099 - Indirect Label Addressing
Q 908101 - Histogram, Bar Graph and Point Plotter
Q 908102 - Cartesian Graph Conversion
Q 908105 - Improved Cartesian Graph
Q 908107 - Data Packing and Unpacking
Q 908114 - Group Register Operations and Moves Utility
KQ 908115 - Utility Routine: Subroutine and Loop Timer
Q 908116 - Alpha Printout System
Q 908117 - Alphabetic Register Loading and Print-out Aid
Q 908118 - 52 to 59 Code Converter
1Q 908119 - Magnetic Card Comparator
1Q 908120 - Utility Routines VI: Conditional Transfers
Q 908121 - Plitter, Op 07 with Background Grid
Q 908123 - Integer Labeling Subroutine
KQ 908125 - Message Encoder and Decoder
1Q 908127 - Alphanumeric List Op Code
Q 908128 - Variable Print Symbol Banner
Q 908130 - Vertical Banner Program
1Q 908133 - Seven Letters from Display Register
Q 908134 - Word Processor
Q 908137 - Coordinate Graphics
Q 908181 - SR-52 Program Converter
Q 908182 - Advanced Register Manipulations
Q 908201 - Indirect Addrsaing with HIR Registers
Q 908204 - Reversible Banner
Q 918004 - Superaleuth
Q 918008 - Star Buster
Q 918009 - Learning Nim
Q 918012 - Baseball
Q 918013 - Parachute
Q 918015 - Yahtzee
Q 918016 - Tic Tac Toe
Q 918018 - Golf
Q 918021 - Craps Revisited
Q 918024 - "3-D" Tic-Tac-Toe
KQ 918028 - Football (Solitaire)
Q 918029 - Time Bombs
1Q 918030 - Poem Machine
Q 918036 - Triple Yahtzee Scorekeeper
KQ 918037 - Electronic Golf
KQ 918038 - Horse Race
Q 918040 - Offensive versus Defensive Football
Q 918047 - Space War
4KQ 918048 - Seven Card Low Poker
Q 918049 - Hexpawn
Q 918051 - Thirty-one; A Card Game
5KQ 918056 - Automatic Blackjack with Options
KQ 918060 - Hamurabi
kQ 918066 - Basketball Game
4KQ 918073 - Automatic Draw Poker
KQ 918075 - Racetrack

More PPX Program Availability - (cont)

Q 918077 - Mars Lander with Display
KQ 918079 - Deluxe Baseball
KQ 918080 - Ant Invasion
KQ 918081 - Game of Blitz
KQ 918083 - Battleship with Grid Plotter
Q 918085 - Astrogator Game
4KQ 918090 - Seven Card Stud Poker
Q 918094 - Daily Biorhythm Compatibility
Q 918096 - Wumpus
9KQ 918097 - Pictorial Black Box
Q 918099 - Space Flight
KQ 918100 - Blackjack for Thirteen Players
1KQ 918101 - TI-59/PC-100A Demonstration
4KQ 918107 - Automatic Crap Game
Q 918108 - Klingon Space Attack
KQ 918115 - Hangman
Q 918116 - Star Trek (Min-version)
4KQ 918117 - Action Craps
KQ 918118 - Advanced Baseball
KQ 918120 - Pattern Planner Game
KQ 918121 - Dungeons and Dragons
KQ 918124 - Wipeout
KQ 918126 - Duel
Q 918128 - Expert Wumpus
KQ 918132 - Road Race
KQ 918146 - Ineractive Battleship
KQ 918151 - Hangman for Two Players
KQ 918162 - Cross Country Auto Race
1KQ 918163 - Hunt the Wumpus (Short Version)
KQ 918164 - Mulit-player Race Car
Q 918174 - Space War (2 Dimensions)
Q 918176 - Tank Battle for Two Players
Q 918178 - Star Wars
9KQ 918185 - Baseball 3
Q 968016 - Single Song/Album Sorter

Code 1 means the programs are available on a loan basis from TI PPC Notes. Send one dollar (two dollars overseas) for each program you wish to borrow. Prompt return is required so the programs will be available to other members.

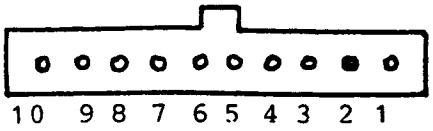
Code D means the programs are available from Gilbert Farrior, 1277 North Warson Road, St. Louis MO 63132. Send a stamped and self-addressed envelope (SASE) for information on specific programs.

Code G means the programs are available from Michael G. Kelley, 11002 W. 21st St., RR #9, Wichita KS 67212. Send a SASE for terms.

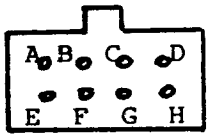
Code K means the programs are available from Albert Smith, 14280 Sandhurst St., Brooksville, FL 33573. Send a SASE for the costs of individual programs.

Code Q means the programs are available from Morris E. Karp, 840 Threadneedle St., No. 181, Houston, TX 77079-2820. Send a SASE for details.

USING CC-40 PERIPHERALS WITH THE TI-74 - Maurice Swinnen



Rear view TI-74



Hex-Bus cable

In spite of reports to the contrary, it is possible to use all of the CC-40 peripherals with the TI-74. All you need is a special interconnecting cable. The connections have been kept a deep secret by certain interested parties. Experimenation revealed the following:

The TI-74 has a 10-pin connector as show above. Diameter of the pins is 0.016" or 0.4 mm. Distance between pins is 0.1" or 2.54 mm. Does that ring a bell? Of course it does! That is the same distance and diameter used with ICs and its inline dip-sockets. The connector on the TI-74 uses only 8 of the 10 pins, pins 1 through 8. To make a good female connector for it, simply saw off half of a 16-pin dip-socket, preferrable one used for wire-wrap, with long, sturdy wire-wrap stems. They allow for neat soldering. If you then shrink enough heat-shrinkable tubing around each stem so as to fill up the space between stems, it is possible to shrink one, large-diameter piece of heat-shrinkable tubing around everything and obtain an almost perfect socket.

