PPX Exchange

Vol. 6 Number 4 Copyright 1982

July/August 1982

Featuring the TI-88

The introduction of the TI Programmable 88 in the June Consumer Electronics Show and the introductory flyer included in the May/June newsletter have brought many questions to the minds of our readers. Here are answers to some of these questions.

Q: When will the TI-88 be available for purchase?

A: In the fourth quarter of 1982.

Q: Where can I buy one?

A: The TI-88 should be available in the same stores where the TI-59 is sold.

Q: Will PPX continue to support the TI-59?

A: Yes, we will fully support the TI-59 and the TI-88 through PPX.

Q: What is the suggested retail price of the TI-88?

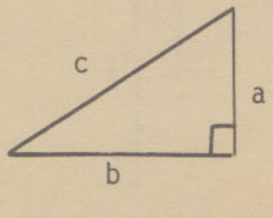
A: $350.00.

Judging from the interest generated by the TI-88's introduction, it is felt that a 'sneak preview' of some of its functions would be appreciated by our members.

EQUATION MODE

Besides an improved Learn Mode, the TI-88 incorporates an Equation Mode which allows the easy insertion of equations. In many cases, the equations may be entered exactly as written. The simple routine discussed here is merely for Equation Mode demonstration. Other features of the TI-88 will be incorporated in future issues of the Exchange.

The Pythagorean Theorem states that the square of the length of the hypotenuse of a right triangle equals the sum of the squares of the lengths of the other two sides. Or,



$$\sqrt{a^2 + b^2} = c$$

The Equation Mode of the TI-88 makes the entry of this equation simple. To enter our equation the following keystrokes will suffice:

Keystroke(s)	Merged Code(s)	Comments
[EQN]		This instructs the calculator to enter the Equation Mode which will allow easy entry of the equation.

continued on page 6

Keeping It Simple

(Editor's Note: Because of the space necessary to explain program development completely, the programming techniques presented in this column will be of little utility save for instructional purposes. The novice programmer can use this information for the development of programming skills, and the more experienced programmer can use the techniques as a refresher course in well structured friendly programming.)

In this, our first attempt at "Keeping it Simple," a program is developed which will store data in sequential registers beginning at any specified register number. The programming techniques outlined will be label addressing, indirect data register addressing and logical charting of program flow. This program can be of special utility when a programmer needs to store data in sequential data registers and simultaneously save program space.

Our first step after problem definition is to outline the flow of program execution. The flowchart (Fig. 1) illustrates the logical sequence of events that must take place in order for our problem to be solved. The keystrokes associated with each event in the flowchart are given for easy reference.

continued on page 7

Symbolic Linear Simultaneous Equations

Andrew J. Fish Jr. Ph.D.
The University of Hartford
College of Engineering

INTRODUCTION: One of the major applications of computers and calculators is in the design of physical systems. The computer is used to generate a numerical solution to the system equations given specific inputs. If the solution is not what the designer desires, the system parameters are altered and a new solution is generated. This process is repeated until the system behaves as desired. Another approach to using the computer in system design is simply to solve design equations. All of these techniques yield specific numerical solutions to specified system parameters. They do not tell the designer how each system element contributes to the solution under all input conditions. This program approaches this problem by solving the system's linear equations in terms of symbols that represent the system elements. The only requirement is that the group of symbols that are used satisfy

continued on page 2

Symbolic Linear Simultaneous Equations...

(cont'd from page 1)

the axioms of a commutative ring with identity. This means that the symbols may include linear operators such as the differential and delay operators. The printer must be used to print out the codes used in the analysis which follows.

USER INSTRUCTIONS

1) Write the system's equations in matrix format:

$$\begin{bmatrix} 0 \\ \cdot \\ \cdot \\ \cdot \\ 0 \end{bmatrix} = \begin{bmatrix} A_{11} & \cdot & \cdot & \cdot & A_{1j} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ A_{i1} & \cdot & \cdot & \cdot & A_{ij} \end{bmatrix} \begin{bmatrix} X_1 \\ \cdot \\ \cdot \\ \cdot \\ X_n \end{bmatrix}$$

using *symbols* rather than numbers. This system matrix will be referred to as matrix A.

2) Assign an integer to each element of the matrix A, creating matrix B. For the constants 0, 1 and -1, in matrix A, no changes need be made; simply transfer them into matrix B. Assign integers for each of the symbols used in matrix A. For example, C(d/dt) would have two integers assigned. Let's arbitrarily choose a 3 for C and a 2 for (d/dt). Thus, the entry in matrix B for C(d/dt) would be 32. A little bookkeeping is all that is necessary to keep the numbers straight.

3) When Matrix B is complete, read in the cards containing the program and press [2nd] [CMS].

4) Enter the number of columns in the matrix A and press A. Then enter the number of rows in the matrix A and press R/S.

5) Enter the largest positive integer used and press R/S.

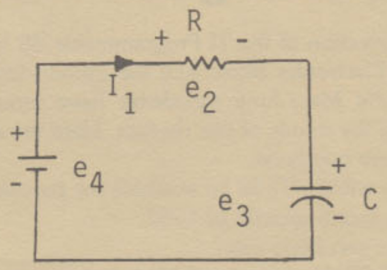
6) The elements of the matrix B are now ready to be entered. To enter the row and column of the entry use the following format: (row.column) and press R/S. Then enter the corresponding integer from matrix B and press R/S again. Continue in this manner until all the elements of Matrix B have been entered. Any element which is identically zero need not be entered if the memories in the calculator were cleared in step 3.

7) When all of Matrix B has been entered correctly, press B to generate the codes which will be used to specify the unknowns in the system.

8) When the listing containing the system codes has been generated, the program is ready to solve the system of equations. The program will equate the i_{th} unknown to a linear combination of the other unknowns. More specifically, the program will equate X_i (i-m) to a linear combination of X_{m+1} through X_n over the integers, where m is the number of rows and n is the number of columns. To calculate this i_{th} unknown, enter i and press C. This will generate a second set of codes which can be used to write down the system's equations.

9) The second set of codes are decoded as follows: The integers in the left hand column of the listing are the codes for the unknowns. The integers in the right hand column are the encoded coefficients of the unknowns. Find the coefficient in the first table of codes produced by looking at the numbers under the arrows. Then replace the encoded coefficient with the entire code above the entry in the table. You may notice that the replacement code does not correspond to any of the original codes in matrix B. It will be necessary to further decode the coefficient in the same manner as before until the coefficient of the unknown contains only codes entered in matrix B. This must be done for each coefficient until the system equation desired is defined.

Example:



The network equations that represent this circuit diagram are:

$$e_2 + e_3 - e_4 = 0$$
$$e_2 = R(I_1)$$
$$I_1 = C(d/dt)e_3$$

The first step is to construct matrix A:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 & 0 & C(d/dt) & 0 \\ R & -1 & 0 & 0 \\ 0 & 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} I_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix}$$

The second step is to assign integers to each element of this matrix creating matrix B. Arbitrarily let the resistor (R) be represented with the integer 4, the capacitor (C) be represented with the integer 3, and the (d/dt) operator be represented with the integer 2, so that the integer for C(d/dt) is 32.

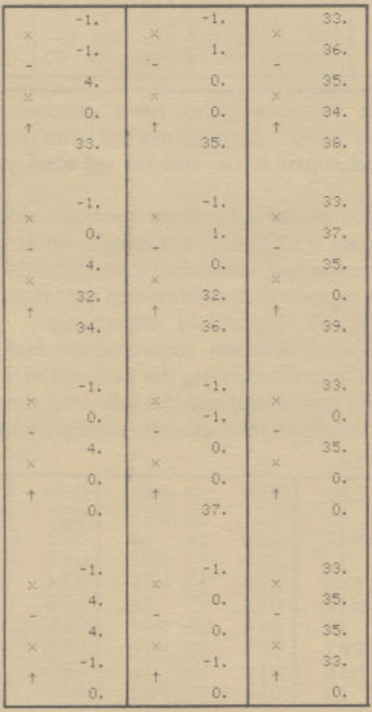
$$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 32 & 0 \\ 4 & -1 & 0 & 0 \\ 0 & 1 & 1 & -1 \end{bmatrix} \begin{bmatrix} I_1 \\ e_2 \\ e_3 \\ e_4 \end{bmatrix}$$

Sample Problem

1) Read in the program from your previously recorded cards.
2) Enter the number of columns (4) and press A.
3) Enter the number of rows (3) and press R/S.
4) Enter the largest positive integer (32) and press R/S.
5) Enter the elements of matrix B by row and column.

