

# THE NATIONAL NINTY-NINER

VOL II - NO. 1 - JANUARY, 1985

MONTHLY NATIONAL NEWSLETTER OF THE

THE 99ER'S ASSOCIATION  
3535 SO.H ST., #93  
BAKERSFIELD, CALIF. 93304  
(805) 397-4361  
DON VEITH - EDITOR/PRESIDENT

CREATED FOR TI 99/4A HOME COMPUTER OWNERS AND USERS GROUPS

## ANNOUNCEMENTS

### OUR SECOND YEAR By Don Veith

This issue marks the beginning of our second calendar year. The newsletter has grown at a steady pace with additional subscriptions appearing from both Users Groups and individuals. The staff has been increased from an early total of five persons to our present 26. This number includes Coordinators, Advisors, Staff Writers, Librarians, Secretary, Printer/Coordinator, Writer/Coordinator, and the Editor.

The past 15 months, since TI withdrew from the home computer market, have been eventful and unusual. The most interesting aspect of this period is that more TI Users Groups have been started than were disbanded. Our current Users Group list has 260 plus clubs on it. Several have given notice that they are reorganizing, but still exist. We have removed seven from our list in the 15 months. Two (2) were converted to commercial Users Groups (?) and ultimately were discovered and exposed by other owners and Users Groups. The remaining five simply disbanded due to a lack of interest by their members.

A number of firms have ceased making products for the 99/4A. In this area, many new firms with totally new and innovative products have appeared. The useful life for the 4A has been extended beyond what the most optimistic "diehard nuts" had envisioned possible. Some excellent commercial software has appeared during this period, for example, the TI-Writer Spelling Checker.

The greatest explosion of new programs has appeared in the public domain sector. We all owe a tremendous THANK YOU to the many talented individuals who have provided these excellent programs!! The programs are developed at their own expense and time. Many of the programs could have been simply left in that person's software library and never distributed except to a few personal acquaintances. Let's hope this outpouring of quality Public Domain software continues to appear for the benefit of every 99/4A owner.

This issue has a slightly different content than our regular newsletter. Two of our regular writers, Jim Peterson and Edgar Dohmann, appear with their regular contributions. The remainder of the staff will be back next month. John Clulow presents an excellent article on a Real Time Clock and a review of the CorComp 9900 Micro-Expansion System. The Holidays pre-empted my article on the Morning Star CP/M Card. Darryl Ingold, a Staff Analyst and President of TEX-BUG, adds reviews on Bon Vivant and the CorComp 9900 DS/DD Controller Card, and Homework Helper +. The remainder of the issue is composed of tips from Users Group newsletters and articles by new contributors to this newsletter.

On behalf of all of our staff members, We want to wish each of you the very best for 1985. Thank you for your continued support for our efforts to keep TI-99/4A Home Computer owners informed.

We plan, with this issue, to get ourselves back on a timely schedule delivering the newsletter in the month it is dated. If we are late, remember that we do not get paid for our efforts and the newsletter staff is composed TOTALLY OF VOLUNTEERS!! Try and remember that if the newsletter should arrive late sometime in the future.

### MEMORY AND DISK ANALYZER

Miller's Graphics has a new Memory and Disk Diagnostics program available for the 99/4A. The program is compatible with **BOTH** the TI and CorComp Disk Controller Cards. The Diagnostics program does a complete memory check for VDP, Scratch Pad, and Memory Expansion RAM reporting any errors that are located. A complete disk diagnostic check is performed including:

1. Check and display the disk drive motor speed in RPM's.
2. Check the bit map for bad sectors reporting failing numbers.
3. Check the disk for fractured files and report their names.
4. Seek out any sector or track on command - read or write to either with a verify.
5. Allow the selection of different head step times to test your drives and determine the optimum timing.
6. Edit any sector in hexadecimal or ASCII with full Screen Editor capabilities.

This utility will eliminate any questions you may have about the status of your system's memory, diskettes, or disk drives. The cost for this utility program is \$19.95 plus \$1.50 for shipping and handling. The Memory and Disk Diagnostics program should be available by the time you are reading this article. Obtain your copy from:

MILLERS GRAPHICS  
1475 W. CYPRESS AVE.  
SAN DIMAS, CA 91773

### MINI MAG 99 - A NEW MAGAZINE FOR TI-99/4 OWNERS

A new magazine for our computer will be making its debut on January 20, 1985. A complimentary issue is available for interested parties. The first issue will carry a date of February, 1985. Contact Leslyn Tepper at the address below to obtain your complimentary copy:

SOS Publications  
Suite 203  
21777 Ventura Blvd.  
Woodland Hills, Ca. 91364

### BACK ISSUES OF THE NATIONAL NINTY-NINER

BACK ISSUES OF THE NATIONAL NINTY-NINER WILL BE AVAILABLE THE LAST WEEK OF JANUARY, 1985. WE ARE CURRENTLY PREPARING THE ISSUES FOR OUR PRINTER TO COPY. ALL THE BACK ISSUES OF OUR NEWSLETTERS (13 TOTAL) WILL BE INCLUDED IN ONE BINDER FOR YOUR CONVENIENCE.

IF YOU DESIRE TO ORDER YOUR SET OF BACK ISSUES OF THE NATIONAL NINTY-NINER, PLEASE FORWARD YOUR CHECK FOR \$10.00 TO:

THE 99'ERS ASSOCIATION  
DEPT BNL  
3535 SO. H ST., #93  
BAKERSFIELD, CA 93304

PLEASE ALLOW FOUR (4) WEEKS DELIVERY TIME AFTER FORWARDING YOUR CHECK TO OUR ORGANIZATION. ALL SHIPPING AND HANDLING EXPENSES ARE INCLUDED IN THE \$10.00 CHARGE.