If you prefer to have all ten pins used, cut one side of a 24-pin dip-socket and trim off two pins. That will give you a 10-pin socket. You could even glue a small hump on the top, to prevent reverse plug-in. (I tried reverse plug-in; it doesn't harm anything, but, of course, it doesn't work either)

For the hex-bus connector you will have to sacrifice a dual-plug hex-bus cable. Take a long one, cut it in two, and share the other half with a friend. When you strip the wires, you will find, of course, 8 of them. All will be soldered to corresponding pins on your TI-74 connector, except one, the green one. Insulate it, and forget it. Wires are soldered as follows:

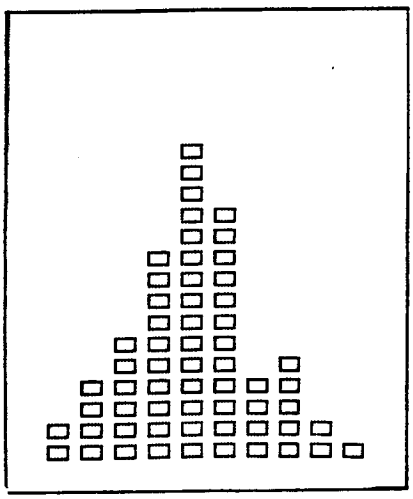
TI-74 connector	Hex-Bus connector	Color
1	D	orange
2	not connected	
3	C	red
4	E	brown
5	H	blue
6	G	black
7	B	yellow
8	A	grey
9	not connected	
10	not connected	

I tried it with the following peripherals with great success: Printer 80, RS232 Interface, the Disk Drive and the Printer-Plotter. I tried to have one TI-74 to talk to one CC-40, without success. The TI-74 is willing, but the CC-40 acts finicky and tells me that its memory may be lost. Well...

Editor's Note: I made the cable and successfully demonstrated use of My TI-74 with the HX-1000. This cabling has not been approved by TI and use could result in loss of warranty.

BAR GRAPHS AND HISTOGRAMS - Walter Bodenmuller wrote for information on programs which would plot bar graphs and histograms. A quick search of back issues of TI PPC Notes found a simple bar graph plotter subroutine for the TI-59/PC-100 by Bill Beebe in V6N9/10P10.

```
000  03  3      022  69 DP     044  03  3      066  00  0
001  02  2      023  02  02    045  02  2      067  95  =
002  03  3      024  03  3     046  69 DP     068  50 I×I
003  02  2      025  02  2     047  04  04     069  42 STO
004  03  3      026  03  3     048  69 DP     070  00  00
005  02  2      027  02  2     049  05  05     071  83 GO*
006  03  3      028  03  3     050  92 RTN    072  00  00
007  02  2      029  02  2     051  76 LBL    073  76 LBL
008  03  3      030  03  3     052  11  A     074  12  B
009  02  2      031  02  2     053  69 DP     075  42 STO
010  69 DP      032  03  3     054  00  00    076  20  20
011  01  01     033  02  2     055  55  ÷     077  98 ADV
012  03  3      034  69 DP     056  05  5     078  76 LBL
013  02  2      035  03  03    057  75  -     079  13  C
014  03  3      036  03  3     058  04  4     080  73 RC*
015  02  2      037  02  2     059  85  +     081  20  20
016  03  3      038  03  3     060  59 INT    082  11  A
017  02  2      039  02  2     061  55  ÷     083  97 DSZ
018  03  3      040  03  3     062  05  5     084  20  20
019  02  2      041  02  2     063  95  =     085  13  C
020  03  3      042  03  3     064  65  ×     086  98 ADV
021  02  2      043  02  2     065  01  1     087  91 R/S
```

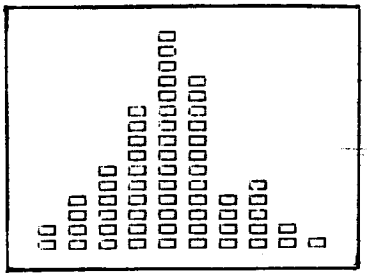


To use the routine place the values for the bar graph in sequence starting at data register R01. Place the number of the highest data register used in the display and call subroutine B . The chart at the right above was made using this routine. The program places the bottom of the bar graph at the right side of the paper. Thus, if the values for the bars are located starting in data register R01 and ending in data register RXY, then the use of the number of the highest data register to start the Dsz loop will yield the desired histogram.

The first 73 steps in the listing above are from V6N9/10. Some very clever programming is involved to calculate an indirect address such that the program jumps to the proper location to generate the needed values in the display for use by the Op 01 through Op 04 commands. The user must control the values to be plotted to be in the range from zero to twenty. If a plot value larger than 20 is used a wrap-around effect occurs; for example, for a value of 27, a bar of 13 units will be printed. For negative plot values unpredictable problems will occur, including entry into "do-nothing" loops .

A comparable TI-95 program is less complex since the manipulation to accommodate the Op 01 through Op 04 commands is not required. Rather, a sequence of alphanumeric 0's is placed at the beginning of the program list, and the bar graph subroutine selects the proper point at which to enter the chain. The data to be plotted is stored in sequence starting at data register 001, and the number of the highest register used is entered at entry to the routine. The routine shown below builds the bargraph starting at the left side of the paper. Therefore, the values to be plotted must be stored in reverse order if a Dsz command is used to sequence through the values to be plotted.

```
0000 GTL AA `0000000000`     0046 LBL AA CLR ADV BRK
0013 `00000000000000`        0052 STO 020
0027 PRT RTN                 0055 LBL BB RCL IND 020
0029 LBL HI +/- +27=         0062 SBL HI DSZ 020
0037 STO 000 CE              0068 GTL BB HLT
0041 SBR IND 000 RTN
```



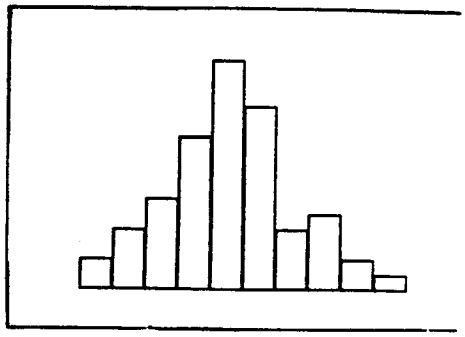
Bar Graphs and Histograms - (cont)