ENTER	PRESS
1.1	R/S
-1	R/S
1.3	R/S
32	R/S
2.1	R/S
4	R/S
2.2	R/S
-1	R/S
3.2	R/S
1	R/S
3.3	R/S
1	R/S
3.4	R/S
-1	R/S

If any errors are introduced, simply reenter the row.column and the corresponding entry.

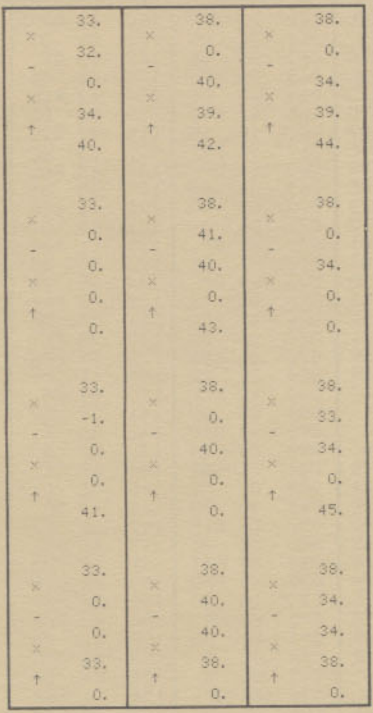


```
       -1.         -1.         33.
 ×           ×           ×
       -1.          1.         36.
 -           -           -
        4.          0.         35.
 ×           ×           ×
        0.          0.         34.
 ↑           ↑           ↑
       33.         35.         38.

       -1.         -1.         33.
 ×           ×           ×
        0.          1.         37.
 -           -           -
        4.          0.         35.
 ×           ×           ×
       32.         32.          0.
 ↑           ↑           ↑
       34.         36.         39.

       -1.         -1.         33.
 ×           ×           ×
        0.         -1.          0.
 -           -           -
        4.          0.         35.
 ×           ×           ×
        0.          0.          0.
 ↑           ↑           ↑
        0.         37.          0.

       -1.         -1.         33.
 ×           ×           ×
        4.          0.         35.
 -           -           -
        4.          0.         35.
 ×           ×           ×
       -1.         -1.         33.
 ↑           ↑           ↑
        0.          0.          0.
```



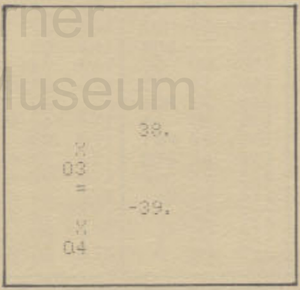
```
       33.         38.         38.
 ×           ×           ×
       32.          0.          0.
 -           -           -
        0.         40.         34.
 ×           ×           ×
       34.         39.         39.
 ↑           ↑           ↑
       40.         42.         44.

       33.         38.         38.
 ×           ×           ×
        0.         41.          0.
 -           -           -
        0.         40.         34.
 ×           ×           ×
        0.          0.          0.
 ↑           ↑           ↑
        0.         43.          0.

       33.         38.         38.
 ×           ×           ×
       -1.          0.         33.
 -           -           -
        0.         40.         34.
 ×           ×           ×
        0.          0.          0.
 ↑           ↑           ↑
       41.          0.         45.

       33.         38.         38.
 ×           ×           ×
        0.         40.         34.
 -           -           -
        0.         40.         34.
 ×           ×           ×
       33.         38.         38.
 ↑           ↑           ↑
        0.          0.          0.
```

6) Press B. This will produce the above codes:
7) In order to find e_3 in terms of e_4 enter 3 and press C.



```
         38.
  X
  03
  =
        -39.
  X
  04
```

which is read: $38e_3 = -39e_4$. To decode this equation, find 39 in the code generated in the listing. It is the tenth entry on the printout. Replace the integer 39 by the expression above it. In this case, 33 × 37 − 35 × 0. And do the same with 38. The equation now becomes:

$$(33 \times 36 - 35 \times 34)e_3 = -(33 \times 37)e_4$$

Now find 33 on the code listing. 33 is replaced by $-1 \times -1 - 4 \times 0 = 1$. 36 is replaced by −1, 35 is replaced by 1, 34 is replaced by 4 ×32, and 37 is replaced by 1. The equation is now in terms of the integers that were used to replace the original symbols. Consequently, replacing these integers by their respective symbols gives the answer:

$$(1 + RC(d/dt))e_3 = e_4$$

I1 and e2 can be determined next by entering the appropriate integer, and pressing C, and decoding the equation.

```
000  76 LBL
001  11   A
002  42 STO
003  06  06
004  91 R/S
005  42 STO
006  07  07
007  91 R/S
008  42 STO
009  00  00
010  91 R/S
011  42 STO
012  01  01
013  59 INT
014  42 STO
015  03  03
016  43 RCL
017  01  01
018  22 INV
019  59 INT
020  65   ×
021  01   1
022  00   0
023  95   =
024  42 STO
025  04  04
026  71 SBR
027  00  00
028  35  35
029  91 R/S
030  72 ST*
031  05  05
032  61 GTO
033  00  00
034  10  10
035  53   (
036  43 RCL
037  03  03
038  65   ×
039  43 RCL
040  06  06
041  75   -
042  43 RCL
043  06  06
044  85   +
045  43 RCL
046  04  04
047  85   +
048  01   1
049  01   1
050  54   )
051  42 STO
052  05  05
053  92 RTN
054  43 RCL
055  02  02
056  42 STO
057  03  03
058  42 STO
059  04  04
060  71 SBR
061  00  00
062  35  35
063  73 RC*
064  05  05
065  42 STO
066  10  10
067  99 PRT
068  69 OP
069  00  00
070  05   5
071  00   0
072  69 OP
073  01  01
074  69 OP
075  05  05
076  43 RCL
077  08  08
078  42 STO
079  03  03
080  43 RCL
081  09  09
082  42 STO
083  04  04
084  71 SBR
085  00  00
086  35  35
087  73 RC*
088  05  05
089  99 PRT
090  49 PRD
091  10  10
092  69 OP
093  00  00
094  02   2
095  00   0
096  69 OP
097  01  01
098  69 OP
099  05  05
100  43 RCL
101  08  08
102  42 STO
103  03  03
104  43 RCL
105  02  02
106  42 STO
107  04  04
108  71 SBR
109  00  00
110  35  35
111  73 RC*
112  05  05
113  99 PRT
114  42 STO
115  11  11
116  69 OP
117  00  00
118  05   5
119  00   0
120  69 OP
121  01  01
122  69 OP
123  05  05
124  43 RCL
125  02  02
126  42 STO
127  03  03
128  43 RCL
129  09  09
130  42 STO
131  04  04
132  71 SBR
133  00  00
134  35  35
135  73 RC*
136  05  05
137  99 PRT
138  49 PRD
139  11  11
140  69 OP
141  00  00
142  06   6
143  00   0
144  69 OP
145  01  01
146  69 OP
147  05  05
148  43 RCL
149  08  08
150  42 STO
151  03  03
152  43 RCL
153  09  09
154  42 STO
155  04  04
156  71 SBR
157  00  00
158  35  35
159  53   (
160  43 RCL
161  10  10
162  75   -
163  43 RCL
164  11  11
165  54   )
166  29  CP
167  67  EQ
168  01  01
169  74  74
170  69 OP
171  20  20
172  43 RCL
173  00  00
174  99 PRT
175  98 ADV
176  98 ADV
177  72 ST*
178  05  05
179  92 RTN
180  76 LBL
181  12   B
182  29  CP
183  01   1
184  42 STO
185  02  02
186  53   (
187  43 RCL
188  02  02
189  75   -
190  43 RCL
191  06  06
192  54   )
193  22 INV
194  67  EQ
195  02  02
196  03  03
197  01   1
198  42 STO
199  09  09
200  61 GTO
201  02  02
202  11  11
203  53   (
204  43 RCL
205  02  02
206  85   +
207  01   1
208  54   )
209  42 STO
210  09  09
211  53   (
212  43 RCL
213  02  02
214  75   -
215  43 RCL
216  07  07
217  54   )
218  22 INV
219  67  EQ
220  02  02
221  28  28
222  01   1
223  42 STO
224  08  08
225  61 GTO
226  02  02
227  36  36
228  53   (
229  43 RCL
230  02  02
231  85   +
232  01   1
233  54   )
234  42 STO
235  08  08
236  71 SBR
237  00  00
238  54  54
239  53   (
240  43 RCL
241  09  09
242  75   -
243  43 RCL
244  02  02
245  54   )
246  67  EQ
247  02  02
248  70  70
249  53   (
250  43 RCL
251  09  09
252  75   -
253  43 RCL
254  06  06
255  54   )
256  67  EQ
257  02  02
258  64  64
259  69 OP
260  29  29
261  61 GTO
262  02  02
263  36  36
264  01   1
265  42 STO
266  09  09
267  61 GTO
268  02  02
269  36  36
270  53   (
271  43 RCL
272  08  08
273  75   -
274  43 RCL
275  07  07
276  54   )
277  22 INV
278  67  EQ
279  02  02
280  87  87
281  01   1
282  42 STO
283  08  08
284  61 GTO
285  02  02
286  89  89
287  69 OP
288  28  28
289  53   (
290  43 RCL
291  08  08
292  75   -
293  43 RCL
294  02  02
295  54   )
296  67  EQ
297  03  03
298  21  21
299  53   (
300  43 RCL
301  09  09
302  75   -
303  43 RCL
304  06  06
305  54   )
306  22 INV
307  67  EQ
308  03  03
309  16  16
310  01   1
311  42 STO
312  09  09
313  61 GTO
314  02  02
315  36  36
316  69 OP
317  29  29
318  61 GTO
319  02  02
320  36  36
321  53   (
322  43 RCL
323  06  06
324  75   -
325  43 RCL
326  07  07
327  54   )
328  77  GE
329  03  03
330  46  46
331  53   (
332  43 RCL
333  02  02
334  75   -
335  43 RCL
336  06  06
337  54   )
338  67  EQ
339  03  03
340  61  61
341  69 OP
342  22  22
343  61 GTO
344  01  01
345  86  86
346  53   (
347  43 RCL
348  02  02
349  75   -
350  43 RCL
351  07  07
352  54   )
353  67  EQ
354  03  03
355  61  61
356  69 OP
357  22  22
358  61 GTO
359  01  01
360  86  86
361  92 RTN
362  53   (
363  43 RCL
364  04  04
365 .55   ÷
366  01   1
367  00   0
368  54   )
369  59 INT
370  85   +
371  01   1
372  95   =
373  65   ×
374  01   1
375  00   0
376  00   0
377  85   +
378  53   (
379  53   (
380  43 RCL
381  04  04
382  55   ÷
383  01   1
384  00   0
385  54   )
386  22 INV
387  59 INT
388  65   ×
389  01   1
390  00   0
391  85   +
392  01   1
393  54   )
394  95   =
395  69 OP
396  01  01
397  69 OP
398  05  05
399  92 RTN
400  76 LBL
401  13   C
402  42 STO
403  03  03
404  42 STO
405  04  04
406  71 SBR
407  00  00
408  35  35
409  73 RC*
410 .05  05
411  99 PRT
412  69 OP
413  00  00
414  04   4
415  04   4
416  69 OP
417  01  01
418  69 OP
419  05  05
420  71 SBR
421  03  03
422  62  62
423  06   6
424  04   4
425  69 OP
426  01  01
427  69 OP
428  05  05
429  43 RCL
430  06  06
431  32 X:T
432  43 RCL
433  07  07
434  22 INV
435  77  GE
436  04  04
437  42  42
438  00   0
439  99 PRT
440  98 ADV
441  92 RTN
442  29  CP
443  43 RCL
444  07  07
445  42 STO
446  04  04
447  69 OP
448  24  24
449  71 SBR
450  00  00
451  35  35
452  73 RC*
453  05  05
454  94 +/-
455  99 PRT
456  69 OP
457  00  00
458  04   4
459  04   4
460  69 OP
461  01  01
462  69 OP
463  05  05
464  71 SBR
465  03  03
466  62  62
467  53   (
468  43 RCL
469  04  04
470  75   -
471  43 RCL
472  06  06
473  54   )
474  67  EQ
475  04  04
476  41  41
477  61 GTO
478  04  04
479  47  47
```

from the Analyst's Desk

• Lem Matteson has noted an error in his program #908181 "SR-52 Program Converter". The correction should be made at location 345 by replacing the instruction [RCL] with [STO]. The program code from locations 338 through 350 should read as follows:

```
338  87 IFF
339  00  00
340  03  03
341  45  45
342  42 STO
343  22  22
344  16  A'
345  42 STO
346  40  40
347  17  B'
348  87 IFF
349  01  01
350  19  D'
```

• For those members who have purchased program #918237 "So Sorry", the program has been revised. Please send your old copies to us, and we will send you the revision.

• A revision is available for those members who have purchased program #918239 "Anagrams". An error has been discovered concerning conflicting data registers. To correct the problem, replace each reference to data registers 4, 5, 6, and 7 with 38, 39, 43, and 41 respectively.

• Member Gene Bahnsen recommends that individuals having difficulty understanding the purpose of the heirarchy register ([HIR] command) use the following program to examine the contents of the heirarchy registers when performing calculations.

```
000  76 LBL
001  11   A
002  82 HIR
003  11  11
004  99 PRT
005  82 HIR
006  12  12
007  99 PRT
008  82 HIR
009  13  13
010  99 PRT
011  82 HIR
012  14  14
013  99 PRT
014  82 HIR
015  15  15
016  99 PRT
017  82 HIR
018  16  16
019  99 PRT
020  82 HIR
021  17  17
022  99 PRT
023  82 HIR
024  18  18
025  99 PRT
026  92 RTN
027  00   0
028  00   0
029  00   0
```

Analyst Desk... *(cont'd from page 4)*

During the execution of calculations from the keyboard or in a program, this subroutine can be implemented to trace the activity in the heirarchy registers. For simplicity we utilized label [A] to begin the program. The user may utilize any label available, and in the absence of a printer, the [PRT] instructions may be replaced with [Pause]. To execute the program code, call the user-defined key from the body of the program.

- We have had a special request for programming assistance in the Minneapolis/St. Paul area. Anyone in this area interested in assisting a fellow TI-59 user please send your name and address to the Exchange Editor. We would like to renew our request for any PPX member interested in tutoring. Our requests in any area are always open to continuing member input.

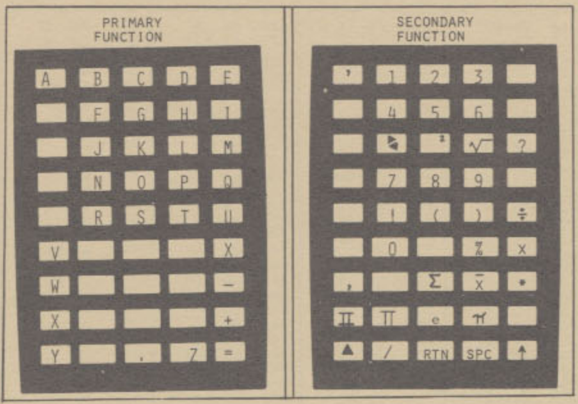
Typewriter PC-100A(C) Required

It is not necessary to decipher print code tables or key in print codes to enable your TI-59 to simulate a typewriter. PPX member, Mr. Michael Mulak has designed a program which allows the user to access the entire sixty-four character print code table by simply pressing one key per character. This program demonstrates the techniques of expanding the user-defined keys, self-editing code, and accessing the printer keys within a program.

Steps 330 through 340 are the key to this program as they utilize the self-editing technique. The program code first repartitions to 399.69 (7 [2nd] [OP] 17) and stores the number 7.1 in register 60. By returning to standard power-up partition, a [SBR] instruction is placed in program location 479. When the program pointer encounters the [SBR] instruction, it returns an error condition to the display. The program has run to the end of the partition and the subroutine instruction is left incomplete. By pressing the key corresponding to your desired character, the appropriate print code is stored for future execution.

