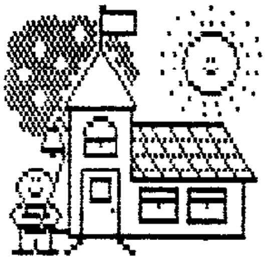
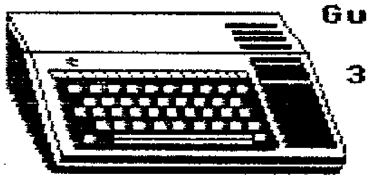
GUILFORD 99'ERS NEWSLETTER



SCHOOL DAYS

SUPPORTING THE TEXAS INSTRUMENTS TI-99/4A COMPUTER



GUILFORD 99'ERS UG
3202 CANTERBURY DR
GREENSBORO NC
27408

TO:

VOLUME 7 NUMBER 9 SEPTEMBER 1990

Bob Carmany, Pres. (855-1538) Emmett Hughes, Vice Pres. (584-5108)
Mack Jones, Sec/Treas (288-4280) Bill Woodruff,Pgm/Library (228-1892)
BBS: (919)621-2623 --ROS

The Guilford 99'er Users' Group Newsletter is free to dues paying members (One copy per family, please). Dues are $12.00 per family, per year. Send check to:LF Jones, 3202 Canterbury Dr., Greensboro, NC 27408. The Software Library is for dues paying members only. (George von Seth, Ed.: 292-2035)

OUR NEXT MEETING

DATE: Sept 4, 1990 Time: 7:30 PM. Place: Glenwood Recreation Center, 2010 S. Chapman Street.

Program for this meeting will be a demonstration of the new XB extension THE MISSING LINK. For a look at some outstanding graphic applications and some other "goodies", make sure that you come to the meeting!!

MINUTES

The August 7th meeting of the Guilford 99er Users' Group was held at the Glenwood Recreation Center in Greensboro. There were 8 members present.

The meeting was called to order by Pres. Bob Carmany at 7:45 P.M. Since the Sect. was absent for the July meeting, no minutes were read. The treasury report was given. For old business, Bob told members he has gotten the chips for his second Quest card and now has 1meg of memory space. As for new business Bob gave the price of the Quest cards now being sold by the land of OZ. The bare Quest board sells for $60. U.S. and you provide the sockets and chips. Or you may buy the completed board but it would be best to write to Hunter Valley for current prices as chip prices vary. He also mentioned that the Hunter Valley newsletter will be going bi-monthly. The 4.3 version of FW is now on ROS BBS for anyone wishing to download.

At the close of the business meeting, the program was turned over to Tony Kleen who gave a demo of his recent 3.01 version of TI Base with modifications. Good job Tony.

The meeting was adjourned at 9:00 P.M.

Respectfully submitted,
L.F."Mac" Jones, Sec/Treas.

RAMBLING BYTES

By "Mac"

First off I wish to apologize to any reader that I offended by inserting the "mouse" article on the last page of the August newsletter. I found that when the Editor delivered the master for me to have printed that we were one page shy of a full count. I had been looking over some old papers that I had from work and since I remembered the laughs we all got from this article, I thought it would be an amusing filler for the empty last page. I realize now it was in very poor taste for a newsletter and was in no way the fault of your Editor. Please forgive my bad taste and I promise not to try to help our competent Editor out in his job in the future.

That out of the way, I would like to say what a great job Tony is doing with TI Base. It's hard to believe how that lad has gone into this program! I guess he has just "hit" on the way to do it, that is, pick a program you like and can use and build it to your needs (if you have the smarts like Tony!!)

Well, I finally got that Gram Kracker that I could afford and now I have to read up on just how to use it. There is a lot to read about it in the manual so I have a lot to look forward to. Until the September meeting, enjoy the good TImes.

CALL PEEK(PRES)

After several weeks of waiting, I finally got the "package" to complete the ultimate system safeguard. One of my "mates", Ron Kleinschafer, put together the latest modification of his Eprommer for me. All the circuitry and chips were in place and all that was left was to get an appropriate transformer to run the thing. After a bit of searching I finally found the 15 V center tap with two 7.5 V outputs from a local electronics firm. The only problem was that it had to be ordered from the factory. Anyway, it finally arrived on August 23rd and I spent the better part of the evening installing it and getting the Eprommer ready to go.

