MSP 99 USER GROUP

THE MSP 99 NEWSLETTER

QUESTIONNAIRE RESULTS

by Joel Gerdeen

The following is a summary of the MSP 99 User Group Survey conducted in the past few months. The questionnaire was included in the October issue of the club newsletter. Of the 280 paid-up members, 66 members or 24 percent returned the questionnaire. At the December meeting, Walt Thompson won the $50 drawing from the returned forms. The information gathered will be invaluable to our officers in planning the direction of the group in 1985. Each member is assured that their specific response will be kept confidential to the group officers.

While the details of the survey are listed below, a few observations are in order. In the following discussion, the term 'members' means those responding and does not imply that such a percentage applies to all club members.

It is not surprising that over 94 percent of members have cassette recorders and joysticks. But it is interesting that over 68 percent have a complete system (PEB, Disk, 32K and RS232) though actually 73 percent have a disk drive. Over 45 percent can be considered serious users that have a modem and editor/assembler. Twenty-seven percent have mini-memory. Another level of skill indication shows that 44 percent consider themselves

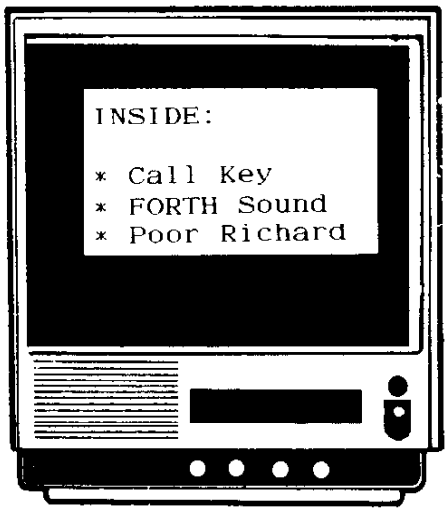
(Continued on page 2)

Noiseless Peripheral Expansion Box

By Bob Hubel

Are you distracted and disturbed by the tornadic roar of your Peripheral Expansion Box? TI engineered the box to provide sufficient cooling capacity for the most strenuous of circumstances -- all 8 card slots occupied and under heavy, continuous usage. Since my use didn't even approach the design limits, I began experimenting with ways to slow the fan down, and I was successful in reducing the noise level down to a barely-perceptible purr! I have been testing this change for a sufficient period of time, and I now feel confortable in recommending this modification to others. In fact, I have even run under light loads for moderate periods of time without any fan at all -- but I don't personally advise going to that extreme.

(Continued on page 4)



The MSP 99 USERS GROUP meets each month for discussions and presentations that enable its members to be better informed about their computers. Users group members share and exchange information. Some members have a broad range of computer expertise; others are just beginning. We are not affiliated with or sponsored by any other group or company. Membership dues are $12 a year for a family, $10 for an individual, and $50 for a sponsor member. You're welcome to visit a meeting before you join. Call or write for more information.

USERS GROUP MEETINGS are held the third Tuesday of each month at 7 p.m. at Dunwoody Industrial Institute, 818 Wayzata Blvd., Minneapolis, MN 55403.

MSP 99 USERS GROUP
P.O. BOX 12351
ST. PAUL, MINNESOTA 55112, U.S.A.

The MSP 99 NEWSLETTER is published eleven times per year, on a monthly basis except during July, by the MSP 99 Users Group. Members are encouraged to contribute articles for publication. Opinions expressed are those of the writer and not necessarily those of the MSP 99 Users Group, its officers, editor, or members. Materials accepted by the editor for publication in the MSP 99 Newsletter, including software listings, are believed to be in the public domain. Newsletter articles may be reproduced by other users groups if appropriate credit is given to the author (if one is listed) and to the Minneapolis-St. Paul 99 Users Group.

Newsletter Editors:
Gary Gese & Mike Kabala 780-8719

DEADLINE FOR NEXT ISSUE:
JANUARY 15

COMMITTEE VOLUNTEERS

If you want to work on committee (Education, Equipment, Program, Publicity, Software, Newsletter), or have an idea for a program, contact one of the officers.