To execute the program, key in the program in power-up partition and record on magnetic cards, if desired. Initialize by pressing [2nd] [E'] and the display will return a flashing "20". This indicates that you have a line of twenty characters to fill. After pressing the "typewriter" key corresponding to your first choice, the calculator display will return with a flashing "19". The program will continue to count down to zero and print the entire twenty character line. It will then return another flashing "20" indicating that it is ready for the second line. Should you choose to complete a line prematurely, simply press [Adv] on the printer or [2nd] [Adv] on the calculator. The buffered characters will be printed, and you may continue with the next line. To terminate the program press [LRN] [LRN].

The Checksum printout is shown below to help you in verifying the program code. Also shown below is the keyboard configuration and an example to enable you to turn your TI-59 into a typewriter!



```
                 EXAMPLE


  PRESS     DISPLAY        PRESS     DISPLAY
    )          19            EE          9
    E          18             ÷          8
   RST         17           1/X          7
    A          16             E          6
    (          15           STO          5
  PRINT        14             )          4
   CLR         13             (          3
   STO         12             -          2
    (          11           LOG          1
    )          10           PRD

         TEXAS INSTRUMENTS-59
```

```
000  72 ST*    053  76 LBL    106  81 RST    159  05  5
001  00  00    054  99 PRT    107  76 LBL    160  06  6
002  24 CE     055  00  0     108  52 EE     161  81 RST
003  69 OP     056  81 RST    109  03  3     162  76 LBL
004  20  20    057  76 LBL    110  05  5     163  17 B'
005  43 RCL    058  24 CE     111  81 RST    164  02  2
006  00  00    059  02  2     112  76 LBL    165  81 RST
007  75  -     060  03  3     113  53  (     166  76 LBL
008  02  2     061  81 RST    114  03  3     167  18 C'
009  01  1     062  76 LBL    115  06  6     168  03  3
010  95  =     063  25 CLR    116  81 RST    169  81 RST
011  50 I×I    064  02  2     117  76 LBL    170  76 LBL
012  29 CP     065  04  4     118  54  )     171  19 D'
013  67  EQ    066  81 RST    119  03  3     172  04  4
014  98 ADV    067  76 LBL    120  07  7     173  81 RST
015  61 GTO    068  32 X:T    121  81 RST    174  76 LBL
016  04  04    069  02  2     122  76 LBL    175  27 INV
017  79  79    070  05  5     123  55  ÷     176  05  5
018  76 LBL    071  81 RST    124  04  4     177  81 RST
019  11  A     072  76 LBL    125  01  1     178  76 LBL
020  01  1     073  33 X²     126  81 RST    179  28 LOG
021  03  3     074  02  2     127  76 LBL    180  06  6
022  81 RST    075  06  6     128  61 GTO    181  81 RST
023  76 LBL    076  81 RST    129  04  4     182  76 LBL
024  12  B     077  76 LBL    130  02  2     183  29 CP
025  01  1     078  34 ΓX     131  81 RST    184  07  7
026  04  4     079  02  2     132  76 LBL    185  81 RST
027  81 RST    080  07  7     133  71 SBR    186  76 LBL
028  76 LBL    081  81 RST    134  04  4     187  47 CMS
029  13  C     082  76 LBL    135  03  3     188  08  8
030  01  1     083  35 1/X    136  81 RST    189  81 RST
031  05  5     084  03  3     137  76 LBL    190  76 LBL
032  81 RST    085  00  0     138  81 RST    191  48 EXC
033  76 LBL    086  81 RST    139  04  4     192  09  9
034  14  D     087  76 LBL    140  04  4     193  81 RST
035  01  1     088  42 STO    141  81 RST    194  76 LBL
036  06  6     089  03  3     142  76 LBL    195  49 PRD
037  81 RST    090  01  1     143  91 R/S    196  01  1
038  76 LBL    091  81 RST    144  04  4     197  02  2
039  15  E     092  76 LBL    145  05  5     198  81 RST
040  01  1     093  43 RCL    146  81 RST    199  76 LBL
041  07  7     094  03  3     147  76 LBL    200  67  EQ
042  81 RST    095  02  2     148  94 +/-    201  01  1
043  76 LBL    096  81 RST    149  04  4     202  81 RST
044  22 INV    097  76 LBL    150  06  6     203  76 LBL
045  02  2     098  44 SUM    151  81 RST    204  89  π
046  01  1     099  03  3     152  76 LBL    205  05  5
047  81 RST    100  03  3     153  58 FIX    206  03  3
048  76 LBL    101  81 RST    154  05  5     207  81 RST
049  23 LNX    102  76 LBL    155  05  5     208  76 LBL
050  02  2     103  45 YX     156  81 RST    209  38 SIN
051  02  2     104  03  3     157  76 LBL    210  07  7
052  81 RST    105  04  4     158  59 INT    211  00  0
```

```
212  81 RST   269  78 Σ+    326  06  6    383  00  0
213  76 LBL   270  07  7    327  81 RST   384  00  0
214  39 COS   271  07  7    328  76 LBL   385  00  0
215  05  5    272  81 RST   329  10 E'    386  85  +
216  02  2    273  76 LBL   330  07  7    387  69 OP
217  81 RST   274  79  x̄    331  69 OP    388  20  20
218  76 LBL   275  06  6    332  17  17   389  73 RC*
219  93  .    276  07  7    333  07  7    390  00  00
220  04  4    277  81 RST   334  93  .    391  65  ×
221  00  0    278  76 LBL   335  01  1    392  01  1
222  81 RST   279  57 ENG   336  42 STO   393  00  0
223  76 LBL   280  07  7    337  60  60   394  00  0
224  75  -    281  03  3    338  06  6    395  00  0
225  02  2    282  81 RST   339  69 OP    396  00  0
226  00  0    283  76 LBL   340  17  17   397  85  +
227  81 RST   284  86 STF   341  01  1    398  69 OP
228  76 LBL   285  07  7    342  42 STO   399  20  20
229  65  ×    286  04  4    343  00  00   400  73 RC*
230  05  5    287  81 RST   344  69 OP    401  00  00
231  00  0    288  76 LBL   345  00  00   402  65  ×
232  81 RST   289  87 IFF   346  02  2    403  01  1
233  76 LBL   290  07  7    347  00  0    404  00  0
234  85  +    291  06  6    348  61 GTO   405  00  0
235  04  4    292  81 RST   349  04  04   406  85  +
236  07  7    293  76 LBL   350  79  79   407  69 OP
237  81 RST   294  16 A'    351  76 LBL   408  20  20
238  76 LBL   295  06  6    352  98 ADV   409  73 RC*
239  95  =    296  05  5    353  25 CLR   410  00  00
240  06  6    297  81 RST   354  42 STO   411  95  =
241  04  4    298  76 LBL   355  00  00   412  84 OP*
242  81 RST   299  90 LST   356  01  1    413  21  21
243  76 LBL   300  06  6    357  42 STO   414  01  1
244  50 I×I   301  00  0    358  21  21   415  44 SUM
245  07  7    302  81 RST   359  69 OP    416  21  21
246  02  2    303  76 LBL   360  20  20   417  05  5
247  81 RST   304  37 P/R   361  73 RC*   418  32 X:T
248  76 LBL   305  06  6    362  00  00   419  43 RCL
249  70 RAD   306  02  2    363  65  ×    420  21  21
250  05  5    307  81 RST   364  01  1    421  67  EQ
251  01  1    308  76 LBL   365  00  0    422  04  04
252  81 RST   309  96 WRT   366  00  0    423  27  27
253  76 LBL   310  07  7    367  00  0    424  61 GTO
254  88 DMS   311  05  5    368  00  0    425  03  03
255  05  5    312  81 RST   369  00  0    426  59  59
256  04  4    313  76 LBL   370  00  0    427  69 OP
257  81 RST   314  30 TAN   371  00  0    428  05  05
258  76 LBL   315  07  7    372  00  0    429  47 CMS
259  97 DSZ   316  01  1    373  85  +    430  01  1
260  06  6    317  81 RST   374  69 OP    431  42 STO
261  03  3    318  76 LBL   375  20  20   432  00  00
262  81 RST   319  69 OP    376  73 RC*   433  02  2
263  76 LBL   320  06  6    377  00  00   434  00  0
264  76 LBL   321  01  1    378  65  ×    435  69 OP
265  05  5    322  81 RST   379  01  1    436  00  00
266  07  7    323  76 LBL   380  00  0    437  61 GTO
267  81 RST   324  60 DEG   381  00  0    438  04  04
268  76 LBL   325  06  6    382  00  0    439  79  79
```

Featuring the TI-88... *(cont'd from page 1)*

[2nd] [CEQ] Clears the dedicated Equation Mode memory.