### WORLD OF DOOM By SYMBIOTECH

Randy Romano has created a series of programs for all the dedicated Dungeons and Dragons types among the 99/4A owners. The "World Of Doom" package contains the following items:

The First Scenario - Staff Of Power	\$21.95
The Second Scenario - Ring Of The Dark Lord	\$24.95
The Third Scenario - Dragon Slayers	\$29.95
Character Generators - Doom Characters	\$ 7.95
Adventure Pack - All Three Games	\$53.95
Mini Adventure Pack - Second/Third Scenarios	\$40.95

For additional information on the listed games and other new items, please write the Symbiotec at:

SYMBIOTECH INC.  
P.O. BOX 320  
ROSCOE, ILL 61073

### REMEMBER WHEN TI ANNOUNCED LOWER PRICES???

The drastic price reductions are shown below:

<u>ITEM</u>	<u>PRICE</u>
DISK MEMORY DRIVE	\$399.95
DISK CONTROLLER CARD	\$249.95
MICROSOFT MULTIPLAN	\$ 99.95
TI WRITER	\$ 99.95
RS-232 CARD	\$174.95
MEMORY EXPANSION CARD	\$299.45
PASCAL CARD	\$249.95

These drastic new prices were announced in the NEW JUE NEWS (newsletter of The New Jersey Users Group) in June, 1983. We wish to thank them for this humorous and informative piece of nostalgia, published in their November, 1984 newsletter.

### SST SOFTWARE HAS REDUCED PRICES

SST has reduced prices on its Basic and Expanded Basic Compilers. The Basic Compiler is now \$49.00 and the Expanded Basic Compiler is \$59.00. The Expanded Basic Compiler now has a high resolution graphics package available for the new price.

The high resolution graphics package is available as part of a new add-on utility package. The package includes a text mode and two additional commands. The utility package is \$30.00. You may also purchase a back-up copy of the compiler for \$30.00. The back-up copy will be the newest version of their compiler which continues to compile even though an error was found. All errors found are listed at the end of the Compilers run. Earlier versions stopped completely after one error was located. Now, the new utility package and the compiler update can be purchased together for \$45.00

Write or call SST software for their latest combined newsletter and product update sheet at:

SST SOFTWARE, INC.  
P.O. BOX 26  
CEDARBURG, WI 53012  
(414) 771-8415

# ARTICLES

## TIPS FROM THE TIGERCUB - #17

Copyright 1984 By  
TIGERCUB SOFTWARE  
156 Collingwood Ave.,  
Columbus, OH 43213

Distributed by Tigercub Software to TI-99/4A Users Groups for promotional purposes and in exchange for their newsletters. May be reprinted by non-profit Users' Groups, with credit to Tigercub Software.

My new catalog #5 is now available for \$1.00, which is deductible from your first order. It contains over 130 programs in Basic and Extended Basic at only \$3.00 each (plus \$1.50 per order for cassette, packing and postage, or \$3.00 for diskette, PP&M). The entire contents of Tips from the Tigercub Nos. 1 through 14, with more added, are now available as a full disk of 50 programs, routines and files for only \$15.00 postpaid.

**Nuts & Bolts** is a diskfull of 100 (that's right, 100!) XBasic utility subprograms in MERGE format, ready for you to merge into your own programs. Contents include 13 type fonts, 14 text display routines, 12 sorts and shuffles, 9 data saving and reading routines, 9 wipes, 8 pauses, 6 music, 2 protection, etc., etc., all for just \$19.95 postpaid! And if you send an order before 31 December 1984 and mention your user group, you may take a 10% discount.

My 28-Column Converter, published in Tips #15, has a bug which causes a line to disappear if the wrap-around causes it to begin with a period and you are using the formatter option. Here is the fix -

```
Change line 300 to read: 300 FOR N=1 TO 5 :: READ CH$,R$
Change line 280 to read: 280 DATA @,{,&},^,*,$,@,.,&
```

In other words, your DATA items will be the "at" sign above the 2, the left brace on the front of the F key, the ampersand on the 7 key, the right brace on the front of the G, the carat sign above the 6, the tilde on the front of the W, the asterisk above the 8, the whatsit? on the front of the A, the period, and the backslash on the front of the Z.

A couple of other changes will automatically turn off the automatic fill and adjust, and turn it back on. At the end of line 180, add :: PRINT #2:".NF" and change line 270 to NEXT J :: PRINT #2:".FI;AD;" :: CLOSE #2 :: CLOSE #1 :: END

Now, as long as the text strings in your program don't contain those oddball characters, all should be well. However, the program has one more bug which is common to all 28-column converter programs, and for which I can find no really good fix. If a program line is exactly 80 characters long, the next program line will follow immediately after it instead of starting on the next line. So, load the file in the Editor mode and scan it before you print it. If any of you whiz kids (or whiz grandpas) can figure out a way to program around that problem, please let me know!

A challenge in Tips #9 was to write a 1-line XBasic program which would take only 70 seconds to scramble the numbers from 1 to 255 into a completely random sequence without duplication. Richard Mitchell, the editor of Super 99 Monthly, came up with an algorithm which is shorter than mine and runs about 10 seconds faster - but it sure does chew up a lot of memory!

```
1 DIM A(255),C(254):: RANDOM
IZE :: CALL PEEK(-31808,B)::
IF B=0 OR A(B)=B THEN 1 ELS
E C(D)=B :: A(B)=B :: D=D+1
:: IF D=255 THEN END ELSE 1
```

And if you're not subscribing to Super 99 Monthly, you should be! It's only \$12 a year, and full of very useful programs, routines and tips. The address is Bytemaster Computer Services, 171 Mustang Street, Sulphur LA 70663. Also be sure to get the National Ninety-Niner from the 99ers Users Group Association (3535 So. H St. #93, Bakersfield CA 93304), also only \$12 a year. Their roster of writers is beginning to look like the Who's Who of the TI world.

