* T I P P C N O T E S *

V5N8, 1980.

NEWSLETTER OF THE TI PERSONAL PROGRAMMABLE CALCULATOR CLUB.

9213 Lanham Severn Road, Lanham, Maryland, 20801.

In this issue you will find a follow-up on the calendar printing contest between the RPN and AOS programmers. Thanks to the fast mode we are still ahead. But now comes the difficult task of shaving off a few seconds here and there. To show what can be done, Richard Snow has investigated various aspects of the calendar program and he reports that the TI-59 executes sometimes while the PC100 is still printing. The same phenomenon was reported by Roger Hill for the HP41C and its printer. Thus, says Richard, if an ADV follows immediately upon an OP 05, the program is delayed momentarily until the OP 05 instruction is finished, before executing the ADV instruction. The speed of some versions of the calendar program may be increased by even 2 seconds per year by merely moving the ADV instruction away from the OP 05 command at several locations.

I especially like Roger Hill's algorithm for determining the weekday on which a month begins. I haven't figured out, though, how it works. I would love to see an explanation of it. It could be substituted directly in our speedy calendar program and it would save twenty steps. Unfortunately, that sequence is used only once, it is not part of a loop, so that it saves only a fraction of a second of program execution.

It looks like the HP41C is better adapted at handling alpha-numeric data than the TI-59 and, if it wasn't for the discovery of the fast mode, they would be well ahead of us in the race.

In the latest issue of the HP PPC Journal, the editor, Richard Nelson, suggest that I print a sample print-out of our calendar, so that anyone who wants to try to improve on Roger's and Palmer's programs could have some guidelines. So, here it goes.

```
       1980.     JAN
 S  M  T  W  T  F  S
       1  2  3  4  5
 6  7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31

       1980.     FEB
 S  M  T  W  T  F  S
                1  2
 3  4  5  6  7  8  9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29
```

By the way, I would suggest, rather than take the time of just any year, to make the program so that it prints several years in succession. One could then take the average running time over the years 1976 through 1980 as the actual time. This method is fair and more accurate.

As far as accepting the other challenges proposed by the RPN programmers, I have received consent from several OAS programmers to accept the challenge on alphanumeric sorting: write a program that sorts in the shortest possible time 99 five-letter words stored in registers in reverse order, (the worst case) and in correct order.(the easiest case) No print-out is needed, just let's see how fast it can do program execution with your favored sorting algorithm. Of course, you print it out after the sorting is completed to verify its accuracy, but let's not include the printing time.

Somebody gave me a problem that has me completely stomped: "Produce a list of integers between two predetermined limits.'Randomize' them in such a way that there is no repetition. In other words, each 'randomly' generated integer is unique. A minimum of 100 different integers are required but any larger series is accepted with gratitude. I will publish the best ones, of course.

Now, just sit back and enjoy this issue. I hope you do.

Maurice E.T. Swinnen

TABLE OF CONTENTS.

CALENDAR CONTEST, Palmer Hansen, Richard Snow..............2
CALCULATOR CALCULUS. George McCarthy..........................3
MODULE SELECTOR,..3
RE-REACTION TIME TEST, Jeff Rosedale..........................4
BRAIN TEASERS, Bod Baldassano, Palmer Hansen.................4
FAST MODE, Palmer Hansen..4
SOLUTION TO 3 EQ. IN 3 UNKNOWNS, COMPLEX, Bill Beebe.....5 - 6
RPN MODULE, Don Jones..6
RE-SBR CONVERSION, Jared Weinberger............................6
PASSIVE CROSS-OVER NETWORKS, Maurice Swinnen.............7 - 8
SUPERPLOT II, Robert and Richard Snow...................9 -10
TAX COMPUTATION MODULE,..9
LABELS TO DIR.ADDRESSES, P.Galidas, Rich Snow, J. Rosedale..10
MORE QUIRKS, Patrick Acosta.......................................11
BIDDING SYSTEMS, Richard Jameson................................11
BOOKS, BOOKS, BOOKS, Steven Grisham............................12
ERRATUM,..12
OPTICS,..12
FINANCIAL SYSTEMS REPORT, Vernon K. Jacobs....................12
COVERAGE PREDICTIONS,Ed Westenhaver............................12
FIRST CLASS MAIL, Richard Snow...................................12
TI-57 LISTING ON A TI-59, Bill Skillman.........................13
RECURSIVE SIMPSON RULE, Norman Herzberg...................14-15
SPEEDY FACTOR FINDER, TI-57, Bill Skillman....................15
ML LIBRARY LABEL PROGRAM, Jerome Porter......................16
DIAGNOSTIC, Marion Brown..16
CROSS-COMPILER BASIC TO TI-59,..................................16

CALENDAR CONTEST.- The July-August 1980 issue of the PPC Calculator Journal (which arrived at our offices in the beginning of September) carries an article and program by Roger Hill on a fast calendar by means of the HP41C. It takes that program 2 min 17 sec to print one full year, clearly beating Panos Galidas' program of 2 min 32 sec. But, unknowingly to Roger, that same week I sent out our issue 7 with Palmer Hansen's version of Panos' program in Martin Neef's Fast Mode. That one is still the champion with 1 min 31 sec to print one full year.

Roger also computes the absolute minimum time the present HP printer will need in order to print the equivalent of one full year printing. That means, load the registers with garbage and print, nothing else. Palmer calls it "to keep the printer printing in a manner analogous to the old rule we used with drum type computers in the early 1960's where the rule was to keep the processor multiplying." Roger's estimate of an average of 62.46 lines per year for the HP41C-printer combination leads to the following estimate for the TI-59/PC100A:

$$62.46 \text{ lines X } \frac{15 \text{ seconds}}{13 \text{ lines}} = 72 \text{ seconds for normal mode and}$$

$$62.46 \text{ lines X } \frac{15 \text{ seconds}}{26 \text{ lines}} = 36 \text{ seconds in fast mode.}$$

By comparison, the theoretical minimum for the HP41C should be, using Roger Hill's figures:

$$62.46 \text{ lines X } \frac{73 \text{ seconds}}{106 \text{ lines}} = 43 \text{ seconds for the PH-41C/printer.}$$

Palmer arrived at these figures by running specific sequences both in normal and in fast mode and noting the time differences.

The foregoing would seem to imply that fast mode will be required if we are to stay ahead in the calendar print competition, but that we don't have much of an edge.

Palmer also sent me a recent PPX submission of a fast calendar version in which he demonstrated the use of code 46 as a dummy operator. He submits for discussion in the TI PPC NOTES the following:

"v5n6p3 reported that the IND command (code 40) could be used in place of the CE command (code 24) as a dummy operator to avoid clearing error indications. The INS command (code 46) can also be used. The use of it would seem to have the advantage that it will not typically be used for any other program function, and that "insert" appropriately defines what is being accomplished. Of course, either the IND or the INS commands must be synthesized, say be the sequence RCL 46 BST BST DEL SST for the INS command. While working on this technique I found that a synthesisez IND command (code 40) after a GTO (code 61) converts the sequence to the equivalent of a GTO.(code 83) Similar equivalences seem to apply for other merged codes. I challenge the reader to find a use for that feature."*

Richard Nelson, the editor of the HP PPC Journal, adds a note to Roger Hill's article, saying that "if the TI-59 programmers want a real challenge, then...". I am sending copies of that challenge to many of the active members and will ask their advice about it, if we will accept the challenge and under what conditions if any. I will keep you informed.

As was to be expected, I didn't even have to wait till issue 7 was mailed. Even before that I received "oh's and ah's" from the Snow brothers, obviously very impressed with Palmer's latest scoop.

Richard Snow even sent me his version of a fast calendar. Although he didn't shave off any additional seconds of the running time, it has some interesting friendly-to-the-user features, such as " next month printing" by simply pressing R/S after the first year has been printed, or the printing of any number of months from the entry of a starting month and the year, followed by an entry of the number of months required. Richard even takes care of card misreadings, which can sometimes take the calculator out of the fast mode. LBL E' will always be there to re-initialize and thus put the calculator in fast mode again.

Richard further notes that any(fast)program can be stopped only if there is a soft display. That is, any display that can be cleared with a CE. As soon as the function hardens the display, the program starts up again. Old timers, 52-User club members, will undoubtedly remember the discussion we had about what constitutes a hard or a soft display, both in the 52 and the 58/59. Thus, entering the mag cards causes a hard display and the (fast) program starts running right away.

CALCULATOR CALCULUS.- by George McCarty, the EduCALC Publications, Box 974, Laguna Beach CA, 92652, 254 pages, soft-bound, 1980.

This book is, in my opinion, an ideal textbook, either for class use or for self-study, as it explains calculus in an easy to understand way. If you ever heard of Simpson's rule but haven't got it all sifted out how exactly it works, just consult this book and you'll never forget it. The book contains a wealth of algorithms and other computing methods on anything imaginable that can be solved by means of calculus.

Although the book was not intended for programmable calculator use per se, any program can easily be written from the clear algorithms. Each section ends with several problems for you to solve and some solutions are also given.

The book is subdivided into 14 sections: 1. Squares, square roots and the quadratic formula, 2. More functions and graphs, 3. Limits and continuity, 4.Differentiation, derivatives and differentials, 5. Maxima, minima and the mean value theorem, 6. Trigonometric functions, 7. Definite integrals, 8. Logarithms and exponentials, 9. Volumes, 10. Curves and polar coordinates, 11. Sequences and series, 12. Power series, 13.Taylor series, 14. Differential equations. Appendix: Some calculation techniques and machine tricks, reference data and formulas, bibliography.

The book may be obtained by members of the TI PPC CLUB at a 10% discount from the above address. Mention on the outside envelope "TI PPC NOTES OFFER".Send $ 13.45 plus $ 1 shipping. (this is a 10% discount price from the normal $ 14.95 + $ 1.00)

MODULE SELECTOR.- Finally somebody has been using his brain and has come up with an extremely useful accessory to the TI-58/59. This device permits selection, either from the keyboard or under program control, of programs from up to four modules. Thus, for example, an EE might have plugged into the device his favored EE module, the ML library, the M/U module and the Statistics chip. His user programs may now access any of the above four. A dream come true.

