-092
8706

Vol.5 No. 6 JUNE 1987

This newsletter is published jointly by OH-MI-TI and New Horizons TI-99/4A Home Computer Users' Groups. Material may be reproduced without permission provided the author and source are acknowledged. For more information concerning TI Users' Groups in the Northwest Ohio area, contact:

Robert Peters
President, OH-MI-TI
225 S. Wheeling
Oregon, OH 43616
(419) 693-7934

(419) 385-7484
TICOMM BBS
>>> 24-HRS <<<
SYSOP
> MILLS - TURNER <

Don Turner
President, New Horizons
5533 Fleet Street
Toledo, OH 43615
(419) 537-1454

Meeting; 13 Jun '87 Sat
Unity Church Secor Road

THE NEWSLETTER STAFF

Bill Tiep	Kent Sheets
Phil Bennis	Marilyn Schafstall
Dave Burkette	Earl Hoffsis

LOCAL CONTRIBUTIONS BY;

KENT SHEETS BILL SAGER STEVE PATTERSON

Bill Sager
612 Meadow Springs.
Maumee 43537
OHIO

Edmonton 99'ers Computer
PO Box 11983 Users
Edmonton, Alberta
Canada T5J 3L1

THE PRESIDENTS PAGE

NEW HORIZONS NEWS

by Don Turner

This months joint meeting is getting close, I hope everyone is planning to attend. If you plan to attend, get ready for an exciting meeting with the members from OH-MI-TI. Also this month we are featuring a TI video after that the swap-n-shop will begin. If you are planning to bring some goodies to add to the goodie table and didn't get a chance to let us know, don't worry, bring it anyway I'm sure everyone will enjoy it. Both clubs will supply the soft drinks and coffee. Anyone wishing to help set up please come early. Also we will need some help tidying up after the meeting. If you have any equipment,modules,software etc bring it maybe you can strike a bargain.

This months *meeting* will be on Saturday June 13 at Unity Church on Executive Pkwy. Executive Pkwy can be accesed off of Central near Sears or of of Secor next to Wendy'. The meeting will start at 2:00 PM.

The software the club ordered (MEMO CAL) is in now. We need someone from the club to demo it at the September meeting. Anyone wishing to do the demo please let *me know* and I will get the software to you. As in the past there will be no July and August meetings, however we will resume the meetings in September. This includes the newsletters as well.

In closing I would like to say thanks to all who have helped in the first half of this year and to wish everyone a safe and pleasant summer !!

Don

PLATO

Kent Sheets 6/87

OH-MI-TI

After last months demo of the Plato software, and the Supercart loader for Plato software, several members indicated their interest in Plato.

Anyone who is interested in the Plato learning series just give me a phone call or drop me a note indicating your interest.

KENT SHEETS 1673 N.Curtice Rd.
Curtice, Ohio 43412 836-7877

PRESIDENT'S CORNER

by BOB PETERS
OH-MI-TI

The meeting this month will be a joint meeting with New Horizons at Unity Church off Secor Road on Saturday, June 13 at 2 P.M. Everybody that is going to attend is asked to bring a snack or dish of food as we are having a potluck lunch at this meeting. The meeting will also consist of a swap meet, so bring whatever you have to trade or sell. A video tape will also be shown for those interested.

I would like to thank Bud Mills for his help with my system. The CorComp 9900 is back and running just fine. The ramdisk is completed, with the Captain's Wheel exp. box to finish wiring and test.

There will be no club meetings in July or August. The next regular meeting will be Sept. 11, 1987 at Oregon #2 Firestation, at 7 P.M. At this time I would like to wish you and your families a safe and happy summer.

WARNING!

Kent Sheets 6/87

OH-MI-TI

This machine is subject to breakdowns during periods of critical need.

A special circuit in the maching called a "gritical detector" senses the operator's emotional state in terms of how desperate he or she is to use the maching. The "critical detector" then creates a malfunction proportional to the desperation of the operator. Threatening the machine with violence only aggravates the situation. Likewise, attempts to use another machine may cause it to also malfunction. They belong to the same union. Keep cool and say nice things to the machine. Nothing else seems to work.

Never let anything mechanical know you are in a hurry.

(Sorce unknown)

SUPERCART NWO ASSISTANCE

Kent Sheets 6/87

OH-MI-TI

Several user group members had mentioned that they were interested in building a supercart, BUT had no idea of what to do or how to do it.

Well now you don't have any excuse. Paul Martin of New Horizons has volunteered to assist any user group member who is interested in building a supercart. He has some extra cartridges and other assorted parts. Some of you have already aquired the E/A Chip and,a empty module. Not to worry if you Haven't. What do YOU have to do? Give Paul a phone call or drop him a note today, stating that you are interested, what parts you already have or need, and a time when you would be willing to build your Supercart. It is that easy.

There is a great vareity of super software for the Supercart.

Contact: PAUL MARTIN, 2260 LEHMAN, TOLEDO,OH 43611 729-5911

DO IT TODAY.........

YOU WILL GET A KICK OUT OF YOUR USERS GROUP

COMMON SENSE Part I. The CARE and FEEDING of the 99/4A
or I have met mv enemies and they
are kevboard bounce and lockup'
by Ino Itall

Subheading 1.: Smoke gets in my eyes, or Cooking bacon in the computer room is a NO NO !

WHY ?!! Because greasy, oily smoke insidiuosly deposits itself on every vital part of your beloved 99/4A computer, including the kevboard contacts and the module port lands. Once there, no way to remove it exists except to place your console in the washing machine with half cup of Tide detergent. As you may surmise, the cure is far worse than the desease in that event. "AHA'" you may say, "I do not cook bacon in my computer room, what is this nuttv Italian talking about?" To which Ino answers " BUT you do smoke, and that is just as bad, if not worse!!"

Tobacco Smoke actually deposits some ash along with an oily brown residue. While it is possible to remove this from module and module receotical lands, the keyboard bounce resulting from such deposits on key contacts means a complete new keyboard is the only "fix".

Subheading 2.: "My days and nights in the Kansas Dust Bowl -or- Close the door, Mom, they're coming through the windows."

Dust, while eventually deleterious to the console, is unavoidable. Its effects can be delayed by always using a DUST COVER on the console when not in use, but sooner or later it must be removed. There are two good ways. The first is preventive maintainance. But either a set of soft bristle artist brushes or cosmetic brushes, and regularly remove dust from between the keys, and generally keep the console clean. The second is to blow the dust out with a jet of compressed air. First try this without disassembling the Console, if that doesnot work, more drastic measures are required.