I'm sure that there are some among you who wonder why I bothered with an Eprommer. There are a couple of reasons. First, Ron's Eprommer (hereafter called 'Ep') comes with another excellent software program that allows you to read eprom chips (or GROM and DSR's) and save them to disk as well as program eproms. With that in mind, it is a relatively simple matter to read the TI Disk Controller DSR (or RS232) and save them to disk in case the original on your card abruptly "dies". Or, how about the GROM chips in the console? The addresses are general knowledge and saving each of them to disk as insurance against failure is a straightforward process. Since the DSR's and GROM chips are about the only specialty chips in our little orphan, being able to create your own replacements makes the ol' TI darn near immortal!

Another advantage to having Ep is the ability to replace the volitile RAM chip in your RAMdisk used for DSR storage with an eprom. If you have been having problem with your RAMdisk DSR acting flaky (like George's Horizon) the prospect of a permanent replacement suddenly becomes attractive. Once the RAMdisk DSR has been stabilized and no more changes are anticipated, copy it to disk and blow an eprom to replace it.

Couple the Ep with some A/L programming expertise and you can customize your console GROM chips. Change the title screen without the GRAMKRACKER or make other changes if you like. Imagination is the only real limitation. While you are customizing things, make your own cartridges by writing the code and stuffing it into an eprom chip on an old CAR WARS cartridge. It makes for some interesting experimentation!

Anyway, I'm sure that these cool fall evenings will be shorter with Ep plugged in and the ol' TI warmed up.

TML—WHAT IT ISN'T

By Bob Carmany

Before we get started, just a brief word about what this article ISN'T. It isn't a review of THE MISSING LINK but rather an overview. Even the most ardent publicist will admit, when backed into a corner, that there are drawbacks to whatever it is that he is selling. Often times, these shortcomings are glossed over or ignored completely. It is up to the purchaser to determine whether the product suits his needs. With that in mind, let's continue.

The latest entry in the world of Extended BASIC extensions is a program authored by Harry Wilhelm and distributed by Texaments called THE MISSING LINK. It, like all of the others, provide assembly language support for XB that greatly increase XB's power and flexibility. After the first "blush" of a new program is over, we must determine if there are any drawbacks. Let's take a look!

THE MISSING LINK (or TML) operates by changing the screen to the bit-map mode when it is running. While this allows for some spectacular programming advantages, it also leads to some limitations. Before we look at some of these limitations, we'll take a look at what TML CAN do.

There is a rather common series of routines for general screen management that allow one to clear the screen, color each pixel, and assign a color to the entire screen. These are the usual routines found in almost every extension of XB in one form or another. So far, just basic stuff!

A series of "pen" routines follows for the assignment of foreground and background colors, inverse pixels (video), and turning the "pen" off and on are next. These, by themselves, aren't all THAT spectacular but they are options that are not found in other XB enhancements. When they are combined with the next group of options --- windowing --- we get to see the magic stuff that bit-mapped graphics is made of! The prospect of having five on-screen windows, each with a different foreground and background color, leads to dreams of the "ultimate" XB program. There is a series of Cartesian graphing capabilities for truly amazing screen presentations. The list goes on and on! Sprite handling routines are resident in the program and even the LOGO "Turtle Graphics". TML is a veritable smorgasbord of graphic capabilities!!

Even the text display routines are impressive. The ability to display text at any pixel row or column and alter the size of the characters presents a wealth of possibilities. Text position is now virtually unlimited on the screen. Just a run through the DEMO program will show you what TML will do and do quite impressively. With all of this "magic stuff", there has to be a trade-off somewhere. After all, no program is perfect.

None of the old XB stand-by routines for putting graphics and text on the screen work properly with TML. Since it operates in the bit-map mode, PRINT, DISPLAY AT, INPUT, ACCEPT AT, HCHAR, VCHAR, GCHAR, and SPRITE result in patches of color rather than the characters themselves. No worries, there are replacements for most of them in TML.

I didn't realize how limiting not having these XB character handling routines was until I decided to re-write a program that I had put together. Extended BASIC is really a "weak sister" as far as error handling routines and error control is concerned. It is not nearly as elegant as Assembly Language, for example. One of the best routines to limit or define input, however, is the ACCEPT AT statement. I quickly realized that the VALIDATE constraint that I had used to control input into a specific range was no longer available. A simple "ACCEPT AT....VALIDATE(UALPHA)" became a byte-devouring, several line process. In the end, I decided that the spectacular screen displays that I had anticipated weren't worth the increased programming effort. It might look nice but it would have compromised the stringent error control that I wanted in the program.

Another problem is a trade-off between flexibility and stack space. If you use a lot of variable names in your program, you may very well run out of stack space if you aren't careful. This is particularly so if you use the 16 color mode. This takes some of the attractiveness out of the program --but not much. Arrays are a problem as well with TML. Programming technique and planning now become very important at times.

