

WASHINGTON DC AREA
TI HOME COMPUTER
USERS GROUP

NEWSLETTER

VOL. No. 2

AUGUST, 1993

ISSUE NO. 9

Bill Whitmore, *President*

Richard D. Sturgell, *Editor*

TI CLUB MEETINGS

THE WASHINGTON DC AREA TI HOME COMPUTER USERS GROUP

The Washington DC Area TI Home Computer Users Group meets monthly at the Fairfax (Va.) High School. The regular meeting night is the second Thursday of each month. However due to schedule problems the meeting for AUGUST - WILL BE HELD ON WEDNESDAY the 10th at 7:00PM.

For directions or other info Call Frank Jordan at (301) 899-3882 or Bill Whitmore (703) 777-2017.

TIBUG - THE BALTIMORE USERS GROUP

The Baltimore Group meets the FIRST TUESDAY of each month at the PINEGROVE ELEMENTARY SCHOOL in Parkville, Md. at 7:00PM.

HAGERSTOWN - WILLAIMSPORT TI USERS GROUP

Meetings are held at the WILLIAMSPORT MEMORIAL LIBRARY on the 3rd FRIDAY of each month. Also a free INSTRUCTIONAL MEETING is held on the FIRST FRIDAY each month. Meetings start at 7:00PM. For more info call Sam Williams at (301) 223-8014.

PARALLEL INTERFACE ON THE RS232 CARD

The latest from TI as how to connect the PIO output to a printer is:

Connect TI pin 1 to CEN Connector pin 1, TI pins 2-9 to CEN pins 2-9, TI pin 10 to CEN pin 11, TI pin 11 to CEN pin 29 and TI pin 16 to CEN pin 16. This should work for many printers (Inc. TI and Epson). When outputting to a Smith-Corona TP-1 the signal on pin 1 has to be inverted with the use of an electronic circuit or you may buy a special cable with this inverter built-in. To use an Okidata Printer two lines require the inverted signals - there is also a special cable for this printer. These cables are advertised in THE 99'er.

If you have had your computer interfaced with these printers or any other printer that required special connections either serial or parallel please send the info to us so that we may pass it along to others.

AVAILABLE COMMAND MODULES as of JULY

TI has made its 99/4A Home Computer very easy to use by making many programs available in Command Modules. These Modules are actually Read Only Memories contained in 'chip' or integrated circuit form. They are programmed in various computer languages: Graphic Programming Language; Assembly Language; and UCSD Pascal. Most Modules are in GPL which is a TI 'secret' ability to develop programs which are capable of making graphic programming easier and which runs faster than either BASIC or Extended BASIC. One Module can add up to 30K GROM and 8K ROM to your computer. Here is a list of the available TI Command Modules, released under the TI name. (All Command Modules start with the TI identification of: PHM 30... Those not yet released or announced are left blank.)

00 Diagnostic	44 Personal Report Generator
01 Demonstration	45 SMU Electrical Engineering Library
02 Early Learning Fun	46 Reading On (S,F)
03 Beginning Grammar	47 Reading Roundup (S,F)
04 Number Magic	48 Reading Rally (S,F)
05 Video Graphs	49 Division I (S,F)
06 Home Financial Decisions	50 Numeration 1
07 Household Budget Management	51 Numeration 2
08 Video Chess	52 Tombstone City
09 Football	53 TI Invaders
10 Physical Fitness	54 Car Wars
11 Speech Editor	55 Editor/Assembler
12 Securities Analysis	56 Alpiner
13 Personal Record Keeping	57 Munch Man
14 Statistics	58 Mini-Memory
15 Early Reading (Scott,Foresman)	59 Scholastic Spelling III (Scholastic,Inc)
16 Tax/Investment Record Keeping	60 Scholastic Spelling IV (S)
17 Terminal Emulator I	61 Scholastic Spelling V(S)
18 Video Games I	62 Scholastic Spelling VI (S)
19 Disk Manager	63
20 Music Maker	64 Touch Typing Tutor
21 Weight Control & Nutrition	65
22 Personal Real Estate	66
23 Hunt the Wumpus	67 Othello (Gabriel Industries-CBS)
24 Indoor Soccer	68
25 Mind Challengers	69
26 Extended Basic	70
27 Addition and Subtraction I (S,F)	71
28 Addition and Subtraction II (S,F)	72
29 Multiplication I (S,F)	73
30 A-Maze-Ing	74
31 Attack (Milton Bradley Co.)	75
32 Blasto (MB)	76
33 Blackjack and Poker (MB)	77
34 Hustle (MB)	78
35 Terminal Emulator II	79
36 Zero Zap (MB)	80
37 Hangman (MB)	81
38 Connect Four (MB)	82 Reading Flight (S,F)
39 Yahtzee (MB)	83 Computer Math Games II (Addison-Wesley)
40 TI Logo	84 Computer Math Games I (AW)
41 Adventure (Scott Adams-Adventure International)	85 Computer Math Games III (AW)
42 Tunnel of Doom	86 Computer Math Games IV (AW)
43 Reading Fun (S,F)	87

COMPUTER BUGS -- BUG OFF !

In the June newsletter I wrote an article mentioning some errors that I had noticed using Extended Basic, version 110. There is NO software of this magnitude (operating system, interpreter, device service routines, and utilities) that I know of that is error free. If anyone thinks that other home computer manufacturers aren't having problems with bugs, they are very mistaken. In fact I know of some manufacturers that are having more than their share of hardware AND software problems. Those of you that have talked to owners of other machines are well aware of this. In comparison to the others, our hardware and software is definitely more reliable AND superior. To quote a Washington member, "I'd rather bitch than switch!". Those are my feelings exactly.

The object of that article was to confirm some of the few problems we do have and develop alternate programming techniques for them. To that end it was very successful. John Yantis (you might remember John visited us with Ed last May) saw the article and went out of his way to take time to write me a letter verifying the problems and the reasons for them. My most sincere thanks to John. Now we have some alternate programming techniques in case we run into any of these situations.

John informs me that the users groups are the source of feedback on TI's products and that they are listening to us. I believe this and think what we, in the past, interpreted as 'unconcerned' was not due to being unconcerned, but because users had no coordinated effort to provide TI with the feedback they needed. A liaison officer of all the users groups would be most helpful in this area.

I originally wrote the article because I was not able to find the appropriate place to air my suspicions. I had called the toll free numbers and programming assistance with no help. John informs me that the procedure is to write to Texas Instruments customer Relations, P.O. Box 53, Lubbock, Tx, 79408, or call the TI Consumer Hotline at

(800) 858-4565. I'm sure they (Consumer Hotline) are now aware that they will be taking possible bug reports. I would suggest that you write so you can give a complete description, definitely include your code and include your telephone number. This way TI will be able to "attempt to duplicate the situation, assess the severity of the problem, postulate possible solutions, and, if warranted, place the bug report in the file for that product for inspection by appropriate Quality Assurance organization."

As to the solutions to the problems I previously noted; unfortunately, I will not be able to review all of the solutions at this time because of space restrictions. The first two and most important ones are here and the others will be included next month. The first was a problem using the RUN command in a program line. If the file referenced in the RUN statement is absent from the disk referenced, an uninitialized disk in the drive, or no disk at all in the drive, the program would abort. To quote John's excellent answer: "When RUN is issued with a filename, either as a Statement or as a Command, BASIC first calls the OLD routine, which clears out an area for buffering and processing the new program. The area that BASIC can use which is least destructive to any program in memory is one that also contains the Symbol Table which Prescan builds for the program each time it is RUN from memory. This buffer allocation is done before the filename is checked for validity or for existence. Clearing the Symbol Table results in all variables, subprograms, and user-defined functions being undefined, but the program in memory is otherwise undisturbed. Thus, if the RUN statement or command fails before any portion of a new program is brought into memory, any program already in memory should remain intact. If, however, the RUN statement in an executing program fails, the next reference to a variable, subprogram, or user-defined function will cause a fatal error. In your example, the reason the ERR subprogram would not execute is because BASIC could not find it. Since all subprogram names were undefined at the time (the Symbol Table had been cleared by the RUN statement).

the BASIC interpreter halted the program and issued the appropriate error message, "SUBPROGRAM NOT FOUND IN 1234". This 'bug' has not been corrected since we have not found an area of memory which OLD can use and yet not destroy a program in memory, nor require the use of Memory Expansion. Your advice to check the filenames on a diskette before issuing a RUN statement is the best 'work-around'."

Listed at the end of the article is a subprogram that can be easily tacked on to the end of any program. The subprogram will also help you to 'work-around' another 'bug' that we will talk about next month also.

My second 'bug' is a perfect example of being in the forest so long that you lose sight of the trees. Take a look at the second paragraph of the description of the ON ERROR statement on page 131 of the Extended Basic manual, "Once an error has occurred and control has been transferred, error handling reverts to the normal action, STOP. If you wish to have any new errors handled differently, an ON ERROR statement must be executed again."

I had it right the first time, but running into another error, I took out the second ON ERROR statement, created my own problem, and then ran off yelling 'BUG'. I was wrong but everyone will benefit from it if they check the book first, before putting the problem on the interpreter. Bounce your problems off the users group second. Maybe someone could start a programming problem section in the newsletter. If you have the problem, so does someone else. Don't think you're alone.