"Compressed Air ??" you may ask. "Where do I get compressed air??". The answer is many and several places: The exhaust port of the household vacuum cleaner is one, and that air is filtered. Some good old Italian (What! you thought I was going to say American?) ingenuity can fit a cheap aluminum funnel onto the end of the hose and you have a great dust ejector. Another is to buy a lever/piston type oil can, and just do not put any oil in it. This gives a nice needle shaped air jet ideal for delicate places. A last resort is lungs and puckered mouth 'POOOF'.

Subheading 3.: Coffee, Cocoa, Coca Cola, Chocolate Chip Cookies, Pizza, Pie and Pasta- or Krumbz in the Kon-sole.

Computer keyboards and foods, both liquid and solid, do not mix. A cup of Cocoa placed on top of the PE Box will surely find its way into the Disk Drives. The point is made, let's not beat it to death. Starvation is preferrable to Pesto Sauce under the FCTN key !!

Subheading 4.: your computer and the electric power utility or It is possible to get an electrical engineering degree bv mailorder from a Carribean University.

The following common sense rules are violated at your computer's jeapardy:

(a)Buy or build a good quality surge protector for your AC line to the computer.

(b)Do not use your computer during electrical storms, in fact disconnect it completely from the power by unplugging it. Unplug when you go away on a trip. We know, personally, of one business and one friend that lost expensive computers to lightning hits.

(c)Your modem is not immune from spikes over the phone line, disconnect the modular plug when not in use or get one of the special surge protectors for modems.

(d) Static electricity can produce sparks of incredible voltages that will destroy computer chips and parts. During the dry winter heating season, have a humidifier going in your computer room. If controlled temperature and humidity is good enough for the professionals, why not you? Also, solder a piece of multi-strand lamp wire to a small metal socket. and ground it on the center screw of your wall socket. Touch it before you begin using your computer, after you have walked around the room (especially if it is carpeted) and frequently while using your computer. This may sound like overkill but it is better than killing you computer.

* NOTE : This article has been reprinted from CALL SOUNDS- Newletter of the Central Westchester 99'ers, Oct 1986.

Becuase of the resent problems occuring with several TI systems in the area (Including TI-COMM), I thought it would be good to review some very basic ways of taking care of your computer. Steve

VALIDATING DATES IN XB, CODE AND ALITTLE HISTORY

by Rod Cook
OH-MI-TI

Entering a date as data into a program can be common particularily when inputing information into a database of some kind. But how do you validate the date so that there are between 1 and 12 months, there are no more than 30 days in September or February 29 is only accepted in a leap year? This article will present an algorithm by which dates can be varified, especially with regards to leap years. But as a precursor to the software, I can't resist the opportunity to talk about the history of our current calander in the hopes that you will find it as interesting as I have.

The Egyptian calander was adopted in Rome in 45 B.C. by Julius Ceasar and accordingly it was called the Julian Calander. It was based on a solar year of 365.25 days and every 4th year was a leap year to makeup for the accumulated quarter day. The first month of the Julian calander was March. The number of days alternated between 31 and 30. The last month of February contained only the number of days needed to finish the year at 365. July is named for Julius Ceasar and contains 31 days, befitting an emperor. Augustus Ceasar, not to be outdone, named August after himself and declared it to have 3) days also thus messing up the scheme of alternating months of 31 and 30 days. It is interesting to note that in this calander, September, October, November and December are the 7th, 8th, 9th and 10th months, their names being derived from the Latin septem, octo, novem and decem for seven, eight, nine and ten respectively.

The actual year is 365.242196 days long, which is slightly less than 365.25 days. This may not seem like much but it amounts to 11 minutes 14 seconds per year. By the year 1500 this amounted to 11 days of error. Religious holidays which were based on astronomical events were getting out of sequence with the seasons. Pope Gregory XIII commissioned the calander be reformed to correct for these errors. The result was the Gregorian calander which we commonly use today. Under this new calander, January 1 became the first day of the year. Only those century years that are evenly divisable by 400 will be leap years. Leap years occur every 4th even year beginning at the century year (1600, 1700, etc.). With some calculation you can see that even with a leap year every 4th year, after 100 years a leap year needs to be skipped to keep the error to less then a day. But after 400 years the leap year must be included in the century year to maintain the accumulation of an even smaller error. So the century years of 1600 and 2000 will have a leap year and all others will not. It will be interesting to see how todays' canned software will handle Feb. 29, 2000! Finally, to make up for the accumulated error of 11 days, The Church declared that Oct. 15, 1582 would follow Oct. 4, 1582. Well, this was not accepted everywhere. It wasn't adopted in the American Colonies until 1752 when Sept. 14 followed Sept. 2.

Following is a listing of an XB program to input and validate dates:

```
100 CALL CLEAR
110 Y=1987 :: M=1 :: D=1
120 DISPLAY AT(10,6):USING "THE YEAR:    "&"####":Y
130 DISPLAY AT(12,6):USING "THE MONTH:    "&"##":M
140 DISPLAY AT(14,6):USING "AND THE DAY    "&"##":D
150 ON WARNING NEXT
160 ACCEPT AT(10,19)BEEP VALIDATE(DIGIT," ")SIZE(-4):Y
170 IF 1752>Y THEN 160
180 IF Y/4-INT(Y/4)=0 AND
      (NOT Y/100-INT(Y/100)=0 OR
       Y/400-INT(Y/400)=0)
    THEN LY=-1 ELSE LY=0
190 ACCEPT AT(12,21)BEEP VALIDATE(DIGIT," ")SIZE(-2):M
200 IF M<1 OR M>12 THEN 190
210 ACCEPT AT(14,21)BEEP VALIDATE(DIGIT," ")SIZE(-2):D
220 IF D<1 OR D>31 OR
      (NOT LY AND M=2 AND D>28) OR
      (LY AND M=2 AND D>29) OR
     ((M=4 OR M=6 OR M=9 OR M=11)AND D>30)
    THEN 210
230 ON M GOSUB 500,510 520,530,540,550,
               560,570,580,590,600,610
240 DISPLAY AT(18,8):A$&STR$(D)&", "&STR$(Y)
250 GOTO 120
500 A$="JAN." :: RETURN
510 A$="FEB." :: RETURN
520 A$="MAR." :: RETURN
530 A$="APR." :: RETURN
540 A$="MAY " :: RETURN
550 A$="JUN." :: RETURN
560 A$="JUL." :: RETURN
570 A$="AUG." :: RETURN
580 A$="SEP." :: RETURN
590 A$="OCT." :: RETURN
600 A$="NOV." :: RETURN
610 A$="DEC." :: RETURN
```

The varification is in line 170 where the value of year is confirmed to be greater than 1752. If you will be dealing with dates before 1752, then you will have to add additional code to adjust for the 12 days added in 1752 here in the Colonies.