Danny Michael has written an assembly language program which will dump a graphics screen to a dot matrix printer (Epson or Gemini, and probably others) in less than 50 seconds - and he's giving it away. Just send him an initialized disk in a diskette mailer with an address label back to you and enough return postage. His address is Route 9, Box 460, Florence AL 35630.

Please, can ANYONE tell me where I can buy diskette mailers at a decent price? The cheapest I have found are \$0.65 each for an 11" x 9" piece of cardboard!

Somebody said they liked my Alphabet Song in the last Tips, and somebody else wanted some more routines for the speech synthesizer, so I put it all together and here's what I came up with. If you can type the alphabet without a mistake, you get an encore.

```
100 CALL CLEAR
110 PRINT " ALPHABET S
ONG"
120 FOR J=1 TO 20
130 PRINT
140 NEXT J
150 PRINT " by Ji
 Peterson": "Wait, please"
:
160 OPEN #1:"SPEECH",OUTPUT
280 K2=64
290 CALL KEY(3,K,ST)
300 IF (ST<1)+(K<65)+(K>90)T
HEN 290
310 IF K<>K2+1 THEN 330
320 T=T+1
330 PRINT #1:T$(K-64,1):T$(K
-64,2)
340 CALL HCHAR(12,17,K)
350 K2=K
```

```

170 DIM T$(26,2)
180 DATA 12,12,4,4,1,1,4,7,7
,8,8,10,10,10,10,12,4,4,7,8,
8,10,4,8,8,10
190 FOR J=1 TO 26
200 READ X
210 T$(J,1)="/"&STR$(X)&" "
&STR$(X/10&32)
220 T$(J,2)=CHR$(J+64)
230 NEXT J
240 T$(23,2)="DOUBLE"&"!"&"!
"&"U"
250 CALL CLEAR
260 PRINT "READY - TYPE THE
ALPHABET"
270 T=0

```

```

360 IF K<>90 THEN 290
370 IF T=26 THEN 390
380 GOTO 270
390 FOR K=65 TO 90
400 CALL MCHAR(12,17,K)
410 PRINT #1:T$(K-64,1):T$(K
-64,2)
420 NEXT K
430 PRINT #1:T$(1,1):"NOW IV
E":T$(3,1):"SAID MY":T$(5,1)
:"A B":T$(3,1):"SEEZ"
440 PRINT #1:T$(8,1):"WONT Y
OU":T$(10,1):"COME AND":T$(1
2,1):"PLAY WITH":T$(1,1):"ME
"
450 GOTO 270

```

Terry Atkinson's routine to redefine the cursor has aroused some interest, so I fiddled around and came up with this version to change the cursor automatically to whatever character, normal or redefined, that you input. And of course you can always color the cursor with CALL COLOR(0,5,11) or whatever colors you like.

```

100 !CURSOR CHANGER by Jim P
eterson
110 INPUT A$ :: A=ASC(A$)::
CALL CHARPAT(A,A$):: FOR J=1
TO 16 STEP 2 :: H$=SEG$(A$,
J,2):: CALL HEX_DEC(H$,D)::
T=T+1 :: H(T)=D :: NEXT J ::
120 CALL INIT :: CALL LOAD(8
196,63,248)
130 CALL LOAD(16376,67,85,82
,83,79,82,48,8)
140 CALL LOAD(12288,H(1),H(2
),H(3),H(4),H(5),H(6),H(7),H
(8))
150 CALL LOAD(12296,2,0,3,24
0,2,1,48,0,2,2,0,8,4,32,32,3

```

```

6,4,91)
160 CALL LINK("CURSOR")!THAN
KS TO TERRY ATKINSON
170 SUB HEX_DEC(H$,D):: N=1
:: DEC=0
180 FOR J=1 TO LEN(H$):: A$=
SEG$(H$,LEN(H$)-J+1,1):: IF
ASC(A$)>58 THEN HT=ASC(A$)-5
5 ELSE HT=VAL(A$)
190 DEC=DEC+N*HT :: N=N*16 :
: NEXT J
200 IF DEC<>32768 THEN D=DEC
ELSE D=-165536-DEC)
210 SUBEND

```

Most folks don't seem to know, and some folks refuse to believe, that the Memory Expansion can't store strings. If you are one of the disbelievers, plug in your Memory Expansion and try this -

```

100 FOR J=1 TO 255 :: M$=M$.
CHR$(J):: NEXT J
110 DIM A$(100) :: M$=M$.
$X=M$ :: PRINT X :: GOTO 110

```

Now RUN that. On my console, I get MEMORY FULL when X=43 although the SIZE command shows I have 24399 bytes of program space free (in the Expansion) - but only 204 bytes of free stack (in the console). Without the Memory Expansion I can get X up to 51, and in Basic to 53.

This can be a serious handicap if you are running a program which reads in a large number of strings from DATA statements, or generates strings while running.

Of course, when the Memory Expansion is attached, the program and the numeric variables are stored in the Expansion, leaving all the console memory available for strings - but if you do not generate strings, the console memory remains unused, because numeric data cannot overflow into it!

If your program generates more numeric variables than the Memory Expansion can hold, you can however store them in the console by converting them to strings, using STR\$, and convert them back to numbers with VAL. This will allow you store an additional 700 to 900 or more numbers. Try this - When you get MEMORY FULL, type SIZE.

```

100 DIM A(3040),A$(1000):: F
OR X=1 TO 3000 :: A(X)=99 ::
PRINT X :: NEXT X
110 Y=Y+1 :: A$(Y)=STR$(99)
:: PRINT Y :: GOTO 110

```

Dave Renkenberger sent me a neat little routine, and I played around with it a bit. For you who are not football fans, I'd better explain that the Wave is performed at football stadiums when the cheerleaders get the fans to stand and cheer, one seating section at a time, across the stadium - and those drunks on the roof are usually out of sequence.

```

90 !THE WAVE by David Renken
berger/modified by Jim Peter
son
100 CALL CLEAR :: CALL SCREE
N(4)
110 A$="!!!the wave!!!"
120 DISPLAY AT(4,14-LEN(A$)/
2):A$
130 B$="press any key to sto
p"
140 DISPLAY AT(22,14-LEN(B$)
/2):B$
150 B$="995A3C3C3C3C2466"
160 A$="0000181B7EBD3C3C"