[2nd] [CMS] Clears the contents of the data memories.

[2nd] [Dfn] [A] The new 'Define' function allows the input of variables with alphanumeric prompting. During execution, the user will be prompted in the display of the calculator by "A: 0." This will be the cue for the user to input the value for the variable a by pressing [ENT]. After input, the value will reside in register 000, the numeric equivalent of register A.

[2nd] [Dfn] [B] The code for the input of variable b.

[√‾‾] The new 'unary' square root function allows entry of the function *before* the argument. The square root function will be pending and will not execute until a completing operation such as [+], [−], [÷], [×], or [=] is encountered.

[( ] Opens a set of parentheses to allow the evaluation of $a^2 + b^2$ before the square root function is executed.

[RCL] [A] A The sequence [RCL] [A] instructs the calculator to recall the contents of register A (register 000).

[( )^n] 2 A^2 Appends the integer power 2 to the variable a. The two merged codes, "A" and "2", appear as A^2 in the Equation mode. The calculator interprets this as "recall the contents of register 000 and square the displayed value."

[+] Common addition.

[Rcl] [B] [( )^n] 2 B^2 Recall the contents of register 001, and square the displayed value.

[ )] Completes $a^2 + b^2$.

[=] Completes the pending square root.

[EQN] Leaves the Equation Mode.

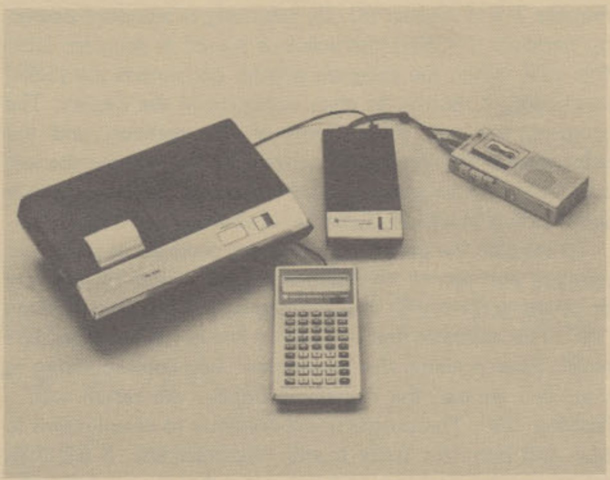
Now that the equation has been entered into the Equation Mode buffer, it can be evaluated by merely pressing the [EVAL] key. After pressing the [EVAL] key, the display will briefly flash "EVAL", and then prompt for the first variable with

A: 0.

To enter the variable a, key in the value and press the User Response key [ENT]. The display will clear and display the prompt for the variable b.

B: 0.

Again, to enter the value of the variable b use the [ENT] key.



The display will clear and the answer will appear. Should the user desire to evaluate the equation with different values for a and b, he can do so by pressing [EVAL]. This will initiate the evaluation for new values of a and b as the equation remains in the dedicated Equation Mode memory.

continued on page 10

Keeping It Simple... *(cont'd from page 1)*

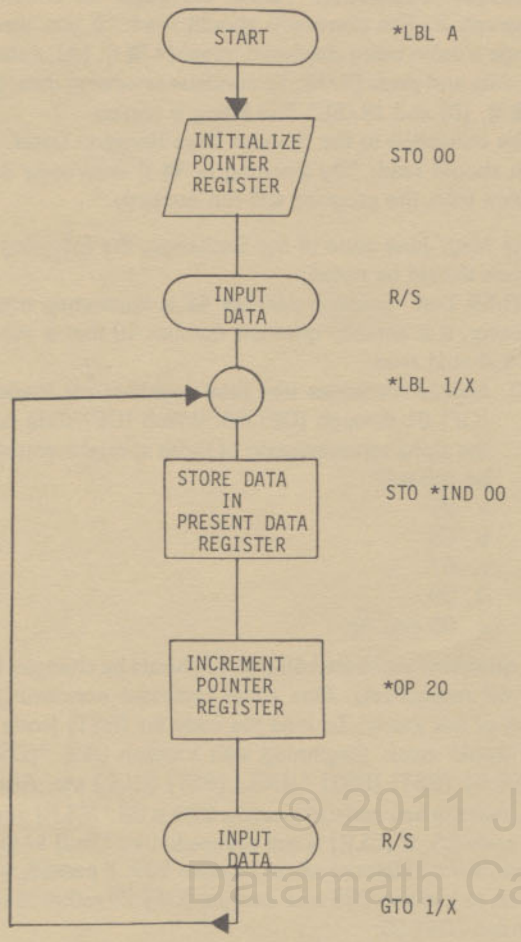

START
*LBL A
INITIALIZE POINTER REGISTER
STO 00
INPUT DATA
R/S
*LBL 1/X
STORE DATA IN PRESENT DATA REGISTER
STO *IND 00
INCREMENT POINTER REGISTER
*OP 20
INPUT DATA
R/S
GTO 1/X

FIG. 1
FLOWCHART

[LRN] Enters the Learn Mode.

*[LBL] [A] Labels are used to mark entry points in the program for later reference. There are ten *user-defined* labels available on the TI-59 [A] through [E']. The choice of the user-defined key [A] allows starting of the program by merely pressing [A].

[STO] 00 The value in the display is stored in data register 00. This value will be the beginning data register number of the data block.

[R/S] This instruction has a dual purpose. If the program is running, it will stop when the [R/S] instruction is encountered. If the program is stopped, it will begin execution at the instruction immediately following the [R/S] command. The [R/S] command is placed here to allow the user to input data before proceeding.

*[LBL] [1/x] This *common* label is referenced later by a [GTO] instruction. The structure created by enclosing a sequence of instructions within *[LBL] [1/x] and [GTO] [1/x] is called a loop and may be executed repeatedly. In our example, the loop is executed each time [R/S] is pressed.

[STO] *[IND] 00 The value in the display is stored in the data register that corresponds to the number in data register 00. The indirect function makes the calculator look at the contents of data register 00 for the number of the data register in which to store the displayed value. This makes it easy to access sequential data registers.

*[OP] 20 The "20" following the [OP] function instructs the calculator to increment the contents of data register 00 by one. The "2" represents addition, and the "0" represents the data register number to be incremented. The alternative to this sequence is 1 [SUM] 00 which not only takes more steps, but leaves an undesired "1" in the display. Saving one step in a program may not be much, but if the capability is available, why not use it? The addition of 1 to register 00 (the pointer into the data block) will cause the next [STO] *[IND] 00 to store the displayed value into the next register in the data block.

[R/S] Stops the program and allows the user to key in the next value to be stored.

[GTO] [1/x] Forces the calculator to begin execution at the next step following [LBL] [1/x].

[LRN] Exits the Learn Mode.

*Denotes [2nd] function

User Instructions

Note that data register 00 is used as the indirect memory

continued on page 8

Precis... *(cont'd from page 11)*

908234I Cumulative Event Recorder

Records and stores exact times of occurrence of a single repeating event relative to the start of a timing period. Repetitions must occur more than one second apart and accuracy decreases for fewer than ten repetitions of the event. Maximum capacity up to 95 repetitions over a maximum time period of about 30 years. Easy counter and response calibration procedures ensure high accuracy. Output in seconds of HMM.SS as desired, for each recorded occurrence of the event. Printer is optional.
Robert C. Becklen, Awarthmore, PA
320 Steps

908235I Number Sorting Descending Order

Ranks up to 99 numbers (39 on the TI-58) in a descending (highest to lowest) order.
William F. Umek, Levittown, PA
87 Steps

918316I Compatibility

This program is like having your own dating service or marriage counselor. Each person answers a series of 24 questions (A-E). He/she can then be compared with up to 43 different persons. A score of 100 is the "perfect" match.
John R. Gibson, Colorado Springs, CO
160 Steps

988060I Sports League Table

Generates a league table for sports. Up to 16 teams can be handled, 8 games per turn. Arrangement of table depends on results of last week's games. Output includes printing of games with names and points, and then output of participating teams in decreasing order from the best at the top of the table. And at last, tabular listing of names and achievements for each team.
Dar Bahatt, TelAviv, Israel
1280 Steps, PC100A, Mod 10

pointer and cannot be used as the beginning register number.

