

# THE NATIONAL NINETY-NINER

VOL III - NO. 1 - JANUARY, 1986

COPYRIGHT 1986 BY

THE 99ER'S ASSOCIATION  
3535 SO.H ST., #26  
BAKERSFIELD, CALIF. 93304  
(805) 397-4361

**\$1.50**

DON VEITH - EDITOR  
COMPUSERVE ID #: 72257,3671

CREATED FOR TI 99/4A HOME COMPUTER OWNERS

## TABLE OF CONTENTS

<u>SECTION/ARTICLE</u>	<u>AUTHOR</u>	<u>PAGE</u>
ANNOUNCEMENTS		1
BACK ISSUES - VOL. I (1983-1984)		1
SPECIAL SUBSCRIPTION EXTENDED		1
REVIEWS		1
IRS TAX GUIDE	DARRELL INGOLD	1
ARTICLES		2
THE CRAFT OF PROGRAMMING * HIGH TECH FILE *	CARNEY MIMMS	2 3
MULTIPLAN ARTICLES		5
TIPS FOR USERS OF MULTIPLAN	DON VEITH	5
MULTIPLAN	L. R. THOMPSON, SR.	5
MULTIPLAN	CURTIS PROVANCE	6
PRINTING WITH MULTIPLAN	HOMER CRABTREE	6
CHANGING MULTIPLAN'S DATA DISK/PRTR OUTPUT		7
TI-MULTIPLAN TIPS	B. J. MATHIS	7
WORD PROCESSING WITH MULTIPLAN	PETE PHILLIPS	8
TIMP TIPS AND TECHNIQUES	STEVE ZIMMERMAN	9
MULTIPLAN A BOOK REVIEW	TED WHOMSLEY	10
TI-WRITER ARTICLES		11
TK-WRITER/QSWRITER AND THE SD COMMAND		11
TK-WRITER REVISION		11
DON'T HESITATE TO TRANSLITERATE	ABDULLAH CLARK	11
TI-WRITER: USING THE SPECIAL CHARACTER MODE	COMPUSERVE	12

VOL III - NO. 1 - JANUARY, 1986

COPYRIGHT 1986 BY

THE 99ER'S ASSOCIATION  
3535 SO. H ST., #26  
BAKERSFIELD, CALIF. 93304  
(805) 397-4361

DON VEITH - EDITOR  
COMPUSERVE ID #: 72257,3671

CREATED FOR TI 99/4A HOME COMPUTER OWNERS

## ANNOUNCEMENTS

BACK ISSUES - VOL. I (1983-1984)

By Don Veith

An apology is in order to all of you who forwarded funds in good faith for Volume I of the Back Issues. A bit of background on what has transpired is perhaps appropriate at this time. A falling out, of sorts, occurred with the previous printer of our newsletter. Details are not important at this late date except to say our problems began early in 1985. The original printer kept repeatedly assuring me that the Back Issues were "almost ready" for mailing to the subscribers. To the best of our knowledge, they were never printed and do not exist. We were left with a dilemma that would test anyone's patience and ingenuity. We have decided to produce the Back Issues rather than refund money forwarded in good faith. The late issues and the much delayed Back Issues are my fault and I accept that responsibility. I should have detected the problem much sooner and taken corrective action to solve it earlier. The problems are now under control with plans to mail the newsletter from Bakersfield by the 10th of each month.

Back to the subject we started with, the Back Issues. We have replaced our previous printer with a local printer here in Bakersfield. He has been doing the newsletter since the May, 1985 issue. We had to double up the last two issues due the printer's extended vacation during October and November, 1985. The October and November-December, 1985 issues were mailed late on December 20, 1985 due to an extensive backlog after the printer returned from vacation. The local printer will start printing the Back Issues during January, 1986. The printing process should be completed by March 1, 1986. We will be able to mail copies of this much promised document about March 10, 1986. The Back Issues will be 50 double-sided pages printed exactly like the newsletter. An index of articles will be included for your convenience.

Luci, myself, and the complete staff of THE NATIONAL NINETY-NINER hope each of you enjoyed your best holiday season ever. We plan to continue publishing this newsletter. Articles on the 4A will continue to appear and be available while our volunteer writers are willing to give their time to create them. May 1986 be the best year any of us have experienced. Thank you for your patience and support!

## SPECIAL SUBSCRIPTION EXTENDED

We have decided to extend the time period for acceptance of the special \$9.00 subscription offer. The offer was mailed to each Users Group with the DataBioTics National Advisory Panel letter. The offer has been extended to a new expiration date of February 28, 1986. All conditions outlined in the letter still apply. If you are not aware of the offer, contact your Users Group officers or write to us for details at our address on the top of this page.

## REVIEWS

### IRS TAX GUIDE

By Darrell Ingold

This program by Frederick Rose is designed to aid those of us who do our own taxes. WHAT IT IS: 1) a line by line input with prompts, 2) a calculator for most tax computations, 3) a print-out (screen or hardcopy) of each line entry on form 1040 and listed supplements. WHAT IT IS NOT: 1) a tutorial on how to do your own taxes, 2) a complete program for printing out completed tax forms, 3) a fool-proof way of doing taxes. System requirements: Basic version - a disk drive is the only additional peripheral; Extended Basic version - 32K memory expansion, disk drive, Extended Basic module and a printer if you wish a hardcopy print-out.

Now, with these facts firmly in mind, let's examine how the IRS TAX GUIDE works. It is imperative that you have a current copy of each form you intend to use PLUS all the pertinent information that will be required for each form before you begin. The program uses line numbers ONLY !!! You must refer to the forms to know what input is required. Don't forget that income averaging requires the previous several year's tax information. The program does compute the following items: Form 1040, Schedules A, B, G & W; worksheets for Social Security/Railroad Retirement, pensions/annuities, unemployment, IRA, state/local tax, contributions, earned income credit & excess FICA.

When using the program (which I did run several times on my own 1984 tax information), I noted several items. First, be advised that if you enter the wrong information into the lines, the wrong results will be obtained. This is just a friendly way to say, double check your input. When entering data always enter the amount as a whole number without commas, dollar signs or cents; e.g. enter 32568 for the amount of \$32,567.86. This method of rounding off is commonly used by accountants and is perfectly acceptable to the IRS.

Next, pay attention to the instructions: sometimes you will be asked if there is an entry for a certain line number. The response requested is displayed on the screen (Y/N). The very next line may simply ask for an entry. I found this to be a bit confusing at times but necessary because the lines that require additional calculation such as excess FICA, need to go to a subroutine only if you have an entry for that line. Other lines are simply a numerical entry from your records. It is not a difficult thing to handle but you must pay attention to the directions! One more thing, if you have no entry for a line that asks for an entry...enter 0. Any other entry (such as just hitting the enter key) will give an error message (but will not stop the program).