```

```

170 FOR CH=91 TO 118 :: CALL
CHAR(CH,A$):: M$=M$&CHR$(CH
):: NEXT CH :: FOR R=8 TO 12
:: DISPLAY AT(R,1):M$ :: NE
XT R
175 FOR T=1 TO 26 STEP 5 ::
DISPLAY AT(22,T):SEG$(M$,T,1
):: NEXT T
180 FOR CH=91 TO 123 :: CALL
CHAR(CH,B$):: CALL CHAR(CH-
5,A$):: CALL SOUND(-999,-7,5
$RND):: CALL KEY(3,K,ST):: I
F ST<>0 THEN STOP
190 NEXT CH :: GOTO 180

```

MEMORY FULL

Happy hackin'

Jim Peterson

## PASCAL NOTES (ADVANCED TOPIC)

By Edgar Dohmann -- JSC User's Group (JUG)

I have taken the general approach of writing this series of articles in a tutorial format primarily to help those of you who are just learning to use the P-System. Once we have covered the operation and use of the P-System features and utilities I will progress into actual Pascal programs. However, I realize there are a number of experienced users out there and from time to time I will try to write an article with some information of interest to you and hopefully the less experienced folks will file these advanced topics away for future reference.

This month we will extend my last article on the Filer a bit and examine the structure of P-System files and directories. If you have a disk analyzer program like the Disk Fixer, you might want to dump sectors 0 through 11 of a P-System disk on a printer to help you follow this article a little easier. Keep in mind that P-System blocks are 512 bytes in length while TI disk sectors are 256 bytes long. Thus each block occupies 2 sectors and always starts on an even numbered sector.

P-System disks are formatted in the same format as other TI disks. The Filer's ZERO utility must then be used to set up the disk for the P-System and this is when the disk becomes different from other TI disks.

**SECTOR 0** -- Sector 0 will be similar to other TI disks. It will have the disk name and the characters "DSK" which the Disk Manager looks for to determine if the disk has been initialized. The bitmap for all sectors on the disk will be set to mark the entire disk as being allocated. The purpose for this is to avoid accidental overwriting of P-System files from a Basic program or with the Disk Manager. The P-System does not use this sector for any other purpose.

**SECTOR 1** -- This sector will have only one entry. The first word will contain a value of 2 to indicate that the only file directory on the disk is in sector 2. The Disk Manager uses this sector to store the directory sectors of files in alphabetic order for cataloging and rapid access purposes. The P-System does not use this sector for any other purpose.

**SECTOR 2** -- This sector is a "pseudo-directory" for a file called PASCAL. The information is in standard format for TI file directories but it indicates that all sectors from 3 to the end of the disk are allocated to this file. Again, this is just to ensure that P-System files will not be accidentally destroyed by Basic or the Disk Manager. If you catalog a P-System disk with the Disk Manager, only this pseudo-file will be listed and the catalog will indicate that it occupies all space on the disk. In addition to providing protection for your Pascal files, this usage of sectors 0, 1, and 2 allows you to use the Disk Manager to make backup copies of your P-System disks.

**SECTOR 3** -- This sector will be all zeroes. It is not used and is left blank because P-System blocks must start on even numbered sector boundaries.

**SECTORS 4 through 11** -- These sectors are also referred to as blocks 2 through 5 of the disk and they contain the directories of the P-System files which are stored on the disk. If you are using the duplicate directory option, then sectors 12 through 19 will contain an exact copy of sectors 4 through 11. Actual file storage will start in either sector 12 or sector 20 depending on whether or not you are using duplicate directories.

Each P-System directory is 26 bytes long so there is room for 78 directories in the 8 sectors (4 blocks) used for this purpose. It is interesting to note that unlike the TI File Manager, the P-System File Manager allows records to overlap sector boundaries. This avoids wasted space at the end of a sector if your record blocking does not precisely match sector sizes.

The first directory is used as a disk header for the P-System similar to the way the TI File Manager uses sector 0. The format of this first directory is as follows:

**WORD 1** : The first disk block allocated for directory use. Even though the disk directory starts at block 2, this word is always 0 so the information in blocks 0 and 1 will not be destroyed.

**WORD 2** : The last disk block allocated for directory use. Actually the value of this word is one greater than the last block used and as such it really marks the first block available for the first file. If you are using duplicate directories, the value in this word will be 10 (>000A) and if you are not using duplicate directories, the value will be 6.

**WORD 3** : Only the lower nibble (4 bits) of this word is used and it indicates the type of file. For directories the file type is 0 so this word is always 0.

**WORDS 4-7** : The disk name. The high byte of word 4 specifies the length of the disk name (maximum of 7 bytes). The name starts in the low byte of word 4.

**WORD 8** : The end of volume block. This word specifies the maximum number of blocks on the disk. For example a DS/DD disk will have 1440 sectors or 720 blocks so the value of this word will be >02D0.

**WORD 9** : The number of files currently stored on the disk.

**WORD 10** : This word appears to be reserved for possible storage of system boot time. I suppose a manufacturer could allow system time to be set from the Date command like the month/day/year can be set. While I do not think TI uses this word the format is : Hours in the high 4 bits, Minutes in the next 6 bits, and Seconds in the low 6 bits. A time of 6:59:42 would be >6EEA. This format is good only for a 12 hour clock.

**WORD 11** : The date of the last boot. Actually this value is updated only if you use the Date option of the Filer to set a new date. The date you enter is stored as follows : Year in the high 7 bits, Day in the next 5 bits, and Month in the low 4 bits. A bias of 1930 is added to the year value in the high 7 bits. Thus a date of Nov. 9, 1984 would be stored as >A89B.