Below is the subprogram for determining whether a file is present so that a RUN statement should be executed. Using the exact code below, it will only check for a program image by the filename specified. This subprogram can easily be modified to do any file checking you'd like. We'll discuss it more next month.

Classified ads are free to User Group members. Your ad could have been here.

```

910 SUB DS(A$,B$,C)
915 ONERROR 940 :: OPEN #11:"DSK"A$
      ".", INPUT, RELATIVE, INTERNAL :: INPUT
      #11: D$,C,C,C
920 FOR E=1 TO 127 :: INPUT #11:D$,X,
      Y,Y :: IF D$=B$ AND X=5 THEN C=1 ::
      GOTO 935
925 IF LEN(D$)=0 THEN C=2 :: GOTO 935
930 NEXT E :: C=4
935 ON ERROR 940 :: CLOSE #11 ::
      SUBEXIT
940 C=3 :: CALL ERR(E,F,F,F) ::
      A$=STR$(E) :: B$=STR$(F)
945 SUBEND

```

The subprogram only expands your program by about 250 bytes. The possibilities for this subprogram are endless. Some sort of this subprogram should be used if you access files in your program. DO NOT USE THE SAME CHANNEL (#11 here) ANYWHERE ELSE IN YOUR PROGRAM. There is another error that you could experience aside from the RUN statement problem (to be talked about next month) but we'll use this subprogram to provide info .

The subprogram in it's present form will look for a program image with the filename you specify on the disk that you also specify. To call the subprogram use the statement:

```

MMM CALL DS(A$,B$,C) :: ON ERROR NNN

```

Where 'MMM' is the statement line number, 'DS' is the name of the subprogram that must match the name in the SUB statement (line 910 in the above list), 'A\$' is the disk drive number to pass to the subprogram, 'B\$' is the filename you are looking for to pass to the subprogram, and 'C' is the return status.

You may use any string and number variable name to pass the information to the subprogram. These string variables do double duty because they also will return the error code and line number in the event of an error. When 'C' is returned it will be in the range of 1-4, this provides for easy branching via the ON GOTO statement depending on the status. A status of 1 indicates that the filename is present on disk A\$ and is a program image. Status 2 is returned if the file is not present AND/OR not a program image. Status 3 indicates an

YOU -- THE COMPUTER DOCTOR !

Computers are really great little things because they do exactly what you tell them to do. That is, except when they get sick. Large companies pay huge sums of money to retain their own computer doctor. I'm not so well off and when my machine gets sick I can't just call a doctor. Some may immediately look to the Exchange Center as their TI given computer doctors. This is not true, they're not the doctors, they just provide the parts. The doctor is you!

Don't panic! It's all really not that hard. I've been at the Exchange Center when there has been a line that started at the elevators! All those people with all that equipment and as it turned out most problems could have been taken care of at home by reading the book or cleaning the equipment. If they had known what to look for and do, many could have saved themselves the trip. (By the way, 11am to 1pm are the busiest times.) Doing it yourself means less people in line, so if you do have take it in, you won't be waiting in line behind as many people. Even if you don't find your problem, you'll have saved time doing the preliminary checking and cleaning and will have some valuable information for Don when you get there.

One of the reasons I chose TI when I bought my computer, was the excellent service I receive when I have a problem. I've received alot of good service from Don, Tom and Debbie at the Exchange Center. Since Don has so much experience with TI products and the problems experienced by TI home computer owners, I've asked him to share some of the techniques for diagnosing some problems with me. In this and coming articles, I'll be sharing that info with you. If you need the Exchange Center, use it; but remember they are there to help us AFTER we've exhausted our options trying to find the problem.

The problems are varied; hardware, software, intermittent problems, and complete failures. Let's start with complete failures. Ruling out the hardware is the first place to start. There are a few 'matter of course' things

that we can do immediately. The FIRST thing to do is clean the electrical connections. The most probable culprit is dirty contacts. DO NOT USE ANY ABRASIVE CLEANERS. Liquid cleaners such as denatured alcohol and those used to clean magnetic tape heads work very well. You'll also need some business cards and Q-tips. A soft pencil eraser will do in an emergency but is not a substitute for a proper cleaning. Be careful of getting eraser crumbs in you system.

Edge connectors (the boards with the contact areas) are a part of 1. the plug in modules, 2. the I/O port on the right side of your computer or plug in (train) peripherals (don't forget the speech synthesizer), 3. the peripheral cards that plug into the expansion box, and 4. the disk controller card contacts (and cables) for DSK2 and DSK3. The female connections are the cartridge port, the peripheral interface cable, the receptacles for peripherals in the Expansion Box and the left side of plug in peripheral boxes, and the female connections on the disk drive cables.

First, clean the easiest and most obvious, the modules and the cartridge port. With constant use, these are the places that you will find that get contaminated the heaviest and the most often. To clean these, simply moisten the Q-tip with the cleaner and rub the edge connectors of the module (you might as well do all of them). Now fold a business card in half longways, moisten it and insert it into the cartridge port. If you can't move it left and right fairly easily, DO NOT force it. Withdraw the card, move it the left or right, and re-insert it. Be careful not to damage the contacts in the port and with reasonable care, you won't. DO NOT use too much cleaner either. Clean it, but don't give it a bath. If you don't have the cleaner, rub the edge connectors with the eraser. You won't be able to clean the port, but removing the dirt on the modules may get you up until you can get an appropriate cleaner. You'll be surprised with the amount of dirt and oxides that build up in a short period of time on the modules and in cartridge port. You may also like to clean the contacts on the peripheral interface cable on the

right side of the computer at this time. This normally will not have to be cleaned as often as the cartridges, just don't forget it.

If you still experience problems, start by removing the hardware that you are NOT using in that application. Remove the speech synthesizer, extended memory, RS232 interface, P-code card, or disk controller that you do not need, ONE AT A TIME and try the application again. *** IT IS IMPORTANT THAT YOU WAIT AT LEAST 2 MINUTES BEFORE REMOVING OR REPLACING ANY PERIPHERAL CARD IN THE EXPANSION BOX. THIS CANNOT BE OVER EMPHASIZED *** Be sure to clean the contacts of each peripheral as you take it out of the system. If your problem clears up, re-insert each peripheral ONE AT A TIME and IN THE ORDER YOU REMOVED THEM and try the application each time. If you experience the problem after re-inserting a peripheral (that has been cleaned) take that peripheral back out of the system. Now verify that the system does work again. This peripheral should be cleaned again, re-inserted, and re-tested. The object is to definitely isolate the problem before taking it to the Exchange Center. Also, if your system now works without the peripheral, you'll probably only need to take the tested (and confirmed) peripheral to the Exchange Center for Don to test on his system and replace if necessary. Call and ask first whether or not you need to bring the whole system if you have isolated the problem to one peripheral.

If the problem still persists, remove, clean the contacts, and replace each peripheral in your minimum configuration and re-try the application. *** AGAIN, OBSERVE THE 2 MINUTE TIME REQUIREMENT FOR REMOVING AND INSERTING PERIPHERAL CARDS IN THE EXPANSION BOX *** If you still have not been able to cure the problem, think about using a friend's identical software and hardware on your system. Swap in your friend's software, test, and if necessary, swap a peripheral one at a time. Make sure that you clean all of the contacts on the new hardware and software you are swapping into your system. If you don't have a friend easily accessible you'll have to pack it all (the minimum configuration that is

not working) up and make the trip to the Exchange Center. If nothing works, take everything, no use making a second trip.

Before you go to the Exchange Center, get together ALL of the facts. Have the following written down: 1. your problem and configuration, 2. ALL the symptoms you've observed so far, 3. the procedures you've gone through to rectify the problem, and 4. any unusual occurrences. Go armed to the teeth with information. This will significantly reduce the time necessary to isolate your problem.

Next time we'll talk about new systems, intermittent failures, and software problems. If you have any questions, I can normally be found at (703) 920-3955 during the day.

Mike Lambert

BUGS! From page 5

error, in most probability you had no disk in the drive, a disk not initialized in the drive, or a device error. Status 4 will only be returned if the disk header is full and the file has not been found.

Your program should do the following upon completion of the subprogram:

1. Specify a new error handler (ON ERROR MMM).
2. Transfer appropriate control for a status 1 or 2.
3. Notify you immediately if the disk header is full in case you have more files to make later in the program.
4. Check A\$ and B\$ in the event of a status 3 being returned. If a status 3 is returned, A\$="130" and B\$="915", channel #11 cannot be used anymore, a file error will result. This is another bug and I'll discuss it next month and give an alternate programming technique.

Mike Lambert

BIT AND SINCERELY BYTE'S CORNER

Dear BIT

My friends in TI Land want to know the what, the how, and the why about the TI computer. I ask you for information and advice and what do you do. Send me two quotations. What relationship does the thrill and secrets of your computer and having fun and enjoyment have to do with:

"For the ladies, the little things count, not diamonds."

Even more preposterous is:

"The way to a mans heart is through his User's Reference Guide followed by his stomach."

Bah-Humbug!

I hope you have a good explanation for these "gems" as you call them. If not, your 99/4A may leave home because of ridicule.

Sincerely BYTE'S

P.S. In response to last month, I can not divulge my sources concerning all the inside information I discover about you. When do we hear the story of the waiter and the computerless check?