Everytime that I ran the program and entered the correct information, the final results were absolutely accurate based on my tax data for 1984. I have been using Microsoft Multiplan for my month to month record keeping. By just taking the totals from it and entering them into this program, I had my taxes done in short order. This guide is very handy for the calculations but it is still imperative that you know what you are doing yourself! One outstanding feature of the IRS TAX GUIDE is that it has the tax table calculations built into the program. I did not have to refer to the tax table to obtain my amount of tax...it just appeared at the right time!

IRS TAX GUIDE is available from Frederick Rose, P.O. Box 510148, Melbourne Beach, Florida 32951-0148. Cost is \$22.50.

## ARTICLES

### THE CRAFT OF PROGRAMMING

By Carney Mims

It must be part of our nature that we never are fully comfortable with a new activity until we have compared it to some old familiar one we understand. No doubt our ancestors, who first took up the newfangled practice of shooting game with bows and arrows, must have told one another that, "When you think about it, it's not so different from throwing rocks."

Lately computer programming seems to have been going through the process of being made familiar, as writers have compared it to a bewildering array of activities ranging from doing crossword puzzles to composing music. More often than not, the comparisons and analogies derived from everyone's favorite metaphor, that of computer programming as a form of language.

Before we get too comfortable with this comparison, we might take a closer look at the similarities between computer programming and the traditional crafts. (An idea proposed and explored with great insight by Dave Bolter in his fine book TURINGS MAN). All I will do here is add a few thoughts of mine along the way.

The metaphor of computer programming as a form of language has now become so pervasive that we tend to forget both its obvious implications and its limitations. The main implication is that when programming, we are communicating with another someone in his or her own language. The less glamorous reality asserts itself when we realize that WE are doing all the communicating. What is coming back is only the product of our instructions to the machine. For further reminder of this reality, we need only look at the struggles of artificial intelligence researchers as they try to program their computers to understand and respond to human communication, or as they call it "natural" language.

Club members who have tried our TI version of the famous ELIZA computer psychiatrist program, have seen a dramatic demonstration of both the power of and the illusion of communicating with the machine, and its underlying limitations. One has to only listen to dedicated programmers, professional or amateur, to see that their talk has less to do with some mysterious interaction with the machine than it does with a pride in their craft.

This attitude makes more sense when seen in its historical context: Each new technology produces new crafts. And so just as early advances in metalworking and the control of fire produced woodworking, or as the advent of fiberglass produced modern boatbuilding, so the technology of the computer has produced the craft of programming.

This correspondence between programming and earlier crafts is often obscured by computer programming's roots in the engineering disciplines, some of whose practitioners, calling themselves "software engineers" still treat programming as if it were merely an engineering technique, - like assembling the parts of a machine. While this may have been true in the days when programming consisted of writing short but stupefying detailed sets of machine language instructions for the early mainframe computers, even a short acquaintance with a modern computer language, such as TI Basic or Extended basic illustrate this is no longer true.

The most vivid demonstration of the significance of individual programming style and technique lies in the fact, so often remarked by our program director, that given the same programming problem to solve, several programmers arrive at completely different yet equally valid solutions. Many experienced programmers, and just about all fanatic hackers, will go much further and claim that they can identify the author of a program if they are familiar with his/her style. In their minds, a particular programmer's decision to use, for example, a GOTO-based loop rather than FOR-NEXT statements in certain leaves as distinctive a mark on the work as a woodworker's choice of a chisel or saw. It is this element of personal style, and the tools necessary to express it, which lift computer programming out of the realm of cookbook application of rules and into that of a skilled craft.

It is programming as craftsmanship that, I think, best explains its attractions for amateur programmers like ourselves. The satisfactions of programming are akin to those of working by hand in wood: the feeling of mastery and self-expression. The other inducements often offered for learning to program: a better understanding of computers, write your own business software, etc., may be beside the point. Best of all, unlike woodworking or other traditional crafts, you do not have to master or reinvent ten thousand years of accumulated wisdom.

So, have at it! And remember, don't leave any marks on the parts that show!!

Article copied from the September, 1985 issue of the CALL SOUNDS newsletter produced by The Central Westchester 99'ers of Hawthorne, New York.

\$ HI TECH FILE \$

GETTING A LINE ON YOUR PROGRAM

Have you ever wondered how your TI console stores and reads the BASIC program you type in? Well, I have! It all started when I began to have problems with one of my consoles. Every time I EDITed an existing line, other lines got "messed up" or even MOVED to some other part of the program! Eventually, I had to have the console replaced, but it didn't keep me from becoming curious. I was determined to figure out how this could happen. When I finally found the answer, I learned a great deal about how the TI keeps track of the program we type into it. Below is some info that you can use to get "computer's-eye-view" of the program you have typed in.

IT TAKES TWO TO TANGO

Any BASIC operating system has to keep track of every line of code you type in. It not only has to keep track of the program data, but it also has to keep track of the line numbers, too! The TI BASIC system uses two different areas for this: 1) the Line Number Table and 2) the Program Area.

All this info is stored in the highest memory available to the console. For TI BASIC that will usually be somewhere around address 14228 and for X-BASIC with MEMORY EXPANSION, somewhere around address -28. This all depends on the number of files open (as in CALL FILES(1)) and whether the disk drives are attached, etc.

PROGRAM FIRST

The actual lines of program code are stored first. Each line of code is stored in a series of bytes. Most all reserved words like PRINT, INPUT, REM, etc., are stored as a single byte value called a TOKEN. For instance, the TOKEN values for the the above three words are 156(>9C), 146(>92) and 154(>9A). TI BASIC also keeps track of variables, constants and jump references in a similar way. Each line is preceded by a length byte and, at the end of each line, the value 0 (zero) is placed to indicate the end of the line.

As each line is typed in, the BASIC operating system translates it into these byte values and stores it at the top end of the available memory. When you EDIT a line or INSERT a new line between existing ones, that new line is stored at the "end" of the Program Area, not inserted into the middle. This is important to remember when trying to locate a line of code! Note also that the line numbers (i.e. 100,110,120,etc.) are NOT stored with the line of code. This allows the TI system to easily RESequence your program, and allows the system to place lines at any available memory location instead of in execution order only. So where are the line numbers? In the Line Number Table, of course!

THE LINE NUMBER TABLE

After all the lines of code are stored, the Line Number Table begins. Each line of code has 4(four) bytes of information in the Line Number Table; two bytes for the line number (>0064=line 100), and two bytes for the memory address where the line of code is stored (>FFE7=-28). When the BASIC interpreter is running the program it looks at the Line Number Table, gets the line number and the line address, branches to the address, gets the actual program line and THEN performs the line of code! Not so simple, eh?

Since the Line Number Table is stored at the end of all the program lines, each time you add a new program line, the entire table has to be "shifted" downward into lower memory. This is why, especially in a long TI BASIC program, it takes quite a while for the cursor to "come back" after you have EDITed a line. The operating system must code the line, shift all the line numbers and addresses downward and then insert the new line number and address into the table. That would take anybody a little time!