A routine for the TI-74 and PC-324 is even less complex due to the availability of the RPT$ command. See the program at the right. To use the plot routine the values to be plotted must be stored in sequence in the Array U. Lines 1020 through 1040 of the routine use Maurice Swinnen's technique for entering values as strings, and for marking the end of input with entry of an "E". The routine in lines 1050 through 1070 print the bar graph, where line 1060 uses the RPT$ command to generate a sequence of 0's of the required length. The bar graph is the same as that from the TI-95 and PC-324.

```
1000 DIM U(20)
1010 OPEN #1,"12",OUTPUT
1020 INPUT X$
1030 IF X$="E"OR X$="e"T
HEN 1050
1040 K=K+1:U(K)=VAL(X$):
GOTO 1020
1050 FOR I=1 TO K
1060 PRINT #1,RPT$("0",U
(I))
1070 NEXT I
1080 END
```

For more elegant bar graphs the TI-74 user can use the interface cable from page 13 of this issue to obtain the graphics capability of the HX-1000 Printer/Plotter. A sample program and bar graph appear below.

```
100 DIM X(20)
110 INPUT X$
120 IF X$="E"OR X$
="e"THEN 200
130 K=K+1
140 X(K)=VAL(X$)
150 GOTO 110
200 OPEN #1,"10",O
UTPUT
210 PRINT #1,CHR$(
19)
300 FOR I=1 TO K
310 Y$=STR$(X(I)*1
0)
320 P$="L(0,0),("&
Y$&",0),("&Y$&",-2
0),(0,-20),(0,0)"
330 PRINT #1,"O"
340 PRINT #1,P$
350 PRINT #1,"M(0,
-21)"
360 NEXT I
400 PRINT #1,CHR$(
17)
410 CLOSE #1:END
```



Bar graphs can be generated easily in the display of the fx-7000G using the SD2 mode and the procedures described in pages 83-86 of the Owner's Manual. The example and the resulting display which appear in the manual fail to capture the versatility of the mode; for example,

* The x and the y range settings can be negative values.

* The SD2 mode will accept negative frequencies. Negative values can be combined with a negative Ymin range to yield bar graphs with bars which extend below the x axis.

* Entries with the DT key can be made with any sequence of X values.

* The width determined by the Xmin and Xmax values set by the range function is divided by the number of additional memories set by the Defm command to define the width of the individual cells.

* An entry with a value of X goes into the cell which includes the value of X within its limits. The X inputs are not limited to integers.

* The summations which are used to calculate the mean and standard deviation are accumulated into memories U through W:

$\Sigma x^2y \rightarrow U$ $\Sigma xy \rightarrow V$ $\Sigma y \rightarrow W$

* Entry through the DT key acts to sum the frequencies into the appropriate Z(1), Z(2), ... memory, and also acts to update the statistics summations in memories U, V and W. Thus, the frequencies do not have to be summed by hand for each X value prior to data entry.

Bar Graphs and HIstograms - (cont)

* The Xmin and Xmax range values set the limits of the X input values which can appear in the histogram. Values outside that range will be included in the statistics summations but will not appear in the displayed histogram.

* The y values of the bar graph can be changed by directly storing values into memories Z(1), Z(2), etc. However, the statistics summations in memories U, V, and W will be in error. It would have been convenient if the next use of the DT key would correct the statistics summations, but that is not the case.

An example will illustrate some of the features described above. Set the range to -5, 5, 1, 0, 20 and 2. Use Defm 8 to expand the memories by eight to provide eight cells. The edges of the cells should be at intervals of 1.25 . Now, enter data as follows:

-5:1 DT places a frequency of 1 in the first cell, memory Z(1).

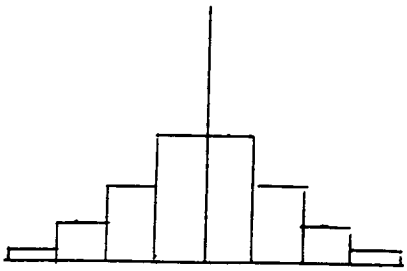
4.9;1 DT places a frequency of 1 in the right hand histogram cell, memory Z(8). An input sequence of 5;1 DT would have increased the summation in the statistics registers (U, V, W), but would not have placed a frequency of 1 in the right hand histogram cell.

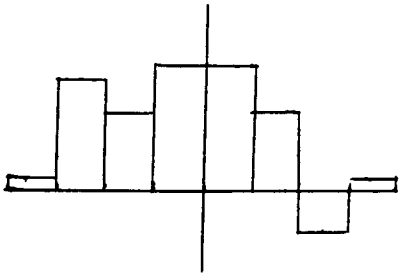
-3;3 DT , -2;6 DT , -1;10 DT , 1;10 DT , 2.49;6 DT , and 2.5;3 DT place frequencies in the remaining cells. Again, the last X value of 2.5 is on the border between the sixth and seventh cells from the left, and is placed in the seventh cell.

Press Graph EXE and see the display at the left below. Press shift $\bar{x}$ EXE and see the mean 0.0335 . Press shift $x\sigma n$ EXE and see the unbiased standard deviation 2.09711534.

Change the range to -5, 5, 1, -6, 12, 2. Increase the frequency in the second cell from the left with the sequence -3;5 DT . Change the frequency in the sixth cell from the left to a negative value with the sequence pressing 3;-5 DT . Press Shift Cls EXE Graph EXE and see the display in the center below. The mean will be -0.7165 and the unbiased standard deviation will be 1.971203376 .

Key in the sequence 3 →Z(2) EXE and 3 →Z(7) . Press Shift Cls EXE Graph EXE and see that the histogram has returned to that at the left below. Check the mean and standard deviation and see that the values did not return to the earlier ones.





Bar Graphs and Histograms - (cont)

A histogram, bar graph and point plotter program for the TI-59 by W. Buechner provides five different plotting formats (PPX 908101). The program listing is on page 18. The instructions for using the programs are:

To clear memory for new data: press 2nd E'.

For data entry, press E. Then enter each frequency in sequence and press R/S. Each item is printed as it is entered. A zero entry is not acceptable since the plot routine looks for a zero to indicate the end of the input data. If you want a zero frequency in the printout, enter a very small non-zero number instead.

The five plotting options are:

1. For a histogram with automatically scaled limits press 2nd A'.

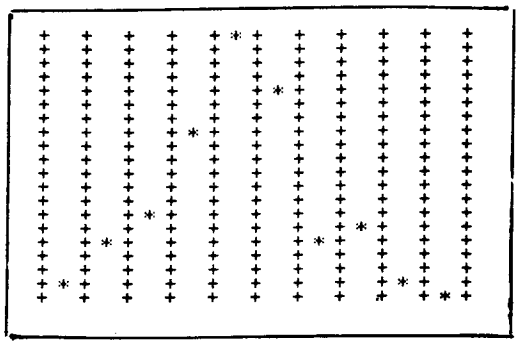
2. For a histogram with markers at the limits, press C, then A'.

