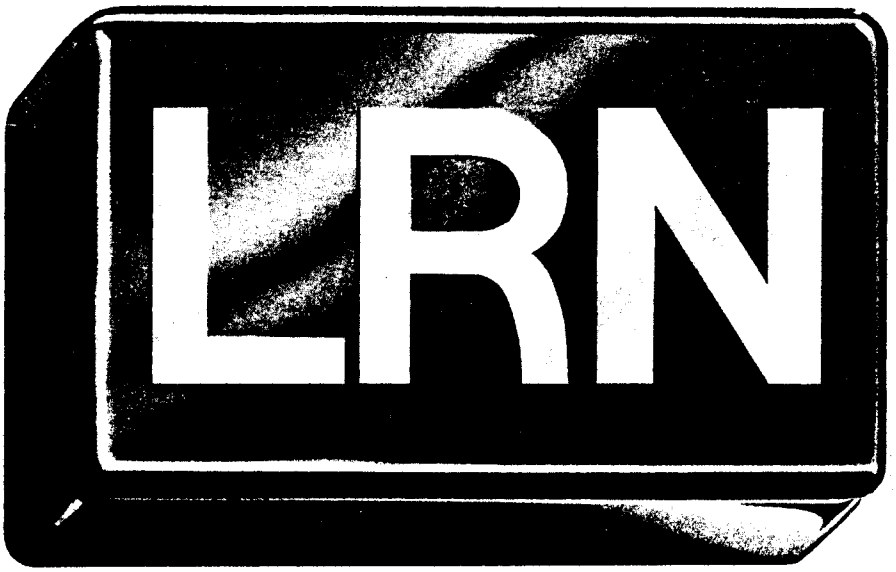


* T I P P C N O T E S * V6N4-5, 1981

****

NEWSLETTER OF THE TI PROGRAMMABLE CALCULATOR CLUB.

9213 Lanham Severn Road, Lanham, Maryland 20801 USA.

To accomodate the members who are constantly asking for a more distinguished heading, here is, in my modest opinion, a most appropriate one. If it weren't for this one, single key, our calculators would just be ordinary SR-51's. The other "hidden" symbolism is that we should constantly be reminded to LEARN more about our marvelous machines and how to program them more effeciently.

The highlight of this issue is the GRAPHICS MODE. Two German calculator fans and an Austrian friend pooled their resources and transformed a strange quirk into something immensely useful. I hope that this new knowledge will give rise to many PRACTICAL plotting programs. With an effective increase of three of the resolution it should now be possible to see so much more detail of curves and graphs.

Richard Snow was so elated about the new technique that it obviously triggered one of his "patriotic" nerves: he produced a demonstration program that draws a finely detailed Stars and Stripes in 45 sec.

IN THIS ISSUE:

Lars Hedlund, of the Swedish Programbiten newsletter, has published a list of proposed routines for our own module. This list is a combination of the wishes of members from our own TI PPC Club and from members from the Programbiten club:

1. *A routine to access the Fast Mode.(maybe also a routine to access Graphics Mode ?)*
2. *A code converter for numbers between -9999 and 99999.*
3. *The Snow Superplotter.*
4. *HIR. Prompted HIR loading of alpha code.*
5. *Labels to direct addresses conversion routine.*
6. *Regression analysis.*
7. *Data register listing with alpha.*
8. *Alphabetical sort routine.*
9. *Fast digital sort routine, ascending and descending.*
10. *13-Digit register routine.*
11. *Packing and unpacking of registers by groups.Clearing by groups. Storing by groups.*
12. *Moving of program segments, entire banks.*
13. *Matrix calculation. Similar to ML-02 and ML-03, but without PRT or ADV commands.*
14. *Complex Determinants.*
15. *Random Number Generator.*
16. *Solutions to equations up to the 4th order.*
17. *Typewriter. An easy way to compose text, and store it on mag cards.*
18. *Prompter. Most commonly used words, printed in any sector. Possibility to form combined prompting words by concatenation.*
19. *Flag test.*
20. *Double precission multiplication and division, up to 20 digits.*
21. *Calculation of polynomials.*
22. *Contour graph plotting.*
23. *Equation systems with three or four unknowns.*
24. *Modulo.(Rest, division)*
25. *A Fast Fourier Analysis.*
26. *Three-dimensional coordinates transformation.*
27. *Base convertion, binary, octal, decimal, hexadecimal.*

This is by no means the final list. Let me know what you think of it. Some of the routines we have already, such as a code converter or superplotter. Others still have to researched Why don't you write me a letter telling me your expertise, if I don't know it already, and which routine(s) you would like to develop or work out some of your,already existing, own. If we all help out, in a short time we will be able to finalize the design. Once done, I publish them all in one single issue. This way you can take a last look at them before the are irrepairably frozen into silicon, and give your comments on them.

Then, and only then, will I start asking for your money. OK ?

In the next issue I plan to bring you a super program: 3-D Luffarschack, Fast Mode. That is Swedish for 3-D Tic-Tac-Toe. The rest you understood. I supose it was written by Lars Hedlund, although no author is mentioned. Only mention is that the print-out of the four levels was supplied by Per-Eric Holmberg and Bjorn Gustavsson. This game beats you all the time. You really have to be on your toes for this one.

My April joke this year wasn't as mean as last year's. I caught people of all nations, except the Flemish and the Dutch, who were forwarned by the meaning of the title GEHEUGENVERLIES,which means "loss of memory" or simply "amnesia."

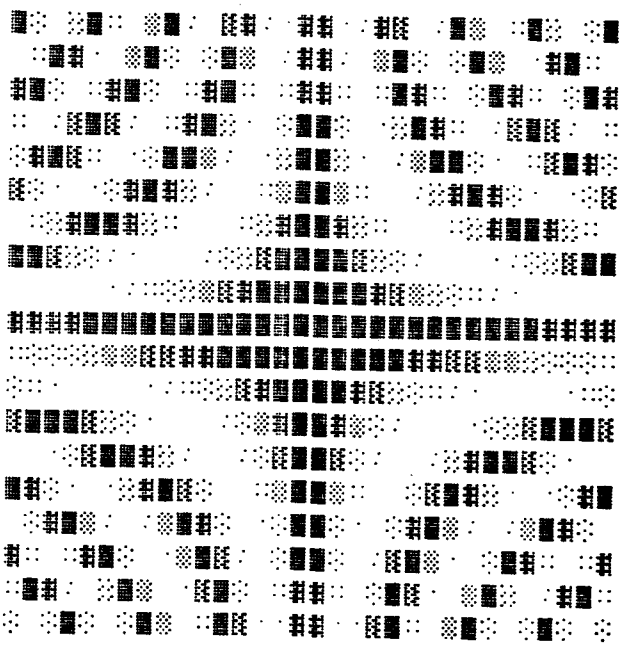
I am still looking for a volunteer to compile all the available and known data on the Fast Mode and write an article about it. I was counting on Palmer O.Hansen doing it, but our friend Palmer has been in the hospital several times lately and is way behind in all his work, especially the one that brings bread on the table. We all wish Palmer a speedy recovery and we look forward to many more of his fine contributions to the NOTES.

I am negociating with Fred Fish to grant me the right to copy his Survival Manual for the ML Library Module. You will then be able to order it directly from the TI PPC Club

I will know for sure next issue.

Maurice E.T. Swinnen

GRAPHICS MODE.- Rather than call it FINE PRINT, as I said in last issue, a more appropriate name for Michael Sperber's discovery is GRAPHICS MODE. In effect the space between lines has been eliminated, just as in Graphics Mode on a line printer. Even the thermal printer on the TI-99/4 has a graphics mode, which consists of simply eliminating the space between lines and the ability to redefine all the ASCII characters to symbols that aid in plotting. To show what can be done with this technique, I have reproduced on the right on this page one of Harald M. Oto's Hoehenliniendiagrams. Mark now that this is done by means of the TI-99/4 home computer on the TI thermal printer. The function here is f(z) = Cos (x*y) with the domain of x ranging from -5 to 5, the domain of y also ranging from -5 to 5 and the z range from -1 to 1.



REDEFINITION OF ASCII CHARACTERS ONLY.

If we just could control each dot on the printing element of the PC100, we would be well on our way to superb plotting. Michael Sperber's discovery brings us one step, and a giant one at that, closer to our goal. His discovery permits us to control three dots in a row at one time and also eliminates the space between lines.

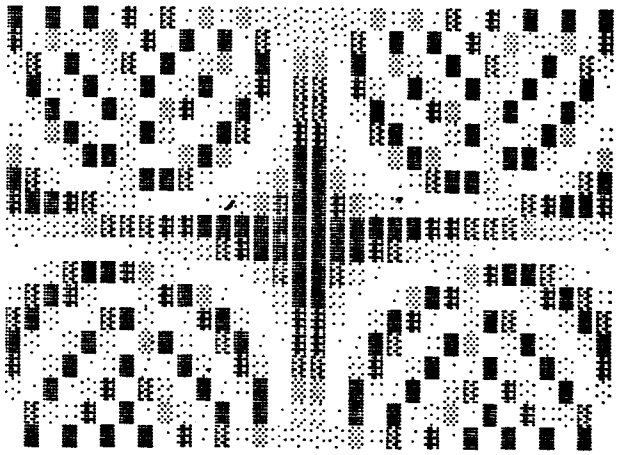
The technique consists of putting a pseudo-code at step 024 of the calculator. I don't know why it doesn't work at any other step. At least I failed to produce it at any other step. You write in your program, in LRN mode, starting at step 024: SUM IND 80 and fill all the steps up to and including 031 either with NOPs or with zeros. Then in keyboard mode you press:

GTO 024 10 OP 17 (4 OP 17 in the TI-58)

CLR PGM 19 SBR 045 P/R LRN INS LRN

RST CLR 6 OP 17 (2 OP 17 in the TI-58)

The INS seems to put the pseudo-code at step 024, forcing the printer to print only part of any character. Somewhere else in this issue you will find Richard Snow's rendition of the PRINT CODE TABLES. In it you can see what the printer does to each of the characters.



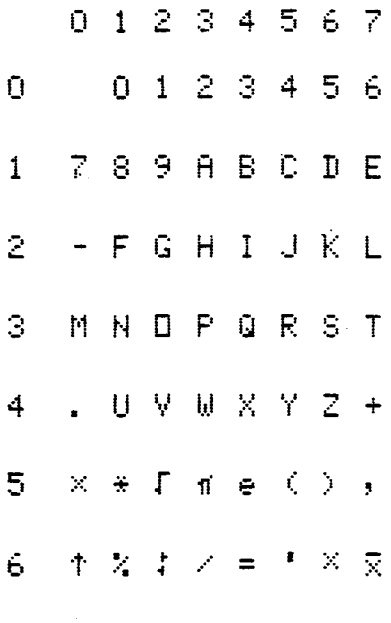
FULL GRAPHICS MODE.

Exploiting this new technique, Richard also gave us a beautiful drawing of the Stars and Stripes flag, never before possible on the PC100. It does this "impossible" something in less than 45 seconds !

And the master himself, Michael Sperber, presents his PLOT 60, a multi-function plotting program. Several examples are given throughout this issue. I hope this new technique will enable us to create a multitude of new and practical applications.

Michael, who lives in Fuerth, West Germany, developed this technique from a "quirk" discovered by his compatriot Gerald Schlueter.(Zeichnen mit dem Printer, von Gerald Schlueter) Together with Johann Berger from Austria, the trio developed everything over a period of several months, all by correspondence.

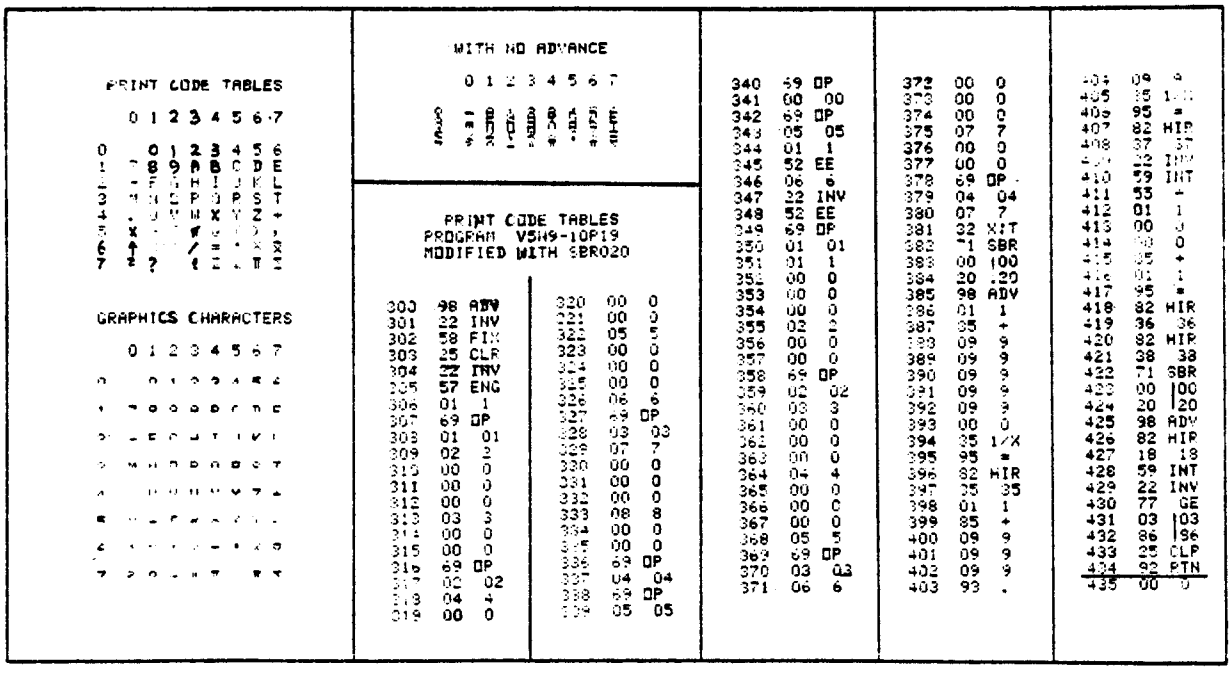
GRAPHICS MODE.- In order to see the effect Graphics mode has on the different print-table characters, Richard Snow modified the Print Code Tables program from v5n9/10p19 with SBR 020, such that it will be executed in Graphics Mode.



On the left is the table as it now prints in Normal Mode, while on the right as printed in Graphics Mode. The new Print Code Tables program listing is given below. The reason I give the two resulting tables full size above is, that after I saw it reduced 50% I found it rather illegible.

Note also that Richard added an ADV at step 300. If you put a NOP there, you will get the result as shown right above the three-line text at the beginning of the listing. Everything runs together.

To key in the Print Code Tables program, first read in PLOT 60. Then key GTO 300 LRN and start keying in the Print Code Tables program. Next initialize PLOT 60 as indicated.(GTO 024 10 OP 17....) Subsequently run the Print Code Tables program by SBR 300.



PRINT CODE TABLES

GRAPHICS CHARACTERS

WITH NO ADVANCE

PRINT CODE TABLES
PROGRAM V5N9-10P19
MODIFIED WITH SBR020

```
300  98 ADV      320  00   0
301  22 INV      321  00   0
302  58 FIX      322  05   5
303  25 CLR      323  00   0
304  22 INV      324  00   0
305  57 ENG      325  00   0
306  01  1       326  06   6
307  69 DP       327  69 DP
308  01  01      328  03  03
309  02  2       329  07   7
310  00  0       330  00   0
311  00  0       331  00   0
312  00  0       332  00   0
313  03  3       333  08   8
314  00  0       334  00   0
315  00  0       335  00   0
316  69 DP       336  69 DP
317  02  02      337  04  04
318  04  4       338  69 DP
319  00  0       339  05  05
```

```
340  69 DP       372  00   0
341  00  00      373  00   0
342  69 DP       374  00   0
343  05  05      375  07   7
344  01  1       376  00   0
345  52 EE       377  00   0
346  06  6       378  69 DP
347  22 INV      379  04  04
348  52 EE       380  07   7
349  69 DP       381  32 XIT
350  01  01      382  71 SBR
351  01  1       383  00 |00
352  00  0       384  20 |20
353  00  0       385  98 ADV
354  00  0       386  01   1
355  02  2       387  85   +
356  00  0       388  09   9
357  00  0       389  09   9
358  69 DP       390  09   9
359  02  02      391  09   9
360  03  3       392  09   9
361  00  0       393  00   0
362  00  0       394  35 1/X
363  00  0       395  95   =
364  04  4       396  82 HIR
365  00  0       397  35  35
366  00  0       398  01   1
367  00  0       399  85   +
368  05  5       400  09   9
369  69 DP       401  09   9
370  03  03      402  09   9
371  06  6       403  93   .
```

```
404  09   9
405  35 1/X
406  95   =
407  82 HIR
408  37  37
409  22 INV
410  59 INT
411  55   ÷
412  01   1
413  00   0
414  00   0
415  85   +
416  01   1
417  95   =
418  82 HIR
419  36  36
420  82 HIR
421  38  38
422  71 SBR
423  00 |00
424  20 |20
425  98 ADV
426  82 HIR
427  18  18
428  59 INT
429  22 INV
430  77  GE
431  03 |03
432  86 |86
433  25 CLR
434  92 RTN
435  00   0
```

PLOT 60.- by Michael Sperber. This program allows you to plot simultaneously any number of functions with a resolution of 60 dots across the paper tape , as opposed to the normal resolution of 20 characters across the width of the paper. You may furthermore predetermine the number of paper tapes you are going to use to plot the functions. You will have to glue the paper tapes one next to the others to obtain the entire picture. The more tapes used, the longer the plotting time, of course. Also, the more functions you want to plot simultaneously, the longer the execution time.

Michael sent me at least eight versions of his PLOT 60. The one reproduced here works as a general plotter and is reasonably fast. It is a good compromise among the whole group. Others may be shorter, faster or have some special advantage. But there is in reality very little difference among the them. And they all have to be initialized from the keyboard, once you have read them into user memory. The initialization sequence is : (the ML module should be in place)

GTO 024 10 OP 17 CLR PGM 19 SBR 045 P/R LRN INS LRN RST CLR 6 OP 17

or press 4 OP 17 and 2 OP 17 respectively on the TI-58.

Next you enter the functions you want to plot. Press GTO 224 LRN. At the end of this article you will find all the rules that apply to entering those functions. Let's just use an example here, to see how it is done. Suppose you want to enter the sine, cosine, sine + cosine, sine X cosine functions and the X-axis. The key sequence would be :

SIN A COS A SIN + RCL 05 COS = A SIN X RCL 05 COS = A CLR D GTO 224

As you can see, each function is simply called (SIN, COS, what have you) as the x-value is in the display at the moment of calling. If that value is needed again, such as in "SIN X COS", the second time you need the x-value you'll find it in R05. End your LAST function with D and end the whole routine with GTO 224.

Now press LRN again, to go out of that mode and enter the functions parameters through E. In our example that would be as follows:

Enter the number of points to be plotted	81	E
Enter Ymin	1.5 +/- ...	R/S
Enter Ymax	1.5	R/S
Enter Xo, the x-starting point	0	R/S
Enter delta-x, the increment in x	4.5	R/S
Enter the number of paper tapes	1	R/S
Start the actual plotting		R/S

If you entered more than "1" for the number of paper tapes, you'll see a "1", then a "2", a "3", etc. printed on each successive tape. It will help you compose the entire picture at the very end.

Besides the ML module, Michael suggests to use other modules as well:

Master : PGM 19 SBR 045
Statistics: PGM 14 SBR 024
Math/Utilities: PGM 06 SBR 099
Leisure: 53 STO 53 0 PGM 21 SBR 331
RPN simulator: PGM 03 SBR 529

And be sure that the display is clear when pressing P/R, otherwise it won't work.

Register use in this program is as follows:

R00	n (counter	R07	Position of actual Y in OP1,2,3,4
R01	Ymin	R08	number of tapes
R02	delta-Y	R09	OP 01
R03	Xo	R10	OP 02
R04	delta-X	R11	OP 03
R05	X (actual value)	R12	OP 04
R06	n (save)	R13	Print code and OP of actual Y.