Depending on the programming application, the loss of the standard XB character handling routines and stack size limitations can be truly crippling. On the other hand, the routines in TML allow for some spectacular screen displays and a welcome group of extensions to the XB environment.

So, where does this leave us in our discussion of THE MISSING LINK? It is definitely worth having and using. Just remember that the greatest improvements to the XB environment have been made in the realm of graphics capabilities. In fact, it will produce some of the best XB graphic screen displays that I have ever seen. Conversely, there has been a significant reduction in the error and range control abilities of XB. Where data must be input into a program or prompts responded to, it might be best to use plain ol' XB. You decide which programming applications are best for TML. Just remember, it won't do everything!

HOME FINANCIAL MANAGEMENT

By Herman Geschwind

Home Financial Management with the home computer is more than balancing your checkbook (which is easier done using a calculator and pencil anyway). One of the most useful functions of a home computer is its ability to assist in record keeping. It used to be that family record keeping was a real chore. Receipts and checks were kept in a shoe box, bank statements could be found in a dresser drawer, and so on..

For years I kept a monthly record of income and expenditures in a ledger book but totaling and crossfooting all these entries with a hand calculator was a time consuming chore.

The benefits of record keeping are that at tax filing time there is a convenient summary record of deductible expenses which considerably eases the logistics of tax form preparation. Further, having a running summary of various types of expenses gives you a chance to better manage your financial affairs.

Now, all this has been done for years using a simple bookkeeping system using spreadsheets, pencil and calculator. With the coming of the personal computer and appropriate software this can be greatly simplified.

The simplest form of personal financial record keeping is to keep a running account of actual expenses in various categories. For this kind of project the <u>Tax/Investment Record Keeping</u> Command Module PHM 3016 is ideal. It allows you to set up a complete home accounting system with Assets, Liabilities, Income and Expense categories. While a number of categories have been predefined and set up as defaults, the system allows that any of the defaults can be overridden with items of your own choice.

PMH 3016 is disk based and will capture as many as 660 transactions on one disk. If more than one disk is needed, additional disks can be started provided that all categories on the previous disk have been summarized on the continuation disk. This means in effect that the number of transactions can be unlimited.

The contents of each record can be customized and not only include account, amount and date but also such information as payee, document number and several lines of comments.There is a provision for categories and sub-categories so that any level of detail can be captured.

This module has extensive search, select and reporting capabilities , either on screen or to an attached serial printer. Unfortunately the module will not support non-serial printers so that those without official TI printers are out of luck...the only bad part about this cartridge.

PMH 3016 is still on the market and can be ordered by mail from such places as Triton who list this module at $5.95 in their current catalogue.

Another command module which can be used for this purpose is PMH 3007,Household Budget Management. This module differs from the previous one in several respects: (1) In terms of hardware it will run either with cassette or disk, (2) The module itself includes extra memory chips so that expansion memory is not required, (3) There is no provision for print output at all, (4) Records are written to tape or disc in program format so that even with a separate Basic program it is not possible to create printed reports.

The added benefit of this module is that in addition to actual expense, budgets can be created for all reporting items so that it is possible to track not only actual expenses but to compare these to pre-established budgets. A convenient way to make use of this facility is to use last year's actual as current year budget as a convenient way to tell at a glance how various items are trending.

Household Budget Management is more limited in its customization features. There are 99 possible income and expense categories available of which 34 may be active.

In reporting the outstanding capabilities of this module are graphics at each level of detail and a "project" feature which allows the quick annualization of any level of detail based upon year-to-date experience.

This module is also listed by Triton at $5.95.

MULTIPLAN. A spreadsheet template for actual expense tracking can be set up easily using Multiplan. This worksheet allows for the capture of 90 income and expense items per month, with a sub-total for income and expense and a gain/loss total for the month. The worksheet automatically creates year-to-date totals for all items and subtotals. The "window" feature of Multiplan allows the viewing of selected months and year-to-date. Print output can be created with any attached printer (compressed print recommended). This worksheet is available to all members if you supply me with a diskette (Multiplan and memory expansion required).

TI LIGHT PEN

OK, I've been promising this thing for several months, so here are the plans for building a lightpen along with a sample program. I have other programs which will work with the pen that I hope to share with you in future months.

First, there are some parts you will need, but fear not, the construction is simple.