Let's look in on the operating system as it does all this stuff! Here's what we'll do:

1) Enter X-BASIC (or TI BASIC with ED/A or MINIMEM module) and type:

```
100 PRINT "THIS IS A TEST"
```

2) Find Line Number Table: To find the Line Number Table, PEEK into the CPU RAM PAD at address -31950; get two consecutive bytes; convert these bytes to decimal; subtract 3 bytes (overhead) and, if your are using X-BASIC with MEMORY EXPANSION, subtract 65536; you now know the address where the line table starts!

```
CALL PEEK(-31950,A1,A2)
LT=A1*256+A2-3-65536
```

In X-BASIC, the value should be -47 and in TI BASIC it should be 14273. As each line of code is typed in, the table will move farther down into memory.

3) Get the line and address: Using the start of the Line Number Table, get the first line of code and the address where it is stored. In X-BASIC this is stored in MEMORY EXPANSION so you use the following:

```
CALL PEEK(LT,L1,L2,A1,A2)    LN=L1*256+L2    LA=A1*256+A2-65536

LT=Line Table Address ( -47)
LN=Line Number      ( 100)
LA=Line Address     ( -42)
```

In TI BASIC, the program is stored in the console itself, therefore you must look into the VDP area with the command PEEKV:

```
CALL PEEKV(LT,L1,L2,A1,A2)
LN=L1*256+L2
LA=A1*256+A2

LT=Line Table Address (14273)
LN=Line Number      ( 100)
LA=Line Address     (14278)
```

4) Get the line length:

Each line of program code is PRECEDED by a length byte. This tells the system how many bytes to interpret. To find the line length, read the byte PRECEDING the Line Address:

```
X-BASIC: CALL PEEK(LA-1,LL)
TI-BASIC : CALL PEEKV(LA-1,LL)
```

In both cases, LL=18.

5) Read the Program Line: Now all we have to do is read in 18 bytes of code starting at the Line Address (LA):

```
X-BASIC:
CALL PEEK(LA,A,B,C,D,E,F,G,H,
I,J,K,L,M,N,O,P,Q,R)
PRINT A;B;C;D;E;F;G;H;I;J;K;
L;M;N;O;P;Q;R

TI-BASIC:
CALL PEEKV(LA,A,B,C,D,E,F,G,H,
I,J,K,L,M,N,O,P,Q,R)
PRINT A;B;C;D;E;F;G;H;I;J;K;
L;M;N;O;P;Q;R
```

In both cases, the values A-R will equal:

```
156 199 14 84 72 73 83 32 73 83
32 65 32 84 69 83 84 0
```

```
156 - Token value for PRINT
199 - indicates a string value
14 - length of string
```

The rest of the bytes are ASCII values for the string (84=I, 72=H, 73=I, 83=S, etc.).

Now let's see if we can't reverse the process. This time we'll do a few POKES into memory that will make the computer think you typed in a line of code! Here's what we'll do:

1) Clear memory and type CALL INIT.

2) Set up Line Table: The system needs to know where the line table starts, so:

```
CALL LOAD(-31950,A1,A2)
X-BASIC: A1=255, A2=220
TI-BASIC A1= 55, A2=204
```

3) Insert line # and address: Next we need to fill up the line table with line and address data:

```
X-BASIC: CALL LOAD(-39,0,100,255,222)
TI BASIC: CALL POKEV(14281,0,100,55,206)
```

4) Insert line length and line code. Now all we need to do is insert the line data:

```
X-BASIC:
CALL LOAD(-35,10,156,199,6,
72,69,76,76,79,33,0)

TI BASIC:
CALL PEEKV(14285,10,156,199,6,
72,69,76,76,79,33,0)
```

5) LIST YOUR PROGRAM!

Because the system has not been properly informed as to where the program ENDS, this is not "RUNable" code. But at least you can see your work!

Below is a program that will "read itself." Using the tools we learned above, this routine will read the Line Number Table, and find the Line Address and Line length. After each line is found, the program waits for you to hit a key before it reads the next line.

NOTE: This program is written for TI BASIC with ED/A or MINIMEM modules. To use it with the X-BASIC and MEMORY EXPANSION you need to make a few changes:

- 1) Line 220 - add -65536 to the end of the line.
- 2) Line 260 - change PEEKV to PEEK.
- 3) Line 290 - add -65536 to the end of the line.
- 4) Line 330 - change PEEKV to PEEK.

Armed with this info, you can begin to learn more about how the TI system works and you can use these tools to help read AND write your own programs. A number of programmers place routines like this into their programs as a way of "protecting" valuable code sequences or algorithms from prying eyes. It's usually called "imbedded code." You can also use routines like this to translate a text line into program format once you learn to decode all the token values!

Next time, we'll talk about how to write a routine that writes it's own RUNable program!

```

100 REM *****
110 REM *
120 REM * PROGRAM PEEKER *
130 REM *
140 REM *****
150 REM
160 REM      11/84
170 REM      SUBFILE99
180 REM
190 CALL INIT
200 CALL CLEAR
210 REM
220 REM *FIND LINE TABLE*
230 REM
240 CALL PEEK(-31950,A1,A2)
250 LT=A1*256+A2-3
260 REM
270 REM *GET LINE & ADR*
280 REM
290 CALL PEEKV(LT,L1,L2,A1,A
2)
300 LN=L1*256+L2
310 IF LN=0 THEN 570
320 LA=A1*256+A2
330 REM
340 REM *GET LINE LENGTH*
350 REM
360 CALL PEEKV(LA-1,LL)
370 REM
380 REM *****
390 REM *PRINT THE DATA*
400 REM *****
410 REM
420 PRINT "LINE";LN
430 PRINT "ADDR";LA
440 PRINT "LEN ";LL
450 PRINT
460 CALL SOUND(150,1400,0)
470 CALL KEY(0,K,S)
480 IF S=0 THEN 470
490 REM
500 REM *UPDATE LT*
510 REM
520 LT=LT-4
530 GOTO 290
540 REM
550 REM *END*
560 REM
570 PRINT "END OF PROGRAM"
580 PRINT
590 END

```

## MULTIPLAN ARTICLES

### TIPS FOR USERS OF MULTIPLAN

This section contains articles from various newsletters to provide tips on how to increase your use of this powerful software package. Many of you have already completed setting up your checkbook and household budget in Multiplan. Some other uses I have observed from reading various newsletters are keeping of statistics for any sports activity, inventory of your household items, and statistics for a college class.