Dear SINCERELY BYTE'S

WE computer owners sometimes have a tendency to miss the diamonds and beauty of our computer by overlooking the little things that count or that deserve our attention for an hour or two. Often we treat lightly or ignore valuable information that is available at our fingertips. A worse tragedy is we give up and banish the computer to some dark closet. Alas, we have another instance that Puff the Magic Dragon suffered. Why? We simply are so anxious to operate a new computer, we miss 50 percent of the fun of computing because we did not use the User's Reference Guide. Guess what. This applies to us old timers who have owned a computer for several months or more. This long introduction leads us to:

Part One-The Addendum, Table of Contents, Glossary, and Index and How To Use Them.

If you have not done so, locate and put your User's Reference Guide into a notebook. Find a pencil, or marker of some kind. I use a red pen and yellow marker. Let's begin with the Addendum. It is a compact source of valuable information. The addendum is normally published for at least three reasons and they are: 1) Advise the user of information omitted from the original documentation. 2) To warn users about possible problems. 3) To update or change information contained in the current User's Reference Guide. Because the addendum contains the most current information, you should study it-yes I said study it not just read the words. To study an addendum, begin by reading it and paying special attention to words or phrases similiar to the following:

DO NOT	ALWAYS
MUST	SHOULD
IMPORTANT	BE SURE THAT....
NEVER	NOTE THAT...
MAY NOT...	PLEASE BE SURE TO...
WOULD NOT...	CAN NOT

These types of words are sometimes called "red flags". They are warnings that something important is being said. The "red flags" discuss what will affect or what is different or how some function should be accomplished when using your computer. Therefore, underline the important parts with the red pencil. Underlining is a time saver because it permits you to quickly locate some "gem" of reference information you need when operating the computer.

For the first few months you own your computer, review the underlined parts every week or two. Thereafter, read them every month for at least a year. Your discovery of little things to enhance your computer enjoyment and knowledge will soon be worth the value of a diamond to you.

What else can we learn from our addendum? Good information about using the TI Solidstate cartridge (new name for command modules). You may be having trouble using software in that the instructions state to use the shift and V keys, but the 99/4A shift key will not work because the instructions are for a 99/4 computer. you should press the FCTN and the number 6 keys for the 99/4A computer. This difference between the two computers is explained in the addendum.

What about definitions? You will find the words Wired Remote Controllers in the addendum. Checking the glossary for a definition, we do not find these words but we do find the the words in our Table of Contents. Reading the description on page I-8 does not fully explain what wired remote controllers are. However, in the index we find that page 90 contains information about them. After reading page 90, we learn that wired remote controllers are a fancy name for joysticks that are used to play games etc. Now I know, after reading paragraph two of the addendum, that if the ALPHA LOCK key is pressed down, the Wired Remote Controllers may not work properly. Many owners have cursed their computer because the joysticks did not work properly and yet the answer was in paragraph two on page 1 of the addendum.

One last thought on the subject of addendum. Did you know you have a legal right to the Texas Instruments newsletter and other information made available by TI? You may be missing out on some great information and TI has said "...automatically entitles you to". If you are not getting the TI newsletter, read your, you guessed it addendum!

Sincerely,

BIT

P.S. Maybe next month we will discuss the computerless check.

BEGINNERS AND INTERMEDIATE CHALLENGE

July answers

The beginners answer to last month's challenge is contained in line 160

160 PRINT TAB (10) ;A\$

WHY?

BEGINNERS CHALLENGE

When using the cassette recorder, and you are asked:

*CHECK TAPE(Y OR N)?

The single-letter responses(Y, N, R, etc.) must be _____ when saving or loading data on a cassette tape.

Write a program that will print a DIM of 22 numbers.

INTERMEDIATE CHALLENGE

Write a program that will print a sentence diagonally across the screen. (From the upper left corner to the lower right corner).

If you specify a VAL string that is ___ or the _____ is longer than _____, the message " " is displayed and the program stops running.

PROS CHALLENGE

The intermediate challenge of last month is this months PRO challenge. Write a program that will tell me how many ways you can make change for a single dollar.

Write a program that will identify how many triangles are in the figure.

BELIEVE IT OR NOT

The architecture of the TMS9900 microprocessor is based on words. However, most memory of semi-conductors is usually organized in bytes. One TI memory word is 16 bits or two bytes.

Bit will give a \$50 savingsbond to the first member or group of members (17 or under) to solve the problem of making change for a single dollar. (Help limited to hints only from adults or professional programmers please) Also, a \$100 savingsbond will be given for the best program to solve the PRO challenge triangle problem. Contest runs to September 30, 1983. Award will be based on program uniqueness, originality, graphics and sound. Judges decision will be final. All contestants must be members of the Washington DC Area TI Home Computer User's Group.

copyrighted 1983

ADDENDUM

TI RS232 Interface Card Owner's Manual

If additional ports are required, the Peripheral Expansion System will accept a second RS232 Interface Card. However, this second card must be permanently modified. It can then be addressed as RS232/3 and PIO/2 or, if a Y-cable is connected to the second card, as RS232/3, RS232/4, and PIO/2. This modification will be done without charge at a Texas Instruments Service Facility, or, if you prefer, you may order a modified card directly from Texas Instruments Consumer Relations.

The modification is very simple and if you are in a hurry to convert one of the RS232's and your electronic fabrication and assembly expertise is sufficient, read on--

First open the case, use a Phillips screwdriver of the proper size to remove the four(4) screws located at the corners of the RS232 package. There is no need to remove the wire handles, carefully lift the upper half of the case as you would open a book. Just open the case only as far as required, this action should not damage the label at the top of the card case, and remove the printed circuit card assembly.

CONT PAGE 13

SO WHAT'S INSIDE THE 99/4A?

For the brave and adventurous, the inside of your 99/4A is a "jungle", although a very neat one, of electronic parts and pieces. There are 47 Integrated Circuits, 10 transistors, 2 crystals and resistors, diodes and a few capacitors. Let me try to let you in on what's what (watt?) by focusing only on the 2 crystals and the 47 IC's.

- 1--48MHz crystal (48,000,000 cycles per second): This is the main "heart-beat" for the computer's timing and control signals.
- 1--10.7MHz crystal (10,700,000 cycles per second): Controls the timing for our video/monitor/TV output signal.
- 1--TMS9900: 16 bit Central Processor Unit, the "brain" that operates and controls the 99/4A.
- 1--TMS9901: NMOS serial In/Out controller which routes all signals to and from the CPU.
- 1--TMS9918A: The fantastic composite video display processor which lets us have four modes of graphics, sprites with 3-dimensional depth, etc.
- 1--TMS9919(SN76489N): The 8/16 bit complex sound generator/processor which lets us have up to 3 tones and 1 noise at the same time.
- 8--MK4116N: 16K x 1 bit MOS dynamic RAM. Out 16K of user memory wherein our Basic programs reside (hide?).
- 2--MCM6810P: 128 x 8 bit RAM, totaling 256 bytes of static RAM used as a "scratch pad" for Register Memory.
- 3--TMC0430(1-CD2155NL; 2-CD2157NL): 6K x 8 bit ROM for a total of 18K Rom, which is GROM 0,1, & 3 (up to five others can be in Command Modules). These contain the graphic abilities and most of the TI Basic Interpreter.
- 2--TMS4732(C51397): 4K x 8 bit MOS static ROM for a total of 8K ROM holding our Graphic Programming Language, the operating system and part of the TI Basic Interpreter. This 8K ROM plus the above 18K ROM/GROM give us the advertized 26K ROM.
- 1--TMS9904(SN74LS362N): The clock driver which takes the 48MHz crystal input and changes it into 4 phases of 3.5MHz that the CPU needs to operate.

The above are the main IC's. (TMS=Metal-oxide Semiconductor/MOS) The remaining IC's just keep things moving along--literally! Along the data buses by changing 16 bits to 8 or 4 or 3 or whatever is needed to make the whole thing work.

(SN=standard circuits; 74LS=0°C to 70°C, low-powered Schottky TTL circuits)

- 3--SN74LS00N: Quad dual NAND gate
- 3--SN74LS04N: Hex inverter
- 1--SN74LS74AN: Dual flip/flop
- 2--SN74LS138: 3 to 8 bit decoder
- 1--SN74LS156N: Dual 2 to 4 bit decoder
- 1--SN74LS194: 4 bit shift register, parallel In/Out
- 2--SN74LS244: Octal 3 state bus driver/buffer
- 1--SN74LS245: Octal 3 state, bi-directional bus transceiver
- 3--SN74LS367: Hex bus driver
- 1--SN74LS373: Octal dual flip/flop latch
- 1--RC4558P: High performance dual op-amplifier
- 2--TIL 119: Darlington opto-coupler/isolater for our cassette recorder control
- 1--uA7812C: +12 volt DC voltage regulator
- 1--uA79MO5C: -5 volt DC voltage regulator
- 1--uA723C: +5 volt DC voltage regulator
- 1--TL 331: DC voltage comparator

I hope this helps the inquisitive, but who cares as long as you turn it on and it works--at least mine has for two years. (Mean Time Before Failure is said to be over 4 years.) Also, even though TI is the #1 IC manufacturer in the world (followed by IBM and everyone else!) all the IC's in my computer were NOT by TI. Some were MOSTEK, Motorola, Ratheon, Synertec, Signetics and American Microsystems. With over 1.3 million 99/4A's is the world TI would have needed 61+ million IC's--the old song of supply and demand.