As you can see, any register equal to or higher than 14 may be used for other purposes in your functions.

The t-register may also be used in your definitions of functions, but bear in mind that the t-reg is cleared each time A or D is called, so you will have to saveguard its contents each time just before those calls.

CONTINUED ON NEXT PAGE

Plot 60, continued.

The rules (nothing is definite, so find some more of them if you care) are:

1. The program should be keyed in and recorded on one card side.
2. The number of functions you may define is unlimited.
3. When you enter functions, the x-value is in the display. When you need the x-value again in the same function, it can be found in register 5.
   When ending your functions with A or D, the value of Y = F(x) must be in the display.
4. DO NOT USE ANY COMMON LABELS IN YOUR FUNCTIONS.
5. Do not use instructions such as CMS, RST, ADV, PRT, OP 05.
6. After initializing the program you may not call any user-defined label except A, D or E. Calling any other one sends your calculator to never-never land, from which it can be called back only by turning the whole mess off and on again.
7. Values of Y which exceed the limits if Ymin and Ymax are not plotted.
8. In case functions cross or touch each other, only the earlier- defined function will be plotted.
9. With the TI-59 any partitioning between and including 2 OP 17 and 9 OP 17 is allowed. With the TI-58 this becomes 2 OP 17 and 3 OP 17.
10. Use only registers equal to or higher than 14.
    Use the t-reg if needed, but safeguard its contents before each A or D call.
11. You may use any step equal to or higher than 224 to start writing your functions.

A few closing remarks, mainly the words of Richard Snow: The initializing process really clobbers steps 024 to 032. The INS step does insert a step, although you can't tell from the display at the time. Although step 024 displays a "25", it is definitely NOT a CLR instruction. It is shown as a single zero in a program listing. The instruction is apperently a hexadecimal pseudo-code.

The OP 05 st step 021 starts printing characters, but the pseudo-code interrupts the printer before the complete characters are printed. The program then jumps to step 000. This explains the RTN at that step. Instead of returning directly to the subroutine call, the program first returns to step 025, then finally to the main program.

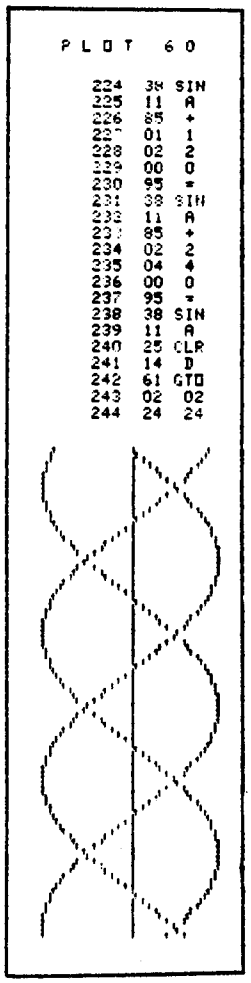
Since the pseudo-code is responsible for modifying characters, rather than a calculator mode, other OP 05 instructions put into the same program will print normally. The program can therefore print a mixture of high resolution graphics and normal alphanumeric characters, be it on different lines.

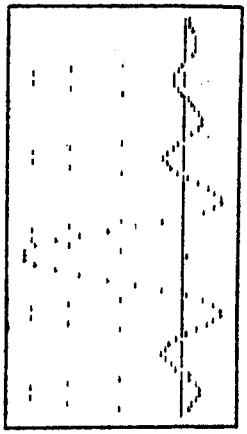
```
000  92 RTN
001  76 LBL
002  15 E
003  42 STO
004  06 06
005  92 RTN
006  61 GTO
007  01 01
008  83 83
009  76 LBL
010  11 A
011  61 GTO
012  00 00
013  33 33
014  76 LBL
015  14 D
016  11 A
017  61 GTO
018  01 01
019  35 35
020  25 CLR
021  69 OP
022  05 05
023  68 NOP
024  74 SM*
025  80 80
026  02 2
027  68 NOP
028  68 NOP
029  68 NOP
030  68 NOP
031  68 NOP
032  22 INV
033  58 FIX
034  75 -
035  43 RCL
036  01 01
037  95 =
038  55 ÷
039  01 1
040  32 X:T
041  43 RCL
042  02 02
043  65 ×
044  77 GE
045  01 01
046  30 30
047  29 CP
048  22 INV
049  77 GE
050  01 01
051  30 30
052  04 4
053  85 +
054  09 9
055  75 -
056  59 INT
057  42 STO
058  13 13
059  95 =
060  65 ×
061  05 5
062  42 STO
063  07 07
064  85 +
065  59 INT
066  94 +/-
067  44 SUM
068  07 07
069  95 =
070  65 ×
071  03 3
072  95 =
073  52 EE
074  59 INT
075  67 EQ
076  00 00
077  87 87
078  32 X:T
079  02 2
080  22 INV
081  67 EQ
082  00 00
083  90 90
084  94 +/-
085  85 +
086  02 2
087  07 7
088  95 =
089  65 ×
090  93 .
091  00 0
092  01 1
093  95 =
094  44 SUM
095  13 13
096  73 RC*
097  13 13
098  55 ÷
099  43 RCL
100  07 07
101  22 INV
102  28 LOG
103  33 X²
104  52 EE
105  22 INV
106  57 ENG
107  42 STO
108  07 07
109  95 =
110  88 DMS
111  82 HIR
112  18 18
113  59 INT
114  39 COS
115  22 INV
116  67 EQ
117  01 01
118  30 30
119  43 RCL
120  07 07
121  65 ×
122  43 RCL
123  13 13
124  22 INV
125  59 INT
126  95 =
127  74 SM*
128  13 13
129  25 CLR
130  29 CP
131  43 RCL
132  05 05
133  92 RTN
134  43 RCL
135  09 09
136  69 OP
137  01 01
138  43 RCL
139  10 10
140  69 OP
141  02 02
142  43 RCL
143  11 11
144  69 OP
145  03 03
146  43 RCL
147  12 12
148  69 OP
149  04 04
150  71 SBR
151  00 00
152  20 20
153  71 SBR
154  02 02
155  13 13
156  43 RCL
157  04 04
158  44 SUM
159  05 05
160  05 5
161  97 DSZ
162  00 00
163  01 01
164  30 30
165  98 ADV
166  25 CLR
167  43 RCL
168  08 08
169  99 PRT
170  43 RCL
171  02 02
172  44 SUM
173  01 01
174  98 ADV
175  71 SBR
176  02 02
177  04 04
178  97 DSZ
179  08 08
180  00 00
181  00 00
182  81 RST
183  42 STO
184  01 01
185  42 STO
186  02 02
187  92 RTN
188  22 INV
189  44 SUM
190  02 02
191  92 RTN
192  42 STO
193  03 03
194  92 RTN
195  42 STO
196  04 04
197  92 RTN
198  42 STO
199  08 08
200  94 +/-
201  22 INV
202  49 PRD
203  02 02
204  43 RCL
205  06 06
206  42 STO
207  00 00
208  43 RCL
209  03 03
210  42 STO
211  05 05
212  32 X:T
213  00 0
214  42 STO
215  09 09
216  42 STO
217  10 10
218  42 STO
219  11 11
220  42 STO
221  12 12
222  32 X:T
223  92 RTN
```

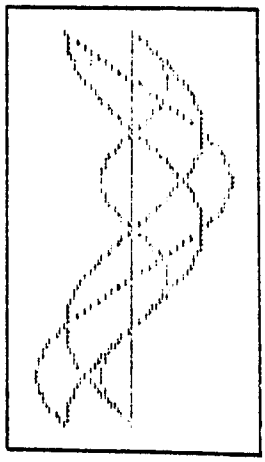
```
P L O T  6 0
AFTER INITIALIZATION
000  92 RTN
001  76 LBL
002  15 E
003  42 STO
004  06 06
005  92 RTN
006  61 GTO
007  01 01
008  83 83
009  76 LBL
010  11 A
011  61 GTO
012  00 00
013  33 33
014  76 LBL
015  14 D
016  11 A
017  61 GTO
018  01 01
019  35 35
020  25 CLR
021  69 OP
022  05 05
023  68 NOP
     0
025  92 RTN
026  42 STO
027  17 17
028  68 NOP
029  69 OP
030  68 68
031  97 DSZ
032  68 68
033  22 INV
034  58 FIX
035  75 75
036  43 RCL
037  01 01
038  95 =
```

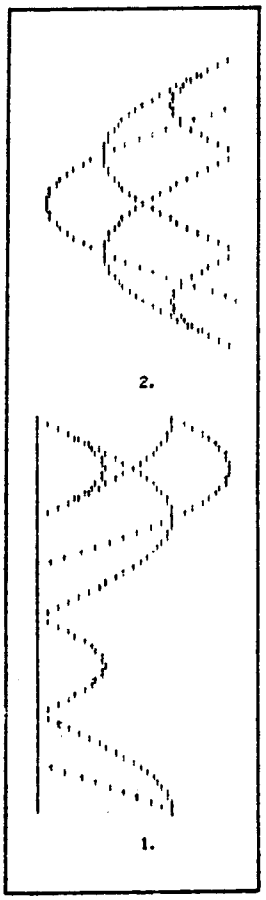
Plot 60, continued.

Here then are a few examples of what can be done with Plot 60. The leftmost one even has the necessary keystrokes starting at step 224. The others are simply demonstrations of curves possible with this multi-plotting program. Note the nice sine-cosine curves plotted on two tapes.









PATENTS.- Two more patents have shown up that may be of interest to our club members:
------- The first one is # 4,006,455, FEB.1, 1977, ERROR CORRECTION SYSTEM IN A PROGRAMMABLE CALCULATOR, Inventor: Stephen P. Hamilton, Garland TX. Filed Oct, 10, 1975. The main subject in this patent applications is the TMC-0594 Magnetic Card Interface Chip (Integrated Circuit to the layman, chip to the cognoscenti)
In the description of the invention itself, the background information, the inventor mentions patent application # 622,280 (the one for this patent was 622,288) filed at the even (same) date by the same author. It concerns a programmable calculator. We have not been able yet to lay our hands on this last patent. We don't even know if it has been granted, but we surmise it has. We don't know the patent number of it, however. We only know its application number.

The second patent we received is a thick book: # 3,900,722, Aug. 19, 1975, Inventors: Michael J. Cachron , Richardson and Charles P. Grant, Dallas, TX, application # 397,060, Sept.13, 1973. MULTI-CHIP CALCULATOR SYSTEM HAVING CYCLE AND SUBCYCLE TIMING GENERATORS. If I am not mistaken, this is the patent for the SR-51, a non-programmable scientific calculator.

STARS AND STRIPES.- by Richard Snow. According to Richard it was written only as a demonstration of the graphics capability of the PC100, using the newly discovered pseudo-code which interrupts the printing in the middle of a line.

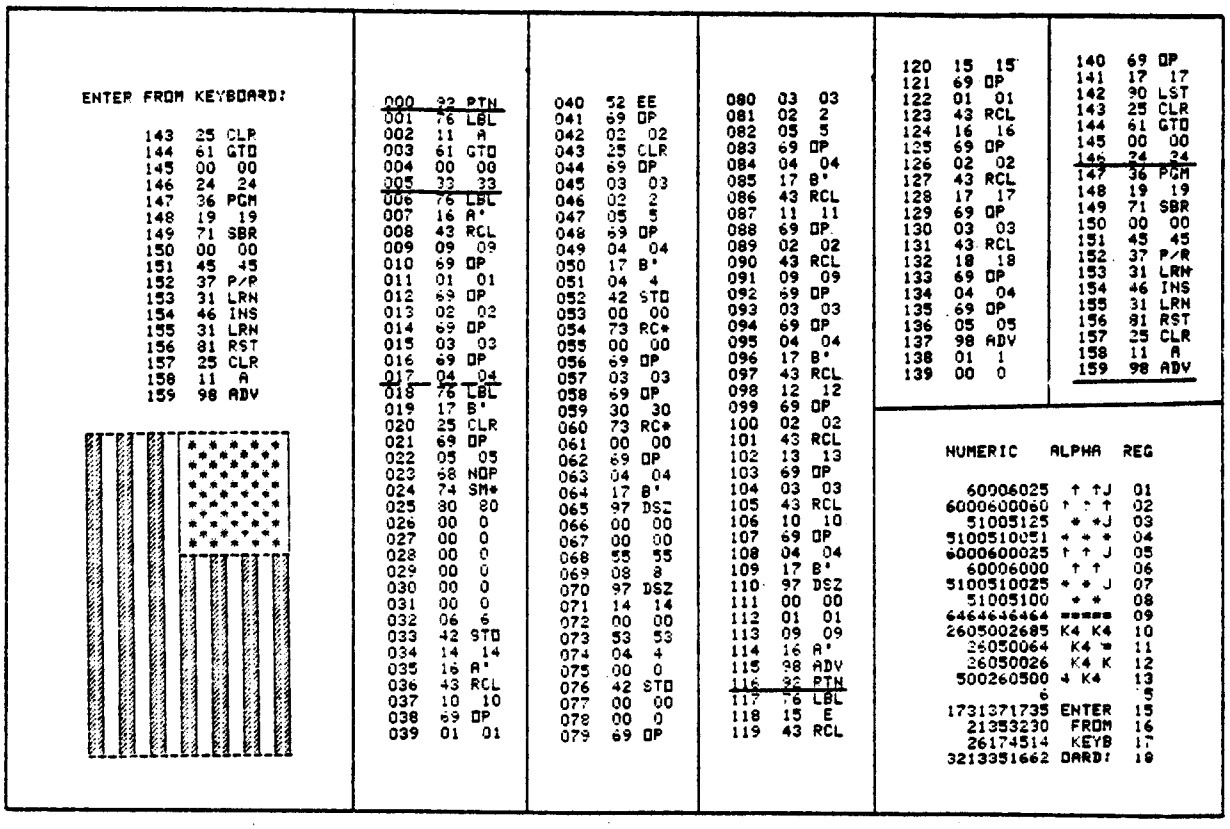
Since the initialization process is somewhat complex, and must be accomplished from the keyboard, Richard included label E to list all the necessary steps. Thus, when you press E to start things rolling, you'll see the listing as shown on the left. Pressing finally A will start the plotting. The ADV is just there to be accomplished after the plotting so you can tear off your beautiful drawing and dazzle your friends with it.

The pseudo instructions is quite interesting and touchy, remarks Richard. ANY USER CALLS TO A LABEL PAST THE PSEUDO CODE SEEMS TO PUT THE CALCULATOR INTO AN ENDLESS LOOP. (Never-never land!) That is the reason why in all programs, and especially in PLOT 60, all the user labels are so close to the beginning. But you may call direct addresses without punishment.

One interesting feature of the pseudo-code is that if OP 05 is NOT used, the buffer contents of the PC100 (A,B,C or D) printer are printed. Example: Press

pi PRT CLR OP 00 SBR 023

after initialization of any Graphics Mode program. It is surprising to see "pi" printed a second time when it no longer exists in the TI-59 memory (the OP 00 took care of that) nor in the display (it was wiped out by the CLR). OP 05 messages may be printed out a second time as well this way.



ENTER FROM KEYBOARD:

```
143  25 CLR
144  61 GTO
145  00  00
146  24  24
147  36 PGM
148  19  19
149  71 SBR
150  00  00
151  45  45
152  37 P/P
153  31 LRN
154  46 INS
155  31 LRN
156  81 RST
157  25 CLR
158  11  A
159  98 ADV
```

```
000  32 PTN
001  76 LBL
002  11  A
003  61 GTO
004  00  00
005  33  33
006  76 LBL
007  16 A'
008  43 RCL
009  09  09
010  69 OP
011  01  01
012  69 OP
013  02  02
014  69 OP
015  03  03
016  69 OP
017  04  04
018  76 LBL
019  17 B'
020  25 CLR
021  69 OP
022  05  05
023  68 NOP
024  74 SM*
025  80  80
026  00  0
027  00  0
028  00  0
029  00  0
030  00  0
031  00  0
032  06  6
033  42 STO
034  14  14
035  16 A'
036  43 RCL
037  10  10
038  69 OP
039  01  01
```

```
040  52 EE
041  69 OP
042  02  02
043  25 CLR
044  69 OP
045  03  03
046  02  2
047  05  5
048  69 OP
049  04  04
050  17 B'
051  04  4
052  42 STO
053  00  00
054  73 RC*
055  00  00
056  69 OP
057  03  03
058  69 OP
059  30  30
060  73 RC*
061  00  00
062  69 OP
063  04  04
064  17 B'
065  97 DSZ
066  00  00
067  00  00
068  55  55
069  08  8
070  97 DSZ
071  14  14
072  00  00
073  53  53
074  04  4
075  00  0
076  42 STO
077  00  00
078  00  0
079  69 OP
```

```
080  03  03
081  02  2
082  05  5
083  69 OP
084  04  04
085  17 B'
086  43 RCL
087  11  11
088  69 OP
089  02  02
090  43 RCL
091  09  09
092  69 OP
093  03  03
094  69 OP
095  04  04
096  17 B'
097  43 RCL
098  12  12
099  69 OP
100  02  02
101  43 RCL
102  13  13
103  69 OP
104  03  03
105  43 RCL
106  10  10
107  69 OP
108  04  04
109  17 B'
110  97 DSZ
111  00  00
112  01  01
113  09  09
114  16 A'
115  98 ADV
116  32 PTN
117  76 LBL
118  15  E
119  43 RCL
```

```
120  15  15
121  69 OP
122  01  01
123  43 RCL
124  16  16
125  69 OP
126  02  02
127  43 RCL
128  17  17
129  69 OP
130  03  03
131  43 RCL
132  18  18
133  69 OP
134  04  04
135  69 OP
136  05  05
137  98 ADV
138  01  1
139  00  0
```

```
140  69 OP
141  17  17
142  90 LST
143  25 CLR
144  61 GTO
145  00  00
146  24  24
147  36 PGM
148  19  19
149  71 SBR
150  00  00
151  45  45
152  37 P/R
153  31 LRN
154  46 INS
155  31 LRN
156  81 RST
157  25 CLR
158  11  A
159  98 ADV
```

```
NUMERIC    ALPHA  REG
  60006025  ↑ ↑J   01
6000600060  ↑ ↑ ↑  02
  51005125  * *J   03
5100510051  * * *  04
6000600025  ↑ ↑ J  05
  60006000  ↑ ↑    06
5100510025  * * J  07
  51005100  * *    08
6464646464  =====  09
2605002685  K4 K4  10
  26050064  K4 =   11
  26050026  K4 K   12
 5002605005  4 K4   13
           6        5
1731371735  ENTER   15
  21353230  FROM    16
  26174514  KEYB    17
3213351662  OARD:   18
```

TI EXCHANGE CENTER.- Albert Smith asks me to pass on the following information to members living in the greater Cleveland, Ohio area:

If any member has trouble locating any TI products from local stores, there is a place where they may purchase these items:

Texas Instruments Exchange Center.
23408 Commerce Park Road
Beachwood , Ohio 44122
Tel.: 216-464-5288

STANDARD VALUE RESISTORS.- *This program by Wallace A. Agy permits you to calculate the nearest standard value resistor given any odd value entry. It will compute the standard values in the standard tolerances of 20, 10, 5, 2, 1 and .5 %.*

Instructions: Enter any of the above tolerances and press A.
Enter the odd value and press B.
See the standard value (STD) printed.
If the second nearest standard value is required, press C.
See second standard (2ND) printed.

