

Boston Computer Society TI-99/4A User Group Meeting Newsletter April 1987

May Meeting

The next meeting of the Boston Computer Society TI-99/4A User Group will be held on May 20 at the Massachusetts College of Art building on Huntington Ave in Boston at 7:30 PM (or so). There is on street parking available. We intend to meet at this building through the summer. I'm not sure of the room number that we will be using, but look for signs downstairs or ask the omni-present security guard. The software library will be present as usual and will contain several new disks that we picked up during the faires this past month.

The meeting topic is, unfortunately, a bit up in the air. We will definitely be taking a look at what happened at the Ottawa and LA shows which will take place the weekend of May 16. Also, we will have the 80 column versions of TI-Writer and Multiplan for the Geneve to display. We will also have a look at some of the current flock of page layout programs for the /4a. These programs allow you to use fancy fonts and graphics in your documents with varying degrees of flexibility and complexity. A new version of DM1000 is slated to be released at the Ottawa show, so we should have it in our library by the meeting as well. This should certainly be enough to fill one meeting.

THINGS MY MODEM TOLD ME by Walt Howe

Well, the Fayuh has come and gone for 1987, and although we do not know the bottom line on the earnings vs. expenses yet until the school bills are in, it certainly was a success on other grounds. The attendance was good, the sales were brisk, the attendees were enthusiastic, the speakers were excellent, and there were few glitches in what is a very complex undertaking for the Fayuh Committee and the many volunteers who helped. Those who travelled a long way to the Fayuh all seemed to think their trip was worthwhile.

Actually, two faires have taken place since our last meeting - our own and the New Jersey TICOFF, and some of us diehards went to both. Even if the Fayuh did no better than break even on costs vs. expenses, the BCS TI-99/4A User Group made a lot of money. At TICOFF the week before, we made about \$1000 after expenses, and we topped that at the Fayuh. Tom Ward provided incredible support for the sales from the BCS library, by copying well over 1000 disks in two weeks.

Although most of you were at the Fayuh, I'll single out a few things just in case you missed them. One was the new rival to the RAVE keyboard from RI. Like the RAVE, it provides an interface that plugs into the console. Unlike the RAVE, it does not come with a keyboard, allowing you to use whatever IBM compatible model you wish. For those who already have a keyboard they can use, it saves on the expense. The new model also adds several new CALLs to BASIC.

I picked up four new Adventure module games by Lucille Rock on one disk for \$5 from the North Eastern 99ers. That one was quite a bargain! I have worked through one adventure so far and it was excellent. It was really priced to sell! There is one puzzler about it, though. The disk sleeve had a *blacked out* copyright statement on it, attributing the copyright to TRIO+ Software of Liscord, IA. I'm sure they had permission to sell it, but why was the copyright statement *blacked out*? Does it mean that they are in the public domain now?

Lou Phillips and the MYARC crew as usual attracted lots of attention. Although we in the BCS have been spoiled by regular advance showings of the 9640, lots of people in attendance were seeing it for the first time. Even for those who have seen it before, there was a lot new on view for the system. We saw MULTIPLAN in 80 columns for the first time, we saw a demonstration of multitasking involving the 9640 and a dumb terminal running two different programs

simultaneously, and we saw some great new graphics drawn with the mouse-supported draw program. The 9640 was delayed again by some 300 bad pc boards that had to be replaced by the manufacturer. It's supposedly really on the way now, but if I don't have one at the meeting, we'll know something else popped up to delay delivery.