COMMERCIAL
ADVERTISEMENT RATES

Business firms that want to communicate with our members may do so by placing an ad in the newsletter. Rates are: Full Page (7 1/2 X 10 1/2) $40; Half Page (3 1/2 X 10 1/2) $30; Quarter Page (3 1/2 X 5) $22. Each ad must be camera ready in sizes indicated and paid in advance. Inserts (printed by advertiser on 8 1/2 x 11) may be inserted in the newsletter at $20 per sheet. Contact the editor for information or to reserve space.

CHANGE OF ADDRESS

Before you move, please mail a change of address to the group at the above address.

(Continued from page 1 column 1)

proficient programmers in BASIC and XBASIC. Twenty percent feel they are advanced programmers. Note especially in this skill catagory as well as other questions , some members checked multiple catagories (182 percent response).

A high percentage (65 %) indicate that they often attend the monthly meetings and a majority (50% of responses - 77% of attendees) feel the meetings are 'Better than Average'. On 'Personal Applicability' ,members were less satisfied, 51% of attendees indicated the meetings could be better. Members though are happy with the newsletter with 56 percent feeling that it is 'Better than Average' and 29 percent actually saying it is 'Fabulous'.

Members make some use of the software library. Sevety-two percent have gotten at least one program from the library, but only six percent have gotten more than 25 programs. They also spend a fair amount of time using their computer. Sevety-one percent use it more than five hours per week.

Forty three percent of the members have been less than one year while twenty percent have been more than two years.

(Chart on back cover)

Absentee ballots can be found in this issue for those out-of-towners or anyone that can not make it to January's meeting to vote in person. Mail them in before Midnight Jan. 15.

MSP 99 Calendar of Events

JANUARY 15 -- (Tuesday) 7:00-9:00 PM — Election night -- Be there to vote for your favorite candidate. We may be (just) a users group, but these elections are as important to us as the U.S. Presidential election is to the country.

FEBRUARY 19 -- (Tuesday) 7:00-9:00 PM — To be announced (as soon as we find out what our new officers have in mind)

Subgroup Meetings

ASSEMBLY LANGUAGE--First Tuesday of month, 7:00 p.m., Bryant Community Center, Bryant Ave. and 31st St.

BUSINESS--Second Tuesday, 7-9 p.m., Norwest Bank, Hopkins. Call Bob DeMars (544-6219) or Dick Clemetson (926-8083).

EDUCATION--At monthly MSP meetings.

YOUTH GROUP--At monthly MSP meetings.

AVALON HILL--At monthly MSP meetings. Chair: John DiIorio.

Committee Chairs

Equipment--We need someone.
Newsletter--Gary Gese & Mike Kabala 780-8719
Program--Dick Dunbar, 488-0153
Publicity--Dave Wunderlin, 544-8266
Software--Ed Neu, 425-8744

(Continued from page 1 column 2)

I had considered 2 options -- purchase a quieter fan, or just slow down the present fan. A quieter fan costs about $20 or more, so I quickly discarded that option. I could reduce the fan speed with either a special solid - state voltage regulation device or merely install a power resistor in series with the fan. Since I have no experience with such solid-state devices and the components would likely cost around $10, I elected for the latter method at a cost of $0, using components from my electronics junk box. If you had to purchase the components at retail, the cost would amount to no more than $2 - $3.

The PEB fan is rated at 14 Watts. I have found that inserting a 500 - 700 Ohm power resistor (10 Watts power dissipation) reduces the speed to a very acceptible level (700 Ohms is my own preference). (Since a 700 Ohm resistor may be hard to find, you can combine 2 or more resistors which add up to 700 Ohms.) This resistor "steals" the energy that was originally intended for the fan and thus the fan doesn't work as hard. However, in doing so, the resistor must shed this extra energy itself, and does so by producing heat. Therefore, you MUST mount the resistor on the OUTSIDE of the PEB, immediately behind the fan, thus allowing the circulating air to cool it. Do NOT mount it inside the cabinet. Although it may be more aesthetically pleasing, it will add unwanted heat inside the cabinet.