Rev. Keith G. Koch

TEXAS INSTRUMENTS: Why TI?

by BARRY TRAVER

As you may have noticed, PACS now has a TI Users Group. Since Texas Instruments advertisements tend to be long on personality (Bill Cosby -- like James Garner -- is appealing!) but short on facts, many PACS members may be asking the legitimate question, "Why TI?"

The purpose of this article is to survey briefly some of the facts on the TI 99/4A personal home computer.

The basic console, available for less than \$200 under TI's rebate offer, comes with 16K RAM and 26K ROM, including a built-in version of BASIC with capabilities for color graphics (with 16 colors available), music (with three-tone harmony), and a number of user-friendly utilities not automatically found in microcomputers (e.g. RESEQUENCE, extensive EDIT, TRACE, BREAK, etc.).

The console, incidentally, makes use of a 16-bit microprocessor (rather than the less sophisticated 8-bit microprocessor) and has a typewriter style (rather than mylar touch) keyboard. An RF modulator (or TV adapter) is ordinarily included in the package at no additional cost; an inexpensive cable makes storage of data and programs possible on an ordinary cassette recorder.

The ROM of the console can be easily expanded by 36K (without loss of any RAM) through use of a cartridge or command module (e.g., Extended BASIC, about which more will be said in a moment), while the RAM may be expanded by 32K (using TI's own memory expansion) or even by 128K (using third-party materials), allowing a total of 144K or more if desired.

No expansion of memory is needed to make use of most of the benefits of the Extended BASIC module, which greatly expands the capabilities of the TI BASIC of the console. Similar to other expanded BASICs, Extended BASIC contains the expected (many new statements, capability for multiple-statement lines, expanded IF-THEN-ELSE statements, multiple variable assignments, trailer comments, direct screen control of input and output, etc.) and the perhaps unexpected (e.g., the ability to write and use true subprograms--not merely subroutines--and up to 28 true sprites, i.e., smoothly-moving, high-resolution graphics whose shape, color, and motion can be thoroughly controlled).

Some special features, however, of Extended BASIC require certain peripherals for full utilization: use of the built-in speech editor requires the speech synthesizer; the ability to load and run one program from another requires a disk drive and disk drive controller; and support for loading and running assembly language programs requires the 32K memory expansion.

These and other peripherals (such as RS-232 interface, telephone modem, and an Epson MX-80 printer with TI's name on it) are presently available for the TI 99/4A through the Peripheral Expansion System, and lower-cost versions of many of these will also be available through TI's recently-announced Hexbus line of peripherals, which are also intended for use with the TI 99/2 (a speedy new computer, also with a 16-bit microprocessor, for less than \$100!).

Languages presently available for the TI 99/4A include not only assembly language and two versions of BASIC (as already mentioned) but also Pascal, Pilot, Forth and two versions of LOGO (most of these require the possession of certain peripherals).

How about software for the TI 99/4A? TI, itself, has available more than 125 business, educational, and technical programs and more than 175 game, adventure, and arcade titles. In addition--although third-party support for the 99/4A has been somewhat limited in the past--there are indications that a software explosion is currently taking place with, for example, Walt Disney, 20th Century Fox, CBS,

Milton Bradley, Parker Brothers, and others producing game cartridges for the 99/4A.

In the educational area, the TI 99/4A is one of only three home computers that will be making available programs from Control Data's extensive PLATO courseware library. (By the way, TI's versions of the PLATO programs should provide superior graphics, since they will be the only ones making use of sprites!) In addition, TI--in a special agreement with Control Data--holds exclusive rights to more than 500 PLATO programs, claimed by TI to be "the pick of the litter."

Other educational software for the TI 99/4A is being produced by such well-known names as Scholastic, Addison-Wesley, Milliken, DLM Academics, the Minnesota Educational Computing Consortium, and Scott, Foresman. TI also has its own word processing program (the TI-Writer) and its own electronic spreadsheet program (the Microsoft Multiplan) for those who may work with words or numbers for a living.

Although--like its competitors--the TI 99/4A is not a perfect computer (next month's article may comment on some of its shortcomings), perhaps enough facts have been given here to show that Bill Cosby does have real reason to smile in his friendly advertisements for the TI 99/4A.

"Why TI?" It's not just a matter of Bill Cosby's appealing personality but of specific facts that have persuaded some people who know computers that the TI 99/4A is an excellent one for their specific intended purposes.

Interested in knowing more? Whether or not you own a TI, you are invited to attend our TI Users Group meetings to find out for yourself. (Skeptics are especially welcome!)

SOFTWARE ---

TEXAS INSTRUMENTS: Short and Long Comings by BARRY TRAVER

Last month's column surveyed some of the impressive strengths of the TI 99/4A home computer. Because of TI's new \$50 rebate offer, this computer, complete with 16-bit microprocessor, 16K RAM, 26K ROM, built-in BASIC, 16-color graphics, three-voice music, typewriter-style keyboard, RF modulator, etc., will be available throughout the rest of 1983 for less than \$100. For those who want more than a computer console to attach to their television set, there are many peripherals available (e.g., speech synthesizer, 32K RAM expanded memory, RS-232 interface, printer, telephone modem, up to three disk drives, etc.) as well as a number of programming languages (Extended BASIC, Pascal, Forth, two versions of LOGO, and assembly language).

Like any of its competitors, the TI 99/4A also has some weaknesses (even if we are all persuaded that the few disadvantages are outweighed by the many advantages!). This month's column will comment briefly on some of those weaknesses and some of Texas Instruments' attempts to correct them.

Although TI has done a commendable job in supporting the TI 99/4A (and before that the TI 99/4) with software made available by Texas Instruments, there has been a real shortage of significant third-party software. Why were third parties not writing software for the TI (apart from TI's earlier lack of aggressiveness in the marketplace, something that has radically changed)? Because earlier the only way to go so was to invest in a \$50,000 minicomputer development system, and few were persuaded of the wisdom of such a financial investment, to say the least.

The situation has changed for a number of reasons, including the introduction of TI's editor/assembler package. The May 1983 issue of **Creative Computing** in its in-depth review of the TI 99/4A had this to say about that package:

"this module plus disk set is an extremely powerful software tool--in reality a minicomputer assembler in microcomputer form." "Some of the remarkable capabilities that the Editor/Assembler brings to the microcomputer field are these: * Writing of relocatable, linkable code. The programmer does not need to worry about absolute addresses and can write his programs as independent subprograms to be linked together by the loader. * Extraordinary editing capabilities. The Editor is very complete and offers a wide variety of conveniences for the programmer. The system looks rather like a word processor for assembler code. * Links to Utility Routines. The utilities give the programmer the ease of programming exhibited by the Extended Basic while allowing the speed of machine-assembled code."

No longer is a \$50,000 minicomputer development system necessary in order for third parties to write impressive programs for the TI 99/4A; the editor/assembler package is available for less than \$50!

Although we should thus expect to see much more cassette- and disk-based software for the TI 99/4A being produced by third parties, Texas Instruments has just made an announcement that--unfortunately--may discourage third-party development of cartridge-based software in the future. In essence, as I understand it, TI wants exclusive rights to produce and market software modules for the TI 99/4A (although third-party authors of such would receive appropriate royalties). While such an arrangement is not uncommon in the book or record industry, it is rather unusual for computer software. Although a change in the architecture of the new TI 99/4As coming off the assembly line will give TI the technical control to eliminate third-party module production, I believe that this development will prove to be a step backwards for TI.

In addition to these software "shortcomings" of TI, there are also some software "longcomings" that

provide occasion for complaint. Much of the software announced for release during the first quarter of 1983 has not yet appeared, and other items seem "long coming" as well. TI's much-awaited version of PLATO courseware (the only version to make use of sprites) is still on the way, even though scrapping of the Minnesota Educational Computing Consortium project should make more people available for the PLATO project. If TI cannot keep on schedule for production of its own software projects, there is all the more reason to wonder why TI wants to add production of third-party software modules to its already-impossible list of "things to do."

To be fair to Texas Instruments, TI already has its own software library of more than 300 programs presently available for the TI 99/4A. And to be sure, there are other things to talk about than software availability (watch future columns), but that is a matter of importance to TI 99/4A owners. In summary, we are disappointed in TI's failure to release its own new software on schedule and to properly encourage third-party production of software modules, but are encouraged that--because of TI's increased share of the marketplace and its excellent editor/assembler--the situation is very promising concerning greater availability of cassette- and disk-based software for this computer.

RS232 from page 10

Look for the integrated circuit marked U7, and then find R5. This 100 OHM resistor will have to be moved to the set of pads adjacent to its present position, the new location is marked PTH1.

* S*M*A*R*T LIBRARY NEWS (301) 681-5065 *
* TI SYSOP, Mike Lambert (703) 920-3955 *

In the near future the Washington Area users Group will be attempting to gather representatives from all interested user groups for the purpose of facilitating the coordinated compilation and dissemination of information to users of all represented groups. We believe that if user groups were linked by representatives, information that can increase the quality of our home computing life will be more available.

I believe that systems such as we are using (SMART), can play a major role toward helping to exchange information. In that regard, if anyone knows of any system (in the continental US) where TI computer users get together, please write or call me so I can make this information easily accessible to as many people as possible. My address is:

Mike Lambert, 306 S. Courthouse Rd.,
Arlington, Va, 22204
(703) 920-3955

In the same respect, if anyone has any information of any kind that would be of any help to other home computer users, get it published or send it to me and I'll get it published for you.