A sample printout in each of the tolerances is given, with the listing below.

```
      20.      %
456.123
    470.    STD
    330.    2ND

      10.      %
456.123
    470.    STD
    390.    2ND

       5.      %
456.123
    470.    STD
    430.    2ND

       2.      %
456.123
    464.    STD
    442.    2ND

       1.      %
456.123
    453.    STD
    464.    2ND

     0.5       %
456.123
    459.    STD
    453.    2ND
```

```
000  76 LBL
001  11  A
002  42 STO
003  06  06
004  98 ADV
005  06  6
006  01  1
007  69 OP
008  04  04
009  43 RCL
010  06  06
011  69 OP
012  06  06
013  32 RTN
014  76 LBL
015  12  B
016  99 PRT
017  28 LOG
018  42 STO
019  05  05
020  03  3
021  06  6
022  03  3
023  07  7
024  01  1
025  06  6
026  69 OP
027  04  04
028  43 RCL
029  05  05
030  29 CP
031  77  GE
032  00  00
033  37  37
034  75  -
035  01  1
036  95  =
037  59 INT
038  75  -
039  48 EXC
040  05  05
041  95  =
042  94 +/-
043  22 INV
044  28 LOG
045  42 STO
046  07  07
047  43 RCL
048  06  06
049  33 X²
050  65  ×
051  93  .
052  00  0
053  00  0
054  04  4
055  85  +
056  43 RCL
057  06  06
058  65  ×
059  01  1
060  93  .
061  05  5
062  85  +
063  01  1
064  95  =
065  59 INT
066  65  ×
067  93  .
068  00  0
069  01  1
070  01  1
071  09  9
072  09  9
073  02  2
074  07  7
075  95  =
076  42 STO
077  04  04
078  03  3
079  93  .
080  02  2
081  75  -
082  43 RCL
083  04  04
084  65  ×
085  03  3
086  95  =
087  59 INT
088  94 +/-
089  85  +
090  43 RCL
091  07  07
092  28 LOG
093  95  =
094  59 INT
095  42 STO
096  03  03
097  22 INV
098  28 LOG
099  35 1/X
100  65  ×
101  43 RCL
102  07  07
103  95  =
104  42 STO
105  08  08
106  01  1
107  42 STO
108  00  00
109  14  D
110  02  2
111  42 STO
112  00  00
113  14  D
114  43 RCL
115  08  08
116  55  ÷
117  53  (
118  43 RCL
119  01  01
120  65  ×
121  43 RCL
122  02  02
123  54  )
124  34 ΠX
125  85  +
126  01  1
127  95  =
128  42 STO
129  00  00
130  73 RC*
131  00  00
132  65  ×
133  43 RCL
134  03  03
135  22 INV
136  28 LOG
137  65  ×
138  43 RCL
139  05  05
140  22 INV
141  28 LOG
142  95  =
143  69 OP
144  06  06
145  32 RTN
146  76 LBL
147  14  D
148  43 RCL
149  08  08
150  23 LNX
151  55  ÷
152  43 RCL
153  04  04
154  95  =
155  59 INT
156  85  +
157  43 RCL
158  00  00
159  75  -
160  01  1
161  95  =
162  65  ×
163  43 RCL
164  04  04
165  95  =
166  22 INV
167  23 LNX
168  85  +
169  93  .
170  04  4
171  09  9
172  05  5
173  95  =
174  59 INT
175  73 ST*
176  00  00
177  93  .
178  01  1
179  06  6
180  04  4
181  65  ×
182  73 RC*
183  00  00
184  75  -
185  01  1
186  93  .
187  02  2
188  08  8
189  04  4
190  75  -
191  93  .
192  00  0
193  00  0
194  03  3
195  03  3
196  05  5
197  65  ×
198  73 RC*
199  00  00
200  33 X²
201  85  +
202  93  .
203  00  0
204  00  0
205  00  0
206  00  0
207  01  1
208  08  8
209  08  8
210  65  ×
211  73 RC*
212  00  00
213  45 YX
214  03  3
215  95  =
216  65  ×
217  53  (
218  43 RCL
219  04  04
220  65  ×
221  03  3
222  85  +
223  93  .
224  08  8
225  54  )
226  59 INT
227  95  =
228  59 INT
229  74 SM*
230  00  00
231  08  8
232  03  3
233  32 X⇄T
234  73 RC*
235  00  00
236  22 INV
237  67  EQ
238  02  02
239  44  44
240  08  8
241  02  2
242  72 ST*
243  00  00
244  32 RTN
245  76 LBL
246  13  C
247  03  3
248  03  3
249  01  1
250  01  1
251  06  6
252  69 OP
253  04  04
254  43 RCL
255  00  00
256  59 INT
257  94 +/-
258  85  +
259  03  3
260  61 GTO
261  01  01
262  27  27
```

TRANSFORM ANALOG FILTERS INTO DIGITAL EQUIVALENTS.- Henrique Sarmento Malvar, Professor of Electronics, Universidade de Brasilia, 70.910 Brasilia, DF, Brazil, Electronic Design, April 30, 1981, pp.145-148. Designing an infinite-impulse-response digital filter from the analog equivalent in the "s" domain normally requires a tedious transformation to the "z" domain. A program for the TI-58/59 simplifies the job.

The program works alright and is fast, because of direct addresses. But the author uses a lot of sequences such as 1 SUM 01, 1 SUM 04, etc. The program is just brimming with those. If he had also the TI-58 users in mind, as he says in the subtitle, he should have remembered that it can be shortened by OP 21 and OP 24. And his longest: 1 +/- SUM 04 might be more user-friendly written as: OP 34.

Also sequences such as))) = are not necessary. Just = is sufficient. An equal sign terminates any pending open parenthesis.

The program, as is, has 240 steps and therefore fits on one card side in the 59.

CALCULATORS AND PERSONAL COMPUTERS,- Special report, by Phil Koopman, Electronic Engineering Times, Issue 207, Monday May 4/11, 1981, pp. 67-94. A well researched and comprehensive article about personal computing devices in the price ranges $ 200.00 to about $ 2500.00.

MORTGAGE.- Wallace Agy is the author of this program. It permits you to get a print-out of each month's payment, the interest part of it, the principal part of it and the remaining balance. All you have to do is enter the original amount of the loan, the yearly interest rate in % and the number of payments. With the listing of the program, below, is shown a sample print-out for $ 1500.00 at 16.25 % over 14 months. As can be seen, the program makes subdivisions of 12 months or one full year.

Instructions: 1. Enter the loan amount and press A.
2. Enter the interest rate in % and press B.
3. Enter the number of months and press C. Payment in display.
4. To start the print-out, press D. Or enter new payment and press D.

```
1500.00  AMT
  16.25  I%
  14.00  NO
 118.34  PMT

 118.34
   1.
  20.31  INT
  98.03  PRC
1401.97  BAL
   2.
  18.99  INT
  99.36  PRC
1302.61  BAL
   3.
  17.64  INT
 100.70  PRC
1201.91  BAL
   4.
  16.28  INT
 102.07  PRC
1099.85  BAL
   5.
  14.89  INT
 103.45  PRC
 996.40  BAL
   6.
  13.49  INT
 104.85  PRC
 891.55  BAL
   7.
  12.07  INT
 106.27  PRC
 785.28  BAL
   8.
  10.63  INT
 107.71  PRC
 677.57  BAL
   9.
   9.18  INT
 109.17  PRC
 568.41  BAL
```

```
  10.
 7.70  INT
110.64  PRC
457.76  BAL
  11.
 6.20  INT
112.14  PRC
345.62  BAL
  12.
 4.68  INT
113.66  PRC
231.96  BAL

 1.00   YR

   1.
 3.14  INT
115.20  PRC
116.76  BAL
   2.
 1.58  INT
116.76  PRC
  0.00  BAL
```

```
000  76 LBL
001  11  A
002  58 FIX
003  02  02
004  47 CMS
005  42 STO
006  00  00
007  01  1
008  42 STO
009  06  06
010  01  1
```

```
011  03  3
012  03  3
013  00  0
014  03  3
015  07  7
016  69 OP
017  04  04
018  43 RCL
019  00  00
020  69 OP
021  06  06
022  91 R/S
023  76 LBL
024  12  B
025  42 STO
026  01  01
027  02  2
028  04  4
029  06  6
030  01  1
031  69 OP
032  04  04
033  43 RCL
034  01  01
035  69 OP
036  06  06
037  55  ÷
038  01  1
039  02  2
040  65  ×
041  93  .
042  00  0
043  01  1
044  95  =
045  42 STO
046  01  01
047  91 R/S
048  76 LBL
049  13  C
050  42 STO
```

```
051  05  05
052  03  3
053  01  1
054  03  3
055  02  2
056  69 OP
057  04  04
058  43 RCL
059  05  05
060  69 OP
061  06  06
062  43 RCL
063  00  00
064  55  ÷
065  53  (
066  53  (
067  01  1
068  75  -
069  53  (
070  01  1
071  85  +
072  43 RCL
073  01  01
074  54  )
075  45  YX
076  43 RCL
077  05  05
078  94 +/-
079  54  )
080  55  ÷
081  43 RCL
082  01  01
083  95  =
084  42 STO
085  02  02
086  03  3
087  03  3
088  03  3
089  00  0
090  03  3
```

```
091  07  7
092  69 OP
093  04  04
094  43 RCL
095  02  02
096  69 OP
097  06  06
098  98 ADV
099  91 R/S
100  76 LBL
101  14  D
102  42 STO
103  02  02
104  99 PRT
105  76 LBL
106  16  A'
107  01  1
108  02  2
109  32 X:T
110  22 INV
111  58 FIX
112  69 OP
113  23  23
114  43 RCL
115  03  03
116  99 PRT
117  58 FIX
118  02  02
119  02  2
120  04  4
121  03  3
122  01  1
123  03  3
124  07  7
125  69 OP
126  04  04
127  43 RCL
128  00  00
129  65  ×
130  43 RCL
```

```
131  01  01
132  95  =
133  69 OP
134  06  06
135  94 +/-
136  85  +
137  43 RCL
138  02  02
139  95  =
140  22 INV
141  44 SUM
142  00  00
143  42 STO
144  04  04
145  03  3
146  03  3
147  03  3
148  05  5
149  01  1
150  05  5
151  69 OP
152  04  04
153  43 RCL
154  04  04
155  69 OP
156  06  06
157  01  1
158  04  4
159  01  1
160  03  3
161  02  2
162  07  7
163  69 OP
164  04  04
165  43 RCL
166  00  00
167  69 OP
168  06  06
169  32 X:T
170  25 CLR
```

```
171  77  GE
172  17  B'
173  01  1
174  02  2
175  32 X:T
176  43 RCL
177  03  03
178  77  GE
179  18  C'
180  61 GTO
181  16  A'
182  76 LBL
183  18  C'
184  04  4
185  05  5
186  03  3
187  05  5
188  69 OP
189  04  04
190  43 RCL
191  06  06
192  98 ADV
193  69 OP
194  06  06
195  69 OP
196  26  26
197  25 CLR
198  42 STO
199  03  03
200  98 ADV
201  61 GTO
202  16  A'
203  76 LBL
204  17  B'
205  22 INV
206  58 FIX
207  25 CLR
208  91 R/S
```

BANKING AND FINANCE MODULE.- Michael E. Shanok, a TIPPP instructor and TI PPC Club member, sent me a special module which he produced. He is also willing to assist the club in cutting its own module. The finance module is of interest to all club members in that it contains a wealth of practical utility routines: Fourteen alpha codes, mostly financial terms, selective flag setting and resetting; labeling routines (OP 04 X:T OP 06 & OP 04 X:T FIX 2 OP 06 INV FIX) ; convert 5 numbers to print code (3.5 sec !);convert numbers to code and print with OP 04 label, 0 to 99,999,999.99; same as above, including punctuation with commas; selective clearing of OP 01 through OP 04 registers; multiply and divide by 100 to 10,000; print line of dashes, asterisks or triangles; given a code, prints in all four quadrants; days between dates , with labeled printout, given early and late date; print out day of week and given date; future date, given early date and number of days.

The market price for this module is usually $ 235.00, but Michael will send it to any TI PPC Club member for $ 40.00 and will make selected listings available. The module cannot be loaded down. It is protected.

Please write to Michael E. Shanok, Plant Engineering Consultants, 2405 Whitney Ave., Hamden, CT, 06518, USA. Or call him at (203) 281-6551.

6800 Microprocessor Disassembler.- Electronic Design, March 19, 1981, pp 243-247, by John C. Vermeulen, IBM Corp. Boulder, CO, 80302, USA. The TI-59 hooked up to the PC100A printer translates machine code of the 6800 into assembly language. It is extremely valuable for debugging microcode, especially for checking program modification made by hand.

DYNAMIC NIM-B ON THE TI-57.- Rick Wenger succeeded to cram a dynamic NIM-B game program on a 57. His instructions are:

1. Key CLR RST and store the initial pile size STO 0. (between 12 and 100)
2. Key your move, which must be smaller than the pile size for your first move, smaller than or equal to the claculator's move thereafter)
3. Press R/S. See reduced pile size (1 pause) calculator's move (which will be smaller than or equal to twice your last move) (2 pauses) and finally re-reduced pile size.
4. First player to get all the cookies wins. Repeat step 2 until:
   a. Flashing zero = user wins.
   b. Flashing 9's = calculator wins.
   c. Flashing anything else = calculator wins by user's default. Move smaller than or equal to zero or larger than twice calculator move.
   User moves effective only in their integral parts.

GO TO 1 any time for a new game.

```
00: STO 7  *LBL 0  *INT  STO 6  INV SUM 0  X:T  INV *X ≥ T  +/-  *LOG
09: RCL 0  *PAUSE  *C.T.  *X=T  GTO 9  X:T  STO 1  2  *PROD 6  RCL 6 *X ≥ T  *LOG
22: *LBL  1  RCL 1  INV  SUM 1  INV SUM 7  1
27: *LBL 2  SUM  1  *EXC 1  INV  * X ≥ T  GTO 2  INV  *X=T  GTO 1
34: RCL 6  X:T  * X ≥ T  *X=T  X  2  STO 7  1 =  INV SUM  0  * PROD  7
45: *PAUSE  *PAUSE  RCL 0  R/S  GTO 0
```

Register allocation:
R0: pile size
R1: Fibonacci number generator
R6: User move- calculator limit
R7(t): User limit (at R/S) - potential calculator move (see 09, 14, 25) - calculator limit .(see 34, 35)

FRACTION REDUCTION.- Under the title "Brueche snell gekuertzt (TI-59)" I found this program by Kai Rohleff of West Germany. The original program was intended for TI-59 (or TI-58) only. In fact, the author doesn't seem to believe in user-defined labels either, as you had to start the program with RST R/S !

A little rework on it produced acceptable descriptors. It even gives you now the largest common denominator, which was buried somewhere in R01 and could be produced, according to author, by pressing RCL 01 !!!!

Instructions:

1. *Enter the numerator of your fraction and press A.*
2. *Enter the denominator of your fraction and press R/S.*
3. *See both printed followed by the reduced fraction, followed by the largest common denominator. (LCD)*

For example, enter 4562/123789 and the program tells you that it cannot be reduced any further , gives 1 as the LCD. On the other hand, enter 12345/456789 and it is promptly reduced to 4115/152263 with an LCD of 3.

```
000  76 LBL
001  16 A'
002  03  3
003  09  9
004  04  4
005  01  1
006  03  3
007  08  8
008  69 OP
009  04  04
010  32 X:T
011  69 OP
012  06  06
013  92 RTN
014  76 LBL
015  17 B'
016  01  1
017  06  6
018  01  1
019  07  7
020  03  3
021  09  9
022  61 GTO
023  00  00
024  08  08
025  76 LBL
026  11  A
027  42 STO
028  01  01
029  42 STO
030  03  03
031  32 X:T
032  16 A'
033  91 R/S
034  42 STO
035  02  02
036  42 STO
037  04  04
038  32 X:T
039  17 B'
040  43 RCL
041  02  02
042  29 CP
043  67 EQ
044  00  00
045  69  69
046  43 RCL
047  01  01
048  55  ÷
049  43 RCL
050  02  02
051  95  =
052  22 INV
053  59 INT
054  65  ×
055  43 RCL
056  02  02
057  85  +
058  93  .
059  01  1
060  95  =
061  59 INT
062  48 EXC
063  02  02
064  42 STO
065  01  01
066  61 GTO
067  00  00
068  40  40
069  43 RCL
070  03  03
071  55  ÷
072  43 RCL
073  01  01
074  95  =
075  32 X:T
076  16 A'
077  43 RCL
078  04  04
079  55  ÷
080  43 RCL
081  01  01
082  95  =
083  32 X:T
084  17 B'
085  43 RCL
086  01  01
087  32 X:T
088  02  2
089  07  7
090  01  1
091  05  5
092  01  1
093  06  6
094  71 SBR
095  00  00
096  08  08
097  98 ADV
098  91 R/S
```

BUILDING SYSTEMS DESIGN WITH PROGRAMMABLE CALCULATORS by Sital Daryanani, P.E.

Available from Architectural Record Books, P.O. Box 682, Hightstown, New Jersey 08520. Price: $29.50

This book is for all architects and engineers who design buildings. It contains 22 programs, developed over a period of 2 years, to aid the designer by relieving him of hand computations, allowing him time to concentrate on design and analysis. The quickness of the programmable calculator will hopefully encourage the designers to test alternative methods and come up with the least expensive, most energy efficient design, which was formerly not cost effective for the designer.

The author cites an instance in which he reduced the design time on 26 apartment buildings from 3 days per building by hand to only 3 hours per building using a programmable calculator.

The programs were written in small units (subroutines) which can be quickly understood and modified, if desired, by the user. Mr. Daryanani has been able to give only 2 hours of instructions to junior engineers and they were able to use these programs.

Users are encouraged to modify these programs to meet their specific needs or use them as building blocks to develop more complex design programs.

Use of the printer is encouraged to provide a permanent record. Since programs are written with printer dependence, use of the TI59 only would require slight modifications. All input data and intermediate answers are printed with alpha descriptors as well as the final results.

The engineering procedures are mainly adapted from the standard industry technical manuals and references are given.

The programs are well written and are loaded with alphanumerics. Each program has a general description, equations, references, instructions, several examples, explanation of labels, subroutines and memories as well as actual listings (not typed) for both the TI59 and HP97 calculators.

The programs are categorized as follows:

PIPING DESIGN PROGRAMS: General Piping System Design, Water Piping System Design, Piping System Volume and Expansion Tank Sizing.

AIR DUCT DESIGN PROGRAMS: Air Duct Sizing, Air Duct Design (Static Regain Method), Air Duct Friction Loss, Air Duct Heat Loss/ Gain.

LIGHTING DESIGN PROGRAMS: Lighting Power Budget, Lighting Design (Lumen Method).

SANITARY ENGINEERING PROGRAMS: Storm Water System Pipe Sizing, Soil and Waste Water System Pipe Sizing, Water System Pipe Sizing, Gas System Pipe Sizing.

HEAT TRANSMISSION COEFFICIENT PROGRAMS: Heat Transmission Coefficient, Over-all Thermal Transmittance Value.

SOLAR ENERGY PROGRAMS: Solar Shading, Monthly and Annual Average Insolation on Tilted Surfaces.

ENGINEERING ECONOMIC ANALYSIS PROGRAM: Life Cycle Cost Analysis Program, Part I and II.

GENERAL UTILITY PROGRAMS: Preliminary Design Conditions, Space Planning, Distribution Network.

(Reviewed by Frank Blachly)

LIFE CYCLE COSTING WITH A PROGRAMMABLE CALCULATOR.- by Warren NG, PE, Mecanical Engineer, Plant Engineering Dept., Lawrence Berkeley Laboratory, University of California, Berkely CA, Heating/Piping/Air Conditioning, April 1981, pp90-94.

Life cycle costing can be defined as the determination of the total cost of owning, operating, and maintaining a planned project. Most LCC studies do not account for the basic rate of inflation. But the differential cost escalations of dominant factors, such as money and fuel costs, relative to inflation should be considered in any study. A special uniform woth factor (UPWF) must be used.

A TI-59 program provides a means to compute the UPWF involved in any period, taking into account the differential ciscount rate and one or more differential fuel escalation tates hypothesized within the sudy period.