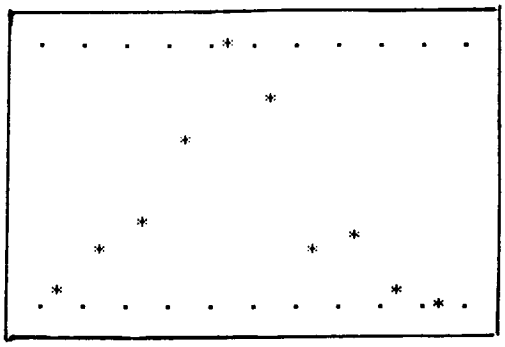
3. For a bar plot, press D, then A'.

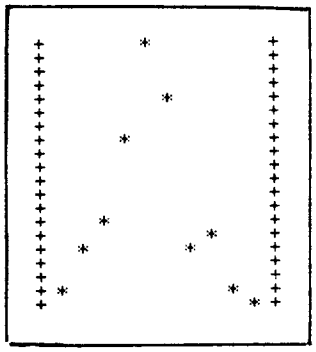
4. For a point plot with a scale on the edges press B', then A'.

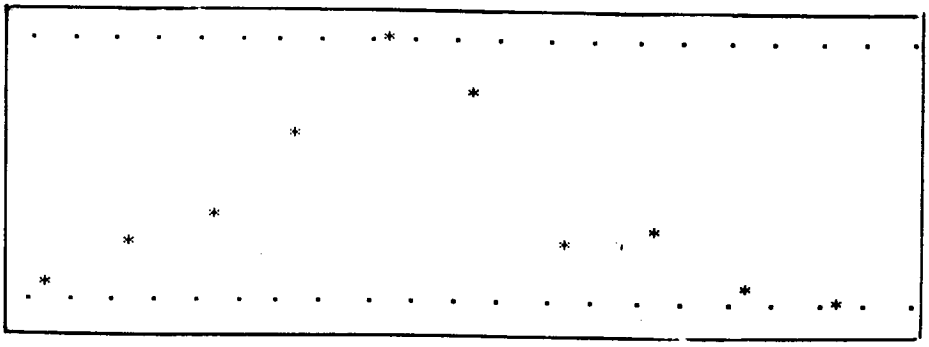
5. For a point plot with marks on the edges press C', then A'.

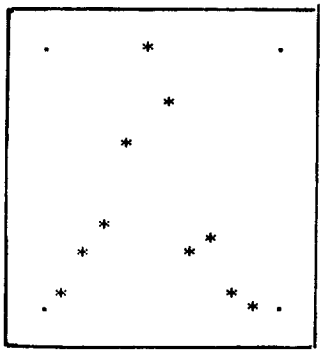
To make more than one kind of plot with the same data, press E before going to the next plot. To set your own limits, then before pressing A' you must enter the upper limit and press A, enter the lower limit and press R/S. Illustrations of the five plots follow.











Bar Graphs and Histograms - (cont)