RS232 Cont

To remove the resistor, use solder wick to absorb the excess solder around the resistor and the empty holes where the resistor will now be placed. Now very carefully bend the resistor leads so that they are not holding it, then turn the card over and lift the resistor out of the present mounting holes. Do not pry against the printed circuit card as damage to the printed circuit may result. Push the resistor leads thru the holes at the new position and then bend them as they were originally to provide a good mechanical mounting and then solder on the bottom side very carefully, do not over heat. Be sure to also solder the holes that you removed the resistor from as these are plated thru holes and in some cases the electric circuit could be opened when removing components. When you have completed this modification the soldering should look as neat as when you started. Be sure that only a solder with rosin flux is used.

Cont Page 15

Advanced Basic Programming: Using Strings as Data

by Gene Harter of Not-Polyoptics

(the author has written several commercially available programs, including "Ships!" and "Sengoku Jida ". He hopes that this will be the first of three articles on advanced basic programming techniques.)

Trying to write good commercial software for a micro-computer limited to 16 k of memory has led Not-Polyoptics, chiefly through the efforts of our head programmer Mike Capobianco, into some twilight areas of Basic programming, just to push as much as we can into that limited amount of space. One of the techniques we have discovered, which I will communicate in this article, is using strings instead of data statements for storing data.

Without reviewing all of the string functions offered for our use in TI Basic, I would remind you that the VAL of a SEG\$ gives you a number from a string; that POS can tell you if a character occurs inside a string or not; that STR\$ can take a number and put it back into a string; and that string variables can be subscripted.

As an example, you can use a string to write music.

```
100 A$="11319214783165741756 51965622047247382622"  
110 FOR I=1 TO 36 STEP 5  
120 CALL SOUND(VAL(SEG$(A$,I,1))*100,VAL(SEG$(A$,I+1,3)),  
VAL(SEG$(A$,I+4,1)))  
130 NEXT I
```

This program plays the scale from C to C with varying lengths and volumes. You can see that it does the same thing as a data statement, without restore statements or read statements. Also it works better. A similar idea is used in our game "Hordes", where 21 data files consisting of strings of numbers are manipulated to provide information about the mapboard characters and colors and other data.

A string can hold information concerning a variety of conditions that you may want to check in your program, and more efficiently than variables or subscripted variables. A string of 10 characters may for example hold status information that would otherwise take up 10 variables in a program, and can be changed within the string with only a little effort.

```
100 A=3  
110 A$="1234567890"  
120 A$=SEG$(A$,1,4)&STR$(A)&SEG$(A$,6,LEN(A$)-5)
```

This routine replaces the 5 in A\$ with a 3, or whatever you want A to equal. Similarly, data can be accumulated by a program and operated on by using strings. Since a string may be 256 characters long within the computer this amounts to quite a lot of information.

Manipulating strings solves the age-old problem of how to move a character using the arrow keys.

```

100 CALL CLEAR
110 X=10
120 Y=10
130 CALL HCHAR(X,Y,42)
140 A$="8769828368908867"
150 B$="-1-1-10000010101"
160 C$="-10001-101-10001"
170 CALL KEY(O,K,S)
180 IF S=0 THEN 170
190 IF POS(A$,STR$(K),1)=0 THEN 170
200 CALL HCHAR(X,Y,32)
210 X=X+VAL(SEG$(B$,POS(A$,STR$(K),1),2))
220 Y=Y+VAL(SEG$(C$,POS(A$,STR$(K),1),2))
230 CALL HCHAR(X,Y,42)
240 GOTO 170

```

This program lets you move an asterisk around the screen with the 4 arrow keys ESDX and the 4 diagonal keys WRZC. It is much shorter than programs using If-Then or On-Goto statements to accomplish the same task.

In conclusion then, strings can be added to or changed while a program is running, operated on without For-Next loops that data statements need, provide a means for holding detailed information without using variables, yet like variables they can be used over and over again, or set to null.

YUOR USER'S GROUP

The User Groups meet monthly for a program of discussion and presentation that will enable you to be better informed about your TI Computer. The Members share and exchange information. Some members have a broad range of computer expertise that they are willing to share with others. We are owners of TI Computers, and share an enthusiasm for these machines. You are welcome to visit a meeting before you join.

The Newsletter is published each month to serve the need of the Home Computer Users of TI Computers. Members are encouraged to contribute articles for publication. Opinions expressed are those of the writer and not necessarily those of the Users Group, it's officers, editor or members.

Articles appearing in this Newsletter may be reproduced if appropriate credit is given to the author and the Users Group.

RS232 from pg 13

Now replace the card into the case and secure with the four screws.

PLEASE NOTE: IF YOU DO THIS YOURSELF THE TI WARRANTY WILL BE VOID. So be careful or get help from someone OR have TI do it for you.

If your RS232 has been changed to give 4 RS232 outputs - it may not work with TI Writer in the Edit mode of Files, PrintF. However it will access RS232/3 or 4 when it is in the Text Formatter mode.

THE CASSETTE RECORDER AS A MASS STORAGE DEVICE FOR THE TI-99/4A

By: C. H. Cook, 10713 Lochridge Dr.
Silver Spring, Md. 20901

1. INTRODUCTION. A good quality cassette tape recorder can be an effective device for mass storage if certain cares are taken. However, one cannot expect too much because of the very nature of the way a cassette recorder does its recording. A cassette tape recorder pulls the cassette tape across the recording and playing heads at a constant rate of speed and constant tension. This is altogether different from discs which are continuously rotating. As data is recorded on a cassette tape each successive piece of data gets physically further and further from the first piece of data. That is, a file of data can only be organized sequentially.

But, all and all, cassette storage of data files can be the answer if sufficient care and planning of the organization of your files is adhered to.

2. CASSETTE TAPES. One cannot hope to have a reliable tape storage system if certain cares are not taken. Always use a high quality tape! You generally get exactly what you have paid for. In manufacturing literature you will find the following tips for using cassette tapes. ADHERE TO THEM!

1. Keep your cassette tapes inside a protective plastic box when not in use.
2. Do not touch the tape with your fingers.
3. Make sure that the tape is wound snugly on the cassette reels before playing.
4. Avoid exposing cassettes to excessive heat, moisture, direct sunlight, and strong magnetic fields.

As for your cassette recorder, you should clean the capstan, pinch roller, and recorder heads at intervals specified by the manufacturer.

Cassette tape tends to mangle and crease at the very beginning of the tape. To avoid recording or playing errors rewind the tape to the very beginning and set your tape counter to zero. Then forward your tape to 5 on your tape counter. Start your recording at that point.

3. FILES AND FILE PROCESSING. As mentioned earlier, magnetic tape is only accessible sequentially. It can be likened to a scroll. A scroll is read by unwinding it from one end; in order to read a paragraph in the middle it is necessary to roll through all the preceding paragraphs. That is, it is not possible to access the middle paragraphs without skipping through the preceding paragraphs one at a time. The paragraphs of a scroll correspond to records on a tape. A chapter would correspond to a file.

A file on a tape consists of a collection of records. A record consists, in general, of a collection of fields. A field is a portion of a record that consists of a logical unit of data. For example, a record might be the employment information of an employee. All the employee records form a file; the employment information for an employee would consist of fields such as name, address, classification, salary, etc. Fields can be either all alphabetic, all numeric or a combination of the two. On a tape it is also necessary to be able to spot the end of the file. For the TI-99/4A this will be somewhat of a problem. It will be necessary to write a "dummy record" to detect an end-of-file. The length of a record is measured in characters and will always be fixed within a file. However, we will be able to choose record lengths of 64, 128 or 192 characters. The format of a record consists of a precise description of the data in the fields and the position of the fields in the record.

In TI BASIC there are two formats for data stored on a tape. These formats are called file-types. The first file type is the DISPLAY-type format. This format refers to printable(ASCII) characters. The DISPLAY format is used when you want the data to be saved exactly as it would appear in a DATA statement. The second type of format is the INTERNAL format. The INTERNAL format saves the data exactly as the computer stores the data internally. In general, the INTERNAL format is easier to use and is more efficient for recording data on a cassette tape.

4. BASIC STATEMENTS FOR FILE PROCESSING. TI BASIC supports 5 statements for file processing, namely 1) OPEN, 2) CLOSE, 3) INPUT, 4) PRINT and 5) RESTORE. We shall only consider the first 4 of these statements. The OPEN statement is used to open a communication link to the recorder. The OPEN statement contains parameters that describe the file to the computer. The CLOSE statement closes the association of a file and the program. The INPUT statement permits the computer to read data from the cassette tape. The PRINT statement allows the computer to write data onto cassette tape.

OPEN The OPEN statement is of the form
OPEN #A; FNC, FOJC, FTJC, OMJC, RTJL
WHERE

A is the file number-any number between 1 and 255
FN is the file name-either "CS1" or "CS2"
FO is the file organization-sequential
FT is the file type-either internal or display
OM is the open mode-either input or output
RT is the record type-fixed
L is the record length-any integer between 1 and 192

Example of an open statement.

```
190 OPEN #1:"CS1",OUTPUT,INTERNAL,SEQUENTIAL,FIXED L
```

PRINT The PRINT statement is of the form
PRINT #A: C PLJ
where

A is the file number
PL is the print list

Example of a print statement:

```
200 PRINT #1:M,N
```

INPUT The input statement is of the form
INPUT #A: C VLJ
where

A is the file number
VL is the variable list-those variables which are assigned values when the the input statement is performed.