Procedure to disassemble the PEB to access the fan lead wires:

1) Unplug power cord.
2) Remove lid.
3) Remove disk drive (4 screws - 2 on top 2 on bottom; disconnect 2 cables).
4) Remove all slide-in cards.
5) Remove 2 screws on cabinet sides.
6) Remove 6 screws on back (but not the 2 screws holding the lid clips).
7) Remove 5 screws on the perimeter of the base, plus the 1 near the middle.
8) Slide base out.

The next step requires a little bit of ingenuity and personal customization. You must disconnect 1 of the fan leads and extend those 2 wires to the outside back of the box through a drilled hole. Splice the resistor(s) between those 2 wires, then mount them on the box next to the fan exhaust. I can't tell you exactly how to do this for there are many possible ways. If you are interested in looking at my example, give me a call and I can show you how I did it. Be sure that all wires are insulated as this is a 115V circuit. After that is done, reverse the above steps to reassemble the box.

Since the fan air intake is through the card cage, I have cut foam to fit in the empty slots, at the far forward side of the card cage. This then forces the reduced airflow to flow through the existing cards, thereby increasing their cooling.

The resistors will run warm and possibly may be uncomfortable to touch -- this is normal. Don't cheat and use less than 10 Watt resistors as then they may get too hot and cause problems.

If anyone knows of a cheap solid-state equivalent which will perform the same function, please notify me as it would be the preferred method, if inexpensive.

You'll be amazed at the considerable reduction in the noise pollution. My telephone number is (612) 894-1924.

Good Luck!

Poor Richard's Peripheral Roundup

By Dick Dunbar

Rumors and Other Fluff

There is a persistent rumor making the rounds that there will be a successor to the TI99/4A seeing the light of day around February of 1985. This new machine is rumored to be coming from Myarc, Inc. instead of CorComp, who was supposed to be laboring in that direction. In the past, I would have been reluctant to lend any credibility to such rumors, but there have been a few developments lately which make it possible that there might be some shred of truth to it. Myarc is a company which has not exactly been a household word in the micro world, but which did and does build the only (so far as I know) hard disk unit for the TI99. They also came out with an expansion system and double density disk controller and other peripheral items after TI's folding act. And they did this all without a lot of advance fanfare - they didn't talk about it, they just went ahead and did it. Frankly, I don't know whether to believe this rumor or not, but it has my hopes up a bit higher.

TI-Pro Special Deal

This one is no rumor - it's a sure thing. Texas Instruments is making a pitch to current owners of TI99/4As who are "software developers" to sell them TI Professional computers at special reduced prices. Their definition of software developers is someone who has developed software for the TI99 and who has either contributed software to a user exchange library, such as the MSP99 software library or the International 99/4 Users Group (yes, it is still in existence - more about that later), or who has commercially marketed such software. To get the special developer's prices, you must agree to transport your existing TI99/4 software, or develop new software to operate on the TI-Pro, and submit that software to a users group exchange library or market it within 12 months of the purchase of your TI-Pro. For this minor set of considerations, you can get one of these computers for prices as shown below. These prices include a monitor, either monochrome or color as indicated. The 256K units also include a TI 855 printer and cable in the price shown.

Desk Top model:

128K, mono, 1-dsk	- $1622
128K, mono, 2-dsk	- 1882
256K, mono, 2-dsk, prtr	- 1992
256K, color, 2-dsk, prtr	- 2233

Portable model:

128K, mono, 1-dsk	- $1492
128K, mono, 2-dsk	- 1752
256K, mono, 2-dsk, prtr	- 1862
256K, color, 2-dsk, prtr	- 2252

What's New Out There

Regarding my reference to the International 99/4 Users Group: yes, it still exists, though it was a long time from the last IUG publication to the current one, which arrived the week before Christmas. It appears that their financial problems have resulted in a much reduced output. The current publication is a combination software catalog and advertising vehicle. The Enthusiast 99 magazine appears to be a thing of the past, at least for the time being. But it's nice to see something from them, at least. Especially a publication which contains ads from companies marketing goodies for the TI99!