Line 180 determines if the year is a leap year and sets a flag, LY, accordingly. This flag will be used to test the value of day later on. LY is set to -1 if the leap year test is true and 0 if the leap year test is false. The section titled "Logical Expressions", page 42 in the TI XB Manual will explain why -1 is used for a true condition and 0 is used for a false condition. A leap year occurs when the year is evenly divisable by 4 AND it is not a century year except when the century year is evenly divisable by 400. A century year occurs when it is evenly divisable by 100. The test in line 180 is true (LY=-1) when both expressions on either side of the Boolean AND are true. Y/4-INT(Y/4)=0 is true when the year is evenly divisable by 4. The expression (NOT Y/100-INT(Y/100)=0 OR Y/400-INT(Y/400)=0) is true when the year is not evenly divisable by 100 OR it is evenly divisable by 400.

Line 200 tests if the value of month is between 1 and 12, (only if life were always this simple).

Line 220 tests the value of day. The value of day is not accepted when any of the following four tests are true:

1. day is less then 1 OR greater tneo 31,

OR

2. it is not a leap year AND month is 2 AND day is greater than 28,

OR

3. it is a leap year AND month is 2 AND day is greater than 29,

OR

4. month is 4 OR 6 OR 9 OR 11 AND day is greater than 30.

It is in tests 2 and 3 where the flag for leap year (LY) is used. The choice of setting LY to -1 or 0 allows the use of LY as a logical expression rather than a mathematical expression.

You now can develope your software with the assurance that it will not error in the year 2000. Speaking of the year 2000, I hope you now realize this century year is special in that it will contain February 29. This has not happened for 400 years and never since the Colonies organized.

WRITERease Mini Review

Kent Sheets 6/87

OH-MI-TI

"WRITERease The ONLY Word Processor/Dictionary for the TI 99/4A." That's whats on the cover of the instructions. The instructions are breif, with the main information in the center, the Command Summary listing.

WRITERease does use D/V 80 files, so it is compatible with TI-Writer files. WriterEASE program includes two disks, the DICTIONARY disk and the PROGRAM disk. The DICTIONARY disk should be backed-up. "The PROGRAM diskette is protected and cannot be backed up." according the the instructions. None of the recent track copy programs I have would touch it. CorComp will provide one backup for the cost of $7.00.

I tried the dictionary on several D/V80 files, checking the spelling for the complete file and also just checking for only one word. This is a 30,000 word dictionary. The screen brings up a list showing where the word it is checking should be in sequence that you can choose from. If you miss the first or second letter of the word you are checking you may never find the correct spelling.

The dictionary allows you to scan thru the dictionary list, going forward or backwards thru the list of words. I did not see any instructions or the ability to add additional words to the spelling dictionary. This would be a major drawback of this program if you wanted to add special words or proper names to the spelling dictionary list. The dictionary file is 358 sectors in INT/FIX 255 format.

The COMMAND KEYS are a combination of CTRL and FCTN keys. Unlike TI-Writer or TK-Writer or other similar word processing programs WRITERease only uses a few of the numeric keys with CTRL/FCTN.

The following are some of the command keys used in WRITERease.

TEXT MANIPULATION
CTRL-F:Find word or string
CTRL-K:Kill end of line
CTRL-L:Load document from dsk
CTRL-M:Merge text file
CTRL-P:Print Document
CTRL-R:Replace word or string
CTRL-S:Save document to disk
CTRL-2:Paragraph reform
FCTN-M:Text COPY,DELETE,MOVE

CONTINUED ON PAGE 5

BIT BY BIT
A BOOK REVIEW

by Rod Cook
OH-MI-TI

If your interested in the history of computers, particularily the hardware aspects of computer development, then I recommend the book titled "BIT by BIT - An Illustrated History of Computers" by Stan Augarten. The price of the book is about $17. Rarely do I find a book that I can't put down when it's nice outside, but this was one.

The book begins tracing the history of mechanical calculators from the Abacus, Napier's Rods, Logarithmic sliding rules, and the Pascaline developed by Blaise Pascal in 1644. The discussion continues to the early to mid 1800's and talks about the analytical engines of Charles Babbage. Around the same time Joseph-Marie Jacquard invented a loom that was controlled by pre-programmed punched tablets. The analytical engines eventually led to mechanical adding machines and calculators of the early 1900's. Punched cards were used to tabulate information in the 1890 census and was considered a technical breakthrough with regards to data handling and storage.

Through the mid 1940's, computing machines became increasingly sophisticated and complex. IBM topped them all with the MARK 1. It was 8 ft. tall, 51 ft. long and 2 ft. thick, weighed 5 tons and used about 750,000 parts. It was an electromechanical machine in that it used electronic devices such as relays as well as mechanical devices such as motors and rotating shafts. The MARK 1 was the first programmed controlled calculator. But it operated in the decimal numbering system and could not perform conditional jumps (IF statements).

Around the late 1930's, at Iowa State University, John Vincent Atanasoff and graduate student Clifford Berry invented the first digital electronic calculator specifically designed to solve linear simultaneous equations (what ever they are!). This machine used all electronic components, no relays, and capacitors were used as storage elements. It used the binary numbering system, timing of the various parts of the device were controlled by a system clock and it used punched cards for input/output. This device has become known as the ABC for Atanasoff-Berry Computer. Although the patentability was recognised by Atanasoff and Iowa State, the device was never patented.

John William Mauchly and Atanasoff exchanged ideas in 1941. Mauchly was commissioned by the Ballistics Research Laboratory to construct an electronic calculating machine. It seems that manual calculation of ballistics data could not keep pace with the development of new weapons. Mauchly and J. Presper Eckert constructed a machine called ENIAC at the Moore School of Electronics in the University of Pennsylvania. ENIAC was not the product of a scientists' or engineers' frustration with solving equations with mechanical calculators, but of a military imparitive during a national crisis. ENIAC contained 17468 tubes, 70000 resistors, 10000 capacitors, 1500 relays and 6000 manual switches. It was eight feet high, eighty feet long, weighed thirty tons, and consumed 174000 watts of power. ENIAC operated in the decimal numbering system and is considered in a modern sense as a very fast program-controlled calculator. But it bridged the gap between calculators and computers.