Example of an input statement:

```
530 INPUT #1:M,N
```

CLOSE The close statement is of the form
CLOSE #A
where

A is the file number.

Example of a close statement:

```
640 CLOSE #1
```

It is perhaps a little easier to remember the parameters that enter into an OPEN statement if we write it as follows:

```
OPEN #A: { "CS1" } { INTERNAL } , { INPUT } , FIXED L  
         { "CS1" } { DISPLAY } , { OUTPUT }
```

It is not necessary to use sequential as a parameter in the OPEN statement since it is the default value of the statement.

5. THE PENDING INPUT AND PENDING PRINT CONDITIONS. It is very important to understand these conditions. The TI-99/4A USER'S REFERENCE GUIDE is quite clear on this subject; cf. II-125 and II-135. When the computer reads records from a file, it stores each complete record internally in a temporary storage area called an input/output (I/O) buffer. A separate buffer is provided for each open file-number. Values are assigned to variables in the variable list from left to right, using the data in this buffer. Whenever a variable-list has been filled with corresponding values, any data items left in the buffer

discarded unless the INPUT statement ends with a trailing comma. Using a trailing comma creates a "pending" input condition. When the next INPUT statement using that file is encountered then the computer uses the data in the I/O buffer beginning where the previous INPUT statement stopped. The computer behaves in a similar way when a PRINT statement does not end with a print-seperator (comma, semicolon, or colon); the record is immediately written onto the file from the I/O buffer. If the PRINT statement ends with a print separator, the data is held in the buffer and a "pending" print condition occurs. When the next PRINT statement using the file is encountered, the computer places the data in the I/O buffer immediately following the data already there.

A Case For Protecting Your Investment

Computer owners usually start out with a basic system configuration, then, before they realize it, they have added peripherals and software that make transporting and even storing the system a frustrating problem. Of course, by that time, the system has started to look like a formidable investment. There are two things the user can do to get the most out of this investment. First, increase usage. Second, physically protect the equipment. H/S Enterprises has developed a custom carrying case that does just those things. This hard case fits easily under an airline seat yet carries everything most users wish to use when travelling. The only items which don't fit are the CRT/monitor and memory expansion accessories.

Space utilization is optimal, yet all equipment is well cushioned and readily available. The upper and lower compartments can hold the console, speech synthesizer, A/C adapter, TV modulator, two joysticks, and take your choice, either 14 command modules/cassette tapes or a cassette recorder and 7 command modules/cassette tapes. Cables and books were not forgotten either — they fit nicely in the custom well beneath the console.

Because of our large membership, H/S Enterprises has agreed to provide this custom case to our members for \$53.95, including shipping costs (it normally sells for \$59.95). Order directly from H/S Enterprises, P.O. Box 128, Suffern, NY 10901. Make sure you tell them you are a Washington D.C. member, and send your check or money order with your order.





TERMINAL INPUT

Names &
Addresses:

Cook, C.H.
10713 Lockridge Dr
Silver Spring, MD
593-7090

Koch, Keith
8505 Nicholson St
New Carrollton, MD
459-6826

Minton, Allen
3019 Kramer St
Wheaton, MD
949-4085

PROGRAM #1
(Line 90)

TAPE
Name &
Address
File

TAPE
Name &
Address
File

PROGRAM #2
(Line 5000)

TERMINAL
List Of
Names &
Addresses

PRINTER
List Of
Names &
Addresses

NAME AND ADDRESS TAPE:

Record #1 Record #2 Record #3 Record #4

Cook	Koch	Minton	Stop
---------------	---------------	-----------------	---------------

<---Start
of Tape

End --->
of Tape

```
90 REM THIS PROGRAM READS NAME AND ADDRESS RECORDS FROM
92 REM THE TERMINAL AND WRITES THEM TO A DATA TAPE.
94 REM
96 REM THE USER SHOULD ENTER THE NAMES IN ALPHABETICAL
98 REM ORDER SINCE THE PROGRAM WILL NOT SORT THEM.
99 REM
100 OPEN #1:"CS1",OUTPUT,INTERNAL,SEQUENTIAL,FIXED 192
110 INPUT "LAST NAME, FIRST NAME ":LNAME$,FNAME$
120 IF LNAME#="STOP" THEN 180
130 INPUT "ADDRESS ":ADDRESS$
140 INPUT "CITY, STATE ":CITY$,STATE$
150 INPUT "PHONE NUMBER ":PHONE$
160 PRINT #1:LNAME$,FNAME$,ADDRESS$,CITY$,STATE$,PHONE$
170 GOTO 110
180 PRINT #1:"STOP",",",",",",",",".,.""
190 CLOSE #1
```

```
5000 REM THIS PROGRAM LISTS THE NAME AND ADDRESS FILE
5010 PRINT "DO YOU WANT A HARD COPY? (Y/N)"
5012 CALL KEY(0,HARDC,STATUS)
5014 IF STATUS=0 THEN 5012
5020 OPEN #1:"CS1",INPUT,INTERNAL,SEQUENTIAL,FIXED 192
5024 IF HARDC=78 THEN 5037
5025 OPEN #5:"PIO"
5027 PRINT #5:"NAME";TAB(25);"ADDRESS";TAB(60);"PHONE"
5028 PRINT #5
5037 INPUT #1:LNAME$,FNAME$,ADDRESS$,CITY$,STATE$,PHONE$
5040 IF LNAME#="STOP" THEN 5999
5045 IF HARDC=78 THEN 5073
5050 PRINT #5:LNAME$;TAB(10);FNAME$;TAB(25);ADDRESS$;TAB(60);PHONE$
5060 PRINT #5:TAB(25);CITY$;", ";STATE$
5070 PRINT #5:""
5073 PRINT
5074 PRINT "-----"
5075 PRINT LNAME$;TAB(20);FNAME$
5076 PRINT
5077 PRINT ADDRESS$
5078 PRINT CITY$;", ";STATE$
5079 PRINT PHONE$
5080 GOTO 5037
5999 CLOSE #1
6000 IF HARDC=78 THEN 6010
6005 CLOSE #5
6010 END
```

**TAPE #1
(INPUT)**
Name &
Address
File

**TERMINAL
INPUT**
Names &
Addresses
To be added:
Hughes, Larry
1884 Columbia Rd
Washington, D.C.
667-3574

PROGRAM # 3
(Line 500)

**TAPE #2
(OUTPUT)**
Updated Name
& Address File

TAPE #1 (INPUT)

Record #1 Record #2 Record #3 Record #4

Cook	Kach	Minton	Stop
---------------	---------------	-----------------	---------------

TAPE #2 (OUTPUT)

Record #1 Record #2 Record #3 Record #4 Record #5

Cook	Hughes	Kach	Minton	Stop
---------------	-----------------	---------------	-----------------	---------------

```

500 REM THIS PROGRAM READS NAME AND ADDRESS RECORDS FROM
501 REM A DATA TAPE AND FROM THE TERMINAL AND MERGES THE
502 REM TWO FILES. THE MERGED FILE IS WRITTEN TO A NEW TAPE.
503 REM
504 REM ASSUMPTIONS:
505 REM (1) THE DATA ON TAPE IS IN ALPHABETIC ORDER BY
506 REM LAST NAME. DITTO FOR THE DATA FROM TERMINAL.
507 REM (2) THE LAST RECORD ON THE TAPE HAS "STOP" IN THE LAST
508 REM NAME FIELD.
509 REM (3) TERMINAL INPUT IS STOPPED BY TYPING "STOP"
510 REM WHEN PROMPTED FOR LAST NAME.
511 REM
512 REM EOF0$ AND EOF1$ ARE LOGICAL VARIABLES INDICATING
513 REM END-OF-FILE ON TERMINAL AND INPUT TAPE RESPECTIVELY.
514 REM
550 EOF0$="FALSE"
560 EOF1$="FALSE"
570 REM
580 REM OPEN TAPE FILE 1 FOR INPUT
590 REM AND TAPE FILE 2 FOR OUTPUT
600 OPEN #1:"CS1",INPUT ,INTERNAL,SEQUENTIAL,FIXED 192
610 OPEN #2:"CS2",OUTPUT,INTERNAL,SEQUENTIAL,FIXED 192
620 REM
630 REM PROCESSING DEPENDS ON VALUES OF EOF0$ AND EOF1$
640 REM AS FOLLOWS:
650 REM
660 REM EOF0$ EOF1$
665 REM -----
670 REM FALSE FALSE IF (NEWNAME<=OLDNAME)
680 REM WRITE NEWNAME RECORD TO TAPE.
690 REM READ NEWNAME RECORD FROM TERMINAL.
700 REM OTHERWISE (NEWNAME>OLDNAME)
710 REM WRITE OLDNAME RECORD TO TAPE.
720 REM READ OLDNAME RECORD FROM TAPE.
730 REM
731 REM FALSE TRUE OUT OF RECORDS ON OLD TAPE. CONTINUE
732 REM TO READ RECORDS FROM TERMINAL AND
733 REM WRITE TO NEW TAPE.
734 REM
735 REM TRUE FALSE OUT OF RECORDS AT TERMINAL. CONTINUE
736 REM TO READ RECORDS FROM OLD TAPE AND
737 REM WRITE TO NEW TAPE.
738 REM
739 REM TRUE TRUE HAVE PROCESSED ALL RECORDS FROM BOTH
740 REM TERMINAL AND OLD TAPE. WRITE "STOP"
741 REM RECORD TO NEW TAPE.
742 REM
750 REM

