

5712  
2801 (63)



# Newsletter

## December & January

-----

The Winnipeg 99/4 User Group is a non-profit organization formed by computer hobbyists for users of the Texas Instruments 99/4A Home Computer and compatibles. The content of this publication does not necessarily represent the view of the Winnipeg 99/4 User Group.

Next General Meeting - Date : To Be Announced  
Time : 1:00 P.M.  
Place: 1334 Aikins St.

### Executive 1987:

President:	Sheldon Itscovich	633-0835
Treasurer:	Bill Quinn	837-7758
Newsletter Editor:	Darrell Wyatt	339-9742
Contributing Editor:	Marie	339-9742
Contributing Editor:	Rick Lumdsen	253-0794
Publisher:	Peter Gould	889-5505
Publisher:	Sheldon Itscovich	633-0835
Publisher:	Dave Wood	895-7067
Publisher:	Bill Quinn	837-7758

Mailing Address: NEWSLETTER EDITOR  
1334 Aikins St.  
WINNIPEG, MANITOBA  
CANADA, R2V 2C7

- LETTER FROM THE EDITOR -  
by Darrell Wyatt

Merry Christmas everyone! December is here and we all have

visions of ramdisks and programs dancing through our heads...well some of us anyway. This month I have started to make a few small changes to the newsletter such as:

1. Notice that the date is December and January. It has come to my attention that we are publishing the newsletter one month behind so from now on the newsletter received at the end of the month will be dated for the coming up month. Ex. January 31 we will receive the February newsletter.

2. Starting next month there will be a small title page change.

Well anyway we are all starting to add to the new library by pooling our separate libraries together and the reinstating of membership fees (at the new rate of \$20) were the main issues settled last meeting. Since there is no meeting this month due to the holidays I will keep this newsletter brief so once again Merry Christmas and a Happy New Year.

- MISCELLANIA - by Doug Howe

#### USING CALL KEY

by Glenn Davis

(from Computer Shopper, June 1986)

Although everybody uses a key-scan now and then, many people do not realize there are some pretty handy tricks that can be used to make the rest of their programming easier. CALL KEY, as implemented on the TI-99/4A, has six modes to operate in: 0-5. As described in the "User's Reference Guide" the modes are as follows:

0 -Scans in the same mode scanned immediately before it.

1 -Scans left side of the keyboard.

2 -Scans right side of the keyboard.

3 -TI 99/4 mode. This has important applications! TI-Writer uses this mode in the text formatter.

4 -Pascal mode. This is true 7-bit ASCII and any code 0-127 can be generated off the keyboard (and more than 30 others). The 99/4 did not have this mode.

5 -TI BASIC standard mode. This is the mode used by BASIC, and if CALL KEY(0,K,S) is performed initially, it behaves like a CALL KEY(5,K,S).

Modes 1 and 2 are used the same way you would use the joysticks via CALL JOYST. The key-codes for modes 1 and 2 generate codes that are in the "User's Reference Guide" on page III-4. Two people can then operate games on opposite sides of the keyboard without interfering with each other. But since people use modes 1 and 2 for games already, I won't bother with an in-depth discussion.

Mode 3 acted differently on the 99/4. It behaved like a CALL KEY(5,K,S) on the 99/4A. Several character codes do not exist in this mode, however: ASCII 0, 16-31, and 96 and greater are not available in mode 3. ASCII 96 is " ` ", the accent grave. The codes 16-31 are normally ASCII control characters.

