The Boston Computer Society TI-99/4A User Group June 1988 Meeting Newsletter Edited by J. Peter Hoddie

The Ugliest Title Page in Ages

The reason this newsletter looks nothing like any newsletter since last June is mostly because of a people overflow at the BCS office when I went to do the layout. There was no Macintosh free for enough time for me to do the layout (about 2 or 3 hours) so I mostly faked it. The content is still there, things just ain't as pretty as usual. Hopefully things will be back to normal next month.

The JULY Meeting

The next meeting of the BCS TI-99/4A User Group will be on July 20 (always the third Wednesday of the month) at 7:30 PM (more or less) at the Massachusetts College of Art on Huntington Avenue in Boston. The topics will include the Gramulator from CaDD Electronics, classic programs and cartridges, and new developments in for the 9640 and 99/4A. There are some really exciting products on the immediate horizon so this meeting should be a good one. If you have questions or suggestions give me a call at 861-8733 during reasonable hours.

The Software Libary News

There are three new disks in the library this month. Supplies are limited so get yours early. Remember pricing is still \$3 per disk, or 4 for \$10. I'm not sure on the numbering of the new disks but the contents are as follows. The first new disk is called GEE and is a graphics programming language (not GPL) out of Australia. It is written in assembly, and allows essentially the creation of BASIC programs that can do full bit map graphics. It has the capability to load and save TI-Artist screens and you can do some animation effects with it. This is a really neat program, and for anyone who is into BASIC programming and/or graphics, it should be a lot of fun. It also includes an integrated program editor.

The other two disks are Jack Sughrue's PLUS! system. This system is pretty much an add on for Funnelweb, containing templates and programs aimed at making your word processing more productive. The documentation is extensive, and the user response has been extremely positive. The system received an excellent review in the latest MICROpendium and was introduced at the last New England TI Fayuh. More details to follow.

There are plans for at least one disk of 9640 material next month, as well as at least two more disks of 99/4A material. There will also be updates to Funnelweb, Birdwell's Disk Utilities, and probably Disk Manager 1000. Bring along your copies of these disks (if you purchased them from the BCS) next month along with \$1 each to receive the latest updates.

The BCS Address

BCS, TI-99/4A User Group, One Center Plaza, Boston, MA 02108 . . . I always forget to print this somewhere.

The Boston Computer Society 71-99/4A User Group June 1988 Meeting Newsletter Edited by J. Peter Hoddie

The Ugliest Title Page in Ages

The reason this newsletter looks nothing like any newsletter since last June is mostly because of a people overflow at the BCS office when I went to do the layout. There was no Macintosh free for enough time for me to do the layout (about 2 or 3 hours) so I mostly faked it. The content is still there, things just ain't as pretty as usual. Hopefully things will be back to normal next month.

The JULY Meeting

The next meeting of the BCS TI-99/4A User Group will be on July 70 (always the third Wednesday of the month) at 7:30 PM (more or less) at the Massachusetts College of Art on Huntington Avenue in Boston. The topics will include the Gramulator from CaDD Electronics, classic programs and cartridges, and new developments in for the 9640 and 99/4A. There are some really exciting products on the immediate horizon so this meeting should be a good one. If you have questions or suggestions give me a call at 861-8733 during reasonable hours.

The Software Libary News

There are three new disks in the library this month. Supplies are limited so get yours early. Remember pricing is still \$3 per disk, or 4 for \$10. I'm not sure on the numbering of the new disks but the contents are as follows. The first new disk is called GEE and is a graphics programming language (not GPL) out of Australia. It is written in assembly, and allows essentially the creation of BASIC programs that can do full bit map graphics. It has the capability to load and save TI-Artist screens and you can do some animation effects with it. This is a really neat program, and for anyone who is into BASIC programming and/or graphics, it should be a lot of fun. It also includes an integrated program editor.