The mathematical formula for this special UPWF together with 30 year forecasts of fuel costs and fuel differential cost escalation rates are included in the recently published LCC methodology of the Department of Energy. (Federal Register, Vol.45, No.16, January 23, 1980: 10 code of Federal Regulations, Part 436, Subpart A, and Federal Register, Vol.45, No.209, October 27, 1980, 10 Code of Federal Regulations, Part 436.)

MODULO 210 SPEEDY FACTOR FINDER IN FAST MODE.-Patrick Acosta did it again! He developec the fastest SFF program to date. Patrick uses a TI-58 and writes the programs for a TI-59. No way for him to check them. But he is that confident they will work, he submits them anyway! This one will not run in fast mode as it was written originally, because it needs 879.09 partitioning. Unfortunately, one of the unpublished ideosyncrasies of the fast mode entry is, that it returns any partitioning to the turn-on one. (6 OP 17) Since you can't change the partitioning from the keyboard without losing fast mode, you must provide the repartitioning in the program. Palmer O. Hansen, who reviewed the program, made the necessary changes in the first 26 steps. This way it produced the fastest solutions I have seen to date. In fact the HP PPC JOURNAL in v8n2p29, March-April 81, mentions some impressive times achieved by the HP-41C and compares them to TI-59 achievements. Those comparisons have to be reviewed in the light of this new program. Another advantage of this program is, that it may be run also in normal mode. This way it permits to get factors from 12-digit numbers, not possible in fast mode. I don't know if it can be done on the HP-41C. I am very curious to find out.

The solution times for the standard problems are:

	Normal mode.	Fast mode.
987654321 = 3X3X17X17X379721	111 sec	58 sec
103569859 = 463X467X479	90 sec	45 sec
9999999967 = prime	not measured	2 hr 8 min 25 sec

Instructions: Cards are loaded in turn-on partitioning. Use this sequence:

1. Turn on.	
2. Load side 1.	See a 1 in the display.
3. Set up fast mode by RST R/S	See a 0 in the display
4. Re-load side 1	See a flashing 1 then a steady 1 in the dsiplay.
5. CLR	See a 0 in the display
6. Load side 2	See a short flash of a 2 the a steady 2 in the display.
7. CLR	See a 0 in the display.
8. Load side 3	See a short flash of a 3, then a steady 3 in the display.
9. CLR	See a 0 in the display.
10.Load side 4	See a 4 flash in the display, then a steady 0. This means we are ready to accept test numbers.
11. Enter test number and press R/S.	Test number and factors are printed and pause-displayed. The calculator stops with either a 1 or the highest factor in the display. It does not print the unwanted 1.

12.To test an additional number, press R/S and see a 0 in the display.

MODULO 210 SFF FAST MODE, Patrick Acosta

000	00	0	107	43	RCL	214	00	00	321	02	02	428	68	NOP	535	67	EQ	642	34	fX
001	00	0	108	01	01	215	53	(	322	95	=	429	43	RCL	536	00	00	643	32	XIT
002	00	0	109	55	÷	216	02	2	323	22	INV	430	01	01	537	55	÷	644	43	PCL
003	00	0	110	43	RCL	217	44	SUM	324	59	INT	431	55	÷	538	04	4	645	02	02
004	00	0	111	02	02	218	02	02	325	67	EQ	432	43	RCL	539	44	SUM	646	95	=
005	36	PGM	112	95	=	219	43	RCL	326	00	00	433	02	02	540	02	02	647	22	INV
006	02	02	113	22	INV	220	01	01	327	53	(	434	95	=	541	43	RCL	648	77	GE
007	71	SBR	114	59	INT	221	55	÷	328	06	6	435	22	INV	542	01	01	649	00	00
008	02	02	115	67	EQ	222	43	RCL	329	44	SUM	436	59	INT	543	55	÷	650	93	.
009	39	39	116	00	00	223	02	02	330	02	02	437	67	EQ	544	43	RCL	651	29	CP
010	09	9	117	59	59	224	95	=	331	43	RCL	438	00	00	545	02	02	652	22	INV
011	00	0	118	01	1	225	22	INV	332	01	01	439	53	(	546	95	=	653	59	INT
012	22	INV	119	44	SUM	226	59	INT	333	55	÷	440	04	4	547	22	INV	654	67	EQ
013	58	FIX	120	02	02	227	67	EQ	334	43	RCL	441	44	SUM	548	59	INT	655	00	00
014	01	1	121	43	RCL	228	00	00	335	02	02	442	02	02	549	67	EQ	656	59	59
015	66	PAU	122	01	01	229	51	51	336	95	=	443	43	RCL	550	00	00	657	02	2
016	91	R/S	123	55	÷	230	04	4	337	22	INV	444	01	01	551	55	÷	658	44	SUM
017	66	PAU	124	43	RCL	231	44	SUM	338	59	INT	445	55	÷	552	08	8	659	02	02
018	91	R/S	125	02	02	232	02	02	339	67	EQ	446	43	RCL	553	44	SUM	660	68	NOP
019	66	PAU	126	95	=	233	43	RCL	340	00	00	447	02	02	554	02	02	661	68	NOP
020	91	R/S	127	22	INV	234	01	01	341	53	(	448	95	=	555	43	RCL	662	68	NOP
021	66	PAU	128	59	INT	235	55	÷	342	06	6	449	22	INV	556	01	01	663	68	NOP
022	01	1	129	67	EQ	236	43	RCL	343	44	SUM	450	59	INT	557	55	÷	664	68	NOP
023	69	OP	130	00	00	237	02	02	344	02	02	451	67	EQ	558	43	RCL	665	68	NOP
024	17	17	131	57	57	238	95	=	345	43	RCL	452	00	00	559	02	02	666	68	NOP
025	25	CLR	132	02	2	239	22	INV	346	01	01	453	53	(	560	95	=	667	43	RCL
026	91	R/S	133	44	SUM	240	59	INT	347	55	÷	454	06	6	561	22	INV	668	01	01
027	68	NOP	134	02	02	241	67	EQ	348	43	RCL	455	44	SUM	562	59	INT	669	55	÷
028	42	STO	135	43	RCL	242	00	00	349	02	02	456	02	02	563	67	EQ	670	43	RCL
029	01	01	136	01	01	243	51	51	350	95	=	457	43	RCL	564	00	00	671	02	02
030	98	ADV	137	55	÷	244	06	6	351	22	INV	458	01	01	565	57	57	672	95	=
031	99	PRT	138	43	RCL	245	44	SUM	352	59	INT	459	55	÷	566	06	6	673	22	INV
032	02	2	139	02	02	246	02	02	353	67	EQ	460	43	RCL	567	44	SUM	674	59	INT
033	42	STO	140	95	=	247	43	RCL	354	00	00	461	02	02	568	02	02	675	67	EQ
034	02	02	141	22	INV	248	01	01	355	55	÷	462	95	=	569	43	RCL	676	00	00
035	29	CP	142	59	INT	249	55	÷	356	02	2	463	22	INV	570	01	01	677	55	÷
036	61	GTO	143	67	EQ	250	43	RCL	357	44	SUM	464	59	INT	571	55	÷	678	06	6
037	01	01	144	00	00	251	02	02	358	02	02	465	67	EQ	572	43	RCL	679	44	SUM
038	07	07	145	57	57	252	95	=	359	43	RCL	466	00	00	573	02	02	680	02	02
039	08	8	146	02	2	253	22	INV	360	01	01	467	53	(	574	95	=	681	43	RCL
040	05	5	147	44	SUM	254	59	INT	361	55	÷	468	08	8	575	22	INV	682	01	01
041	00	0	148	02	02	255	67	EQ	362	43	RCL	469	44	SUM	576	59	INT	683	55	÷
042	61	GTO	149	43	RCL	256	00	00	363	02	02	470	02	02	577	67	EQ	684	43	RCL
043	00	00	150	01	01	257	53	(	364	95	=	471	43	RCL	578	00	00	685	02	02
044	82	82	151	55	÷	258	02	2	365	22	INV	472	01	01	579	57	57	686	95	=
045	08	8	152	43	RCL	259	44	SUM	366	59	INT	473	55	÷	580	04	4	687	22	INV
046	06	6	153	02	02	260	02	02	367	67	EQ	474	43	RCL	581	44	SUM	688	59	INT
047	04	4	154	95	=	261	43	RCL	368	00	00	475	02	02	582	02	02	689	67	EQ
048	61	GTO	155	22	INV	262	01	01	369	53	(	476	95	=	583	43	RCL	690	00	00
049	00	00	156	59	INT	263	55	÷	370	06	6	477	22	INV	584	01	01	691	57	57
050	82	82	157	67	EQ	264	43	RCL	371	44	SUM	478	59	INT	585	55	÷	692	06	6
051	01	1	158	00	00	265	02	02	372	02	02	479	67	EQ	586	43	RCL	693	44	SUM
052	85	+	159	55	÷	266	95	=	373	43	RCL	480	00	00	587	02	02	694	02	02
053	01	1	160	04	4	267	22	INV	374	01	01	481	55	÷	588	95	=	695	43	RCL
054	85	+	161	44	SUM	268	59	INT	375	55	÷	482	04	4	589	22	INV	696	01	01
055	01	1	162	02	02	269	67	EQ	376	43	RCL	483	44	SUM	590	59	INT	697	55	÷
056	85	+	163	43	RCL	270	00	00	377	02	02	484	02	02	591	67	EQ	698	43	RCL
057	01	1	164	01	01	271	51	51	378	95	=	485	43	RCL	592	00	00	699	02	02
058	85	+	165	55	÷	272	06	6	379	22	INV	486	01	01	593	57	57	700	95	=
059	53	(	166	34	fX	273	44	SUM	380	59	INT	487	55	÷	594	06	6	701	22	INV
060	43	RCL	167	32	XIT	274	02	02	381	67	EQ	488	43	RCL	595	44	SUM	702	59	INT
061	02	02	168	43	RCL	275	43	RCL	382	00	00	489	02	02	596	02	02	703	67	EQ
062	55	÷	169	02	02	276	01	01	383	55	÷	490	95	=	597	43	RCL	704	00	00
063	02	2	170	95	=	277	55	÷	384	04	4	491	22	INV	598	01	01	705	57	57
064	01	1	171	22	INV	278	43	RCL	385	44	SUM	492	59	INT	599	55	÷	706	04	4
065	00	0	172	77	GE	279	02	02	386	02	02	493	67	EQ	600	43	RCL	707	44	SUM
066	54	)	173	00	00	280	95	=	387	43	RCL	494	00	00	601	02	02	708	02	02
067	22	INV	174	93	.	281	22	INV	388	01	01	495	55	÷	602	95	=	709	43	RCL
068	59	INT	175	29	CP	282	59	INT	389	55	÷	496	02	2	603	22	INV	710	01	01
069	65	×	176	22	INV	283	67	EQ	390	43	RCL	497	44	SUM	604	59	INT	711	55	÷
070	05	5	177	59	INT	284	00	00	391	02	02	498	02	02	605	67	EQ	712	43	RCL
071	04	4	178	67	EQ	285	53	(	392	95	=	499	43	RCL	606	00	00	713	02	02
072	95	=	179	00	00	286	04	4	393	22	INV	500	01	01	607	59	59	714	95	=
073	59	INT	180	55	÷	287	44	SUM	394	59	INT	501	55	÷	608	02	2	715	22	INV
074	65	×	181	02	2	288	02	02	395	67	EQ	502	43	RCL	609	44	SUM	716	59	INT
075	01	1	182	44	SUM	289	43	RCL	396	00	00	503	02	02	610	02	02	717	67	EQ
076	04	4	183	02	02	290	01	01	397	55	÷	504	95	=	611	43	RCL	718	00	00
077	85	+	184	68	NOP	291	55	÷	398	02	2	505	22	INV	612	01	01	719	57	57
078	01	1	185	68	NOP	292	43	RCL	399	44	SUM	506	59	INT	613	55	÷	720	02	2
079	00	0	186	68	NOP	293	02	02	400	02	02	507	67	EQ	614	43	RCL	721	44	SUM
080	07	7	187	68	NOP	294	95	=	401	43	RCL	508	00	00	615	02	02	722	02	02
081	95	=	188	68	NOP	295	22	INV	402	01	01	509	55	÷	616	95	=	723	43	RCL
082	42	STO	189	68	NOP	296	59	INT	403	55	÷	510	04	4	617	22	INV	724	01	01
083	04	04	190	68	NOP	297	67	EQ	404	34	fX	511	44	SUM	618	59	INT	725	55	÷
084	43	RCL	191	43	RCL	298	00	00	405	32	XIT	512	02	02	619	67	EQ	726	43	RCL
085	02	02	192	01	01	299	53	(	406	43	RCL	513	43	RCL	620	00	00	727	02	02
086	22	INV	193	55	÷	300	02	2	407	02	02	514	01	01	621	57	57	728	95	=
087	49	PRD	194	43	RCL	301	44	SUM	408	95	=	515	55	÷	622	04	4	729	22	INV
088	01	01	195	02	02	302	02	02	409	22	INV	516	43	RCL	623	44	SUM	730	59	INT
089	99	PRT	196	95	=	303	43	RCL	410	77	GE	517	02	02	624	02	02	731	67	EQ
090	66	PAU	197	22	INV	304	01	01	411	00	00	518	95	=	625	43	RCL	732	00	00
091	83	GD+	198	59	INT	305	55	÷	412	93	.	519	22	INV	626	01	01	733	57	57
092	04	04	199	67	EQ	306	43	RCL	413	29	CP	520	59	INT	627	55	÷	734	04	4
093	01	1	200	00	00	307	02	02	414	22	INV	521	67	EQ	628	43	RCL	735	44	SUM
094	32	XIT	201	53	(	308	95	=	415	59	INT	522	00	00	629	02	02	736	02	02
095	43	RCL	202	04	4	309	22	INV	416	67	EQ	523	55	÷	630	95	=	737	43	RCL
096	01	01	203	44	SUM	310	59	INT	417	00	00	524	02	2	631	22	INV	738	01	01
097	67	EQ	204	02	02	311	67	EQ	418	53	(	525	44	SUM	632	59	INT	739	55	÷
098	01	01	205	43	RCL	312	00	00	419	06	6	526	02	02	633	67	EQ	740	43	RCL
099	01	01	206	01	01	313	53	(	420	44	SUM	527	43	RCL	634	00	00	741	02	02
100	99	PRT	207	55	÷	314	04	4	421	02	02	528	01	01	635	57	57	742	95	=
101	98	ADV	208	43	RCL	315	44	SUM	422	68	NOP	529	55	÷	636	06	6	743	22	INV
102	66	PAU	209	02	02	316	02	02	423	68	NOP	530	43	RCL	637	44	SUM	744	59	INT
103	91	R/S	210	95	=	317	43	RCL	424	68	NOP	531	02	02	638	02	02	745	67	EQ
104	61	GTO	211	22	INV	318	01	01	425	68	NOP	532	95	=	639	43	RCL	746	00	00
105	00	00	212	59	INT	319	55	÷	426	68	NOP	533	22	INV	640	01	01	747	57	57
106	25	25	213	67	EQ	320	43	RCL	427	68	NOP	534	59	INT	641	55	÷	748	06	6

MODULO 210 SFF FAST MODE, Patrick Acosta.

```
749  44 SUM
750  02  02
751  43 RCL
752  01  01
753  55  ÷
754  43 RCL
755  02  02
756  95  =
757  22 INV
758  59 INT
759  67  EQ
760  00  00
761  59  59
762  02  2
763  44 SUM
764  02  02
765  43 RCL
766  01  01
767  55  ÷

768  43 RCL
769  02  02
770  95  =
771  22 INV
772  59 INT
773  67  EQ
774  00  00
775  57  57
776  06  6
777  44 SUM
778  02  02
779  43 RCL
780  01  01
781  55  ÷
782  43 RCL
783  02  02
784  95  =
785  22 INV
786  59 INT

787  67  EQ
788  00  00
789  59  59
790  04  4
791  44 SUM
792  02  02
793  43 RCL
794  01  01
795  55  ÷
796  43 RCL
797  02  02
798  95  =
799  22 INV
800  59 INT
801  67  EQ
802  00  00
803  59  59
804  02  2
805  44 SUM

806  02  02
807  43 RCL
808  01  01
809  55  ÷
810  43 RCL
811  02  02
812  95  =
813  22 INV
814  59 INT
815  67  EQ
816  00  00
817  57  57
818  04  4
819  44 SUM
820  02  02
821  43 RCL
822  01  01
823  55  ÷
824  43 RCL

825  02  02
826  95  =
827  22 INV
828  59 INT
829  67  EQ
830  00  00
831  57  57
832  02  2
833  44 SUM
834  02  02
835  43 RCL
836  01  01
837  55  ÷
838  43 RCL
839  02  02
840  95  =
841  22 INV
842  59 INT
843  67  EQ

844  00  00
845  57  57
846  01  1
847  00  0
848  44 SUM
849  02  02
850  43 RCL
851  01  01
852  55  ÷
853  43 RCL
854  02  02
855  95  =
856  22 INV
857  59 INT
858  67  EQ
859  00  00
860  39  39
861  02  2
862  44 SUM

863  02  02
864  43 RCL
865  01  01
866  55  ÷
867  43 RCL
868  02  02
869  95  =
870  22 INV
871  59 INT
872  67  EQ
873  00  00
874  45  45
875  01  1
876  00  0
877  61 GTO
878  01  01
879  61  61
```

CALCULATOR STATUS.- One more in the series of candidates for our own module. This program, by Bill Beebe, Jr., allows you to determine the status of the following calculator operations:

1. Angular mode. 2. Partitioning set. 3. Flag status. 4. Fix-decimal status. 5. EE or ENG exponent set. 6. Printer connected.

In addition to being able to determine the status of the above operations, you can also store the coded value of the first five status operations in data register 00, by pressing E. The status may then be moved to another register. At a later time the coded value may then be returned to R00, so that by pressing E' the calculator can be forced to assume the status set by the status code.

1. Angular mode.-Press A to display the code for the current angular mode. Degree mode gives a 0, radian a -1 and grad a 1. A simple convention for remembering this code is that "radian is the second function of the minus key and grad is the second fynction of the plus key."

2. Partitioning set.- Press B to display the partitioning status. It will be displayed as an integer 0 to 6 for the TI-58 and 0 to 9 for the TI-59. It is the same number required to set the current partitioning via OP 17.

3. Flag set status. Pressing C will produce a unique code of ones and zeroes in the display, which indicates the set/reset status of all 10 flags. The code is interpreted as follows:

Digit	*x*	*.*	*x*	*x*	*x*	*x*	*x*	*x*	*x*	*x*	*x*	*(watch the decimal point)*
Flag #	*0*		*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	

For example, setting flags 0, 1, 2, and 5 and pressing C will produce 1.11001 in the display. Note that the display should not be in EE nor ENG format.

4. Fix-decimal status.- Pressing D displays the current fix. Display of 9 means INV FIX. This routine will remove EE format, but not ENG format. Also, if either EE or ENG format is on display, and if the display is in either FIX 8 or FIX 9, the routine will incorrectly produce a 7 for both fixes. Howver, for FIX - through 7, the routine works perfectly.

5. EE/ENG set.- Pressing B' displays the code for the current display exponent set. A 0 (zero) indicates neither was set, 1 means EE is set and 2 means ENG was set. A simple convention for remembering is that "ENG is the second function of of the EE key and EE is the primary one. Second = 2, primary = 1.)

6. Printer connected status.- Press N A' (N is a single digit from 0 to 9) while the calculator is connected to the printer. It will cause flag N to be set. If the printer is not connected, flag N is reset.

This routine use HIR 8. Therefore do not use more than seven pending operations or an alphanumeric previously stored via OP 04, when calling this routine. In addition, with the printer connected or not, the display is zeroed, NOT CLEARED. If it is not, the display contains a 1.