There has been a lot of controversy on the information services lately. Dr. Ron Albright, author of "The Orphan Chronicles," has taken the position that the 9640 is only for 99/4A owners and the day has gone when the 9640 could have offered any appeal to others. It is too far behind the AMIGA, the ATARI ST's, and the new MACINTOSH models in software development, if not in capabilities, to offer any outside appeal. In making his position clear, he has belittled MYARC's claims that a broader range of software will be coming from companies that write for other computers. He is actively advising that if you are not a hacker or a strong 4A hobbyist, if what you want is better application software, you should not consider the 9640 unless and until a lot of software is available. Ron's position makes some sense, and if he wasn't quite so negative in his outlook, I would find it easier to agree with him. If the only reason I wanted a computer was to run application programs, I would probably wait a while to buy the 9640. But I can live with the software that is already available, particularly with the new 80-column versions, and I really want to try my hand at programming with the new capabilities. I don't want to buy a much more expensive computer and start to amass software from the beginning again. So the 9640 is definitely for me! I'm impressed enough with MYARC's business record and product support to believe the 9640 and MYARC will be around for a long time to come. Another argument to wait is that there will in all likelihood be an even more powerful MYARC computer somewhere in the future, but I have learned not to keep waiting for the next generation. If today's version is what you want, don't wait for tomorrow's promises no matter how good they sound.

The 4A has seen some incredible utilities and hardware developed in the past year, but very little application software. Without a secure software market, 4A programmers are writing for themselves more than ever, and the utilities are the result. Some are complaining that the 9640 is drawing away programming talent, but I really don't see that happening. More significant is that there is no such thing as a standard 4A anymore. Once the standard was 16k console with tape recorder. Then it was 48k, PE Box, and a single-sided, single density drive or two. But what is it now? With new RAMdisks, GRAM devices, operating systems, quad drives, and 80-column cards the hardware available has far

outstripped the old standards. What capabilities should a programmer write for? Should he write for 48k only and ignore the added features, or should he limit his market by using them? I see that more than the 9640 as a limiting factor. Perhaps it will take the new standard set by the 9640 to get development moving again towards applications software. I don't see much more being written for the 4A, unless they are 9640 offshoots!

So why should anyone stay with the 4A? There are lots of answers to that one. If you are happy with everything it can do and do not need any more software or features, why spend the money? Lots of people fall into that description. Or if you enjoy learning about computers and you feel you have much more to learn about the 4A, why complicate things by getting a more complicated computer? The 4A's design was very far seeing in some ways. I've been told that if the computer had not been built with an 8-bit buss and 6 wait states, it could have run like a minicomputer. It essentially uses mini or main-frame assembly language as it is. Some believe that the hardware limitations were deliberate to keep it from being too capable, just as the IBM PC Jr was introduced with a chiclet keyboard and other limitations originally to keep it out of the business world. The point of this is not the limitations, but the largely unexploited capabilities that remain. There is enough in the 4A to keep a developer who wants to break new ground busy for a long time.

c.COLUMN

by Donald L. Mahler

This month, rather than giving a teaching program, we will discuss some interactions of FW and c99. The older version of FW (3.3) will lead from the menu directly (using c99b) to the c compiler included on the disk. When compiling is finished, you are led directly back to the FW menu ready to assemble. With FW 3.4(Feb87), the c compiler is not included; the instructions are to copy c99c,c99d, and c99e from the c system disk, and change file names to CP, CQ, and CR (which will load in following the file CO). If you do this, call the c.compiler from the FW menu, and attempt to compile a c.file with an "#include dskn.file", the error message "bad file name" appears! However, if you run "CO" from the program format loader, everything works as it should. The slight hitch is that when you are finished compiling, you are returned to the Title Screen (or the BOOT menu if you are using the HD and JPH's op system) rather than the FW menu. All is not lost; when the Assembler is loaded, the file name has been retained in memory (actually, the name of the original c.code rather than the compiled SRC code). I have

written to Tony McGovern about this "glitch".

If you are a FW and c99 user, you may have noted another problem; if you change object code to "program image format" using the "SAVE" option of Editor-Assembler (found on the B disk that came with the E/A cartridge), it cannot be run with the EA5 option of FW. You must use the E/A cart EA5. This applies also to any c program image files that you may have obtained from disks (e.g. my DCHANGE) or BBS's. However, there is a new version of the C99PFI file on the FW3.4(Feb87) disk. To change a c object code to program image:

- 1) Call "load and run" from FW
- 2) Load C99PFI
- 3) Load your obj code
- 4) Load other needed o.codes
(e.g. PRINTF)
- 5) Load CSUP
- 6) Load C99PFF
- 7) Load FWSAVE (instead of SAVE)

The resulting program can be run from the option 5 (program format) loader of FW. Since running my c programs converted to image files had been the last remaining use for my E/A cart, it has now taken its place in the box with TIW, TEII, and other historical relics!