The other two disks are Jack Sughrue's PLUS! system. This system is pretty much an add on for Funnelweb, containing templates and programs aimed at making your word processing more productive. The documentation is extensive, and the user response has been extremely positive. The system received an excellent review in the latest MICROpendium and was introduced at the last New England TI Fayuh. More details to follow.

There are plans for at least one disk of 9640 material next month, as well as at least two more disks of 99/4A material. There will also be updates to Funnelweb, Birdwell's Disk Utilities, and probably Disk Manager 1000. Bring along your copies of these disks (if you purchased them from the BCS) next month along with \$1 each to receive the latest updates.

The BCS Address

BCS, TI-99/4A User Group, One Center Plaza, Boston, MA 02108 . . . I always forget to print this somewhere.

- 5. Time of day makes a difference in download times, particularly on CIS. The busy evening hours are substantially slower than weekends during the day or any day in the wee small hours of the morning. If you are cutting costs, plan NOT to download during the evening.
- 6. Xmodem is considered an older slow protocol now in the computer world. All the services except GENIE have quicker options. CIS has its own CompuServe B and Quick B protocols. DELPHI supports ymodem, zmodem, and others. The Source supports KERMIT.
- 7. Ymodem, CompuServe B and Quick B will be available in TELCO 2.0. KERMIT is available now, thanks to Al Beard's porting a FORTRAN program over to the 4A. I have made no time comparisons yet of these other protocols except CompuServe B, which is about 20% faster than xmodem.
- 8. All these tests were made at 1200 baud. 300 baud may minimize differences, but it is hard to say without actually testing. 2400 baud will almost certainly not be twice as fast as 1200 baud.
- 9. For those who live in the Boston calling area, DELPHI appears to offer the best return for your money, judged strictly on download times. For those outside the Boston calling area, many other factors come into play depending on what access means are available to you and how much it costs. This should not be taken as a recommendation of DELPHI over others, since personalities of each, areas of particular expertise, charges at different times of day, charges at different speeds, and so on provide some real differences that you can only judge for yourself.
- As I complete tests on the different services, I will make more of the data available. At present, I have not done enough controlled downloading of identical files across different services to make confident numerical comparisons. When I have completed this data, I should be able to give some indexes by which you can compare hourly charges and access charges to arrive at the least costly means for your own situation. When I do, it will only apply for a short while anyway. One thing is sure in the telecommunications world. Whatever is true today will change tomorrow!

NEW SOFTWARE ON THE INFORMATION SERVICES

Recent new software to appear on all four networks are:

- 1. FUNNELWEB 4.1, dated 29 May 88.
- 2. DSKU 4.1
- 3. GEE, a new fast bit-map graphics programming language from Australia. It is said to be as simple as LOGO to learn.
- 4. TELCO 2.0 (maybe? It almost went up Sunday night, but another bug popped up)
- 5. QDDK, Clint Pulley's Quick and Dirty Development Kit for developing 9640 software from MDOS.

poT-pourrI by Mike Wright

Once upon a time there was a large manufacturer who made a first-class home computer. But, even though its home computer was technically superior to those of its competitors, the large manufacturer was forced to cease production of the computer.

That was all a long time ago. Unfortunately the story is not too becoming as a fairy tale since it lacks the mandatory happy ending.

Today things are different. Today there is a small manufacturer who makes what is claimed to be a first-class computer. It is also considered to have technical features that are better than or equal to those of its competitors.

The ending to this story still has to be written. If you like happy endings you should take printhead to paper and let the small manufacturer know your feelings.

This is what John Johnson did on April 18 this year. John is the author of Remind-Me!, a desk calendar for the TI-99/4A that keeps track of what you should be doing. He is also responsible for the BOOT program that is used by many ramdisk owners.

John's letter was addressed to: Myarc Incorporated, PO Box 140, Basking Ridge, NJ 07920. It reads:

"Dear Mr Phillips,

"You told us many things about the 9640. Things like included speech, ability to run TI programs from an A> prompt, A COMPLETELY OPEN COMPUTER, a good Basic, etc.

