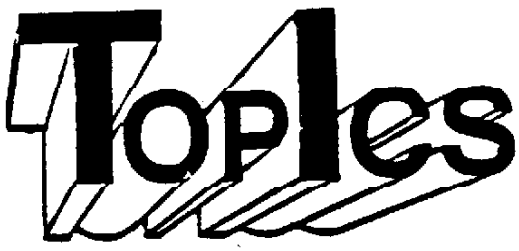

LA 99er COMPUTER GROUP
VOL 8 NO 4 LOS ANGELES CA APR 1989
Newsletter

TERRIES CORNER

Ever try to write and format a column with a 15 month crying soulfully in the background? Last month I appealed for help in getting this newsletter ready for printing. Steve Mehr and clearly by the return of double column, Tom Freeman have given me a hand. Unfortunately I need more. There must be a couple of you out there who are interested in seeing the integrity of this newsletter continue.

At the moment and for the last couple of months I have the responsibility of my two grandchildren from dawn until late night, my daughter-in-law clearly unable to sustain sobriety. That coupled with my weekends with my parents are a large load to handle.

I was unable to attend Ticoff or the Fayuh, and am also going to miss Ottawa. Well priorities prevail.

Now that my cards are once again openly on the table, I am once again requesting assistance.

CRAIG MILLER

I personally, and the club also have recently again been sneeringly called "Miller worshippers". Well why not, I am quite happy and honored to have Craig and his family in my circle of friends. Craig has contributed greatly to this community, and continues to support and stand behind ALL his products.

Most of the Miller bashing spreads like manure in a rain storm from one VERY jealous questionable person. He has the unfortunately misguided desire to elevate himself by maligning others in the guise of "community interest". So sick. The rest of the bashers are primarily sheep.

Craig, in order to support his family did what was necessary to produce programming that he legitimately was paid for. The 4A market no longer did this. The appreciation for Craigs products, turned into a challenge to break and crow. Sadly the two main persons involved in this quest for glory are very talented but far more immature than their young years.

Why two young (other than above) software/hardware vendors chose to waste energy in a no win situation is unfortunate. Too soon old, to late smart.

Yes, LA 99ers strongly respect Craig Miller, and are proud he will be our June speaker. Craig hopes by showing what is done in the PC world, there will be new ideas for people such as Rodger Merritt, Steve Mehr, Steve Doran, or any other programmers in our midst. This will be a pro 4A, not convert to PC meeting. We welcome your participation.

NEWS RELEASE
Dated April 9, 1989
by Steve Mehr, UG member

The L.A. 99er's Computer Group is proud to welcome back Craig Miller of MG at their June 28th meeting. At this special meeting, Craig will be demonstrating a professional Desktop Publishing system run on an IBM AT., says Steve Mehr, Program Chairman of the group. "The purpose of inviting Craig to our group for a special meeting is to help provide ideas for TI programmers. This will present a good way for those unfamiliar with other computer operating systems to see what else is out there.", Steve said.

MG's releases included Advanced Diagnostics, Explorer, Diskassembler, The Smart Programming Guide for Sprites, The Orphan Chronicles, The Smart Programmer, Night Mission, Gram Kracker, The Turbo XT and Triton's Super Extended Basic. Steve also said that everyone is welcome to attend the meeting, especially those interested in programming for the TI or Desktop Publishing for their business. "Attending this meeting may be the encouragement needed to turn someones programming idea into reality.", Steve commented.

"This special meeting is one of many projects our group will be involved with in the months to come.", Steve added. The meeting will be free of charge, as are all their meetings, and will be held in the meeting room of the Torrance Public Library, 3031 Torrance Blvd. Torrance, Ca. 90503, and will start at 7:30 P.M. For more information, please contact Steve Mehr at (805) 379-2937.

* WHAT YOU MAY HAVE MISSED * at last months meeting was me! Yes I did make it but had to leave early. Inventory was cancelled, but work was not. "I got the message!" Hopefully, notes on the demonstrations are elsewhere in the newsletter.

+ +

An Evening with Ray Bradbury

On April 8, 1989, Ray Bradbury appeared at the Thousand Oaks Library and spoke to a standing room only audience. Ray Bradbury has authored such famous Science Fiction novels as Something Wicked This Was Comes, Dandelion Wine, The Illustrated Man, The Martian Chronicles, Fahrenheit 451, and on and on and on. He shared highlights of his life, several of his philosophies, and cautioned against listening to doomsdayers, around many of his humorous stories. I was fortunate to have a front row seat, although I wish I had a tape recorder and a note pad too. After his hour and a half talk, Mr. Bradbury answered several questions from the audience.

Some of his books were on sale that evening and after his talk, you could purchase them (autographed!) including his latest hard cover, The Toynbee Convector. Mr. Bradbury arranged quite a discount for The Friends of the Library, the organization who sponsored his talk, to sell the books, and all proceeds went directly back to them. You see, Mr. Bradbury has been a friend of the library since the age of 12! By the way, did you know that Fahrenheit 451 was written in the basement of UCLA's library? How about that!

P.S. For your pleasure, we are reprinting an article about Ray Bradbury that ran in the Thousand Oaks newspaper, The News Chronicle, elsewhere in this newsletter. Enjoy.

HOLLY McFARLAND/News Chronicle

Author Ray Bradbury spoke to a crowd of about 300 at the Thousand Oaks Library on Saturday night.

Audience gets a charge out of Bradbury's tales

By LINDA COOPER
News Chronicle

Science fiction writer Ray Bradbury's career came as a shock. He was struck by the muse when he was electrocuted at a carnival at the age of 12.

It was a fateful day when the carnival rolled into his town in November 1932, Bradbury told a standing room only crowd of about 300 people Saturday at the Thousand Oaks Library.

Two things made a deep impression on the young Bradbury that day — the death of his uncle in a holdup and Mr. Electrical at the carnival. Mr. Electrical sat on a wired chair, and when he flipped a switch, his hair stood on end and sparks flew out of his ears. He then took a blazing sword and touched Bradbury, and the same things happened to him.

"Then he pointed at me and said, 'Live forever,' " Bradbury said. And while he thought that was great advice, he wasn't sure how to go about it.

The next day, he went to his uncle's funeral, and on the way back he made his parents stop at the carnival so he could visit Mr. Electrical. The man took a liking to the boy and gave him a tour of the carnival, then the two sat by a lake and philosophized.

Mr. Electrical told Bradbury that they were old friends. In fact, he said Bradbury's soul was that of a friend of his who died in a war in 1918. Needless, to say, Bradbury was impressed.

He later found out that Mr. Electrical was a defrocked Presbyterian minister from Illinois. It didn't matter. That meeting sparked Bradbury's imagination, and three months later he was writing full time. He has never regretted it.

"I'm one of the luckiest people I've ever known. It all worked out," he said. Bradbury urged his listeners to go with their dreams and never listen to doomsayers.

Bradbury follows his own advice. When the first space vehicle landed on Mars, Bradbury was at the Jet Propulsion Laboratory in Pasadena to view it. A reporter came up to him and asked him how it felt to realize that his stories about life on Mars were wrong.

"I said, 'Fool, fool. There is life on Mars, and it is us. Now get out of here,' " Bradbury told the audience.

NEWS CHRONICLE/Thousand Oaks, Calif./Sunday, April 9, 1989

You can turn ON any of the 15 bits (0-14) with the OR function without worrying about the status of the other bits. Example: F=F OR 4. This turns on bit number 2 without affecting the other bits.

You can test any bit with the AND function. Example: IF F AND 4 THEN ... bit number 2 is on.

You can turn OFF a bit with the AND and NOT functions. Example F=F AND NOT 4 (which is the same as F=F AND -5) This turns off bit number 2 without affecting the other bits.

And, you can turn ON a bit that is OFF and turn OFF a bit that is ON (toggle the bit to its opposite state without knowing the state) with the XOR function. Example: F=F XOR 4. This will turn OFF bit 2 if it is ON and it will turn ON bit 2 if it is OFF.

With this in mind lets put it to work in the previous example program but first we will map out our Flag variable F.

```
OUTPUT = Set equals Output to printer
NOLOOP = Set equals Don't execute loop
COLOR  = Set equals Color Monitor

Bit no.       15 --- 3     2       1       0
Value         --------     4       2       1

Condition     not used   COLOR   NOLOOP  OUTPUT
Old variable               D       C       B
```

And now lets apply them to the previous program example but now we only need one flag variable.

```
160 IF F AND 4 THEN CALL COLOR(x,x,x) ELSE CALL COLOR(y,y,y)
 .
 .
230 IF F AND 1 THEN PRINT #3:A$,B$
240 PRINT A$,B$
 .
 .
400 IF F AND 2 THEN RETURN
410 FOR I=1 TO 20 :: ......... :: NEXT I :: RETURN
```

In your program the IF F AND 4 statement uses the same number of bytes as IF F=1. However, since we have eliminated 2 other variables (flags C & D) and replaced B with F we have reduced the running size of the program. Also, the fewer the variables in your program the faster it runs! If your program had a few more flag variables in it they too could be eliminated by assigning their flag condition to one of the unused bits in F.

When you want to test for more than one condition at the same time simply add up the values for the bits you want to test and perform your AND and compare the result to the total value. Example:

```
IF F AND 6=6 THEN .... both the COLOR and NOLOP bits are set  ....
             ELSE .... one or none of the bits are set.
```

This type of multiple testing replaces the normal code of:

```
IF D=1 AND C=1 THEN .... both conditions are set.
```

To replace the normal code of IF D=1 OR C=1 add up the bits again but do not compare the result of AND to the total. Example:

```
IF F AND 6 THEN .... both or either bit could be set ....
           ELSE .... neither bit is set.
```

5. It can be used to easily round off a floating point number into an integer.

Since the logical expressions of AND, OR XOR and NOT work on binary integers you can use AND to round a floating point number into an integer. Normally your code to perform this may look like:

```
B=INT(B+.5)
```

By using AND you can save 3 bytes and slightly speed up your program by replacing the above code with:

```
B=B AND-1
```

This forces the computer do the rounding internally for you so it can operate on an integer. When you use this method your floating point value must be in the range of -32768.5000 through 32767.4999.

We used the AND value of -1 because -1 is the same as NOT 0 or ALL BITS ON. With all the bits turned on, your AND or filter allows all bits to pass through it and just return the result of the computer's internal rounding.

You can carry this one step further and use it to round for dollars and cents or any number of decimal places you want. For example, the normal code used to round to 2 decimal places of:

```
B=INT(B*100+.5)/100
```

Could be replaced with:

```
B=(B*100 AND-1)/100
```

But this makes the valid range limit -327.6850 through 327.6749 since we are multiplying by 100 before the AND is executed. Also, this method only saves 1 byte and with all the other floating point operations in this statement the time saved is very slight. So, we recommend that you only use this method of rounding for obtaining integers and not for rounding to decimal places.

