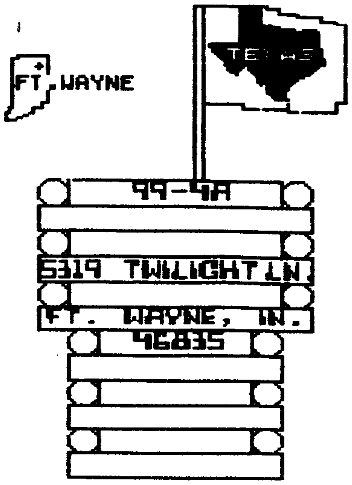


THE FORT'S USER GROUP

President D.Bradtmueller-483-0896	Secretary Denny Przybyla - 432-1228
Treasurer Bud Darr - 693-3341	Editor/Publisher Pat Murphy - 485-2623
Vice-president Bud Darr - 693-3341	Librarian Tom Carson - 485-7086

February 1988

PRESIDENT'S COMMENTS - BY DAVE BRADTMUELLER

We had a good turn out again. I am glad to see so many people are renewing there memberships. I have two requests to make. First, if you didn't notice in the last news letter, Pat gave notice that he will be stepping down as editor of this news letter in 3 to 4 months. So far we have not had any volunteer to take over. If when Pat steps down no one has come forward, this letter will stop being printed. So think about it, do you want this newsletter to continue? If so Volunteer!!!!!!!!!!!!!!!

Second, I am running out of ideas for demo's. If I do not get some volunteers soon our meetings will be very short and very dull. It's up to you as to how our meetings will go.

On to bigger and better things. I have been contacted by Dave Szippl (president of the Lima Area TI 99/4a User Group). On Saturday May 21 from 10 am to 6 pm. at the Lima Campus of O.S.U. the Lima group will be sponsoring a user group conference and swap meet. The event is totaly free and will be a good place to get to know all the faces that go with the names that we see in the Ohio newsletters. The club has been offered a table at the conference. This would be at no charge to us. I need your input. Do you think that we should set up a table, and can we get enough volunteers to man it? Let me know if you would be interested in volunteering or if you would like to go. We will be trying to set up car pools if we have enough people.

The next meeting will be on February 13, at 9:30 am. The doors open at 9:00 so try to be there as early as possible. See you all there.

RENEW YOUR MEMBERSHIP NOW!

MINUTES

Meeting of January 09, 1987
Submitted by Dennis B. Przybyla, Secretary

President Dave Bradtmueller called the meeting to order shortly after 9:30 am. There were nine members and one guest in attendance. The Secretary's minutes of the December meeting and the Treasurer's report for January were published in the January Newsletter. The formal presentation of these reports at the meeting were omitted but submitted as published. There were no additions or corrections to the reports of the Secretary and Treasurer and the reports were approved as published.

The election of officers for 1988 were held at this meeting. The slate of nominees selected at the December meeting and published in the January Newsletter were presented to the group for a vote. No additional nominations were presented. The nominees received unanimous approval. Your elected officers for 1988 are Dave Bradtmueller, President; Bud Darr, Vice President and Treasurer; and Dennis Przybyla, Secretary.

Your 1988 renewal membership dues are now due.

It was pointed out that Pat Murphy definitely wants to be relieved from his Editor/Publisher duties. If this group wants a newsletter, someone must step in and continue on with the superb work that Pat has done. Think about it. It could be satisfying, as well as fun!

Tom Carson got the okay to purchase 200 diskettes for the library. These diskettes will also be available for sale to the membership

Again, our group had a very interesting open discussion period, covering a variety of topics. It does pay to attend the meetings! There is always a good exchange of news and ideas.

Following the open discussion period, a demonstration was given by Dave Bradtmueller on Millers Graphics' Advanced Diagnostics, a disk utility program.

The door prize winner at this meeting was Bud Darr, again! He received two diskettes. Delbert Turner, a past member, was a guest at the meeting.