The next computer, EDVAC, is the direct descendant of ENIAC. EDVAC was the first stored-program computer and it used binary numbers. The discussion continues with further developments that eventually led to the large mainframe computers promenant in the 60's and 70's.

Of particular interest is a patent infringement lawsuit that resulted when the Patent Office issued Mauchly and Eckert's ENIAC patent in 1964. Sperry Rand became owner of the patent and immediately proceded to notify the other computer companies that they were violating the ENIAC patent. Sperry offered patent licenses at 1.5% of the selling price of their competitors' equipment. No one accepted Sperry's offer. Sperry sued Honeywell and Control Data. The lawyers for these two companies eventually learned of Atanasoff's ABC and showed that many of Atanasoff's ideas embodied in the ABC were used in ENIAC. The patent was invalidated in 1973. Eckert and Mauchly were stripped of their legal claim to the invention of the computer, but it was their genious that led to the stored program computer and the birth of the computer industry.

The last fourth of the book deals with the development of the integrated circuit (Jack Kilby at TI and at Fairchild), the minicomputer (Digital's PDP-1) and finally the early microprocessors (Intel's 4004 chip set) around 1970. Further developments towards the personnal computer are outlined in discussions about Edward Roberts, William Gates, Stephen Wozniak, Steven Jobs and A. C. Markkula.

This books appears to be well researched and the facts accurately presented. I am sure you will agree that once you've read this book your time has been well spent.

NORTHWEST
OHIO 99ER
NEWS

FCTN-0:Purge Document
FCTN-1:Delete one character
FCTN-3:Delete line
FCTN-8:Insert line

FCTN-H:HELP Menu

CTRL-A:DICT Check entire Doc
CTRL-C:DICT Check one word
CTRL-D:File Directory
CTRL-O:Oops!
CTRL-U:Set printer codes
CTRL-3:Set screen and character colors

Since most users who will be buying this program already have access to and are already using TI-Writer type programs, I am surprised that the developer didn't at least keep the CTRL and FCTN keys that are used with the other programs. The above contral 'keys are nice for those users who want to learn their association of key to use, but should be in addition to the standard that is already being used. As I mentioned above the other concern is the ability to update or add another distionary.

This program was purchased by the users group for a demonstration at one of our upcoming meetings. If you have any questions prior to purchasing this program, just give me a call. You can even use it if you want to check it out from the users group.

NEW ADDRESS

by STEVE R PATTERSON
NEW HORIZONS

NO, I have not moved to Oklahoma of Idaho, but I have stumbled upon a very interesting location (Address) in VDP memory that is accessable from Extended Basic using the CALL LOAD and CALL PEEK commands. It is address -31952.

I discovered the address in a Genial Traveler program on the first disk of the diskazine. It was used to delete the program in memory and then the program ended. I began to fool around with this because of my great curiousity.

What I found, I think, is something that will benefit all that have the 32k Memory Expansion.

First, I would like you to try a little experiment to demonstrate the use of the Call Load.

1) Load any program into Extended Basic.
2) CALL PEEK(-31952,W,X,Y,Z)
3) PRINT W,X,Y,Z
4) Write down those four numbers.
5) CALL LOAD(-31952,255,0,255,0)
6) Try to list or run the program.
7) Now, CALL LOAD(-31952,W,X,Y,Z)
W-Z are numbers from call peek in 2.
8) List or run the program.
9) Then, NEW
10) Repeat 6-8
11) Then, press FCTN '+' or 'QUIT'
12) Return to XB and Repeat 6-8.
13) Then, type 'BYE'
14) Return to XB and Repeat 6-8.
15) Then, with widget, Hit Reset.
16) Return to XB and Repeat 6-8.
17) Turn off computer.
18) Return to XB and Repeat 6-8.

If you followed the directions you should have been able to retrieve the program without having to reload the program itself, except for 18, I wasn't serious.

Not only can you delete the lines that are in memory and bring them back with just one statement but you can also delete just several lines from either the end of the program or the begining. I have been unable to delete lines from the center of a program but maybe there will be something in the next newsletter.

Deleting lines begining at end

First you must get the four numbers from the address using the call peek because most programs are not the same length and these will differ with differetn sized programs. Do this by a command in the edit mode:

CALL PEEK(-31952,W,X,Y,Z) :: PRINT W,X,Y,Z

Then count the number of lines you wish to delete starting at the end and moving towards the begining. Then times that number by 4. Input this number plus 'X' into the call load as 'X'. If the number exceeds 255 then you must add 1 to 'W' and 'X' becomes the amount over 255.

Example:
W=254, X=231, Y=255, Z=13
You wish to delete 8 lines from the end of the program.
8*4=32
231+32=263
263>255 by 13
So, W=255 and X=13
CALL LOAD(-31952,255,13)
If you only wished to delete 2 lines.
2*4=8
231+8=239
239<255
So, W=254 and X=239

CALL LOAD(-31952,254,239)

Deleting lines begining at start

First you must get the four numbers from the address using the call peek Do this by a command in the edit mode:

CALL PEEK(-31952,W,X,Y,Z) :: PRINT W,X,Y,Z

Then count the number lines you wish to delete starting at the first line and moving towards the end. Then times that number by 4. Input this number minus 'Z' into the call load as 'Z'. If the number is below 0 then you must subtract 1 from 'Y' and 'Z' becomes 255 minus the amount under 0.

Example:
W=254, X=231, Y=255, Z=13
You wish to delete 8 lines from the start of the program.
8*4=32
13-32=-9
-9<0 by 9
So, Y=254 and Z=246
CALL LOAD(-31950,254,246)
If you only wished to delete 2 lines.
2*4=8
13-8=5
5>0
So, Y=255 and Z=5
CALL LOAD(-31950,255,5)

I hope that you will find this article of use and I hope I can figure out a method of deleting lines from the center of a program, just not the end and begining. If you decide to work with this Call Load and find a way of deleting lines in the center of a program be sure to let me know or write an article and let everyone know.

Both examples used in this article were based upon a program containing only 10 lines. This Call Load is capable of deleting as many lines as are present in memory.