6. Easily converts Lower case to Upper case or visa versa.

To convert lower case or uppercase to uppercase value just AND it with 95.
Example: 97 AND 95 returns 65 .... 65 AND 95 also returns 65.
a A A A

To convert uppercase or lower case to lower case value just OR it with 32.
Example: 65 OR 32 returns 97 ..... 97 OR 32 also returns 97.
A a a a

To exchange case, upper to lower & lower to upper, just XOR it with 32.
Example: 65 XOR 32 returns 97 ..... 97 XOR 32 returns 65.
A a a A

One practical application for this is with CALL KEY statements. Example:

```
CALL KEY(0,K,S):: IF K AND 95=89 THEN .... Y or y was pressed.
```

However, on the 99/4A, TI gave us a CALL KEY that only returns uppercase values so, CALL KEY(3,K,S):: IF K=89 THEN .... works the same as the above example and K will equal 89 whether Y or y was pressed.

We hope that the previous pages have helped shed some light on the potential uses for AND, OR, XOR and NOT. With a little experimentation in your programs you may find many more areas that these logical expressions can be used to help speed up your programs and save a few bytes of code.

You can turn ON any of the 15 bits (0-14) with the OR function without worrying about the status of the other bits. Example: F=F OR 4. This turns on bit number 2 without affecting the other bits.

You can test any bit with the AND function. Example: IF F AND 4 THEN ... bit number 2 is on.

You can turn OFF a bit with the AND and NOT functions. Example F=F AND NOT 4 (which is the same as F=F AND -5) This turns off bit number 2 without affecting the other bits.

And, you can turn ON a bit that is OFF and turn OFF a bit that is ON (toggle the bit to its opposite state without knowing the state) with the XOR function. Example: F=F XOR 4. This will turn OFF bit 2 if it is ON and it will turn ON bit 2 if it is OFF.

With this in mind lets put it to work in the previous example program but first we will map out our Flag variable F.

```
OUTPUT = Set equals Output to printer
NOLOOP = Set equals Don't execute loop
COLOR  = Set equals Color Monitor

Bit no.       15 --- 3     2       1       0
Value         --------     4       2       1

Condition     not used   COLOR   NOLOOP  OUTPUT
Old variable               D       C       B
```

And now lets apply them to the previous program example but now we only need one flag variable.

```
160 IF F AND 4 THEN CALL COLOR(x,x,x) ELSE CALL COLOR(y,y,y)
 .
 .
230 IF F AND 1 THEN PRINT #3:A$,B$
240 PRINT A$,B$
 .
 .
400 IF F AND 2 THEN RETURN
410 FOR I=1 TO 20 :: ......... :: NEXT I :: RETURN
```

In your program the IF F AND 4 statement uses the same number of bytes as IF F=1. However, since we have eliminated 2 other variables (flags C & D) and replaced B with F we have reduced the running size of the program. Also, the fewer the variables in your program the faster it runs! If your program had a few more flag variables in it they too could be eliminated by assigning their flag condition to one of the unused bits in F.

When you want to test for more than one condition at the same time simply add up the values for the bits you want to test and perform your AND and compare the result to the total value. Example:

```
IF F AND 6=6 THEN .... both the COLOR and NOLOP bits are set  ....
             ELSE .... one or none of the bits are set.
```

This type of multiple testing replaces the normal code of:

```
IF D=1 AND C=1 THEN .... both conditions are set.
```

CUE CARDS
JUG TEXAS

EDITOR OPERATIONS

FILE: !EDIT-CUES | EITHER !Comm

CURSOR MOVEMENT	!CTRL	!FCTN(14 WAYS!)
Fast Move		
(^) to line #1	C	9 !S(E)0(E)
(v) to End of text	C	9 !S(E)E(E)
(LN)to any Line #	C	9 !S(E)LN(E)
(?) to a Word or Para.Number		!FS /XXXX/
(<) to LM	V	!$
(>) to RM		7 !NoTabs set
(\) home(screen)	L	!$
Slow Move		
(^) up	E	E
(v) down	X	X
(>) right	D	D
(<) left	S	S
Beyond Margins		
(<) release LM	Y	!$
(<) to col.#1	!V:Y:V!	
(>) to col #80:must reset Margin(RM=80)		
Para. Moves		
(v) to next para.	4,J	
(^) to above para.	6,H	

WINDOWS	!CTRL	!FCTN
(^) up	B	6
(v) down	A	4
(>) right		5

TABS		
Forward (>)	I	7
Back----(<)	T	!$
Word----(>)	W,7	

TOGGLES ON/OFF	!CTRL	!FCTN
Word Wrap	0(zero)	
ON-solid cursor		!WW Mode
OFF-hollow cursor		!Fixed Mode
Line #'s		0(zero)
Color(screen)	3	
Prnt Codes-CTRL U (blinking (_)-Special		

DELETE		
Characters	F	1
Line	N	3
End of line	K	!$
Block of lines	C	9 !D(E):FL LL

INSERT	!CTRL	!FCTN
Characters	G	2
(Reformat)	R,2	
Line	O	8
Block of lines	C	9 !C(E):FL LL
Merge from DSK (see CMD-CUES card)		

OOPS!	Z,1	
DUPE Line	5	

FORMATTER CUES

FILE:!FMTR-CUES

MAIN CODES(these work for most TEXT)
(The DOT (.) must be first on a LINE
and/or after the semicolon(;).
ENTER codes as follows at TOP of TEXT

.FI;.LM 8;.RM 72;.PL 69;.IN +2;.LS 2

CODES may also be VERTICALLY listed. A
"cr" after each code prevents REFORMAT
(CTRL-2) from scrambling the codes.

Explanation of above CODES:
.FI -FILL-must be used to REFORMAT text
.LM 8 -Sets Left Margin at COL. 8.
.RM 74 -Sets Right Margin at COL. 74
(Delete .LM and .RM for 80 COL. print)
.PL 69 -Sets Page Length at 69 LINES
(Delete for 66 line page-default)
.IN +2 -INdents new paragraph 2 spaces
from LM.(after each "CR".
(Delete for BLOCK style format)
.LS 2 -Causes printout to doublespace
(Delete for singlespace printout)

FIXED FORMAT CODE (for TABLES/CHARTS)
.NF -Place on blank line BEFORE text
-NO FILL-Turns OFF:(.FI),(.LM), .IN)
-Prints TEXT as entered on EDIT SCREEN
-USE(.FI)to restore(.LM)and(.IN)
(Place .FI on blank line after text)

OPTIONAL CODES-
(Use separate line for codes that turn
functions ON/OFF and for .CO)
.CO-COmments(text after.CO won't print)
.AD -ADjust (justify) RM (evens it up)
(must have (.FI) ON)
.NA -No Adjust-Turns(.AD) OFF

SPECIAL CODES
.BP -Begin Page-starts new page
.HE t -Headers text(Chapter Title,etc.)
.FO t -Footers text(Footnotes,etc.)
(If "%" is entered in HE/FO,FORMATTER
will automatically insert page nos.)
.PA n -PAge # reset for .HE/.FO
.SP n -Skip "n" lines in printout
(leave space for pictures,etc.)
.IN -n -Outdent "n" columns from LM
(turn OFF with .IN +n)
.IN n -Indent "n"columns from col. #1
(use to place address/signature blocks
Tables,Charts,etc.on right side of pg.)
(Turn OFF with .IN +n)
.IF DSK?.FILENAME -Include File
(use to print MANY files as ONE BOOK)

```
CASE CHANGE        !CTRL !FCTN
  To lower case    ! .   !   !
  To upper case    ! ;   !   !
-------------------------------
SEPARATORS         !     !   !
  End Para.(cr)    !     !   !<ENTER>
  New Para.(cr)    ! N,B !   !
  New Page (PAcr)  ! P,9 !   !
-------------------------------
MODE CHANGE        !CTRL !FCTN
  To CMD Mode      ! C   ! 9 !‡‡
  To EDIT Mode     !     !   ! E<ENTER>
-------------------------------
‡ Not On TEMPLATES-mark keys with Felt
  Tip pen or Stick Ons:(CTRL T,Y,K,L,V)
‡‡ CTRL C is PREFERRED- in most cases!
```

Clean those print heads

"It takes only three things to get good, dark, crisp print from you printer," writes Chuck Reinhart, of Bellaire, New York. 1. A properly adjusted printer

3. A clean print head.

The guide for the fine print wires gradually gets clogged with a mixture of lint, ink and oils from the ribbon. As this dirt builds up and dries out, the pin wires drag in the guide. The result is you get light, low-contrast print even from a new ribbon. The following is a procedure for cleaning the print head that is quick, simple and does not require removal of the print head.

• Obtain an aerosol can of Color TV Tuner Cleaner (Radio Shack No. 64-2328 or equivalent). Make sure the label states that it contains silicone, that it will not harm plastic and that it has a plastic tube to plug into the spray nozzle.

• Turn off the printer. Leave paper in the printer, but remove the ribbon. Gently move the print head to the middle of the carriage.

• Cut a two-inch square from a lint-free cotton handkerchief. Fold the cut cloth over on top of itself a couple of times until it is about the width of your printer ribbon and is about four layers.

• Insert the cloth into the print head exactly where the ribbon was, between the pin guide and the ribbon shield. The cloth should not fit too tightly.

• Insert the tube into the aerosol spray cap. Put the end of the tube in contact with the cloth next to the pin guide of the print head and give a short quick press to wet the cloth.

• Turn on the printer and send a page of print to the printer (self-test may be used). Now, move the cloth a little to the side so that you have a clean spot. If necessary, give the cloth another shot of fluid and print out another page.

• Remove the cloth from the print head and print a page (without the ribbon). If you see any printing on the paper, put the cloth back into the print head and repeat the whole process until the page prints clean.

• Finally, install the ribbon and enjoy the improved print.

Courtesy
MICROPENDIUM

```
 (use to "call in" print code commands)
.TL nn:aa,oo,pp,qq,etc. -TransLiterate
 (use to make one KEYSTROKE call up any
  other key or a long STRING of keys-
  phrases,graphics,print commands,etc.)
-------------------------------------
MAIL LIST CODES(see TI-Writer Manual)
.ML f   -MailList (VALUE fILE)
.DP n:t -Define Prompt
&n&     -AlterNate input
-------------------------------------
PRINT CODES (recognized by FORMATTER)
@ Overstrike (insert in EDIT)
& Underline  (insert in EDIT)
^ Required Space (insert in EDIT)
  (use to tie words together for @,&
     also use to keep FORMATTER from
        moving TEXT to left margin.)
-------------------------------------
PAGE FORMAT (automatic/manual control)
 6 blank lines at top of each page.
    (.HE will print in line #3(if used)
 3 Blank lines at bottom of each page.
    (.FO will print in 2nd line from
       bottom as set by .PL )
(can't override 6/3 blank lines format)
 Page length set by .PL(66 is default)
 Margins set by.LM and.RM(0 is default)
 Fixed Format(.NF) ignores .LM,.IN,.RM
-------------------------------------
PROCEDURE
 1.Enter Formatter Codes at top of TEXT
 2.Save file to DISK
 3.QUIT Editor
 4.LOAD Formatter & turn on printer
 5.Follow Formatter Cues
 6.Make sure .LF is in Output name
TO STOP PRINTING:  HIT FCTN 4
TO RETURN TO EDITOR: HIT <ENTER>
-------------------------------------
```

BEGINNING FORTH #11 By Earl Raguse

FLOATING POINT ARITHMETIC

Last time I promised to get into FPA or Floating Point Arithmetic. This will permit us to do graphics which require the use of square root, trancendental and trignometric functions. Please read the TIFM Chapter 7. I'm not going to repeat all that. Read at least the first three pages anyway. The last two are pretty technical, and do not have to be understood to use FPA.