"We finally got the machine, and the promises kept coming. We supported you and bought the machine, even though the price climbed from the initial \$350 you and we planned on (remember the initial poll you put out on CompuServe back in 85?), to the \$500 or so that you now ask for it. We supported you as part of the team. one of the orphans. We didn't buy clones, even though we knew that it would have been the most economical thing to do, and believe it or not, there were a few of us that were willing to take the beating because we wanted to support one of our own.

"After all, you knew what it was like, having to struggle through a completely closed system, learning it from scratch, and making the most of it. Well, you gave us real power, brought us out of the orphanage, and we appreciate that. We're still supporting you.

"Even though we still haven't seen ADVANCED BASIC, or the finished MDOS, or the PASCAL, we still stick with you waiting. Waiting while you develop more software for the computer, software that you can sell to us, and probably expect to sell to us before the above products are completed.

"You talk about a protection card to go into the P-box that will prevent piracy. Why are you wasting time on a piece of hardware that won't contribute to the computer? On the same note, why are you wasting time on software programming at all?

"You are taking the same course as TI, aren't you? I hear a lot of rumors about your plans to not give out much technical information on the computer or the operating system. I find it hard to believe that you could be so stupid as to try that on us! Incredibly stupid is what you are if you stab us in the back by trying to keep the machine closed.

"Tlers have learned to program in the last 8 years. Why won't you let us and third party software developers handle the software, and you handle the hardware? I feel I'm a fair programmer on the 4A, but I don't know squat about the 9640. And I don't really have the

time to learn it. You won't provide information on how to program it, again, because you want to monopolize the software market. If you had half a brain, you'd have a technical manual full of information to any and all who requested it, at no charge. If you would have, there would be no software shortage for the machine, as there is now. Instead, you choose to limit the number of programmers to damn few, and you've managed to get on their bad side too.

"So we have My-art. So what? With documentation on the computer, by now we could have had 10 different drawing programs. People don't buy an IBM (or clone) because of the software IBM sells.

"I really enjoyed my 99/4A, and I do see a huge advantage to having a machine like the Geneve. I probably could never go back to a 4A, but if you decide to continue with the foolish marketing system of a closed machine, with you writing software alone, I will not buy any software or hardware from Myarc. I won't sell the machine, or threaten to make you buy it back or any other foolishness, but I will do my best to persuade others not to do business with Myarc.

"I will continue to use the 9640 with whatever software comes along in the form of freeware, third party sales, etc. I will continue to support my peers (all the Tlers and 9640 owners) as best as I can, but I feel you really screwed us. You gave us a compatible computer to upgrade to, and we supported you with our dollars. We put up with your limp excuses as to why products were constantly late. We put up with you labor disagreements with your programmers, while products bought and paid for weren't delivered. Now you are pulling a TI all over again. I have just about had it.

"So I guess what I'm trying to say is that I have lots of patience. I can put up with delays, tap dancing, etc, but when you start talking about a closed machine that I can't program, that's when I call it quits. I hear that your new managing genius, Riley, is responsible for this type of marketing. Straighten him out; let him know that it won't work with this 9640 owner.

"I'm also posting this letter on GEnie, and I'll encourage others to write to you.

"With much bitterness,

"John A. Johnson.

Rave 99 MX01 memory enhancement

Rave 99 is developing a reputation for quality hardware products for the TI-99/4A and Geneve. The company first came to attention with its 99/105 keyboard enhancement. This was offered as a replacement for the often maligned 48-key standard TI keyboard. (Incidentally, much of what is claimed about TI's "small" keys and their spacing just isn't true if you actually measure the TI keyboard.)

Rave then followed up with its speech synthesizer adapter card. This allowed you to move the speech synthesizer from the side of the console into the P-box. For Geneve owners, this was the only way to have speech work at all since there is no way to attach the standard TI speech synthesizer to the Geneve.