Multiplan does have a sort feature for any column you designate. It will sort either numeric, alphabetic, or alphanumeric information. Design your data input into the spreadsheet carefully. Each item of data you wish to use as a sort field must be given its own column in the spreadsheet. A file containing names, addresses, telephone numbers, etc. would probably be sorted on these major categories. You would need a column for the person's last name while combining the first name and middle initial into another column. Plan every possible way you might wish to extract or sort data from the file. Do this exercise on paper to work out the problems before working with Multiplan. I have found that it best to think out a file plan over several days before beginning data entry. Several new approaches to the problem will "pop into your head" at the oddest times and places. I hope you enjoy the articles and find the information both useful and interesting. Don Veith, Editor

### MULTIPLAN

By Larry R. Thompson, Sr. - Chester County 99'ers

A lot of people are of the opinion that Multiplan is strictly a program used for financial analysis, probably because it is used to such a great extent in this regard. However, there are any number of uses which Multiplan or any spreadsheet can perform.

Multiplan may be used anyplace where numbers are mathematically related to one another, and it is necessary to calculate a result based on changing one or more elements relating to the output. One possible example could be someone who enjoys cooking but has to constantly calculate how much of each ingredient to use based on the number being served. It would be a simple matter to set up a sheet where all you had to do was enter the number of servings and the sheet would automatically calculate how much of each ingredient to use. Once this sheet was laid out it would be a simple matter to change it for any recipes you may have.

Another possible use would be someone who does custom carpentry, for example windows. Once the rough dimensions of a window were entered onto the sheet, the sheet could be set up to calculate the size of all pieces needed to build the window.

A third and last example would be for the electronic tinkerer. The gain of an operational amplifier circuitry or the frequency of timers is dependent on other circuit values. It would be a simple matter to lay out a sheet to allow you to change any one value and determine the effect on the output.

The main point is that Multiplan is a very flexible tool that almost anyone can adapt to their individual purpose. Oh, it also does financial work, like keeping and balancing your checkbook.

### MULTIPLAN

By Curtis Provance - New Hampshire 99'ers UG

I prepare several reports each month for my church and have come up with some time savers that pertain to Income Statements, Balance Sheets, etc. The files have several hundred formulas and cells, so shortcuts had to be found quickly.

1. Load formulas first, copy where necessary. When you copy a cell, you get cell formats plus its contents. Format after you have copied the cells.
2. Find the LOWER RIGHT; delete unneeded rows and columns and increase your percentage (%) of memory available.
3. A single cell can be recalculated without doing the complete spreadsheet. Place the cursor over the cell to be recalculated, then depress EDIT followed by the ENTER key.
4. If you have large strings. use continuous columns instead of FORMATTing a large width for a single column. If you use a single large wide column, you may not be able to view that column with any others. With several small columns, you can SPLIT vertically keeping a portion of your strings visible.
5. I have tried to save some disk space by NAMEing the entire sheet, then EXTERNAL copying it onto a blank sheet. What this did was remove all of the formulas, formats, etc. I reformatted and saved the new sheet. The results? Not impressive. I went from a 56 sector file to 53 sector file, and the copying task took almost one half hour. My advice, don't waste your time to save space this way. One thing you can do is generate a look up table this way. I had three (3) years of historical data that I wanted averaged for each month and displayed as individual months with a cumulative total. Once the data was computed, I did not need the formulas, only the results. I did NAME the results and copy back onto a blank sheet. When you copy a column, row or whatever from another sheet, you only get numbers and strings; the format remains the same as whatever you are copying into. The manual states that you can only copy into BLANK cells, and this is true. You can, however, format blank cells and they will still be BLANK.

Copied from The New Hampshire 99'ers newsletter of July, 1985.

### PRINTING WITH MULTIPLAN

By Homer Crabtree

When I give my opinion of Microsoft Multiplan used with the 99/4A, I state that it is as good as it is with other computer except you do not have printer control. For an example, you cannot have an enlarged header with emphasized titles. Well, things have changed thanks to other users of our computer.

The trick is accomplished by creating a spreadsheet, saving it as a SYLK file, and then use DISKO (or some other type of disk fixer software) to insert the printer into the SYLK file. Then by copying the command to other spreadsheets, you can change fonts, pitch, and other features your printer may possess.

The following steps illustrate how to create such a file:

1. With Multiplan, create a spreadsheet with "ABC" in cell R1C1.
2. Use the name command to name this cell "cond" for condensed.
3. Press T, then O, and select Symbolic.
4. Press T then to save to a newly initialized disk.
5. With DISKO, look in sector 22 for "ABC" or Hex >41 >42 >43 and change it to >1B >42 >33. (Hex=Hexadecimal)
6. Go to Multiplan and Transfer, Option (Symbolic), Transfer Load file, then T, O (normal), T, S to a new file name.

You are now ready to use the file.

1. Load any spreadsheet. (E.G. SPENCER)
2. In an empty cell press Xtern, Copy, enter the name of the printer control file, TAB (CTRL 2), enter cond (name of cell to copy).
3. Print your spreadsheet. It should be condensed.

The above works for a Gemini. The control codes vary with each printer and should be listed in the printer reference manual.

This is a list of printer control codes in my print control spreadsheets.

CC	ABBR	PRINTER FUNCTION	CC	ABBR	PRINTER FUNCTION
AS	FNTASC	ASCII FONT	DS	DBLSTK	DOUBLE-STRIKE
IT	FNTILT	ITALIC FONT	CD	CANDBLSTK	CANCEL DOUBLE STRIKE
PC	PITCOND	PITCH CONDENSED	EM	EMPH	EMPHASIZED
PE	PITELITE	PITCH ELITE	CE	CANEMPH	CANCEL EMPHASIZED
PN	PITNORM	PITCH NORMAL	BS	SUBSCP	SUBSCRIPT
PP	PITPICA	PITCH PICA	PS	SUPSCP	SUPER SCRIPT
BD	BIDIR	BI-DIRECTIONAL	CS	CANSUBSCP	CANCEL EITHER SCRIPT
UD	UNIDIR	UNI-DIRECTIONAL	UL	UNDLIN	UNDERLINE
EJ	EJECT	EJECT PAGE	CU	CANUNDLIN	CANCEL UNDERLINE
L6	LPI6	LINES PER INCH=6	M5	LMAR5	LEFT MARGIN 5
L8	LPI8	LINES PER INCH=8	M10	LMAR10	LEFT MARGIN 10
IN	INIT	INITIALIZE PRINTER	M16	LMAR15	LEFT MARGIN 15

#### CHANGING MULTIPLAN'S DATA DISK AND PRINTER OUTPUT

This article was taken from a **BITS AND PIECES** column in the Ozark 99'er News of October, 1985. The newsletter is published by the Ozark 99'er Users Group of Springfield, Missouri.

The default values for your printer and disk drive can be changed on the Multiplan Program Disk so that it is not necessary to set them at the beginning of each session. These can be changed using any "Disk Fixer" program. (Advanced Diagnostics was utilized by the author.) Looking at Sector 272 will reveal 'DSK1' in Byte 92. This is where your spreadsheet is loaded from when the Transfer Option is selected. If you have two disk drives, set this value to 'DSK2'.