Floating Point numbers are represented in the computer with 8 (count em, eight) bytes, in Radix 100 notation. This is one of the reasons FPA is slow. This is equivalent to four sixteen bit words (64 bits total). Radix 100 representation uses the same built-in computer firmware capability that XBASIC or ASSY uses. If you are curious about this, read the TI EDITOR/ASSEMBLER Manual page 279.

The process of multiplying, for example, a pair of numbers represented by four sixteen bit numbers each is substantially more complex than integer multiplication of a pair of sixteen bit numbers. I will not explain in detail how this is done. I could say that is because you would probably be bored anyway, but the real fact is that I don't know. I have read books on computer design from the library on the general principle, but nowhere have I seen a discussion on exactly how TI does it. Fortunately it is not necessary to know in detail how the computer works to make it work for us.

The TI version of FPA provides every operation available to us for integer arithmetic, and then some. All the stanard arithmetic and relational operators like +, -, *, /, >, <, =, 0=, and 0< are provided, by prefixing them with F. This applies to the stack manipulators too like FDUP, FOVER, and FSWAP. For some reason, FROT is not provided, but if needed, it could easily be defined. Fetch, Print and Store are done similarly as, F@, F. and F!. We also have some new words, some of which I will be using, SIN, COS, TAN, ATAN, SQR, LOG, PI and the exponential ^, all of which are identical to XBASIC commands of the same spelling.

All very good, but how do we get numbers into FP notation? TI has povided us with S->F and F->S to move numbers from S (Single) on the stack to F (Floating) on the stack, and vise versa. In addition, we may use >F from the keyboard or on a screen to convert a string of digits to an FP number. Note also, the nice words F.R, FF., and FF.R, explained on page 3 of the TIFM, Chapter 7, as a means of controlling number formatting in lists.

TI Forth also provides >FAC, >ARG, FAC>, S->FAC, FAC->S and FAC>ARG, and other related words like FADD, FSUB, FMUL, and FDIV, all of which have the same meaning as the similar words in ASSY. I will not use or explain these. If you understand such stuff, read on in the TIFM Chapter 7, and the E/A manual.

GRAPHICS EXAMPLES

This brings me to the point where I am forced to provide some examples. Screens #66 through #6B are examples of of both DOT and LINE graphics using FPA, which I did back when I was just beginning to figure out Forth. The first Screen #66 defines several variables, namely Q (angle), R1 (radius, can't use R that's a resident Forth word), XX (X axis), YY (Y axis), W (width), H (height), and Z (a parameter). The words LOC (LOCate), MENU> (to the MENU), E1 (EXIT word), and QUIT? (looks for FCTN 4), should be obvious.

HOT, GALAXY and their supporting cast, H1 through H8, actually do the graphic work. Don't ask why I named them that way, I had to have names, so I made them short. When you see GALAXY you will understand that one. HOT is an infinite loop using MYSELF (Remember those loops from BFORTH #5?), which initializes R1 and Z, H7 announces, and H8 puts you into a DO +LOOP which calls H5 which plots a DOT after H2, H3, and H4 compute the coordinates. You can exit H8 with FCTN 4. The formulae excuted by H2, H3 and H4 are:

x=r cos Q + W and y=r sin Q + H

You all remember that from your high school Trigonometry classes don't you? These days they don't teach kids about trigonometry, after all, you don't need to know that to sell insurance, or used cars, so they have to make it up to attend most colleges.

Execute HOT (as in, Some Like It) and write down the numbers of the patterns you would like to see again. That is what GALAXY does. it expects a number on the stack, then performs the galaxy that HOT did when is displayed that number.

What kind of patterns are they? Well, they vary, I have never seen one exactly duplicated, and I have watched pattern numbers up into the hundreds. I don't really pretend to understand exactly what is happening. Galaxy #6 is what I set out to program originally, but the thing got out of hand. I didn't really give it much thought before I started, I was thinking degrees, but the computer was thinking radians. These are related by an infinitely non-repeating decimal number called PI.

You will discover the the numbers 11, 33, 55, 77, 99, etc tend to be related. Also the numbers 22 (the weirdest), 44, 66, and 88 produce another series. These all have the common factor 11, but that doesn't help much toward understanding, just enjoy.

The basic principle of the algorithm is simple, I am just using a loop and computing coordinates XX YY at the

end of a radius vector, using SIN and COS, at which a DOT is plotted. The vector is rotated each time by incrementing the angle Q and the length of the radius R1. The values W and H are the X and Y coordinates of the center, and you can contol these (see the menu for CIRC) before executing GALAXY. however, HOT resets them using LOC to the middle of the CRT.

On Screen #67, we see how x and y coordinates may be entered from the keyboard using words X and Y. Two new VARIABLEs RSQ and SQT are defined to hold the values of R squared, and a square root, to be used in computing x=(R^2-y^2)^(-1/2), which can be used to plot a circle. I wanted to see if the square root aproach was faster than the SIN COS approach. You be the judge. All the computation is done by ARC, which computes two points on each side of the vertical axis of a circle. The JOT series just uses ARC in two loops to draw circles on the CRT, they may be a little confusing, but not mysterious if you study them.

Screen #68 is a repeat of #66 except that I now use LINE to connect pairs of DOTs. This often changes the appearance of the figure compared to plain dots. To make this simpler, I had to invent 2ROVER, which saves the point at the end of the current line and pushes it down into the stack as the 5th and 6th numbers which will become the top when the LINE is drawn. To do 2ROVER, I had to invent 2SWAP, both of these are on my Stack Manipulator #33 screen of BFORTH #8, OVER OVER is the same as 2DUP. One new thing here is that instead of H1, I have G1 which changes the values in DCOLOR and SCREEN color. G2, G3 and G4 are the same as the H version. FIG, like HOT, lets you see all the patterns, and DOODLE shows any specific number that you put on the stack. Note that if you decide to GALAXY and DOODLE often, they have abbreviated to G and D. Forth is nice that way.

If you have any questions about how all this works, please don't hesitate to ask me. Next time I am going to show how we can speed all this up by substituting approximate values for the FPA stuff.

CU next time; May the Forth be with U!

```
SCR #66
 0  ( DOT GRAFIK ER 4 85 )  F6 IT : IT ; 0 VARIABLE Q
 1    >F 2 VARIABLE R1 6 ALLOT 0 VARIABLE XX 0 VARIABLE YY
 2    128 VARIABLE W  90 VARIABLE H  0 VARIABLE Z CLS
 3    5 12 AT ." LOADING, AIN'T WAITING FUN ? "
 4  : LOC 128 W ! 90 H ! ;  : MENU> TEXT 69 LOAD QUIT ;
 5  : E1 8 9 AT  ." TO EXIT ENTER <FCTN 4> " 1 WAIT ;
 6  : QUIT? ?TERMINAL IF TEXT 0 Z ! MENU> ENDIF ;
 7  : H1 GRAPHICS2 -1 DCOLOR ! 0 DMODE ! 1 SCREEN ;
 8  : H2 R1 F@ >F .4 F+ >F 1.01 F*  R1 F! ;
 9  : H3 R1 F@ Q @ S->F COS F* F->S W @ + XX ! ;
10  : H4 R1 F@ Q @ S->F SIN F* F->S H @ + YY ! ;
11  : H5 H2 H3 H4 XX @ YY @ DOT ;
12  : H7 TEXT 10 7 AT ." DOT GALAXY NO. " Z @ . E1 ;
13  : H8 H1 128 Z @ * 0 DO I Q ! H5 QUIT? Z @ +LOOP ;
14  : HOT LOC >F 2 R1 F! 1 Z +! H7 H8 1 WAIT MYSELF ;
15  : GALAXY ( Z --) >F 2 R1 F! Z ! H7 H8 MENU> 0 Z ! ; -->

SCR #67
 0  ( EXPERIMENT IN DOT GRAPHICS 4 85 )
 1   0 VARIABLE RSQ  0 VARIABLE SQT 6 ALLOT
 2  : X W ! ; : Y H ! ;
 3  : ARC  R1 @ DUP 1 + SWAP 1 MINUS * DO RSQ @ I I * - S->F
 4   SQR SQT F! SQT F@ W @ S->F F+ F->S I H @ + DOT QUIT?
 5   SQT F@ MINUS  W @ S->F F+ F->S I H @ + DOT LOOP ;
 6  : CIRC  DUP R1 ! DUP * RSQ ! H1 ARC MENU> ;
 7  : JOT1 35 W ! ;
 8  : JOT2 60 H ! ;
 9  : JOT3 15 DUP R1 ! DUP  * RSQ ! ;
10  : JOT4 W @ 25 + W ! ;  : JOT5 H @ 25 + H ! ;
11  : JOT6 3 1 DO JOT5 ARC QUIT? LOOP ;
12  : JOT7 7 1 DO JOT2 JOT4 JOT6 LOOP ;
13  : JOT  CLS E1 H1 JOT1 JOT3 JOT7 MENU> ;
14    CLS 12 12 AT ." ALMOST DONE!"
15  -->

SCR #68
 0  ( SPIRAL GRAPHIC ER 9 27 85 )
 1  : D3 CLS 12 7 AT ." DOODLE NO. " Z @ . E1 ;  HEX
 2  : 2SWAP   ROT >R ROT R> ;
 3  : 2ROVER ( n1 n2 n3 n4 -- n3 n4 n1 n2 n3 n4 )
 4     OVER OVER >R >R 2SWAP R> R> ;
 5  : G1 GRAPHICS2 F0 DCOLOR ! 0 DMODE ! 4 SCREEN ; DEC
 6  : G2 R1 F@ >F .3 F+ >F 1.03 F* R1 F! ;
 7  : G3 R1 F@ Q @ S->F COS F* F->S W @ + XX ! ;
 8  : G4 R1 F@ Q @ S->F SIN F* F->S H @ + YY ! ;
 9  : G5 G2 G3 G4 XX @ YY @ QUIT? ;
10  : G6 G1 G5 OVER OVER DOT ;
11  : G7 G6 G2 Z @ * 0 DO G5 2ROVER
12       I Q ! LINE QUIT? Z @ +LOOP ;
13  : FIG  LOC >F 5 R1 F! 1 Z +! D3 G7 1 WAIT MYSELF ;
14  : DOODLE >F 5 R1 F! Z ! D3 G7 0 Z ! MENU> ;
15  : D DOODLE ;  : G GALAXY ;   -->

SCR # 69
 0  ( DOT GRAPHICS  INSTRUCTIONS) CLS
 1  3 3 AT ."        TO DRAW A CIRCLE  "
 2  3 4 AT ."    OF RADIUS R CENTER @ X,Y "
 3  3 5 AT ."   DEFAULT IS AT CENTER SCREEN "
 4  3 7 AT ." ELSE, ENTER x COORDINATE THEN X  "
 5  3 8 AT ."       ENTER y COORDINATE THEN Y "
 6  3 9 AT ."       THEN PRESS <ENTER>. "
 7  3 11 AT ."  NEXT ENTER THE CIRCLE RADIUS "
 8  3 12 AT ."     THEN <CIRC> <ENTER> "
 9  3 15 AT ." ELSE JUST FOR FUN TRY <JOT>, or  "
10  3 16 AT ." <HOT>,<n GALAXY>,<FIG>,<n DOODLE> "
11  3 17 AT ." THEN <ENTER>. "
12
13
14
15
```

FOUR-A/TALK

Random ramblings
about things TI.

by Bill Gaskill

April 1989

MYARC'S HARD DISK SYSTEM

Because I am a died-in-the-wool 99/4A loyalist (though my wife has another name for it) I took the plunge in September 1988 and purchased the Myarc Hard and Floppy Disk Controller (HFDC) card. I also purchased a Seagate 125-1 20 megabyte hard drive and power supply from Myarc to go with it. Total price was $735. A lot of bucks for a 99/4A computer peripheral you say? Yes, but read on before you pass judgement.