ENTER	PRESS	DISPLAY
Beginning register number, R_{xx}	A	xx
Value to be stored in R_{xx}	[R/S]	Stored value
Value to be stored in $R_{xx}+1$	[R/S]	Second value

Continuing in this manner will store the desired values in successive registers until the highest available data register has been used. The result of trying to store a value in an invalid register will be a flashing display.

Sample Problem

In this example we will store the following values in data registers 1 - 3 and 30 - 32.

Register Number	Value
1	pi
2	e
3	100
30	pi^2
31	e^2
32	7

Enter	Press	Display
1	[A]	1
•[π]	[R/S]	3.141592653
1 [INV] [lnx]	[R/S]	2.718281829
100	[R/S]	100
30	[A]	30
• [π] [x^2]	[R/S]	9.869604401
1 [INV] [lnx] [x^2]	[R/S]	7.389056099
7	[R/S]	7

To verify the results recall the contents of the registers.

```
000  76 LBL     008  00  00
001  11  A      009  25 CLR
002  42 STO     010  09   9
003  00  00     011  02   2
004  91 R/S     012  00   0
005  76 LBL     013  91 R/S
006  35 1/X     014  61 GTO
007  72 ST*     015  35 1/X
```

OOPS!

(Editor's Note: It seems that no matter how we may try, errors creep into our newsletter. Below, you will find corrections to the March/April and May/June issues of the Exchange and a minor correction to the H Addendum. I would like to thank the membership for their kind comments and patience during this editor transition.)

• Program number 998022H "Analysis for Tabex Swimming Pool Chemicals" as listed in the H addendum should be numbered 998051H.

The following corrections to the March/April issue of the Exchange should be noted:

• "Simple Subroutines" error on page 8 column 2 paragraph 2. The correction should read: "If you want to change a value being displayed, press [x ⇄ t], [A], enter the new data and press [R/S]. To continue reviewing data, press [x ⇄ t], [B] and [R/S]." The listing is correct.

• The correction to the "Inverse Days Between Dates" program should read: "By inserting [x ⇄ t] (key code 32) at location 092, the program will run correctly."

In the May/June issue of the Exchange, the following corrections should be noted:

• "TI-59 Test" question number 42 is apparently missing. However, it is actually question number 40 that is missing, and it should read:

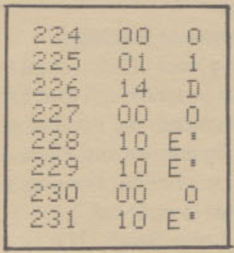
40. Special memories (the print registers) are loaded by [OP] 01 through [OP] 04. Which [OP] code causes the alpha representation of those special memories to be printed?
    a. 05
    b. 06
    c. 07
    d. 09
    e. 05 and 06

The questions numbered 40 and 41 should be changed to 41 and 42 respectively. One other reminder concerning the listing of this article. To load the code for [BST] (code 51), you would enter; (beginning with location 049) •[DSZ] 0 [STO] 51 [BST] [BST] •[DEL] [SST] [RCL] etc. Also, be sure that the key code at location 229 is 20 •[CLR] and not key code 25. If [CLR] is not preceeded by [2nd] in this instance, the conditional test at location 209, if passed, would branch to location 228 and execute [OP] 25 rather than the intended [OP] 20.

• The following corrections should be made to the "There's Gold in Those Guard Digits" article:

1. The second equation on the first page should be
   916323117 + .4437 =
2. The first equation in the left hand column on page three should read:
   1,111,111,111 × 100 + 11 =
3. The first equation in the right hand column on page 3 should have [RCL] 02 inside the parentheses rather than [RCL] 01.
4. The first term in the second equation in the right hand column on page 3 should be [RCL] 01 instead of [RCL] 02.
5. The fourth equation in the right hand column on page 10, (the equation using flag testing) should have a magnitude of X not an X which might be mistaken for a multiply sign between [LBL] [A] and [STO] 00.
6. The last equation in the right hand column of page 10 should have a [+ / −] between the second equal sign and the [RTN].

• The following program code was omitted from the "13 Digit Register Lister Results." The correction is for locations 224 through 231 only. We would like to thank PPX member Michael K. Brady for bringing this error to our attention.



```
224   00   0
225   01   1
226   14   D
227   00   0
228   10  E'
229   10  E'
230   00   0
231   10  E'
```

potpourri

- We are finding that many members are not receiving their newsletters and catalog updates. The reason, more often than not, is that members have had a change of address which has never been forwarded to us. The newsletters and addendums to the catalog are mailed third class bulk rate, and are therefore returned to PPX when the labeled address is not correct. As mailing costs continue to rise, we at PPX are making every effort to refrain from passing these costs on to you, but we need your help. Please notify our membership coordinator at P.O. Box 109 as soon as possible with any change of address.
- We have had many inquiries concerning the F Update. The F Update was combined in the complete Software Catalog published in September 1980. There is no separate F Update.
- We would like to remind our international members that we can accept payment in U.S. Dollars only. All checks and money orders must be drawn on an United States bank or an U.S. correspondent bank. The orders that we receive in any other currency will be returned unfilled, and to avoid the inconvenience and time lag, we suggest that you send all international payments in money orders.
- In recent months, we have noticed a sizable increase in the number of orders which have not included the required $2.00 postage and handling fee and applicable state and local taxes. PPX is required by state and local regulations to pay taxes in all states except Alaska, Montana, New Hampshire, Oregon, and Delaware. We strive at all times to keep our costs down so that we may offer our members, at the lowest possible price, membership in the Exchange, programs, and accessories. When members neglect to pay the postage and handling fees and their state and local taxes, which we must pay regardless, it hurts everyone. Beginning immediately, all orders which do not include the postage and handling fee and all applicable state and local taxes will be returned unfilled requesting that the required amounts be remitted.

PROGRAMMING CORNER

(Editor's Note: This column serves a dual purpose. It informs members of what non-PPX software is currently available and also lists descriptions of programs our members would like to see. The non-PPX software listed in this column is not available from Texas Instruments or PPX, and all inquiries concerning the pricing and availability of these items should be directed to the contacts listed below.)

PROGRAMS WANTED

The program requests for this issue are listed below. All submissions to fill these requests should be postmarked no later than September 15, 1982.

- A program to take data stored in "pseudo" registers of the Math Utilities program 08 (Data Packing) and allow data from any 8 adjacent blocks of the pseudo registers (i.e. one through eight, nine through sixteen, etc. . . .) to be recalled by keying in a three digit code. The eight data items should then be stored in eight normal data registers, for example, R_{53} through R_{60}.
- A program which provides sun/moon conjunction time for any month past or future, coupled with time of average sun/moon conjunction.
- A program to calculate the pull force required to pull electrical cable through conduit.
- A program for New Jersey payroll that includes all necessary deductions for current tax changes, both federal and state.
- A program which will allow one to convert from rectangular or polar coordinates to spherical and then change the angle of view by 30, 45, and 60 degrees.
- A program to compute a complete hydraulic analysis of fire sprinkler systems following the guidelines established by National Fire Protection Association Manual no. 13. The program should incorporate the use of the Hazen-Williams formula and handle as many as 50 loops, or incorporate a method of reducing the number of loops by equal size/length subroutines.
- A program to eliminate the need for no-decompression and if possible, include decompression tables for scuba diving.