The next meeting will be held on Saturday morning, February 13, at the Shawnee Branch Library. The meeting start time will be at 9:30, with the doors opening at 9:00 am.

$$

The February issue of the Computer Shopper magazine's TI Forum should be of interest to all TI enthusiasts. Included, is Warren Agee's report on the Chicago TI Faire. I spent most of the day at the Faire, and now I realized how much I missed. Also of interest is the half page ad covering Myarc's GENEVE 9640. MYARC is serious about their new computer!

A lot of good articles in the January issue of MICROpendium. A few of the topics in this issue are Regena on Basic, c99, Working with Logo, Rejuvenating troublesome cartridges, a new box, Archivers conference, three articles on the Geneve 9640, reviews on Quik Font and EZ-Keys, plus much more. If you don't subscribe to the MICROpendium, you might consider doing so. It's the best media for keeping up on what's new in the TI-99/4A community.

RENEW YOUR MEMBERSHIP NOW!

$$

Membership/Treasury
by Bud Darr

As Tony Miller mentioned , in the last news letter, it is **SIGN-UP TIME!** He has reported a good number of renewals at the time of his last writing. But, since the last meeting, only one additional has come to my attention. So, if some of you can't always make it to the meetings, but still want to keep your membership in good standing, you can send your "**CHECK**" to me at my home...addrress as noted. Bud Darr, 10 Holiday Park, Churubusco, IN 46723-2221. Please, no cash by mail. Your continued interest in the club is greatly appreciated.

I am sorry to say that at this time, full transition of treasury records have not been made and won't be until Tony Miller returns from an educational leave in Chicago (about two weeks).

As a final word...if you have a friend or have knowledge of a person who is into the **TI-99/4A** and is not upgraded, the club can be a great help in getting the most out of the ole' TI!!!

On to the treasury report for the month of JANUARY:

DEBITS	
OLD BALANCE................................	$393.71
BALANCE....................................	$393.71
CREDITS	
NEW BALANCE................................	$393.71

RENEW YOUR MEMBERSHIP NOW!

$$

>>>EDITOR'S NOTES<<<
by P. Murphy

The first article comes from TIGERCUB Software in Columbus, Ohio. The TIGERCUB has just the best of things to sell you. This article is just one of the many interesting things from the TIGERCUB. If anyone is interested in the very useful items from them...contact TIGERCUB SOFTWARE, 156 Collingwood Ave., Columbus, OH 43213. Now, here is the article...very good...thanx Jim Peterson!

DEBUGGING

by Jim Peterson

When you have finished writing a program, the next thing you should do is to run it. And, very probably, it will crash!

Don't be discouraged. It happens to the very best of programmers, very often.

So, the next thing to do is to debug it. And you are lucky that you are using a computer that helps you to debug better than some that cost ten times as much.

There are really three types of bugs. The first type will prevent the program from running at all - it will crash with an error message. The second type will allow the program to run, but will give the wrong results.

And the third type, which is not really a bug but might be mistaken for one, results from trying to run a perfectly good program with the wrong hardware, or with faulty hardware. As for instance, trying to run a Basic program, which uses character sets 15 and 16, in Extended Basic.

First, let's consider the first type. The smart little TI computer makes three separate checks to be sure your program is correct. First, when you key in a program line and hit the Enter key, it looks to see if there is anything it can't understand - such as a misspelled command or an unmatched quotation mark. If so, it will tell you so, most likely by SYNTAX ERROR, and refuse to accept the line.

Next, when you tell it to RUN the program, it first takes a quick look through the entire program, to find any combination of commands that it will not be able to perform. This is when it may crash with an error message telling you, for instance, that you have a NEXT without a matching FOR, or vice versa.

And finally, while it is actually running and comes to something that it just can't do, it will crash and give you an error message - probably because a variable has been given a value that cannot be used, such as a CALL HCHAR(R,C,32) when R happens to equal 0.