WORDS 12-13 : Unused in the directory so their value is 0.

The next 77 directory records define files that are stored on the disk. The P-System File Manager requires that a file occupy a contiguous set of blocks on the disk. This restriction makes the directory format simpler than the TI File Manager which allows files to be stored in a number of segments. As a result, if you have a lot of file addition and deletion activity on a P-System disk, you may want to run the Krunch option of the Filer from time to time to ensure that your disk is not overly fragmented. The P-System file directories are stored in the order the files are placed on the disk. An alphabetic index is not maintained like it is by the TI File Manager. This is why P-System catalogs do not appear in alphabetic order (although a sorting algorithm in the Filer should not have been too much to ask).

The format of the file directories is as follows:

WORD 1 : The first block used by the file.

WORD 2 : The last block used by the file plus 1. Like word 1 of the disk header directory, this points to the first block of the next file.

WORD 3 : The high order bit is some sort of status bit but I am not sure of its purpose or if TI uses it. The file type is stored in the lowest 4 bits of this word. As mentioned above, the disk header directory has a file type of 0. Executable program files have a type of 2, Text files are type 3, and Data files are type 5.

WORDS 4-11 : The file name. The length (maximum of 15 characters) is stored in the high byte of word 4. The file name starts in the low byte of word 4.

WORD 12 : Last byte used. This appears to indicate the last byte used on the last block allocated to the file. This is probably used to determine where to start appending to an existing file.

WORD 13 : Date of last update. The month/day/year of the last file update will be stored in the same format as described above for the file header directory. The date stored will be the same as the date in the file header directory which will only be current if you used the Date option to update the system date to the current date.

### THE MAKING OF THE CORCOMP 9900 DISK CONTROLLER CARD

By W. R. Moseid

When the decision was made to provide a new disk controller card to the TI-99/4A world, we began what turned out to be a long and arduous trip. The metal case (clamshell) was easy. Just use the one that had been designed for the CorComp 32K and RS-232 cards. Then, modify it slightly for a slot to accommodate the new circuit board which would project through the back of the case. This part would hold the connectors for the cables to the internal and external floppy disk drives.

Since all disk controller integrated circuits (chips) can control up to four floppy disk drives, we decided not to restrict people to three drives. With some careful planning, design and care a card could be produced which would support up to four disk drives with any combination from the list below:

- ‡ Single or Double Sided
- ‡ Single or Double Density
- ‡ 35 or 40 Tracks per Diskette Side
- ‡ Any Combination of the Above
- ‡ Automatic Density Recognition

In order to allow the use of a variety of disk drives which people had available, a feature was selected which allowed the owner to set the Head Step Time (Head step time is the time it takes the disk drive read/write heads to step from one track to another). You can select one of four step times for each disk drive in your system. The times selected were 15 milliseconds (ms) 10 ms, 6 ms, 3 ms. This timing is set by positioning a set of DIP (Dual-In-line Pack) switches to various on/off settings. The decision was made to place the DIP switches inside the case. Even though this meant the owner would have to remove the case to set all the DIP switches, this approach was selected for several reasons.

- ‡ Safety for the Card (Makes sure power is OFF when the DIP switch is set)
- ‡ Safety for the DIP switch (Chances of something hitting it or changing the settings were minimized)
- ‡ Lower costs to the Consumer (Assembly Time and Material Costs were lower)
- ‡ The 10ms Factory Setting works with most Drives

When the 99/4A Power Up sequence was examined, several interesting things were discovered:

- ‡ Plato, TE II, and other Command Modules have a special sequence that runs at that time. (once they get control they do not give it up and the power up scan is not completed.)
- ‡ In order to allow the owner the ability to select the CorComp Disk Manager from the Title screen, a special power up Screen had to be made to allow our rapid loader to execute on the single key press.
- ‡ Because of the way Plato, TE II and other Command Modules operate during PowerUp, a choice of two different Menu Screens had to be provided.

When the early timing studies were done, they were conducted with direct I/O using an Assembly Language program (NO GROM). At that time we calculated that the CorComp Disk Controller in double density could run 2 to 4 times faster than the TI Disk Controller. A fact proven by the speed at which the 98 sector Disk Manager program loads into expansion memory. This is based on the way that the CorComp Disk Controller card accesses the diskette and transfers the information into the computer. However, when TI designed their disk memory system, they made a decision that the memory expansion would not be required. This way, with the controller (old stand alone disk controller) and a console, you could utilize BASIC with the disk system. In order to do this, the TI Disk System and ALL of the Command Modules which use disk expect the information read from the disk to be in the console (VDP RAM) memory. For example, when BASIC, EXTENDED BASIC or the EDITOR/ASSEMBLER load from disk, they expect the information to be in the console memory. If memory expansion is attached then EXTENDED BASIC will move the information to the memory expansion after it is loaded. This moving process is very time consuming. Remember, each sector (256 characters) read or written must be passed through VDP RAM in order to be compatible with TI firmware/software. In the following table are some of the latest timing tests using GROM and VDP RAM:

#### Time To Load In Seconds

FILE TYPE	# OF SECTORS	TI CONTROLLER		CORCOMP DISK CONTROLLER			
		SD		SD		DD	
		L/O	READY	L/O	READY	L/O	READY
BASIC PROGRAM	47	7.2	24.5	7.2	24.5	5.0	22.3
X-B PROGRAM	39	6.3	8.8	6.3	8.8	3.6	6.3
X-B I/V 254	52	---	18.2	---	18.2	---	14.9
E/A PROGRAM	25	---	6.3	---	6.3	---	4.2
E/A D/F 80	181	---	55.8	---	54.3	---	47.8

\* L/O = CONTROLLER OUT      READY = CURSOR BACK ON SCREEN

NOTE: The tests were conducted with default interlace selections. Timing may be improved with different interlace selections for the various modules and languages.