The latest offering from Rave is the MX01 memory enhancement. This is a two-layer card (without clamshells) that mounts in any available slot in the P-box. It replaces the standard TI 32K memory expansion card.

The MX01 is offered in three models:

MX01/64 contains 8K of memory at >6000->7FFF, 8K of non-DSR memory, 8K of system DSR memory, 8K of user DSR memory and 32K of main memory. Price is \$199.95.

MX01/288 is an MX01/64 with an additional 256K of main memory. Price is \$309.95.

MX01/544 is an MX01/64 with an additional 512K of main memory. Price is \$419.95.

If you buy any lower model there are upgrade kits to convert it into the next higher one. For this review we will be looking at the MX01/544.

Rave offers the following programs which are sold separately: Keyboard Macro loader, Ramdisk software, Print spooler (not yet released), and a modified version of Myarc's Extended Basic II which will run without a cartridge.

The card uses 32K x 8 CMOS static RAMs (NEC D43256C-12LL). The -12 shows they are rated at 120 nanoseconds. Being CMOS they draw very little power. To retain the contents of memory when the P-box is switched off, you can install an optional 3-volt lithium battery. However, the card has what Rave calls a super-capacitor. The idea is that the cap will charge while the P-box is in use, and then retain a working voltage for up to 5 days. All you have to do is give the cap an initial continuous charge for 14 hours. For practical purposes, the card's memory is retained if you only turn your system on once a week.

Once you have all that extra memory, what can you do with it? The most practical application is a ramdisk. With Rave's MX01 ramdisk software you can configure the MX01/544 card to contain the equivalent of a DSDD disk (1440 sectors), plus a SSSD disk (360 sectors), plus a non-standard disk of 119 sectors.

The memory at >6000->7FFF can be used for any supercart application. This means that programs that will run in a supercart, such as DataBioTics Super Space II, will run on the MX01. Standard supercart applications include: Ryte Data's Command DOS, R.A. Green's Macro assembler, and DataBioTics software support loader.

You could also write your own DSR to supplant or replace the one supplied by Rave. However, there is no technical information supplied with the MX01 on how to do this.

The macro loader available from Rave allows multiple keystrokes to be called out with a single key press. For example, you could define CTRL-T to put out the string "Texas Instruments Home Computer". This macro can then be used from within any program loaded under E/A5. This includes TI-Writer. Macros are not available in Basic or Xbasic.

After you mount the MX01 in your P-box, it must be programmed before it can be used. As with most things, a high degree of flexibility carries with it some extra complexity. This axiom applies to loading the card. To make things easier you can use the following to get your card up and running:

Loading the DSR:

- + System diskette in drive 1.
- + Editor/Assembler cartridge.
- + E/A5 DSK1.LOADER.
- + >1000 default.
- + For DSR load DSK1.XMEMDSR.

Loading the ramdisk software:

- + Ramdisk diskette in drive 1.
- + Editor/Assembler cartridge.
- + E/A5 DSK1.CFG.
- + L = Load

- + N = Do not save files
- + Load DSK1.ROS.
- + Q = Quit.

Installing the Macro loader:

- + Ramdisk diskette in drive 1.
- + Editor/Assembler cartridge.
- + E/A3 DSK1.MACRO.
- + FCTN-9 to return.

Configuring the ramdisks:

- + Ramdisk diskette in drive 1.
- + Extended Basic cartridge.
- + RUN "DSK1.CONFIG".
- + Y = yes to Rave memory card.
- + R = search for ramdisks.
- + Set A=1440, B=360, C=119.
- + Format each disk.

You can now use E/A5 to load a supplied version of Disk Manager 1000 and copy files to your ramdisks.