You will need the following:

ONE - 9 Pin D-plug (joystick plug)
TWO - lengths of 2 conductor wire
TWO - CDS photocells
TWO - junk FLAIR pens with cap (or anything big enough to hold the CDS cells)

Now that you have the above, let's get started. First, gut the pens out and cut off the end that the tip was in, then punch out the end of the cap. Feed the wire through the bottom of the pen and out the tip. Solder the two wires to the leads of the CDS cell, and place the cell into the cap and put it on the old pen body. Tape the wire around the pen to help prevent it from pulling out by accident.

Grab the D-plug and hard-wire the leads (pin layout is below) for pin #1 to pin #7 and #9 (this should be the right direction with CALL JOYST), with pin #2 solder it to pins #2 and #5 (CALL JOYST left direction). If you have not figured it out yet, you can add up to four more pins using the other moves of the joystick routine. Wrap tape around the plug to protect the wiring.

Now, convert the program below so that you can test the pens. As you can tell, the program name is "DOT" and you just touch the dots on the screen with the pen. Depending upon your TV/monitor, you may need to adjust the contrast/brightness.

Oh yeah, before you get too involved playing with these new toys, I must give credit to Edwin McFall of Aberdeen, WA. He is the one who came up with this contraption.

```
JOYSTICK PORT PINOUT
====================
  -------------
 \ 1 2 3 4 5 /
  \ 6 7 8 9 /
   ---------
```

PIN	USED FOR
1	NOT USED
2	COMMON LINE JOYSTICK #2
3	UP
4	FIRE
5	LEFT
6	NOT USED
7	COMMON LINE JOYSTICK #1
8	DOWN
9	RIGHT

MULTIPLAN TIPS

by
Steve Zimmerman

One of the most useful features of Multiplan (and one that I miss on my Tandy 200 portable!) is WINDOWS! Windows are a BIG THING now--just read the magazine reviews and ads for programs such as Framework and Symphony. Windows enable you to see two parts of the screen, in different areas of the worksheet, at the same time. Sounds simple enough, but what good is it? Well, let's say that you have a worksheet with about 60 rows of labels in Column 1, and that you enter data for each day in a month in columns 2-32. Since your basic column width shows only 4 columns, you can see the labels only while entering the first 3 days data. If you make the columns narrower, of course, you can see more--if the numbers aren't too long! If the numbers are too long, and you make the columns narrow, you just see "#####", an error symbol which means that you have too many numbers to display in that column width. To enable you to enter later columns of data in the proper rows, you need to create a window.

To do this, position your cursor (the cell pointer) in R1C2 (one column to the right of your labels), and hit the W key (for Windows). On the next menu, you want to key S (for Split). On the next menu, key V for vertical, and Multiplan will respond with "at column 2", Linked yes no, with the NO highlighted. Use the Tab (Ctrl 2) to move the command cursor from the 2 in "at column " to the Linked field, key Y to link the windows, and then key Enter. With practice, this command sequence becomes W S V Tab Y Enter. You now have a window, linked vertically. As you move the cell pointer down in the active window (the one you just created), the labels move along. As you move the cell pointer to the right, columns will disappear on the left side of the second window, but the labels will remain in view.

To 'uncreate' this window, key W (for Window) C (for Close); Multiplan will respond with #2, key Enter, and the window closes.

You can do the same thing with horizontal windows, if you have labels across the top and you want to see them as you move up and down. One thing you cannot do, however, is to link windows which intersect in the same cells. R1C1 cannot be in 2 linked windows, one for R1 (horizontal) and one for C1 (vertical).

You can move the cell pointer between windows by using Ctrl 6 (change window), which makes a different window the active window. Ctrl 6 again will take you back to the original active window.

I hope that this 'raises the shades' for those of you with windowing questions! To cover some other questions, I don't know of any way to load the recalc routines when you start out in Multiplan. This will, of course, cause those with only one drive to have to swap disks, starting with the Multiplan disk, then your data disk, and then back to the Multiplan disk to load the recalc routines. The program will also have to access the system disk to set up Names, to do Xternal copies, to show you your disk directory, and to load the routines for logical operators. It appears that these are handled by some type of overlay, which is loaded from the system disk only when needed. I don't know for sure, but it is possible that one or more of these routines uses the same (or overlapping) addresses in memory. Oh, yes, one more thing which requires access to the Multiplan system disk is the Help files. These can be quite handy, though! To get Help, just move your command line cursor to the command you need Help with, and key Fctn ?. To get Help with Help, move the command cursor to Help and hit Enter.

One more handy hint on recalculation--to recalculate a single cell, move your cell pointer to that cell, key E (for Edit), DON'T CHANGE ANYTHING, and hit Enter. This will cause the contents of THAT CELL ONLY to be recalculated. Of course, if you left recalc turned on....