The TI has a wide variety of error messages to tell you when you did something wrong, what you did wrong, and where you did it wrong. But, it can be fooled! For instance, try to enter this program line (note the missing quotation mark).

```
100 PRINT "Program must be s aved in:"merge format."
```

And, sometimes you may be told that you have a STRING-NUMBER MISMATCH when there is no string involved, because the computer has tried to read a garbled statement as a string.

Also, the line number given in the error message is the line where the computer found it impossible to run the program; that line may actually be correct but the variables at that point may contain bad values due to an error in some previous line.

If the error occurs in a program line which consists of several statements, and you cannot spot the error, you may have to break the line into individual single statement lines. This is the easiest way to do that. Be sure the line numbers are sequenced far enough apart. Bring the problem line to the screen, put a ! just before the first ::, and enter it. Bring it back to the screen with FCTN 8, retype the line number 1 higher, use FCTN 1 to delete the first statement and the ! and ::, put a ! before the first ::, and continue. Then, when you have solved the bug, just delete the ! from the original line and delete all the temporary lines.

Pages 212-215 of your Extended Basic manual list almost all the error codes, and almost all the causes of each one - it will pay you to consult these pages rather than guessing what is wrong.

You may create some really bad bugs when you try to modify a program that was written by someone else - especially if you add any new variable names or CALLs to the program. Your new variable might be one that is already being used in the program for something else, perhaps in a subscripted array. I have noticed that programmers rarely use in a variable name, so I always tack it onto the end of any variable that I add to a program.

Also, the program that you are modifying may have ON ERROR routines, or a prescan, already built in. The ON ERROR routine was intended to take care of a different problem than the one you create, so it could lead you far astray - you had better delete that ON ERROR statement until you are through modifying.

The prescan had better be the subject of another lesson, but if the program has an odd-looking command !@P- up near the front somewhere, it has a prescan built in. And if so, if you add a new variable name or use a CALL that isn't in the program, you will get a SYNTAX ERROR even though there is no error. One way to solve this is to insert a line with !@P+ just before the problem line, and another with !@P- right after it.

When a program runs, even though it crashes or is stopped by FCTN 4 or a BREAK, the values assigned by the program to variables up to that point will remain in memory until you RUN again, or make a change to the program, or clear the memory with NEW. This can be very useful. For instance, if the program crashes with BAD VALUE IN 680, and you bring line 680 to the screen and find it reads CALL HCHAR(R,C,CH) just type PRINT R;C;CH and you will get the values of R, C and CH at the time of the crash. You will find that R is less than 1 or more than 24, or C is less than 1 or more than 32, or CH is out of range.

In Extended Basic, you can even enter and run a multi- statement line in immediate mode (that is, without a line number), if no reference is made to a line number. So, you can dump the current contents of an array to the screen by FOR J=1 TO 100::PRINT A(J);: : NEXT J - or you can even open a disk file or a prin- ter to dump it to.

You can also test a program by assigning a value to a variable from the immediate mode. If you BREAK a program, enter A=100 and then enter CON, the program will continue from where it stopped but A will have a value of 100.

You can temporarily stop a program at any time with FCTN 4, of course (the manual says SHIFT C, but it was written for the old 99/4), and restart it from that point with CON. Or you can insert a temporary line at any point, such as 971 BREAK if you want a break after line 970. Or, you can put a line at the beginning of the program listing the line numbers before which you want breaks to occur, such as 1 BREAK 960,970,980 Note that in this case the program breaks just BEFORE those listed line numbers. You can also use BREAK followed by one or more line numbers as a command in the immediate mode.

The problem with using BREAK and CON is that BREAK upsets your screen display format, resets redefined characters and colors to the default, and deletes sprites. So, it is sometimes better to trace the assignment of values to your variables by adding a temporary line to DISPLAY AT their values on some unused part of the screen. If you want to trace them through several statements, it will be better to GOSUB to a DISPLAY AT. And if you need to slow up the resulting display, just add a CALL KEY routine to the subroutine.