The modules are labeled 0 through 3. To select, say module 3, one would have to code 3 77 OP 04 OP 05, followed by the normal PGM xx. Four individual Light Emitting Diodes (LEDs) are provided, which light up when a particular module has been accessed. Besides those four LEDs there are four additional ones, which may be controlled either from the keyboard or, preferrably, under program control, to signal a particular situation, as a prompting indicator, such as to tell the user to enter a card or to start a particular program sequence, or what have you.

The device comes completely assembled, with a flat cable ending in a connector that looks exactly like a module. That one is plugged into the back of the calculator instead of the usual module. The device itself has four sockets to accomodate up to four modules of your choice. The module selector does not need battery power nor does it have to be plugged into a wall socket. It derives its power entirely from the calculator. Besides an automatic module selector, that is one that may be accessed under program control as well as from the keyboard, there is "keyboard only" version, called the manual one. Its price is about half that of the automatic one.

Prices are $ 99.95 for the manual one and $ 199.95 for the automatic one, but if you send in your order with the mention "TI PPC NOTES" the prices will be reduced by ten dollars for each model, to $ 89.95 and 189.95 respectively.

The module selector has been designed and is being built by the well known American Micro-Products Inc in Richardson, Texas. This firm also produced the CROM simulators for Texas Instruments itself, the CROM emulators, the dual-disk drive accessory for the TI-59, the line printer for the TI-59, etc.

The device is marketed by Systems 7, 1020 Park Ave, Suite 101, Cranston RI, 02910. The people behind both the manufacturing and the marketing are all former TI employees and still have a perfect working relationship with that company.

I used both the manual and the automatic versions and, perceived from "an old programmer point of view" it is "heaven on earth.

The System 7 people are very well known for production and marketing of costum modules, such as for tax calculation, oil drilling and insurance computation.

So, if you just want information or to order one, write to Systems 7, NOT TO ME.

Re-REACTION TIME TEST.- In v5n3p13 at the end of the description of the Reaction Time Test, I stated that it would be a good challenge among the members to produce a general routine that would give the leading number and the amount of zeros trailing it of a number which had the following form: N000....000 . That is, a digit N followed by a number of zeros. Jeff Rosedale has developed such a routine:

LBL LOG DIV LOG INT STO 00 INV LOG) INT RTN

enter the number and press or call SBR LOG. The number in the display will be the leading digit of the number entered and the number of zeros will be stored in R00.

BRAIN TEASERS.- After v5n7 was typed, but before it was mailed, I got a few more solutions to the brain teasers.(see v5n7p5) Bob Baldassano has these routines to get .1415924534:

COS INV LNX DMS INT RAD COS +/- INV COS RTN (8 steps, not counting INV)

COS INV LNX + CE LNX) INT RAD COS +/- INV COS (11 steps)

And to obtain 197 he offers these two:

COS INV LNX INV LNX INV LNX $\sqrt{x}$ $\sqrt{x}$ GRD INV TAN + CE) INT (12 steps)

INV LNX INV LNX INV LNX INV LNX $\sqrt{x}$ $\sqrt{x}$ GRD INV TAN + CE) INT (12 steps)

Palmer Hansen was equally intrigued and sent these two: First to get .1415926536 :

RAD Σ+ +/- INV COS INV INT (5 steps) and to get 197:

Σ+ Σ+ X INV LOG - Σ+ = (7 steps)

FAST MODE.- Palmer O. Hansen has been trying to find an explanation as to how the fast mode works, ever since Martin Neef discovered this very useful state. Credit goes to Palmer for having been the first one to put this thing to practical use by writing several programs in such a way that they can easily be used with this unique mode. One of his more spectacular ones is the Fast Calendar which we presented in issue v5n7. Here are some more of Palmer's thoughts on the subject:" I have been catching up on back issues of 52-Notes. I found that very soon after the 59 came out, someone had found that the ML-16 program computed factorials almost twice as fast when called from the memory module as when called from a downloaded equivalent. This didn't seem to correspond with my earlier testing. I had compared the run times of the diagnostic in ML-01 as called by the sequence PGM 01 SBR... and found them to be nearly identical. Careful recheck showed a run of 14 seconds from the module and 16 seconds from user memory, hardly the two-to-one ratio. .I had also previously made a test of relative speeds using the determinant calculation from ML-02 and found them to be approximately equal. You may ask how I did this,since it requires a partitioning of 959.00 to download ML-02. The answer is that, after the download you return to the turn-on partitioning, and if you limit the size of the determinant, you can perform the calculations in user memory, since the determinant calculations do not use the upper half of the memory, at least for the program instructions. I tried the test again and got approximately equal run times. Furthermore, I got approximately equal run times for computing factorials with ML-16. After some reflection I realized that I was not returning the calculator to user memory after downloading, say by an RST or a PGM 00. As soon as I did I got the approximate two-to-one ratios for factorial calculations with ML-16 and determinant calculations with ML-02. But I couldn't approach the ratio with the diagnostic. The problem there seems to be related to the heavy use of statistics and conversions in the diagnostic. All of this leads to a refinement of my statement on fast mode which appeared in QUIRKS on v5n6p4 of TI PPC NOTES, namely that fast mode puts the calculator in the same mode as when it is accessing RAM memory for execution of the statistics and conversionss functions, or when accessing a program in the memory module."

SOLUTION OF THREE EQUATIONS IN THREE UNKNOWNS, COMPLEX.- Bill Beebe, of Lilburn, Georgia, contributed this excellent program. It will solve three equations in three unknowns with complex coefficients. This is very handy if you do circuit analysis. (EEs note!) Element entry follows convention established by ML-02. Thus, the Master Library module is required. Given three equations in the following form:

$$1x + 4y + 7z = 10$$
$$2x + 5y + 8z = 11$$
$$3x + 6y + 9z = 12$$

The digits 1 through 12 represent the complex coefficients and indicate their order of entry.
Data are entered in registers 5 through 28, which remain undisturbed by program execution. Bill used Crammers method for solving three equations in three unknowns. The program fits on one side of a mag card, for the TI-59. It will also work well on the TI-58/58C if partitioned to 239.29. Both ML or EE modules may be used. (Yes, TI had the forsight of putting identical PGMs 04, 05 and 06 on both modules)
Instructions: Initialize, press E. This clears all memories.
Coefficients have to be entered in POLAR form, in the order indicated above.So, if, for example, you have to enter (10 + j5)x + (-7 + j4)y+ 3/180°z you key in: 10 x:t 5 INV P-R A 7 +/- x:t 4 INV P-R A 3 x:t 180 A (note that the last coefficient is already in polar form, thus no INV P-R is needed)
Once entry 12 has been entered, the program automatically computes the value of the determinant, delta, in complex form. It will place the real part in the display and you may obtain the imaginary part by pressing x:t. The only conceivable error condition is division by zero, but it is avoided by checking the results of delta.

If during data entry you make a mistake, enter the number of the entry to be corrected and press 2nd A'. The program will come back with the number of the register in which the magnitude is stored. Ignore this, but enter the correct value through key A. If during entry of the last element or even while the program is computing delta, you notice an error, simply press R/S and CLR. Then enter 12 and press 2nd A'. You will find that the program computes delta again.
Now , if you find that there is no danger of division by zero, obtain X, Y and Z. To obtain X, press B to obtain the real part, followed by x:t to obtain the imaginary part. Do the same to obtain Y (press C) and Z. (press D)

If you use the TI-59 only, this program is more than satisfactory. But, in combination with the PC100, you could produce a program that is more "anwendungsfreundlich" that is one which requires much less button pushing. I use my calculator most at the office, and it is always attached to the printer. (I even lost the battery pack somewhere) I am also very bad at copying long numbers correctly. So, I prefer to give that task to the printer, after which I cut the tape and paste it in my note book.
I re-worked Bill's program so that it will give a satisfactory printout. Do not try to achieve OP 04 OP 06 printing. The P-R conversion uses HIR8 and will mess up your OP 04 OP 06 printing, which in turn will distort your results. I can guarantee you, it will look messier than the five o'clock traffic on the freeway on a rainy day.
Entry of the printer program is much simpler. Do not touch the x:t key. Instead, enter, for example (10 + j5) x + 3/180° ....as 10 A 5 B 3 A 180 C
Thus, rectangular data is always entered through A B, while polar is entered A C. Once all 12 coefficients are entered the program will print delta, first the real part, followed by the imaginary part, then X, Y, Z, each time the real part first. The sample problem shows what to expect. If you don't want to waste all that paper, you could eliminate printing of the input data. Simply over-write steps 018-023-028 with NOPs. (Do NOT delete, because of the direct addresses)
Error recovery is identical as in the "TI-59-only" program.

THREE EQUATIONS IN THREE UNKNOWNS, COMPLEX COEFFICIENTS.