Calculator status (continued)

Save Calculator Status.- Press E to assemble the current status of all the flags, the partinioning, the angular mode, the fix setting and the EE/ENG setting in data register R00. This routine stops with the code in the display. The code is assembled as A. P P D E F F F F in which A is the angular mode, PP the partitioning set, E the exponent mode and FFFF a soecial number representing the combined set/reset status of all the flags. The flags in this number carry the following decimal "weight" or value:

Flag #	0	9	8	7	6	5	4	3	2	1
Decimal value	1	2	4	8	16	32	64	128	256	512

If a flag is set, its value is summed into the total flag number. Thus the largest possible number is 1023 for all 10 flags set.

This routine removes all FIX and EE/ENG formats. The angular status A is the status number produced by routine a plus 1.

Initialize Status.- Pressing E' with a valid status code in R00 will force the calculator to assume the characteristics encoded in the status number. In the case of the flag status, all flags are set or reset according to the code, regardless of any prior setting.Routine stops with zeroed (not cleared) display.

NOTES: The absolute value function at steps 002, 026 and 112 is used to harden the display and to prevent unwanted results.

The INS (insert) instruction at step 260 is synthesized with STO 46, after which the STO is deleted.INS serves as a dummy operator to supply the missing operand between the left parenthesis and the divide operator.

```
000  76 LBL      040  00  0 00     080  76 LBL      120  08  08     160  93  .      200  03  03     240  48  48
001  11  A       041  43  43       081  13  C       121  53  (      161  01  1      201  80 GRD     241  01  1
002  50 I×I      042  52  EE       082  53  (       122  02  2      162  49 PRD     202  32 X↕T     242  52  EE
003  09  9       043  32 RTN       083  09  9       123  82 HIR     163  00  00     203  54  )      243  67  EQ
004  00  0       044  76 LBL       084  42 STO      124  48  48     164  22 INV     204  53  (      244  02  02
005  39 COS      045  14  D        085  00  00      125  22 INV     165  58 FIX     205  22 INV     245  48  48
006  69 DP       046  53  (        086  22 INV     126  87 IFF     166  12  B      206  58 FIX     246  57 ENG
007  10  10      047  93  .        087  87 IFF     127  40 IND     167  44 SUM     207  22 INV     247  32 X↕T
008  32 RTN      048  03  3        088  40 IND     128  00  00     168  00  00     208  57 ENG     248  54  )
009  76 LBL      049  35 1/X       089  00  00     129  01  01     169  93  .      209  65  ×      249  53  (
010  12  B       050  75  -        090  00  00     130  34  34     170  00  0      210  01  1      250  29 CP
011  53  (       051  52  EE       091  98  98     131  82 HIR     171  01  1      211  00  0      251  65  ×
012  93  .       052  22 INV      092  43 RCL     132  18  18     172  49 PRD     212  42 STO     252  01  1
013  01  1       053  52  EE       093  00  00     133  95  =      173  00  00     213  00  00     253  00  0
014  85  +       054  54  )        094  22 INV     134  97 DSZ     174  11  A      214  33 X²     254  42 STO
015  35 1/X      055  28 LOG      095  28 LOG     135  00  00     175  44 SUM     215  75  -      255  00  00
016  65  ×       056  59 INT      096  35 1/X     136  01  01     176  00  00     216  22 INV     256  33 X²
017  69 DP       057  50 I×I      097  85  +      137  22  22     177  69 DP      217  59 INT     257  33 X²
018  16  16      058  32 RTN      098  97 DSZ     138  00  0      178  20  20     218  49 PRD     258  54  )
019  22 INV      059  76 LBL      099  00  00     139  54  )      179  43 RCL     219  00  00     259  53  (
020  59 INT      060  16 A'       100  00  00     140  42 STO     180  00  00     220  54  )      260  46 INS
021  54  )       061  42 STO      101  86  86     141  00  00     181  32 RTN     221  69 DP      261  55  ÷
022  59 INT      062  00  00      102  01  1      142  04  4      182  76 LBL     222  17  17     262  02  2
023  32 RTN      063  01  1       103  87 IFF     143  22 INV     183  10 E'      223  58 FIX     263  75  -
024  76 LBL      064  82 HIR      104  00  00     144  28 LOG     184  53  (      224  40 IND     264  22 INV
025  17 B'       065  08  08      105  01  01     145  35 1/X     185  43 RCL     225  00  00     265  59 INT
026  50 I×I      066  00  0       106  08  08     146  49 PRD     186  00  00     226  53  (      266  22 INV
027  03  3       067  69 DP       107  00  0      147  00  00     187  50 I×I     227  43 RCL     267  67  EQ
028  35 1/X      068  04  04      108  54  )      148  17 B'      188  75  -      228  00  00     268  02  02
029  52  EE       069  82 HIR      109  32 RTN     149  44 SUM     189  59 INT     229  22 INV     269  71  71
030  00  0       070  18  18      110  76 LBL     150  00  00     190  32 X↕T     230  59 INT     270  22 INV
031  00  0       071  39 CP       111  15  E      151  93  .      191  00  0      231  65  ×      271  86 STF
032  22 INV      072  67  EQ       112  50 I×I     152  01  1      192  70 RAD     232  01  1      272  40 IND
033  52  EE       073  00  00      113  01  1      153  49 PRD     193  67  EQ     233  00  0      273  00  00
034  34 fX       074  76  76      114  00  0      154  00  00     194  02  02     234  75  -      274  54  )
035  34 fX       075  22 INV      115  42 STO     155  22 INV     195  03  03     235  59 INT     275  97 DSZ
036  34 fX       076  86 STF      116  00  00     156  57 ENG     196  60 DEG     236  32 X↕T     276  00  00
037  59 INT      077  40 IND      117  93  .      157  14  D      197  01  1      237  00  0      277  02  02
038  29 CP       078  00  00      118  05  5      158  44 SUM     198  67  EQ     238  67  EQ     278  59  59
039  67  EQ      079  92 RTN      119  82 HIR     159  00  00     199  02  02     239  02  02     279  92 RTN
```

CODE CONVERTER.- This routine will convert a five-digit number to alpha code in 6 sec.
---------- The author is Richard Snow. One more candidate for our own module.
A user-defined key, such as A' should be used in the module, rather than LBL PRT. It will save the user one more step when calling the routine. Also, the routine should be a subroutine of a larger routine in the module, maybe of an Alpha Register List ?

```
000  76 LBL      006  01  1       012  48 EXC      018  59 INT      024  00  0       030  59 INT      036  14  14
001  16 A'       007  94 +/-      013  00  00      019  55  ÷       025  00  0       031  44 SUM      037  25 CLR
002  52  EE       008  42 STO      014  85  +       020  05  5       026  49 PRD     032  01  01     038  48 EXC
003  55  ÷       009  00  00      015  93  .       021  54  )       027  01  01     033  97 DSZ     039  01  01
004  52  EE       010  95  =       016  01  1       022  65  ×       028  34 fX      034  00  00     040  92 RTN
005  00  0       011  28 LOG      017  85  +       023  01  1       029  75  -       035  00  00
```

HIR-LIST.- Another candidate for our own module is this HIR-list by Frederic De Mees. It is extermely handy when you are debugging a program or simply trying out different key strokes before writing the actual program and finding out their effect on the HIR registers. Press A to start.

000 76 LBL 001 16 A' 002 69 OP 003 06 06 004 01 1 005 82 HIR 006 38 38 007 92 RTN 008 76 LBL	009 11 A 010 82 HIR 011 18 18 012 32 X:T 013 01 1 014 02 2 015 22 INV 016 28 LOG	017 82 HIR 018 08 08 019 02 2 020 03 3 021 02 2 022 04 4 023 03 3 024 05 5	025 01 1 026 01 1 027 82 HIR 028 38 38 029 82 HIR 030 11 11 031 16 A' 032 82 HIR	033 12 12 034 16 A' 035 82 HIR 036 13 13 037 16 A' 038 82 HIR 039 14 14 040 16 A'	041 82 HIR 042 15 15 043 16 A' 044 82 HIR 045 16 16 046 16 A' 047 82 HIR 048 17 17	049 16 A' 050 32 X:T 051 69 OP 052 06 06 053 82 HIR 054 08 08 055 00 0 056 98 ADV 057 92 RTN

FLAG STATUS.- One more possible inclusion in our own module, this flag status test was written by Dave Leising. As opposed to Bill Beebe's tester, somewhere else in this issue, this one was intended to print the flag status only. Press A for the first time. After that, R/S will do.

000 76 LBL 001 11 A 002 22 INV 003 57 ENG 004 58 FIX 005 09 09 006 09 9 007 93 . 008 01 1	009 48 EXC 010 00 00 011 82 HIR 012 04 04 013 25 CLR 014 82 HIR 015 03 03 016 01 1 017 82 HIR	018 02 02 019 22 INV 020 87 IFF 021 40 IND 022 00 00 023 00 00 024 29 29 025 82 HIR 026 12 12	027 82 HIR 028 33 33 029 01 1 030 00 0 031 82 HIR 032 42 42 033 97 DSZ 034 00 00 035 00 00	036 19 19 037 82 HIR 038 14 14 039 42 STO 040 00 00 041 01 1 042 02 2 043 03 3 044 04 4	045 05 5 046 06 6 047 07 7 048 08 8 049 09 9 050 98 ADV 051 99 PRT 052 82 HIR 053 13 13	054 99 PRT 055 98 ADV 056 98 ADV 057 98 ADV 058 25 CLR 059 92 RTN 060 61 GTO 061 00 00 062 02 02

EE/ENG TESTER.- Michael Sperber wrote this routine because PGM 20 of the MU module doesn't have one. Besides giving a -1 for INV ENG, a 0 for ENG and a 1 for EE mode, you can force the calculator into one of thses modes by entering the appropriate -1, 0 or 1 and pressing B. Testing for any of the modes is done by pressing A.

The disadvantage of this routine is, that it changes any FIX format to FIX 7. On the other hand, you cannot use MU-20 B' to save the FIX format, because it contains the sequence EE INV EE, and that would spoil things. So Michael added a small FIX format tester which works allright for FIX 0 through FIX 6 and INV FIX, but not for FIX 7 nor FIX 8. They are regarded as INV FIX by the routine.

000 76 LBL 001 11 A 002 53 (003 07 7 004 -32 X:T 005 89 π 006 75 - 007 88 DMS 008 22 INV	009 88 DMS 010 54) 011 50 I×I 012 28 LOG 013 50 I×I 014 22 INV 015 77 GE 016 22 INV	017 09 9 018 76 LBL 019 22 INV 020 42 STO 021 00 00 022 58 FIX 023 07 07 024 01 1	025 00 0 026 69 OP 027 04 04 028 82 HIR 029 18 18 030 50 I×I 031 28 LOG 032 53 (	033 59 INT 034 85 + 035 04 4 036 54) 037 94 +/- 038 58 FIX 039 40 IND 040 00 00	041 92 RTN 042 76 LBL 043 12 B 044 29 CP 045 57 ENG 046 67 EQ 047 52 EE 048 22 INV	049 57 ENG 050 22 INV 051 77 GE 052 52 EE 053 52 EE 054 76 LBL 055 52 EE 056 92 RTN

QUIRK- John Miranda (the youngest member, I suppose, at 14) found this quirk in the TI-58. I tested it and found it to work in the TI-59 as well. With the printer attched, it puts the calculator in TRACE mode. RST kills this state.
From turn-on press:

CP LRN LBL A FIX 0 R/S LRN 999 + .00000999 = A STF IND 7 INV LRN

The display nows says 9993 00. You can SST through it to 99999 when it reverts to step 000 and you see your program LBL A FIX 0 R/S again.

If you BST, you loose the display, but stay with the TRACE mode. But it is impossible to go back into LRN at this point, unless you press GTO 000 then LRN.
The "listing" of your program you now get, if you press SST repeatedly, is SR-52-like, no mnemonics, just the program step and the key code.
Further investigating invited from the membership.

TWIN-T FILTER DESIGN.- In v5n3p8 we brought you, as part of the decimal point article, a program by that name. In fact, there were two versions: one for those in a hurry, a calculator-only version. The other one had all the bells and whistles,(at least I thought it had, until I saw the present program) and it printed the results with descriptors in the right hand margin.

The idea for the original program came from Robert Trost one of our members in Holland. Robert wrote me, asking if I had any idea how to mechanize the twin-T filter, as he was using it often in designs of devices for the handicapped.

Once I had the program developed and Robert had his copy I decided to put it also in the notes. Why not let everybody have a crack at it. And that is what Ruud Paap from Papendrecht, a compatriot of Robert, did. His program is a real gem. It is one of those interactive routines that guide users all the way.Only a true "programmer at heart" could have produced such a jewel.

The price you always pay for this adherence to detail is a long program, but not necessarily a slow one. Ruud used direct addresses throughout, for speed and the execution time is within very reasonable limits.

It further fits on two mag cards and is written in a typical "mainframe-programmer" style: you write a bunch of subroutines, then you link everything together with a main routine. Even on the 59 that approach can be very effective, as this program proves it.

You key in the program and load the print code data registers in 4 OP 17 partitioning. Then you record four card sides in 6 OP 17 partitioning.

To use the program:

1. *Load all four card sides in 6 OP 17 partitioning.*
2. *If you know the frequency, press A.*
   If you know the R and C value, press D.
   If you know the R and the frequency, press E.
   If you know the C and the frequency, press E'.
   In any of the four cases, the schematic diagram is drawn. Then the first prompt appears on the printer. Enter the requested value and press R/S.
   The last prompt is always the bandwidth. After you press R/S again, the calculator computes and the printer prints the remaining unknown values.
   R is always entered in kilo-ohms and C in nanofarads. (sure, you must remember good, old nanofarad?)
3. *For a new calculation, go back to 2.*

Happy filtering,

```
TWIN-T-FILTER DESIGN

    --R1----+-R2--
  -:        :    :-
   --C1----+-C2--
         :  :
        R3  C3
         :  :
        ------

FREQUENCY IN KHZ?
       32.000
BANDWITH 1-100?
           1.

FREQUENCY:
       32.000      KHZ

RESISTORS:
        R1:
        2.230     KOHM
        R2:
        2.230     KOHM
        R3:
        1.115     KOHM

CAPACITORS:
        C1:
        2.230       NF
        C2:
        2.230       NF
        C3:
        4.460       NF



TWIN-T-FILTER DESIGN

    --R1----+-R2--
  -:        :    :-
   --C1----+-C2--
         :  :
        R3  C3
         :  :
        ------

RESISTANCE IN KOHM?
        2.200
CAPACITY IN NF?
        1.000
BANDWITH 1-100?
           1.

FREQUENCY:
       72.343      KHZ

RESISTORS:
        R1:
        2.200     KOHM
        R2:
        2.200     KOHM
        R3:
        1.100     KOHM

CAPACITORS:
        C1:
        1.000       NF
        C2:
        1.000       NF
        C3:
        2.000       NF
```