With the speed increase indicated in the previous table, we naturally were curious how time is consumed having to use the VDP RAM as an intermediate storage area. The table below shows the time required to copy a disk. When the CorComp Disk Controller copies a disk, VDP RAM does not have to be used as an intermediate storage place saving time.

#### Time To Copy 360 Sectors (3 Files) in Seconds

Type of Copy	TI Disk Controller	CorComp Disk Controller	
		NO/TURBO	W/TURBO
SD to SD	151	143	70
SD to DD	---	135	61
DD to DD	---	123	51

Measuring performance increase figures is always a challenge. This is due to the fact that the "statistics people" can make them do what they want. But, you can see a performance increase in using the CorComp Disk Controller Card and DiskManager of up to approximately 296% depending on several factors which are:

- \* Diskette Density
- \* Diskette File Type (DIS/VAR-worst)
- \* Kind Of Loader (X/B, Ed/Asm, etc.)
- \* The Operation Being Done
- \* Language Used (GROM - worst case)
- \* File Sector Location On Diskette

When the Utilities were designed we found that TI changed Extended Basic to prevent its scanning the peripheral DSR's for CALL's (ie; CALL FILES) from running a program. In order to allow the utilities to function with Extended Basic, a list of the utilities had to be provided in the Link Table in extended memory. Thus the syntax for using the CorComp Utilities in X/B is CALL LINK(utility name)(etc.....).

Console Basic did not possess this constraint. In Basic, the Utility syntax is CALL (utility name, etc.)

The Disk Manager was written especially for the CorComp Disk Controller. We sat down and thought of all the features one would like in a Disk Manager. Just about all of the ideas are in the current Disk Manager. A decision was made to publish the Disk Manager on diskette for the following reasons:

- \* Easy to Release an Update
- \* Lower Cost to the Customer

The task of figuring out all the technical details of how to achieve compatibility with the TI hardware, software and firmware was a very hard and time-consuming effort. At times, some of the issues seemed almost too much to overcome. In the end, our perseverance and determination were rewarded and the Disk Controller Card reached the market. All the known problems, as of this writing (11/10/84), have been resolved. All other 3rd Party cards are fully compatible with the CorComp 9900 Disk Controller Card.

While writing the manual, we decided to try and present the material in a manner that would allow the beginning user to sit down and follow the guide in a logical step-wise manner. This method allows the user to learn how to utilize the card from its manual in a very straight-forward manner. The manual was also designed for the technically minded individual who could read to the level of detail desired. It required 700 hours of effort preparing the manual before these objectives were achieved.

### LOADING CORCOMP DISK MANAGER FROM EDITOR/ASSEMBLER

This bit of information is reprinted from Miller's Graphics publication "The Smart Programmer". It was printed on page 5 of the July, 1984 issue. For everyone with a CorComp Disk Controller card, Tom Knight sent us the following program. This short program allows you to load the CorComp Disk Manager from the Editor/Assembler module with option 3 - Load and Run. Thanks Tom.

```

IDT 'LOADMGR'
AORG >2700
DEF MGR
MGR  LWPI >83E0
      MOV R11,@>8300
      LI R12,>1100
      SBO >0000
      SBZ >000B
      BL @>44F2
      NOP
      SBZ >0000
      MOV @>8300,R11
      B $R11
      END

```

### FORTH DO LOOPS

By Gene Thomas - Little Rock 99'ers

Not long ago I came down with a disease known as Forth Syndrome! There is no known cure, so I must learn to live with it.

One problem I immediately ran into was the difficulty in creating a delay. In BASIC, FOR I= 1 TO 500 :: NEXT I, simple. A DOA/LOOP comes to mind, but then the difficulties arise. Even a large count such as, : 10000 0 DO LOOP,; does not create a delay by any practicle assessment. So, let's nest DO-LOOPS. That turns out to be a bad idea for several reasons.

1. The numbers you have to test to find the delay you want are too big.
2. Numbers for three or more DO-LOOPS will have to be put on the stack.

Also, that method doesn't look very elegant. Here is what I have defined to do the deed (attached printout).

The stack effect is, (n---). You will find that numbers such as 5, 10, 15,...50 DELAY work well.

I am a novice at FORTH so I fully expect that this method could be improved upon, or done in a completely different way. In the meantime, it is all I have. Hope you find some use for the program, or that my efforts will challenge someone to create an improvement.

```

SCR #86
0 ( EXT. SCREEN #2 FIG (TI) GT NOV84) 1
2 0 VARIABLE WAIT & SET VAR FOR DELAY
3
4 : DEL.INC 1 WAIT +! ; & INC DELAY VARIABLE
5
6 : DEL.CON WAIT 200 < IF & DELAY 200 COUNTS FOR
7 DEL.INC MYSELF THEN & EACH DO-LOOP COUNT
8 0 WAIT ! ; & RE-INITIALIZE WAIT
9 : DELAY ( N --- ) & #'S FROM 5 THRU 50
10 0 DO DEL.CON LOOP ; & WORK WELL.
11
12

```

### A REAL TIME CLOCK FOR THE CORCOMP 9900 MICRO-EXPANSION SYSTEM By John Clulow and Ron Bries, New Horizons

Recently the New Horizons and OH-MI-TI Users Groups (Northwest Ohio) obtained a CorComp Micro Expansion System (MES) for use in our Bulletin Board System, "TI COMM". The MES contains the equivalent of a 32K memory card, RS232 card, and a CorComp double-sided, double-density disk controller. The MES is only about 5.5" X 5.5" square, weighs a mere 1 lb. 10 oz., and does not require a cooling fan. We are so favorable impressed with the heavy-weight performance of the MES unit that we decided to change the name of our Bulletin Board to COR COMM! It has run our BBS perfectly now for over 350 continuous hours, and exhaustive testing with software like TI-Writer, Editor/Assembler, TE II, and Extended BASIC has yet to uncover a bug. Best of all, our BBS now has 1440 sectors of disk space on-line with only one double-sided, double-density drive. (The MES can handle up to 4 DSDD drives.)