On the same sector (272) at Byte 196, you will see 'RS-232.BA=300'. This is the device that your spreadsheet will be printed on when the print option is selected. You may want to change the baud rate or change the output to a parallel printer such as PIO.LF.

**EDITOR'S NOTE:** The procedure works exactly as outlined in the article. I used Advanced Diagnostics by Miller's Graphics to modify the sector and bits outlined in the article. The procedure is very easy and only takes a few minutes to complete. If you do not have any type of disk fixer or do not care to attempt the procedure, forward a formatted single-side, single-density disk to our address on page one (1). The only condition is that you must include the postage for its return to your address. If your printer uses an output other than PIO.LF or RS232.BA=300, please advise me and I will alter that output to meet your printer requirements.

#### TI-MULTIPLAN TIPS

By B. J. Mathis

The next three articles were written by the author listed above. The articles were copied from the April through June, 1985, newsletters of the SOUTHWEST NINETY-NINERS from Tucson, Arizona.

#### TITLES

Have you ever tried to place a title on your Multiplan worksheet? If so, you know how frustrating it can be. You type and enter one word at a time in each of the upper cells. If you later insert a new column or delete a column, or simply widen or narrow a column, you have to retype part if not all of your title. Well, here is something to put an end to that!

Simply choose the Formatting command. Then type R1 (or whichever you want your title in), then TAB (CTRL 2) to the alignment section. Select 6 (for general), TAB again to the code section, type C for continuous, and press Enter. Your title will now adjust itself to any amount of deletions amounts of deletions, insertions, narrowing or widening of a column.

If you discover a mistake in your title, place your cursor in the first cell of your title, and press E fo Edit. Multiplan will display the entire title for you to edit.

You probably already know that to back your cursor up, you have to use FCTN 9, however this also takes out whatever it passes over. You don't have to use this to edit. You can use FCTN 4 to backspace by single position or FCTN 5 to backspace and CTRL 4 or CTRL 5 to go forward. Multiplan will automatically allow you to insert your corrections once you have reached the proper point in the text. However, it will not get rid of letters unless you use the Delete forward CTRL 0 keys. Delete Forward deletes anything that is highlighted at the time it is chosen so make sure that you are highlighting the correct part. All these keys may be used to edit formulas and values, also.

## RECALC

Have you been trying to figure out how to use Multiplan? Well, this column is meant to help you solve some of the problems you're having. Before you do anything else, please be sure you have selected Options ("O") and changed the Recalc" to "NO". This will save you a lot of time and frustration. As Multiplan will even "Recalc" for text entered. When you have all of your figures and text typed in then reselect Options ("O") and turn on recalc, press "Y" for "YES" and press enter.

If some of your formulas give you an answer of "#REF!" then recheck your formulas to see if you are trying to add a cell that has text in it. Multiplan does not like to add text to numbers! Remember, to turn off the Recalc again before you try to fix those formulas. Once Recalc is off, use the Edit ("E") to look at each formula for possible problems. Use Back Char (FCTN 4), Forward Char (FCTN 5), Back Space (FCTN 9), and Delete Forward (FCTN 0) to change the formula as needed or completely rebuild the formula.

You may also see an error message of "#VALUE!", think about the formula. You may be trying to add, subtract, etc. your total from itself. If you insert a row or column within a formula's parameters, or move a row or column containing a formula, the formula may automatically change and calculate only part of the information you intended. A formula copied across may try to add each column to the next. This will result in an unbelievable total in the last column.

In order to correct these problems, you may have to check each formula on the entire worksheet. After checking each formula, select Options again and turn back on the Recalc. Having Recalc off when entering material will speed up operations significantly. So think about what you need to do and then decide whether Recalc should be on or off. Good luck!!

## THE COPY COMMAND

When using Multiplan, the copy command is probably one of the most useful commands. It is especially useful for setting up a spreadsheet. You can Copy Right, Down, or From. Each of these commands is useful for copying formulas or other information to another part of your worksheet.

Copy From lets you copy an entire rectangular area to another part of the worksheet. You can copy from one single cell to another single cell or you can copy from a group of cells to another group. In order to copy from one block of cells to another, you first designate the boundaries of the group you wish to copy. For instance, if you wish to copy the information in Row 3 Column 2 through Row 7 Column 5 to the area with the boundaries of row 24 column 9 through Row 28 Column 12, you first choose the Copy command, then chose From. Now type R3C2:R7C5, tab to the next section, type R24C9, press Enter. Notice it is not necessary to give the computer the boundaries of the area you wish to Copy the data/formulas to, you only have to tell it where to start copying to, it will figure out the rest.

If you want the information in one cell to appear in another group of cells, Copy From can accomplish that also. Let us say that you want the information on Row 5 Column 3 to appear in Row 6 Column 7 through Row 10 column 9. You will again choose the Copy From command. The command line should read "COPY FROM cells: R5C3 to:R6C7:R10C9".

Copy Right will copy cells to the right in the same row. If you want an item that is in Row 8 Column 10 to appear in Row 8 Columns 11 through 20, use the Copy Right command. To do this, it is best to position your cursor at Row 8 Column 10, press "C", press "R", enter your number of columns, in this case, 10 and press Enter. Copy Down is much the same as Copy Right, however it copies down the column instead of across the row.

When copying with RECALC turned off, the numeric value that was in the original cell will appear in the new position, however when you activate RECALC, the numeric value will change based on the information Multiplan finds in the cells referenced by the formulas. When you Copy a formula that contains references to a specific row or column, you may have to change that reference to relate to the specific row and column in it's new position. Copy can also be used to copy text, this will not change when RECALC is turned on.

## WORD PROCESSING WITH MULTIPLAN

By Pete Phillips - San Antonio 99ers

Word Processing with Multiplan?? Why not? Multiplan has many advantages over TI-Writer or the Editor/Assembler's Editor. For instance, Multiplan will allow you to format your document in columnar layout and print it in condensed text, providing for a larger amount of text on a given page. In addition, Multiplan will center your text where desired, and allow for the movement of blocks of text in a much more flexible format. Using Multiplan as a Word Processor does have it's drawbacks. Among these are the lack of a global editor, editing of text is a bit more difficult (you cannot simply type over your text), and fast typists will have to learn to slow down a little due to the program's relatively slow processing speed. Despite these drawbacks, however, for many applications Multiplan may be the easiest way to solve the problem at hand.

I do not propose to go into a full tutorial on the use of Multiplan, for that I would refer you to the Multiplan Manual. I realize that many people find this a formidable document, but for use as a text processor, only a general knowledge of Multiplan is necessary. Therefore, in this discussion I will merely cover what I have found to be the easiest steps to follow in setting up and using the worksheet.