If you are like me you want your computer to be productive right out of the box. None of this having to be an electronics engineer to assemble it first, for me. I want components that say NO ASSEMBLY REQUIRED. I am simply not a hardware hacker and I am more than willing to pay a few extra dollars for the convenience and time and frustration savings realized in buying "ready-to-run" components.

When I decided to investigate the HFDC I called Myarc at their Georgia office and talked to Jack Riley. I expressed my concerns about "turn-key" components to him and was assured that Myarc could provide me with a system that required a minimal amount of assembly. Thus convinced, I placed the order and anxiously awaited its arrival. About three weeks later I called back and inquired about the status of the order. I talked to Jack again and he informed me that I would receive the power supply from Myarc-Georgia and the controller card and hard disk from Myarc-New Jersey within the next 4-5 days. I didn't listen close enough to what he said. If I had, I might have asked how a "turn-key" system could be delivered from two different locations, especially when the power supply would come from one place and the hard disk that goes in the power supply would come from another shipping point.

True to Jack's prediction, the hard disk and controller card arrived four days later, the power supply a day after that. When I opened the box containing the power supply I discovered that it was an off-white, steel cased enclosure measuring about 6 inches wide by 18 inches deep and 4 inches high. A handsome piece of equipment that made a nice looking addition to my workstation. When I looked for the instructions on what to do with it there were none. None meaning NOT ONE! A slight trickling of panic seeped into my thoughts. I then opened the box containing the controller card and found a 3-ring binder with about 80 pages of some of the most confusing documentation I have ever read. But not one word of it told me what to do with the power supply nor how it was to be hooked up to the hard drive. So much for "turn-key" systems.

Well I rolled up my sleeves, dug in, and figured out a couple of things without too much hassle. First, there are two power connectors inside the power supply box (white plastic connectors on the end of red wires) and you can use either one, since the unit is built to handle two hard drives. Second, the cover plate I got that fills the blank space for the second hard disk I didn't buy, can be installed without a degree in mechanical engineering. The electrical cord for the power supply only plugs in one way so I couldn't very well do too much damage there. On to the hard disk.

The Seagate ST125-1 that I ordered from Myarc was represented as a 28 millisecond access time hard disk in my conversation with Jack Riley. I find that difficult to believe since Computer Shopper ads put the ST125 at 40 milliseconds. The ST125 sells for around $269. A 28 millisecond Seagate goes for just under $500. You can draw your own conclusions. Anyway, the drive is a 3.5 inch masterpiece that bolts easily to the base of the power supply. The back of it has two edge connectors for two ribbon cables that I could not find in any of the boxes that were delivered to my house. So I set the entire unit aside and moved on to the controller card.

In an act totally out of character for me, I read the manual before proceeding with the installation of my HFDC. When I was sure that I got all that my tiny brain could get out of the documentation I called Myarc-Georgia for help. I just couldn't make enough sense out of what was being said to risk damaging my $735 investment. It turned out that the phone call was about the first really correct thing that I had done (Jack Riley to the rescue). I immediately asked about the missing cables and was told to unfold the two panels in the HFDC box that kept the 3-ring binder and card from moving around. Low and behold, two ribbon cables, one 20 pin and one 34 pin. Boy did I feel stupid. From that point on, I decided to invest in the cost of whatever number of telephone calls it took to get Jack to walk me through the set up of the whole system. That was the second correct thing I did.

The installation went fairly smooth once I had a guiding voice on the phone to keep me from making any mistakes. In fact, the entire set up is rather painless once you go through it about two times, which I did. I don't know what went wrong the first time, but things got so scrambled on the hard disk that nothing would work right. Jack Riley offered the opinion that the cause was not turning everything on and off simultaneously with a master switch. I'm not totally convinced of that though since I have been able to turn the hard disk on last and off first and have had no problems. But I suggest that you stay with what the manufacturer tells you to do, so that your warranty isn't jeopardized.

There are some things about the set up, particularly the formatting of the hard disk, that I don't think I could ever have figured out on my own. When the FORMAT command is accessed from the Myarc Disk Manager 5 you must respond to a series of questions like volume name, sectors per track, number of heads, cylinders, write pre-compensation, interlace and the like. It all sounded pretty foreign to me. But with my guide on the other end

of the telephone line I tackled the job as any brave soldier would who was facing the enemy's ground troops from behind an M60 tank. No problem!

In looking back, I again realize the correctness of making the telephone call to Myarc. Because the MDM5 disk manager program has gone through several updates and the documentation has not been brought totally up to date with the program, the manual tells you things that are no longer correct or that are incomplete or are in some way different than what you are staring at on your monitor. Had I not had the benefit of Jack Riley's immediate tutledge over the phone I would probably still be wondering how to get the HFDC and hard drive working. I must fault Myarc for that. There is no excuse for not providing the most correct and up to date information in the manual. The documentation appears to be about two years old as far as I can tell and it gives such erroneous information as "the cables (connecting the hard disk to the HFDC) can only be put on one way. Wrong! The red strip on the edge of both of my cables had to be at the bottom when installed on the HFDC and facing to the right when attached to the power supply. Physically, they can be installed on the edge connectors either way, but only one way is correct. The manual really needs a going over to bring it up to speed with the current requirements of the set up process.

Back to the formatting. John Kolean wrote two nice articles in the September '88 issue of MICROpendium on the HFDC and its use with a hard drive. Some of the information helped me but not enough of it to get the system up and running. Here is what I did with my hard drive formatting. Maybe it will help you.

VOLUME NAME: 1 (The Volume Name is just like the disk name that you give a floppy disk. WDS1. already appears on the screen. You merely type in a name after the WDS1. to name the ROOT directory of the hard drive. If you name it the number 1 (the actual number 1) you will be able to read and write to your hard drive from programs that will not support pathing, which is most of the programs I own. I guess few software authors ever believed that we would ever get out of the DSK arena. Too bad. More on this later.

SECTORS PER TRACK: 32
HEADS: 4
CYLINDERS: 615
RESERVED DIR/FILE SECTORS: 2048
WRITE PRECOMPENSATION: 0 (erase the default then enter zero).
REDUCED WRITE CURRENT: 0 (erase the default and enter zero).
SECTOR VERIFICATION RETRIES: 8
INTERLACE: 2
STEP RATE: 0

These settings may not be valid for your drive if you have a different hard disk or an earlier or later version of MDM5. My copy of MDM5 is V1.25. I noticed that in the MICROpendium articles mentioned above John Kolean has V1.23 and the prompts he lists in the article are different than the ones that appear on my FORMAT screen. Versions 1.26, 1.27 and V1.28 are pretty much the same as 1.25, especially as far as what you are prompted for when formatting the hard drive. When I last talked to Jack Riley (March 13th) V1.29 had just or was about to be released.

When you have completed entering the above values type in the word FORMAT to format the hard disk. While the process of formatting is taking place MDM5 will dislay a bunch of information on the screen to give you something to stare at while it takes the six minutes or so to get the job done. If you begin noticing more than 1 or 2 BAD SECTORS appearing during the format you might have a problem with some kind of electronic garbage hindering the process. This happened to me so I tried reformatting again as John Kolean suggested in his article, but the problem only worsened. The formatting really slowed down and it seemed as if every sixth sector or so showed up as being bad. I KNEW that couldn't be right so I finally powered down the entire system and let it sit for about 10 minutes to clear it of any garbage and to let it cool down. That seemed to work as I was able to reformat the disk successfully on the next attempt, with no bad sectors.

Once the hard disk is formatted you can leave it as it is or you can create subdirectories on it. Either way, the system is usable once the formatting is complete. With my system I chose to build subdirectories to take advantage of the DSK1 emulation available with the hard drive. I also put a copy of MDM5 on my disk for lightening quick access. The configuration listed below has worked flawlessly thus far. If you can't get yours to work feel free to try mine if you like. I give no guarantees though, as I know only enough about this piece of hardware at this point to be truly dangerous.

Subdirectories:

WDS1. (the root directory)

DSK.
TIMP. (this is a nested subdirectory of the DSK subdirectory. It contains the files from my Multiplan disk: MPBASE, MPCHAR, MPDATA, MPHLP, MPINTR, OVERLAY).

DSK1. (this subdirectory contains the MDM, MDM5 and MDMXB files from the Myarc Disk Manager).

UTIL. (UTIL has the MDM, MDM-BACKUP, MDM-DELETE, MDM5, MDM6, MDM7 MDM8, MDM9, MDMCBACKUP and MDMXB files in it).

The process of making the subdirectories and copying the files to the various areas is fare for another article, but I will say that using the MDM5 program was and is a most enjoyable experience. It is truly a neat piece of software that is pretty easy to understand as it is. Not much is needed in the way of docs. Wonder who wrote it?

I still have a long way to go before I completely understand the workings and capabilities of my new piece of hardware, but I am absolutely thrilled to have it as

an addition to my workstation. I now have what I will boldly claim to be the largest 99/4A data base in existence. 6515 records (2448 sectors) in a single file. I certainly could never have done that on any other storage media available for the TI.

THIS MONTH IN 4A HISTORY:

1983:

-On April 25, 1983 Texas Instruments began offering a free Peripheral Expansion Box to anyone purchasing any three of the following; an RS232 card, a disk controller card, a disk drive, a 32K memory expansion card, a p-Code card, TI-Writer or Multiplan.

-Navarone Industries introduced the "Select-a-Cart" module expander, more commonly known as the Widget and finally named the Cartridge Expander.

-Model Masters of Fullerton, California, now doing business as DataBiotics, introduced JOYPRINT, a low-cost but ill-fated RS232 printer hookup that operated out of the joystick port.

-Tom Wynne, a Seattle area 99er who created the MOLIST utility for XB programmers and then gave it away in Barry Traver's Genial Traveler, made it into the 99er Hall of Fame with a score of 1,009,600 points in Tombstone City.