SOFTWARE AVAILABLE

- Z-COMP™

Davis-Gilbert Company of San Antonio has designed three Solid State Software™ modules dealing with finance and more specifically with Regulation Z disclosure computations. The programs will calculate:

- Installment loans
- Single payment loans
- True interest
- Add-on rates to APR and vice-versa

Also included are subroutines for the calculation of payments with credit insurance, odd days interest, and changes in the term of the loan and payment amount necessary to produce a specified balloon. Interested persons may contact:

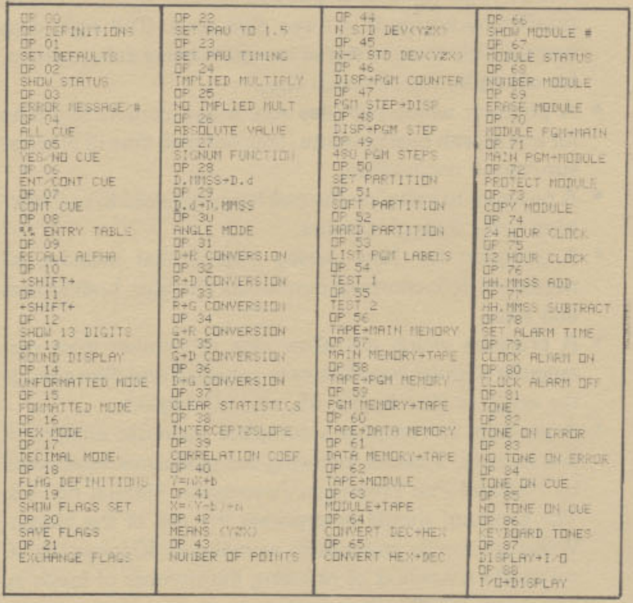
Davis-Gilbert Company
The Landmark, Alamo Plaza
Suite 110
San Antonio, TX 78205
(512) 227-7777

- The Weatherford Company has incorporated several programs into a custom Solid State Software™ module for insurance applications. The programs include:
  - Annuity Planning
  - Mortgage Redemption
  - Illustrations
  - Single and Flexible Premiums
  - Whole Life
  - Universal Life
  - Passenger Auto Ratings

Featuring the TI-88... *(cont'd from page 5)*

OP CODES

The following is a list of the OP codes that are available on the TI-88. The [OP] codes have been grouped into similar categories to enable the user to remember frequently used codes. Also incorporated in the TI-88 is the [INV] [OP] function which will allow a user to view the definition of the requested [OP] code.



```
OP 00
OP DEFINITIONS
OP 01
SET DEFAULTS
OP 02
SHOW STATUS
OP 03
ERROR MESSAGE/#
OP 04
ALL CUE
OP 05
YES/NO CUE
OP 06
ENT/CONT CUE
OP 07
CONT CUE
OP 08
%% ENTRY TABLE
OP 09
RECALL ALPHA
OP 10
→SHIFT→
OP 11
←SHIFT←
OP 12
SHOW 13 DIGITS
OP 13
ROUND DISPLAY
OP 14
UNFORMATTED MODE
OP 15
FORMATTED MODE
OP 16
HEX MODE
OP 17
DECIMAL MODE
OP 18
FLAG DEFINITIONS
OP 19
SHOW FLAGS SET
OP 20
SAVE FLAGS
OP 21
EXCHANGE FLAGS
```

```
OP 22
SET PAU TO 1.5
OP 23
SET PAU TIMING
OP 24
IMPLIED MULTIPLY
OP 25
NO IMPLIED MULT
OP 26
ABSOLUTE VALUE
OP 27
SIGNUM FUNCTION
OP 28
D.MMSS→D.d
OP 29
D.d→D.MMSS
OP 30
ANGLE MODE
OP 31
D→R CONVERSION
OP 32
R→D CONVERSION
OP 33
R→G CONVERSION
OP 34
G→R CONVERSION
OP 35
G→D CONVERSION
OP 36
D→G CONVERSION
OP 37
CLEAR STATISTICS
OP 38
INTERCEPT&SLOPE
OP 39
CORRELATION COEF
OP 40
Y=mX+b
OP 41
X=(Y-b)÷m
OP 42
MEANS (Y&X)
OP 43
NUMBER OF POINTS
```

```
OP 44
N STD DEV(Y&X)
OP 45
N-1 STD DEV(Y&X)
OP 46
DISP→PGM COUNTER
OP 47
PGM STEP→DISP
OP 48
DISP→PGM STEP
OP 49
480 PGM STEPS
OP 50
SET PARTITION
OP 51
SOFT PARTITION
OP 52
HARD PARTITION
OP 53
LIST PGM LABELS
OP 54
TEST 1
OP 55
TEST 2
OP 56
TAPE→MAIN MEMORY
OP 57
MAIN MEMORY→TAPE
OP 58
TAPE→PGM MEMORY
OP 59
PGM MEMORY→TAPE
OP 60
TAPE→DATA MEMORY
OP 61
DATA MEMORY→TAPE
OP 62
TAPE→MODULE
OP 63
MODULE→TAPE
OP 64
CONVERT DEC→HEX
OP 65
CONVERT HEX→DEC
```

```
OP 66
SHOW MODULE #
OP 67
MODULE STATUS
OP 68
NUMBER MODULE
OP 69
ERASE MODULE
OP 70
MODULE PGM→MAIN
OP 71
MAIN PGM→MODULE
OP 72
PROTECT MODULE
OP 73
COPY MODULE
OP 74
24 HOUR CLOCK
OP 75
12 HOUR CLOCK
OP 76
HH.MMSS ADD
OP 77
HH.MMSS SUBTRACT
OP 78
SET ALARM TIME
OP 79
CLOCK ALARM ON
OP 80
CLOCK ALARM OFF
OP 81
TONE
OP 82
TONE ON ERROR
OP 83
NO TONE ON ERROR
OP 84
TONE ON CUE
OP 85
NO TONE ON CUE
OP 86
KEYBOARD TONES
OP 87
DISPLAY→I/O
OP 88
I/O→DISPLAY
```

Programming Corner... *(cont'd from page 9)*

- Homeowner Ratings

and numerous other insurance applications. Although the Module is designed for specific insurance companies, interested persons may contact:

Weatherford Company
1020 South Arroyo
Pasadena, CA 91105
attn: Jerry Conrad
(213) 829-3463
(213) 682-3641

Précis

This column presents the abstracts of some of the new PPX programs which have been recently accepted. The programs were selected by our analysts as being ones that would be of special interest to our members. You can purchase these programs at a cost of $4.00 each. Send your order to: Texas Instruments: PPX Department, P.O. Box 109, Lubbock, TX 79408. Include an additional $2.00 for postage and handling plus applicable state tax.

If you have a need for a specific program, send a note to PPX. There is a chance that the program may have already been written. If it has, we will put the abstract in the next issue of the Exchange. Requests for programs not yet written will be placed in the "Programming Corner" column.

148022I Retirement Savings

Given unchanging rates of interest on savings and inflation of cost of living, the program will find the one of the following three that is not entered: the number of years savings will last if increased only by interest and decreased only by cost of living as increased by inflation, first year's cost of living, and initial saving. Printer is optional.
Stanley Becker, Long Beach, NY
258 Steps

188050I Appreciation Rate

Using the trend-line analysis, the current rate of appreciation, for each of 13 or fewer prices, based on the last 2 to 6 daily prices, is calculated. The inputs are today's price, number of days (2-6) on which the calculation will be based, and the register number for checking a price entry. Outputs are appreciation rate, list of the last 6 prices and the register number and price stored there.
John E. Binns, Stuart, FL
156 Steps, PC-100A, Mod 1

398280I Muller Zeroes of Functions

Finds all roots of up to a 37th order polynomial. Also leaves 240 steps free for an arbitrary function to be solved. No initial guess is required unless the function is user-defined and more than one root (or a complex pair) is needed. Almost quadratical convergence in the vicinity of a root. Finds both real and complex roots even when they are not simple. Printer is optional.
Markus Sveinn Markusson, Gardabar, Iceland
720 Steps, Mod 1

398281I Precision Division 24 to 120 Digit

Dividend, divisor, and quotient can have a maximum of 120 digits. User can select any multiple of 12 digits from 24 to 120. Program has provisions for the verification of the accuracy of the data entry. Quotient returns can be repeated.
Laurence M. Leeds, Sun City, AZ
455 Steps

398282I Area Bounded by One or Two Curves

Will find the area bounded by a curve and the x-axis, even if the value of the function becomes negative in the given interval. The user may also find the area between two curves.
Tomas P. Weithofer, Cincinnati, OH
207 Steps