EXAMPLE: GIVEN : (10 + j5)x + (-7 + j4)y + 3/180° z = 325/30°
(-7 + j4)x + (19 -j7)y + (12 + j8)z = 494.2/27.08°
3/180°x + (12 + j8)y + (19 + j14)z = 585/90°

```
000  76 LBL     050  01  01     100  19  19     150  06  06     200  01  1      250  97 DSZ
001  18 C'      051  42 STD     101  85  +      151  95  =      201  61 GTD     251  02  02
002  85  +      052  03  03     102  43 RCL     152  32 X:T     202  02  02     252  02  02
003  01  1      053  99 PRT     103  09  09     153  43 RCL     203  06  06     253  40  40
004  08  8      054  32 X:T     104  19 D'      154  15  15     204  25 CLR     254  92 RTN
005  00  0      055  99 PRT     105  43 RCL     155  85  +      205  05  5      255  76 LBL
006  76 LBL     056  98 ADV     106  18  18     156  43 RCL     206  42 STD     256  15  E
007  19 D'      057  61 GTD     107  65  X      157  19  19     207  29  29     257  69 DP
008  95  =      058  01  01     108  43 RCL     158  85  +      208  71 SBR     258  00  00
009  37 P/R     059  82  82     109  08  08     159  43 RCL     209  02  02     259  47 CMS
010  44 SUM     060  76 LBL     110  65  X      160  05  05     210  31  31     260  01  1
011  02  02     061  17 B'      111  43 RCL     161  18 C'      211  17 B'      261  76 LBL
012  32 X:T     062  00  0      112  16  16     162  43 RCL     212  36 PGM     262  16 A'
013  44 SUM     063  42 STD     113  95  =      163  22  22     213  04  04     263  75  -
014  01  01     064  01  01     114  32 X:T     164  65  X      214  18 C'      264  01  1
015  92 RTN     065  42 STD     115  43 RCL     165  43 RCL     215  48 EXC     265  03  3
016  76 LBL     066  02  02     116  17  17     166  08  08     216  29  29     266  85  +
017  11  A      067  43 RCL     117  85  +      167  65  X      217  71 SBR     267  42 STD
018  99 PRT     068  06  06     118  43 RCL     168  43 RCL     218  02  02     268  01  01
019  32 X:T     069  65  X      119  07  07     169  12  12     219  31  31     269  85  +
020  92 RTN     070  43 RCL     120  85  +      170  95  =      220  01  1      270  02  2
021  76 LBL     071  14  14     121  43 RCL     171  32 X:T     221  94 +/-     271  09  9
022  12  B      072  65  X      122  15  15     172  43 RCL     222  49 PRD     272  95  =
023  99 PRT     073  43 RCL     123  19 D'      173  21  21     223  04  04     273  42 STD
024  22 INV     074  22  22     124  43 RCL     174  85  +      224  43 RCL     274  00  00
025  37 P/R     075  95  =      125  10  10     175  43 RCL     225  29  29     275  02  2
026  76 LBL     076  32 X:T     126  65  X      176  07  07     226  99 PRT     276  00  0
027  13  C      077  43 RCL     127  43 RCL     177  85  +      227  32 X:T     277  02  2
028  99 PRT     078  05  05     128  14  14     178  43 RCL     228  99 PRT     278  00  0
029  98 ADV     079  85  +      129  65  X      179  11  11     229  98 ADV     279  02  2
030  72 ST*     080  43 RCL     130  43 RCL     180  18 C'      230  92 RTN     280  00  0
031  00  00     081  13  13     131  18  18     181  92 RTN     231  42 STD     281  02  2
032  69 DP      082  85  +      132  95  =      182  71 SBR     232  00  00     282  00  0
033  20  20     083  43 RCL     133  32 X:T     183  02  02     233  02  2      283  02  2
034  32 X:T     084  21  21     134  43 RCL     184  04  04     234  03  3      284  00  0
035  72 ST*     085  19 D'      135  09  09     185  71 SBR     235  42 STD     285  69 DP
036  00  00     086  43 RCL     136  85  +      186  01  01     236  01  01     286  04  04
037  69 DP      087  12  12     137  43 RCL     187  98  98     237  06  6      287  69 DP
038  20  20     088  65  X      138  13  13     188  71 SBR     238  42 STD     288  05  05
039  97 DSZ     089  43 RCL     139  85  +      189  01  01     239  02  02     289  25 CLR
040  01  01     090  20  20     140  43 RCL     190  92  92     240  73 RC*     290  91 R/S
041  00  00     091  65  X      141  17  17     191  92 RTN     241  01  01
042  20  20     092  43 RCL     142  18 C'      192  25 CLR     242  63 EX*
043  17 B'      093  10  10     143  43 RCL     193  01  1      243  00  00
044  43 RCL     094  95  =      144  16  16     194  07  7      244  72 ST*
045  02  02     095  32 X:T     145  65  X      195  61 GTD     245  01  01
046  42 STD     096  43 RCL     146  43 RCL     196  02  02     246  69 DP
047  04  04     097  11  11     147  20  20     197  06  06     247  20  20
048  32 X:T     098  85  +      148  65  X      198  25 CLR     248  69 DP
049  43 RCL     099  43 RCL     149  43 RCL     199  01  1      249  21  21
```

```
INITIALIZE, PRESS E

ENTER       10.      A
             5.      B
   26.56505118

            -7.      A
             4.      B
   150.2551187

             3.      A
           180.      C

            -7.      A
             4.      B
   150.2551187

            19.      A
            -7.      B
  -20.22485943

            12.      A
             8.      B
   33.69006753

             3.      A
           180.      C

            12.      A
             8.      B
   33.69006753

            19.      A
            14.      B
   36.38435182

           325.      A
            30.      C

          494.2      A
          27.08      C

           585.      A
            90.      C

          3199.    δ A
          2018.      JB

   51.15376437     X A
  -18.51217268       JB

   45.35337899     Y A
   8.327285169       JB

  -9.380847583     Z A
   10.42320496       JB
```

RPN MODULE.- Don Jones, Salisbury, NC, offers a few comments on the correct use of some aspects of this module.

1. The translation by the module of " Print: Space, 35-84" as NOP is certainly wrong. It should be ADV.
2. At times it may be desirable to skip a particular translation, while still incrementing the count of HP and TI steps, so that the original count sequence is maintained. This would be the case when the HP PRT code is skipped to avoid the five pauses generated by the module. To accomplish this, key ADV ADV ADV OP 23 OP 28 , write in the desired PRT step, then continue to key in HP code.
3. A listing of HP-97 and HP-67 code equivalents would be helpful. The HP programs usually list only for HP-97. Does anybody have such a list?
4. Comments on relative speed of translated versus original programs would be interesting. Has anybody ever investigated this aspect?

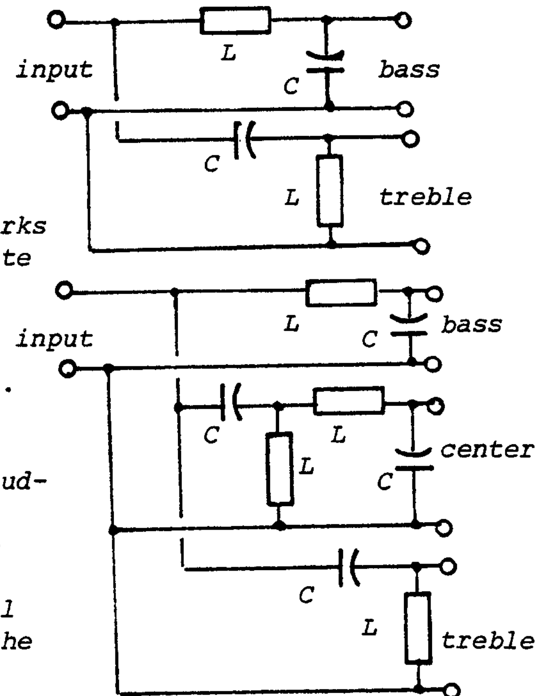
RE-Is converting a certain sequence into a SBR worth the trouble?-(v5n1p3) Jared Weinberger, Bologna, Italy, has optimized the routine some more, such that instead of 3 it uses only 2 registers and it has two steps less than the original one.(40 vs 42 steps) :

LBL A STO 00 R/S STO 01 R/S LBL E 2 - RCL 01 + LBL D 2 X LBL C RCL 01 +/- - RCL 01 + RCL 00 X RCL 01 - 3 + LNX X 0 = RTN

PASSIVE CROSS-OVER NETWORKS.- *(Maurice E.T. Swinnen) This program allows an audio designer to compute the parameters of inductances and capacitances in order to achieve the required cross-over frequencies of networks used in loudspeaker systems. Although electronic cross-over networks are available and are often used by manufacturers of hi-fi systems, the "true" amateurs, those that swear by tubes as opposed to semi-conductors for its rich reproduction tonality, also shun electronic cross-over networks and prefer the passive L-C kind.*

This program computes the values of L and C and the winding data for the coil(s). A conservative DC current density of 2 Amp/mm² wire cross section for copper wire is used. This value may be increased or decreased by changing the number at step 159. Coil dimensions in which the inside diameter is 1/2 of the outside diameter and in which the winding length is again 1/2 of the inside diameter seem to give the best results.

Although only two- and three-way networks are depicted here, you can device and compute the values of networks with any number of frequency cross-over points.



Two- and three-way parallel filters, 12 dB/octave.

The program will fit on one mag card and may be used with or without the printer. In the latter case you will have to do a little more button pushing.

It goes without saying that all the loudspeakers have to have the same impedance.

Instructions: (see also example at the beginning of the program listing)

1. *Initialize, press 2nd E'. This clears all memories, the t-reg and sets flag 7 if the printer is attached.*
2. *Enter the amplifier power in watts and press A.*
3. *Enter the loudspeaker impedance and press B.*
4. *Enter the cross-over frequency and press C. Required capacitance displayed and printed.*
5. *Enter closest normalized and available capacitor value in uF and press D. No printer: display shows inductance of required coil in mH. Press R/S. Diameter of required wire in mm displayed and printed.*
6. *Enter closest and normalized (preferrably next higher value) diameter of available lacquer-covered copper wire. Press E.*
   No printer: number of required turns. Press R/S to continue.
   No printer: length of coil in mm. Press R/S to continue.
   No printer: diameter of coil in mm. Press R/S to continue.
   No printer: resulting coil inductance in mH. Press R/S to continue.
   No printer: DC resistance of coil in ohms. Press R/S to continue.
   No printer: resulting cross-over frequency. Press R/S to continue.
   Percent difference between entered and resulting cross-over frequency.

With the printer there is no need to press R/S. All the above values are printed with suitable descriptors.

In case of multiple-way cross-over networks the next cross-over frequency point may now be entered. Go to step 4, above. DO NOT RE-INITIALIZE.

(Ref: PASSIVE LAUTSPRECHERWEICHEN, Burkhardt Mueller, Elrad, heft 3, 1980, pp 43-45.)