Having had some experience running the BBS with the TI PEB system, we realized the need for a real time clock. The initial version of our BBS used an interrupt driven "real" time clock, but the intermittent disabling of interrupts in the course of program execution greatly affected the clock's accuracy. The next step was to build a real-time clock card for the PEB. We based our card upon a design for a clock and A/D converter by Gary Emmich and Tony Albanese of the No. NJ 99er User's Group with modifications in the power supply circuit and address decoding (required for the CorComp DSDD controller).

The design presented here is a further modification for use with the CorComp MES. The device plugs into the 44 pin bus and may be used with the console and Mini Memory or Editor Assembler (although it is designed specifically for the 9900 Micro-Expansion System.) Used with the MES, the card may be accessed through TI BASIC or Extended BASIC using CALL LOAD and CALL PEEK statements. Alternatively, an assembly language program (below) may be used to access the clock with simple CALL LINK("SET",...) and CALL LINK("TIME",...) statements. Instructions for use of the A/L routine are given in the source code.

To access the clock through BASIC, CALL LOAD statements are used to set the clock registers and CALL PEEK to read the time and date. In this case conversion from BCD to decimal is required within the BASIC program. In addition, when accessing the clock from BASIC the sound generator must be reset by using a CALL SOUND statement after the access. To access the clock through BASIC, use the decimal versions of the addresses shown in the A/L program (e.g., Seconds = -31228). The byte you write or read at each address must be in BCD format. That means that each digit of the number is expressed separately in hexadecimal. For example, the number 31 would be equivalent to 3\*16 + 1 or 49. So if you wanted to write a 31 to the Day of Month register, you would write 49 instead of 31: CALL LOAD(-31220,49). Similarly, the month 12 would be represented as 18 (1\*16 + 2). To convert a number back from BCD to decimal, divide 16 into it. The integer result is the first digit and the remainder the second. For instance if you obtained an 88 from the seconds register -- CALL PEEK(-31228,X) -- INT(88/16) is 5 with a remainder of 8 so the decimal number is 58 seconds. And remember to execute a CALL SOUND statement after you access the clock. The A/L routine takes care of BCD conversions for you and also returns string representations for day of week and month.

The parts list is shown below. The 22/44 pin edgecard connector must be modified by cutting off the tabs on the ends (so it will fit into the 44 pin bus on the right side of the MES). The solder tails should be spread apart and the ribs in the slot on the back of the connector removed so that the grid board can be epoxied to the connector. It will be necessary to cut a 1/4 inch notch in each side of the grid board to make it fit. We recommend the use of wire wrap wire in constructing the card even if you plan to solder because it allows for more dense circuit configuration. Note that three "AAA" Nickel-Cadmium cells are used in series as battery backup for the clock.

Pay particular attention to pin numbering on both DIPs and the 44 pin edgecard. Note that each IC and socket have a notch in one end. The pin to the left of this notch, looking at the DIP or socket from the top, is Pin 1. Pins are then numbered down that side, across, and then back up the opposite side. The last pin on the IC will be immediately opposite Pin 1. Some ICs may have a spot adjacent to Pin 1 instead of a notch on one end. Looking at the console or MES unit, the 44 pin edgecard is numbered with Pin 1 on the lower left and Pin 2 on the top left. Pin 43 is on the bottom right and Pin 44 on the top right. Odd numbered pins are on the bottom - left to right and even numbered pins on the top. The ground used in the circuit consists of pins 21, 23, 25, and 27 of the I/O bus. These pins should be connected to a single ground bus wire. That wire should then be used in making all connections to ground including the battery ground. The supply voltage +5 for all ICs is connected to pin 1 of the 44 pin edgecard. This is the voltage supply for the speech synthesizer. It is rated at 50 mA. The completed clock card only draws 8 mA but YOU SHOULD NOT ATTEMPT TO USE THE CLOCK CARD IN COMBINATION WITH THE SPEECH SYNTHESIZER: To do so may exceed the current limit.

Before installing the chips in their sockets check the continuity between pin 1 (+5 V) and ground to make sure that there are no dead shorts. Then do a visual check of all solder joints at IC and 44 I/O pins to insure there are no shorts. Make sure the polarity of the tantalum capacitor is correct. Install the chips making sure the pin numbers are correct with respect to the sockets. Install the three Ni Cad batteries in series making sure that polarity is correct. Plug the card into the 44 pin bus with the console power off. When you power up the console, you should get the normal title screen. Any abnormality in the screen or sound indicates a problem with the card.

If you have had some previous experience building circuits such as this, you will not find the board particularly difficult to construct. However, we do not recommend that this project be undertaken if you don't feel sure of yourself and in any case we assume no responsibility for any damage to your equipment or consequential damage arising from use of the clock card. If you want to build a card and feel uncertain about some aspect of its construction, you may be able to get help from someone in your users group who is more familiar with electronics.

Price	Supplier	Cat. No.	Description
\$ .49	Radio Shack	270-398	"AAA" Cell Battery Holder (2)
\$ .39	"	270-404	"AAA" Cell Battery Holder (1)
\$ 1.95	"	276-158	Grid Board
\$ .79	"	272-1437	22 MFD Tantalum Capacitor
\$ .99	"	267-1123	Germanium Diodes (pk of 10)
\$ .19	"	271-023	1K Ohm Resistor (pk of 2)
\$ .19	"	271-049	220K Ohm Resistor (pk of 2)
\$ 2.39	"	278-501	Wire-Wrap wire (spool)
\$ 4.49 ea	"	23-127	NiCad Batteries (need two pks)

For other components...