Random Stuff

By J. Peter Hoddie

I'm worn out. Two computer shows in two weeks. School. Exams. Sickness. Newsletter. Lab reports. These are all factors that have contributed to the lack of length in this newsletter. At least the articles by Walt and Donald have enough good content to make up for this collection of unrelated thoughts that I am now setting to screen.

The New Jersey show was in NJ. That says alot. But actually, despite some rather huge technical problems, that I would never have anticipated, during my talk, and a scramble to set up, I had a pretty decent time. The high light of the show for me was seeing a group of friends in from LA, namely Tom Freeman, Terrie Masters, and George Steffan. George is one of the best kept secrets of the TI Community. George knows everything about everything. He wrote the "GRAM Disk" version of TI-Writer and Editor/Assembler for the Miller Graphics GRAM Kracker utilities, is a regular contributor to the LA 99'ers newsletter, a really fantastic assembly programmer, and a pleasure to talk with. Tom Freeman, who I had met once before at Barry Traver's home this summer, is the author of Diskassembler, the only disassembler I've ever actually enjoyed working with. Tom also writes a

monthly column for the LA newsletter which often contains near commercial grade programs and excellent accompanying text. For the two shows, the LA group put together a booklet collection and disk of Tom's material, with the set going for \$9. If you are into programming in BASIC or assembly in any way or are looking for some good utilities, this booklet is for you. If you are interested in a copy let me know, and I'll order some copies from LA for the next meeting.

The Boston show (New England Fayuh, if you insist on the formality (?!)) was fun. I was completely worn out before it opened but all the same, it was a good time. To be completely honest, we had a terrible time dealing with the "staff" of Waltham High School, which made the show a bit of an ordeal, but the spirit of the show was great. Attendance was down somewhat from last year, but not significantly. There were twice as many dealers as last year, selling some very interesting products - as Walt noted. People came from all over the country (world!) to be at this show. There were people from California, Ottawa, Sherbrook (Canada), Nova Scotia (I suspect I spelled that one wrong), Tennessee, New Jersey, Pennsylvania, Maine, New York, Maryland, Washington DC, the Netherlands (really and truly!, the long distance award), Utah, and all of the New England states as well. It appears that Boston has become "the" show to go to, at least in this part of the country. I wonder if it will happen again next year? It is really too early to tell, because those of us that did most of the work are still very busy recovering our sanity, sleep, and shattered lives. Many thanks go to all of the BCS members who helped out during the show. Several out-of-towners commented to me that the spirit of the people who were working the show was great. That no one took the attitude "that isn't my problem" - but instead when something needed to be done, it got done. *Our show had very little (if any) formal organization, but thanks to the enthusiasm of our members (you), it worked out great.*

On related fronts, a huge thanks to Tom Ward who copied over 1500 disks for our group to sell at the two shows. Total sales were close to \$2500 between the shows. This money gives our group some room to do what we want, without having to worry about budgeting problems. One of the things we are considering doing is building a batch of a new peripheral card designed by John Clulow. This card contains 16K of battery backed RAM and allows you to have a program run automatically at power up on your /4A. This is very similar to what is being done with the Horizon RAM Disk, but at about \$50-\$60, it is about 1/3 the price. The card also allows for 8K more memory for assembly programs in the >4000 to >5FFF space which could lead to some nifty

applications. We may have one of these to demonstrate at the May meeting. The reason this is in the same paragraph as the note on Tom, is that he will probably end up building these things if we get it going. If you are interested in building one of these cards yourself, let me know. I have a copy of the schematics and construction instructions that John sent me and I would be glad to pass a copy along. They are also scheduled to be printed in the next issue(s) of the Ryte Data newsletter.