Sometimes, your program will appear to be not flowing through the sequence of lines you intended (perhaps because it dropped out of an IF statement to the next line!) and you will want to trace the line number flow. This can be done with TRACE, either as a command from the immediate mode or as a pro- gram statement, which will cause each line number to print to the screen as it is executed. If used as a command, it will trace everything from the beginning of the program, so it is usually better to insert a temporary line with TRACE at the point where you really want to start. Once you have implemented TRACE, the only way to get rid of it is with UNTRACE.

TRACE has its limitations because it can't tell you what is going on within a multi-statement line, and it will certainly mess up any screen display. Sometimes it is better to insert temporary program lines to display line numbers. I use CALL TRACE( ) with the line number between the parentheses, and a subprogram after everything else 30000 SUB TRACE(X)::DISPLAY AT(24,1):X :: SUBEND

Some programmers use ON ERROR combined with CALL ERR as a debugging tool, but I can't tell you much about that because I have never used it. ON ERROR can give more trouble than help if not used very carefully, and I cannot see that CALL ERR gives any information not available by other means.

Sometimes you can debug a line by simply retyping it. It is only very rarely that the computer is actually interpreting a line differently than it appears on the screen, but retyping may result in correcting a typo error that you just could not see. In fact, most bugs turn out to be very simple errors.

When you are debugging a string-handling routine, don't take it for granted that a string is really as it appears on the screen - it may have invisible characters at one or both ends. Try PRINT LEN(M$) to see if it contains more characters than are showing; or PRINT "*"&M$&"*" to see if any blanks appear between the asterisks and the string.

There is no standard way to debug a program. Each problem presents a challenge to figure out what is going wrong, to devise a test to find out what is really happening.

Don't debug by experimenting, by changing variable values just to see what will happen, etc. Even if you succeed, you will not have learned what was wrong so you will not have learned anything - and if your program contains lines that you didn't understand when you wrote them, you will have real problems if you ever try to modify the program. (Believe me, I speak from experience!).

************ PROGRAM WITH PRE-SCAN**************

by PATRICK MURPHY

```
DO NOT BEND
FLOPPY DISK ENCLOSED
DO NOT XRAY
```

Have you ever sent or received a floppy disk through the U.S. Mail? If so, you have probably noticed the warning on the mailing envelope about the enclosed floppy disk and "DO NOT X-RAY"!!! I have included a program that will print this warning out on any EPSON compatible printer (STAR Micronics Too). I have included two versions. The first is as it cames from the author (thanx) and the second has been "Compressed", "Shrunk", and most important for program start up time "Pre-Scanned"!!!

I believe the first printing came from "Computer Shopper". The second, I personaly condensed to show the difference in program length. Had the program been a long program, this saving would be more evident than what appears !! For those of you that do not know Pre-Scan, this program will take any Extended Basic program and re-write it to turn off the SCAN and turn it on when necessary. This SCAN is required with Extended Basic for memory allocation before executing a program. With Pre-Scan by T. J. Hoddie, this feature is controlled to help speed up start time of an Extended Basic program. Excellent program. Look for the !@P- & !@P+ signs in the second program listing. Included in the program are the options to remove REM statements, rename variables and scan sub-programs!! Here they are:

```
100 ! ********************
110 ! *                  *
120 ! *   DON'T LABELS   *
130 ! * BY: Ron Rutledge *
140 ! * Central Iowa UG  *
150 ! *                  *
160 ! ********************
170 !
180 ESC$=CHR$(27)!ESCAPE CODE
190 EMP$=ESC$&"E" ! EMPHASIZED PRINT
200 NOR$=ESC$&"W"&CHR$(0)! NORMAL PRINT
210 ENL$=ESC$&"W"&CHR$(1)! ENLARGED PRINT
220 UON$=ESC$&"-"&CHR$(1)! TURN UNDERLINE ON
230 UOF$=ESC$&"-"&CHR$(0)! TURN UNDERLINE OFF
240 OPEN #1:"PIO"
250 PRINT #1:EMP$&ENL$&UON$&"DO NOT BEND"
260 PRINT #1:NOR$&" FLOPPY DISK ENCLOSED"
270 PRINT #1:ENL$&UOF$&"DO NOT XRAY"
280 PRINT #1: : : :
290 CALL KEY(0,R,S)! ARE YOU PRESSING A KEY?
300 IF S=0 THEN 250 ! IF NOT THEN PRINT ANOTHER
310 CLOSE #1
320 END
```