After you are up and running you have access to the following CALL subprograms: CALL DN.X(Y), change ramdisk drive number; CALL TD, toggle ramdisks; CALL CO/CF, turn cartridge RAM on or off; CALL AO/AF, turn auto-boot on or off; CALL BOOT to run the BOOT program.

Now, how about quirks? After loading the system DSR you can use CALL CSE/CSD for cartridge space enable/disable. This feature is used to test the card by loading two supplied demos: LINES, and CIRCLES. However, once the ramdisk is loaded, you must use CALL CO/CF to access the same feature. It would be more consistent if these calls were the same.

A second quirk that may not bother too many people is that the TI Thermal Printer will not work when the MX01 is installed.

How about service? Initially, my MX01 did not perform as stated in the manual. I telephoned Rave and left a message on their answering machine. That night I received a call from Rave's John McDevitt suggesting further things to try. Nothing helped and I returned the card under warranty. Ten days later it was back and was accompanied by a full explanation of the cause of failure (two traces were touching). Telephone help and a quick return. In today's TI world it is hard to find better support.

How about documentation? The supplied documentation is more than adequate. It includes full switch settings in case you wish to add up to three more boards, giving you a total of 2Mb of backed-up memory (and a further \$1,200 negative cash flow). However, I would like to have seen further technical information on implementing your own DSR, or have the offer of buying the current source listing.

How about compatibility? The MX01 works with the TI Disk Controller, Myarc Disk Controller, TI RS232, TI P-code card, Horizon ramdisk, and Rave speech adapter. Other cards may work, but have not yet been tested. Note that the MX01 does not work with the Geneve.

Are there any missing features? I would like just one to be added. The ability to do a CALL DIR, as you can with the Myarc disk controller. This would it make it very convenient to find out what is being stored in all that memory.

The impossible dream

When it comes to writing a really zippy program you just can't make do with Basic. And yet, Basic offers so many easy-to-use features that most people are usually reluctant to go through the pain of learning assembly. Even after the initial pain, you are always reminded of how easy it is to do certain things in Basic and how hard it is to duplicate them in assembly. For example, try writing the equivalent of DISPLAY AT or ACCEPT AT in assembly. [Hint: there is a minimal display at starting on page 56 of the Mini Memory manual and continuing through page 63.]

Many people have thought about this. One popular solution is to only write the slow parts of the Basic program in assembly and access these routines with a CALL LINK.

But a better solution would be to have a program that would take your Basic program and convert it to assembly — a Basic compiler. To my knowledge, there have been two attempts along this line. One was available from SST, the other was from Ryte Data. However, both programs had restrictions on the Basic coding. For example, SST required the use of LET to load a variable. Neither program could deal with DEFs, and so on.

To my way of thinking, a real compiler would take any Basic or Extended Basic program and, without ifs or buts, simply convert it to source TMS 9900 assembly language.

So why has no one from our talented community come forward with this most desirable piece of software. According to our own J. Peter Hoddie, he and Paul Charlton came to the conclusion that it is simply not possible.

The reasoning goes like this. The compiled program would have to carry an overhead equivalent to ROM 0 (8K) and GROMs 1 and 2 (12K). These contain the Basic interpreter itself, and the many built-in resources it can use. By the time you add the necessary parts of this overhead to a program, there simply would not be enough memory left to load the equivalent compiled program. [Consider that there is only 48K of RAM available in the 99/4A, and that the Extended BASIC cartridge contains over 35K of programming, must of it written in GPL which is generally more compact than assembly code - jph]

The opinions rendered by Peter and Paul should constitute enough authority for anyone. But, one can still dream...

Where's the go in Logo?

Once you've OLDed a Basic program into memory you can usually get it to run by typing RUN. Although this may seem infernally simplistic, it can be circumvented by making the first line in the program:

100 END

Since RUN causes the Basic interpreter to begin execution at the lowest line number, the program will immediately halt. In this case, you would have to type:

RUN [line number]

The problem is that unless you had documentation, you would have to know what line number to use. If you wanted to be really nasty, you could set a variable on a required line number, check for it at various points in the program, and halt if you found an inconsistency. Fortunately, few Basic programs make use of this technique. However, this does seem to be the de facto standard in Logo.