Mode 3 has several applications. One of which is accepting a single character from the keyboard, regardless of the case. In this manner, a "Y" (versus a "y") will always be detected when the "Y" key is pressed, whether or not the ALPHA-LOCK is depressed. Many programmers assume (by coding the program in a certain way) that the ALPHA-LOCK is not depressed or prompt the user that it must not be depressed (or vice-versa). Mode 3 allows you to get around that. By specifying CALL KEY(3,K,S) the 99/4A will accept only upper-case, so when the keyboard is scanned, it won't matter if the ALPHA-LOCK is up or down. Amazingly, this also works on INPUT, LINPUT and ACCEPT statements too! A short program can illustrate this.

```
100 CALL KEY(3,K,S)
110 CALL CLEAR
120 INPUT "SEE! ONLY UPPER-CASE : NO lower-case":A$
130 GOTO 120
```

Try the ALPHA-LOCK in both positions and you'll see that it is impossible to get lower-case on the screen by typing off the keyboard! In either case, the tilde ("~") and other special characters cannot be entered. Try inserting LINPUT and ACCEPT in place of INPUT if you have TI Extended BASIC.

Another application for mode 3 appears when dealing with files on the disk system and printer. Programs are written to prompt for "device.filename". "Device" must be in upper-case for the 99/4A to recognize it as a valid device name. "Filename" can be any combination of upper- and lower-case characters. If you have either a Disk Memory System or an RS232 interface, you should know this. "The Disk Memory System" manual recommends that filenames be only upper-case because the Disk Manager (I) Cart-ridge won't display lower-case correctly. In a program where the user enters the filename, just add a CALL KEY(3,K,S) before the INPUT, LINPUT or ACCEPT (in TI Extended BASIC) statements. This allows entry of a correct filename without fumbling with the shift-key or depressing the ALPHA-LOCK. Beware, though, that if the program is retrieving a disk file that is named with lower-case, the disk controller will not find it if the name is entered this way.

Mode 4 has a few applications on an elementary level. All ASCII codes in 7-bit format are enabled (0-127), unlike modes 3 and 5, where 16-31 are disabled. As when switching to mode 3, switching to mode 4 affects the INPUT and ACCEPT statements. Typing in mode 4 will allow data in files to be generated that couldn't be generated in modes 3 or 5 (i.e. with control characters in them that were entered via the keyboard. If the values are entered through the program with the CHR\$ function, no difference will appear). One note of

caution: ENTER and CTRL-M both return the same character code in mode 4 ( which makes code 141, normally returned by CTRL-M, unavailable ). If this is undesirable, you'll have to program around it. I do not know of any TI software that uses this mode, since it is incompatible with the 99/4. Other codes are also generated.

Mode 4 has some other side-effects that may have some application, although I haven't found many yet. One is that in both TI BASIC and TI Extended BASIC the CLEAR key (FCTN 4) will not stop a program on an INPUT or ACCEPT statement. Try this:

```
100 CALL KEY(4,K,S)
110 INPUT "TRY IT":A$
120 GOTO 110
```

Can't break the program? Hmm...Bee, how do you break such a program? Don't turn you 99/4A off. Just let the program sit. The answer comes indirectly from the "User's Reference Guide". It tells us the code for CLEAR in mode 5 (normal BASIC mode) is 2. In mode 4 (Pascal) a 2 code must be produced by some key, so which one? Well, looking to the appendix in the "User's Reference Guide", we see that a CTRL-B will give you a 2 code. Now RUN the above program again. Have you noticed some other unique things.... ERASE (FCTN-3) doesn't work either..... and neither do DELETE and INSERT (FCTN- 1 and -2). Look up the codes for those keys in the " User's Reference Guide ". Try them in the INPUT lines. Neat huh? Try FCTN-S. So, you can't backspace either! Now if any of you are familiar with any other computer systems, main-frames or telecommunications, then you might be able to guess what keystroke will result in a backspace like FCTN-S will: CTRL-H. Try this one too. Try some others and tell the rest of us if you find out something interesting!

If you intend on using mode 4 in conjunction with files on the Disk Mem-ory System or the RS232, remember to push the ALPHA-LOCK down. Otherwise the lower-case might give you a headache.

Mode 5 is the normal BASIC mode, which most people should be familiar with. If not, it was discussed in depth in "Compute!" of November 1983. There are some differences between TI BASIC and TI Extended BASIC as far as applica- tions of mode 5. TI BASIC allows graphic definitions up to ASCII 159, while TI Extended BASIC only allows definitions up to ASCII 143 ( 16 fewer; the memory issued to keep track of sprites ). When PRINTing graphics to the screen instead of the slower CALL HCHAR or CALL VCHAR, no codes past CTRL-0 should be used for TI Extended BASIC. Codes up through CTRL -9 may be used in TI BASIC.