Program Listing:

```
000  91 R/S    080  44 SUM    160  02  2     240  42 STO    320  00  0     400  06  06
001  76 LBL    081  56  56    161  69 DP     241  00  00    321  00  0     401  87 IFF
002  12  B     082  01  1     162  04  04    242  22 INV    322  00  0     402  02  02
003  61 GTO    083  44 SUM    163  43 RCL    243  86 STF    323  00  0     403  04  04
004  04  04    084  57  57    164  56  56    244  05  05    324  00  0     404  08  08
005  12  12    085  43 RCL    165  55  ÷     245  22 INV    325  69 DP     405  61 GTO
006  76 LBL    086  55  55    166  43 RCL    246  86 STF    326  01  01    406  02  02
007  13  C     087  87 IFF    167  57  57    247  06  06    327  04  4     407  81  81
008  86 STF    088  00  00    168  95  =     248  87 IFF    328  00  0     408  98 ADV
009  01  01    089  00  00    169  69 DP     249  04  04    329  69 DP     409  61 GTO
010  91 R/S    090  92  92    170  06  06    250  03  03    330  04  04    410  02  02
011  76 LBL    091  99 PRT    171  98 ADV    251  13  13    331  69 DP     411  48  48
012  14  D     092  87 IFF    172  98 ADV    252  87 IFF    332  05  05    412  42 STO
013  86 STF    093  05  05    173  69 DP     253  01  01    333  87 IFF    413  59  59
014  01  01    094  00  00    174  00  00    254  03  03    334  06  06    414  68 NOP
015  86 STF    095  66  66    175  02  2     255  13  13    335  03  03    415  68 NOP
016  02  02    096  87 IFF    176  07  7     256  04  4     336  98  98    416  68 NOP
017  91 R/S    097  06  06    177  02  2     257  07  7     337  87 IFF    417  68 NOP
018  76 LBL    098  01  01    178  04  4     258  04  4     338  05  05    418  91 R/S
019  15  E     099  06  06    179  03  3     259  07  7     339  03  03    419  42 STO
020  61 GTO    100  42 STO    180  00  0     260  04  4     340  52  52    420  58  58
021  04  04    101  58  58    181  69 DP     261  07  7     341  61 GTO    421  43 RCL
022  28  28    102  42 STO    182  02  02    262  04  4     342  02  02    422  53  53
023  76 LBL    103  59  59    183  02  2     263  07  7     343  81  81    423  42 STO
024  16 A'     104  86 STF    184  04  4     264  04  4     344  87 IFF    424  00  00
025  42 STO    105  06  06    185  03  3     265  07  7     345  04  04    425  61 GTO
026  03  03    106  43 RCL    186  07  7     266  69 DP     346  03  03    426  01  01
027  61 GTO    107  59  59    187  03  3     267  01  01    347  84  84    427  72  72
028  00  00    108  32 X:T    188  06  6     268  69 DP     348  87 IFF    428  01  1
029  61  61    109  43 RCL    189  00  0     269  02  02    349  03  03    429  01  1
030  76 LBL    110  55  55    190  00  0     270  69 DP     350  03  03    430  42 STO
031  17 B'     111  77  GE    191  00  0     271  03  03    351  79  79    431  00  00
032  86 STF    112  01  01    192  00  0     272  69 DP     352  00  0     432  42 STO
033  03  03    113  26  26    193  69 DP     273  04  04    353  42 STO    433  53  53
034  91 R/S    114  43 RCL    194  03  03    274  69 DP     354  56  56    434  25 CLR
035  76 LBL    115  58  58    195  69 DP     275  05  05    355  42 STO    435  86 STF
036  18 C'     116  32 X:T    196  05  05    276  87 IFF    356  57  57    436  07  07
037  86 STF    117  43 RCL    197  04  4     277  05  05    357  42 STO    437  91 R/S
038  04  04    118  55  55    198  01  1     278  03  03    358  58  58    438  86 STF
039  91 R/S    119  22 INV    199  03  3     279  52  52    359  42 STO    439  00  00
040  76 LBL    120  77  GE    200  03  3     280  29 CP     360  59  59    440  72 ST*
041  19 D'     121  01  01    201  03  3     281  73 RC*    361  98 ADV    441  00  00
042  86 STF    122  31  31    202  05  5     282  00  00    362  98 ADV    442  69 DP
043  08  08    123  61 GTO    203  69 DP     283  67  EQ    363  98 ADV    443  20  20
044  91 R/S    124  00  00    204  04  04    284  03  03    364  98 ADV    444  99 PRT
045  76 LBL    125  66  66    205  43 RCL    285  44  44    365  87 IFF    445  61 GTO
046  10 E'     126  42 STO    206  59  59    286  75  -     366  07  07    446  04  04
047  61 GTO    127  59  59    207  69 DP     287  43 RCL    367  03  03    447  37  37
048  03  03    128  61 GTO    208  06  06    288  58  58    368  75  75    448  42 STO
049  69  69    129  00  00    209  02  2     289  95  =     369  47 CMS    449  55  55
050  76 LBL    130  66  66    210  07  7     290  65  ×     370  98 ADV    450  69 DP
051  11  A     131  42 STO    211  04  4     291  43 RCL    371  98 ADV    451  20  20
052  42 STO    132  58  58    212  03  3     292  54  54    372  98 ADV    452  73 RC*
053  59  59    133  61 GTO    213  03  3     293  95  =     373  25 CLR    453  00  00
054  86 STF    134  00  00    214  05  5     294  69 DP     374  81 RST    454  67  EQ
055  05  05    135  66  66    215  69 DP     295  07  07    375  04  4     455  04  04
056  91 R/S    136  98 ADV    216  04  04    296  69 DP     376  07  7     456  73  73
057  42 STO    137  69 DP     217  43 RCL    297  20  20    377  95  =     457  75  -
058  58  58    138  00  00    218  58  58    298  87 IFF    378  81 RST    458  43 RCL
059  68 NOP    139  07  7     219  69 DP     299  04  04    379  86 STF    459  55  55
060  68 NOP    140  07  7     220  06  06    300  02  02    380  05  05    460  95  =
061  98 ADV    141  69 DP     221  98 ADV    301  81  81    381  61 GTO    461  69 DP
062  43 RCL    142  04  04    222  98 ADV    302  87 IFF    382  02  02    462  30  30
063  53  53    143  43 RCL    223  43 RCL    303  01  01    383  48  48    463  68 NOP
064  42 STO    144  56  56    224  59  59    304  03  03    384  86 STF    464  68 NOP
065  00  00    145  69 DP     225  75  -     305  93  93    385  05  05    465  68 NOP
066  29 CP     146  06  06    226  43 RCL    306  87 IFF    386  61 GTO    466  68 NOP
067  73 RC*    147  03  3     227  58  58    307  03  03    387  03  03    467  68 NOP
068  00  00    148  01  1     228  95  =     308  02  02    388  13  13    468  72 ST*
069  67  EQ    149  69 DP     229  35 1/X    309  81  81    389  87 IFF    469  00  00
070  01  01    150  04  04    230  65  ×     310  61 GTO    390  05  05    470  61 GTO
071  36  36    151  43 RCL    231  01  1     311  02  02    391  06  06    471  00  00
072  87 IFF    152  57  57    232  09  9     312  48  48    392  52  52    472  76  76
073  08  08    153  69 DP     233  93  .     313  69 DP     393  86 STF    473  69 DP
074  04  04    154  06  06    234  05  5     314  00  00    394  06  06    474  30  30
075  48  48    155  01  1     235  95  =     315  04  4     395  61 GTO    475  72 ST*
076  42 STO    156  03  3     236  42 STO    316  00  0     396  02  02    476  00  00
077  55  55    157  04  4     237  54  54    317  00  0     397  48  48    477  61 GTO
078  69 DP     158  02  2     238  43 RCL    318  00  0     398  22 INV    478  01  01
079  20  20    159  02  2     239 -53  53    319  00  0     399  86 STF    479  36  36
```

MULTIPLE LINEAR REGRESSION WITH TWO VARIABLES - P. Hanson. I needed a linear regression with two variables for an application where I work. I modified a Model 100 program for regression with user defined functions similar to that on V12N1P14. I needed a sample problem to check the capability of the modified program and remembered that there was an equivalent program and a sample problem in the Statistics Library book reviewed on page 4 of this issue. When I entered the values from the table on page 50 of the book I got answers that were different from those listed on page 51 of the book. I knew that the different numerical precision of the Model 100 and the Sharp machines and the different methods of solution would lead to somewhat different results, but the actual difference seemed too large. I then converted the program from the book for the Model 100, entered the sample problem from page 50 of the book, and got an answer which agreed with the result with my modified version of the user defined function program to seven decimal places. After a considerable amount of rechecking of programs and data entries I found that there was a typographic error in the table in the book. The first y value in the table was listed as -9.9 when it should have been -9.99 .