The formulas used in this program are:

$f = 1 / 2\cdot\pi\cdot\sqrt{L\cdot C}$ *(f= cross-over frequency, L=coil inductance in mH, C=capacity in (uF*

$C = 10^6 / 2\cdot\pi\cdot f\,\sqrt{2\cdot Z_L}$ *(ZL = loudspeaker impedance in ohms) C in uF*

$L = 10^9 / (2\cdot\pi\cdot f)^2\, C$*, coil impedance in mH.*

$d = 2\cdot P / \sqrt{8\cdot\pi\cdot P\cdot Z_L}$ *in which P=amplifier power in watts, d= wire diameter in mm.*

$A = d^2\cdot\pi / 4$ *(mm²) wire cross section*

(OVER)

CROSS-OVER NETWORKS, cont)

$\ell = L / 33 \cdot N^2$ *(mm)* ℓ*= winding length of coil, N= number of turns*

$Rd = 12 \cdot \ell \cdot N \cdot / 5600 \cdot d^2$ *(ohms) coil DC resistance,*

$$N = \sqrt[2.5]{\frac{L}{\sqrt{\frac{(d/2)\cdot 2\cdot \pi}{0.6}}\; 33 \cdot 10^{-6}}} \quad \textit{(number of turns)}$$

$\Delta\% = (f_1 - f_2) \cdot 100 / f_1$ *in which* f_1*=required f and* f_2*=resultant frequency.*

The voltage requirements for C are $2 \cdot \pi \cdot \sqrt{8 \cdot P \cdot Z_L}$ *(volts) in which P in watts and* Z_L *in mH. This would be easy to add to the program if needed.*

```
INITIALIZE 2ND E' ---------------

P=60, ENTER 60 A        60.      WATT
LS IMPEDANCE  8 B        8.      OHMS
FREQUENCY   500 C      500.      F HZ
C  IN UF                28.135   C UF
NORMALIZED C 30 D       30.000   C'UF
L-COIL MH                3.377   I MH
D-WIRE MM                0.795   D MM
NORMALIZED D .9 E        0.900   D'MM
# OF TURNS             100.      TURN
LENGTH COIL MM          10.3     L MM
DIAMETER COIL MM        20.6     D MM
NEW L-COIL MH            3.382   I'MH
DC RESIST COIL           0.272   ROHM
NEW FREQUENCY          500.      F'HZ
DIFFERENCE %             0.000     %

FREQUENCY  2000 C     2000.      F HZ
C IN UF                  7.034   C UF
NORMALIZED C 10 D       10.000   C'UF
L-COIL IN MH             0.633   I MH
D-WIRE IN MM             0.785   D MM
NORMALIZED D .9 E        0.900   D'MM
# OF TURNS              51.      TURN
LENGTH COIL IN MM        7.4     L MM
DIAMTR COIL IN MM       14.8     D·MM
NEW L-COIL MH            0.637   I'MH
R-COIL IN OHMS           0.099   ROHM
NEW FREQUENCY         1994.      F'HZ
DIFFERENCE %            -0.300     %
```

```
000  76 LBL
001  16 A'
002  55  ÷
003  01  1
004  02  2
005  22 INV
006  28 LOG
007  85  +
008  01  1
009  54  )
010  82 HIR
011  08  08
012  32 X:T
013  52 EE
014  22 INV
015  52 EE
016  69 OP
017  06  06
018  22 INV
019  58 FIX
020  92 RTN
021  76 LBL
022  11  A
023  42 STO
024  00  00
025  32 X:T
026  04  4
027  03  3
028  01  1
029  03  3
030  03  3
031  07  7
032  03  3
033  07  7
034  61 GTO
035  16 A'
036  76 LBL
037  12  B
038  42 STO
039  01  01
040  32 X:T
041  03  3
042  02  2
043  02  2
044  03  3
045  03  3
046  08  8
047  03  3
048  06  6
049  61 GTO
050  16 A'
051  76 LBL
052  13  C
053  42 STO
054  09  09
055  32 X:T
056  02  2
057  01  1
058  00  0
059  00  0
060  02  2
061  03  3
062  04  4
063  06  6
064  16 A'
065  65  ×
066  02  2
067  65  ×
068  89  π
069  65  ×
070  42 STO
071  06  06
072  43 RCL
073  01  01
074  65  ×
075  02  2
076  34 √X
077  55  ÷
078  01  1
079  52 EE
080  06  6
081  95  =
082  22 INV
083  52 EE
084  35 1/X
085  58 FIX
086  03  03
087  32 X:T
088  01  1
089  05  5
090  00  0
091  00  0
092  04  4
093  01  1
094  02  2
095  01  1
096  61 GTO
097  16 A'
098  76 LBL
099  14  D
100  42 STO
101  06  06
102  58 FIX
103  03  03
104  32 X:T
105  01  1
106  05  5
107  06  6
108  05  5
109  04  4
110  01  1
111  02  2
112  01  1
113  16 A'
114  65  ×
115  43 RCL
116  06  06
117  33 X²
118  55  ÷
119  01  1
120  52 EE
121  09  9
122  95  =
123  22 INV
124  52 EE
125  35 1/X
126  42 STO
127  02  02
128  58 FIX
129  03  03
130  87 IFF
131  07  07
132  23 LNX
133  91 R/S
134  76 LBL
135  23 LNX
136  32 X:T
137  07  7
138  04  4
139  00  0
140  00  0
141  02  2
142  08  8
143  02  2
144  03  3
145  16 A'
146  43 RCL
147  00  00
148  55  ÷
149  53  (
150  24 CE
151  65  ×
152  43 RCL
153  01  01
154  65  ×
155  08  8
156  54  )
157  34 √X
158  65  ×
159  02  2
160  55  ÷
161  89  π
162  95  =
163  34 √X
164  58 FIX
165  03  03
166  32 X:T
167  01  1
168  06  6
169  00  0
170  00  0
171  03  3
172  08  8
173  03  3
174  08  8
175  61 GTO
176  16 A'
177  76 LBL
178  15  E
179  42 STO
180  07  07
181  58 FIX
182  03  03
183  32 X:T
184  01  1
185  06  6
186  06  6
187  05  5
188  03  3
189  08  8
190  03  3
191  08  8
192  16 A'
193  33 X²
194  65  ×
195  89  π
196  55  ÷
197  02  2
198  93  .
199  04  4
200  95  =
201  42 STO
202  05  05
203  34 √X
204  65  ×
205  03  3
206  03  3
207  52 EE
208  06  6
209  94 +/-
210  55  ÷
211  43 RCL
212  02  02
213  22 INV
214  52 EE
215  95  =
216  35 1/X
217  45 YX
218  93  .
219  04  4
220  95  =
221  42 STO
222  04  04
223  58 FIX
224  00  00
225  87 IFF
226  07  07
227  42 STO
228  91 R/S
229  76 LBL
230  42 STO
231  32 X:T
232  03  3
233  07  7
234  04  4
235  01  1
236  03  3
237  05  5
238  03  3
239  09  9
240  16 A'
241  65  ×
242  43 RCL
243  05  05
244  95  =
245  34 √X
246  42 STO
247  03  03
248  58 FIX
249  01  01
250  87 IFF
251  07  07
252  43 RCL
253  91 R/S
254  76 LBL
255  43 RCL
256  32 X:T
257  02  2
258  07  7
259  00  0
260  00  0
261  03  3
262  08  8
263  03  3
264  08  8
265  16 A'
266  65  ×
267  02  2
268  95  =
269  58 FIX
270  01  01
271  87 IFF
272  07  07
273  44 SUM
274  91 R/S
275  76 LBL
276  44 SUM
277  32 X:T
278  01  1
279  06  6
280  00  0
281  00  0
282  03  3
283  08  8
284  03  3
285  08  8
286  16 A'
287  55  ÷
288  02  2
289  95  =
290  65  ×
291  03  3
292  03  3
293  52 EE
294  06  6
295  94 +/-
296  65  ×
297  43 RCL
298  04  04
299  33 X²
300  22 INV
301  52 EE
302  95  =
303  42 STO
304  02  02
305  58 FIX
306  03  03
307  87 IFF
308  07  07
309  32 X:T
310  91 R/S
311  76 LBL
312  32 X:T
313  32 X:T
314  07  7
315  04  4
316  06  6
317  05  5
318  03  3
319  08  8
320  02  2
321  03  3
322  16 A'
323  43 RCL
324  03  03
325  65  ×
326  43 RCL
327  04  04
328  65  ×
329  01  1
330  02  2
331  55  ÷
332  05  5
333  06  6
334  52 EE
335  03  3
336  55  ÷
337  43 RCL
338  07  07
339  33 X²
340  22 INV
341  52 EE
342  95  =
343  58 FIX
344  03  03
345  87 IFF
346  07  07
347  33 X²
348  91 R/S
349  76 LBL
350  33 X²
351  32 X:T
352  03  3
353  05  5
354  03  3
355  02  2
356  02  2
357  03  3
358  03  3
359  08  8
360  16 A'
361  43 RCL
362  02  02
363  55  ÷
364  01  1
365  52 EE
366  09  9
367  65  ×
368  43 RCL
369  08  08
370  95  =
371  34 √X
372  65  ×
373  02  2
374  65  ×
375  89  π
376  22 INV
377  52 EE
378  95  =
379  35 1/X
380  58 FIX
381  00  00
382  87 IFF
383  07  07
384  34 √X
385  91 R/S
386  76 LBL
387  34 √X
388  32 X:T
389  02  2
390  01  1
391  06  6
392  05  5
393  02  2
394  03  3
395  04  4
396  06  6
397  16 A'
398  75  -
399  43 RCL
400  09  09
401  95  =
402  65  ×
403  01  1
404  00  0
405  00  0
406  55  ÷
407  43 RCL
408  09  09
409  95  =
410  58 FIX
411  03  03
412  87 IFF
413  07  07
414  22 INV
415  91 R/S
416  76 LBL
417  22 INV
418  32 X:T
419  07  7
420  05  5
421  00  0
422  00  0
423  06  6
424  01  1
425  00  0
426  00  0
427  16 A'
428  98 ADV
429  91 R/S
430  00  0
431  35 1/X
432  00  0
433  91 R/S
434  76 LBL
435  10 E'
436  47 CMS
437  29 CP
438  02  2
439  00  0
440  69 OP
441  07  07
442  69 OP
443  19  19
444  25 CLR
445  02  2
446  00  0
447  02  2
448  00  0
449  02  2
450  00  0
451  02  2
452  00  0
453  02  2
454  00  0
455  69 OP
456  01  01
457  69 OP
458  02  02
459  69 OP
460  03  03
461  69 OP
462  04  04
463  69 OP
464  05  05
465  69 OP
466  00  00
467  98 ADV
468  91 R/S
```

SUPERPLOT II.- The Snow brothers, Robert and Richard, wrote this program last year. It has been circulated among the members of the local club and also among some of my friends in Germany. It is about time it is published in the TI PPC NOTES! The Roman II, of course, means that it is an enhancement of Superplot, the fantastic program written by Don O'Grady and contained in the Math/Utility module. As opposed to Don's program, the Snow version has less program steps: 239 vs 444. This way it allows the user to put in more and longer functions. Speed is about at par with MU-05, allowing for one being in module and the other one in user memory. But its main advantage is that allows not just ten functions as MU-05 does, but ANY number of them !