Usually, when CALL KEY is used for routine-jumping, two methods are employ- ed. The first method uses a series of IF -THEN statements to check each condition individually. This procedure is, well, slow. IF-THEN allows the 99/4A to make as few as one scan per second. Depending on how it was coded of course. The sec- ond method normally used is a CALL KEY ON exp GOTO where the command keys are consecutive, as in A-6 or I-9. This method is quite fast, but a programmer must use consecutive keys, which makes the mnemonic

(memory-aiding) value of such keys poor.

But really, just two statements can handle all the branching from the key- scan to subroutines. How can just two lines do that? The secret is in the two statements, often used seperately, but rarely together: POS and ON exp GOTO (or GOSUB). The POS function searches a string for the value returned by CALL KEY, and the ON exp GOTO performs a cal- culated jump, i.e. a jump determined by the value of exp. The string should be defined early in the program (once only to avoid wasting time) using string lit- erals and/or CHR\$ function. The actual characters contained in the string are one greater (ASCII code) than the value required. For example, the "TE" in the string below represents "S" and "D".

```
100 STRING$=CHR$(0)"TE"
110 CALL KEY(3,K,S)
120 ON POS(STRING$(K+1),1)+1 GOTO (ill- egal or
unused key routine),[no key rou-
tine],[ "S"-routine],[ "D"-routine]
```

The full code is presented here:

```
100 CALL KEY(3,K,S)
110 STRING$=CHR$(0)"TE"
120 CALL KEY(0,K,S)
130 ON POS(STRING$,CHR$(K+1),1)+1 GOTO
140,160,180,200
140 PRINT "ILLEGAL KEY"
150 GOTO 120
160 PRINT "NO KEY"
170 GOTO 120
180 PRINT "S KEY"
190 GOTO 120
200 PRINT "D KEY"
210 GOTO 120
```

When you run this program, leave the ALPHA-LOCK off. You'll see why shortly. The character STRING\$ will represent the key-codes retrieved by the KEY subpro- gram. One (1) must be added to this ex- pression because the KEY statement re- turns negative one (-1) when no key is pressed, which cannot be used as an arg- ument for the CHR\$ function. The arg- ument for CHR\$ [K+1] then becomes zero when no key is pressed, and zero is in the first position of STRING + (check the definition in line 110). This makes the routine loop back if no key is pressed to scan the keyboard again. If an illegal ( unused ) key is pressed the POS function will return a zero, mean- ing that a match was not found of the strings. Therefore, one (1) must be add- ed to the expression because the ON exp GOTO cannot use zero as an argument. The line, therefore, becomes as is shown in line 120 above. To add more routines to a program, all that is necessary (beyond the routine's code) is to add another line number to the list in the ON exp GOTO and an appropriate character to the string.

The keyboard can be scanned up to ten scans per second this way, and it catches virtually every key when

waiting for a key to be pressed ( the routine above does not; BASIC is slow at scroll- ing ).

Now, finally, we'll cover mode 0, an unusual mode. In the program above, mode 3 was specified once, followed by call- ing mode 0. This "locks" the computer into mode 3 so all successive mode 0 scans are in mode 3 (refer to the defin- ition of the modes above) until a non- zero mode is called. This is useful in applications programs, for example, an Extended BASIC database program that on- ly uses upper-case characters as command letters. A single CALL KEY(3,K,S) at the beginning of the program will "lock" the entire program into upper-case. Likewise modes 1 and 2 are affected by mode 0 in the same way. Unfortunately, CALL JOYST doesn't recognize mode 0 (it returns a # BAD VALUE error).

This should give programmers of all levels a little more insight into the powerful TI-99/4A.