One thing that this exercise did prove is how versatile the program from V12N1P14 is. The only changes required to accept two independent variables W and X, and one dependent variable Y are:

1. Add W(50) in line 100.

2. Add the following lines:

```
143  A$="W"&STR$(I)&" = ":INPUT A$;W(I)

146  IF PN<>0 THEN PRINT #PN,A$,W(I)
```

3. For a linear regression with two variables enter the following user defined functions at the end of the program:

```
810 F(1)=1
820 F(2)=W(L)
830 F(3)=X(L)
840 RETURN
```

The printout from a modified TI-74 program for the sample problem from the Statistics Library book appears at the right. Some other solutions for the same sample problem are:

```
W1 =          1
X1 =         -9.99
Y1 =          40.16

W2 =          2
X2 =         -4.98
Y2 =          28.74

W3 =          3
X3 =          0
Y3 =          17.32

W4 =          4
X4 =          5.01
Y4 =          5.9

W5 =          5
X5 =          9.98
Y5 =         -5.53

W6 =          6
X6 =          15
Y6 =         -16.95

W7 =          7
X7 =          19.98
Y7 =         -28.37

A1 =  53.55750988
A2 = -12.08011949
A3 =  .1317364727

d1 = -.0013430286
d2 = -.0012232672
d3 =  .0028485882
d4 =  .0029683496
d5 = -.0016424302
d6 = -.0028400336
d7 =  .0012318219

Mean =  4.285714E-13

S.E. =  .0028507199
```

Sharp from the book, pages 50-51	53.55751069	-12.08011976	0.1317365269
Model 100, book method	53.5575106929	-12.080119760479	0.13173652694611
Model 100, converted from V12N1P14	53.53750998009	-12.080119522791	0.13173647935745

On the next page we will see a more dramatic illustration of the differences between solutions on different machines.

MANY DIGIT SQUARE ROOT IN BASIC - Larry Leeds

Larry writes Basic programs on the Radio Shack Model 100 which is a fourteen digit, base 10 machine. Conversion for use on the CC-40 and TI-74, which are 13-14 digit, base 100 machines, is typically straightforward. But this program for calculation of multipreciaion square roots was different. The program listing is at the right. Larry's instructions were "Move the decimal point two places at a time until the remaining integer is one or two digits. If one digit, partition with seven digits in the first group, with the remaining groups six digits each. If two digits, partition with six digits in each group. Add trailing zeroes to the last group if needed." In response to the program prompts you should provide twice as many digits for N as for the square root of N.

Line 70 is provided to make it possible to generate square roots of integers without entering all of the trailing zeroes. For $\sqrt{3}$ simply remove the REM indication. For square roots of other integers make a suitable replacement for the 3.E+06 in line 70.

Page 21 includes a short addition to the program which makes it possible to run a test problem which yields an answer of 20 blocks of all nines. With steps 800-870 in place simply enter RUN 800 in the display, press ENTER, and wait about minute for the solution to appear in the display.

The first problems that I tried with this program were square roots of single digit integers. The square roots of 2, 6, and 8 were normal, but the square roots of 3, 5, and 7 yielded negative values in the third block of the output. Tests of the square roots of 2, 3, 5, 6, 7, and 8 for 100 blocks (600 digits) did not show negative values in the output at any other position. The program was modified to work with five digit blocks to see if the problem would be cleared. (V12N2P12 had reported that another program had problems when converted from the Model 100 to the TI-74, and the problems were cleared by changing from six digit blocks to five digit blocks.) The solutions for $\sqrt{3}$ and $\sqrt{7}$ were correct, but the solution for $\sqrt{5}$ continued to have a negative value for the third block. The printouts for the first six blocks of $\sqrt{3}$ and $\sqrt{5}$ are shown below for both the six digit and five digit block programs.

1732050
807569
-122707
527446
341505
872366

2236067
977500
-210304
409173
668731
276235

173205
08075
68877
29352
74463
41505

223606
79775
0-211
69640
91736
68731

The same problem did not appear with the Model 100 implementation. Comparative testing isolated the source of the TI-74 problem to the first statement in line 150. Testing with the statement D=INT(C*Z)+1 changed to D=INT(C*Z) showed that the negative output problem had been eliminated for the square roots of 3, 5, and 7. See the printouts for $\sqrt{3}$ and $\sqrt{5}$ on the next page.

```
10 DIM N(200)
20 Z=1.E+06:J=1
30 PRINT "*** SQR(N) ***
":PAUSE 2
35 INPUT "Use Printer Y/
N ? ";Z$
40 IF Z$="N"OR Z$="n"THE
N 50
45 PN=1:OPEN #1,"12",OUT
PUT
50 INPUT "Number of grou
ps in Sqr(N) ? ";W
60 INPUT "Number of grou
ps in N ? ";X
70 REM N(1)=3.E+06:GOTO
115
80 FOR I=1 TO X
90 N$="N("&STR$(I)&") =
"
100 INPUT N$;N(I):NEXT I
115 PRINT "wait"
120 A=N(J)*Z^3+N(J+1)*Z^
2
130 B=SQR(A)/Z
140 C=B-INT(B)
150 D=INT(C*Z)+1:B=B-C
160 N(J+1)=N(J)*Z+N(J+1)
-B^2
170 B=B*2:N(J-1)=B
180 N(J)=D:F=W-1:G=W+1
200 H=G-F:L=H:J=H
210 N(J+1)=N(J)*Z+N(J+1)
:N(J)=0:J=J+1
220 J=J+H-1:K=H-1:A=K:Q=
J:C=Z
250 M=N(J)+C-Z-N(K)*D
260 C=M/Z+Z
270 P=C-INT(C)
280 C=C-P:N(J)=P*Z
290 K=K-1:J=J-1:H=H-1
300 IF H<>0 THEN 250
310 IF C=Z THEN 400
320 M=(C-Z)*Z:J=J+1:N(J)
=N(J)+M
330 IF M>=Z THEN 400
340 C=0:D=D-1:H=L:K=A:J=
Q:N(J)=N(J)+D
350 FOR Y=H TO 1 STEP -1
360 C=(N(K)+N(J)+C)/Z
370 M=C-INT(C)
380 N(J)=M*Z:C=C-M:K=K-1
:J=J-1:NEXT Y
390 J=J+1:N(J)=N(J)+C*Z:
N(A)=D
400 H=L-1:K=A:N(K)=N(K)+
D
410 FOR Y=H TO 1 STEP -1
420 IF N(K)<Z THEN 460
430 C=N(K)/Z:M=C-INT(C)
440 N(K)=M*Z:C=C-M:K=K-1
450 N(K)=N(K)+C:NEXT Y
460 F=F-1:IF F<>0 THEN 6
00
470 H=L:K=0:C=0
490 N(K)=(N(K)+C)/2
500 M=N(K)-INT(N(K))
510 N(K)=N(K)-M:C=M*2*Z
520 K=K+1:H=H-1:IF H<>0
THEN 490
530 PAUSE ALL:FOR I=0 TO
 W-1
540 L=6-LEN(STR$(N(I))):
IF L<0 THEN L=0
550 N$=RPT$("0",L)&STR$(
N(I))
560 PRINT #PN,N$:NEXT I:
END
600 K=A+2
610 M=((N(K)*Z+N(K+1))*Z
+N(K+2))*Z+N(K+3)
615 K=K+3
620 S=((N(0)*Z+N(1))*Z+N
(2))
630 T=M/S:D=INT(T):A=A+1
:N(A)=D
640 GOTO 200
```