Other handy hints--to Un-Name Names, enter the Name, and delete the reference in the "to refer to" field--this deletes the name by making it refer to nothing. To delete an Xternal reference or link, the same procedure is used. You enter X (for Xternal), C (for Copy), from sheet (sheetname), Name (enter the name), to: (delete anything in this field), Linked (yes), Enter. This redefines the Xternal copy link to refer to nothing, and the value placed in the cell by the Xternal Copy command will disappear, and the cell will now be unlocked. To change the target of an Xternal copy, use the same procedure, but specify the new cell or cells in the "to:" field. Since each cell or range on the supporting sheet can have only one target (on the dependent sheet), the old link will be replaced by the new.

BASIC TI-WRITER

by Tom Kennedy

How many of you have a typewriter, please raise your hand, keep your hand up if your typewriter has interchangeable text. How about automatic bold and underline. Or some amount of memory storage (for letter heads, etc.). How about an erase key? Those of you left have probably got a pretty expensive piece of machinery, but TI-WRITER has ten times the functions, or features of the best typewriters. With TI-WRITER, your only limitation is your own creativity. To start off with, what will you need to operate your Word Processor? You must have the 99/4A console (TI-WRITER won't work with the 99/4), a TV or monitor, the cartridge and disk package, the disk system, memory expansion, the RS232 interface, and a printer. In other words, the whole works. The printer is something you definitely want to be careful in choosing because all of your work will be in vain if you can't print out exactly what you type in, and with an attractive appearance.

First, let's look at the command line. That's the line at the top of the screen when you're in the command mode. There are seven commands shown and sixteen sub-commands that are options of the main seven. The commands are selected by typing only the letters that are capitalized in the word. For instance: "F" for Files, "SH" for SearcH, or "LF" for Load File. That's an interesting point, you can access any of the sub-commands from the main command menu. In other words, to ShowDirectory (which is a disk catalog) you would enter the command mode, (FCTN 9), and either type "F" for files, and "SD" for ShowDirectory, or just type "SD" immediately. This feature saves a lot of time and keystrokes.

The first command is Edit. This simply enters you into the text-edit mode in which text is created.

Next is Tabs. When you hit "T", the top part of your text is shown with a scale across the top showing the current tabs and margins. Changes are made by simply typing over existing entries with the appropriate symbol (L,R,T, or I).

"F" for files allows you to work with your text file as a whole, to Load, Save, Delete, Print, Purge, or ShowDirectory. "PF" for print file is not what you'll get when you print out through the text formatter, it just prints a "hard copy" of the whole file, just as you see it on the screen. It doesn't print with any of the modifications made by the format commands (more on those later). "PF" is useful for making a fast copy of a long letter, or whatever, in order to check for errors without having to scroll back and forth or up and down. Purge simply erases the file from memory to prepare for a new entry. It is similar to the "NEW" command in BASIC.

Next is "L" for Lines. This allows you to work with whole lines or groups of lines by moving them to somewhere else in the text, copying to somewhere else and leaving the original intact, to delete groups of lines, or to quickly move the cursor to some line in the text with the ShowLines option.

SearcH (or "SH") gives you the option of either the FindString routine or the ReplaceString routine. FindString will move the cursor to the first and/or each successive use of the word string you give. ReplaceString searches the text for a given string and replaces all or one occurrence with the new string. This is great for correcting a repetitive spelling error.

RecoverEdit is a failsafe repair in case the text buffer was purged in either the File or Quit command. It will pull back everything but the first line and restore the file. I guess the loss of the first line is the penalty paid for accidentally erasing a file, which can't be done very easily.

Finally, Quit, as the name implies, blows it all apart and leaves you with the title frame. But before it goes, all open files are closed (such as to disk or printer) so no data is lost. Fortunately, it first gives you the option of saving your file (in case you forgot to do that already) or just purging the file and going back to the edit mode. But if you really want to quit, you type "E" for Exit and it shuts down.

Now let's go over the keyboard. TI-WRITER makes extensive use of the FCTN and CTRL keys and uses every possible function of the top line of keys (the numbers). There are also many functions that have duplicate methods of keystrokes to activate them. For instance, to enter the command mode, you either press FCTN 9 or CTRL C. The reason for this duplication is to allow you to choose which is easiest to use depending on where your fingers are at. The problem though, is that it can be very confusing trying to remember the fifty different key combinations that activate the thirty functions. A better method is to just pick which keys you're going to use for what function and ignore the rest. What I do is use the number line keys for anything shown on the overlay strip and just memorize the few functions hidden down in the keyboard. Let's start by going down the overlay strip, left to right.