1984:

-Compute!'s First Book of TI Games made it debut in Compute! Magazine.

-Mikel Laboratories released their RS232 standalone unit for the 99/4A and at the same time announced plans to build a peripheral expansion box for our computer.

-Joseph Nocera's now classic "Death of a Computer" article on the demise of the 99/4A appeared in the Texas Monthly magazine.

-Texas Instruments still had 14% of the home computer market, despite the bailout five months earlier. Commodore had 37%.

1985:

-INTELPRO released a French version of their COMPANION word processor.

1986:

Unisource Electronics of Lubbock, Texas, a major TI-99/4A retailer and mail order house since March of 1983, goes out of business.

-PILOT 99 author Thomas Weithofer dies at age 22.

-Amerisoft, a Georgia based supplier of TI products, and a supporter from the early days of the 4A, leaves the TI market.

1987:

-Monty Schmidt, author of the Techie BBS program, releases Command DOS for the 99/4A.

-The Asgard Helpline, run by former International Users Group librarian Dr. Guy Steffen-Romano, receives an award from the Front Range 99ers of Colorado Springs, Colorado, for continued support of the TI community.

1988:

-Bob Lawson 1344 Boston Ave. Bayshore, New York 11706, announces that the Forti Music Card, formerly offered by the now defunct Unisource Electronics, is still available through him.

TRIVIA:

Did you know that...

-LA99ers User Group member forever George F. Steffen wrote the 99/4A version of Beanstalk Adventure and that he also wrote so of the assembly software for Craig Miller's Gram Kracker conversion utilities?

-Funware, the Richardson, Texas company started by an ex-TI employee that produced Ambulance, Henhouse, Rabbit Trail etcetera, also planned to introduce another cartridge game called Galactic Barrier, but never did?

-Moonbean Software of Northampton, Massachusetts, long-time supporter of the 4A community with a stable of game programs, used to be called HI-FI Exchange?

-TI community personalities Cheryl Whitelaw, Steven Shaw and Tom Wynne all made it into the 99er Hall of Fame with their game scores in Munchman (178,950), Pinball (10,028,010) and Tombstone City (1,009,600) respectively?

-The Turbo Pasc 99 project marketed by L.L. Conner Enterprise, was originally undertaken by Barry Boone under the support of Steve Lamberti of Texaments? Wonder what happened? Texaments even advertised it in MICROpendium a couple years back. Must have taken too long to convert it all over from the German language that it was originally written in.

-Lou Phillips (who?) was once a VP at the Chase Manhattan Bank in New York?

-Windows99 from CSI Design Group in St. Louis, Mo. was written by Ken Dibble (who?)?

Next month; an in-depth look at Barry Traver's Genial Traveler Diskazine and what you are missing if you are not a subscriber, more 4A history and of course... more trivia!

Until then....

Count the Words by Alan Whiteman - LA 99ers

My 99/4A is often used with TI-Writer to produce book reports, essays and the like for school assignments. Usually the assignments set some specific number of words and invariably I get the question "how many words is this, daddy?" I simply guessed at the average number of words on a line or a page, since TI-Writer does not have a feature to count words. So, I thought, "there ought to be a program."

The Extended Basic program which follows is set up for the DV80 file created by TI-Writer, and it counts by assuming that each word is followed by a space (ASC 32) or the end of the line. The main loop uses the POS function to locate spaces, calculates an effective length for each word and stores the number of words in an array by word length.

It isn't really necessary to count words separately by length to get the total number, but this enables counting only those words greater than say three letters - often a requirement for school assignments. Besides, this makes it easy to produce the distribution by length which is also not needed but does make it somewhat more of a programming novelty.

The program does not look for individual characters or punctuation marks, so the word lengths are sometimes off by one. Also, using special characters to underline, for example, will create artificially long looking "words." I arbitrarily cut the maximum length to 20 characters which does not really affect the intended application.

The program is designed to read the files as created by the Editor, so Line 190 checks for Formatter instructions with a leading period (".") and Blank lines created by the Carriage Return (Character 13), provided they are in colu@n 1, and those lines are deleted in both the line count and the word count. The program also reads the Formatter output, assuming this has been written as a disk file, but it will take much longer to count with the many blank lines and spaces added in formatting. To see how much additional work is done with a formatted file, modify the program to display W(0), or enter PRINT W(0) after the program is stopped. W(0) counts words of zero length (spaces) and in the formatted file there are often more spaces than words.

Admittedly the program in Extended Basic is slow. In fact Line 250 is a placebo displaying a line counter just to let you know that something is really happening; similar to watching sector numbers when initializing a disk! A version using an assembly subroutine runs much faster and can more easily correct the shortcomings of the basic program. This will be the subject of another article.

There is probably a program out there somewhere that already does all this and more. However, I needed a word count program one weekend and the challenge was to create one and see it work.

(Ed. Note: The XB Program has been reformatted to 56 col]

```
100 ! WORDB Text File Word Counter
110 DIM W(20):: ON WARNING NEXT
120 GOTO 130 :: CALL KEY :: CALL CLEAR :: A$,F$,Y$,I,J,K
,L,M,N,S,Y,Z :: !@P-
130 DISPLAY AT(2,1)ERASE ALL:"TEXT FILE WORD COUNTER": :
"FILENAME? DSK1.": :"PRESS ENTER TO STOP"
140 FOR I=0 TO 20 :: W(I)=0 :: NEXT I
150 ACCEPT AT(4,14)BEEP SIZE(-12):F$ :: IF LEN(F$)<3 THE
N 470
160 OPEN #1:"DSK"&F$,INPUT
170 DISPLAY AT(8,1):"PROCESSING LINE" :: N=0
180 LINPUT #1:A$ :: IF EOF(1)THEN 270
190 IF ASC(A$)=13 OR ASC(A$)=46 THEN 180
200 N=N+1 :: L=LEN(A$):: Z=1
210 Y=POS(A$," ",Z):: IF Y=0 THEN Y=L+1
220 J=Y-Z :: IF J>20 THEN J=20
230 W(J)=W(J)+1
240 IF Y<L THEN Z=Y+1 :: GOTO 210
250 DISPLAY AT(8,20):N
260 GOTO 180
270 DISPLAY AT(12,1):"MINIMUM WORD SIZE?(1-9):":"ENTER 0
 TO EXIT"
280 DISPLAY AT(16,1):"WORD SIZE >=":"WORD COUNT"
290 ACCEPT AT(12,25)BEEP SIZE(1)VALIDATE(DIGIT):M
300 IF M=0 THEN 350
310 S=0 :: FOR I=M TO 20 :: S=S+W(I):: NEXT I
320 DISPLAY AT(16,14):USING "##":M
330 DISPLAY AT(17,12):USING "#####":S
340 GOTO 290
350 DISPLAY AT(23,1):"WANT THE DISTRIBUTION? Y"
360 ACCEPT AT(23,24)BEEP SIZE(-1)VALIDATE("YN"):Y$
370 IF Y$="N" THEN 450
380 DISPLAY AT(1,3)ERASE ALL:"SIZE -WORD- COUNT"
390 S=0
400 FOR I=1 TO 20
410 S=S+W(I)
420 DISPLAY AT(I+1,4):USING "##          ####":I,W(I)
430 NEXT I
440 DISPLAY AT(23,1):USING "TOTAL ""WORDS"" ####":S
450 DISPLAY AT(24,1):"PRESS ANY KEY WHEN READY" :: CLOSE
 #1
460 CALL KEY(0,K,S):: IF S=0 THEN 460 ELSE 130
470 CALL CLEAR :: STOP
480 END :: !@P+
```

Quick - Count the Words by Alan Whiteman - LA 99ers

The Word Counter described in a previous article was written in Extended Basic and is predictably slow. To speed things up an assembly language subroutine replaces the main loop of the XBasic program. The source code, which follows, is liberally annotated and probably needs little further explanation.

An XBasic program is still used to read each string from the text file using the LINPUT function. Each string is then passed via a CALL LINK to the assembly routine STRING. Where the XBasic program used the POS function to locate spaces, the assembly routine compares each byte of the string with the space character. At the same time it checks for non-alphabetic characters (ASCII less than "A") and adjusts the pointers to obtain true word lengths. As before, the words are counted and stored by word length in an array.

As each byte of the string is read it is little more code to add the Basic Bias and display the string. This is not needed but gives one something to look at as the program is running. Also, the routine checks for blank lines or format instructions by the first character (CR,LF,".") in each the line - though not necessarily in column 1; the blank lines, format lines and total lines are counted.

After the file is read, a CALL LINK to the COUNT routine passes the counts for each word length back to the Basic program for display. Actually, much of the code in COUNT is the subroutine CIF which is needed to convert the integer values produced in assembly into the floating point format used by XBasic. For some reason XBasic does not load the resident CIF routine so I had to include one. CIF only handles positive integers up to 9999, but this is enough for most applications since each counter is for a particular word length and the total file could easily be 40000 or more words - more than the buffer for TI-Writer and far more than most school assignments.

Like the Extended Basic program described previously, this program is designed to run with output from the editor. It also runs with a formatted output file but this takes much longer since the formatter adds many lines and spaces. The final program display includes the number of blank lines and spaces, so you can compare the results for both the editor and formatter files, as well as comparing run times.

At the recent FEST WEST in San Diego, Barry Traver described the use of the ALSAVE routines which appeared in the Genial Traveler (Vol 1-3). ALSAVE is a neat utility for loading assembly code with the the host XBasic program and it really speeds up the process. Our thanks to Barry and to creator Todd Kaplan for this utility which cuts load time from about 6 seconds to something under 1 second.

The "latest" version (a program is rarely final!) of the word count program is thus called FASTCOUNT - a little pretentious, but it does move along at about 5300 words a minute - faster than I can type, or read!