398283I Precision Division

The divisor and dividend can each be numbers with a maximum of 168 digits; the quotient will always return as 168 digits. The accuracy of data entry can be verified prior to execution; quotient returns can be repeated. May take up to 9 hours.
Laurence M. Leeds, Sun City, AZ
399 Steps

628240I Multistory Frame Response to Earthquake

Gives bending moments in columns and girders of multistory building frames due to earthquake lateral forces. Program can be repeated to add stories when original storage capacity is exceeded. Supplementary program gives U.B.C. earthquake input. Factor Method, a modified slope-deflection procedure, is more accurate than either the portal or cantilever method. Can also be used for wind analysis. Analysis

can start at any level for as many stories as desired.
Paul Fischer, Birmingham, AL
206 Steps

658172I Vertical Refractivity Profile
Calculates Z (height), N (refractivity), N per 1000 feet from significant levels of a Raiosonde Sounding. Inputs of P (mb), T, Td Dewpoint, or T-Td Dewpoint Depression for each level and Z altitude of lowest level, results in outputs either hand-held or printed; of P (mb), Z (ft), N (Refractive Index), and N per 1000 feet for each level. Handles up to 17 significant levels, and the printer is optional.
W.B. Hollis, Seattle, WA
399 Steps

668190I Design of Round Ducts Under Internal Neg. Pressure
Will Calculate the required shell thickness, given the internal negative pressure. Will also calculate the required ring stiffeners, spacing, and fastening. Reference Round Industrial Duct Construction Standards, A.M.A.C.N.A., 1977. Written by Dr. Michael C. Soteriades.
Curtis W. Olsen, Milwaukee, WI
771 Steps

698035I Buckling Pressure of Isotropic Cylinders
Based on elastic theory, predicts the buckling pressure of cylinders fabricated of isotropic materials subjected to either hydrostatic or lateral external pressure. Lateral buckling pressure is equivalent to hydrostatic except for cylinders having a "Z" factor less than 70. Both types of pressure capability and the stress levels associated with each is printed. The purpose of the stress level is to enable the user to ascertain that pressure stress is less than the proportional limit of the material and should therefore present valid results. Additional analysis must be employed to predict capability in the inelastic stress range.
Robert S. Savage, Huntsville, AL
390 Steps, PC100A

708012I Woodwinds Acoustics Formulae
Produces dimensions for proper winds construction for needed staples, bocals or damaged necks for oboes, bassoons and saxophones (respectively) by students or repairmen. Also calculates the hole correction length for flutes and woodwind sound radiation properties. Printer is optional.
Ross D. Litman, II, Ft. Wainwright, AK
157 Steps

718010I Lighting Calculations (Zonal Cavity Method)
By inserting the height, length and width of the cavity, the CR (cavity ratio) is computed. Using the CR and manufacturers' fixture charts, the following data is inserted: L/F (lumens per fixture), LLd (lamp lumen depreciation), LLD (lamp dirt depreciation), CU (coefficient of utilization), and I (illumination level desired). Calculates the exact number of fixtures required. The actual number of fixtures is then inserted and the exact resultant illumination is calculated. English or metric units may be used.
Arthur Belefant, Melbourne Beach, FL
186 Steps, PC100A

738029I Lithium Bromide
Calculates the relationship between the solution vapor pressure, temperature, and enthalpy for a given concentration of lithium bromide (LiBr) salt in water, as used in domestic and industrial refrigeration units. The range covered is 32 to 350 degrees Farenheit, 45% to 70% (weight) LiBr, and pure water. The correlating equations used are those of L.A. McNeely (ASHRAE Transactions, 1979, Part I, P. 413; ASHRAE Handbook of Fundamentals, 1979), published by the American Society of Heating, Refrigerating and Air-conditioning Engineers, Inc.
Marcel J.P. Bogart, Whittier, CA
221 Steps

778042I Bearing Rotation
Given a base line bearing in one system and a bearing of the same base line in another system, computes the angle difference, between the two systems. Then using the computed angle difference, any bearing in the first system may be converted to a bearing in the second system. An alternative method permitted is to enter a rotation angle and all subsequent bearings entered are rotated by the entered angle.
Joseph D. Canning, Boise, ID
253 Steps, Mod 4

788065I Galactical/Equatorial Conversions
Converts Galactic coordinates to equatorial coordinates and vice-versa, for astronomy calculations.
John J. Garner, Grand Portage, MN
390 Steps

798061I Printing Price "20"
Collects labor and rates from up to 20 cost centers. It sums material cost from each area and computes labor hours, labor costs, subtotal (materials and labor), and customer charge (subtotal and profit). It additionally computes percent of job that is labor cost and the unit cost. Provides user prompting.
David G. Vequist, Pittsburg, KS
250 Steps, PC-100A

908233I Histogram and Bar Graph Plotter
Makes a single or multitape histogram and bar graph for up to 49 input points. Any value between the upper and lower vertical limits can be specified as the base for plotting.
W.W. Buechner, Arlington, MA
400 Steps, PC-100A

continued on page 7

ADDRESS CHANGES

In order to ensure uninterrupted service, please submit address changes to PPX at lease six weeks prior to the effective date of the change. Send your name, membership number, old and new addresses to:

PPX
P.O. Box 109
Lubbock, TX 79408

TI-59 Programming Seminars

There may be a seminar coming to your area. These seminars are open to anyone with a TI-59 regardless of programming background. The seminars provide both beginning and intermediate programming training on the TI-59 in a "hands on" fashion. Tuition for the two day class is $150.00 per person. This includes the instruction, workbook, and luncheon for the two days. You should supply your own TI-59. To register send your check for $150.00 payable to Texas Instruments to:

TI-59 Seminar
Texas Instruments
P.O. Box 10508 MS 5820
Lubbock, TX 79408

If you have any further questions regarding the seminars or if you would like information on setting up a company seminar, please contact Professional Calculator Division at 806—741-2202. The schedule of the upcoming seminars is listed below.

Seminar Dates	**Location**
August 5-6	Miami, FL
August 12-24	Omaha, NE
August 19-20	Houston, TX
August 23-24	Pittsburg, PA
September 2-3	Cincinnati, OH
September 7-8	San Francisco, CA
September 13-14	Seattle, WA
September 16-17	Sacramento, CA
September 30-October 1	Kansas City, KS
October 5-6	Chicago, IL
October 11-13	Detroit, MI
October 14-15	Salt Lake City, UT
October 21-22	Milwaukee, WI
October 25-26	Montreal, Canada
October 28-29	Rochester, NY
November 4-5	Albany, NY
November 11-12	Philadelphia, PA
November 18-19	Birmingham, AL
November 22-23	Akron, OH
December 2-3	Indianapolis, IN
December 9-10	New York City, NY

Membership Renewals

Is your membership about to expire? To ensure that you will miss no newsletters, catalogs, or ordering privileges, check the renewal table to find out if your membership will expire soon. (If your number is not included in the range of the table, it is not time for you to renew). The next issues of the Exchange will list additional renewal dates.

A renewal card and reminder will be sent to each member before the time to renew. Return the card promptly to PPX with your check or money order for $20.00. Please do not procrastinate in returning your renewal material as our membership coordinator must remove delinquent members from our computer listing. Be sure to include your membership number on both your card and your check and mail to: Texas Instruments PPX Department, P.O. Box 109, Lubbock, TX 79408.

Membership Number	**Renewal Date:**
909350 - 910093	September 30
920097 - 920478	September 30
927573 - 928026	September 30
932168 - 932308	September 30
910094 - 910895	October 31
920479 - 928270	October 31
928027 - 928270	October 31
932309 - 932935	October 31
950001 - 958000	October 31

The PPX Exchange is published bimonthly and is the only newsletter published by Texas Instruments for TI-59 owners. Members are invited to contribute articles and items of general interest to other TI-59 users. Authors of accepted feature articles for the newsletter will receive their choice of either a one year complimentary PPX membership or a Solid State Software^{TM} module. Please double-space and type all submissions, and forward them to:

Texas Instruments, PPX
P.O. Box 53
Lubbock, Texas 79408
Attn: PPX Exchange Editor

TEXAS INSTRUMENTS
INCORPORATED
PPX • P.O. Box 53 • Lubbock, Texas 79408
U.S. CALCULATOR PRODUCTS DIVISION

ADDRESS CORRECTION REQUESTED