To put in your own functions is extremely simple: just press E and BST once. You will see the code for LRN (31).Overwrite that one with the first step of your own function. Use the same rules as the ones for MU-05: When the program calls the user-defined subroutine, x is in the t-register; when the subroutine is completed, the alpha code for the plotting character should be in the t-register. The f(x) should then be in the display. (see manual page 20)

The only difference with these rules is as follows:
End each subroutine with D, instead of with RTN as is done in MU-05. End the last of of your subroutines with RST. Do not start each routine with a label as in MU-05. Instead, simply write all your subroutines in one long chain, with D as a separator and RST at the very end.

Then, to use the program, that is to enter the different parameters, do as follows: Enter initial x-valye and press A.
Enter x-increment and press R/S.
Enter y-minimum and press B.
Enter y-maximum and press R/S.
Enter the number of points to be plotted and press C.
Enter the number of tapes desired and press R/S.

In the example, two functions are plotted simultaneously: sin(x) and sin(x) / x Both are plotted in radian mode, starting at -4π. The x-increment is π/8 while the y-limits are -.25 and 1. The number of points is 60 and we want just 2 tapes, which get pasted side by side to reveal the entire plot.

For sin(x) plotting we use "e", code 54, while for sin(x)/x plotting we employ the "0" symbol, code 81. (just to be different, friends, no other reason) Both subroutines are written in one, long chain, following LBL E, steps 232 to 253.

will always be plotted first. The program start plotting automatically after the last entry, i.e. number of tapes desired.

If you write only one, single subroutine, consider that one to be "the last one" and consequently end it with RST.

Record the entire program on one bank of a mag card.

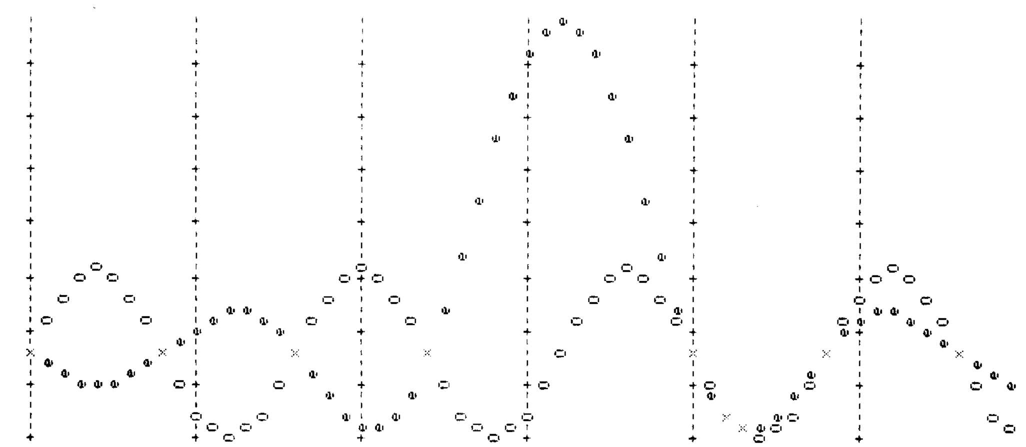
The key strokes for the example are:

4 +/- × π = press A
π ÷ 8 = press R/S
.25 +/- press B
1 press R/S
60 press C
2 press R/S

SEE PROGRAM ON NEXT PAGE.

TAX COMPUTATION MODULE.- The Tax Management Inc., 1231 25th Street, NW, Washington DC, 20037 sent me a flyer, announcing the availability of two tax calculation modules: the Federal Income Tax module and the Estate Tax module. The first one computes regular income tax, maximum tax on personal service income, income averaging tax, minimum tax, alternative minimum tax, medical expense deduction, earned income credit, etc, while the second one specializes in marital deduction, federal estate tax, interrelated marital and/or charitable deduction amount, current value of portfolio investment , etc. Both for use with the TI-59/PC100A. No prices given for the modules or the accompanying manuals. I have not seen these modules, nor do I know the quality or lack thereof. Please write to the address above if interested.

```
230  76 LBL
231  15  E
232  05  5
233  04  4
234  53  (
235  32 X:T
236  70 RAD
237  42 STD
238  26  26
239  38 SIN
240  55  ÷
241  43 RCL
242  26  26
243  54  )
244  14  D
245  08  8
246  01  1
247  53  (
248  32 X:T
249  38 SIN
250  55  ÷
251  04  4
252  54  )
253  81 RST
```



```
000  14  D
001  43 RCL
002  01  01
003  69 DP
004  01  01
005  43 RCL
006  02  02
007  69 DP
008  02  02
009  43 RCL
010  03  03
011  69 DP
012  03  03
013  43 RCL
014  04  04
015  69 DP
016  04  04
017  69 DP
018  05  05
019  43 RCL
020  11  11
021  44 SUM
022  10  10
023  97 DSZ
024  09  09
025  02  02
026  01  01
027  98 ADV
028  97 DSZ
029  15  15
030  01  01
031  41  41
032  00  0
033  92 RTN
034  76 LBL
035  14  D
036  75  -
037  43 RCL
038  12  12
```

```
039  54  )
040  55  ÷
041  43 RCL
042  13  13
043  54  )
044  58 FIX
045  00  00
046  52 EE
047  22 INV
048  52 EE
049  22 INV
050  58 FIX
051  55  ÷
052  05  5
053  85  +
054  05  5
055  32 X:T
056  82 HIR
057  05  05
058  82 HIR
059  14  14
060  54  )
061  42 STD
062  14  14
063  77  GE
064  01  01
065  37  37
066  32 X:T
067  00  0
068  77  GE
069  01  01
070  37  37
071  32 X:T
072  22 INV
073  59 INT
074  65  ×
075  01  1
076  00  0
077  75  -
```

```
078  08  8
079  54  )
080  22 INV
081  28 LDG
082  52 EE
083  22 INV
084  52 EE
085  82 HIR
086  08  08
087  65  ×
088  73 RC*
089  14  14
090  54  )
091  59 INT
092  55  ÷
093  01  1
094  00  0
095  00  0
096  54  )
097  22 INV
098  59 INT
099  67  EQ
100  01  01
101  29  29
102  65  ×
103  01  1
104  00  0
105  00  0
106  55  -
107  32 X:T
108  82 HIR
109  18  18
110  54  )
111  22 INV
112  74 SM*
113  14  14
114  02  2
115  00  0
116  67  EQ
```

```
117  01  01
118  29  29
119  04  4
120  07  7
121  67  EQ
122  01  01
123  29  29
124  05  5
125  00  0
126  61 GTD
127  01  01
128  31  31
129  82 HIR
130  15  15
131  55  ÷
132  82 HIR
133  18  18
134  54  )
135  74 SM*
136  14  14
137  43 RCL
138  10  10
139  32 X:T
140  92 RTN
141  04  4
142  82 HIR
143  54  54
144  43 RCL
145  16  16
146  42 STD
147  09  09
148  65  ×
149  43 RCL
150  11  11
151  54  )
152  22 INV
153  44 SUM
154  10  10
155  61 GTD
```

```
156  02  02
157  05  05
158  76 LBL
159  11  A
160  42 STD
161  10  10
162  92 RTN
163  42 STD
164  11  11
165  92 RTN
166  76 LBL
167  12  B
168  42 STD
169  12  12
170  94 +/-
171  42 STD
172  13  13
173  94 +/-
174  92 RTN
175  44 SUM
176  13  13
177  92 RTN
178  76 LBL
179  13  C
180  42 STD
181  16  16
182  42 STD
183  09  09
184  92 RTN
185  42 STD
186  15  15
187  65  ×
188  02  2
189  00  0
190  75  -
191  01  1
192  82 HIR
193  04  04
194  54  )
```

```
195  22 INV
196  49 PRD
197  13  13
198  61 GTD
199  02  02
200  05  05
201  97 DSZ
202  17  17
203  02  02
204  18  18
205  01  1
206  00  0
207  42 STD
208  17  17
209  04  4
210  07  7
211  02  2
212  00  0
213  02  2
214  00  0
215  02  2
216  00  0
217  02  2
218  00  0
219  42 STD
220  01  01
221  42 STD
222  02  02
223  42 STD
224  03  03
225  42 STD
226  04  04
227  43 RCL
228  10  10
229  32 X:T
230  76 LBL
231  15  E
232  31 LRN
```

```
REGISTERS USED:

R01  DP01
R02  DP02
RD3  DP03
R04  DP04
R09  # POINTS
R10  X-VALUE
R11  X-INCREMENT
R12  Y-MINIMUM
R13  Y-RESOLUTION
R14  PLOT DSZ
R15  TAPES DSZ
R16  # POINTS DSZ
R17  +---- 10 DSZ
```

```
TO SYNTHESIZE
CODE 31 (LRN)

PRESS STD 31
BST BST DEL SST
```

LABELS-TO-DIRECT ADDRESSES .- *In v5n7p2/3 we presented John Worthington's program by that name. Lots of favorable comments have arrived here. But at the same time, some members question the wisdom of publishing the other three programs we have received so far. They claim "not everybody is that fanatic about it." So, I have asked the printer to produce a nice, presentable package of the three programs in question. If you want them, send me a letter or a card. The description of them follows:*

Labels-to-Direct-Addresses Conversion Routine. Panos Galidas, Rockville MD. Five pages of instructions, two pages of program listing in 50% reduction format. Instructions also contain the algorithms the program is based on.

Label-to-Direct-Address Program Aid.- Richard Snow, Vallejo, CA. A 27-step routine computes the address if you don't mind entering the print code from the keyboard. Otherwise a longer program will do everything automatically. Two pages of instructions and three pages of full-size program listing.

Optimizing a Program? Let the Calc' Handle it.- Jeff Rosedale, Yorktown Heights, NY. Four pages of full-size text and program listings. Instructions, examples.

I'll send you this 16-page compilation by first-class mail for $ 4.00 US. Members living outside the US (except Canada and Mexico) please send an extra dollar if you want it air mailed to you.