TWIN-T FILTER, Ruud Paap.

```
000  06   6
001  02   2
002  58 FIX
003  02   02
004  69 OP
005  02   02
006  58 FIX
007  06   06
008  76 LBL
009  81 RST
010  69 OP
011  03   03
012  69 OP
013  05   05
014  69 OP
015  00   00
016  22 INV
017  58 FIX
018  92 RTN
019  76 LBL
020  71 SBR
021  25 CLR
022  91 R/S
023  58 FIX
024  03   03
025  99 PRT
026  22 INV
027  58 FIX
028  65   ×
029  01   1
030  00   0
031  00   0
032  00   0
033  95   =
034  92 RTN
035  02   2
036  00   0
037  02   2
038  00   0
039  69 OP
040  01   01
041  87 IFF
042  00   00
043  00   00
044  51   51
045  02   2
046  52 EE
047  09   9
048  22 INV
049  52 EE
050  85   +
051  43 RCL
052  10   10
053  95   =
054  69 OP
055  02   02
056  87 IFF
057  00   00
058  00   00
059  43 RCL
060  39   39
061  85   +
062  43 RCL
063  11   11
064  95   =
065  69 OP
066  03   03
067  02   2
068  00   0
069  02   2
070  58 FIX
071  07   07
072  69 OP
073  04   04
074  69 OP
075  05   05
076  69 OP
077  00   00
078  25 CLR
079  22 INV
080  58 FIX
081  92 RTN
082  04   4
083  69 OP
084  17   17
085  69 OP
086  00   00
087  43 RCL
088  12   12
089  69 OP
090  01   01
091  43 RCL
092  13   13
093  69 OP
094  02   02
095  43 RCL
096  14   14
097  69 OP
098  03   03
099  43 RCL
100  15   15
101  71 SBR
102  00   00
103  73   73
104  98 ADV
105  71 SBR
106  00   00
107  35   35
108  43 RCL
109  38   38
110  69 OP
111  01   01
112  06   6
113  02   2
114  58 FIX
115  06   06
116  69 OP
117  03   03
118  06   6
119  02   2
120  02   2
121  58 FIX
122  05   05
123  71 SBR
124  00   00
125  73   73
126  86 STF
127  00   00
128  71 SBR
129  00   00
130  35   35
131  71 SBR
132  00   00
133  00   00
134  03   3
135  05   5
136  00   0
137  04   4
138  58 FIX
139  02   02
140  69 OP
141  02   02
142  75   -
143  43 RCL
144  39   39
145  95   =
146  58 FIX
147  04   04
148  71 SBR
149  81 RST
150  71 SBR
151  00   00
152  00   00
153  43 RCL
154  16   16
155  69 OP
156  02   02
157  58 FIX
158  06   06
159  71 SBR
160  81 RST
161  22 INV
162  86 STF
163  00   00
164  98 ADV
165  92 RTN
166  69 OP
167  00   00
168  43 RCL
169  17   17
170  69 OP
171  01   01
172  43 RCL
173  18   18
174  69 OP
175  02   02
176  92 RTN
177  71 SBR
178  01   01
179  67   67
180  43 RCL
181  19   19
182  69 OP
183  03   03
184  43 RCL
185  20   20
186  61 GTO
187  00   00
188  73   73
189  69 OP
190  00   00
191  43 RCL
192  21   21
193  69 OP
194  01   01
195  43 RCL
196  22   22
197  69 OP
198  02   02
199  92 RTN
200  69 OP
201  00   00
202  43 RCL
203  24   24
204  69 OP
205  01   01
206  43 RCL
207  25   25
208  69 OP
209  02   02
210  92 RTN
211  71 SBR
212  02   02
213  01   01
214  75   -
215  06   6
216  02   2
217  95   =
218  69 OP
219  02   02
220  43 RCL
221  26   26
222  69 OP
223  03   03
224  43 RCL
225  27   27
226  61 GTO
227  00   00
228  73   73
229  65   ×
230  02   2
231  65   ×
232  89   π
233  95   =
234  36 1/X
235  92 RTN
236  95   =
237  69 OP
238  06   06
239  71 SBR
240  02   02
241  55   55
242  43 RCL
243  28   28
244  69 OP
245  04   04
246  58 FIX
247  03   03
248  93   .
249  00   0
250  00   0
251  01   1
252  65   ×
253  92 RTN
254  22 INV
255  58 FIX
256  01   1
257  00   0
258  00   0
259  44 SUM
260  09   09
261  43 RCL
262  09   09
263  69 OP
264  00   00
265  69 OP
266  02   02
267  69 OP
268  05   05
269  92 RTN
270  95   =
271  69 OP
272  06   06
273  71 SBR
274  02   02
275  55   55
276  03   3
277  01   1
278  02   2
279  01   1
280  69 OP
281  04   04
282  58 FIX
283  03   03
284  01   1
285  52 EE
286  09   9
287  22 INV
288  52 EE
289  65   ×
290  92 RTN
291  76 LBL
292  11   A
293  71 SBR
294  00   00
295  83   83
296  71 SBR
297  02   02
298  12   12
299  71 SBR
300  30   30
301  71 SBR
302  42 STO
303  02   02
304  71 SBR
305  02   02
306  30   30
307  34 FX
308  65   ×
309  01   1
310  52 EE
311  06   6
312  22 INV
313  52 EE
314  95   =
315  42 STO
316  00   00
317  43 RCL
318  00   00
319  65   ×
320  43 RCL
321  02   02
322  71 SBR
323  02   02
324  30   30
325  42 STO
326  01   01
327  61 GTO
328  04   04
329  55   55
330  76 LBL
331  12   B
332  71 SBR
333  01   01
334  78   78
335  71 SBR
336  71 SBR
337  48 EXC
338  00   00
339  65   ×
340  43 RCL
341  01   01
342  55   -
343  43 RCL
344  00   00
345  95   =
346  42 STO
347  01   01
348  61 GTO
349  04   04
350  55   55
351  76 LBL
352  13   C
353  71 SBR
354  06   06
355  10   10
356  48 EXC
357  01   01
358  65   ×
359  43 RCL
360  00   00
361  55   -
362  43 RCL
363  01   01
364  95   =
365  42 STO
366  00   00
367  61 GTO
368  04   04
369  55   55
370  76 LBL
371  14   D
372  71 SBR
373  00   00
374  83   83
375  71 SBR
376  01   01
377  78   78
378  71 SBR
379  71 SBR
380  42 STO
381  00   00
382  71 SBR
383  06   06
384  10   10
385  42 STO
386  01   01
387  65   ×
388  43 RCL
389  00   00
390  71 SBR
391  02   02
392  30   30
393  42 STO
394  02   02
395  61 GTO
396  04   04
397  55   55
398  76 LBL
399  15   E
400  71 SBR
401  00   00
402  83   83
403  71 SBR
404  01   01
405  78   78
406  71 SBR
407  71 SBR
408  42 STO
409  00   00
410  71 SBR
411  02   02
412  12   12
413  71 SBR
414  71 SBR
415  42 STO
416  02   02
417  65   ×
418  43 RCL
419  00   00
420  71 SBR
421  02   02
422  30   30
423  42 STO
424  01   01
425  61 GTO
426  04   04
427  55   55
428  76 LBL
429  10   E'
430  71 SBR
431  00   00
432  83   83
433  71 SBR
434  06   06
435  10   10
436  42 STO
437  01   01
438  71 SBR
439  02   02
440  12   12
441  71 SBR
442  71 SBR
443  42 STO
444  02   02
445  65   ×
446  43 RCL
447  01   01
448  71 SBR
449  02   02
450  30   30
451  42 STO
452  00   00
453  76 LBL
454  16   A'
455  69 OP
456  00   00
457  43 RCL
458  39   39
459  69 OP
460  01   01
461  43 RCL
462  30   30
463  69 OP
464  02   02
465  43 RCL
466  31   31
467  71 SBR
468  31 RST
469  25 CLR
470  91 R/S
471  99 PRT
472  98 ADV
473  42 STO
474  03   03
475  55   ÷
476  53   (
477  24 CE
478  85   +
479  01   1
480  95   =
481  42 STO
482  04   04
483  43 RCL
484  00   00
485  65   ×
486  43 RCL
487  03   03
488  95   =
489  42 STO
490  05   05
491  43 RCL
492  00   00
493  65   ×
494  43 RCL
495  04   04
496  95   =
497  42 STO
498  06   06
499  43 RCL
500  01   01
501  55   ÷
502  43 RCL
503  03   03
504  95   =
505  42 STO
506  07   07
507  43 RCL
508  01   01
509  55   ÷
510  43 RCL
511  04   04
512  95   =
513  42 STO
514  08   08
515  71 SBR
516  02   02
517  01   01
518  69 OP
519  05   05
520  43 RCL
521  32   32
522  59 INT
523  04   04
524  43 RCL
525  02   02
526  65   ×
527  93   .
528  00   0
529  00   0
530  01   1
531  95   =
532  58 FIX
533  03   03
534  69 OP
535  06   06
536  98 ADV
537  71 SBR
538  01   01
539  67   67
540  43 RCL
541  33   33
542  71 SBR
543  02   02
544  66   66
545  43 RCL
546  34   34
547  42 STO
548  09   09
549  71 SBR
550  02   02
551  40   40
552  43 RCL
553  00   00
554  71 SBR
555  02   02
556  37   37
557  43 RCL
558  05   05
559  71 SBR
560  02   02
561  37   37
562  43 RCL
563  06   06
564  95   =
565  69 OP
566  06   06
567  22 INV
568  58 FIX
569  98 ADV
570  71 SBR
571  01   01
572  90   90
573  43 RCL
574  35   35
575  69 OP
576  02   02
577  43 RCL
578  36   36
579  71 SBR
580  81 RST
581  43 RCL
582  37   37
583  42 STO
584  09   09
585  71 SBR
586  02   02
587  74   74
588  43 RCL
589  01   01
590  71 SBR
591  02   02
592  71   71
593  43 RCL
594  07   07
595  71 SBR
596  02   02
597  71   71
598  43 RCL
599  08   08
600  95   =
601  69 OP
602  06   06
603  22 INV
604  58 FIX
605  25 CLR
606  98 ADV
607  98 ADV
608  98 ADV
609  91 R/S
610  71 SBR
611  01   01
612  90   90
613  43 RCL
614  33   33
615  71 SBR
616  81 RST
617  71 SBR
618  71 SBR
619  55   ÷
620  01   1
621  52 EE
622  01   1
623  02   2
624  22 INV
625  52 EE
626  95   =
627  92 RTN
```

```
150462.          09
1502202020.      10
2047201503.      11
3743243120.      12
3720212427.      13
3717350016.      14
1736242231.      15
   202020.       16
3517362436.      17
3713311517.      18
  24310026.      19
3223307100.      20
1513331315.      21
2437450024.      22
3100312171.      23
2135173441.      24
1731154562.      25
2431002623.      26
4671000000.      27
  26322330.      28
1413311643.      29
2437220002.      30
2002010171.      31
   262346.       32
3732353662.      33
   350162.       34
2437323536.      35
6200000000.      36
   150162.       37
   206200.       38
     2000.       39
```

SPOCK, BEAMING ABOARD.- The "idle doodle" has become a program. The reponse was over-whelming. But of all the programs, the one by Serge Borodin was judged a "hands down" winner. A quick and dirty way of beaming Spock aboard in just under 53 seconds, not counting the mid-waist stop to reload beam tapes.

Instructions are very simple: Key in the top half of Spock in 6 OP 17. Don't forget the 60 alpha registers. Record it on two cards. Then key in Spock's bottom half, same partition, also on two cards, four sides.

To use it, read in Spock's top half, press RST R/S and the drawing starts. When it stops around Spock's waist, read in the two remaining cards, press RST and R/S again to complete the drawing.

On the off chance that one of our foreign members is utterly puzzled and doesn't know what I am talking about: For a long time there was a fantastic shown running on American TV called Star Trek. It dealt with an international crew and their adventures aboard a star ship called Enterprise. One of the most colorful characters among that crew was Spock, the intelligence officer. Spock was a Vulcan and had rather pointed ears and no emotions whatsoever. In fact, Spock was the epitome of logic. One of the favored excursions of the senior crew members was visiting alien planets on their way. Their way of traveling from the Enterprise to the planet and back was by "beaming", a sort of "we-desintegrate-you-then-we-compose-you-again" scheme. The TV viewer was usually shown a slowly fading picture of three crew members (always Spock among them) while the background remained perfectly intact. The same thing, in reverse, then happened on the planet. In our picture here Spock is almost back to normal shape again, in the process of being "beamed aboard."

SEE THE PROGRAMS ON THE NEXT TWO PAGES.

ALPHA + NUMERIC DATA REGISTER.- Re-v6n2p10. Michael Sperber says about this program: "This program prints non-significant zeros if: a. log X = -3, b. the 12th digit of X is zero (leading zeros not counted as digits, c. the 13th digit of X is not zero.("")

You may print a maximum of up to 11 non-signifiant zeros if you enter this number: (0.001 + 1 EE 15 +/- INV EE)

The explanation? The program has 15 places to print X in. One place is reserved for the decimal point, remain 14. The program switches to EE-printing if X is smaller than or equal to 0.001, which means that a maximum of two 'leading' zeros after the decimal point may take away another two places. Remain 14 - 2 = 12 places for a 13-digit number. That means that the number cannot be printed fully. Now, you might ask, why does the program not stop after having '0.001' printed if it cannot print the '0.001000000000001' ?

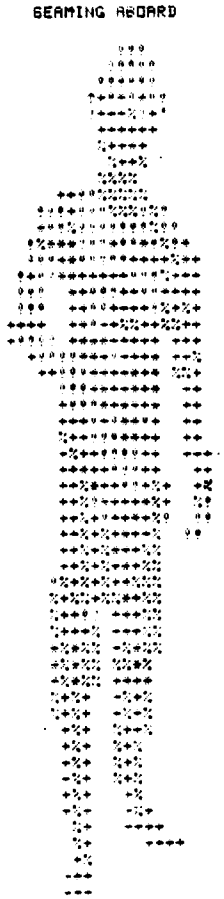
That is exactly the quirk in the program. The program stops printing (printing here means 'converting to print codes and storing in OP's) a number, if the number is zero.(program steps 217-222) That means that the program does convert the 16th digit (you may recall it from Reg 66) but there is a safeguard in the program to prevent the register number in OP 04 from being cleared by the 16th digit of X (program steps 235-241)

You might ask why I choose just these three conditions a, b and c. Condition a is to prevent the program from switching into EE-printing, because the EE-mode-sensor in program steps 115-121 becomes active only if X is equal to or smaller than 1E-3 and to use a maximum of two digits taken away. Condition b is to make sure that the last digit is zero. Otherwise it would not be non-significant. Condition c is to make sure that the program does convert the non-significant zero, because X ≠ 0.

Of course, you could cure this effect by changing program step 117 or you could change condition a to log X = -2 for negative numbers, etc. etc."

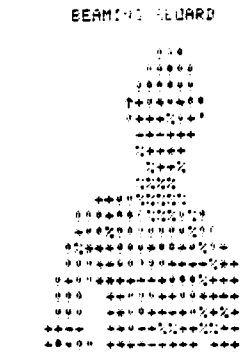
TI-59 RESERVOIR ENGINEERING MANUAL.- by Reuven Hollo, Haresh Fidara Keplinger Associates, 220 pages, 1980, $ 50.00, Pennwell Books, P.O. Box 21288, Tulsa, Oklahoma, 74121. Toll free number 800-331-4429.
Contains 27 reservoir engineering programs for the TI-59. Has all the necessary equations, flow charts, examples, and program listings. Contents: Basic Reservoir Engineering, Economic Analysis, Geology, Log Interpretation, Advanced Reservoir Engineering.

SPOCK, top half, by Serge Borodin




```
000  69 ΣP
001  00  00
002  43 RCL
003  01  01
004  69 ΣP
005  01  01
006  43 RCL
007  02  02
008  69 ΣP
009  02  02
010  43 RCL
011  03  03
012  69 ΣP
013  03  03
014  43 RCL
015  04  04
016  69 ΣP
017  04  04
018  69 ΣP
019  05  05
020  99 ADV
021  69 ΣP
022  00  00
023  43 RCL
024  05  05
025  69 ΣP
026  03  03
027  69 ΣP
028  05  05
029  43 RCL
030  06  06
031  69 ΣP
032  03  03
033  69 ΣP
034  05  05
035  43 RCL
036  07  07
037  69 ΣP
038  02  02
039  43 RCL
040  08  08
041  69 ΣP
042  03  03
043  69 ΣP
044  05  05
045  43 RCL
046  09  09
047  69 ΣP
048  02  02
049  43 RCL
050  10  10
051  69 ΣP
052  03  03
053  43 RCL
054  11  11
055  69 ΣP
056  04  04
057  69 ΣP
058  05  05
059  43 RCL
060  12  12
061  69 ΣP
062  02  02
063  43 RCL
064  13  13
065  69 ΣP
066  03  03
067  43 RCL
068  14  14
069  69 ΣP
070  04  04
071  69 ΣP
072  05  05
073  69 ΣP
074  00  00
075  43 RCL
076  15  15
077  69 ΣP
078  02  02
079  43 RCL
080  16  16
081  69 ΣP
082  03  03
083  69 ΣP
084  05  05
085  43 RCL
086  17  17
087  69 ΣP
088  02  02
089  43 RCL
090  18  18
091  69 ΣP
092  03  03
093  69 ΣP
094  05  05
095  69 ΣP
096  00  00
097  43 RCL
098  19  19
099  69 ΣP
100  03  03
101  69 ΣP
102  05  05
103  43 RCL
104  20  20
105  69 ΣP
106  02  02
107  43 RCL
108  21  21
109  69 ΣP
110  03  03
111  69 ΣP
112  05  05
113  43 RCL
114  22  22
115  69 ΣP
116  02  02
117  43 RCL
118  23  23
119  69 ΣP
120  03  03
121  69 ΣP
122  05  05
123  43 RCL
124  24  24
125  69 ΣP
126  01  01
127  43 RCL
128  25  25
129  69 ΣP
130  02  02
131  43 RCL
132  26  26
133  69 ΣP
134  03  03
135  43 RCL
136  27  27
137  69 ΣP
138  04  04
139  69 ΣP
140  05  05
141  43 RCL
142  28  28
143  69 ΣP
144  01  01
145  43 RCL
146  29  29
147  69 ΣP
148  02  02
149  43 RCL
150  30  30
151  69 ΣP
152  03  03
153  43 RCL
154  31  31
155  69 ΣP
156  04  04
157  69 ΣP
158  05  05
159  43 RCL
160  32  32
161  69 ΣP
162  01  01
163  43 RCL
164  33  33
165  69 ΣP
166  02  02
167  43 RCL
168  34  34
169  69 ΣP
170  03  03
171  43 RCL
172  35  35
173  69 ΣP
174  04  04
175  69 ΣP
176  05  05
177  43 RCL
178  36  36
179  69 ΣP
180  01  01
181  43 RCL
182  37  37
183  69 ΣP
184  02  02
185  43 RCL
186  38  38
187  69 ΣP
188  03  03
189  43 RCL
190  39  39
191  69 ΣP
192  04  04
193  69 ΣP
194  05  05
195  43 RCL
196  40  40
197  69 ΣP
198  01  01
199  43 RCL
200  41  41
201  69 ΣP
202  02  02
203  43 RCL
204  42  42
205  69 ΣP
206  03  03
207  43 RCL
208  43  43
209  69 ΣP
210  04  04
211  69 ΣP
212  05  05
213  43 RCL
214  44  44
215  69 ΣP
216  01  01
217  43 RCL
218  45  45
219  69 ΣP
220  02  02
221  43 RCL
222  46  46
223  69 ΣP
224  03  03
225  43 RCL
226  47  47
227  69 ΣP
228  04  04
229  69 ΣP
230  05  05
231  43 RCL
232  48  48
233  69 ΣP
234  01  01
235  43 RCL
236  49  49
237  69 ΣP
238  02  02
239  43 RCL
240  50  50
241  69 ΣP
242  03  03
243  43 RCL
244  51  51
245  69 ΣP
246  04  04
247  69 ΣP
248  05  05
249  43 RCL
250  52  52
251  69 ΣP
252  01  01
253  43 RCL
254  53  53
255  69 ΣP
256  02  02
257  43 RCL
258  54  54
259  69 ΣP
260  03  03
261  43 RCL
262  55  55
263  69 ΣP
264  04  04
265  69 ΣP
266  05  05
267  43 RCL
268  56  56
269  69 ΣP
270  01  01
271  43 RCL
272  57  57
273  69 ΣP
274  02  02
275  43 RCL
276  58  58
277  69 ΣP
278  03  03
279  43 RCL
280  59  59
281  69 ΣP
282  04  04
283  69 ΣP
284  05  05
285  91 R/S
286  00  0
287  00  0
288  00  0
```

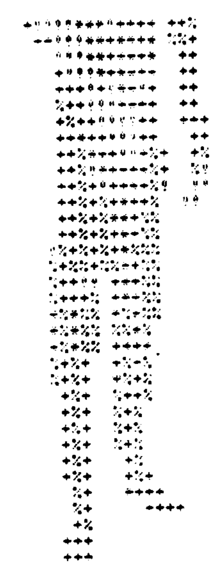
```
                0.  00
             1417.  01
       1330243122.  02
         13143213.  03
       3516000000.  04
          73737300.  05
       7373737373.  06
             73.  07
       7373737373.  08
             6047.  09
       7351735173.  10
       7300000000.  11
             7347.  12
       4747617347.  13
       6500000000.  14
               47.  15
       4747474747.  16
               61.  17
       4747514700.  18
       6147476100.  19
               61.  20
       6161610000.  21
       4747737361.  22
       6161616100.  23
             7373.  24
       7347737373.  25
       6161617361.  26
       7300000000.  27
             5173.  28
       7361737373.  29
       7373737361.  30
       7373000000.  31
             7361.  32
       5151737373.  33
       5173735173.  34
       6173510000.  35
          737373.  36
       5151737373.  37
       7373474747.  38
       4761514700.  39
          73477373.  40
       5151474747.  41
       4747737373.  42
       6147474700.  43
          73737300.  44
          51477373.  45
       7347477373.  46
       5147474700.  47
          73737300.  48
          47517373.  49
       4747474773.  50
       6147614700.  51
       4747474700.  52
          4751737.  53
       4761614747.  54
       6161474700.  55
       4773737373.  56
          47515151.  57
       4747474747.  58
          47474700.  59
```



METALWORKING- Cybermation MPS-10 Manual Programming System for NC metalworking. Pro-
---------- grammable calculator makes it easy to convert engineering drawings to punched tape. Eleven pre-programmed cards for the TI-59 are available:
Line-line intersection, line-circle intersection, circle-circle intersection, Corner rounding of intersecting straight lines, corner rounding of intersecting straight line and circle, corner rounding of intersecting circles, tangents to circles, circle through two points with given start and finish direction, tangent from point to circle, definition of circle through three given points, curve-fitting program.
Please write for more information to: Cybermation Inc. 377 Putnam Ave,, Cambridge MA, 02139, tel. (617) 492-8810.