Starting with an empty worksheet, your first step should be to select the OPT or OPTIONS command and turn off the Recalc or Recalculation Option. Since you will be doing no mathematical calculations, this will eliminate the considerable delay incurred as the program searches for numbers or formulas. Next select the FORMAT option, then DEFAULT (select the option the cursor is highlighting by pressing the Enter key) on the sub-menu, and finally WIDTH on the next menu, and set the default column width at 30 characters. I realize that it is possible to set the width up to 32 characters, but by setting it at 30 we will later be able to widen it to 32 creating a margin between each each column of text.

The next advisable setup step is to again select the FORMAT, DEFAULT option. This time select the CELLS option on the third menu. In the alignment column, select L for Left. Remember, when Multiplan is displaying the ALPHA/VALUE prompt, entering a number as the first character in a line will select the VALUE option rather than the ALPHA option. Therefore, if the first character in a line is a numeric one, you must first hit ENTER TWICE TO specifically select the ALPHA command. If you forget and the only characters entered on that line are numeric ones, this will prevent them from being right justified.

The final setup step I use is to select the WINDOW option and place a border around the one open window. You may then use this border as a line length guide while typing. You may type up to but not including the column containing the right border without having the end of your text cut off.

You are now ready to begin entering your text. Start at row one (1), column one (1), and enter one line after another in column one. I prefer to enter all of my text in column one and format it later, since this makes it somewhat easier to move data around. Another advantage is that you do not have to worry about keeping track of where you are located on the page.

Once the entering of text is completed, you are ready to format the data into columns. Since the maximum column width on the TI printer is 132, we will divide the text into four (4) equal columns of 32 characters each and have a two (2) column border on the left and right margins.

Assuming we are working with one page as an example, there are two (2) ways you can format the text. One would be to simply divide it into 54 rows per column (assuming your page length is 66), and leave whatever may be left over in the fourth column. You may also decide that you would like the columns to be even in length, in which case you would simply divide the total number of rows by four (4), and make each column that length.

For example, let us assume the total number of rows, when the document is formatted in one column is 200. 200 divided by 4 equals 50. We would therefore make each column 50 lines long. To do this, we would copy from row 51 to 100 and place the copy in Row 1, Column 2. Next we would copy from Row 101 to 150 and place the copy in Row 1, Column 3, and finally, we copy from Row 151 to 200 and place the copy in Row 1, Column 4.

The entire document is now in rows 1 through 50 and columns 1 through 4, but copies of columns 2 through 4 below row 50 in column 1. To get rid of these, use the delete command. Now, change the default width to provide 2 spaces between columns. You are now ready to print the file. To do this, first save the file to disk. Next, exit Multiplan and select TI BASIC, then enter the following commands:

```
OPEN #1:"PI0.CR" (Use double quotes)
PRINT #1:CHR$(15)
END
```

If your printer is not connected to the parallel port of your RS-232, you will have to supply the proper file-name. This procedure sets up the TI printer to print in condensed text. Next, re-enter Multiplan and select PRINT, OPTIONS. Enter your printer name in the setup field and return to the PRINT menu. Now, select MARGINS and set the left margin to two (2) and change the print width to 132.

All that needs to be done now is to select the printer command and your document should come out in four even columns. I admit this procedure is a bit tedious, but it is the most flexible means I know to format text into columnar form. I have made several attempts to devise a program to translate a TI-Writer file into a Multiplan file using the symbolic link file format, but so far all of my attempts have proved to be fruitless. I am still working on it, so if I have any success I will let you know.

This article was copied from the August, 1985 issue of the NINE T NINE USERS GROUP newsletter. The group is located in Toronto, Canada.

### TIP TIPS AND TECHNIQUES

By Steve Zimmerman

This time around I am going to talk about a variety of of small and not-so-small tips to help you avoid trouble with your spreadsheet models. First, if you choose to lock the formulas in a worksheet, make an unlocked backup copy BEFORE YOU LOCK THEM! Locking the formulas in a worksheet is easy. Unlocking them requires you to do it piece-by-piece! So, just in case you need to make changes in the future, keep your unlocked backup available! If someone else will be doing data entry on a complex worksheet, it is a good idea to have them working with a locked copy--this avoids problems such as having someone enter a number or label in a cell which contains formulas or information used elsewhere in the worksheet.

When building a worksheet, work on one area at a time. This allows you to enter numbers to check each small area of the worksheet does what you want it to do. Using this technique, you can build up each individual area so that it works, and then link the areas to produce subtotals, grand totals, and the like. This is like programming in FORTH--you define and test a word. Then use the word in further definitions once it has been tested.

Deleting your numbers can help you see if an error condition results. You can then correct the cause of any error condition in setting up your formulas. Use relative references whenever possible in building a worksheet (I seldom use absolute references for anything). The greatest advantage of relative references is that they allow you to easily copy formulas which will be used over and over. An absolute reference cannot be copied and used in another area of the worksheet without being edited. A time consuming process prone to errors.

#### Editor's note:

- (1) Relative Reference - A reference to a cell relative to the cell containing the reference, as R1-1C meaning "the row above, in this column." Alternative to an absolute reference.
- (2) Absolute Reference - A reference to a cell that uses specific row and column numbers -- for instance, R17C12.

I use relative references for adding up columns of numbers, for example, to obtain a daily total. Adding up R1-1+R1-3C+R1-5C adds the number one row up, three rows up, and five rows up, in THIS column (and so, may be copied to any column and do the same thing!), and places the totals here. A formula of R13C2+R1C2+R9C2 will add the values in those three cells and place the total in whatever cell the formula is located in. If copied from column 2 to column 3, the total will STILL add up the numbers IN COLUMN 2 -- NOT IN COLUMN 3! To get it to add up the numbers in column 3, 4, 5, or whatever it is copied to, it must be EDITed to correct the column references after it is copied. This is, to put it mildly, a pain!

The most accurate way to enter formulas in a spreadsheet is by pointing. Some spreadsheets do not allow this, but we are fortunate because Multiplan does. In fact, in Multiplan, we can point not only in entering but also in editing formulas. This is sometimes referred to as the "wander mode". Using this method of building a formula is simple and provides the advantage of creating relative references.

To use this method, begin by placing the cursor or cell pointer in the cell where you want the formula to be placed and the end result will appear. Key either V or = to start entering a formula. Use the <FCTN> E, S, D, X (the arrow keys) to move the cursor to the first cell you want to add subtract, or whatever. As you progress, a relative reference will appear on the command line after the word VALUE: -- if you move the cell pointer up one cell, it will show VALUE: R1-1C. If this is the first cell you want to use, press the key for the operator (+, -, \*, /) required (in this case, let us use +). The + will appear after the cell reference on the command line and the cell pointer will dropback to the cell you are building the formula in. To select the cell you want to add to the one just above, move the cell pointer once more using the arrow keys. As you move it, you will see the relative reference on the command line change. When you have found the next cell, again enter the next operator continuing in this manner until the formula is complete. When you have pointed to the last cell to be used in the formula, press <ENTER>. The cell pointer will drop back to the cell you are working in, and the number created by your formula will appear in that cell. In the lower left-hand corner of your screen, you will see the coordinates of the cell you are in, and the first 16 positions of the formula just created (or the entire formula if it is less than 16 positions long).