```

```

752 REM GET FIRST RECORDS.
760 GOSUB 1000
770 GOSUB 2000
780 REM CHECK EOF INDICATORS.
800 IF (EOF0$="TRUE")+(EOF1$="TRUE")THEN 850
805 REM EOF0$="FALSE" AND EOF1$="FALSE"
810 IF NEWLNAME$>OLDLNAME$ THEN 830
815 GOSUB 3000
820 GOSUB 1000
825 GOTO 800
827 REM NEWLNAME$<=OLDLNAME$
830 GOSUB 4000
835 GOSUB 2000
840 GOTO 800
850 IF (EOF0$="TRUE")+(EOF1$="FALSE")THEN 900
855 REM EOF0$="FALSE" AND EOF1$="TRUE"
860 GOSUB 3000
870 GOSUB 1000
880 GOTO 800
900 IF (EOF0$="FALSE")+(EOF1$="TRUE")THEN 950
910 REM EOF0$="TRUE" AND EOF1$="FALSE"
920 GOSUB 4000
930 GOSUB 2000
940 GOTO 800
950 REM EOF0$="TRUE" AND EOF1$="TRUE"
960 PRINT #2:"STOP", "", "", "", "", ""
970 CLOSE #1
980 CLOSE #2
999 END
1000 REM READS A NEW DATA RECORD FROM TERMINAL.
1005 REM IF NEWLNAME = "STOP", SET END OF FILE INDICATOR.
1010 INPUT "NEW LASTNAME, FIRSTNAME ":NEWLNAME$, NEWFNAME$
1020 IF NEWLNAME$<>"STOP" THEN 1050
1030 EOF0$="TRUE"
1040 RETURN
1050 INPUT "NEW ADDRESS ":NEWADDRESS$
1060 INPUT "NEW CITY, STATE ":NEWCITY$, NEWSTATE$
1070 INPUT "NEW PHONE NUMBER ":NEWPHONE$
1080 RETURN
2000 REM READS AN OLD DATA RECORD FROM TAPE.
2005 REM IF OLDLNAME = "STOP", SET END OF FILE INDICATOR.
2010 INPUT #1:OLDLNAME$, OLDFNAME$, OLDADDRESS$, OLDCITY$, OLDSTATE$, OLDPHONE$
2040 IF OLDLNAME$<>"STOP" THEN 2060
2050 EOF1$="TRUE"
2060 RETURN
3000 REM WRITE NEW DATA RECORD TO TAPE
3010 PRINT #2:NEWLNAME$, NEWFNAME$, NEWADDRESS$, NEWCITY$, NEWSTATE$, NEWPHONE$
3020 RETURN
4000 REM WRITE OLD TAPE RECORD TO NEW TAPE
4010 PRINT #2:OLDLNAME$, OLDFNAME$, OLDADDRESS$, OLDCITY$, OLDSTATE$, OLDPHONE$
4020 RETURN

```


TI 99/4A

RESIDENT DISK DIRECTORY

No Need to Use the DISK MANAGER
or other specialized programs to find out
what is on your disk.

MACHINE LANGUAGE DIRECTORY PROGRAM, resides in
memory. Can be called ANY TIME, even FROM PROGRAMS
to display your disk directory WITHOUT AFFECTING
ANY BASIC PROGRAMS you may have in memory at the time

Three versions available:

1. Extended Basic (requires Expansion Memory)
2. Console Basic and Expansion Memory (requires either the TI Mini Memory or Editor/Assembler Cartridges)
3. Console Basic and Mini Memory (no Expansion Memory required).

Note: Versions 2 and 3 can be run directly from the LOAD & RUN
Option of both the Mini Memory and the Editor/Assembler.

All Versions use approx. 2000 bytes of Machine Language Memory
(not normally available to Basic Programmers) and work on Single
Sided and Double Sided Disk Drives

INTRODUCTORY PRICE ~~\$24.95~~ ^{\$19.95} EACH Version on disk (please specify)
or ~~\$49.95~~ ^{\$39.95} all three Versions on one disk

Send Check or Money order (VA Residents include 4% Sales Tax) to:

FARAH SOFTWARE - 2516 Fernwood Drive - Vienna, VA 22180
Telephone (703) 281 3550

★ ★ ★ WABASH ★ ★ ★ VERBATIM ★ ★

Disk Connection Ltd.

★ ★ OKIDATA ★ ★ ★ HAYES ★ ★ ★

(703) 451-5342

INFLATION FIGHTING PRICES
PURCHASE BY THE CASE AND SAVE

WABASH	5 1/4"	PER BOX	PER CASE/100
SS/SD		\$21.50	\$175.00
SS/DD		24.50	205.00
DS/DD		35.00	310.00
96 TPI		51.00	470.00
available in Soft, 10 and 16 sector			
WABASH	8"		
SS/SD		\$25.00	\$225.00
SS/DD		34.00	300.00
DS/DD		47.00	430.00
VERBATIM	5 1/4"		
SS/DD		\$27.00	\$250.00
DS/DD		39.00	350.00
SS/DD w/Cleaning Kit			35.00
ULTRA MAGNETICS	5 1/4"		
SS/DD		\$27.00	\$240.00
DS/DD		36.00	310.00
3M	5 1/4"		
SS/DD		\$27.00	\$235.00
DS/DD		39.00	350.00
MAXELL		\$ CALL	\$ CALL

DECISION SUPPORT SOFTWARE PRODUCTS

THE ACCOUNTANT	LIST	OUR \$
Apple version	\$129	\$99.50
IBM version	195	155.00
DB CALC (Visicalc interface)	20	15.00
THE BUSINESS ACCOUNTANT		
Apple version	\$255	\$210
IBM	295	245
Business DB Calc	40	30
EXPERT CHOICE		
LIST \$495 (IBM - PC)	INTRO \$245	\$225
MODEMS		
HAYES SMARTMODEM 300	\$295	\$235
HAYES SMARTMODEM 1200	699	\$ CALL
PRINTERS		
OKIDATA 82, 84, 92, 93		\$ CALL
Flip 'N File		\$22.00
Plastic Library Boxes		\$3.50

PROFESSIONAL SOFTWARE

ASHTON-TATE PRODUCTS	LIST	OUR \$
dBASE II w/application manual	\$700	\$ CALL
FRIDAY	295	245
THE FINANCIAL PLANNER	700	560
THE BOTTOM LINE STRATEGIST	400	350
FOX & GELLER		
QUICKCODE	\$295	\$245
dGRAPH	295	245
dUTIL	99	80
SATURDAY NIGHT		\$ CALL
HUMANISOFT		
DBPlus	\$125	\$99
dBASE II application manual		24.50
TJ/MAKER III	\$295	\$245
Word processor, spreadsheet, list processing graphics, data transfer, file management system — ALL built into one system		

Virginia Residents 4% sales tax
VISA & MASTER CARD accepted
ADD \$3.50 minimum shipping
C.O.D.'s add \$1.65

Phone (703) 451-5342
9 a.m. - 7 p.m. M - F EST
10 a.m. - 2 p.m. SAT EST

FOUR OF MY OBSERVATIONS/EXPERIENCES ON TAPE STORAGE
by D.L.Shie

ONE. If the remote switch polarity is wrong on your cassette recorder, an alternative to the external adapter is to rewire the connections internally to the mini-jack--if you're brave enough to take your cassette recorder apart, and its out-of-warranty. There are three soldered connections to center(C), Middle(M), and Outside(O) parts of the jack. The trick is reversing all three connections, although if you can see the connection between C and O, it can be left in--but beware--both connections between power and C and O must be severed--and this is the rub--if your case in which the jack is mounted is metal, instead of plastic, it may be impossible to disconnect it for your case is then part of the circuit. But I've seen none like that--only mounts in plastic.

TWO. The above is also a good solution if your beautiful recorder has everything BUT a remote jack--as long as it's transistorized and has a DC motor and the two wires to the motor can be found--snip ONE (or better yet, unsolder it from a board) and splice in two wires to a jack you mount into a hole you drill into the plastic case near the microphone hole (DON'T drill through before checking the other side for clearance needed--you don't need a hole in its motor either). Solder one to C, the other to M, and a short wire soldered to connect C and O. One of the diagrams will work--how to tell in advance I have no idea--the other will give you a great set up for regular recording but not for the TI computer--and we don't want an adapter hanging outside--so if you guess wrong, do procedure ONE and switch your wires for the opposite case--or maybe easier, reverse the two connections on your same split wire instead of the three at your jack. If you wire your jack wrong, testing it CAN'T blow it as its effect is to connect ALL THREE wires anyhow for the recorder's motor to work!

I did procedure TWO first, getting it right the first time, then went to work on an old Panasonic with the wrong switch polarity. The jack was mounted directly on a circuit board so I had to scratch off two copper pathways, relocate my red motor wire, and solder a jumper from M to the area from which I had detached C and O. (I also did it wrong first and scratched off a pathway between C and O so I had to solder a wire path between them also.)