A REVIEW OF MENU v.06 -

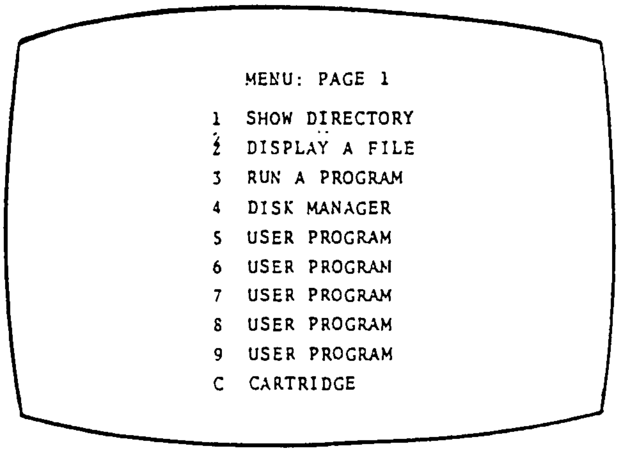
By Mark F. Armstrong
Bluegrass 99 Computer Society, Inc.

MENU V.6.8 is a Randisk Operating System for the New Horizon Randisk. Even if you do not as yet own a New Horizon Randisk, you might read this review to see that the full promise of the TI-99/4A is subtantially fulfilled by this program. We have all read before and this program proves it, the people at TI did not know how good their product is. MENU V.6.8 stretches the TI-99/4A to its limits and in so doing gives the user a truly first class computer system.

> "the full promise of the ti-99/4a is substantially fulfulled."

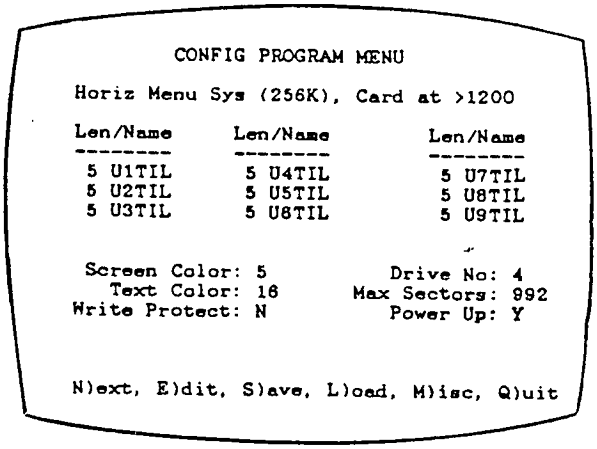
After the program is installed, the user has the option of CALLing A0 or AF. CALLing A0 turns the automatic power up routine On; AF turns it off. The Automatic power up routine takes over the usual power up routine so that you do not see the familiar color bars and TI logo. Instead you are presented with a Menu of 9 choices. The first choice allows you to View the Directory of any disk drive. This directory can be displayed on the monitor or sent to your printer using Shift 1 rather than the single Keystroke 1. In addition, it is important to note that a disk directory can be printed to a disk drive. This will allow you to create a disk library quite simply. The second choice allows you to display any text file, e.g., a TI Writer file. This file may also be printed by entering Shift 2 rather than just the Keystroke 2.

The third choice will Run a Program. When the keystroke 3 is entered, a default of DSK1.UTIL1 appears on the screen. If this is the name and disk drive desired, simply press enter. If a different program is desired, the default drive and filename can simply be overwritten and entered. One of the very handy features of this third option is that the default drive and filename can be easily changed.



```
                MENU: PAGE 1

           1   SHOW DIRECTORY
           2   DISPLAY A FILE
           3   RUN A PROGRAM
           4   DISK MANAGER
           5   USER PROGRAM
           6   USER PROGRAM
           7   USER PROGRAM
           8   USER PROGRAM
           9   USER PROGRAM
           C   CARTRIDGE
```

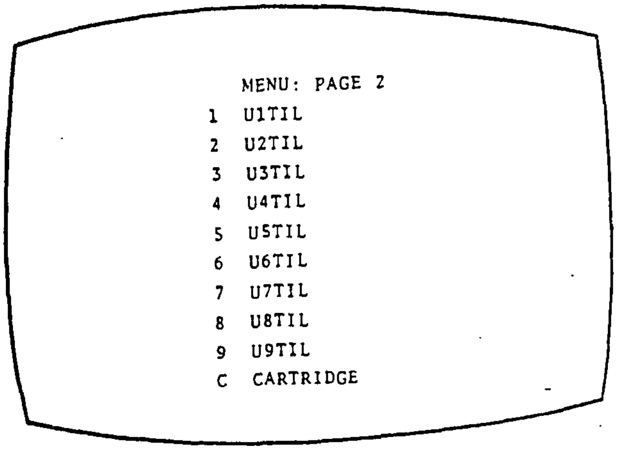
When you select choice one, View the Directory, after the directory is displayed on the CRT, by pressing the space bar, each filename in the directory is sequentially highlighted. When the file which you want to run is highlighted, simply press enter. Then, when the third choice is selected, Run a Program, the default name will be the disk drive and file name previously highlighted in the View the Directory choice. The only limitations on the Run a Program choice is first, that the appropriate cartridge must be in the cartridge port or selected in the widget. Second, only Option 5 Assembly language programs will run. I have had only one assembly language program which will not run in choice three. I have had some Extended Basic programs which I have had to load twice to get a successful load and run. There are technical notes in the documentation which come with the program. In these notes, assembly language program requirements and other technical details are discussed. One program which will not run without some modification is Rapid Copy; however, the fix is clearly set forth in the technical notes and should not prove to be difficult to execute.



```
                 CONFIG PROGRAM MENU

      Horiz Menu Sys (256K), Card at >1200

      Len/Name        Len/Name          Len/Name
      --------        --------          --------
       5 U1TIL         5 U4TIL           5 U7TIL
       5 U2TIL         5 U5TIL           5 U8TIL
       5 U3TIL         5 U6TIL           5 U9TIL


        Screen Color: 5              Drive No: 4
          Text Color: 16          Max Sectors: 992
      Write Protect: N               Power Up: Y


      N)ext, E)dit, S)ave, L)oad, M)isc, Q)uit
```

In the third choice, Run a Program, you can type in the program name, use the default value or create your own default value. Each of these methods require several key strokes. By customizing choices five through nine, you can load an assembly language program assigned to the selection. For example, my selection 5 is BA-Writer and choice 6 is Turbo, a copy program. To select one of these programs, merely enter the number and the program loads with this one keystroke. This customization requires that the source code be modified and then assembled and the ROS reloaded. The procedure for accomplishing this is set out in the documentation which is clear and even a user who does not program in assembly language can do it.