SPOCK, bottom half, by Serge Borodin.

```
000  04  4
001  07  7
002  07  7
003  03  3
004  07  7
005  03  3
006  69 OP
007  01  01
008  07  7
009  03  3
010  07  7
011  03  3
012  05  5
013  01  1
014  05  5
015  01  1
016  05  5
017  01  1
018  69 OP
019  02  02
020  43 RCL
021  58  58
022  69 OP
023  03  03
024  43 RCL
025  57  57
026  69 OP
027  04  04
028  69 OP
029  05  05
030  43 RCL
031  59  59
032  69 OP
033  01  01
034  43 RCL
035  56  56
036  69 OP
037  02  02
038  43 RCL
039  55  55
040  69 OP
041  03  03
042  43 RCL
043  54  54
044  69 OP
045  04  04
046  69 OP
047  05  05
048  69 OP
049  00  00
050  43 RCL
051  01  01
052  69 OP
053  02  02
054  43 RCL
055  02  02
056  69 OP
057  03  03
058  43 RCL
059  03  03
060  69 OP
061  04  04
062  69 OP
063  05  05
064  43 RCL
065  04  04
066  69 OP
067  02  02
068  43 RCL
069  02  02
070  69 OP
071  03  03
072  43 RCL
073  03  03
074  69 OP
075  04  04
076  69 OP
077  05  05
078  43 RCL
079  05  05
080  69 OP
081  02  02
082  43 RCL
083  06  06
084  69 OP
085  03  03
086  43 RCL
087  03  03
088  69 OP
089  04  04
090  69 OP
091  05  05
092  43 RCL
093  07  07
094  69 OP
095  02  02
096  43 RCL
097  08  08
098  69 OP
099  03  03
100  43 RCL
101  03  03
102  69 OP
103  04  04
104  69 OP
105  05  05
106  43 RCL
107  09  09
108  69 OP
109  02  02
110  43 RCL
111  10  10
112  69 OP
113  03  03
114  43 RCL
115  11  11
116  69 OP
117  04  04
118  69 OP
119  05  05
120  43 RCL
121  12  12
122  69 OP
123  02  02
124  43 RCL
125  13  13
126  69 OP
127  04  04
128  69 OP
129  05  05
130  43 RCL
131  14  14
132  69 OP
133  02  02
134  43 RCL
135  15  15
136  69 OP
137  03  03
138  43 RCL
139  16  16
140  69 OP
141  04  04
142  69 OP
143  05  05
144  43 RCL
145  17  17
146  69 OP
147  02  02
148  43 RCL
149  18  18
150  69 OP
151  03  03
152  43 RCL
153  19  19
154  69 OP
155  04  04
156  69 OP
157  05  05
158  43 RCL
159  20  20
160  69 OP
161  02  02
162  43 RCL
163  21  21
164  69 OP
165  03  03
166  43 RCL
167  22  22
168  69 OP
169  04  04
170  69 OP
171  05  05
172  43 RCL
173  23  23
174  69 OP
175  02  02
176  43 RCL
177  24  24
178  69 OP
179  03  03
180  43 RCL
181  25  25
182  69 OP
183  04  04
184  69 OP
185  05  05
186  69 OP
187  00  00
188  43 RCL
189  23  23
190  69 OP
191  02  02
192  43 RCL
193  26  26
194  69 OP
195  03  03
196  69 OP
197  05  05
198  43 RCL
199  27  27
200  69 OP
201  03  03
202  69 OP
203  05  05
204  43 RCL
205  28  28
206  69 OP
207  01  01
208  43 RCL
209  29  29
210  69 OP
211  02  02
212  43 RCL
213  30  30
214  69 OP
215  03  03
216  69 OP
217  05  05
218  43 RCL
219  31  31
220  69 OP
221  01  01
222  43 RCL
223  32  32
224  69 OP
225  02  02
226  43 RCL
227  33  33
228  69 OP
229  03  03
230  69 OP
231  05  05
232  43 RCL
233  34  34
234  69 OP
235  02  02
236  43 RCL
237  27  27
238  69 OP
239  03  03
240  69 OP
241  05  05
242  43 RCL
243  35  35
244  69 OP
245  02  02
246  69 OP
247  05  05
248  43 RCL
249  36  36
250  69 OP
251  01  01
252  43 RCL
253  37  37
254  69 OP
255  02  02
256  43 RCL
257  38  38
258  69 OP
259  03  03
260  69 OP
261  05  05
262  43 RCL
263  39  39
264  69 OP
265  03  03
266  69 OP
267  05  05
268  43 RCL
269  40  40
270  69 OP
271  03  03
272  69 OP
273  05  05
274  43 RCL
275  31  31
276  69 OP
277  01  01
278  43 RCL
279  41  41
280  69 OP
281  02  02
282  43 RCL
283  42  42
284  69 OP
285  03  03
286  69 OP
287  05  05
288  69 OP
289  05  05
290  69 OP
291  00  00
292  43 RCL
293  41  41
294  69 OP
295  02  02
296  43 RCL
297  43  43
298  69 OP
299  03  03
300  69 OP
301  05  05
302  43 RCL
303  44  44
304  69 OP
305  03  03
306  69 OP
307  05  05
308  69 OP
309  05  05
310  69 OP
311  05  05
312  43 RCL
313  45  45
314  69 OP
315  03  03
316  69 OP
317  05  05
318  43 RCL
319  46  46
320  69 OP
321  03  03
322  69 OP
323  05  05
324  43 RCL
325  47  47
326  69 OP
327  02  02
328  43 RCL
329  48  48
330  69 OP
331  03  03
332  69 OP
333  05  05
334  43 RCL
335  49  49
336  69 OP
337  03  03
338  43 RCL
339  50  50
340  69 OP
341  04  04
342  69 OP
343  05  05
344  69 OP
345  00  00
346  43 RCL
347  45  45
348  69 OP
349  02  02
350  69 OP
351  05  05
352  43 RCL
353  51  51
354  69 OP
355  02  02
356  69 OP
357  05  05
358  69 OP
359  05  05
360  98 ADV
361  98 ADV
362  43 RCL
363  52  52
364  69 OP
365  02  02
366  43 RCL
367  53  53
368  69 OP
369  03  03
370  69 OP
371  05  05
372  98 ADV
373  98 ADV
374  98 ADV
375  91 R/S
376  00  0
377  00  0
```



SPOCK

```
          0.   00
 7373735151.   01
 4751514751.   02
     474700.   03
 4773737351.   04
 4747477347.   05
 7351517347.   06
 6147477373.   07
 7351514747.   08
 4761474773.   09
 7373734747.   10
     474747.   11
 4747514747.   12
       4747.   13
 4747615147.   14
 4773734761.   15
 4700004761.   16
 4747617351.   17
 4747475161.   18
 4700006173.   19
 4747614773.   20
 4747514761.   21
 7300007373.   22
 4747614761.   23
 4751474761.   24
     737300.   25
 5151476161.   26
 4751476161.   27
        73.   28
 6147614761.   29
 4751616161.   30
         61.   31
 4761614761.   32
 6151476161.   33
 4747737300.   34
 4747476100.   35
         51.   36
 6151616100.   37
 5161516161.   38
 6161516100.   39
 4751475100.   40
 4761470000.   41
 4761476100.   42
 6147476100.   43
 6147610000.   44
   47610000.   45
   47614700.   46
   61470000.   47
   51474747.   48
       4747.   49
 4747000000.   50
 4747470000.   51
       3633.   52
 3215260000.   53
   61614700.   54
 4751514751.   55
 7373735151.   56
   47476100.   57
 7347474751.   58
       4747.   59
```

RANDOM INTEGER PROGRAM.- Re-v6n1p9. Michael Cassidy, a psychologist with the Oceanside California Unified School District tells me that he had been looking for a long time for such a program and that it had him "stomped" too.
He uses these programs in education to sample student populations when evaluating the effectiveness of various projects and interventions.

CALCULATOR STATUS ROUTINES.- Somewhere else in this issue Bill Beebe Jr. presents his calculator status routines. Michael Sperber, who reviewed them, tells me that they are so "tight" they leave very little room for improvement. The only thing he suggests: subsitute steps EE INV EE by means of DMS INV DMS. It is slower but a little friendlier towards the user, beacause it doesn't interfere with the user's choice of display format.

A-MAZE-ING.- TI has a new module for the TI-99/4 by that name. It allows you to play
---------- the role of a mouse in a maze, persued by one or two cats, standard or smart ones, pouncing at slow, medium or high rate. You may either try to escape or go on a cheese hunt, etc. etc. In my opinion, this is one of the best games TI has ever written. It appeals to young and old and is really action packed.

This brought to mind the MAZE program I acquired once. I played with it for a while, rewrote it, played some more, gave copies of it to the local members in the hope they would improve upon it (no dice) and finally gave up in disgust. It produces beautiful mazes in acceptable periods of time. Then, suddenly, without warning, it produces "duds." (mazes without connection between entry and exit.)

So, fellas, here is your chance. Try to find out what is wrong with it and make it work. I know you all like challenges, witness the enormous response I got when I said that no program existed to beam in Spock.

Instructions: Key in the 720 program steps with partitioning 3 OP 17. Re-partition to 6 OP 17 and record three card sides.

To use the program: 1. Read in card sides 1, 2 and 3 in turn-on partitioning.
2. Initialize by pressing E. You will see 719.29 in the display.
3. Enter the number of mazes you want, from 1 through 9, and press A.
4. Enter the number of lines for the first maze.(think about it: each line, in spite of direct addressing, still takes about 90 sec to compute and print) press R/S.
5. Enter the number of lines for the second maze, followed by R/S, etc. etc. The display will show each time the number of mazes still left to be specified. If you enter too much, the display will flash. No harm done, however. Just press CLR to stop the flashing.
6. Enter a seed, any number, and press B.
7. Start program by pressing C. NOW, HAVE PATIENCE !

Parameters that may be changed in the program:

Address	Effect	Range	Preset in program
130-131	printing symbol	01 to 99	32 (square 0)
259-260	more cross pieces (⌈¯ T ¯⌉)	01 to 99	60
304	less cross pieces (T)	0 to 9	5
430	longer horizontal pieces	1 to 9	7
472-473	less cross pieces (L T J)	01 to 99	20
671-672	shorter solution paths	00 to 99	80

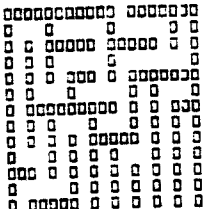
You will find the program listing and a few samples on the next page.

TAPE MEASURE.- I admit, this is a "crazy" program. It was written by another one of
------------ our younger members, Jeff Rosedale. It will provide you with approximate pieces of tape of a length you enter. A is for inches, B is for cm, C is for feet, and D is for meters. Enter the required length and press either A, B, C or D. Enter, for example, 25 and press D. See your paper supply shoot out of your printer.

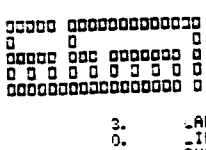
```
.2304311512   02
4040404040.   03
    243115.   04
2017360000.   05
      3117.   06
1737000000.   07
1517393724.   08
2817371735.   09
    301737.   10
1735000000.   11

000  76 LBL     007  02  2      014  42 STO     034  30  30     054  95  =      074  12  12     094  12  12
001  16 A'      008  95  =      015  01  01     035  69 OP      055  16 A'      075  61 GTO     095  69 OP
002  55  ÷      009  58 FIX     016  69 OP      036  05  05     056  06  6      076  00  00     096  00  00
003  43 RCL     010  00  00     017  31  31     037  92 RTN     057  42 STO     077  93  93     097  69 OP
004  02  02     011  52 EE      018  43 RCL     038  76 LBL     058  12  12     078  76 LBL     098  02  02
005  75  -      012  22 INV     019  03  03     039  11  A      059  61 GTO     079  14  D      099  01  1
006  93  .      013  52 EE      020  69 OP      040  99 PRT     060  00  00     080  99 PRT     100  44 SUM
                                021  01  01     041  16 A'      061  93  93     081  55  ÷      101  12  12
                                022  69 OP      042  04  4      062  76 LBL     082  93  .      102  73 RC*
                                023  02  02     043  42 STO     063  12  B      083  00  0      103  12  12
                                024  69 OP      044  12  12     064  99 PRT     084  02  2      104  69 OP
                                025  03  03     045  61 GTO     065  55  ÷      085  05  5      105  03  03
                                026  69 OP      046  00  00     066  02  2      086  04  4      106  69 OP
                                027  04  04     047  93  93     067  93  .      087  95  =      107  05  05
                                028  69 OP      048  76 LBL     068  05  5      088  16 A'      108  98 ADV
                                029  05  05     049  13  C      069  04  4      089  01  1      109  98 ADV
                                030  98 ADV     050  99 PRT     070  95  =      090  00  0      110  22 INV
                                031  97 DSZ     051  55  ×      071  16 A'      091  42 STO     111  58 FIX
                                032  01  01     052  01  1      072  08  8      092  12  12     112  98 ADV
                                033  00  00     053  02  2      073  42 STO     093  73 RC*     113  25 CLR
                                                                                                114  91 R/S
```

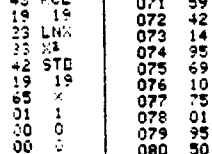
```
            1.      LAB
           20.      LIN
3.141592654         RND
```



```
            2.      LAB
            0.      LIN
4.721051104         RND
```



```
            3.      LAB
            0.      LIN
9.318024891         RND
```



```
000  76  LBL
001  19  D'
002  53  (
003  43  RCL
004  19  19
005  23  LNX
006  33  X²
007  42  STO
008  19  19
009  65  ×
010  01  1
011  00  0
012  00  0
013  00  0
014  54  )
015  22  INV
016  69  OP
017  92  RTN
018  76  LBL
019  14  D
020  55  ÷
021  00  0
022  32  X:T
023  22  INV
024  28  LOG
025  95  =
026  22  INV
027  59  INT
028  65  ×
029  01  1
030  00  0
031  95  =
032  59  INT
033  92  RTN
034  76  LBL
035  10  E'
036  01  1
037  74  SM*
038  00  00
039  43  RCL
040  00  00
041  65  ×
042  43  RCL
043  15  15
044  22  INV
045  28  LOG
046  52  EE
047  22  INV
048  52  EE
049  95  =
050  44  SUM
051  14  14
052  92  RTN
053  76  LBL
054  18  C'
055  04  4
056  42  STO
057  00  00
058  71  SBR
059  01  01
060  21  21
061  87  IFF
062  01  01
063  01  01
064  04  04
065  43  RCL
066  14  14
067  55  ÷
068  01  1
069  00  0
070  75  -
071  59  INT
072  42  STO
073  14  14
074  95  =
075  69  OP
076  10  10
077  75  -
078  01  1
079  95  =
080  50  I×I
081  65  ×
082  71  SBR
083  01  01
084  30  30
085  97  DSZ
086  15  15
087  00  00
088  97  97
089  71  SBR
090  01  01
091  21  21
092  22  INV
093  97  DSZ
094  00  00
095  01  01
096  15  15
097  87  IFF
098  01  01
099  01  01
100  04  04
101  71  SBR
102  01  01
103  30  30
104  97  DSZ
105  15  15
106  00  00
107  65  65
108  71  SBR
109  01  01
110  21  21
111  97  DSZ
112  00  00
113  00  00
114  65  65
115  69  OP
116  05  05
117  22  INV
118  86  STF
119  01  01
120  92  RTN
121  00  0
122  48  EXC
123  16  16
124  84  OP*
125  00  00
126  05  5
127  42  STO
128  15  15
129  92  RTN
130  03  3
131  02  2
132  65  ×
133  53  (
134  05  5
135  75  -
136  43  RCL
137  15  15
138  54  )
139  22  INV
140  28  LOG
141  33  X²
142  95  =
143  44  SUM
144  16  16
145  92  RTN
146  01  1
147  00  0
148  42  STO
149  00  00
150  69  OP
151  30  30
152  73  RC*
153  00  00
154  22  INV
155  67  EQ
156  01  01
157  50  50
158  92  RTN
159  65  ×
160  09  9
161  42  STO
162  16  16
163  01  1
164  65  ×
165  93  .
166  01  1
167  95  +
168  22  INV
169  59  INT
170  22  INV
171  67  EQ
172  01  01
173  76  76
174  43  RCL
175  00  00
176  52  EE
177  09  9
178  95  =
179  59  INT
180  22  INV
181  52  EE
182  97  DSZ
183  16  16
184  01  01
185  64  64
186  92  RTN
187  01  1
188  01  1
189  42  STO
190  00  00
191  25  CLR
192  73  ST*
193  00  00
194  97  DSZ
195  00  00
196  01  01
197  92  92
198  92  RTN
199  19  D'
200  65  ×
201  08  8
202  95  =
203  59  INT
204  22  INV
205  28  LOG
206  95  =
207  52  EE
208  22  INV
209  52  EE
210  42  STO
211  10  10
212  65  ×
213  01  1
214  00  0
215  85  +
216  01  1
217  95  =
218  42  STO
219  14  14
220  18  C'
221  92  RTN
222  76  LBL
223  16  A'
224  01  1
225  42  STO
226  12  12
227  43  RCL
228  12  12
229  42  STO
230  13  13
231  32  X:T
232  43  RCL
233  11  11
234  14  D
235  67  EQ
236  03  03
237  74  74
238  42  STO
239  00  00
240  01  1
241  44  SUM
242  12  12
243  43  RCL
244  12  12
245  32  X:T
246  43  RCL
247  11  11
248  14  D
249  22  INV
250  67  EQ
251  02  02
252  40  40
253  19  D'
254  45  Y×
255  53  (
256  73  RC*
257  00  00
258  55  ÷
259  06  6
260  00  0
261  54  )
262  65  ×
263  01  1
264  93  .
265  09  9
266  75  -
267  93  .
268  01  1
269  42  STO
270  15  15
271  65  ×
272  22  INV
273  74  SM*
274  00  00
275  73  RC*
276  00  00
277  22  INV
278  59  INT
279  69  OP
280  10  10
281  65  ×
282  43  RCL
283  12  12
284  85  +
285  01  1
286  93  .
287  01  1
288  95  =
289  42  STO
290  16  16
291  22  INV
292  97  DSZ
293  16  16
294  03  03
295  39  39
296  19  D'
297  65  ×
298  02  2
299  95  =
300  59  INT
301  75  -
302  19  D'
303  45  Y×
304  05  5
305  95  =
306  50  I×I
307  65  ×
308  53  (
309  43  RCL
310  12  12
311  85  +
312  93  .
313  09  9
314  75  -
315  43  RCL
316  15  15
317  32  X:T
318  43  RCL
319  13  13
320  42  STO
321  15  15
322  95  =
323  59  INT
324  44  SUM
325  15  15
326  43  RCL
327  15  15
328  67  EQ
329  02  02
330  91  91
331  10  E'
332  93  .
333  01  1
334  74  SM*
335  00  00
336  61  GTO
337  02  02
338  91  91
339  01  1
340  44  SUM
341  12  12
342  43  RCL
343  12  12
344  32  X:T
345  09  9
346  77  GE
347  02  02
348  27  27
349  44  SUM
350  14  14
351  43  RCL
352  14  14
353  55  ÷
354  01  1
355  00  0
356  95  =
357  59  INT
358  42  STO
359  10  10
360  09  9
361  42  STO
362  00  00
363  73  RC*
364  00  00
365  59  INT
366  72  ST*
367  00  00
368  97  DSZ
369  00  00
370  03  03
371  63  63
372  18  C'
373  92  RTN
374  43  RCL
375  12  12
376  32  X:T
377  43  RCL
378  10  10
379  14  D
380  42  STO
381  00  00
382  22  INV
383  67  EQ
384  02  02
385  53  53
386  71  SBR
387  01  01
388  46  46
389  43  RCL
390  13  13
391  42  STO
392  15  15
393  10  E'
394  61  GTO
395  03  03
396  39  39
397  76  LBL
398  15  E
399  03  3
400  69  OP
401  17  17
402  92  RTN
403  76  LBL
404  17  B'
405  01  1
406  42  STO
407  12  12
408  43  RCL
409  12  12
410  32  X:T
411  43  RCL
412  10  10
413  14  D
414  42  STO
415  00  00
416  22  INV
417  67  EQ
418  04  04
419  23  23
420  71  SBR
421  01  01
422  46  46
423  09  9
424  87  IFF
425  02  02
426  04  04
427  29  29
428  65  ×
429  93  .
430  07  7
431  65  ×
432  19  D'
433  95  =
434  42  STO
435  13  13
436  22  INV
437  97  DSZ
438  13  13
439  05  05
440  51  51
441  43  RCL
442  12  12
443  85  +
444  01  1
445  95  =
446  32  X:T
447  43  RCL
448  10  10
449  14  D
450  42  STO
451  15  15
452  67  EQ
453  05  05
454  33  33
455  32  X:T
456  43  RCL
457  00  00
458  67  EQ
459  05  05
460  51  51
461  19  D'
462  45  Y×
463  53  (
464  53  (
465  73  RC*
466  15  15
467  85  +
468  73  RC*
469  00  00
470  54  )
471  55  ÷
472  02  2
473  00  0
474  95  =
475  32  X:T
476  93  .
477  05  5
478  22  INV
479  87  IFF
480  02  02
481  04  04
482  84  84
483  01  1
484  22  INV
485  77  GE
486  05  05
487  51  51
488  43  RCL
489  00  00
490  32  X:T
491  43  RCL
492  15  15
493  77  GE
494  04  04
495  98  98
496  48  EXC
497  00  00
498  65  ×
499  93  .
500  01  1
501  49  PRD
502  00  00
503  95  =
504  32  X:T
505  43  RCL
506  10  10
507  71  SBR
508  01  01
509  99  99
510  42  STO
511  10  10
512  43  RCL
513  14  14
514  71  SBR
515  01  01
516  59  59
517  42  STO
518  14  14
519  32  X:T
520  42  STO
521  15  15
522  01  1
523  00  0
524  49  PRD
525  00  00
526  49  PRD
527  15  15
528  00  0
529  63  EX*
530  15  15
531  74  SM*
532  00  00
533  43  RCL
534  12  12
535  42  STO
536  15  15
537  10  E'
538  01  1
539  44  SUM
540  12  12
541  43  RCL
542  12  12
543  32  X:T
544  08  8
545  77  GE
546  04  04
547  36  36
548  61  GTO
549  05  05
550  79  79
551  93  .
552  01  1
553  65  ×
554  73  RC*
555  00  00
556  59  INT
557  69  OP
558  10  10
559  95  =
560  74  SM*
561  00  00
562  01  1
563  44  SUM
564  12  12
565  43  RCL
566  12  12
567  32  X:T
568  08  8
569  77  GE
570  04  04
571  08  08
572  09  9
573  32  X:T
574  43  RCL
575  10  10
576  14  D
577  42  STO
578  00  00
579  93  .
580  01  1
581  74  SM*
582  00  00
583  65  ×
584  44  SUM
585  14  14
586  43  RCL
587  14  14
588  59  INT
589  42  STO
590  14  14
591  95  =
592  59  INT
593  42  STO
594  11  11
595  86  STF
596  01  01
597  18  C'
598  92  RTN
599  76  LBL
600  13  C
601  25  CLR
602  02  2
603  00  0
604  42  STO
605  17  17
606  22  INV
607  86  STF
608  02  02
609  71  SBR
610  01  01
611  87  87
612  01  1
613  44  SUM
614  17  17
615  73  RC*
616  17  17
617  42  STO
618  18  18
619  98  ADV
620  02  2
621  07  7
622  01  1
623  03  3
624  01  1
625  04  4
626  69  OP
627  04  04
628  43  RCL
629  17  17
630  75  -
631  02  2
632  00  0
633  95  =
634  69  OP
635  06  06
636  02  2
637  07  7
638  02  2
639  04  4
640  03  3
641  01  1
642  69  OP
643  04  04
644  43  RCL
645  18  18
646  69  OP
647  06  06
648  02  2
649  22  INV
650  44  SUM
651  18  18
652  22  INV
653  49  PRD
654  18  18
655  03  3
656  05  5
657  03  3
658  01  1
659  01  1
660  06  6
661  69  OP
662  04  04
663  43  RCL
664  19  19
665  69  OP
666  06  06
667  98  ADV
668  71  SBR
669  01  01
670  99  99
671  08  8
672  00  0
673  42  STO
674  01  01
675  17  B'
676  16  A'
677  97  DSZ
678  18  18
679  06  06
680  71  71
681  86  STF
682  02  02
683  17  B'
684  71  SBR
685  01  01
686  99  99
687  97  DSZ
688  20  20
689  06  06
690  06  06
691  92  RTN
692  76  LBL
693  12  B
694  42  STO
695  19  19
696  92  RTN
697  76  LBL
698  11  A
699  42  STO
700  20  20
701  02  2
702  01  1
703  42  STO
704  17  17
705  43  RCL
706  17  17
707  75  -
708  02  2
709  01  1
710  95  =
711  91  R/S
712  42  STO
713  17  17
714  01  1
715  44  SUM
716  17  17
717  61  GTO
718  07  07
719  05  05
```