Now, Pre-Scanned...

```
100 A=0 :: B=1 :: C=2 :: D=3 :: E=4 :: GOTO 110 :: A$,B$,C$,D$,E$,F$ :: ,F,G :: CALL KEY :: !@P-
110 C$=CHR$(27):: A$=C$&"E" :: D$=C$&"W"&CHR$(A):: B$=C$&"W"&CHR$(B)
120 F$=C$&"-"&CHR$(B):: E$=C$&"-"&CHR$(A):: OPEN #B:"PIO"
130 PRINT #B:A$&B$&F$&"DO NOT BEND" :: PRINT #B:D$&" FLOPPY DISK ENCLOSED" :: PRINT #B:B$&E$&"DO NOT XRAY" :: PRINT #B: :
: :
140 CALL KEY(A,F,G):: IF G=A THEN 130 :: !@P+
150 CLOSE #B :: END
```

$$$

From the LA99er's Group comes this news about ITSOFT! This software is put out by Rodger Merritt. Print-It, Rodgers first "IT" software, was described by Peter Hoddie as the "Fairware" version of Font Writer. Picture-It is a companion to Print-It, designed to be used with TI-ARTIST instances. This program features : BANNERS. TI-ARTIST instances are converted for use in your banner. It will also convert your TI-ARTIST instances to view on the screen as character or sprite mode. With character mode, instances can be saved in MERGE format to be used in XB programs! Sprite mode allows Merge format to be used with CALL LOADS!

The great thing that Picture-It does is converts instances to be output through TI-WRITER! Could make great letterheads with this very useful program!!

CATALOGging with this program is also very useful, sorting by file types. With MAX-RLE, almost any graphics art can be converted for use with Picture-It! All this for $10.00!!!! Send $10.00 to:

Rodger Merritt
1948 Evergreen Ave.
Fullerton, CA 92635
(714)-990-4577

I have sent for this fine program and hope to "demo" it at one of the upcomming meetings. It appears that there are still many excellent programs and programmers in the TI community. Thanx to the LA99er's group for this info!

RENEW YOUR MEMBERSHIP NOW!

$$

The following article was forwarded to us by our "GOOD" friends in Lima, Ohio. It was first published by the author Art Byers, in the Jan. 88 issue of CALL SOUNDS, the newsletter of the Central Westchester user group. All programs listed are Public Domain and are available from our library. Thanx to Charles Good and Art Byers.

SUPERXB$V1 - Documentation.

This disk contains a series of TI Extended Basic CALL SUBprograms that attempt to make the TI Extended Basic Module partly compatable with the new commercially sold Super Extended Basic module.

The New Super Extended Basic module will load QUALITYSOFT's Draw 'n Plot routines into low memory. To obtain this to use with the TI XB module, you must buy them from quality soft and preload them into lo memory. Complete documentation and instructions come with the purchase.