One of the things found in this particular publication is a special offer from Navarone Industries. Unfortunately, by the time you read this, the offer will be over, since it expires on December 31, 1984. But, If you had known about it in time, you could have gotten a free cartridge expander for purchasing two of Navarone's cartridge software products. I don't know why they bother to put out ads like that when people can't see them in time to take advantage of them.

However, there was one good piece of information found in the IUG catalog. The SST Expanded Basic Compiler, developed and sold by SST Software, Inc., has been reduced in price. According to the ad in the IUG catalog, this piece of software is now available for $49, which is a great deal better than the previous price of $95. With an added Graphics Enhancement Package, the price is $59. The ad states also that you have the ability to build your own features into the compiler, or they will provide an estimate for customization. It would appear from this that source code may be available for the compiler, almost unheard of in commercial software marketed for microcomputers. I plan to look into this. so you may hear more in a later column.

So What Else Is New

Ever watchful for good deals for TI users, I have just sent for a membership in a group called the 99/4A National Assistance Group. They seem to be offering some interesting incentives for members, so I decided to join it and review the membership package for you. I haven't received the package yet, but when I do, I will tell you about it in this column. The ad for this group was in the Home Computer Digest, the ad supplement to the Home Computer Magazine. The ad listed some special pre Christmas discounts for members only. One of the special deals was a box of 10 1st quality, lifetime guaranteed diskettes for $9.75 - your choice of SS/SD, SS/DD, or DS/DD. How they expect you to become a member in time to take advantage of these discounts I don't know - maybe that's the catch. We will find out soon.

The latest Home Computer Digest was a veritable storehouse of information. There were some important developments this month, as evidenced by ads in the HCD. At least two companies have begun to produce and sell Extended Basic modules under license from Texas Instruments. This is an exciting development, as it opens up the possibility that other modules may be forthcoming. The companies making and/or selling these modules are:

MicroPal, made and sold by:
TENEX Computer Express
P.O.Box 6578
South Bend, IN 46660
(219)259-7051 (Info)
(800)348-2778 (Orders)

Sunware, sold by:
Unisource Electronics
P.O.Box 64240
Lubbock, TX 79464
(800)858-4580 (Oper #50)

More From Foundation

Our friends from Foundation Computing are advertising yet another new card to insert into your Pripheral Expansion Box. They have become the second company to advertise a CP/M card for the TI99/4A. The Foundation CP/M card boasts a 4Mhz Z80A, 64K memory, 2 RS232 ports and a double-density disk controller. The best news about the Foundation CP/M card compared to the Morningstar version is the price. Foundation wants

$350 for theirs, while Morningstar wants $595. I don't have complete information on the Foundation version yet. When I do, I will pass it on.

Foundation Computing
74 Claire Way
Tiburon, CA 94920
(415)388-3840

Terminal Developments

Since the column two months ago where I reviewed the TE-1200 and AMA-LINK terminal emulators, there have been some developments. Thanks to Dave Hendricks of the LeHigh 99'er Computer Group, we now have a very nice terminal emulator called TE3, source code and all, which will soon be available through the MSP99 software library. This emulator has among it's features emulation of the Lear-Siegler ADM3A terminal. The ADM3A is a kind of semi-intelligent terminal. It does have the necessary attributes to work with full screen editors on some mainframe computers. This emulator originated with Texas Instruments but was never released by them. It has been extensively modified and extended by independent developers, and is public domain software. TE3 is not perfect but we can make it better as we go along. Joel Gerdeen has been working with it and has discovered and corrected a couple of bugs in it already.

A New 1200-BPS Modem

A new, low priced 1200-BPS modem has recently been introduced by the folks at Anchor Automation. They are the ones who make and sell the popularly priced Signalman series of modems. This new modem is called the Volksmodem 12. It works at 0-300 and 1200 BPS, and it has auto-answer, auto-dial, automatic ANSW/ORIG selection, computer controlled dialing and answer, automatic speed mode selection, etc. - virtually everything you need in a modem, and it can be had for only $179 if you know where to look. Of course, Anchor also make another Hayes compatible 1200-BPS modem, the Signalman XII, which can be had via discount mail order for as little as $229. Frankly, I don't yet know any details on the differences between these units, but I intend to find out soon. I will pass on the information when I get it. If you are interested in the Volksmodem 12, drop me a line or a phone call and I'll inquire for you at the same time.