MORE QUIRKS.- *Patrick Acosta, San Antonio, Texas, has been experimenting on his TI-58C with some very unusual behavior. What follows is a abbreviated version of his description of those quirks. I found that everything works as well on a simple 58 and on a 59.*
The special sequence from the keyboard used in all the different quirks is:

STF IND 3 LIST *(NOTE: this is not 03 but simply 3.)*

The above sequence, from turn-on, will put the calculator in a true FIX 9, not a floating point state.Note, when the printer is connected and you turn on the TRACE mode, how all your entries and results are printed.
In the 58C, when turned off, the FIX state is always preserved, but not this one.
Instead of 3 you can use any single digit, provided the contents of the corresponding register is equal to or less than zero.
It is also possible to set several, or even all flags by means of this sequence.
Use : STF IND 3 INV *(remember NOT 03 but simply 3)*

Flags will be set according to the following rules:

9	*8*	*7*	*6*	*5*	} *Flags*
4	*3*	*2*	*1*	*0*	}
d1	*d2*	*d3*	*d4*	*d5*	*is the # in the display, with d1 = MSD (most significant digit)*

If dn ≡ 0 mod 4, no flags are set.
If dn ≡ 1 mod 4, the corresponding bottom flag will be set.
If dn ≡ 2 mod 4, the corresponding top flag will be set.
If dn ≡ 3 mod 4, both flags will be set.

Note to newcomers: The above notation might not be familiar to you. It is best explained by means of an example. If dn= 7, then we divide 7 so many times by 4 until the remainder is less than 4. Here it becomes simply 3. So, both top and bottom flags will be set. Thus, 3 mod 4 means : 7 divided by 4 = 1. The remainder only is of interest to us here: 3.
Note that only the MSD will control the setting of the flags. As an example:
33333 STF IND 3 INV will set all flags.
11111 STF IND 3 INV will set flags 4, 3, 2, 1 and 0.
22222 STF IND 3 INV will set flags 9, 8, 7, 6 and 5.
22111 STF IND 3 INV will set flags 9, 8, 2, 1 and 0.
The number in the display need not have a minimum of five digits:
33 will set flags 9, 8, 4 and 3.
1444233 sets flags 4 and 5 only.(Only the MSD control the setting, here 14442.
5677 will set flags 4, 8, 7, 2, 6 and 1.

The sequence 22 SIN INV SIN STF IND 3 INV CE *gets you in ROM. If you now press LRN you will see 7992 00. Putting some program steps in steps 7992, 7993.... will show you key codes elevated by 6. But if you first pressed GTO 000 and put the program steps starting at step 000, you will note that the key codes have been reduced by 10. (Do not press RST for GTO 000. It will kill this special state.)*
This sequence leaves a lot of room for experimentation. For example, press

Σ+ 22 SIN INV SIN STF IND 3 INV LRN *and you will see at steps 7992.... the first steps of the INV P/R program, which in ROM follows the Σ+ routine.*
Writing code in steps 7992 to 7999 will put the code in steps 312 to 319 (reg 20)

BIDDING SYSTEMS.- *Richard Jameson brings to my attention the availability of special programming systems that help those who are doing "bidding for jobs" as a regular, professional activity. These programming systems are on mag cards and come with extensive documentation how to use them. In the BIDDING system there is a PHASE I and a PHASE II system. Additionally there is a system called JOB COST and another one called WORKING CAPITAL. For more information and prices, write CAL-Q-TECH, 262 State Street, Augusta, Maine 04330 or call (207) 622-7183.*

BOOKS, BOOKS, BOOKS.- Steven Grisham brings to my attention the existence of the following books, of interest to programmable calculator fans:

The first four books are available from Herbert A. Luft, P.O. Box 91, Oakland Gardens, N.Y. 11364:

* Astronomical Formulae for calculators, by J. Meeus, 185 pages, 1979, soft-bound, limited supply only, European import, $ 8.50.
* Mathematical Astronomy with calculators, by A. Jones, 254 pages, 1978, $ 15.95.
* Practical Astronomy with your calculator, by P. Duffett-Smith, 208 pages, 1979, cloth-bound at $ 22.50, soft-bound at $ 7.75.
* Positional Astronomy and Astro-Navigation made easy, by H.R. Mills, 267 pages, 1978, a new approach using the pocket calculator, $ 24.95.
* Calculator Navigation, by Mortimer Rogoff, 417 pages, hard-bound, $ 24.95. Topics covered are coastwise navigation, sailing, celestial navigation and Loran. Statistical approach to many of the classical navigational techniques. Publisher: W.W. Morton & Co., 500 5th Ave, New York NY, 10110.
* Syhnthetic Hydrograph Computations on small programmable calculators, by Thomas E. Croley II, 236 pages, hard-cover loose-leaf, $ 12.00. Publisher: Institute of Hydraulic Research, The University of Iowa, Iowa City, Iowa, 52042.

ERRATUM- In v5n7p4 I talked about a Fast Fourier Transform program written by member R.E. Harrisson and published in Electronics. That last part of the information was wrong: It was published in Electronic Design. The date of publication mentioned is correct. May 10, 1980. My apologies to Fred Fullam and the many others whom I managed to confuse utterly.

OPTICS.- The Engineering magazine of Electro-Optical and Laser Technology, called EOSD, for Electro-Optical System Design, has consistently published good programs to be used on both RPN and AOS calculators. The main promotor behind the calculator column has been Dr. Robert Pitlak, who is presently with Inrad, 181 Legrand, Northvale NJ, 07647. Some of the programs have used very sophisticated techniques, such as iteration. Once in a while, though, I see offending sequences, such as SBR A, where a simple A would do as well.

In the August 1980 issue of EOSD, Robert publishes a program called SHG Phase-Match Angles, to be used on the TI-59. EOSD is a Kiver/Cahners Publication, 5 South Wabash Ave., Chicago IL, 60603, tel. (312) 372-6880.

FINANCIAL SYSTEM REPORT.-Vernon K. Jacobs, CPA, CLU, is the editor of a magazine by that name. It is a monthly review of new computer programs, products and services for financial and tax advisors. Mr. Jacobs is no newcomer to this business: He is also the editor of Tax Angles, a monthly strategy newsletter for the general public; he writes a monthly column on taxes for Private Praxis, a publication for doctors, and another one for Real Estate Intelligence Report, a newsletter for real estate investors. His book The Taxpayer's Counterattack was published by Kephart Communications (1979) in Alexandria VA. Mr. Jacobs is also the president of Syntax Corporation, the publisher of the Financial System Report. If you are interested in all of this, write Syntax Corp., 4500 West 72nd Terrace, Prairie Village, Kansas 66208 or call (913) 362-9667.

COVERAGE PREDICTIONS USING PROGRAMMABLE CALCULATORS.- Ed Westenhaver, a TI PPC Club member, is a patent specialist with the Harris Corporation, Broadcast Products Division, P.O. Box 4290, Quincy IL, 62301. Ed has written TI-59, HP-67/97 and Basic language programs to compute coverage in broadcasting for the FM and TV channels. His compnay makes this 28-page article available free to the Braodcast community. In his program Ed uses a rather unique method to curve fit an emperical family of curves, such as the FCC 50,50 field intensity charts. The problem presented to him was a curve fit to a long polynomial, which is, of course, rather impractical to be evaluated manually. Ed will be glad to correspond with anyone who may be interested in his method of arriving at the curve fit equation.

FIRST CLASS MAIL- (v5n7p13) Richard Snow sends me an optional algorithm. Suppose the number of ounces is in R00:

```
.....RCL 00 + INV INT OP 10 ) INT X 13 + 2 = R/S
```

TI-57 LISTING ON A TI-59.- We have had SR-52 and SR-56 listing programs in previous issues. Now Bill Skillman has written this one for TI-57 listing. Somewhere else in this issue you will find a speedy factor finder program for the TI-57, which I listed by means of the present program. Doesn't it look neat? The following instructions apply:

1. Read the two mag cards, banks 1 through 4.
2. Enter the starting step and press A. The digit "1" will flash in the display. DO NOT PRESS CE OR CLR TO STOP THE FLASHING. If no entry is made, the default starting step will be 00.
3. Press key with the same function (name) as on the TI-57. The only exceptions are:
   for c.t. press CP
   for σ^2 press E (prints e^2)
   Non-merged codes are printed immediately. Merged codes display first a steady 0. Then press digit 0 through 9 followed by R/S. Merged code is printed entirely.
   INV is a special case: press INV before main code. The digit -1, flashing, is displayed, without printing. When subsequently the main code is entered, both INV and the main code are printed.
   Numbers are entered by first pressing D. A steady 0 is displayed. Enter number followed by R/S.
   To return to normal keyboard control without flashing, press LRN twice.

```
        100.       02
       1000.       03
      10000.       04
     100000.       05
  100000000.       06
          0.       07
     200000.       08
   24314200.       09
273144.204101      10
  1517.205102      11
152735.206103      12
273222.209104      13
154037.212105      14
371331.301106      15
446237.303107      16
  5070.304108      17
  5070.304109      18
 26350.306112      19
163036.307201      20
336335.308202      21
362431.309203      22
153236.312204      23
    53.401205      24
363732.403206      25
351527.404207      26
364130.405208      27
  4566.406209      28
331341.407212      29
174415.409301      30
333516.412302      31
245024.501303      32
  1717.503304      33
    55.504305      34
    56.505306      35
    72.506307      36
313233.507308      37
212444.509309      38
243137.512312      39
161722.601401      40
223732.602402      41
    50.606403      42
163646.607404      43
351316.701405      44
361435.702406      45
    20.706407      46
446437.707408      47
223516.801409      48
353637.802412      49
    47.806501      50
221700.807502      51
  5470.901503      52
356336.902504      53
    40.904505      54
476320.905506      55
    64.906507      56
271427.907508      57
  7747.909509      58
    67.912512      59
```

```
000  82 HIR
001  36  36
002  01  1
003  32 X:T
004  82 HIR
005  38  38
006  69 OP
007  05  05
008  73 RC*
009  00  00
010  65  ×
011  43 RCL
012  03  03
013  95  =
014  22 INV
015  59 INT
016  65  ×
017  43 RCL
018  05  05
019  95  =
020  69 OP
021  00  00
022  69 OP
023  01  01
024  43 RCL
025  06  06
026  82 HIR
027  08  08
028  69 OP
029  20  20
030  61 GTO
031  04  04
032  76  76
033  76 LBL
034  17 B'
035  25 CLR
036  91 R/S
037  32 X:T
038  09  9
039  32 X:T
040  22 INV
041  67 EQ
042  00  00
043  46  46
044  85  +
045  02  2
046  85  +
047  01  1
048  85  +
049  32 X:T
050  92 RTN
051  76 LBL
052  14  D
053  17 B'
054  43 RCL
055  02  02
056  95  =
057  81 RST
058  76 LBL
059  11  A
060  93  .
061  85  +
062  01  1
063  00  0
064  95  =
065  42 STO
066  00  00
067  25 CLR
068  29 CP
069  69 OP
070  00  00
071  81 RST
072  76 LBL
073  23 LNX
074  43 RCL
075  10  10
076  76 LBL
077  18 C'
078  10 E'
079  81 RST
080  76 LBL
081  10 E'
082  65  ×
083  59 INT
084  32 X:T
085  01  1
086  95  =
087  22 INV
088  59 INT
089  65  ×
090  43 RCL
091  03  03
092  95  =
093  59 INT
094  92 RTN
095  76 LBL
096  42 STO
097  43 RCL
098  35  35
099  76 LBL
100  19 D'
101  10 E'
102  82 HIR
103  36  36
104  32 X:T
105  52 EE
106  04  4
107  42 STO
108  01  01
109  25 CLR
110  43 RCL
111  04  04
112  82 HIR
113  46  46
114  82 HIR
115  48  48
116  87 IFF
117  02  02
118  01  01
119  22  22
120  17 B'
121  43 RCL
122  29  29
123  43 RCL
124  01  01
125  95  =
126  32 X:T
127  81 RST
128  76 LBL
129  22 INV
130  43 RCL
131  09  09
132  69 OP
133  03  03
134  01  1
135  94 +/-
136  32 X:T
137  61 GTO
138  04  04
139  72  72
140  76 LBL
141  24 CE
142  43 RCL
143  11  11
144  18 C'
145  76 LBL
146  25 CLR
147  43 RCL
148  12  12
149  18 C'
150  76 LBL
151  43 RCL
152  43 RCL
153  26  26
154  19 D'
155  76 LBL
156  28 LOG
157  43 RCL
158  13  13
159  18 C'
160  76 LBL
161  29 CP
162  43 RCL
163  14  14
164  18 C'
165  76 LBL
166  30 TAN
167  43 RCL
168  15  15
169  18 C'
170  76 LBL
171  32 X:T
172  43 RCL
173  16  16
174  18 C'
175  76 LBL
176  33 X²
177  43 RCL
178  17  17
179  18 C'
180  76 LBL
181  34 FX
182  43 RCL
183  18  18
184  18 C'
185  76 LBL
186  35 1/X
187  43 RCL
188  19  19
189  18 C'
190  76 LBL
191  88 DMS
192  43 RCL
193  20  20
194  18 C'
195  76 LBL
196  37 P/R
197  43 RCL
198  21  21
199  18 C'
200  76 LBL
201  38 SIN
202  43 RCL
203  22  22
204  18 C'
205  76 LBL
206  39 COS
207  43 RCL
208  23  23
209  18 C'
210  76 LBL
211  89  π
212  43 RCL
213  24  24
214  18 C'
215  76 LBL
216  44 SUM
217  43 RCL
218  27  27
219  19 D'
220  76 LBL
221  45 YX
222  43 RCL
223  28  28
224  18 C'
225  76 LBL
226  66 PAU
227  43 RCL
228  29  29
229  18 C'
230  76 LBL
231  48 EXC
232  43 RCL
233  30  30
234  19 D'
235  76 LBL
236  49 PRD
237  43 RCL
238  31  31
239  19 D'
240  76 LBL
241  50 IXI
242  43 RCL
243  32  32
244  18 C'
245  76 LBL
246  52 EE
247  43 RCL
248  33  33
249  18 C'
250  76 LBL
251  53  (
252  43 RCL
253  34  34
254  18 C'
255  76 LBL
256  54  )
257  43 RCL
258  35  35
259  18 C'
260  76 LBL
261  55  ÷
262  43 RCL
263  36  36
264  18 C'
265  76 LBL
266  68 NOP
267  43 RCL
268  37  37
269  18 C'
270  76 LBL
271  58 FIX
272  43 RCL
273  38  38
274  19 D'
275  76 LBL
276  59 INT
277  43 RCL
278  39  39
279  18 C'
280  76 LBL
281  60 DEG
282  43 RCL
283  40  40
284  18 C'
285  76 LBL
286  61 GTO
287  43 RCL
288  41  41
289  19 D'
290  76 LBL
291  65  ×
292  43 RCL
293  42  42
294  18 C'
295  76 LBL
296  97 DSZ
297  43 RCL
298  43  43
299  18 C'
300  76 LBL
301  70 RAD
302  43 RCL
303  44  44
304  18 C'
305  76 LBL
306  71 SBR
307  87 IFF
308  01  01
309  03  03
310  12  12
311  22 INV
312  86 STF
313  02  02
314  43 RCL
315  45  45
316  19 D'
317  76 LBL
318  75  -
319  43 RCL
320  46  46
321  18 C'
322  76 LBL
323  67 EQ
324  43 RCL
325  47  47
326  18 C'
327  76 LBL
328  80 GRD
329  43 RCL
330  48  48
331  18 C'
332  76 LBL
333  81 RST
334  43 RCL
335  49  49
336  18 C'
337  76 LBL
338  85  +
339  43 RCL
340  50  50
341  18 C'
342  76 LBL
343  77 GE
344  43 RCL
345  51  51
346  18 C'
347  76 LBL
348  15  E
349  43 RCL
350  52  52
351  18 C'
352  76 LBL
353  91 R/S
354  43 RCL
355  53  53
356  18 C'
357  76 LBL
358  93  .
359  43 RCL
360  54  54
361  18 C'
362  76 LBL
363  94 +/-
364  43 RCL
365  55  55
366  18 C'
367  76 LBL
368  95  =
369  43 RCL
370  56  56
371  18 C'
372  76 LBL
373  76 LBL
374  43 RCL
375  57  57
376  19 D'
377  76 LBL
378  78 Σ+
379  43 RCL
380  58  58
381  18 C'
382  76 LBL
383  79  x̄
384  43 RCL
385  59  59
386  18 C'

472  43 RCL
473  08  08
474  86 STF
475  01  01
476  82 HIR
477  36  36
478  32 X:T
479  71 SBR
```

RECURSIVE SIMPSON RULE.- Norman Herzberg, Princeton NJ, who is the author of this program starts his letter as follows:" This is really a fan letter. What good things keep coming from you! Keep up the good work." Of course, flattery will get you into the TI PPC NOTES any time. Keep those good letters coming, Norman! This program might seem similar to others you have seen before. You will soon detect that it is just a little different. Especially when you start checking its speed. It is indeed very fast and very accurate. This program does Numerical Integration. To be just "fastidiously technical", it uses Rhomberg acceleration of Simpson's Rule to achieve accuracy on the order of the 8th power of the step size.

Let us briefly review what is meant by Numerical Integration. Suppose you want to integrate a function f(x) for x between t_0 and t_1. For a "step size" h is selected and then the function to be integrated is evaluated at the points $x=t_0$, then $x=t_0+h$.... until we reach $x=t_1$. Then some formula is applied to the function values to get an estimate of the value of the integral. If the programmer has been careful, he/she will be able to estimate the error made in using the formula. This will depend on the complexity of the formula and on the step size h. Complex formulas and small step sizes yield better estimates. Given a program, and so being locked into a formula, the only way to get a better answer is to use a smaller step zize.

The novelty of this program, besides being an extremely compact one for the great accuracy which it yields, is that it is iterative. Each successive pass through the algorithm decreases the error by about 1/64. Moreover, unlike all the other programs I have seen, none of the computation is wasted. All of the previous computation is used, and needed, to obtain the next estimate. The net result is that, if the user wants greater accuracy he/she must wait only about 1/2 of the time a non-iterative method would demand. Moreover, the change between successive iterations is a good guide to the actual error being made. Since the author's method is a bit opaque (his own words and admission) he has included an error estimate (free): after each pass look in the t-register. In case the explanation is unclear,

A = > enter t0
B = > enter t1
E = > evaluate first estimate
C = > continue first estimate.

Note that half the total number of steps used to evaluate the estimate is briefly flashed after pressing C. The iterative nature of the program allows to save half of the necessary new work, by reusing the old. To see an estimate of the error, look in the t-register after any iteration.

Finally, to get yet greater accuracy one could use the following formula:

Inew + ((Inew - Iold) / 255) . Norman says that there are an infinite number of such formulas . He stopped with a formula that is exactly correct for polynomials up to degree 8. The next formula would be for degree 10, etc.

This program was adapted from one Norman once wrote for the SR-56. Yes, he managed to squeeze that one into 74 steps, leaving 26 steps for f(x).

Norman would like to see some speed trials conducted against any other program you may have. A good benchmark would be to see what the integral of e^{-x^2} is from .5 to 1.5 to 10 places. Or choose your own benchmark.

Norman proposes a very interesting benchmark himself , by using the so-called elliptical integral:

$$E(k) = \int_0^{\pi/2} \sqrt{1 - k^2 \sin^2(x)}\, dx$$

The circumference of the ellipse with minor axis=a and major axis=b is $4 \cdot b \cdot E(k)$ where $k^2 = 1 - (a/b)^2$. To test the program check that $E(1) = \pi/2$.