NEWCOMER'S CORNER.- (To branch or not to branch....) Somebody called me and asked me if I could help him with a program. Why not? His problems were: He had the calculator only three days and he would like to enter a time in hours and minutes at the start of an operation and another one at the stop of the operation. The program he had in mind should then have to compute the number of minutes elapsed. It sounded quite simple: Store both times, subtract the earlier from the latter time, separate the hours, multiply by 60, add to the minutes and display or print.

But, he asked, what if my first time falls in the morning, say at 10.20 and my second time in the afternoon, say at 2.35 ? You will get a negative time. Well, I had the solution for that one too: add twelve hours to afternoon times. No, he said, I want to enter both times in 12-hour format and have the calculator decide if one time falls in the morning, the other in the afternoon. (picky, picky!)

So, we would have to use a comparison, X $\geq$ T or INV X $\geq$ T. The printer prints that as GE and INV GE respectively, so I will stick to that nomenclature, mostly because I won't have to change typing elements this way.

The principle behind the comparison is very simple: you stick one number into the T-register (for Test) by means of the X:T key, (you know, the third key from the top in the second column from the left) then you call or put the second number into the X-register, of which the display shows you the first 10 digits, you say either GE or INV GE and the program will branch to the common label, user-defined label or direct address that follows immediately after the comparison, if the comparison is true. If not true, the program will continue with the over-next instruction.

Your first objection will be: (I know this from teaching TI-59 programming) Where is the X-register ? I don't see a key with that name. The X-register is simply the display. If you key a number into the calculator, it goes into the X-reg. But the display shows only the first 10 digits of that number. The X-reg, just like any other register in the TI-58/59, can hold up to 13 digits. Who cares, you will say. Well, comparisons are done ON ALL 13 DIGITS not just on the 10 we see. So, we should care about those three hidden ones. Press, for example, 2nd π and see in the display 3.141592654. The last digit, the 4, is a rounding off. But, of what? Multiply by 1000, by pressing X 1000 = and see 3141.592654. Now press INV INT to keep only the fractiona part. Now we see also the hidden digits:.59265359 . The last digit should be another zero, but because it is at the right side of the decimal point, it is not shown. Thus, the hidden digits were 590 and the 4 was a rounding off of 3590.

To show you the dangers lurking in the believe that comparisons are only done on what you see in the display, press 2 Y^x 3 = and see 8. Is that really 8? press INV INT and you should see 0, if it is truely 8. What do you see? 1E-12 ! Or our number 8 is in reality 8.000000000001. If you would have compared 8 to that number by means of X=T, it would not have branched, saying in effect "they are not equal", which is true.

Now, let's dispell another myth. When I say "branch" I mean "branch" as in tree That is, a branch is NOT A SUBROUTINE. A branch on a tree parts from the main body of the tree AND NEVER COMES BACK TO IT, unless you tie it with a rope. Thus, if you branch by means of a comparison, it will not do you any good to put a RTN at the end of that branch. This is not a subroutine and it will not return to the main program automatically. You will have to either put R/S or a GTO somewhere. The GTO can be to a common label (such as STO, RCL, SUM, what have you) to a user-defined label (A, B', E') or to a direct address. (123, 034, 674)

As a little side remark: I see programs all the time that have instructions such as "SBR A' ". If you simply call A' it means in reality "SBR A'. But the user-defined keys save you one step, by letting you call simply A'. So, save yourself the trouble of putting the SBR instruction in front of it. It is not needed. But, remember now, that it is a subroutine call. I sometimes get programs to debug that have loops in them with sequences such as ....RCL 03 GE A B ....After doing this loop seven times the program goes haywire, is the usual complaint. Small wonder ! What you are saying there is: If the comparison is true, branch to label A if not do subroutine B. The calculator can do only 7 of those before its return pointer register is full. What you meant to say was: ...RCL 03 GE A GTO B.... (with the emphasis on GTO) Now, there is no problem. No full return pointer register will spoil things and the loop can be done in eternity, or until somebody cuts your power supply, whichever comes FIRST.

Newcomer's corner.(continued)

If we compare by means of X=T things are rather simple: either the two numbers are exactly equal, all 13 digits, or they are not. So, we branch accordingly.

But now we are going to compare with GE and INV GE. Let's write it out one more time (change typing elements, please) X ≥ T and INV X ≥ T. In words, this means: if the number in the X-reg is equal to or larger than the number in the T-reg.... and if the number in the X-reg is not equal to or smaller than the number in the T-reg.

That sounds simple enough. But let me dispell another myth: I still have to meet a seasoned programmer who is 100 % sure of him- or herself when it comes to predict if a certain comparison is going to branch or not. Especially if it involves an INV, most "seasoned" programmers start doubting and experiment a bit before freezing the program in its final form. Including me. I just shake my head and resort to my trusted experimenter's program. It is in reality a very simple routine, that tells you when a certain comparison will branch or not. It is fun to play with it and it helps you understand how comparisons work. Do a comparison in your mind, predict the outcome and write it down. Then use this routine and see what it has to say. Once you start hitting 100 % correct predictions, consider yourself in the "genius" category, get yourself a button that says "my I.Q. is easily above 140" and apply for membership in one of those mutual admiration societies who cater to the above-140-crowd.

Key in the program, 291 steps, and record on two sides of a mag card.
Enter the first number and press A. Enter the second number and press B.
If you want the first number in the T-reg and the second one in the X-reg, press C.
If you want the first number in the X-reg and the second one in the T-reg, press D.
If you want the comparison GE, press 2nd A'.
If you want the comparison INV GE, press 2nd B'.
Once the two numbers are in, you don't have to put in new ones each time. You can move them where you want by keys C and D and do any of the two comparisons by means of A' and B'. You will need a TI-59 mounted on the printer for this routine.

Who writes me a suitable routine for use on the calculator only, preferrably for use with both the 58 and the 59 ? I will publish the best ones.

By the way, if you wondered what happened to the program for my caller, the one that computes the number of minutes elapsed between two times, I had to write two separate routines. My problem was this: how do you interprete entering two exactly equal times? Should the result be zero,the calculator saying in effect "yes, you dummy, those are equal times." Or should the result be 720 minutes?(60 minutes/hour X 12 hrs) meaning that the first time is in the morning and the second one in the afternoon, for example. Thus, I wrote two separate routines and let the user elect which one suits him best.

Both work the same way: Enter the first time and press A.
Enter the second time and press B.
Both entries sould be in the format HH.mm and work for a 12-hour clock only. Thus, 10.40 AM is entered as 10.40 and 2.35 PM is entered as 2.35 and NOT as 14.35.
The programs work for calculator-only and work on both the 58 and the 59.
The result is the number of the minutes elapsed, displayed only.

000 76 LBL 001 44 SUM 002 01 1 003 02 2 004 44 SUM 005 02 02 006 61 GTO 007 42 STD 008 76 LBL	009 11 A 010 42 STD 011 01 01 012 35 CLR 013 92 RTN 014 76 LBL 015 12 B 016 42 STD 017 02 02	018 32 X:T 019 43 RCL 020 01 01 021 [illegible] GE 022 44 SUM 023 76 LBL 024 42 STD 025 43 RCL 026 02 02	027 59 INT 028 75 - 029 43 RCL 030 01 01 031 59 INT 032 95 = 033 65 × 034 06 6	035 00 0 036 95 = 037 42 STD 038 00 00 039 43 RCL 040 02 02 041 22 INV 042 59 INT	043 75 - 044 43 RCL 045 01 01 046 22 INV 047 59 INT 048 95 = 049 65 × 050 01 1	051 00 0 052 00 0 053 95 = 054 44 SUM 055 00 00 056 43 RCL 057 00 00 058 92 RTN

000 76 LBL 001 44 SUM 002 01 1 003 02 2 004 44 SUM 005 02 02 006 61 GTO 007 42 STD 008 76 LBL 009 11 A	010 42 STD 011 01 01 012 32 X:T 013 35 CLR 014 92 RTN 015 76 LBL 016 12 B 017 42 STD 018 02 02	019 22 INV 020 [illegible] GE 021 44 SUM 022 76 LBL 023 42 STD 024 43 RCL 025 02 02 026 59 INT	027 75 - 028 43 RCL 029 01 01 030 59 INT 031 95 = 032 65 × 033 06 6 034 00 0	035 95 = 036 42 STD 037 00 00 038 43 RCL 039 02 02 040 22 INV 041 59 INT 042 75 -	043 43 RCL 044 01 01 045 22 INV 046 59 INT 047 95 = 048 65 × 049 01 1 050 00 0	051 00 0 052 95 = 053 44 SUM 054 00 00 055 43 RCL 056 00 00 057 92 RTN

To branch or not to branch, program listing.

```
        78.      N1
        56.      N2
  N1 IN T-REG.
  N2 IN X-REG.
    X =) T ?
DOES NOT BRANCH

        46.      N1
        79.      N2
  N2 IN T-REG.
  N1 IN X-REG.
    X =) T ?
DOES NOT BRANCH

        12.      N1
        23.      N2
  N1 IN T-REG.
  N2 IN X-REG.
    X =) T ?
 BRANCHES

        23.      N1
        56.      N2
  N1 IN T-REG.
  N2 IN X-REG.
INV X =) T ?
DOES NOT BRANCH

        89.      N1
        45.      N2
  N2 IN T-REG.
  N1 IN X-REG.
INV X =) T ?
DOES NOT BRANCH

        44.      N1
        44.      N2
  N2 IN T-REG.
  N1 IN X-REG.
    X =) T ?
 BRANCHES

        89.      N1
        89.      N2
  N1 IN T-REG.
  N2 IN X-REG.
INV X =) T ?
DOES NOT BRANCH
```

```
000  76 LBL
001  32 X:T
002  69 OP
003  01  01
004  02  2
005  04  4
006  03  3
007  09  9
008  00  0
009  00  0
010  03  3
011  07  7
012  69 OP
013  02  02
014  92 RTN
015  76 LBL
016  69 OP
017  02  2
018  00  0
019  03  3
020  05  5
021  01  1
022  07  7
023  02  2
024  02  2
025  04  4
026  00  0
027  71 SBR
028  34 FX
029  92 RTN
030  76 LBL
031  34 FX
032  69 OP
033  03  03
034  69 OP
035  05  05
036  69 OP
037  00  00
038  92 RTN
039  76 LBL
040  33 X²
041  69 OP
042  01  01
043  02  2
044  04  4
045  03  3
046  09  9
047  00  0
048  00  0
049  04  4
050  04  4
051  69 OP
052  02  02
053  92 RTN
054  76 LBL
055  68 NOP
056  02  2
057  00  0
058  03  3
```

```
059  05  5
060  01  1
061  07  7
062  02  2
063  02  2
064  04  4
065  00  0
066  71 SBR
067  34 FX
068  92 RTN
069  76 LBL
070  11  A
071  42 STO
072  01  01
073  03  3
074  09  9
075  00  0
076  02  2
077  69 OP
078  04  04
079  43 RCL
080  01  01
081  69 OP
082  06  06
083  69 OP
084  00  00
085  92 RTN
086  76 LBL
087  12  B
088  42 STO
089  02  02
090  03  3
091  09  9
092  00  0
093  03  3
094  69 OP
095  04  04
096  43 RCL
097  02  02
098  69 OP
099  06  06
100  69 OP
101  00  00
102  92 RTN
103  76 LBL
104  13  C
105  03  3
106  09  9
107  00  0
108  02  2
109  71 SBR
110  32 X:T
111  71 SBR
112  69 OP
113  43 RCL
114  01  01
115  32 X:T
116  03  3
```

```
117  09  9
118  00  0
119  03  3
120  71 SBR
121  33 X²
122  71 SBR
123  68 NOP
124  43 RCL
125  02  02
126  42 STO
127  00  00
128  25 CLR
129  92 RTN
130  76 LBL
131  14  D
132  03  3
133  09  9
134  00  0
135  03  3
136  71 SBR
137  32 X:T
138  71 SBR
139  69 OP
140  43 RCL
141  02  02
142  32 X:T
143  03  3
144  09  9
145  00  0
146  02  2
147  71 SBR
148  33 X²
149  71 SBR
150  68 NOP
151  43 RCL
152  01  01
153  42 STO
154  00  00
155  25 CLR
156  92 RTN
157  76 LBL
158  16 A'
159  71 SBR
160  22 INV
161  43 RCL
162  00  00
163  77 GE
164  44 SUM
165  61 GTO
166  43 RCL
167  76 LBL
168  17 B'
169  02  2
170  04  4
171  03  3
172  09  9
173  04  4
174  02  2
```

```
175  00  0
176  00  0
177  69 OP
178  01  01
179  71 SBR
180  22 INV
181  43 RCL
182  00  00
183  22 INV
184  77  GE
185  44 SUM
186  76 LBL
187  43 RCL
188  01  1
189  06  6
190  03  3
191  02  2
192  01  1
193  07  7
194  03  3
195  06  6
196  00  0
197  00  0
198  69 OP
199  01  01
200  03  3
201  09  9
202  03  3
203  02  2
204  03  3
205  07  7
206  00  0
207  00  0
208  01  1
209  04  4
210  69 OP
211  02  02
212  03  3
213  05  5
214  01  1
215  03  3
216  03  3
217  09  9
218  01  1
219  05  5
220  02  2
221  03  3
222  69 OP
223  03  03
224  71 SBR
225  52  EE
226  98 ADV
227  92 RTN
228  76 LBL
229  44 SUM
230  01  1
231  04  4
232  03  3
```

```
233  05  5
234  01  1
235  03  3
236  69 OP
237  01  01
238  03  3
239  09  9
240  01  1
241  05  5
242  02  2
243  03  3
244  01  1
245  07  7
246  03  3
247  06  6
248  69 OP
249  02  02
250  71 SBR
251  52  EE
252  98 ADV
253  92 RTN
254  76 LBL
255  22 INV
256  04  4
257  04  4
258  00  0
259  00  0
260  06  6
261  04  4
262  05  5
263  06  6
264  00  0
265  00  0
266  69 OP
267  02  02
268  03  3
269  07  7
270  00  0
271  00  0
272  07  7
273  09  9
274  00  0
275  00  0
276  00  0
277  00  0
278  69 OP
279  03  03
280  71 SBR
281  52  EE
282  92 RTN
283  76 LBL
284  52  EE
285  69 OP
286  05  05
287  69 OP
288  00  00
289  25 CLR
290  92 RTN
```

Re-Newcomer's Corner v6n3p14. - Several members have written me to tell me that they had trouble with the long chain of numbers and the RTN at the very end. Some of you gave the solution: put nine zeros at the very end, just before the RTN. Otherwise you will get zeros rather than the required digits at the end, that is you simply can't access the last nine digits.

Michael Sperber says that another remedy would be to change LBL A to :

LBL A SBR IND 00 EE = EE 0 0 INV EE INT OP 20 R/S

Very clever, but, as Michael notes himself, hardly suitable for the newcomer's corner.

Re-SR-56 Listing on a TI-59.- V6n3p13.Michael Sperber asks if two register contents shouldn't be changed:

R51 is now 1515.0602, and should be changed to 1517.0602
R57 is now 36411435.0603 and should be changed to 36411435.0608

I don't have a 56 anymore. Could somebody please check and let me know?

OPTICS.- Gregory L. Stark recently had a ray-tracing program system in EOSD as announced in v6n2p3. The system consists of three well documented (equations, metho examples, register contents) programs. Each program comes with the listing and recorded magnetic cards. The first program is called RAY TRACE THROUGH TORIC; the next one is SKEW RAY TRACE THROUGH ASPHERIC--SURFACE BY SURFACE, and the third on is called TILTED SURFACE.

Gregory will send a copy of each program, 35 pages in total, and four recorded magnetic cards, if you defray his cost of $ 6.00. This includes the mailing inside the US, Canada and Mexico. If you live outside the US, please send Greg a few dollars extr to cover air mail A.O. Mail Printed Matter postage.

I tried those programs and they work. They are user-friendly and, although some of them use an iteration scheme, they come up with an answer in reasonably short time. I wish all the programs I ever obtained from the PPX catalog would be of this high quality.

Please, do not write to me. Write Gregory L. Stark instead, at 25 Clearview Drive, Pittsford NY, 14534.

By the way, Greg submitted another one of his pet programs to the PPX: MU-08 Modified. It concerns data packing, which is one of Greg's pet projects. It allows the storing and recalling of both positive and negative integers with three pseudo-registers per data register.

ANGLE CONVERTER.- v5n4/5p25 gave a program to convert angles by Ralph Donnelly. It was faster (1.0 sec) than an earlier program by Frank Blachly in v5n3p4 (21 steps, 3.0 sec.) but it will not work with negative numbers.

Robert M. Elliot has written one with 20 steps, 1.5 sec execution time, that will work with negative numbers:

```
LBL +/-  - 360  )  GE  +/-  LBL IFF  +  360 )  INV GE  IFF RTN
```

MICKY MOUSE.- Encouraged by the success of Spock being beamed in, may I challenge you to newer and loftier heights? Would you like to try your hand at this nice drawing of Micky Mouse. Do you think your TI-59 cum PC100A or C could handle it? This one was done on a computer, of course, with a special video line printer.