Many Digit Square Root in Basic - (cont)

```
1732050
807568
877293
527446
341505
872366
```

```
2236067
977499
789696
409173
668731
276235
```

```
800 W=21:X=40:J=1:Z=1.E+
06
810 FOR K=1 TO 19
820 N(K)=999999:NEXT K
830 N(20)=999998
840 FOR K=21 TO 39
850 N(K)=0:NEXT K
860 N(40)=1
870          GOTO 115
```

No other problems have been identified to date with this change in place. Furthermore, no problem has been identified when the same change is made to the Model 100 program. Even so, we were uneasy.

Myer Boland ran the program on his TI-99/4 which has the same 13/14 digit, base 100 mechanization as the TI-74 and CC-40. He found the same negative value problem in the third block of the output. His solution was to change the first part of the user instructions to "Move the decimal point two places at a time until the remaining integer is one or two digits. If one digit, partition with five digits in the first group, with the remaining groups six digits each. ... ". Thus, to solve for the square root of three, the first input block would be 30000. No problems have been identified with this method on either the TI-99/4, the TI-74 or the Model 100. The printouts for the square roots of 2, 3, 5, 6, 7 and 8 are:

```
141421
356237
309504
880168
872420
969807
```

```
173205
080756
887729
352744
634150
587236
```

```
223606
797749
978969
640917
366873
127623
```

```
244948
974278
317809
819728
407470
589139
```

```
264575
131106
459059
050161
575363
926042
```

```
282842
712474
619009
760337
744841
939615
```

In a sense the negative output is correct. Consider the square root of five solution where the second and third lines are 977500 and -210304. If a one is borrowed from the second line it becomes 977499. If the magnitude of the third line is subtracted from the borrowed 1000000 the third line becomes 789696. The modified lines are correct. This suggests another solution for the negative output problem: simply scan the solution before displaying or printing it. If any group is negative perform the borrowing process outlined above.

Although the negative output has only appeared in the third group of any solution that has been tried to date some experimentation will show that negative values have apeared in other groups while the program is running. This ability to reach "correct" solutions even if some of the elements become negative suggests that the algorithm is particularly forgiving. Can one of our members explain?

MORE USED TI-59'S - These calculators are available "as is". They will have passed all of the tests described in pages 4 through 7 of V12N1. Send a check for sixty dollars to the editor. I will give fifty dollars to the owner, ship the calculator to you, and return any part of the ten dollar shipping fee which was not used.

CALCULATORS WANTED - TI-56 and TI-58C. Write to Steve Naiser, Building B, 9051 Mansfield Road, Shreveport, LA 71118.

PC-200'S WANTED - This is the printer used with the TI-66 and the BA-55. Write to Lars Herold Andersen, 5 Ravnsbjerg Hegn, DK 7400 Herning, Denmark.

MORE ON KAPREKAR'S CONSTANT - V12N2P22 presented W. H. Widmer's discussion of Kaprekar's constant together with a program for calculating the constant for the four digit case. V12N2P23 reported that in tests of several values for N(0) the highest number of iterations required for solution was six, and asked if there was any N(0) for which i>7.

Larry Leeds reports: "Using the Radio Shack Model 100 and a program for four digit numbers I have tested all the pertinent numbers from 0001 to 9998. Each number leads to the constant 6174, and the maximum number of iterations never exceeds 7. Professor Widmer asked for a proof; although this is not a theoretical proof, perhaps he will accept this result in lieu of a formal proof ..."

The entry of N(0) for the program on V12N2P22 was cumbersome. V12N2P23 asked if someone would write a program which would permit entry of N(0) as a four digit number and which would run in fast mode. Lars Herold Andersen of Denmark submitted a shorter program which runs in fast mode:

```
000  91 R/S   027  43 RCL   054  00  00   081  86  86   108  65  ×   135  00  0
001  25 CLR   028  04  04   055  77  GE   082  48 EXC   109  09  9   136  00  0
002  32 X:T   029  22 INV   056  00  00   083  03  03   110  00  0   137  00  0
003  99 PRT   030  28 LOG   057  62  62   084  42 STO   111  95  =   138  00  0
004  32 X:T   031  54  )    058  48 EXC   085  02  02   112  69 DP   139  00  0
005  43 RCL   032  59 INT   059  01  01   086  97 DSZ   113  25  25  140  00  0
006  06  06   033  72 ST*   060  42 STO   087  04  04   114  61 GTO   141  76 LBL
007  22 INV   034  04  04   061  00  00   088  00  00   115  00  00  142  11  A
008  67  EQ   035  65  ×    062  43 RCL   089  50  50   116  03  03  143  47 CMS
009  00  00   036  43 RCL   063  02  02   090  43 RCL   117  00  0   144  32 X:T
010  19  19   037  04  04   064  32 X:T   091  00  00   118  00  0   145  06  6
011  06  6    038  22 INV   065  43 RCL   092  75  -    119  00  0   146  01  1
012  69 DP    039  28 LOG   066  01  01   093  43 RCL   120  00  0   147  07  7
013  17  17   040  97 DSZ   067  77  GE   094  03  03   121  00  0   148  04  4
014  43 RCL   041  04  04   068  00  00   095  95  =    122  00  0   149  42 STO
015  05  05   042  00  00   069  74  74   096  65  ×    123  00  0   150  06  06
016  99 PRT   043  23  23   070  48 EXC   097  09  9    124  00  0   151  01  1
017  98 ADV   044  95  =    071  02  02   098  09  9    125  00  0   152  00  0
018  81 RST   045  42 STO   072  42 STO   099  09  9    126  00  0   153  69 DP
019  03  3    046  00  00   073  01  01   100  85  +    127  00  0   154  17  17
020  42 STO   047  03  3    074  43 RCL   101  53  (    128  00  0   155  04  4
021  04  04   048  42 STO   075  03  03   102  43 RCL   129  00  0   156  05  5
022  32 X:T   049  04  04   076  32 X:T   103  01  01   130  00  0   157  30 TAN
023  75  -    050  43 RCL   077  43 RCL   104  75  -    131  00  0   158  33 X²
024  53  (    051  01  01   078  02  02   105  43 RCL   132  00  0   159  86 STF
025  24 CE    052  32 X:T   079  77  GE   106  02  02   133  00  0
026  55  ÷    053  43 RCL   080  00  00   107  54  )    134  00  0
```

To run the program, enter N(0) as a four digit number and press A. See a flashing "1." in the display. Press 7 and then EE to enter fast mode. The sequence N(0), N(1),...,N(i) and the number of iterations (i) is printed. The program stops with the number of iterations in the display. NOTE: No tests against illegal entries such as 1111, 123, or 12345 are made.