RECURSIVE SIMPSON RULE- *by Norman Herzberg.*

```
            0.        T 0
            2.        T 1

      102.4375
   .4375000075
  -.0751953124

   102.4001465
   .4001464798
  -.0017585756

   102.4000006
   .4000005752
  -.0000297575

         102.4
   .4000000103
  -.0000004738

         102.4
   .4000000033
  -.0000000075

         102.4
   .4000000012
  -.0000000002

         102.4
   .3999999974
  -.0000000001

         102.4
   .3999999953
   -3.3333333-11
```

```
000  76 LBL    038  75  -     076  44 SUM    114  75  -     152  01  1
001  14  D     039  43 RCL    077  17  17    115  32 X:T    153  76 LBL
002  61 GTO    040  06  06    078  43 RCL    116  43 RCL    154  93  .
003  01  01    041  55  ÷     079  17  17    117  12  12    155  85  +
004  80  80    042  02  2     080  95  =     118  95  =     156  03  3
005  76 LBL    043  95  =     081  65  ×     119  55  ÷     157  07  7
006  13  C     044  42 STO    082  43 RCL    120  06  6     158  00  0
007  98 ADV    045  00  00    083  06  06    121  03  3     159  00  0
008  86 STF    046  43 RCL    084  55  ÷     122  95  =     160  00  0
009  01  01    047  16  16    085  06  6     123  85  +     161  00  0
010  03  3     048  42 STO    086  95  =     124  32 X:T    162  95  =
011  42 STO    049  09  09    087  42 STO    125  95  =     163  69 OP
012  13  13    050  66 PAU    088  11  11    126  99 PRT    164  04  04
013  01  1     051  66 PAU    089  87 IFF    127  22 INV    165  32 X:T
014  42 STO    052  43 RCL    090  01  01    128  59 INT    166  69 OP
015  16  16    053  11  11    091  99 PRT    129  99 PRT    167  06  06
016  43 RCL    054  42 STO    092  85  +     130  32 X:T    168  91 R/S
017  08  08    055  12  12    093  53  (     131  99 PRT    169  76 LBL
018  42 STO    056  76 LBL    094  24 CE     132  98 ADV    170  12  B
019  06  06    057  47 CMS    095  75  -     133  91 R/S    171  42 STO
020  14  D     058  43 RCL    096  43 RCL    134  76 LBL    172  08  08
021  42 STO    059  00  00    097  10  10    135  15  E     173  32 X:T
022  17  17    060  14  D     098  54  )     136  22 INV    174  02  2
023  43 RCL    061  44 SUM    099  55  ÷     137  86 STF    175  61 GTO
024  03  03    062  18  18    100  01  1     138  01  01    176  93  .
025  22 INV    063  43 RCL    101  05  5     139  02  2     177  00  0
026  44 SUM    064  06  06    102  95  =     140  22 INV    178  00  0
027  06  06    065  22 INV    103  76 LBL    141  49 PRD    179  00  0
028  14  D     066  44 SUM    104  99 PRT    142  06  06    180  45 YX
029  44 SUM    067  00  00    105  48 EXC    143  49 PRD    181  09  9
030  17  17    068  97 DSZ    106  11  11    144  16  16    182  95  =
031  76 LBL    069  09  09    107  42 STO    145  61 GTO    183  52 EE
032  49 PRD    070  47 CMS    108  10  10    146  49 PRD    184  22 INV
033  00  0     071  02  2     109  97 DSZ    147  76 LBL    185  52 EE
034  42 STO    072  65  ×     110  13  13    148  11  A     186  92 RTN
035  18  18    073  43 RCL    111  15  E     149  42 STO
036  43 RCL    074  18  18    112  43 RCL    150  03  03
037  08  08    075  85  +     113  11  11    151  32 X:T
```

```
00   33 1        STO 1
01   15          CLR
02   32 2        STO 2
03   61 0        SBR 0
04   01          1
05   61 9        SBR 9
06   61 0        SBR 0
07   86 7        LBL 7
08   61 8        SBR 8
09   61 8        SBR 8
10   04          4
11   61 1        SBR 1
12   06          6
13   61 9        SBR 9
14   06          6
15   61 1        SBR 1
16   33 2        RCL 2
17   23          4
18   22          X:T
19   33 1        RCL 1
20   76          x≥t
21   51 7        GTO 7
22   84          +/-
23   81          R/S
24   86 8        LBL 8
25   04          4
26   86 9        LBL 9
27   61 1        SBR 1
28   86 0        LBL 0
29   02          2
30   86 1        LBL 1
31   34 2        SUM 2
32   86 5        LBL 5
33   19          C.t
34   33 1        RCL 1
35   45          ÷
36   33 2        RCL 2
37   85          =
38  -49      INV INT
39  -66      INV X=T
40  -61      INV SBR
41   33 1        RCL 1
42   22          X:T
43   33 2        RCL 2
44  -39 1    INV PRD 1
45   66          X=T
46   84          +/-
47   81          R/S
48   51 5        GTO 5
```

SPEEDY FACTOR FINDER FOR THE TI-57.-

Bill Skillman translated his SR-56 speedy factor finder into suitable 57ese. Achieved times are not bad at all:
987654 EE 3 + 321 = INV EE resolves into 3, 17 and 379721 in 2 min 29 sec.
103569859 resolves into 463, 467 and 479 in 1 min 55 sec.

User instructions:

1. Enter program.
2. Enter integer, press RST and R/S.
   Factors will be displayed in ascending order. Press R/S each time to display the next factor. Last factor will be displayed as a negative value.

Repeat 2) for next integer.

Registers used: 1 used -No
                2 trial divisor
                9 t-reg

MASTER LIBRARY LABEL PROGRAM- Jerome K. Porter, Corte Madera, CA, writes that, when he first saw the programs by that name in v5n1 he "was charmed by the eloquence and dismayed at the inconvenience of having to key-punch a different card for each of the programs of the Master Library. (or any other module)." So he set out to write a UNIVERSAL PROGRAM that can be used with any program in practically any module. It will simply print as a descriptor the user-defined key you pressed to enter your problem. It further has a print out of the module number and name and the number of the particular program you have accessed.

Jerome had an initializing step in the original program he sent me that I found "not too friendly to the user." It said "partition 9 OP 17." Then it said "Select PGM and enter nn SBR PGM." As he had a few steps steps left over I wrote an automatic partitioning command under his LBL PGM. It saves the user five key strokes each time.

Record this program on one card side with the normal turn-on partition 6 OP 17.

User instructions:

1. Read-in the program, one card side.
2. Select the program in the module you have plugged in and enter nn (PGM #) SBR PGM.
3. Make entries in accordance with program library instructions.

NOTE: I realize that this "universal program" will not necessarily work with any and all the programs in any module. Some programs require a partition that will interfere with this one. In others, the simple print-out of the user-defined key will not suffice. Again, some modules have their own print commands already written in. But in the great majority it will constitute a "quick and dirty" solution and enhance the module programs greatly.

```
          MASTER
  1.
 24.         PGM

 10.           A
25.4

50.8          A'
 20.
```

```
000  55  -        022  03  3        058  16  A'       094  88  88       130  05  5        166  05  5        202  00  00
001  01  1        023  05  5        059  42  STO      095  01  1        131  06  6        167  71  SBR      203  00  00
002  02  2        024  06  6        060  88  88       096  04  4        132  05  5        168  00  00       204  62  PG*
003  22  INV      025  03  3        061  01  1        097  06  6        133  71  SBR      169  00  00       205  89  89
004  28  LOG      026  03  3        062  03  3        098  05  5        134  00  00       170  62  PG*      206  10  E'
005  85  +        027  06  6        063  06  6        099  71  SBR      135  00  00       171  89  89       207  61  GTO
006  01  1        028  71  SBR      064  05  5        100  00  00       136  62  PG*      172  19  D'       208  00  00
007  54  )        029  00  00       065  71  SBR      101  00  00       137  89  89       173  61  GTO      209  15  15
008  82  HIR      030  00  00       066  00  00       102  62  PG*      138  18  C'       174  00  00       210  76  LBL
009  08  08       031  62  PG*      067  00  00       103  89  89       139  61  GTO      175  15  15       211  36  PGM
010  43  RCL      032  89  89       068  62  PG*      104  17  B'       140  00  00       176  76  LBL      212  32  X:T
011  88  88       033  91  R/S      069  89  89       105  61  GTO      141  15  15       177  15  E        213  09  9
012  69  OP       034  99  PRT      070  16  A'       106  00  00       142  76  LBL      178  42  STO      214  69  OP
013  06  06       035  98  ADV      071  61  GTO      107  15  15       143  14  D        179  88  88       215  17  17
014  92  RTN      036  91  R/S      072  00  00       108  76  LBL      144  42  STO      180  01  1        216  32  X:T
015  99  PRT      037  61  GTO      073  15  15       109  13  C        145  88  88       181  07  7        217  42  STO
016  98  ADV      038  00  00       074  76  LBL      110  42  STO      146  01  1        182  00  0        218  89  89
017  91  R/S      039  18  18       075  12  B        111  88  88       147  06  6        183  00  0        219  36  PGM
018  42  STO      040  76  LBL      076  42  STO      112  01  1        148  00  0        184  71  SBR      220  01  01
019  88  88       041  11  A        077  88  88       113  05  5        149  00  0        185  00  00       221  71  SBR
020  06  6        042  42  STO      078  01  1        114  00  0        150  71  SBR      186  00  00       222  96  WRT
021  00  0        043  88  88       079  04  4        115  00  0        151  00  00       187  62  PG*      223  43  RCL
                  044  01  1        080  00  0        116  71  SBR      152  00  00       188  89  89       224  89  89
                  045  03  3        081  00  0        117  00  00       153  62  PG*      189  15  E        225  42  STO
                  046  00  0        092  71  SBR      118  00  00       154  89  89       190  61  GTO      226  88  88
                  047  00  0        083  00  00       119  62  PG*      155  14  D        191  00  00       227  03  3
                  048  71  SBR      084  00  00       120  89  89       156  61  GTO      192  15  15       228  03  3
                  049  00  00       085  62  PG*      121  13  C        157  00  00       193  76  LBL      229  02  2
                  050  00  00       086  89  89       122  61  GTO      158  15  15       194  10  E'       230  02  2
                  051  62  PG*      087  12  B        123  00  00       159  76  LBL      195  42  STO      231  03  3
                  052  89  89       088  61  GTO      124  15  15       160  19  D'       196  88  88       232  00  0
                  053  11  A        089  00  00       125  76  LBL      161  42  STO      197  01  1        233  71  SBR
                  054  61  GTO      090  15  15       126  18  C'       162  88  88       198  07  7        234  00  00
                  055  00  00       091  76  LBL      127  42  STO      163  01  1        199  06  6        235  00  00
                  056  15  15       092  17  B'       128  88  88       164  06  6        200  05  5        236  98  ADV
                  057  76  LBL      093  42  STO      129  01  1        165  06  6        201  71  SBR      237  91  R/S
```

DIAGNOSTIC. - (Re-v5n2p9) Marion Brown has found that:

1. You need to press CLR or enter a zero after reading card side with 4 +/-, if you want the test to flash zero. Otherwise it will flash -4.

2. If somehow a code 31 (LRN) gets into the program, the counter will barge right ahead and stop at 959. In other words, it fails to pick up an (erroneous) code 31. It will pick up any other code, though.

CROSS COMPILER BASIC TO TI-59.- Systems, 810 Stratford, Sidney, OH 45365 offers a program in Basic to "cross compile Basic to TI-59." The Basic listing costs $12.00, the documentation an additional $ 8.00, while a TRS-80 Level II casette with the listing costs $ 15.00.

See you next time, Maurice