Anchor Automation, Inc.
6919 Valeen Ave.
Van Nuys, CA 91406
(818)997-7758

And To All A Good Night

While I am writing this column, the Christmas holidays have not yet reached their conclusion, although they will have by the time you read it. However, since I didn't think to wish you all a joyous holiday last month, when I could have done so before the fact, I will have to do it now. Please accept my best wishes for the coming year, in hopes that Santa was good to you, and that you will continue to learn about and enjoy your TI99 throughout the coming year.

A Useful Routine

by Gary Gese

Here's a little routine that helps you in defining a long list of characters, or screen locations for graphic scenes, or even to play music using a For-Next loop. This routine can be used anywhere in a program but the longer your list of data the longer it takes to run the routine.

(Continued on page 15)

Using CALL KEY

by Glenn Davis

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 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 keyboard.
2 - Scans right side of 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 (unlike the Apple //e), for example. 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, 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 in upper-case because the Disk Manager (1) Cartridge 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 fewer 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 don't 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? Hmmm ... Gee, how do you break such a program? Don't turn your 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 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 on 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, mainframes or telecommunications, then you might be able to guess what key-stroke 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 conjuction with files on the Disk Memory 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

applications 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 is used to keep track of sprites). When PRINTing graphics to the screen instead of the slower CALL HCHAR or CALL VCHAR, no codes past CTRL-O 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 employed. 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 second method normally used is a CALL KEY ON exp GOTO where the command keys are consecutive, as in A-G or 1-9. 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 separately, 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 calculated 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 literals and/or the 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 represent "S" and "D".

```
100 STRING$=CHR$(0)&"TE"
110 CALL KEY(3,K,S)
120 ON POS(STRING$,CHR$(K+1),1)
    +1 GOTO [illegal or unused
    key routine], [no key
    routine], ["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 subprogram. One (1) must be added to this expression because the KEY statement returns negative one (-1) when no key is pressed, which cannot be used as an argument for the CHR$ function. The argument 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, meaning that a match was not found of the strings. Therefore, one (1) must be added 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. Programs written in this fashion can be easily maintained and modified.

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 scrolling).

Now, finally, we'll cover mode 0, an unusual mode. In the program above, mode 3 was specified once, followed by calling mode 0. This "locks" the computer into mode 3 so all successive mode 0 scans are in mode 3 (refer to the definition of the modes above) until a non-zero mode is called. This is useful in applications programs, for example, an Extended BASIC data-base program that only 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).

The information presented here should give programmers of all levels a little more insight into the powerful TI-99/4A. Take some time to incorporate these ideas into programs of your own and see how much better they RUN!

Want Ads

DISK CONTROLLER - Old style Texas Instruments Disk Controller connects directly to TI99/4: no expansion box needed. $45. Includes Disk Manager, documentation, but no drives. Call Jim Dew at 535-0758, evenings.

FORTH FORUM

Sounding Off In TI FORTH

by Mike Kabala

I'm sure that by now you're probably tired of hearing about how Texas Instruments designed a really fantastic machine, but did such a poor job of marketing it that nobody knew about it until it was too late. That appears to be the case with the sound chip as well. There are a lot more possibilities for this little gem than you are given access to in BASIC. (or EXTENDED BASIC, for that matter.) There's a whole chapter on sound in the Editor Assembler manual that describes these capabilities.

According to this manual, there are two different ways to create sound. The first involves creating a "sound table" in RAM containing information on frequency, attenuation, and duration for each sound you want to produce. The other method is to feed the information directly to the sound processor at address >8400 (hexadecimal).

I chose the second method for two reasons: By poking data directly into the sound chip, you have the ability to change one of the voices (there are three, plus one noise channel) while the other voices just go on doing what they were doing. Also, you have the ability to change the attenuation while a note is playing. This allows you to construct an envelope for each sound as I will demonstrate later in this article.

Wait a minute! This article was supposed to be about creating sounds in FORTH, wasn't it? So why am I spending so much time talking

about assembly language? Well, it appears that the authors of TI-FORTH forgot about sound when they wrote the manual. This would lead most people to believe that TI-FORTH does not support sound generation (as I found out when I attended the November users' group meeting where I was drafted to write this article and to do a presentation at the December Assembly Language meeting) This is definitely not the case. remember, TI-FORTH gives a programmer access to ALL machine resources. That includes address >8400.

Making Noise

All you need to do is store the right values at address >8400. For example, to start a frequency of 110 Hz with an attenuation of 16 DB on voice #1, just type:

```
8900 8400 !  3F00 8400 !
9800 8400 !
```

and to turn it off. type:

```
9F00 8400 !
```

Simple, isn't it?

Okay, okay! So it's not so simple. The only thing that makes it hard is the fact that you have to remember all of those meaningless numbers. It will become a lot simpler if you just define a word to do the remembering for you. The following word will take a frequency value and a voice number and automatically set the proper frequency on that note:

```
HEX

: CREATE-NOTE
   ( freq voice -- voice )
   SWAP OVER 2000 * 6000 + SWAP DUP
   10 / 100 * SWAP ROT SWAP OF AND
   100 * + 8400 ! 8400 ! ;

DECIMAL
```

Now all you have to do is type 1017 1 CREATE-NOTE and you will have set the frequency on voice #1 to 110 Hz. Still not simple enough? Try this. Just define another word called A0 (meaning the note A in octave 0) like this:

```
: A0 1017 ;
```

Then you just have to type A0 1 CREATE-NOTE to do the same thing. (CAUTION: This word actually redefines a hexadecimal number. Be sure you're through with hexadecimal before you define this word.)

Attenuation can be treated in a similar manner. The following two words will allow you to turn a note on and off:

```
HEX

: NOTE-ON ( voice -- )
   2000 * 7000 + 800 + 8400 ! ;

: NOTE-OFF ( voice -- )
   2000 * 7000 + F00 + 8400 ! ;

DECIMAL
```

Now, typing 1 NOTE-ON will turn voice #1 on, and typing 1 NOTE-OFF will turn it off.

It's All In The Timing...

The only thing that remains is to define words to waste time. That can be done with loops. I defined a word called MILLISECONDS to keep track of most of this.

```
: MILLISECONDS ( n -- )
   10 / 155 * 0 DO LOOP ;
```

Place a value on the stack corresponding to the number of milliseconds you wish to delay. for example:

```
A0 1 CREATE-NOTE NOTE-ON
50 MILLISECONDS 1 NOTE-OFF
```

will play the note we defined earlier for a duration of approximately 50 milliseconds.

Notice that you don't necessarily have to turn the note off after the time delay. Let's define a few more notes for this next example.

```
: C1 855 ;
: E1 679 ;
: G1 571 ;
```

Now try the following. (You may wish to define it as a word and then execute it.)

```
C1 1 CREATE-NOTE NOTE-ON
50 MILLIS
E1 2 CREATE-NOTE NOTE-ON
50 MILLIS
G1 3 CREATE-NOTE NOTE-ON
50 MILLIS
1 NOTE-OFF 2 NOTE-OFF 3 NOTE-OFF
```

Notice that the first two notes continue playing as the next note is switched on. This provides a great deal of versatility in creating sounds. (You just have to remember to turn all the notes off when you're done with them!)

On The Attack

Now comes the fun part -- envelope generation! You can create much more interesting sounds if you play with the attenuation a little bit. The way we have it set up now, a note is either on or off with no variations in between. Natural sounds aren't quite so simplistic, however. The sound that a note makes at its beginning is usually very different than the sound it makes a few milliseconds later. That's what enables us to tell the difference between a flute and a piano.

While the TI Home Computer can't reproduce the sounds of these instruments exactly, we can come closer by changing the attenuation (volume) of a note very rapidly at the very beginning and the very end. To do this, I defined some variables to hold my attenuation values.