[Ed.Note: The XB program has been reformatted to 56 col]

```
1 ! "FASTCOUNT" Text File Word Counter
2 ! Assembly Routines Loaded via ALSAVE
10 CALL INIT :: CALL LOAD(8196,63,248):: CALL LOAD(16376
,65,32,32,32,32,32,255,48):: CALL LINK("A")
100 ON WARNING NEXT :: F$="DSK1."
110 GOTO 120 :: DIM W(34):: CALL KEY :: CALL CLEAR :: A$
,I,S :: !@P-
120 F$=SEG$(F$,1,5):: DISPLAY AT(2,4)ERASE ALL:"TEXT FIL
E WORD COUNTER": : :"Filename? ";F$: :"PRESS ENTER TO ST
OP"
130 ACCEPT AT(5,14)BEEP SIZE(-12):F$ :: IF LEN(F$)<3 THE
N 420 ELSE F$="DSK"&F$
140 OPEN #1:F$,INPUT
150 DISPLAY AT(6,1): : :"Processed Lines"
160 LINPUT #1:A$ :: IF EOF(1)THEN 180
170 CALL LINK("STRING",A$):: GOTO 160
180 CLOSE #1 :: CALL LINK("COUNT",W())
190 DISPLAY AT(10,1): : : :"PRESS ANY KEY WHEN READY"
200 CALL KEY(0,I,S):: IF S=0 THEN 200
210 DISPLAY AT(3,8)ERASE ALL:"FILE ";F$: : :"WORDCOUNT v
s MINIMUM SIZE"
220 DISPLAY AT(10,1):"Minimum Word Size?(1-9):1":"ENTER
0 TO EXIT"
230 DISPLAY AT(16,1):"WORD SIZE >=":"WORD COUNT"
240 ACCEPT AT(10,25)BEEP SIZE(-1)VALIDATE(DIGIT):I :: IF
 I=0 THEN 270
250 DISPLAY AT(16,14):USING "###":I :: DISPLAY AT(17,12)
:USING "#####":W(I+24)
260 GOTO 240
270 DISPLAY AT(24,1):"WANT THE DISTRIBUTION? Y"
280 ACCEPT AT(24,24)BEEP SIZE(-1)VALIDATE("YN"):A$
290 IF A$="N" THEN 400
300 DISPLAY AT(1,7)ERASE ALL:"FILE DESCRIPTION"
310 DISPLAY AT(3,3):USING "####  Text Lines":W(21)
320 DISPLAY AT(4,3):USING "####  Blank Lines":W(22)
330 DISPLAY AT(5,3):USING "####  Format Lines":W(23)
340 DISPLAY AT(6,3):USING "####  Total Lines":W(24)
350 DISPLAY AT(7,3):USING "####  Total Words":W(25)
360 DISPLAY AT(8,3):USING "####  Multiple Spaces":W(0)
370 DISPLAY AT(10,4):"WORDCOUNT DISTRIBUTION"
380 DISPLAY AT(12,1):"Size - Count    Size - Count"
390 FOR I=1 TO 10 :: DISPLAY AT(I+12,3):USING "##   ####
      ##   ####":I,W(I),I+10,W(I+10):: NEXT I
400 DISPLAY AT(24,1):" PRESS SPACE BAR OR (BACK)"
410 CALL KEY(0,I,S):: IF S=0 THEN 410 ELSE IF I=15 THEN
210 ELSE IF I=32 THEN 120 ELSE 410
420 CALL CLEAR :: STOP
430 END :: !@P+
```

```
*       TEXT FILE WORD COUNTER
RTN     BSS 2                   *Return Address
NOLET   BSS 2                   *First Letter Flag
FLAG    DATA 0                  *First Time Flag
LF      DATA >0A00              *Line Feed (10)
CR      DATA >0D00              *Carriage Return (13)
SPACE   DATA >2000              *ASCII 32 - Space
DOT     DATA >2E00              *Format Instruction (Period = 46)
ALPHA   DATA >4000              *ASCII 64 - Low Limit for Alphabet
EXPO    DATA >5E00              *ASCII 94 - Formatter Reserved Space
BIAS    DATA >6000              *BASIC Screen Bias
BLANK   DATA >8000              *Space with Screen Bias
MAX     DATA >5200              *Maximum String Length
NUMBER  DATA WS2                *Workspace Pointer for Number Routine
        DATA STARTN             *Address for Number Routine
FAC     EQU >834A               *Floating Point Accumulator
STAT    EQU >837C               *Status Byte
NUMASG  EQU >2000               *Number Routine to BASIC
STRREF  EQU >2014               *String Routine from BASIC
VMBW    EQU >2024               *Write Routine to Screen (VDP)
GPLWS   EQU >83E0               *GPL Workspace
WS1     BSS >20                 *My Workspace
WS2     BSS >20                 *Workspace for Number Crunching
XSTR    BSS >52                 *String Buffer (3 Rows)
ARR     BSS >30                 *24 Word/Line Counters
BRR     BSS >14                 *10 Total Word Counters
CRR     BSS >4                  *Buffer for Screen Numbers
*
        DEF STRING,COUNT        *Entry Points for CALL LINK
*
STRING  MOV 11,@RTN             *Save Return Address to BASIC
        LWPI WS1                *Load Workspace
        MOV @FLAG,@FLAG
        JNE PASS1               *Test Flag for First Time Entry
        INC @FLAG               *Set First Time Flag
        LI 1,>46
SWEEP   DECT 1
        CLR @ARR(1)             *Zero Out ARR & BRR Counters First Time
        JNE SWEEP
*
PASS1   CLR 0
        LI 1,1                  *Point to First Parameter
        LI 2,XSTR               *String Buffer Address
        MOVB @MAX,*2            *Set Maximum String Length
        BLWP @STRREF            *Get String from BASIC
        MOVB *2,@WS1+1          *Save String Length in R0
        CLR 6                   *Temp Store for Bytes
        CLR 7                   *Temp Store for Space Count
        CLR @NOLET              *Reset Till First Letter
        INC @BRR                *Collect Total Line Count
        BLWP @NUMBER            *Routine to Display Line Numbers
*
        LI 4,1                  *Set Prior Position Pointer
        CLR 5
LOOP    INC 5                   *Set Current Position Pointer
        C 5,0                   *Test for End of String
        JGT LAST
        MOVB @XSTR(5),6         *Get a Byte
        MOV @NOLET,@NOLET       *Seen Any Letters Yet?
        JNE LINE
        BL @FIRST               *Scan XSTR for CR, LF, DOT
*
LINE    AB @BIAS,@XSTR(5)       *Rewrite Byte with BASIC Bias
        CB 6,@EXPO              *Is Byte a Reserved Space?
        JEQ PASS2
        CB 6,@ALPHA             *Is Byte A or Higher?
        JGT LOOP
        CB 6,@SPACE             *Is Byte a Space?
        JEQ PASS2
        INC 4                   *Move Pointer If None Of These
        JMP LOOP                *Get Another Byte
PASS2   BL @ADD                 *Do Some Arithmetic First
        JMP LOOP
LAST    BL @ADD                 *The Last Word on a Line
*
CYCLE   MOVB @BLANK,@XSTR(5)    *Blank Rest of the String Buffer
        INC 5
        CI 5,>52
        JLT CYCLE
*
        LI 0,>122               *Set Up for the Screen Display
        LI 1,XSTR+1
        LI 2,27
        BLWP @VMBW              *Write Row 10
        AI 0,>20
        AI 1,27
        BLWP @VMBW              *Write Row 11
        AI 0,>20
        AI 1,27
        BLWP @VMBW              *Write Row 12
*
EXIT    CLR @STAT               *Clear for Take Off
        LWPI GPLWS              *Give GPL Back Its Space
        MOV @RTN,11
        RT                      *Back to Basics!
*
FIRST   LI 9,ARR                *ARR Start Address
        CB 6,@DOT
```

```
       JNE NEXT1
       INC @>2E(9)          *Count Format Instruction Lines
       B @EXIT              *No More Processing
NEXT1  CB 6,@CR
       JNE NEXT2
       INC @>2C(9)          *Count Blank Lines
       B @EXIT
NEXT2  CB 6,@LF
       JNE NEXT3
       INC @>2C(9)          *Count Blank Lines
       B @EXIT
NEXT3  CB 6,@SPACE
       JNE NEXT4
       INC 7                *Count Spaces (Temporarily)
       AB @BIAS,@XSTR(5)    *In Case They're Needed for Screen
       B @LOOP              *Look for More Spaces
NEXT4  INC @NOLET           *We Found Something - Perhaps a Letter
       A 7,*9               *Move Spaces Found into ARR(0)
       INC @>2A(9)          *Real Line Count
       RT
*
ADD    MOV 5,6              *Get Current Pointer
       S 4,6                *Less Prior Pointer = Length
       CI 6,21              *Word Length Less Than 21?
       JLT PASS3
       LI 6,20              *Set to a Maximum of 20
PASS3  A 6,6                *Set Pointer to Proper Counter
       INC @ARR(6)          *Add to the Word Count
       MOV 5,4              *Reset Prior Position Pointer
       INC 4                 for Next Byte
       RT                   *Done for Now
*
STARTN LI 0,>F3             *VDP Row 8, Column 19
       LI 1,CRR+4           *Buffer Address Pointer
       CLR 2                *Counter for Digits in Line Number
       LI 3,10
       MOV @BRR,9           *Fetch Line Number
STARTD DEC 1                *Point to Next CRR Position
       INC 2                *Four Digit Numbers
       CLR 8
       DIV 3,8              *Divide Line Number by 10
       AI 9,>90             *Add >30 and BASIC Bias to Display Digit
       SWPB 9
       MOVB 9,*1            *Move Remainder to CRR Position
       MOV 8,9
       JNE STARTD           *Return If More Digits
       BLWP @VMBW           *Write to Screen
       RTWP
*
*
COUNT  MOV 11,@RTN          *Just Like Before
       LWPI WS1
       LI 9,40
       CLR 10               *Accumulator for Word Totals
TOT1   A @ARR(9),10         *Collect Word Totals by Length
       DECT 9
       CI 9,18
       JGT TOT1
TOT2   A @ARR(9),10
       MOV 10,@BRR(9)       *Distribution Totals 1-9
       DECT 9
       JNE TOT2
       LI 0,>22             *Array Element Pointer (ARR & BRR)
       LI 1,1               *First (Only) Parameter
       LI 2,>44             *Buffer Pointer
ARRAY  DEC 0
       DECT 2
       MOV @ARR(2),@FAC     *Put Integer Into FAC
       BL @CIF              *Fetch Back Floating Point
       BLWP @NUMASG         *Tell BASIC the Number
       MOV 0,0
       JNE ARRAY            *Do All Elements (0-30)
       CLR @FLAG            *Reset Array Next Time
       B @EXIT
*
CIF    MOV @FAC,3           *Get Integer
       JEQ DONE             *Exit if Integer is Zero
       LI 6,>4041           *Set Floating Point Exponent
       LI 10,FAC            *Point to FAC
       CLR *10+             *Clear Out First Two Bytes
       CLR *10+             *Clear Out the Rest of FAC
       CLR *10+
       CLR *10
       CI 3,100             *Check Integer Less Than 100
       JLT SMALL
       SWPB 6               *Set Exponent for 100-9999
       MOV 3,4              *Set Integer in R3 and R4
       CLR 3
       LI 5,100
       DIV 5,3              *Quotient in R3, Remainder R4
       MOVB @WS1+9,@FAC+2   *Write Least Significant Byte
SMALL  MOVB @WS1+7,@FAC+1   *Write Most Significant Byte
       MOVB 6,@FAC          *Write Exponent to FAC
DONE   RT
       END
```

-= THE IBM CONNECTION =-

Last mont we included BATCH files as MACROs. This might be a good time to discuss Batch Files; what are they and how do they work? Should I spend my time learning how to use them?

1. Batch Files are a collection of commands and/or keystrokes that can be executed while you are in the DOS environment. The file can be written using EDLIN, a simple DOS based editor, or any word processors or other Line Editor. It's name must have the extension .BAT (short for BATch). Batch files are one of three types of programs which can be run (in the DOS environment) without having to also type in it's extension. (The other two are .EXE and .COM).