The PEEKVdp ram and POKEVdp ram routines are built into the new Super XB module. The only way you can use these CALLs with the regular XB module is to preload assembly language routines, such as those that appeared in the SMART PROGRAMMER or those available by subscribing to the GENIAL TRAVELER. These routines will reside in Lo-memory and be accessed by CALL LINKs. You may have to rewrite the SUB PEEKV and SUB POKEV subprogrrams to conform with the requirements of those assembly programs. Also, they will cannot be used if "Draw 'n Plot" is in lo-memory, as they will over-write those sub programs.

Similarly, the clock programs are in the Super XB module. To simulate this, you will have to preload an assembly language clock routine such as the one that appeared in the August 1984 Issue of the Smart programmer. Again, if the Draw 'n Plot subprograms are in low memory, these will over write them and probably cause lockup.

What is obvious from the above is that you cannot be 100% compatable with a program written for the new Super XB madule if it makes use of all the above calls in the same program. However, most programs do not use all these subprograms. In addition, the SUB programs provided are all very useful and can be used in your own regular XB programs.

The SUB programs are provided in MERGE form, consecutively numbered. MERGE in only those you need for the program you wish to run or are planning to write. See the TI Extended Basic reference Guide for the proper way to use CALL SUB programs.

This disk contains the following CALLs which are 100% compatable with the CALLs built into the new Super XB module:

CALL ALL(numeric expression)

This call fills the screen with the character of the ASCII number in the perenthesis. ie: CALL ALL(32) fills the screen with blanks, just like CALL CLEAR. CALL ALL(36) fills the screen with dollar signs -$-.

CALL BEEP sounds a beep tone.

CALL BYE is the same as the immediate command BYE except that it is used in a program.

CALL CHIMES sounds a single chime. Although it is possible to duplicate the chimes provided in the Editor Assembler Manual, I have chosen to use a chime from one of Jim Peterson's Tips from the TIGERCUB. Jim is the undisputed KING of XB CALL SUB programs, and this was included as a tribute to him. You can buy over 300 usefull CALL subs in merge form on his three Nuts and Bolts disks - and get a real education on how to program in XB.

CALL CLOCK will only work if you have the clock routine preloaded into Low memory.

CALL COLORS(foreground,background). This call sets the foreground and background colors for all Character sets except set 0. Therefore it does not set the border colors. These can be done by the standard CALL SCREEN(). By setting the background color to TRANSPARENT -1- in CALL COLORS, the color called by CALL SCREEN() will become the background color.

CALL GOSPRT starts all sprites moving after they have been stopped with CALL STSPRT.

WARNING: only ONE of the following can be used in any program as the last line of the CALL SUB XXXX(n) absolutely MUST be the last line of the program. Therefore, they all use the same line numbers, ending in 32767, and will overwrite each other: GOSUB(n), GOTO(n), RESTORE(n), RUNPROG("device,filename").

CALL GOSUB(n) allows you to use a variable with GOSUB. However, remember when you RESequence line numbers, (n) will NOT be resequenced. Also remember that a call to a non existing line number will crash your program.

CALL GOTO(n) same as above but is a GOTO instead of a GOSUB.

CALL RESTORE(n) allows the use of a variable instead of a fixed line number with the same cautions as in GOSUB(n).

CALL RUNPROG("device.filename") circumvents TI XB's disallowance of RUN "DSK1."A$" or RUN A$ where A$ might be "DSK3.MYPROGRAM"

CALL HONK sounds a warning honk.

CALL KEYS("keylist",numeric variable) allows validation of the keys listed by either a predefined string (A$) or listed in between the quotes ie:"1234ABC" and returns a numberic variable. If the key A was pressed the variable would be 5 as A is the fith in the sequence shown. If a key other than those defined is pressed, a honk is sounded and the program awaits a correct key press. IMPORTANT: CALL HONK must be MERGED along with SUB KEYS(). I consider this to be among the most usefull of all the subprograms on this disk.

CALL NEW is the same as the immediate command NEW but can be used in a program.

CALL PEEKV(vdp address, value list) and CALL POKEV(vdp address, value list) are explained above.