The documentation consists of 17 pages. It is extensive, clear and complete. A careful reading of the documentation will lead though all of the features of the ROS. The startup procedure involves loading a Config program. There is an extended basic loader or the program can be loaded through E/A option 5. Once the program is loaded, you are presented with 3 rows of "2 U1TIL" though "2 U9TIL". By entering the edit mode, Keystroke E, you can change these values and names. The names correlate with the second page of the Menu program. In addition to the Menu display discussed above, there is another Menu page which is accessed by pressing the space bar. Then by entering the appropriate keystroke, i.e., 1 through 9, the program selected can be loaded. The program may be either an assembly language program - E/A 5 - or extended basic program. The proper cartridge must be in the cartridge slot, e.g., XB cartridge to run an XB program. The values and names which must be edited in the Config program are the "2 U1TIL" through "2 U9TIL". Change the "UnTIL" through the editor to the name of the program you want to appear on the second page of the Menu. Program names longer than 5 characters must be shortened. The program can then be loaded and run by entering the second page of the Menu display (by pressing the space bar) and then entering the appropriate keystroke, 1 through 9. In addition, the desired program can also be loaded by CALLing it. The number value in "2 UnTIL" refers to the number of characters in the name which will be used in the CALL.



```
        MENU: PAGE 2
    1   U1TIL
    2   U2TIL
    3   U3TIL
    4   U4TIL
    5   U5TIL
    6   U6TIL
    7   U7TIL
    8   U8TIL
    9   U9TIL
    C   CARTRIDGE
```

For example, if you have a program named EMULATOR; Shorten the name to EMULR. If you leave the value of "2 EMULR" then the CALL will be CALL EM. This will load and run the Emulator program.

In addition to editing the second page of the Menu, screen color, disk number, automatic power up routine, maximum number of sectors and write protection can be set. This program automatically adjusts for the size of the ram disk you have. The program will also search for multiple NH ramdisks so that you can install the operating system in the ram disk with the lowest address. After setting the Config program to the values and text desired, merely exiting the program will install it. If desired the Configuration can be saved so that it can be reloaded for future use.

There are several other useful features the ROS program offers. First, on either page of the Menu, pressing C will address the cartridge in the cartridge port. If you have a super cart, pressing SC will address it. Second, pressing T will display the time if you have the Corcomp Triple Tech card. Third, you can return to the color bars by simply pressing Function 9. In the case of a cartridge like the Navarone DBM which has a menu in the cartridge, this allows you to enter the menu. Merely pressing C in this call will not allow you to address the full menu but only one selection of the cartridge menu. Four, you can enter console basic by entering the keystroke B. Fifth, the keystroke D will blank the screen. The documentation suggests that the program's author put this function in just to see if it would work and then simply left in it the final version. This is a useful function. There are numerous times when it is necessary to leave the computer on but unattended. Blanking the screen preserves the working environment without keeping the CRT continually on. Six, the automatic power up routine can be interrupted simply by holding the space bar down during the power up routine. This will return to the color bars and TI logo.

This description does not do full justice to this program. The use of single keystrokes will spoil the user. The excellent documentation gives even the novice user the chance to make full use of the program's features. The program allows full flexibility within a framework which supports any application the user desires. This is an excellent program which will give your TI-99/4A many additional years of useful life. One note of caution. There are some editions of MENU V.6.8 which have not been completely debugged. If you get one of these, do not give up. Keep trying until you get a good copy. The wait and aggravation of loading and then unloading a flawed copy will be worth it.

"the people at texas instruments did not know how good their product is."

One final note: John Johnson and Mike Ballmann have given this program to public domain. It is not even fairware. If you are impressed with it, the author request you make a contribution to the Miami Users Group. Give the contribution - we need to encourage Johnson and Ballmann.

MULTIPLAN™

Manipulating Dates

In addition to having a specific application, the spreadsheet covered in this article uses many different functions, including LOOKUP, MID, IF, VALUE and INT. If you've had difficulties with any of these functions, give this spreadsheet a try.

The spreadsheet is based on manipulating a date in the form MO/DA/YR. The specific application is for a receivable aging, very similar to the one we did in Extended BASIC several months ago.

Though there are several obstacles in setting up a spreadsheet for working with agings, the primary problem is that some years have a leap day. There is a leap day every four years except once every hundred years, such as the years 1900 and 2000. The calendar we use begins with the year 1582. Therefore, all calculations we will use will relate to the year 1582 and take into account all leap days. Our spreadsheet will work only with dates in the 1900's, but a few modifications would allow working with any year after 1582.

Here is the accounts receivable aging report that we will show how to build in this article:

```
Today's Date: 03/25/85

Invoice #    Date     Customer Name Amount  0-30     31-60    61-90   Over 90
---------    ----     ------------- ------  ----     -----    -----   -------
12           08/14/84 ABC Company   $124.00                           $124.00
21           02/28/85 XYZ Company   $175.00 $175.00
5            03/01/84 LMN Company   $147.00                           $147.00
19           01/05/85 RMM Company    $26.00                   $26.00
20           02/21/85 SSS Company   $377.00          $377.00
```