I wouldn't advise this on a recorder on which you have a warranty, and some I haven't gotten the cases off without breaking a few screws or plastic fasteners. This was a 10 cent yard sale bargain I experimented on for reversing polarity--and I only figured out afterwards with circuit diagrams that I risked nothing. Note that this does not reverse the polarity of the motor (your tape would run backwards and jam up)--only the polarity of the switch wires to your computer which TI built to handle only one of the two possibilities.

THREE. This item is a discovery--one of my other yard sale tape recorders is a GE Blind Man's Special whose Fast Forward and Rewind let the tape slip by the playback head close enough to hear the high speed play--so I can exactly pinpoint start and end of files by hearing the gaps(or my vocal descriptions) between files by distinctive sound changes from the computer's two tones. This is the way big systems can search tapes for the special "Tape Mark" noise records between files. I don't have to watch a little counter--in fact I didn't have a recorder with one on till recently --my \$1 yard sale banged up Wards Stereo on which I did procedure

MINI MEMORY
By: Pete Crowell

The recently released Mini Memory from Texas Instrument has, in this writer's opinion, opened an entirely new door in the 99/4(A) computer. Everyone has quite possibly read an article or two about this new module, with the bulk of the text being taken from TI's manual. Being only a novice myself in the area of assembly language for the 9900, I shall only touch the area of assembly usage very lightly.

The line-by-line assembler used in the minimem works quite well and allows mnemonic code to be entered directly into the minimem where it is assembled for execution. Let me just say for those of you who are unfamiliar with assembly language, it is FAST! Any individual who wishes to write a program (or subroutine) with the fastest execution time possible, will certainly want to investigate and learn how to use the assembler.

With the minimem module, standard TI Basic can "call" assembly language routines stored in the minimem or loaded from disk to the expansion memory. This permits the usage of extended basic commands, such as CALL LOAD and CALL LINK, to be used by regular basic, for the minimem has an additional interpreter built into its ROM.

Without getting too carried away with the assembly portion of the minimem, I wish to mention the Easy Bug utility which is also included in the Mini Memory module. This little routine allows the user access to all of the 99/4(A) system memory including the Video Display Processor memory (from whence cometh sprites and all of the standard ASCII characters). Easy Bug not only permits the user to examine the contents of the various portions of memory which are not seen by Basic, but it also allows alteration of the RAM memory. A word of warning about memory alteration—be prepared to have to turn the console off in order for the computer to get its head straight when you are done. This person found it quite interesting to play with the video portion of the 99 and forgot to take note of what values were stored in the various memory addresses prior to altering them. Needless to say, my 99 was rapidly producing colors and unidentifiable characters at rather random screen locations before I discovered that I didn't remember just what I had altered.

When you finally tire of looking into the "mysteries" of what makes your 99 tick, you can settle down to using the minimem. Your favor-

ite program that now resides on your cassette tape (starting at tape counter=50 or somewhere thereabout) can easily be loaded to the 99 by the command OLD CS1. Once your personal program has been loaded (with the minimem module plugged into the GROM slot) you type the new command "SAVE MINIMEM". As long as your program does not exceed the 4088 byte capacity of the minimem, you are now in business. Since the minimem contains a battery pack, the module can now be removed, and your program is saved for future use. To reuse your program, you simply insert the Mini Memory module in your console and turn it on. After you select TI Basic and the console is ready, you type the new command "OLD MINIMEM" and your program is ready to run. The entire process is very fast and saves searching for the proper tape or disk. The speed of loading a program has to be seen to be appreciated, for it is much faster than disk.

There are two additional features of note in the minimem. The first is that it permits access to the expansion memory from basic. This in itself is no great thing, but it permits you to use the memory for a large data file for your program. The second bonus feature is just what many of you with cassette-based systems have been wanting—a method of keeping data files to be used with a program. The minimem itself can be used to store data for access by your programs. This permits a minimum system configuration of the console, a cassette recorder, your monitor, and the minimem to provide a handy way to store, retrieve and manipulate data without the high cost of a disk system.

Many new programming ideas come to mind as you use the Mini Memory and get to appreciate the flexibility which this new module adds to your system.

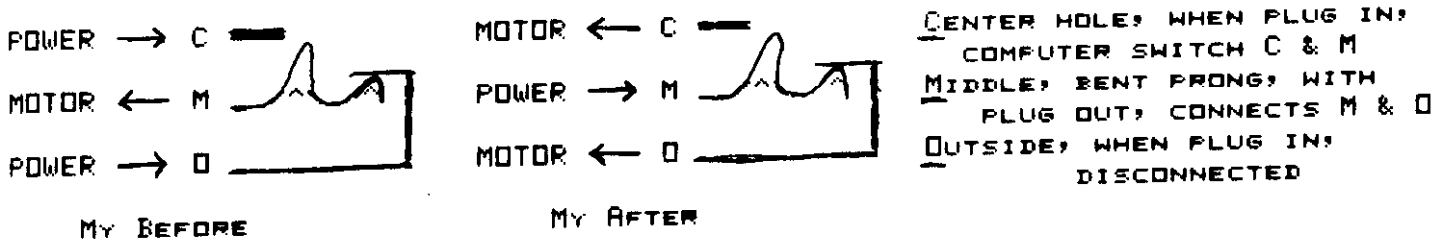
As a rabid fan of my 99, I must say that the new Mini Memory module has heightened my appreciation of just what a great little system you all have residing there in your home. The new Expansion Box has overshadowed a device, which in itself, exceeds nearly any other device which has been made available to the user. If someone out there will just create a course in assembly language, we will all be able to reap the full benefits of this mighty module.

In closing, I would simply like to take my hat off to Texas Instruments for the Mini Memory; and hope that the forthcoming UCSD PASCAL will hold as many pleasant surprises for we, the users, as this new and exciting module does.

TWO. I still like my sound searches best in some cases--I can quickly find a gap and listen to my discription(including program name) then see whether I need to go forward or backward, counting files, to where I need to be.

FOUR. My last item is another procedure. I collected about five recorders until I found one that worked fair--then I discovered a trick to make almost all work well. Most cheap(and old yard sale) recorders have their heads poorly aligned--and all have some sort of screw adjustment on the head to adjust the tracking. The screw is "fixed" in place by a spot of paint at the factory. When you adjust this screw while playing, listen for the best and highest pitched fidelity of a good PRE-RECORDED music tape which has its tracks where they should be. My computer reads that setting fine for almost any recorder--one that failed me had its head too badly worn...it will only now work sporadically, but didn't at all before adjusting. Music tapes sound better to me after this adjustment also! How to do it? My GE blind recorder has a hole for a jeweler's screwdriver which in the play position is right over the screw--so all my stubborn recorders now also have holes drilled neatly through their cases there (CAREFUL, no holes through COMPONENTS under your case, remember!) This is far easier than taking the whole case apart--but it probably voids a warranty with the hole and breaking the paint spot. By the way, you can put your own paint on the screw to help hold it in a place that works best. Bye.

remote JACK Wiring to Reverse Polarity of Switch



FUTURE PLANS: We plan to form a new organization called THE MID-ATLANTIC USER'S GROUP that will be responsible to publish the newsletter. It will be a composite from TI users and other sources and will be available at a regular subscription rate (\$9 to \$10) throught the local user's groups. July 1983 is our target date for the new organization. We will review the possibility of having a TI User's Group fest late this year or early 1984. We want to improve our group purchase efforts and to expand the services of our members.

MEMBERSHIPS: Memberships are available to the general public. Cost for dues depend on the group you join. Membership information for each group is: WASHINGTON D. C. AREA TI USER'S GROUP--Contact Frank Jordan at (301) 899-3707 or mail your dues (\$12) to Frank at 4100 Canterbury Way, Temple Hills, Md. 20748; TI BALTIMORE USER'S GROUP (TIBUG)--Send \$15 to Membership Chairman TIBUG PO BOX 3, Perry Hall, MD. 21128; HAGERSTOWN-WILLIAMSPORT TI USER'S GROUP--Write Sam Williams at PO BOX 376, Williamsport, MD 21795 or call (301) 223-8014; MONTGOMERY COUNTY TI USER'S GROUP--Call Allen Minton at (301) 949-4085.

MEMBERSHIP DUES \$12.00

WASHINGTON D. C. AREA
TI 99/4 USER'S GROUP

MAIL TO: F. Jordan
4100 Canterbury Way
Temple Hills, MD 20748
% TI Membership

NAME _____
ADDRESS _____
CITY _____ STATE _____ ZIP CODE _____
TELEPHONE _____ INTEREST _____

Disk Connection, Ltd's User Group Special



SPECIAL \$60

LIST \$100
40% Discount \$40
Club Price \$60

NAME _____

ADDRESS _____

CITY _____

STATE _____ ZIP _____

Send Payment to: **DISK CONNECTION, Ltd.**
P.O. Box 2489
Springfield, VA 22152
(703) 451-5342

Virginia Residents 4% sales tax
VISA & MASTER CARD accepted
Add \$2.50 shipping • C.O.D.'s add \$1.65

FUTURA SOFTWARE - MAILING LIST

The latest version (5.0P) of the Mail List by Chas. Ehninger of FUTURA SOFTWARE now has been updated so that it will interface with TI-WRITER. This will then insert the name from the mail list in a form letter created with the TI-WRITER.

TI NEWS
P. O. BOX 267
LEESBURG, VA.
22075

TIME SENSITIVE MATERIAL