Once you load a Logo program, how do you know how to start it? You can use PP to print out all the procedures. But how do you know which starts the program? If there are any Logo experts out there I'd be glad to hear from them.

Some people never forget - 2

In the April issue (page 9) I pointed out the similarity in names between the defunct IUG's Master 99 Series and Bytemaster's MASTER series.

Richard Mitchell, editor of The Smart Programmer, took me to task for choosing to "discredit" Bytemaster's MASTER series and said that I had done a "grave disservice to a quality product".

In a letter to Richard I explained that the item was based on an overheard conversation in which the participants seemed confused by the appearance of a new "master" series.

In a letter dated May 26, Richard replied: "...please note that the article leaves some peculiar impressions about the origin of Bytemaster's "MASTER" series of software. The series began with the 1987 release of STRING MASTER, a string manipulation package, tagged with the "master" name as an indicator that its origin was with Bytemaster. When it became apparent that other packages might fit nicely in conjunction with the STRING MASTER environment, Bytemaster's MASTER series was planned. There is absolutely no connection whatsoever wit IUG's Master 99 series. In light of the demise of IUG some four years ago and Bytemaster's well-known poor relations with IUG (as outlined in The Orphan Chronicles), I was quite surprised (appalled!) that anyone would even consider it possible that thre might be any link whatsoever between the products."

Richard also points out that I incorrectly referred to the Nov 86 issue of "MG's" Smart Programmer on page 7. In fact, this should have read "Bytemaster's" Smart Programmer.

Production Gramulators shipped

All placed orders for CaDD Electronics' Gramulator were delivered or shipped during the week of June 5. The company now has sufficient parts in stock to build a further 17 units at the current price of \$190 (+ \$3 shipping and insurance through UPS). This price includes a comprehensive manual and software.

The \$10 price increase is caused by the dramatic increase in the price of memory chips, now that the Japanese have been prohibited from dumping in the US. When the Gramulator was being designed, its three static RAMs could be bought for about \$6 each. The most recent price quoted for the same chips is \$17 each, and this could only be guaranteed for a week. One supplier offered a 52-week lead time.

Another supplier predicted that the price of these RAM chips will double before the end of the year. CaDD is simply unable to absorb this kind of soaring cost and will have to pass it on. Now, you've all heard this kind of sales rhetoric before, but in this kind of out-of-control market you should buy now before the price inevitably rises. CaDD will service orders on a first-come first-served basis.

The address to order your Gramulator from is: CaDD Electronics, 52 Audubon Road, Haverhill, MA 01830. To call, note the change in number: (617)-372-0336

Random Ramblings By J. Peter Hoddie

The policy of this newsletter has always been to print the articles provided by the authors without any large scale editing. From time to time this has lead to phone calls from rather prominent members of the TI community asking me how I could support certain positions enumerated in the newsletter. Most of you are aware that I have been working closely with MYARC on the development of the 9640 for over 18 months now. All of this said, I feel that it is necessary for me to respond to some of the comments made by John Johnson and published in Mike Wright's column this month. Before getting into the details I would like to state that (1) John's program Remind Me! is released through Genial Computerware, a company in which I hold a one-third interest, (2) I have a tremendous respect for John and his programming abilities, and (3) it is not my intention in the following comments to "slam" any party, John or MYARC.

On MYARC

A major complaint in John's letter, and perhaps what I would consider the most important, is that MYARC is attempting to "close" the machine. I have not seen any solid information to back up this claim. Back in Chicago in November 1985, before I had any direct involvement with MYARC, I listened to Lou Phillips promise a completely open machine. Since then I have heard nothing to directly contradict this. As recently as May of this year I have discussed the writing and/or reviewing of technical programming information with individuals at MYARC. MYARC's commitment to providing full technical information can be seen in the forthcoming manual for the new hard and floppy disk controller (HFDC). This manual describes every available software call for both the hard and floppy disk. This amounts to proving information which TI considered "proprietary" and has yet to release to this day.