Note that the data on the left portion of the report is keyed in, while the spreadsheet completely calculates the aging column in which an invoice is to appear. Setting up this type of report in a spreadsheet permits flexibility in formatting the report. The finished report can then be loaded into TI-Writer for minor formatting changes that are either difficult from the spreadsheet or are not of a recurring nature. In building the spreadsheet, we cover the input for the first invoice only. Using COPY DOWN will establish the remainder of the invoice area (change the data).

```
                   1                        2                      3
 1
 2 "Today's Date:"                 "03/25/85"
 3
 4 "Invoice #"                     "Date"              "Customer Name"
 5 "---------"                     "----"              "-------------"
 6 "12"                            "08/14/84"          "ABC Company"
*****************************************************************************
             4             5              6               7
 1
 2
 3
 4 "Amount"        "0-30"         "31-60"         "61-90"
 5 "------"        "----"         "-----"         "-----"
 6 124             IF(RC[+13]<31,R IF(AND(RC[+12]< IF(AND(RC[+11]>
                   C[-1]," ")     61,RC[+12]>30), 60,RC[+11]<91),
                                  RC[-2]," ")     RC[-3]," ")
*****************************************************************************
```

```
        8                 9                10                11
1                 "YY"              "MM"              "DD"
2                 VALUE(MID(RC[-7 VALUE(MID(RC[-8 VALUE(MID(RC[-9
                  ],7,2))+1900    ],1,2))         ],4,2))
3
4 "Over 90"
5 "-------"
6 IF(RC[+10]>90,R VALUE(MID(RC[-7 VALUE(MID(RC[-8 VALUE(MID(RC[-9
  C[-4]," ")      ],7,2))+1900    ],1,2))         ],4,2))
******************************************************************************
        12                13               14                15
1 "ce"            "ko"            "vr"            "adder"
2 RC[-3]-1583     INT(RC[-1]/100) INT(RC[-2]/4)   LOOKUP(RC[-5],R
                                                  1:247C19:R1:247
                                                  C20)
3
4
5
6 RC[-3]-1583     INT(RC[-1]/100) INT(RC[-2]/4)   LOOKUP(RC[-5],R
                                                  1:247C19:R1:247
                                                  C20)
******************************************************************************
        16                17               18
1 "leap"          "days"          "days away"
2 IF(AND(RC[-7]/4 RC[-1]+RC[-2]+( R2C17-RC[-1]
  =INT(RC[-7]/4), (RC[-5]+RC[-4]-
  RC[-7]/100<>INT RC[-3])*365)+((
  (RC[-7]/100),RC RC[-3]-RC[-4])*
  [-6]>2),1,0)    366)+RC[-6]
3
4
5
6 IF(AND(RC[-7]/4 RC[-1]+RC[-2]+( R2C17-RC[-1]
  =INT(RC[-7]/4), (RC[-5]+RC[-4]-
  RC[-7]/100<>INT RC[-3])*365)+((
  (RC[-7]/100),RC RC[-3]-RC[-4])*
  [-6]>2),1,0)    366)+RC[-6]
******************************************************************************
        19                20
 1 "LOOKUP"
 2 1              0
 3 2              31
 4 3              59
 5 4              90
 6 5              120
 7 6              151
 8 7              181
 9 8              212
10 9              243
11 10             273
12 11             304
13 12             334
```

Of course, several of the columns must be widened using FORMAT WIDTH. Also, the columns using monetary amounts should be formatted by using FORMAT CELLS. A company name can be added at the top using INSERT ROW, but there are Absolute References in Column 18 that would require a change to match the new Row that is currently R2C17. A company name can also be added by using the instructions on page 86 of the manual to print a file that can be used in TI-Writer, which would also permit other modifications from TI-Writer.

COPYING A CARTRIDGE TO DISK

The following is for non-modem users and was released to the Spirit of 99 through the courtesy of Bud Wright,SysOp of T.A.B.B.S.

TO DO THIS, MAKE SURE YOU HAVE THE T.I. DISASSEMBLER, A CARTRIDGE WITH ROM CHIPS ONLY(YOU CAN FIND OUT BY OPENING IT UP,THEN LOOK IF IT HAS LARGE CHIPS ONLY. IF IT DOES, IT IS ROM ONLY. IF IT HAS ANY SMALL CHIPS, TOO BAD, YOU HAVE GROM WHICH HASN'T WORKED FOR ME YET SINCE IT IS IN GPL)EITHER A WIDGIT OR COVER PIN 1 OF THE CARTRIDGE BEING COPIED WITH TAPE(PIN 1 IS THE RIGHTMOST PIN ON THE BOTTOM SIDE). ALSO, A PRINTER WOULD NOT BE NECESSARY ,BUT VERY CONVENIENT TO HAVE.

---

(1) COVER PIN 1 OF THE CARTRIDGE BEING COPIED(IF YOU HAVE A WIDGIT, THIS IS NOT NECESSARY)

(2) IF YOU HAVE A WIDGIT, PLUG E/A IN SLOT 1, THE CARTRIDGE YOU WANT COPIED IN IN SLOT 2, TI EXTENDED BASIC IN SLOT 3.

(3) MAKE SURE YOU HAVE A TOTALLY BLANK DISK TO COPY THE CARTRIDGE ON.

(4) SWITCH OVER TO THE E/A CARTRIDGE (EITHER WITH THE WIDGIT OR JUST INSERT IT), INSERT E/A DISKETTE 'A',PRESS 2 (3 ON 99/4) TO SELECT LOAD AND RUN, FOR FILENAME TYPE 'DSK1.DEBUG' AND PRESS ENTER. WHEN IT IS LOADED, PRESS ENTER AGAIN, AND TYPE 'DEBUG' FOR PROGRAM NAME.

(5) WHEN THE DEBUGGER PROMPT APPEARS, SWITCH OVER TO SLOT 2(FOR THE CARTRIDGE BEING COPIED) OR INSERT THE CARTRIDGE BEING COPIED. MAKE SURE THE COMPUTER DOESN'T RESET. IF IT DOES, MAKE SURE PIN 1 IS DEACTIVATED AND START OVER. (WITH THE WIDGIT, DON'T PRESS RESET).NOW MAKE SURE THE DEBUGGER IS STILL OPERATIVE (PRESS ENTER A COUPLE OF TIMES).

(6) NOW TYPE IN 'M6000 8000' AND PRESS ENTER. THE SCREEN SHOULD START SCROLLING UPWARD AND DISPLAY DIFFERENT KINDS OF NUMBERS. IF ALL YOU GET IS '6000=00 00 00 00 00 00 00 00 ********' ,ETC. OR ZEROS ONLY, MAKE SURE YOU HAVE A ROM CARTRIDGE ONLY, AND THAT YOU ARE SWITCHED OVER TO THE CARTRIDGE BEING COPIED(OR IS INSERTED). OTHERWISE, YOU SHOULD BE GETTING ALL KINDS OF NUMBERS. YOU WILL PROBABLY SEE THE TITLE OF THE CARTRIDGE IN THE FEW FIRST LINE AT THE RIGHT. NOW LET THE SCREEN DISPLAY THE NUMBERS FOR A WHILE, AND MAKE SURE THAT EVERYTHING IS O.K.

(7) PRESS QUIT, (FCTN =) TO GO BACK TO THE MAIN TITLE SCREEN.

(8) SELECT 2 FOR THE E/A AGAIN, 3 FOR LOAD AND RUN, THEN INSERT THE DISKETTE WITH THE T.I. DISASSEMBLER ON IT, AND LOAD IT.