OOPS! CTRL 1 CTRL Z This can be a real lifesaver. It recovers, or "backs up" a function that you didn't mean to hit. Like if you goofed and hit "Delete Line" instead of "Insert Character", you just hit "OOPS!" and the line comes back.

Del Char FCTN 1 CTRL F This is the same as "DEL" in console BASIC. It deletes one character under the cursor and pulls the rest of the line up to fill.

Reformat CTRL 2 CTRL R This is used to close up the text after using Insert Character. It deletes all spaces between the cursor and the next word in the text. Then it draws all subsequent words up through the paragraph until it encounters a Carriage Return.

Ins Char FCTN 2 CTRL G In the Word Wrap mode (solid cursor), thirty two blank characters are inserted after the cursor and the bulk of the text is pushed down the line. After insertion of new text, you hit Reformat and any remaining spaces are removed. In the Fixed mode (hollow cursor), this operates the same as in console BASIC.

Screen Color CTRL 3 This allows you to choose which of the five color combinations of text/screen you prefer. The default, for no good reason, is white on dark blue. But I find this hard on the eyes. I prefer to turn down the color on my monitor and use either black on green or black on light blue.

Del Line FCTN 3 CTRL N Deletes the entire line that the cursor is on, including the space of the line.

Next Paragraph CTRL 4 CTRL J This advances the cursor to the beginning of the following paragraph and puts the first line of text at the top of the page.

Roll Down FCTN 4 This is called a "vertical block scroll", which means that the next 24 lines of text are shown. This is handy for scanning quickly down the text to get to some point.

Dupe Line CTRL 5 This creates an exact duplicate of the line the cursor is on and places it directly below. Some have questioned its value in writing text, especially since the Move/Copy function can do the same, but this key makes it faster and easier to create repetitive lines such as a double row of asterisks under a title.

Next Window FCTN 5 This is a "horizontal block scroll". It jumps across to display the next block of 40 characters, in increments of 20. For example, the screen starts out on column one to forty, then twenty to sixty, then forty to eighty.

Last Paragraph CTRL 6 CTRL H The opposite of "Next Paragraph" Paragraph.

Roll up FCTN 6 CTRL B The opposite of "Roll Down".

Word Tab CTRL 7 CTRL W This moves the cursor down the line to the first letter of each word.

Tab FCTN 7 CTRL I Just like on a typewriter, this moves the cursor to next setting, defined using the Tab function on the command line.

New Paragraph CTRL 8 This places a Carriage Return symbol at the end of the line you're on and skips down to the next line. If you have preset an auto-indent, (by using an "I" in Tabs) then it also indents over to the proper column.

Ins Line FCTN 8 CTRL O Inserts a blank line above the line the cursor is on.

New Page CTRL 9 Inserts a blank line with a Np and Cr symbol at the beginning. This causes the printer to feed to the next page.

Command/Escape FCTN 9 CTRL C This is how you exit from the edit mode to get to the command line and the functions above it. It also is used to cancel a command already in progress.

Word Wrap CTRL 0 This switches from the "Word Wrap" mode to the "Fixed" mode. In Word Wrap, when you reach the end of the line the cursor jumps to the next line. If you're in the middle of a word at the end of the line, the whole word you were on moves down too. This allows you to just type continuously without looking up to see when to hit enter. In the fixed mode, when you reach the end of the line your letters just pile on top of each other and you hit enter to move to the next line.

Line Numbers FCTN 0 This removes or displays the four-digit line numbers at the left side of the screen. The numbers are used for reference when manipulating blocks or lines of text, just like when you're editing a BASIC program. You need line numbers to refer to where changes will be made.

Quit FCTN = Quit is the same as in console BASIC. Use Quit option of the Command line to safely exit TI-WRITER.

Back Tab CTRL T The same as Tab except it backs up one setting.

Beginning of Line CTRL V Moves the cursor to the beginning of the line you're on.

Del.End of Line CTRL K This is just like Delete Character (FCTN 1), except it takes out everything to the right of the cursor.

Home Cursor CTRL L This moves the cursor to row 1, column 1, on the screen only. Unfortunately, it doesn't move to first line of text, which would be more convenient when working with a long document and wanted to jump to the top.

Left Mrgn Release CTRL Y Allows you to temporarily back-arrow beyond the left margin when it has been set past zero.

The last four key functions to mention are the cursor arrows: UP, DOWN, LEFT, &RIGHT. These stay the same as in console BASIC.