CALL QUITOFF disables the quit key, FCTN=

CALL QUITON enables the quit key, FCTN=

CALL STSPRT stops all sprite motion. To restart use CALL GOSPRT.

$$ SUB PROGRAMS and documentation written and donated to the public domain by ART BYERS. SYSOP TINET special interest group on DELPHI. 12/16/87. $$$ DV80

RENEW YOUR MEMBERSHIP NOW!

$$

$$ LIBRARY NEWS $$
by Tom Carson

Two new programs from MICROpendium have been added to our library. From the November issue we have the BASIC program SOLITAIRE written by Regena. this program can be addictive. The program feeds you 25 cards - one at a time - which you place on a five by five grid. After all cards have been placed, your horizontal and vertical poker hands are scored. If the combined score is greater than 2,100 you have done well.

The second program comes from the December 1987 issue and is good for making labels for presents. The article titled - Labels with impact - was written by Ed Machonis. The print codes used are for Epson compatible printers. You have a choice of eight different borders. If you wish, you can design a border of your own and substitute one of the origional eight with your own.

We have version 4.0 dated October 1987 of Funnelweb available. This disk contains 716 sectors of programs and documentation. This is a workhorse disk. It contains a TI-Writer, Edit/Assm, DM1000, and many other utilities.

RENEW YOUR MEMBERSHIP NOW!

W.I.S.H. LIST

WISH GAC0686...A terminal emulator program that would transfer a complete disk rather than one file at a time.

WISH TM0686...Would like a ribbon cable connector (female) for a 36 pin .100" card edge connector.

WISH GAC0786...A program that would download different fonts to a dot matrix printer i.e., script, gothic, roman, etc.

WISH JY0886...A program that converts CALL LOAD statements into assembly language source code.

WISH BCD0986..Would like a used expanded system (RS232 Optional), at least one disk drive and 32K memory expansion.

WISH GCC1286..Would like a casette cable and educational programs or modules (primary level).

ANSWER GAC0686...Use FREEWARE program "MASS TRANSFER" available from Stuart Olson, 25322 W. Wayside Place, Lake Villa, IL 60046. Program now in user group library.

ANSWER TM0686...Connectors you want are available from PILGRIM'S PRIDE, 5 Williams Lane, Hatboro, PA 19040

ANSWER JY0886...The program to convert CALL LOAD to ASM. LANGUAGE source or object code has been written by Tom Freeman of LA99ers and is in our library.

ANSWER GAC0886...Program called OLDENG prints any TI-WRITER file in old english letters - also Character Sets and Graphics Design III provides 6 full character sets - can be purchased from TEXAMENTS.

FOR SALE: Used TI System. Contact Tim Wilson on 432-0887 for info on the system and the price.

Improved Video

by, Bob Lawson

In my travels through the Texas Instruments Manuals, specifically the TMS-9918,28,29 Manual, I read,"The load resistor (RL, pin 36 to ground) defines the sharpness of the edges on the video signals. A lower resistor value gives faster fall times and a sharper picture." Hmmm! I don't remember any 330 ohm resistors.