- HELPFUL HINTS -  
by Rick Luadsen

Well, Christmas is almost upon us and since our meeting this month would conflict with the holiday season, it has been cancelled until January but Darrell and his newsletter staff are still putting together our little paper here to tide us over until the New Year.

Sorry fellows, I didn't get a chance at all to brush up on my Disk Fixer program so that tutorial will still be on hold for now. I am still engrossed in

MS-DOS and it seems to be extremely complicated after the TI. I'm slowly getting a handle on it now, but it is slow going. The IBM world is almost unfriendly compared to the TI community, and the Shareware is more strictly controlled. For example, quite often the programs are sent out with just enough

documentation to let you see the program working but not to utilize the advanced features. I guess it does encourage one to pay and register for the use of the program but I still think I would rather see the WHOLE program in action before I spent my money.

A few excerpts from the latest MicroPendum magazine. The new Myarc Gen'evé is slowly cutting out it's own little niche in the TI world. Some little fixes for

various module based programs that had trouble running are now starting to appear. All who have seen it running have nothing but praise and hopefully in the New Year we will get a first hand report on it. Seems one of our former members was in Chicago for the TI Faire and had a close look at it. He plans on

purchasing the computer in the near future. He also said he was very interested

in re-joining our group as it seems he got lost in the shuffle when we moved from the library.

Rave 99, the company that brought us the Rave keyboard has a new product out. It is a small expansion card for the PEB that allows you to remove your Speech

Synthesizer PCB and install it in the expansion box much the same as the CorComp Triple Tech card. This card however does not have the additional features of the Triple Tech, such as the print buffer and clock, but is a cheaper way to get the SS of the side of your console. It is available from Rave 99 for \$49.95 U.S. at 112 Rambling Road, Vernon CT 06066. John Birdwell's excellent Disk Utilities program is now at Version 4.0A for all those interested. This has to be the best Disk Fixer program available for the TI and

to top it, it is a Shareware product. Another Shareware is one called the Music

Pre-Processor. From the review it sounds quite interesting as it will allow you

to control fade rate and chop of any music you create. For a very limited( initially) sound chip found in the 4/A console it is really starting to come alive in these last few years. The music for the TI now is almost unbelievable compared to what the parent company offered 4-5 years ago, but I guess that is true for any of the software for the 4/A now. Horizon Computer has lowered the price on it's EXCELLENT Ram Disk card. It is now priced at \$35 for a bare PCB, \$45 for the PCB, operating system and docs, or \$195 for a 192K complete card. These prices are all U.S. dollars. I have praised this card here before and I still believe it is the best money you will ever spen on your computer. I have heard wind of a further upgrade to take it beyond the 256K limit, right to the full DSDD capacity of 360K. Sounds TERRIFIC!!!

Barry Traver of Genial TRAVELER fame has been barred from computer and reading activities due to retinal surgery so that will put his Disk based magazine a little behind schedule. I'm sure if anyone has seen his magazine, the wait will

be worth i as it is an excellent publication and I wish Barry a speedy recovery as I am sure the whole TI community does.

That's about it for this month. One last hing, I still have a 32K internal expansion for sale. This is the one that fits inside your console and does not require an expansion box. I'll sell it for \$30 or \$35 installed so give me a call if you are interested. In the meantime, have a Merry Christmas and a Happy Safe New Year.

- TI-CHAT -  
Compiled by Marie!

This month's TI-Chat features some useful little Tips.

Reprinted from the MAD HUG November 1987.

DISABLE "CALL WAITING"  
by Mel Myhre  
downloaded by T.Vacha - TI-CHIPS

I am unable to verify the following info which I found on Comuserve regarding "Call Waiting", because I do not have the service myself. (I prefer to get AS FEM

calls as possible, and sure don't want two callers at once). However, I offer the following by Mel Myhre, to those who have modems and the plague called "Call Waiting". The rest is a direct quote.