AIR FORCE SONG

```
1 @=0 :: {=1 :: |=2 :: _=3 :
: \=4 :: GOTO 100
2 A,AF,AS,B,BF,C,CS,D,DF,DS,
E,EF,F,FS,G,GF,GS,HA,HAF,HAS
,HB,HBF,HC,HCS,HD,HDF,HDS,HE
,HEF,HF,HFS,HG,HGF,HGS,J,L,L
A,LAS,LB,LBF,LC,LCS,LD
3 LDF,LDS,LE,LEF,LF,LFS,LG,L
GF,LGS,Q :: CALL DELSPRITE :
: CALL SOUND :: !@P-

100 !
110 !
120 LA=110.00 :: LAS=116.54
:: LBF=116.54 :: LB=123.47 :
: LC=130.81 :: LCS=138.59 ::
 LDF=138.59 :: LD=146.83 ::
LDS=155.56 :: LEF=155.56 ::
LE=164.81
130 LF=174.61 :: LFS=185.00
:: LGF=185.00 :: LG=196.00 :
: LGS=207.65
140 AF=207.65 :: A=220.00 ::
 AS=233.08 :: BF=233.08 :: B
=246.94 :: C=261.63 :: CS=27
7.18 :: DF-277.18 :: D=293.6
6 :: DS=311.13 :: EF=311.13
:: E=329.63 :: F=349.23
150 FS=369.99 :: GF=369.99 :
: G=392.00 :: GS=415.3
160 HAF=415.3 :: HA=440.00 :
: HAS=466.16 :: HBF=466.16 :
: HB=493.88 :: HC=523.25 ::
HCS=554.37 :: HDF=554.37 ::
HD=587.33 :: HDS=622.25 :: H
EF=622.25 :: HE=659.26
170 HF=698.46 :: HFS=739.99
:: HGF=739.99 :: HG=783.99 :
: HGS-830.61
180 Q={
190 L=500
200 CALL SOUND(_*L/\,E,@,C,@
,3.775*C,30,-\,@)
210 CALL SOUND(L/\,G,@,C,@,3
.775*C,30,-\,@)
220 CALL SOUND(L,G,@,AS,@,3.
775*CS,30,-\,@)
230 CALL SOUND(L,G,@,B,@,3.7
75*D,30,-\,@)
240 CALL SOUND(L/_,F,@,B,@,3
.775*LG,30,-\,@)
250 CALL SOUND(L/_,E,@,B,@,3
.775*LG,30,-\,@)
260 CALL SOUND(L/_,D,@,B,@,3
.775*LG,30,-\,@)

270 !
280 CALL SOUND(L,E,@,C,@,3.7
75*C,30,-\,@)
290 CALL SOUND(L,F,@,B,@,3.7
75*D,30,-\,@)
300 CALL SOUND(L,FS,@,C,@,3.
775*D,30,-\,@)
310 CALL SOUND(L,G,@,C,@,3.7
75*E,30,-\,@)
320 ON Q GOTO 340,530,710

330 !
340 CALL SOUND(_*L/\,HA,@,F,
@,3.775*F,30,-\,@)
350 CALL SOUND(L/\,HC,@,F,@,
3.775*F,30,-\,@)
360 CALL SOUND(L,HC,@,F,@,3.
775*C,30,-\,@)
370 CALL SOUND(L,HC,@,HA,@,3
.775*F,30,-\,@)
380 CALL SOUND(L/_,HD,@,A,@,
LA,@)
390 CALL SOUND(L/_,HC,@,F,@,
LA,@)
400 CALL SOUND(L/_,HA,@,F,@,
LA,@)

410 !
420 !
430 !
440 CALL SOUND(L,G,@,E,@,LC,
@)
450 CALL SOUND(L,G,@,AS,@,3.
775*CS,30,-\,@)
460 CALL SOUND(L,G,@,B,@,3.7
75*D.30 -\,@)
470 CALL SOUND(L/_,F,@,B,@,3
.775*G,30,-\,@)
480 CALL SOUND(L/_,E,@,B,@,3
.775*G,30,-\,@)
490 CALL SOUND(L/_,D,@,B,@,3
.775*G,30,-\,@)

500 !
510 Q=Q+{
520 GOTO 200
530 CALL SOUND(_*L/\,HB,@,D,
@,3.775*D,30,-\,@)
540 CALL SOUND(L/\,HD,@,HB,@
,3.775*D,30,-\,@)
550 CALL SOUND(_*L/\,HD,@,G,