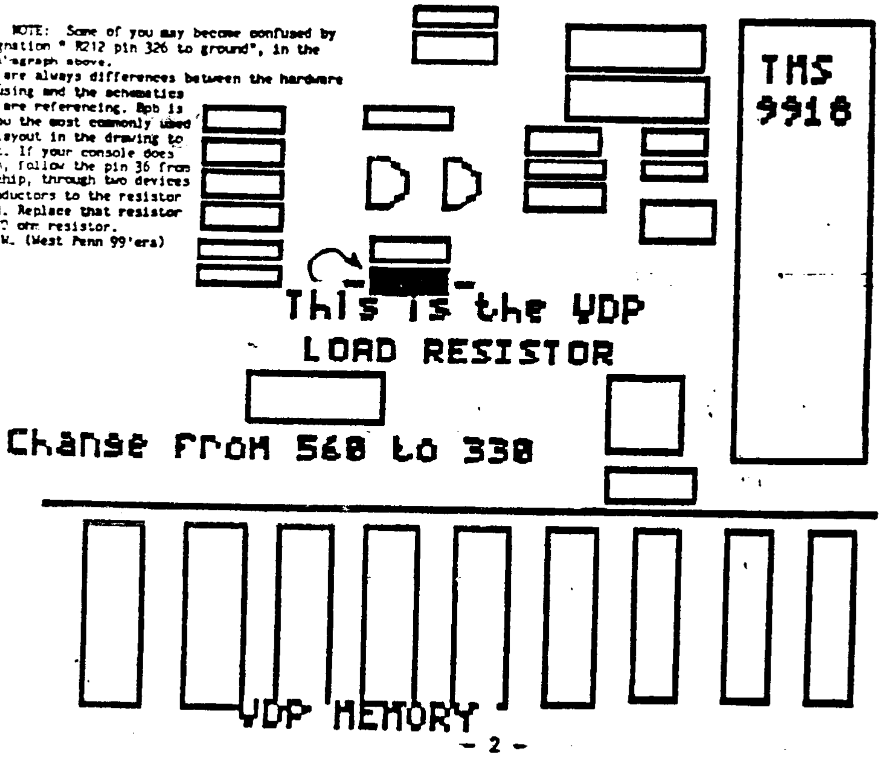
Well, I pulled out the "TI Console and Peripheral Manual, and sure enough, R212 pin 36 to ground was 560 ohms per the schematic. The next step was to check out a console, and well you guessed it, R212 was 560 ohms, not 330 ohms as recommended in the TI Manual!

Next step was to try some different value resistors, 330 ohms seems to be about the best common value resistor to use. I wonder why TI chose to use 560 ohms. I did find one old TI Manual which recommended 390 ohms (1979), but they're sometimes hard to find in 1/4 watt. This 30 cent change gives about a 40%, that's right, I said 40% improvement in the picture. The improvement is so good, you'll wonder where the WHITE SHADOWS WENT.

EDITOR'S NOTE: Some of you may become confused by the designation " R212 pin 326 to ground", in the second paragraph above.

There are always differences between the hardware you are using and the schematics that you are referencing. Bob is giving you the most commonly used console layout in the drawing to the right. If your console does not match, follow the pin 36 from the VDP chip, through two devices called inductors to the resistor to ground. Replace that resistor with a 330 ohm resistor.

J.F.W. (West Penn 99'ers)



CUT OFF AND RETURN TO PAT MURPHY AT THE NEXT MEETING

Wish I Sure Had...

W.I.S.H. (print or type please)

Answer to WISH #______________(# = name and date rec’d...i.e.,GAC0586)

NAME And DATE

____ $15.00 Renewal

____ $15.00 New member

____ $ 7.50 Subscribing Membership

OFFICE USE

____ REC. ____ UNUM

____ PAY ______ DATE

NAME: ______________________

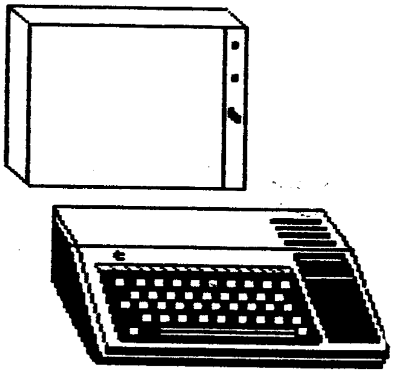
ADDRESS: ______________________

CITY: ______________ STATE: ________

PHONE: (_____) _______-__________ ZIP CODE: ________

INTERESTS: ______________________

The Fort's User Group
5319 Twilight Lane
Ft. Wayne, IN 46835



MIAMI CO. - 99/4A USER GRP
P.O. BOX 1194
PERU IND. 46970