Information on each and every available system call from MDOS has been documented by Paul Charlton in a series of files posted on all the major telecommunications networks and available from the BCS software library since last fall. While I will admit that these files are far from being a a tutorial of any sort, or even the ultimate reference source, these files do provide all the information necessary to write programs that will run under MDOS. Many computer manufacturers publish huge technical reference manuals for their computers, for the Apple Macintosh and Commodore Amiga, as examples. These manuals for the most part are just long lists of system calls with input and output parameters given, which is exactly what the 9640 documents currently do. These manuals do spend some time describing concepts which are part of the particular methodology involved in writing programs for that computer. The 9640 documents do not do that at this time. However, 95% of the calls available from MDOS should be self explanatory to the 99/4A programmer, certainly the intended audience at this time. It is possible to write a program using these documentation. It may not be easy, but then writing software to run in a new environment Is never a trivial task. Several determined individuals have managed to create useful applications under MDOS, including Clint Pulley, Dave Ramses, and Jim Schroeder.

MYARC does not have any full time employee dedicated to writing documentation. Creating good quality documentation is a large job. MYARC is still intent on improving the quality of the available 9640 documentation. However, as John accurately points out, the manufacturer of a computer can never do everything. Apple Computer has some of the best documentation available, yet there are hundreds of books and magazines available on how to program their computers. Some of the most important work done on the 99/4A was done by individual programmers. The information provided by TI, even to their own programmers, never would have allowed for someone to write a program like John's incredible MENU/BOOT system for RAM disks.

The point of this initial discussion, is that I believe sufficient information has been provided for serious work to proceed. This information has been available for many months

now. The documentation can improve, and it probably will. However, those who truly wish to begin work on 9640 specific software should be able to do so. Hopefully they will share what they learn with the rest of us.

John's article implies that MYARC has raised the price of the 9640 in order to make more money. MYARC announced the price of the 9640 at about \$500. Several times Lou Phillips made side comments that no one ever paid list price. When the machine was released it listed for \$500, and many were able to purchase it for about \$400. Since the time when the original costing on the 9640 was done, the prices for certain chips (in particular memory) have increased by over 300%. Since initial release the price has increased to a list of about \$540, and MYARC has provided a substantially improved keyboard as part of this price increase.

The 9640 can still be purchased for under \$500, and considering the capability that the computer provides, this appears to be an exceptional purchase. Consider an IBM style keyboard like the Rave 99 keyboard sells for about \$200, a 512K memory card is another \$200, a 9938 video card roughly another \$200, a GRAM emulator about \$180, and significantly increased speed which really can't be purchased at any price. Total that up and you come out well above \$500 and you still may not have things like a mouse interface, and simultaneous RGB and composite video output. You also end up using several slots in an expansion box. And the 9640 provides MDOS - true command line driven operating system, and the possibility of Advanced BASIC and UCSD Pascal in the near future at no additional charge. MYARC also provides word processing software and an upgrade to the Multiplan package. All this for under \$500 has been considered an exceptional purchase by many.

John makes some brief comments about a protection card for the expansion box. I do not believe in protection schemes. Unfortunately I also understand why they are necessary. MYARC's protection scheme would allow unlimited copies of a piece of software to be made, but they could only be run on only one machine at a time. This will provide no real inconvenience to most 9640 owners. Perhaps in an effort to temper the problem of losing an expansion box slot to a protection card, MYARC has put a connector on the card to allow the use of the speech synthesizer with the 9640. This would also seem to address John's implication that MYARC removed the speech synthesizer from the original 9640 specification. One look at the 9640 card shows that there is no real estate on the 9640 board for the chips required to replace the speech synthesizer. MYARC has what they consider good reason for wanting the protection card. They have figures that they believe show wide spread piracy of the MY-Art program, their first entry into the software business. MYARC invested significant money up front to bring this program to market and did not make back what they expected, in their view, because of piracy. Their hardware protection scheme appears to be a workable solution to this touchy problem.