@,LB,@)
560 CALL SOUND(L/\,HD,@,G,@,
LBF,@)
570 CALL SOUND(L,HD,@,FS,@,L
A,@)
580 CALL SOUND(L/_,HC,@,FS,@
,3.775*D,30,-\,@)

590 CALL SOUND(L/_,HB,@,FS,@
,3.775*D,30,-\,@)
600 CALL SOUND(L/_,HA,@,FS,@
,3.775*D,30,-\,@)

610 !
620 CALL SOUND(L,G,@,B,@,3.7
75*G,30,-\,@)
630 CALL SOUND(L/_,HG,@,G,@,
LG,@)
640 CALL SOUND(L/_,HG,@,G,@,
LG,@)
650 CALL SOUND(L/_,HG,@,G,@,
LG,@)
660 CALL SOUND(L,HG,@,G,@,LG
,@)
670 CALL SOUND(L,G,@,F,@,3.7
75*G,30,-\,@)

680 !
690 Q=Q+{
700 GOTO 200
710 CALL SOUND(_*L/\,HA,@,F,
@,3.775*F,30,-\,@)
720 CALL SOUND(L/\,HC,@,F,@,
3.775*F,30,-\,@)
730 CALL SOUND(L,HC,@,F,@,3.
775*C,30,-\,@)
740 CALL SOUND(L,HC,@,HA,@,3
.775*F,30,-\,@)
750 CALL SOUND(L/_,HA,@,F,@,
3.775*C,30,-\,@)
760 CALL SOUND(L/_,HB,@,F,@,
3.775*C,30,-\,@)
770 CALL SOUND(L/_,HC,@,F,@,
3.775*C,30,-\,@)

780 !
790 CALL SOUND(L,HB,@,DS,@,3
.775*FS,30,-\,@)
800 CALL SOUND(L,HB,@,DS,@,3
.775*F,30,-\,@)
810 CALL SOUND(L,GS,@,E,@,3.
775*E,30,-\,@)
820 CALL SOUND(L,E,@,D,@,3.7
75*LE,30,-\,@)

830 !
840 CALL SOUND(L,HC,@,E,@,3.
775*A,30,-\,@)
850 CALL SOUND(L/|,HC,@,E,@,
LA,@)
860 CALL SOUND(L/|,HC,@,E,@,
LA,@)
870 CALL SOUND(L,HCS,@,E,@,L
A,@)
880 CALL SOUND(L/|,HCS,@,E,@
,3.775*G,30,-\,@)

890 CALL SOUND(L/\,HA,@,CS,@
,3.775*G,30,-\,@)
900 CALL SOUND(L/\,HA,@,E,@,
3.775*G,30,-\,@)

910 !
920 !
930 !
940 CALL SOUND(L,HD,@,F,@,3.
775*F,30,-\,@)
950 CALL SOUND(L/|,HD,@,F,@,
3.775*LF,30,-\,@)
960 CALL SOUND(L/|,HD,@,F,30
,3.775*LF,30,-\,@)
970 CALL SOUND(L,HDS,@,DS,@,
3.775*LF,30,-\,@)
980 CALL SOUND(L,HDS,@,DS,@,
3.775*LFS,30,-\,@)

990 !
1000 CALL SOUND(L/_,HE,@,HC,
@,3.775*LG,30,-\,@)
1010 CALL SOUND(L/_,HD,@,HC,
@,3.775*LG,30,-\,@)
1020 CALL SOUND(L/_,HC,@,G,@
,3.775*LG,30,-\,@)
1030 CALL SOUND(_*L/\,HE,@,E
,@,3.775*A,30,-\,@)
1040 CALL SOUND(L/\,HC,@,E,@
,3.775*A,30,-\,@)
1050 CALL SOUND(_*L/\,HE,@,F
S,@,3.775*D,30,-\,@)
1060 CALL SOUND(L/\,HC,@,FS,
@,3.775*D,30,-\,@)
1070 CALL SOUND(L,HD,@,F,@,3
.775*LG,30,-\,@)

1080 !
1090 CALL SOUND(L/|,HC,@,C,3
0,3.775*C,30,-\,@)
1100 CALL SOUND(L/\,HC,@)
1110 CALL SOUND(L/\,HC,@,G,@
,G,@)
1120 CALL SOUND(L/_,HA,@,A,@
)
1130 CALL SOUND(L/_,G,@,LG,@
)
1140 CALL SOUND(L/_,E,@,LE,@
)
1150 CALL SOUND(L,C,@,LC,@)
1160 FOR J={ TO 150
1170 NEXT J
1189 !@P+
1190 END
```