"I use the following and it has always worked for me: Preface the number with "1170,". The "1170" tells the phone company computer to turn off the call waing and any one who calls the number will get a busy signal. The comma is a pause for any Hayes compatible modem as to give the phone company computer time to react and stabilize the line. Try "1170" manually and you will hear approximately 3 tone bursts and then a normal dial tone after a slight pause. Call waiting is restored as soon as the present connection/call is terminated by the phone company computer so it must be reaccomplished for every number dialed. The Phone Files I use are all prefaced by "1170,". I travel extensively and it has always worked for me in the US(they don't have such things overseas yet)."

Reprinted from our November 1984 newsletter.

Tired of the plain, blinking square called a cursor? Like to see something in a designer cursor? Perhaps in the shape of the state of Texas? According to the Columbus (Ohio) 99/4A Users Group, you can do this with a series of CALL LOADs. This transformation requires a memory expansion and Extended Basic. Here's the program:

```
1 CALL CLEAR :: CALL INIT
2 CALL LOAD(8196,63,248)
3 CALL LOAD(16376,67,85,82,83,79,82,48,8 )
4 CALL LOAD(12288,48,48,63,255,254,124,2 4,12)
5 CALL LOAD(12296,2,0,3,240,2,1,48,0,2,2
,0,8,4,32,32,36,4,91)
6 CALL LINK("CURSOR") :: END
```

Enter RUN and the program loads the redefined cursor into high memory where it will remain until you QUIT or type BYE. Other Extended Basic programs may be loaded and the cursor will remain in its redefined shape. Line 4 carries the workload here. CALL LOAD(12288,0,0,0,0,0,0,252) results in a flat line cursor. Deleting line 4 results in an invisible cursor.

Reprinted from the HUG newsletter of May 1987.

#### SCREEN ROUTINES

1. Tired of the ssame old CALL CLEAR command to clear the screen? Try these instead.

10 CALL HCHAR(1,1,32,768) - This will clear the screen from top to bottom.

10 CALL VCHAR(1,1,32,768) - Clearing the screen from left to right is what tis one does.

1. For a sharper display on a black and white TV add this line to the beginning of a program. CALL

SCREEN(15) - This will disable the color generating circuit in the computer and remove the vertical lines often seen. It also increases the sharpness of the characters.

#### CASSETTE TIPS

If you accidentally enter OLD CSI when you mean to enter SAVE CSI, do not panic... All is not lost. Type SHIFT "E" and hit ENTER. You will get an I/O error message but the program will still be in memory and you will get a second chance to save.

#### LONG FILE LOADING

(Extended Basic Only) To load and run long files that were originally saved without the memory expansion and disk drives try this:

(perform the following in this order.)

```
CALL FILES(1)
NEW
OLD DSK1.FILENAME
SAVE DSKn.FILENAME//,MERGE
TURN OFF COMPUTER, THEN BACK ON
MERGE DSKn.FILENAME//
SAVE DSK1.FILENAME
```

This will convert the file to one which will be able to access the memory expansion and may be run from a menu or "LOAD" program.

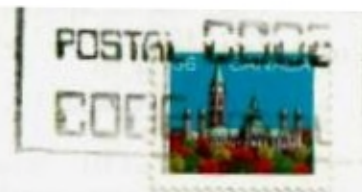
(Steven Foster 8/85)

#### PERSONAL RECORD KEEPING

If you have numerous files on cassette created for the PRK where each page was used and you have tried to load them with the Expansion System turned on giving you an error message when loading, try this. Load the file from cassette, then delete several pages. Then save the file on a different cassette. Turn everything off, then turn on the expansion system and console. Now try to load the new file from cassette. If you deleted enough from the original file, you should now be able to save the file to disk. It seems that when a PRK file is full that it uses all available memory thus making it necessary to delete "Pages" because you cannot do a CALL FILES(1) from the module.

(Scott Medbury 4/87)

Winnipeg 99/4 Ushers Group  
1334 Aikins St.  
Winnipeg Manitoba  
Canada R2V 2C7



Edmonton 99ers  
P.O. Box 11983  
Edmonton Alberta  
T5J 3L1