Using this method, you can easily enter long formulas without trying to remember cell coordinates and without taking the time and memory overhead involved in naming cells. Formulas that operate in one columnar row can be copied across or down and will work properly in any row or column.

When building a model this way, enter your row and column labels first, then enter data (sample data if possible), and last, point to each item to create your formula. Once you have the formula set up, blank the cells with your sample data items, let the sheet recalculate (or tell it to recalculate if you turn off the automatic recalc as I always do). Check for errors, such as #DIV/0! (meaning that you are trying to divide by 0). Save your backup copy when your formulas are completed, then lock in your working copy if you wish, and enter your data. This will help you create error-free models.

Editor's Note: ERROR MESSAGES are described in the Multiplan Help file on disk by selecting the Command FORMULAS.

Article was copied from the Riverside (Calif.) Users Group (TIRUG) newsletter of August, 1985.

### MULTIPLAN - A BOOK REVIEW

By Ted Whosley - BUG News - Brea, Ca - July, 1985

If you have been using the book, "Managing Your Business With Multiplan" by Ruth K. Witkin, this information may be of use to you. The book is written for the IBM computer and uses call-outs for the F1 (Function Key One), F2 etc., keys, plus a few words that are not used on the TI-99/4A.

A list of IBM keys and their equivalents on the 4A are listed below. If you will make a copy of this list and tape it to the inside front cover of the book, it will serve as a handy reference when working with Multiplan.

IBM	99/4A	MEANING	IBM	99/4A	MEANING
F1	CTRL-6	NEXT WINDOW	F6		NO REFERENCE
F2	CTRL-3	NEXT UNLOCKED CELLS	F7	FCTN-5	EDIT WORD LEFT (BACK WORD)
F3	CTRL-7	ABSOLUTE REF REFERENCE	F8	CTRL-5	EDIT WORD RIGHT (FORWARD WORD)
F4	FCTN-8	RECALCULATE	F9	FCTN-4	EDIT CHARACTER LEFT (BACK SPACE)
F5		NO REFERENCE	F10	CTRL-4	EDIT CHARACTER RIGHT (FORWARD CHARACTER)

The IBM uses REPEAT to repeat a character such as dashed lines. On the 4A, you must use COPY to do this. When a TAB is required on the IBM, you use CTRL-2 or CTRL-A. I find CTRL-A is handier because the keys are closer together. IBM CTRL-Pq Up is CTRL-1 or HOME on the 4A. IBM END is CTRL-Z or FCTN-1 (Lower Right). Help on the IBM is ALT H and simply H (or the ?) on our 4A. Even though the book is written for the IBM, I think it is the best text for a beginner. It goes into great detail on the information a beginner needs to know for getting started with Multiplan.

Editor's Note: I have reviewed this book and would recommend it highly to anyone who has no previous experience with spreadsheets as an excellent reference text.

## TI-WRITER ARTICLES

### TK-WRITER OR QS-WRITER AND THE SD COMMAND

The SD Command is used in TI-Writer for Showing the Directory of a disk. This feature is handy for locating files without going to a disk manager module or program. When both TK-Writer and QS-Writer were created, this command was left inoperative. The computer promptly tells you to do things that are unmentionable and returns independently to its master screen. As most of you are aware, TK-Writer and QS-Writer allow you to load TI-Writer without the TI-Writer Command Module using Extended Basic or another cartridge. A solution to disable the SD Command is given in the following paragraphs.

The software may be changed to make the computer ignore the SD Command. This is done by altering the existing software with a disk sector altering program such as Disk Fixer or Disko. The steps are outlined below:

1. Copy the file "EDIT1A" to a blank disk. This file will be altered with a disk fixer and be copied back to your original disk. (Be sure you have a backup copy before you do this).
2. Read Sector 24 with the disk fixer.
3. Alter the byte >14. It is >5344, change it to >2044.
4. Write the altered sector back to sector 24.
5. Copy this file back to your software using the same disk file name.
6. Boot up the modified copy and check for a "BEEP" error tone when you try to use the SD command.

Figures 1 and 2 below, using the Navarone disk fixer, provide an example of how the printout will appear. NOTE: Parenthesis, not normally in a sector display, are used to outline the address to be change from 5044 to 2044.

Figure 1 - BEFORE MODIFICATION

```
NAVARONE IND. *** DISK FIXER V2.0 ** SECTOR DUMP   SECTOR ADDRESS   0024
ADDR = 0 1 2 3 4 5 6 7 8 9 A B C D E F INTERPRETED
-----
0000 = 2D54 5348 3E0F 2D54 5425 3E2C 2FCA 4D20  -TSH>*-TRE>,JM
0010 = 3E84 2F42(5344)3CD4 2E4C 5346 0000 2E8A  >*/BSD<T.LSF***
```

Figure 2 - AFTER MODIFICATION

```
NAVARONE IND. *** DISK FIXER V2.0 ** SECTOR DUMP   SECTOR ADDRESS   0024
ADDR = 0 1 2 3 4 5 6 7 8 9 A B C D E F INTERPRETED
-----
0000 = 2D54 5348 3E0F 2D54 5425 3E2C 2FCA 4D20  -TSH>*-TRE>,JM
0010 = 3E84 2F42(2044)3CD4 2E4C 5346 0000 2E8A  >*/BSD<T.LSF***
```

### TK-WRITER REVISION

Apparently, when going from the Editor to the Formatter in Tom Knight's Extended Basic load program for TK-Writer, the LOAD program reloads the Assembly program and does not check to see if it is in memory. This wait or delay can be avoided by making the following modifications to the LOAD program:

```
100 CALL CLEAR :: CALL INIT           7,65,84,250,132,69,68,73,84,
:: CALL PEEK(-2043,A,B):: IF          79,82,250,22)
A<>84 OR B<>75 THEN 108                104 CALL LOAD(B196,63,232)::
102 CALL LOAD(16360,85,84,73           6010 110
,76,73,84,250,212,70,79,82,7         108 CALL LOAD("DSK1.WRITER")
```

### DON'T HESITATE TO TRANSLITERATE!

By Abdullah Clark

Many members of our User's Group and other TI-99/4A owners with TI-WRITER avoid using the Transliterate Command like it was a foreign language. As thick as the TI-WRITER Manual is, which intimidates some, this command is certainly one subject that deserves further explanation. These tips and applications are probably not the only points to be made about transliteration, and I hope others will come about as a result of this article. To keep it short, no examples are included, but contact me at the meeting if you have any questions.