(9) PRESS ENTER ONCE, THEN TYPE 'START' & ENTER.NOW THE DISASSEMBLER TITLE SCREEN SHOULD APPEAR. NOW SWITCH OVER TO THE CARTRIDGE BEING COPIED OR PLUG IT IN.AGAIN, MAKE SURE IT DOESN'T RESET OR OTHERWISE STOP YOU.FOR STARTING ADDRESS, PRESS '6000'. FOR ENDING ADDRESS , PRESS '6500'. NOW INSERT THE DISKETTE YOU WANT TO COPY IT ON AND TYPE IN 'DSK1.CART1' FOR DEVICE NAME.PRESS ENTER TO START THE DISASSEMBLY PROCESS. THE DISK DRIVE COMES ON,ETC. THE DISASSEMBLED CODE SHOULD APPEAR ON THE SCREEN.
FOR EXAMPLE, THIS MIGHT APPEAR:

```
'61FC    JNE >6F64            >16F4'
```

TO PAUSE DISASSEMBLING, PRESS A KEY TO PAUSE. PRESS IT AGAIN TO CONTINUE. THE PROCESS WILL CONTINUE FOR A WHILE. WHEN THE NUMBER ON THE LEFT='>6500' OR AROUND THAT NUMBER, THE SCREEN WILL STOP SCROLLING.PRESS ENTER TWICE FOR E/A MENU.

(10)

NOW PRESS QUIT. EITHER PLUG IN EXTENDED BASIC OR SWITCH OVER TO IT, AND SELECT EXTENDED BASIC. NOW ENTER THE FOLLOWING PROGRAM:

```
100 CALL CLEAR
110 INPUT "SOURCE FILE?>DSK1.":A$
120 INPUT "OBJECT FILE?>DSK1.":B$
130 OPEN #1:"DSK1."&A$,VARIABLE 80,INPUT
140 OPEN #2:"DSK1."&B$,VARIABLE 80,OUTPUT
150 LINPUT #1:C$
```

```
160 D$=SEG$(C$,6,27)
170 PRINT #2:D$
180 IF EOF(1)THEN 200
190 GOTO 150
200 PRINT "FINISHED."
210 CLOSE #1 :: CLOSE #2
220 DELETE "DSK1."&A$
```

NOTICE THIS CAN NOT BE DONE IN BASIC, SINCE THE 'LINPUT' STATEMENT IS USED.

(11) RUN THIS PROGRAM. FOR SOURCE FILE, INPUT 'CART1'. FOR OBJECT FILE, INPUT 'CART11'. NOW THE DISK DRIVE WILL COME ON, AND IT WILL TAKE A WHILE BEFORE IT IS FINISHED.WHEN IT IS FINISHED , IT WILL SAY SO. IF YOU RECEIVE AN ERROR , CHECK TO SEE IF YOU HAVE THE RIGHT DISK INSERTED AND THAT THE SOURCE AND OBJECT FILES ARE VALID.IF THAT ISN'T IT, CHECK TO SEE IF YOU HAVE ENOUGH DISK SPACE AVAILABLE.IF YOU DON'T,CHANGE LINE 140 FROM 'DSK1.' TO 'DSK2.' IF YOU HAVE A SECOND DRIVE.OTHER WISE, SORRY! YOU MIGHT WANT TO CHANGE LINE 160 SO THAT IT SAVES VARIABLE LENGTH LINES, THUS SHORTENING THE OUTPUT. WHEN DONE,PRESS QUIT AND INSERT THE E/A MODULE OR SWITCH OVER USING THE WIDGIT.

(12) INSERT THE E/A DISKETTE 'A' AND SELECT 1 FOR EDITOR. PRESS 1 FOR LOAD FILE. FOR FILE NAME,USE CART11.

<13> WHEN LOADED,PRESS FCTN-9 FOR THE EDITOR COMMAND LIST(THIS ONE:) *Edit,Tabs,Files,Delete,Insert,Move,Copy, etc. then select R for replace. next type in* V,1000/6/A/ AND PRESS ENTER. NOW PRESS 'Y' WHENEVER THE CURSOR IS ON A JMP 6??? OR A LI 6??? OR A SB,CB,B,BLWP ,LI,AND ALL OTHER ONES EXCEPT TWO-DIGIT(FOR EXAMPLE '6?' OR DATA STATEMENTS. WHEN YOU REACH THE LAST ONE, THE EDITOR WILL BE IN EDIT MODE. PRESS FCTN-9 AND TYPE R FOR REPLACE AGAIN. THIS TIME TYPE IN V,1000/7/B/ AND PRESS ENTER. DO THE SAME THING AS THE LAST TIME. WHEN FINISHED, SAVE THIS IN VARIABLE 80 FORMAT ON THE EDITOR COMMAND LIST.

(14) DO STEPS 8-13 AGAIN, CHANGING THE STARTING ADDRESS FOR DISASSEMBLY PROCESS SECOND TIME AROUND TO 6500 AND THE ENDING ADDRESS TO 7000 THE FILE NAME TO CART2 , SHRINKED VERSION TO CART22 THE THIRD TIME 7000-7500,CART33, THE FOURTH TIME 7500-8000,CART44. WHEN YOU HAVE DONE THIS, GO ON TO NEXT STEP.

(15) TYPE IN THE FOLLOWING PROGRAM IN EDITOR OF THE E/A.

```
DEF START
AORG >A000
START
COPY 'DSK1.CART11'
COPY 'DSK1.CART22'
COPY 'DSK1.CART33'
COPY 'DSK1.CART44'
END
```

SAVE THIS AS CART55 IN VARIABLE 80 FORMAT NOW LOAD THE ASSEMBLER (OPTION 2) AND FOR SOURCE FILE NAME USE CART55,OBJECT FILE CART66. NOW ASSEMBLE UNDER THE 'R' OPTION . WHEN IT IS FINISHED, JUST LOAD AND RUN IT AND FOR PROGRAM NAME USE START. IF IT DOES NOT LOOK RIGHT, USE THE EDITOR TO LOOK AT THE FIRST AND LAST LINES OF THE DISASSEMBLED CODE TO SEE IF THE NEXT FILE NAME HAS IDENTICAL ONES OR IF ANY LINES ARE LEFT OUT. WHEN DONE WITH THIS, ASSEMBLE AGAIN, CHECK AGAIN UNTIL IT WORKS.