2. When you first start up your system, your IBM will first look for a CONFIG.SYS file...then an AUTOEXEC .BAT file (similar to XBasic's LOAD program). This file is usually used to do all the preliminary things you usually do:

Set print mode
Set and display Date and time

But the Batch file is much more powerful than to be used for simple SET-UP work. Let's assume you are using your computer for WordProcessor ing, a Spread Sheet, some programming, and also some games.

First: create a file called MENU.BAT
While in the DOS environment type:

```
COPY CON MENU.BAT
echo off   (keeps commands from be
            ing printed to screen)
cls        (clears the screen)
echo          MY MENU
echo:      (prints a blank line)
echo  1 ... Word Processor
echo  2 ... Spread Sheet
echo  3 ... Basic
echo  4 ... Games
echo:
echo  Enter your choice...
```

Now press Ctrl "Z"

Your program will be saved to the disk in the current drive. To display this file anytime, simply enter:

MENU (or TYPE MENU.BAT)

Our next move is to create a Batch file for each of the entries on our menu. Here is a sample...enter:

```
COPY CON 1.BAT
echo off
cls
cd\wp     (ChangeDiretory to WP..this
           assumes your word processor
           is in a directory named WP)
wordstar  (it is not necessary to type
           in the program's extension)
cd\       (changes the directory to
           the default directory)
menu      (displays your original MENU
           program)
Now enter Ctrl "Z" to save the file.
```

Now what do we have? When you first turn on your machine, after doing its set-up business, the screen will be cleared and your little menu will appear. If you choose #1 (the Word processor), the 1.BAT file will take over, clear the screen, change to the proper directory, and RUN your Word Processor program. AND...when you leave the processor (using the program's QUIT command, the 1.BAT (still in control) will return you to the default directory and display your menu...ready for your next choice!

Neat huh? Practice on 1.BAT until it works properly for you, then repeat the process for each of the other choices. If you haven't figured it out by next month, I'll give you a sample of my GAME MENU (how's that for a clue?).

By the way, if your computer doesn't the date setting routine in it's path you will want to include it in your AUTOEXEC.BAT file. Exam.

```
echo off
cls
setdate
menu
Ctrl "Z"
```

out of coffee,
see you next month

- - C H I C K - -

Did you know that...?

by Chick De Marti 1988

TI-Writer TIP

Ever want to make an entire line or paragraph bold or underlined? Get tired of dozens of @'s to boldface a sentence? Here's what to do; type in the line with normal spaces. Put the cursor at the beginning of the sentence. Then go to the Replace String (RS) command and type in the following: / /@/
When the prompt (All,Yes,No,Stop) appears, select "Yes". The @ will be placed before each word.

A Telecomunication Tip

(Found in a recent (grates) issue of PC today)

If you are novice on BBS's, or a newcomer to CompuServe, consider logging in initially at slow as 300 baud as you will be doing a lot of reading and most people cannot read a stream of words whizing by at 240 characters per sec. (1200 baud). Save that speed for downloading software.

~~~~~~~~~~~~~~~~~~

? ? ? ? ?

CALL LOAD(-32187) will get you 0 line number

Can anyone tell me what it can be used for?

This next one I've used...

```
100 CALL INIT
110 CALL LOAD(_31879,0)
120 CALL PEEK(-31879,T)
130 PRINT T/60
140 ! T = approx. a second
```

Use as a timer!
Thanx HUNTER 99er and Joe Wright. (from an "old TI TIMES newsletter).

FROM THE BBSs

Message #2131
From:ED50
Sub:DF80 FILE NAMES
To:all
THIS IS JUST A NOTE TO ANYONE WHO HAS EVER LOADED UP A DF80 FILE AND DIDN'T KNOW THE NAME TO MAKE IT RUN!
If you load up TI Basic from either E/A or MiniMemory, you can do the following 4 steps to find the name:

```
1) CALL INIT
2) CALL LOAD("DSKx.filename")
3) CALL PEEK(16176,A,B,C,D,E,F)
4) PRINT CHR$(A)&CHR$(B)&CHR$(C)&
   CHR$(D)&CHR$(E)&CHR$(F)
```

The name will be printed & if you want
CALL LINK("name")

~~~~~~~~~~~~~~~~~~

FLAG LOVERS CAN GET REVENGE

People who were outraged over an art exhibit that features a U.S. flag on the floor now can get some measure of revenge. They can walk on the likeness of the artist at another exhibit.
The new piece was put together by a junior high school art teacher who was angered by the original by student "Dread"Scott Tyler of the school of the Art Institute of Chicago.
The protest exhibit feature a life-size sketch of Tyler drawn on cloth in the manner of a police outline of a homicide victim. The cloth is on the floor and a U.S. flags hangs above it.
Viewers are encouraged to step on the sketch of Tyler's face while they write comments on "How to Display "Dread" Scott Tyler"

Thanx Daily Breeze newspaper.

-=> HELP NEEDED TO BUILD A <=-
-=> NEWS-LETTER LIBRARY <=-

(Did You Know ... cont.)

WHAT'S NEW IN THE COMPUTER WORLD?

(*) How about a computer that recognizes your voice and responds to, "PRINT". "SAVE", or "READ THAT LAST PARAGRAPH TO ME"?
(*) How would you like your typewriter to take voice dictation as well as type the letter for you?
(*) A speech module commodore and Apple are working on one), that can speak. MIMICING YOUR VOICE!
(*) A program called "CALM"(for IBM) can help lower your blood pressure by using bio-feedback and a device strapped to your wrist and a pinky monitor. It also helps you to lose weight by using relaxing therapy and sublimemal messages.
(*) And of course ROBOTS...they are here already!

~~~~~~~~~~~~~~~~~~

-= RECYCLE PAPER =-
(by Mike Imel, Indiana FamilyCompute)

How many times has your printer fed through an extra sheet of blank (or almost blank) fanfold paper that you must throw away? Rather than discard this costly perforated paper save it for reuse as printing scrap. Leave the feed holes attached and store. When your ready to use it, tape the pages into a single continuous sheet. They are still usable for testing printer effects or for listing of programs.

~~~~~~~~~~~~~~~~~~

OFFICE IN A LIMO

If money is no object, you can travel in style with Donald Trump. the real-estate tycoon has lent his name to a new line of streches by Cadillac. The Trump Limousine Golden Series costs $80,000 and is the ultimate highpowered, movable office the businessman who must travel from one meeting to another. It features:

(Limo continued...)

Italian Leather seat,
Rosewood veneer cabinets and desk top,
A cellular phone with two hand sets...
(one for his secretary!)
An intercom-system,
A FAX machine, outlets for a Computer,
a Stereo, television, and VCR
(all remote controled),
A hidden safe, and YES, Mr North,
(a paper shredder!)
Made on order only, 38 were sold in '88.
That number is expected to double in 89.
P.S. Mr. Trump got the first one.

~~~~~~~~~~~~~~~~~~

Remember my "CLOSE ENOUGH" program?
Here's a "Probability formula" for long term decisions.

The problem: How many of each of these cars shall I buy for sale in 1989?

Volvo . . .sells best during a bullish economy.
Toyota. . .sells best during a weak economy.
Chevie. . .sells best during a stable economy.
A matrix might be set up thus:

	A	B	C	D	E
1		Estimated Unit Sales:			
2		Weak	Stable	Strong	Project
3					
4	Volvo	500	800	1200	790
5	Toyota	850	900	800	865
6	Chevie	100	1000	450	620
7					
8	Economic Forecast				
9					
10	Weak	30%			
11	Stable	50%			
12	Strong	20%			

The formula for each would be:

B4 * B10 + C4 * B11 + D4 * B12

I'm out of coffee,
see you next month

- - C H I C K - -
~~~~~~~~~~~~~~~~~~

DID YOU KNOW — BONUS PAGE

LATE BREAKING NEWS

I just got to read the December issue of T>I>R>U>G>, the newsletter of the TI Riverside Group and found these items of interest:

(*) Steve Menr and Roger Merritt's Jiffy Flier and Form Shop were first and second place in number of sales for the Chicago (last August) Faire.

(*) Also at the Faire was a blind programmer named Irwin Hott. "He runs special hardware and software to be able to hear all characters and words ... and to know just what is on the screen", the article reported. However while working on his TI at home, Irwin does not use a monitor!

~~~~~~~~~~~~~~~~~

More: GOVERNMENT BBS NUMBERS

Dept. of JUSTICE Washington, DC
Immigration & Naturalization Service

Budget & Finance
data: (202) 786-3640
voice: no voice line indicated
Sysop: not listed
public access: Do not use mid-day

* * * * * * * * *

NOTICE: Home Office Computing reveals "You can still purchase a tuner for about $100, that will convert your monitor to a combination monitor and T.V." (see Dec '88 issue pg 20)

Seen on the GSBUG BBS (in Torrance)

(*) "Woman are meant to be loved, not understood"
(*) "In waking a tiger, use a long stick".
(*) "Clean mind, Clean body: take your pick".
(*) "Gravity doesn't exist, Earth sucks".

JUMP TO in TIWriter

From John Owens, JUG-TX (Jan.1989)
"FAST CURSOR MOVES"

1. Jump to first line of text:
   Fctn-9 "S"(Showline) <Enter>
   0 or 1 <Enter>

2. Jump to End of Text:
   Fctn-9 "S"(Showline) <Enter>
   "E"(End of text) <Enter>

4. Jump to 1st usage of a word:
   Fctn-9 "SF"(FindString) <Enter>
   (string) Exam. "END"

5. Jump to left margin:
   Ctrl-V (Left margin, same line)

6. Jump to right margin:
   Ctrl-I (or Fctn-7)

10. Delete to end of the line:
    Ctrl-k

-= EDITING SPEED-UPS =-

3. DUP LINE:
   Ctrl-5 (handy for setting up vertical lines in a table).

4. CASE CHANGE:
   Ctrl . (period) to lower case.
   Ctrl ; (semi-colon) upper case.

OOPS: Ctrl 1 (Reinstate a line deleted in error).
Fctn 0 (Toggles line numbers... ON or OFF).

XBasic #6 by Jim Swedlow:
MORE IF THEN

Suppose the C$ can only be "Y" or "N" and you want to change C$ without knowing it's value. You could write:
100 IF C$="Y" THE C$="N" ELSE C$="Y"

You don't need to use IF THEN:

100 C$=CHR$(167-ASC(C$))
~~~~~~~~~~~~~~~~~

LIBRARIAN FRED MOORE 7730 EMERSON AVE. LOS ANGELES, CA 213-670-4293

Disks are $2.00 **each** not programs. Many programs takes more than one disk. If you have a **SSSD** drive be sure you get all the disks needed to run the program usally **both A** and **B** disk if the program is over 360 sectors (if available). That comes to $2.00 **each** other wise get the DSSD disk. It pays to have a DSSD drive. And don't forget to include postage if you want it mailed $0.25 for each disk.

0000 **LA99/4A DISKS LIBRARY CATALOG** MARCH 89 : $1.00 either DSSD(699) or 0000A(343) and 0000B(349) for SSSD systems. Special offering.

NEW ADDS FOR APRIL LA99/4A LIBRARY

The Library Committee wish to give thanks to those who donated disks to our Library this month: Ray Green, Earl Raguse, Chick De Marti, Houston Users Group (H.U.G.)