1). Contrary to several published reports, the Transliterate Command does indeed work, and works well if each instance of its use is on a separate line (with a carriage return afterward). The most common reason seen for failure of this command is the lack of the leading period which is required for all format commands. Also, you must use commas, not spaces, when a multiple-character "conversion" is needed. One user even gets it to change printer configuration (instead of the SPECIAL CHARACTER MODE, which does not seem to work), but he uses the ampersand "&" between the multiple characters required.

2). Use the Transliterate Command to transliterate a character back to itself if you want to cancel that conversion later in your text.

- 3). Be aware that any time you use the Transliterate Command to cause one character to produce a set of symbols all at one time, if you are using the Fill and Indent Mode, the Text Formatter will treat those symbols as one character! As examples, consider the sequence given on page 107 of the TI-WRITER Manual to print a tilde over an "n" or the case of using one character to produce a set of ellipses marks (for a partial quotation).
- 4). The instance described in item #3 can be used to your advantage when you have run out of positions when composing a Header or Footer --- just use one "complex" Transliteration Command to specify the additional spaces or other characters needed.
- 5). The Transliterate Command can also be used to increase the spacing between a Header or Footer and the text and/or the end/start of the page, by using a line feed appropriately in the Header or footer, preceded by a proper Transliterate command. However, in this instance, do not use the underline or overstrike symbols in your text.
- 6). Another caution to keep in mind is not to convert any of the "reserved" characters used by TI-WRITER for its own special functions (the caret "^", the ampersand "&", or the at-sign "@") plus the underline. You will not get any error message, but your printout results will indeed be strange!
- 7). The suggestion by the TI-WRITER Manual at pages 106 and 127 to use two (2) at-signs or ampersands together when you want to print one of these symbols did not seem to work. Only transliterating did the trick.
- 8). I haven't tested this for the entire set of available characters, but it appears that the transliterate Command will work with any valid ASCII code as its first parameter (even ASCII 0 to 31), although the second parameter should be a character your printer is capable of printing/processing.
- 9). It's a good idea to use the Find string command to check your document for instances where you may have used a character in the text before you use that same character in transliteration.

Article copied from the TI Riberside Users Group (TIRUG) newsletter of september, 1985. It was copied for printing in their newsletter from HOCUS, the newsletter of MAUG (The Milwaukee Area Users Group).

### TI-WRITER: Using The Special Character Mode

The Special Character of the TI-WRITER software is a method of sending the necessary commands to a printer to activate the many functions, such a different character fonts. Those who are familiar with using the Text Formatter may already know of using the Transliterate Command to do just this. With Special Character Mode, the TL Commands are not used. Instead, a number of "Special" Characters, other than the normal ASCII range of 32 (space) to 127 (DEL), are generated and sent to the printer upon printout with either the PF command or the Text Formatter.

We have all seen the symbol that represents a Carriage Return installed at the end of a sentence or a paragraph in our document, this is one of those "Sp-Ch"'s. The ASCII value of a Carriage Return is 13, and send a CR to a printer you must send character number 13 (not the value 13).

To activate the special Character mode, you hit a CTRL-U, and the cursor symbol changes to an underline character. A second CTRL-U puts you back into a normal mode.

To install a Sp-Ch in your document, you must first know what function of your printer you wish to invoke. Most printer manuals have a chart that lists the various functions and the codes needed to activate them, and it is very handy to have a copy of this list nearby when formatting a document. For instance, the sequence ESCape-"E" (ASCII values 27 & 69) will invoke Emphasized Print on Epson Printers, and if we send an ASCII value of 27, then an ASCII value of 69, the printer switches to the Emphasized Print mode. We know that ASCII-69 is a capital E, but ASCII-27 (ESCape) is not a typeable character. Now we go to the list of Special character on page 146 of the TI-WRITER manual, and we see that ASCII-27 can be operated by typing the FCTN-R while in Sp Ch Mode. The symbols generated by each Sp Ch are also listed, and are almost all are Hexadecimal value of the ASCII code, compressed down to take up only one character. ASCII-27 is Hex-1B and you will see a little "1b". Right after the "ESC" character, you type the "E" for ASCII-69. In summary, the sequence to send a control codes for Emphasized print would be:

```
CTRL-U
(FCTN-R)E
CTRL-U
```

These Special Characters can be installed anywhere in the text, as they do not print upon output, just as the "re-defined" characters used with the Transliterate Command are "invisible"

There are pluses and minuses to using the special Character mode against using the Transliterate command. The TL commands are more versatile, and can be made to send a complicated sequence of ASCII values, where using Sp-Ch mode would get quite tedious each time a lengthy code was sent. Also, a number of TL command can be stored in a separate file, and linked to the document upon printing, thus saving the job of rewriting the commands each time. On the other hand, when a relatively short code sequence is needed, Sp-Ch is much simpler, and the biggest advantage is that you need not load and run the Text Formatter, which can be a major obstacle to many.

As an example, when I want to just write a little note, and I want it in Emphasized, I can simply start with a CTRL-U/FCTN-R/"E"/CTRL-U and when I use the PF command, I have a nice dark print. Another widely used area is when reformatting paragraphs, such as when modifying margins, and you need to install a Carriage-Return symbol at the end. One way is to move the cursor to the point where you need the symbol, hit CTRL-B (New Paragraph) and then edit out the extra line and spaces. A much simpler way is to just locate the cursor and hit CTRL-U/"M"/CTRL-U, which will generate a CR symbol.

Article downloaded from the "TI Forum" of COMPUERVE by Scott Darling, Sysop of BBS #4.

THAT'S ALL FOR THIS MONTH !!!

BULK RATE  
 U. S. POSTAGE  
**PAID**  
 Permit No. 787  
 Bakersfield, Calif.

THE NATIONAL 99'ER  
 3535 SO. H ST., #93  
 BAKERSFIELD, CA 93304

**SUBSCRIPTION FORM**  
**FOR "THE NATIONAL NINTY-NINER"**  
**PUBLISHED BY THE 99'ERS ASSOCIATION**

NAME: \_\_\_\_\_ DATE: \_\_\_\_\_

ADDRESS: \_\_\_\_\_

CITY: \_\_\_\_\_ STATE \_\_\_\_\_ ZIP \_\_\_\_\_

SUBSCRIPTION TYPE	AMOUNT	CHOICE
THIRD CLASS - BULK RATE	\$12.00	_____
FIRST CLASS - US AND CANADA	\$17.00	_____
FIRST CLASS - OVERSEAS	\$22.00	_____

PLEASE MAIL CK./M.O. FOR SUBSCRIPTION CHOICE SELECTED ABOVE TO:

THE 99'ERS ASSOCIATION  
 ATTN: LUCI VEITH  
 3535 SO. H ST., #93  
 BAKERSFIELD, CALIF. 93304