```
0 VARIABLE ATT-LEN
0 VARIABLE ATTACK 18 ALLOT
```

Then I defined a word to make it easier to store these attenuation values away.

```
: STORE-ATTACK ( nn ... n1 n -- )
   DUP ATT-LEN ! DUP 0 DO SWAP
   100 * OVER 1 - I - 2 *
   ATTACK + ! LOOP DROP ;
```

To store an attack, place the attenuation values (numbers between 0 and 15) on the stack followed by the number of attenuation values in the list and then call STORE-ATTACK. For example,

```
12 8 4 6 7 8 6 STORE-ATTACK
```

will produce a piano-like attack. Similarly,

```
13 11 9 8 4 STORE-ATTACK
```

will produce a more organ-like sound.

Now we may re-define NOTE ON to produce an attack on each note corresponding to the stored values. (These values may be changed at any time simply by calling STORE-ATTACK with the desired values on the stack)

```
: NOTE-ON ( voice -- )
2000 * 7000 + ATTACK ATT-LEN @
0 DO OVER OVER SWAP I 2 * + @ +
   8400 ! 5 MILLISECONDS LOOP
   DROP DROP ;
```

Now, if you type NOTE-ON, you will have the attack added to the start of the note.

In Conclusion

Well, I could go on, but the remainder of what I've done has been mostly in the way of embellishments to the stuff I've already mentioned and it wouldn't be much fun if I told you everything. You can probably think up better things on your own anyway. I will make one little suggestion, though. If you make the names of your words a lot shorter than what I've used here, you will be able to fit your sounds on much fewer screens. (you won't have to type in as much either)

I've found it easiest to define a song as a single word called PLAY. Then, after you've loaded your song, you just have to type PLAY to hear it. I also define an empty word called SONG at the beginning of the first screen of a song definition. That way, if I want to load a different song, I just type FORGET SONG before loading the new song and my old song is painlessly removed from memory.

I hope you have as much fun with this as I have. If you come up with any really good ideas, please consider writing an article or a letter to the editor and sharing it with the rest of us. In the mean time, have fun with FORTH sound.

An MSP 99 Library Review

by Gary Gese

Beginning with this issue, the MSP 99 Newsletter will begin running a regular series of reviews of some of the software available to our groups members through the Software Library. Each month, we will feature new reviews of the latest (and not so latest) offerings from our extensive collection of public domain software programs.

In addition to the standard reviews, we will also include a handy ratings chart so you can see at a glance how the program stacks up in the reviewer's opinion. It will list the Title, Library Code, and system requirements, just like in the Software Catalog, to make it easy for ordering.

There will also be a list of items to be rated. They are; Performance, Quality, Documentation, and User Friendliness.

Ratings will use the familiar star chart which goes like this:

\- = No Rating

* = Not Bad

** = Good

*** = Very Good

**** = Fantastic

We hope that this column will help you as you decide what to order from the group's library. (We also hope that it will stimulate more software sales!) If you know of or use any of these programs from our library, and feel that perhaps not enough has been said about it, feel free to contribute your own review of it to this column. We would be more than happy to consider printing it.

Practice That's Fun

Title: Type-ette Timer
Code: E04020
Req : B,XB,G

Performance: ***
Quality: **
Documentation: *
User Friendliness: ***

This little program is for those of you that never got past Typing I back in high school. (Or for the person that went out and bought a computer for all the great things it could do, and then found out he had to learn how to type!)

But don't get me wrong. This program's not just for adults. Nor does it teach you how to type. Kids of all ages can make great use of it, as long as they've had a beginning typing course or, at least, are proficiently self-taught.

It begins with an attractive Title screen, followes by a short list of instructions. A line of type appears on the screen. After the computer beeps you must type in the same line exactly as it appears on the screen. Press enter as soon as you finish and the computer will automaticly tabulate your WPM and whether or not the typing is correct.

For every correct line the computer advances a little graphic bird across the screen. For every incorrect line, the computer deducts a little from your total score. After 10 lines the computer tabulates your average WPM.

But this is more than just a program to gage your typing speed. If used regularly for say half an hour a day, there is no reason anyones typing can not improve. And, it's fun. Many of the random sentences that come out of the computer are rather humorous. One of them made me laugh so hard I wasted time on the line I was trying to type.

Of course, the lines of type are enclosed within data statements so you can change them if you wish, but each line must be kept to within 20 characters including spaces and punctuation.

However, I did discover one fault in the program. I found that when I messed up a too badly, (the program does not allow you to correct mistakes) and decided to quit the line by pressing enter rather than make a worse mess of things, that I recieved a higher score than I normally did.

Pursuing this matter deeper, I found that if I typed 6 or 7 characters of gibberish and hit enter, I could get scores 15 to 20 WPM faster than my average typing speed. Once I even clocked in at a brisk 97 WPM. Needless to say, I did not let this go to my head.

Yet in spite of its one minor flaw, (which I suspect does not really matter except to those that normally type at 97 WPM) it's a great little program for anyone wishing to improve his or her typing skills.

(Continued from page 7)