One new product that was introduced at the Boston show that has gone largely un-noticed is a little card from Rave 99 that lets you put your speech synthesizer into your expansion box. This is nice in that it gets the card out of the way, where it can't be disconnected by accident. It was selling for \$40 at the show. The reason I bought one of these cards was to run with my Geneve. As you may know the Geneve supports speech, but there is no port to plug the speech synthesizer into, as there is no console connector like on the /4a. I thought this card would solve my problems as the CorComp Triple Tech card won't work because it wasn't build to specs. Well, Saturday night after the show, a buffet at the Best Western hotel, and a Chinese dinner, Paul Charlton, Corson Wyman, and me went back to my house (my parents' home to be precise) and tried this card out. It almost works. That is to say it worked with Terminal Emulator II, with Parsec, Alpinar, and that sort of thing. The only thing it failed was the multiple word CALL SAY call in Extended BASIC. It turns out to be a slight timing problem. The Geneve is just too fast for the poor old 1979 (or earlier) technology speech synthesizer. But all is not lost. I contacted Rave 99 and told them about the problem. They are in contact with MYARC in an effort to resolve the problem, which should be quite simple according to Paul. The card does work flawlessly with the 99/4A, but if you intend to use it with a Geneve, I suggest checking with Rave before actually purchasing one.

Another comment. If you really feel a need to abandon the 99/4A for what you consider "greener pastures" - don't buy an IBM or compatible. Buy a Macintosh. Buy an Amiga. Buy an Atari ST. These machines are fun. The IBM is boring. But don't toss your /4A out either. I suspect that you will find it does many things faster, simpler, and more conveniently than many of these allegedly high power machines. And for those of you are reading this paragraph and thinking that I am out to buy another computer - wrong. I haven't considered the prospect, or even priced out options. I'm still having too much fun working with the /4A and the 9640 to bail out now.

I used to devote a large portion of my writings for the BCS to programming languages, techniques, and whatever. However, since I started writing for MICROpendium I have been including less material in this newsletter of that nature. What do you people want? Do you actually enjoy this rambling or would you like some serious discussions of programming, and that sort of thing? Tell me, please? I would like to point out that assembly language programming is not impossible. With some perseverance and a will to learn, it can be done. The first thing to remember is that the Editor/Assembler manual is actually correct 99.8% of the time. If you check out the errata that was packed with the manual you can up this to 99.99% of the time. The next thing is to get your hands on assembly source code written by anyone. Genial Traveler is an excellent place to find such materials. Smart Programmer, MICROpendium, Ryte Data newsletter, and many books published a few years ago are all quite useful. No one book will do. Pick up a batch and check them all out. The way to learn assembly (or at least the approach I took) is to type in a program, even if you don't understand it. Preferably a program that is only a page or so long. Then start making changes and see what the changes do. This approach is much easier than trying to write a space invaders program from scratch as your first assembly program. Set your sights low. Work on one thing at a time. For example first learn how to put text on the screen, then how to get input from the keyboard, how to move an object on the screen, how to display and manipulate numbers, how to access system utilities, how to make the thing talk, or play music, or access the disk drive. But don't try to write DBase III in the first 6 months. Also, if you are getting good, or think you know it all, get a copy of the source code to Fast-term. Paul Charlton is one fantastic coder. If you can understand everything he does in Fast-term, then you have nothing left to worry about in assembly programming. If you think assembly is too tough or too time consuming for you, try Forth. I believe that Forth is the best computer language going. Not that I use it much, because I am too comfortable with BASIC and assembly. However, Forth does have an appeal that is quite unique. In some ways it is like Logo in that you can create words and recursive procedures. It uses reverse polish notation like HP calculators, and is stack oriented which can be fun. It is a very fast and powerful language, ideal for someone who wants access to the full power of the /4a without having to learn the details of assembly. If you plan on learning Forth, get a copy of "Starting Forth" by Brodie. No other book is acceptable.