To save program steps and execution time the digits of the maximum integer are stored in R00 through R03, e.g., for 8765, R0 = 8, R01 = 7, R02 = 6, and R03 = 5. Then to obtain the four digit number you can use

$$1000 \times R00 + 100 \times R01 + 10 \times R02 + R03 = 8765$$

and to get the reversed number you can use

$$1000 \times R03 + 100 \times R02 + 10 \times R01 + R00 = 5678$$

and the difference of those two equations will reduce to

$$999 \times (R00 - R03) + 90 \times (R01 - R02) = 8765 - 5678$$

USING THE HX-1000 WITH THE TI-95 - P. Hanson. The cable described on page 13 of this issue can be used to connect the TI-95 and the HX-1000 Printer/Plotter. You must use the INPUT/OUTPUT menu in the TI-95 to set the device number in the TI-95 to match the device number of the HX-1000, which will be 10 or 11 depending on the position of the device number switch. See page 6-4 of the TI-95 User's Guide for detailed instructions. Once the device numbers match you can use all the normal printing functions which are available with the PC-324.

You can set other modes peculiar to the HX-1000 by transmitting appropriate character codes. Suppose that you want to set the compressed print mode, that is, 36 characters per line. Set ALPHA mode, select CHR from the menu, and enter 018. Go out of ALPHA mode and press 2nd PRINT. The other software control codes on page 21 of the HX-1000 User's Manual can be set in a similar manner.

Selection of the Graphics mode (software control code 019) disables the ability to print in the normal manner, as expected. So far, I have been unable to successfully send graphics mode commands. I can send the Text mode (software control code 017) to take the HX-1000 out of graphics mode.

MORE A.O.S. CALCULATORS FROM HP - Educalc Catalog #38 lists three new calculators from Hewlett Packard, the HP-17B Business Calculator ($84.95), the HP-19B Business Consultant II ($139.95), and the HP-27 Scientific/Business Calculator ($84.95). All three calculators emphasize business applications, and all three use algebraic logic. Old-timers who remember the endless arguments over the relative merits of RPN and AOS have to be surprised at an advertisement for an HP machine which states "It calculates with Algebraic Logic, dealing with equations just as you write them."

BOOK REVIEW - TI-74 BASICALC Technical Data Manual. This 30 page manual was announced on page 8 of the latest issue of Programmable Calculator News. It is available from TI for ten dollars. Call 1-800-TI-CARES for the latest information on ordering.

The manual includes descriptive text, block diagrams, schematics, memory maps, and interface details. A must document for those who want to delve deeply into the workings of the TI-74.

THE ML-02 CONVERSION FOR THE TI-95 - Page 3 shows that Hewlett Ladd has completed a conversion of the TI-59 ML-02 program for use with the TI-95. If you use the 7x7 sub-Hilbert test on the converted program and read out the results to 13 digits you will find that the results are in EXACT agreement with those obtained with the TI-95 Mathematics library. See V12N1P8 for the solution of the 7x7 sub-Hilbert with the Mathematics module.

SIGNUM ON THE TI-95; SGN ON THE TI-59 - In V12N2P11 Charlie Williamson challenged members (1) to write a program for the TI-95 which mechanizes the TI-59's OP 10 (true signum), and (2) to write a program for the TI-59 which mechanizes the TI-95's SGN function.

SIGNUM on the TI-95:

Peter Measer: LBL AA STO A SGN - RCL A +/- SGN = / 2 = RTN 15 steps

Lars Andersen: LBL AA * 1/x ABS = RF 53 RTN 9 steps

SGN on the TI-59:

Charlie, Peter and Lars all submitted a routine which depends on the characteristic of the TI-59 where 0/0 yields a flashing 1:

LBL A (CE / |x|) CE RTN 9 steps

Of course, the CE which is required to clear the flashing 1 will also clear any other error indication which may have been present before the routine was called. Charlie also submitted a longer routine which does not have that limitation:

LBL A (OP 10 - |x| + 1) RTN 11 steps

MYSTERY OF THE LOST BITS - Peter Measer observed that the total of the bytes listed for each TI-95 program and the "Bytes Free" is not equal to the number of bytes that is supposedly available in user memory, namely 5200. For example, suppose that I had four programs stored: ISY 472 bytes, PRI 24 bytes, STX 280 bytes, and e 144 bytes. The bytes free will be shown as 4240. The total of the five numbers is 5160, not 5200. What happened to the missing 40 bytes?

A clue is found in the memory map on page C-17 of the TI-95 Programming Guide. There is a "Catalog" listed at the end of the file space. If I search the end of the file space with the system unprotected as in Appendix A of the Programming Guide I will find that eight bytes (3828 through 382F) are used for bookkeeping, and an additional eight bytes are used for each file. For the example described above the last sixteen addresses in the file space, the contents of the last sixteen addresses, and the interpretation of the contents are:

3820	0		
3821	0		
3822	0		
3823	1		From 3823 and 3824, 1D8 in hexadecimal is equivalent
3824	D8		to 472 in decimal, the bytes for the first program.
3825	49	I	From 3825, 3826, and 3827 the equivalent character codes
3826	53	S	from the table on pages C-4 and C-5 of the Programming
3827	59	Y	Guide yield "ISY", the name of the first program.
3828	0		
3829	0		
382A	3		From 382A and 382B, 398 in hexadecimal is equivalent to
382B	98		920 in decimal, the number of bytes in all programs.
382C	10		From 382C and 382D, 1090 in hexadecimal is equivalent
382D	90		to 4240 in decimal, the number of free bytes.
382E	0		
382F	4		Indicates that four files are stored.