2234 **TI-DOS V6.0** Fairware by Michael Connell uses STAR and FAST-TERM : This set of routines has many of the features of other MS-DOS systems DSSD(408)

2458 **TI-WRITER V4.0** Fairware by R.A.Green: TI-WRITER on disk with many improvements. Faster actions, "No out of memory", printer controls plus many more. SSSD(221)

2459 **STAR NX-1000R** Color Printer Set. By Jerrie Steinberg. Choose and set printer color. 1.Black 2.Red 3.Blue 4.Violet 5.Yellow 6.Orange 7.Green SSSD(14)

2460 **40 COLOMN** By Richard Miutillo : Program includes demo on scrolling and true 40 column screen. Can be used with X/B, E/A or M/M instruction included. SSSD(232)

2461 **FORTH FONT MAIL** Fairware by Howard Arnold : This program will print out mailing labels and disk mailers. Lets you design your own graphic letters. E/A #3. SSSD(360)

2462 **PRINTER PACK** From H.U.G. : X/B-PRINTER A disk full of useful printer programs: ADDRESS LABELS, BILLINGB STATEMENT, LETERHEAD, SLEEVE PRINTER, ELECTRIC TYPEWRITER, ONE LINE UP, GOTHIC PRINT, DISK CATALOG/COMMENTS, JOLLY ROGER LABEL, BANNER MAKER, GEMINI SETUP. SSSD(236)

2463 **TIW-UTILITY** By Ewell Brigham : A disk that has many programs that will be helpful to those who use TI-WRITER. Comes with several formatters, multi-column program, and other utilities. SSSD(307)

2676 **COPY-KAT** By TI Users Group Jackson,MS :Will copy to initialized disk in single or multiple drives, has error checking. SSSD(196)

4545 **MISC #31** By Earl Raguse : An excellent disk of abour 60 XB files. Printers commands, Bar graph, tones, upside down, inverse video, cursor, music, labels, math, catalog, files transfers, 2 column, graphics to mame a few. SSSD(360)

4546 **MISC #32** By Chick De Marti : 7 Programs on disk from the LA99 TOPICS Newletter March 1989. ANIMATION 1,2,3, CAT, CLOSENUFF, CTRL/MENU and PATHDAY. SSSD(56)

4547 **MISC #33** 13 programs from H.U.G : SING A LONG B(computer speak you sing), PANO THE TRAIN B(story)), DISK INDEXER XB(uses call keys), SPACE RIDE B(graphic), PROGRAM LISTER XB(uses disk indexer), LOG ON FILE GENERATOR (for BBS), HEX TO DECIMAL CONVERTER B, BASKETBALL STATS XB-TAPE-PRINT(teams and players has files), TOTAL CATALOG XB-PRT(displays time), GEMINI PRINTER DEMO B, EPSON/TI PRINTER DEMO B, REM REMOVER XB(removes REM and ! from program) SSSD(342)

4548 **MISC #34** 19 programs from H.U.G. : ELECTRIC TYPEWRITER XB(line by line), TEXT TO PROGRAM XB(converts DV/80 file to program), CARTRIDGE DUMPXB, CARTRIDGE TO DISK TIW(docs for Cartridge dump), DV/80 TO PROGRAM XB, , GEMIMI CONTROL XB/PRT(set 22 codes), GOTHIC XB-PRT(large letters), DIGITAL CLOCK XB-32K(time on top left), ORBIT SIMULATION XB(design and orbit missile), PHONE TO NAME B-PRT(and vice versa), WORD WRAP XB(auto wrap XB), PROGRAM CHECKER XB(check two programs), ASSEM TO XB XB(convert assembly to XB), DISK LABELS B(cat on 1"), SLEEVE AND CATALOG XB-PRT, SPEECH FUN B-TEII-SPEECH, SOUND EFFECT XB(short routines), PROGRAM AID XB-PRT, SSSD(342).

4549 **MISC #35** 14 Programs from H.U.G.: X-LATE XB(text to program), FORTY COLUMN TEXT XB-32K, DISK CAT E/A(load and run), BBS MAIL XB(prepare message off line), DISK CAT X/B, BEAM HEADER B(displays MUF between two points), MCS OP AIDE B(amateur radio), CW KEYBOARD B-FILE(1)(morse code), MINIMUF B(Amateur radio), DISK LABEL MAKER B-PRT, JOYSTICK TESTER XB, BASEBALL STATS XB-32K-PRT, AUTO DIALER XB-32K-RS232-MODEM, SOURCE TIME XB-PRT (keep time on "Source"). SSSD(341)

4550 **MISC #36** 17 programs from H.U.G. : DISK CAT XB(screen only), TRI CAT XB-PRT(3 disk catalog side by side), FILE FIXER B(converts INT/FIX 196 to DIS/VAR 80), COMPACTOR XB(make merge smaller), DISK TITLE LABEL XB-PRT(print title labels), BYE CUBE XB(Rubit cube), INITIALIZER E/A(in any format), DISK FIX TIW(repair blown disks), PRO WRITER B(styles), ELI THE ANALYST B-TEII(? and answers), ADDRESS BOOK XB-PRT(for mailing labels), MASTER CASSETTE B-2CASS(keep files on tape), NEW-GEN XB-PRT(create menu, TI KEYBOARD LETTERHEAD TIW-PRT(your Logo at top), ITOD XB-PRT(multiplan Sylink), DB CONVERSION XB(sounnd to DB), RAFFLE B(random drawing). SSSD(336)

4551 **MISC #37** 16 programs from H.U.G. :IF DV80 FILES TIW(31 files for Gemine or SG-10), TI WRITER CODES TIW-PRT(control codes for Gemini/Epson), BURGLAR ALARM XB(useing 99/4A), DISK COPY XB(single drive 2 pass), READER XB-PRT((40 col DV/80), PRINTER INITIALATION XB(save setting to disk), DEFINITION TABLE XB-E/A(names for E/A #3 program), CATALOG XB-PRT. SSSD(349)

4552 **MISC #38** 8 programs from H.U.G. : XTRANFER XB(mass tranfers ffiles), MULTIPLAN TUTORIAL TIW-PRT, FLIPPY CAT XB-PRT(cat 2 disk), SYSTEX XB-PRT(machine language to XB), ACCOUNTANT XB-PRT(balance checkbook), JACKET E/A5-PRT(disk sleeves), BERT AND ERNIE XB-SPEECH(Sesame street), AUTO DIAL XB(Fast Term, P-Term, TE-Plus). SSSD(357)

4553 **MISC #39** 41 Assembly subroutines files from New Horizen Users Group. This disk really shows you the capability of your system. To mame a few files BIKINI, DEFTABLE, DISASSEM, DSRLINK, DUMP, E/ADIS, JUSTFY, LOGO, MMDIS, PLOT, QUICKSRC, RECSCR, SCREEN, SCROLL, SORT, STAT, TIME, VDP, VENUS. DSSD(693)

4554 **MISC #40** 6 programs from H.U.G. : DOCU-GRAM XB-PRT(turns DV/80 into formatted docs), FREZZER TIMP-PRT(inventory frezzer), PRINTER XB-PRT(preset Gemini printer), LABELER XB-PRT(prints out labels/docs), FILE-ALL XB-PRT(name and address), DATA BASE MANAGER XB-PRT(sort and create address lists). SSSD(354)

4555 **MISC #41** 7 programs from H.U.G. :AGGIE TIW-PRT(letter mon to son), MINI DATA MANAGER XB-PRT(has all the funtions of larger DBM), DCOPY E/A5(converts files on disk to if/128 file for modem tran), SESSIONS XB(blink files for transmiting to "The Source"), VFILES XB-PRT(disk catalog), TV TEST XB(adjust TV screen), VCR GUIDE XB-PRT(catalog your video tapes). SSSD(358)

4556 **MISC #42** 8 programs from H.U.G. : ACOUSTIC DESIGN XB-PRT(4176 speaker), BBS40/FT XB(mail preparation for Fast-Term), BBS FILER XB-PRT(create BBS directory), DISK LIST XB(catalog in muti-colors), COMSORT XB(variouyus sorts routines), CHANGE TO 0 XB(add line 0), PRINT XB-PRT(multi-column,display), BUG FIXER EA/3(an dissasembler). SSSD(327)

4557 **MISC #43** 8 programs from H.U.G. : MEMORY MAP TIW-PRT(USES PF COMMAND IN TI-Writer), SOLUTION TIW-PRT(Return to Pirate's Isle), CAR CHART TIW-PRT(mainnntennance chart), VOLKSMODEM DIALER (require Fast Term), TRACK HACK TI XB(copier for TI), TRACK HACK CC XB(copier for CorComp, S CHART E/A3(loader for Super Cart or Gramkracker), DM-1000 for GRAMKRACKER. SSSD(311)

4558 **MISC 44** 11 Programs from H.U.G.: SECTOR TRANSFER E/A3(transfers forth programs), SUPER COPY E/A5(disk duplicator), DV/80 ONLY XB(read TIW files), C99 QUICK REFERENCE CHART TIW-PRT(easy access), MULTIPLAN HOME BUDGET TIMP(with docs), RETURN ADDRESS XB-PRT(print 2 address on 1x3 label), SUPER SAVE E/A3(convert E/A3 files to E/A5 format), GPL DISASSEMBLER E/A5-PRT(P-N to start), UTILITY 1 SORT E/A5-PRT(sort TIW files and save to disk), LABLER XB-PRT(labels up to 6" long), DISK INFORMATION MANAGER E/A5(initialize, catalog, look at sectors). SSSD(348)

4559 **MISC 45** 9 programs from H.U.G.: NEAR LETTER QUALITY XB-PRT(for set Gemimi 10X and 12X), VECTOR BASE XB-PRT(maintain mailing lists, inventories, etc.), VECTOR CALC XB-PRT(a spreadsheet payroll, bills etc.), CORPCOMP TRIPLE TECH CARD (an utility), CDP SCREEN B-XB-PRT(prints out for the 32 x 24 grid screen), READ 64 E/A5(a 64 column reader), MENO CALENDAR XB-PRT(remind you of dates and events need dedicated disk). SSSD(352).

4560 **MISC #46** 4 programs from H.U.G.: DM1002 XB(file reader and disk manager upper and lower case letters), TID E/A5(a mass tranfer for Fast), FAST TERM MOD XB(modified to be used with TID), MEMO CALENDAR XB-PRT(remind you of events, need dedicated disk). SSSD(323)

5046 **SPELLING** By Cliff Donohoo : Teaches spelling ans builds vocabulary. Input your own words. Spelling lessons, flash spell, phase speech, test and more. SSSD(261)

5047 **CHEMESTRY** By John Sewell teaches many of the fundamentals of chemistry. Subjects covered are elementry chemistry, molecules, distill, gas-laws, gas-motion, titration and conversion.. SSSD(238)

6054 **SNOOPY CALENDAR** By J.B. Olivier :XB-PRINTER Prints out month calendar with or without Snoopy picture any year 1800 up DSSD(672)