```
100 RESTORE 150 (enter the line
    number of the first data
    statement for this set)
110 FOR X=1 TO N ("N" being the
    number of items in the data
    list)
120 READ A,A$
130 CALL CHAR(A,A$)
140 NEXT X
150 DATA (Fill in the numeric and
    string variables here in that
    order. Remember to keep them
    in pairs.)
```

Each time the For-Next Loop is done, the computer takes a new pair of numbers and calls them A and A$. You can fill up Data statements with several pairs per line, thereby saving memory.

To use the routine with HCHAR or VCHAR locations, write them in order row,column,char#,repetitions even if that number is only 1.

SURVEY ITEM	TOTAL	%
EQUIPMENT OWNED:		
TI99/4A	66	100
Cassette	65	98
Joystick	62	94
XBASIC	57	86
Speech	52	79
Disk Drive	48	73
TE II	48	73
32K Memory	46	70
Expansion Box	45	68
RS232	45	68
Printer	44	67
TI-Writer	38	58
Editor/Assem	32	48
Modem	30	45
FORTH	27	41
MultiPlan	23	35
MiniMemory	18	27
LOGO	16	24
Other Computer	10	15
P-Code	7	11
TI99/4	2	3
MBX System	1	2
MEETING ATTENDANCE:		
Often	43	65
Seldom	18	27
Never	4	6
SKILL LEVEL:		
Recreational	20	30
Applications	29	44
Beginner	29	44
Intermediate	29	44
Advanced	13	20
MONTHLY MEETINGS:		
CONTENT-		
Fabulous	5	8
Better/Average	35	53
Could be Better	14	21
Needs Help	3	5
PRESENTATION-		
Fabulous	4	6
Better/Average	33	50
Could be Better	17	26
Needs Help	3	5
PERSONAL APPLICABILITY-		
Fabulous	4	6
Better/Average	24	36
Could be Better	22	33
Needs Help	6	9
NEWSLETTER:		
CONTENT-		
Fabulous	19	29
Better/Average	37	56
Could be Better	7	11
Needs Help	0	0
LAYOUT (DESIGN)-		
Fabulous	19	29
Better/Average	39	59
Could be Better	5	8
Needs Help	0	0
READABILITY-		
Fabulous	23	35
Better/Average	33	50
Could be Better	6	9
Needs Help	1	2
PROGRAMS FROM LIBRARY:		
NONE	18	27
1-10	28	42
11-25	16	42
26-50	4	6
Over 50	0	0
HOURS/WEEK ON COMPUTER:		
5 or Less	19	29
6-10	22	33
11-20	17	26
Over 20	8	12
LENGTH OF MEMBERSHIP:		
1-3 Months	5	8
4-6 Months	4	6
6 Months-1 Year	19	29
1-2 Years	24	36
Over 2 Years	13	20

MSP 99 USERS GROUP
P.O. BOX 12351
ST. PAUL, MINNESOTA 55112

Address Correction Requested

TIME SENSITIVE MATERIAL
POSTMASTER - PLEASE DELIVER PROMPTLY