I'm glad to see that John has not given up all hope in the 9640. I think it is unfortunate that he chose to express himself in such a negative manner in a first public communication with MYARC. Perhaps a more diplomatic first letter would have been more appropriate and less damaging to the TI community as a whole, where fighting has already done too much harm.

More Comments

Moving onto other topics, though still commenting on Mike Wright's article. The Rave memory card is, in my opinion, probably the best memory card option going for the 99/4A owner, and initial tests indicate that it will work with the 9640 as a RAM disk with the next release of MDOS. I am working on a complete review of the Gramulator intended for MICROpendium so I won't get into too many details, but if you are looking for a GRAM emulator such as the GRAM Kracker, the Gramulator is an exceptional product.

I don't believe that Mike's observation that Richard Mitchell/Bytemaster's "Master" series of software had the same name as the old IUG's unreleased Master software was intended to make any negative inferences about Bytemaster's products. I have String Master, the first release in the "Master" series, and it is any extremely well executed package, as the recent review in MICROpendium also stated.

XB/AL Programming

Through some courses I have been exposed to while attending (great and glorious?) Boston University over the past three years I have picked up a few ideas which are applicable to programming on the 99/4A. One interesting idea is that of the exception which is used in the Ada language. Essentially an exception is an error condition. However, by forcing an error to happen significant amounts of code can be removed from a program. For example, consider the case of an Extended BASIC program that uses assembly language CALL LINKs to perform certain basic operations. These operations could each generate a particular fixed set of errors. Each time you call any of these CALL LINKs you must check to see if any of the error conditions occurred. This may consist of many IF-THEN constructs to check the condition of a set of error code variables that the CALL LINK returns. This is a time consuming procedure and can often involve large amounts of code.

Instead of using this approach, I have used the capability of an assembly language CALL LINK routine to generate an Extended BASIC error on return from its execution. If the Extended BASIC application contains an ON ERROR statement with a carefully written error handler, the error handler will be called immediately on return from the CALL LINK with no overhead. I tend to use generally un-used error codes (such as IMAGE ERROR) for these cases, as then you can easily determine if the error is in your program or forced from an assembly language call.

MICROpendium and c99

c99 is a good language. It allows the 99/4A owner to create compiled code. Learning to program is c99 seems to be a popular subject these days. This newsletter has been running Donald Mahler's c.Column for 2 years now (accurate?) and MICROpendium has had a monthly column for nearly a year. Unfortunately, the column in MICROpendium is not a great source of information, in my opinion. The author spends time talking about obscure topics (like using negative array subscripts, something may not even work on future version of c99), and bad programming practices. He ignores standard C programming conventions, and many of his programs appear to be little more than bad translations of BASIC programs. If you are interested in learning C, I would suggest reading Donald Mahler's columns here, checking out some of the source code examples available in the public domain (including our own software library), and looking into general publications on C programming. I cannot encourage you to spend any time studying the material published on c99 in MICROpendium at this time. While the programs may appear to "cute" or interesting, I believe that the programming practices and misleading explanations will do more harm than good for those who wish to do any serious working c99.

Despite my distaste for the c99 coverage in MICROpendium, I still encourage anyone reading this to subscribe to the magazine. It is the only consistent monthly source of information on the 99/4A and 9640. It contains regular articles by writers including Mike Dodd, Walt Howe, and Cheryl Regena. They have regular reviews, news, and tips. If you are at all serious about your 99/4A or 9640 (and you must be to still be using it) then you should check out MICROpendium.