\$ 9.60	Digi-Key	MM58167AN	MM58167 Real Time Clock
\$ 2.70	"	KF386	32.768 KHz Min. Crystal
\$ 4.20	"	C5-22 H4	Ser Edgebd Connector 22/44
\$ .54	"	74LS04N	Hex Inverter
\$ .45	"	74LS30N	8-input NAND gate
\$ ----	(don't carry)	74LS245N	Bidirectional Octal Bus Drivr
\$ .15	"	C8914	14 pin sockets (2 required)
\$ .23	"	C8920	20 pin socket
\$ .28	"	C8924	24 pin socket

\$An alternate source...			
\$12.95	JDR Microdevices	MM58167	MM58167 Read Time Clock
\$ 1.95	"	KF386	32.768 KHz Min. Crystal
\$ ----	(don't carry)	-----	H4 Ser Edgebd Connector 22/44
\$ .24	"	74LS04N	Hex Inverter
\$ .25	"	74LS30N	8-input NAND gate
\$ 1.49	"	74LS245N	Bidirectional Octal Bus Drive
\$ .15	"	C8914	14 pin sockets (2 required)
\$ .29	"	C8920	20 pin socket
\$ .30	"	C8924	24 pin socket

Digi-Key 1-800-344-4539  
 Highway 32 South  
 P.O. Box 677  
 Thief River Falls, MN 56701

JDR Microdevices 1-800-538-5000  
 1224 S. Bascom Avenue  
 San Jose, CA 95128  
 CA Res: 1-800-662-6279

DEF SET, TIME

```

*
* This program allows access to the
* clock. To set it use the statement:
* CALL LINK("SET",H,M,S,DW,M,DM) where
* the parameter list is Hours, Minutes,
* Seconds, Day of the Week, Month, and
* Day of the Month. To read the time:
* CALL LINK("TIME",H,M,S,DW,M,DM,D$,M$)
* where D$ and M$ are return variables
* for the day of week and month names
* (e.g., "Saturday" "June").

```

```

* Equates used are for Ext. BASIC
* Change as required
*
*          XBASIC  MINIMEM  ED/ASSM
*          w/BSCSUP
*
NUMREF EQU >200C >6044 REF NUMREF
STRREF EQU >2014 >604C REF STRREF
NUMAS6 EQU >2008 >6040 REF NUMAS6
STRAS6 EQU >2010 >6048 REF STRAS6
XMLLNK EQU >2018 >601C REF XMLLNK
CFI EQU >12B8 >1200 >1200

```

```

* Equates
*
*          hex  decimal
*
THOUS EQU >8600 -31232
HUNDR EQU >8602 -31230
SECOND EQU >8604 -31228
MINUTE EQU >8606 -31226
HOUR EQU >8608 -31224
DAYW EQU >860A -31222
DAYM EQU >860C -31220
MONTH EQU >860E -31218
STATUS EQU >8688 -31096
RESET EQU >8684 -31100
FAC EQU >834A -31926

```

```

* Buffers and data
*
WS BSS 32
HR BYTE 0
MN BYTE 0
SC BYTE 0
DW BYTE 0
MO BYTE 0
DM BYTE 0
SAVRTM DATA 0
D10 DATA >000A
D6 DATA >0006
FAC1 DATA >4000
BUFFER BSS 10
SUND TEXT 'Sunday'
MOND TEXT 'Monday'
TUES TEXT 'Tuesday'
LI R1,7
LI R2,>8000
RS1 MOVB R2,@RESET
SRL R2,1
DEC R1
JNE RS1

```

```

WEDN TEXT 'Wednesday'
THUR TEXT 'Thursday'
FRID TEXT 'Friday'
SATU TEXT 'Saturday'
JAN TEXT 'January'
FEB TEXT 'February'
MAR TEXT 'March'
APR TEXT 'April'
MAY TEXT 'May'
JUN TEXT 'June'
JUL TEXT 'July'
AUG TEXT 'August'
SEP TEXT 'September'
OCT TEXT 'October'
NOV TEXT 'November'
DEC TEXT 'December'
WEEK DATA SUND,MOND,TUES,WEDN
DATA THUR,FRID,SATU
MNTN DATA JAN,FEB,MAR,APR,MAY
DATA JUN,JUL,AUG,SEP,OCT
DATA NOV,DEC,WEEK

```

Read data from XBASIC: Set Clock

```

SET MOV R11,@SAVRTM
LWPI WS

```

```

* LINK Parameter List Format:
* Hr,Min,Sec,Day/W,Mo,Day/M

```

```

* 1) Get XBASIC LINK parameters
* and convert to BCD

```

```

CLR R0
CLR R1
LI R5,6
LI R4,HR
GP1 INC R1
BLWP @NUMREF
BLWP @XMLLNK
DATA CFI
MOV @FAC,R3
CLR R2
DIV @D10,R2
MPY @D6,R2
A @FAC,R3
SWPB R3
MOVB R3,@R4+
DEC R5
JNE GP1

```

```

* 2) Reset all counters

```

```

RD1 INC R1
MOVB @FAC1,@FAC
CLR R2
MOVB @R4,R2
SWPB R2
MOV R2,R6
SRL R2,4

```

```

* 3) Set counters
*
MOV B @MONTH
MOV B @DAYM
MOV B @DAYW
MOV B @HR,@HOUR
MOV B @MN,@MINUTE
MOV B @SC,@SECOND

*
* 4) Return to XBASIC
*
B @RETURN
*
* -----
* Read clock and return data XBASIC
*
* -----
TIME MOV R11,@SAVRTN
LWPI WS

*
* LINK Parameter List Format:
* Hr,Min,Sec,Day/W,Mo,Day/M,Day$,Month$
*
* 1) Get all data from clock
*
READ MOV B @MONTH,@MO
MOV B @DAYM,@DM
MOV B @DAYW,@DW
MOV B @HOUR,@HR
MOV B @MINUTE,@MN
MOV B @SECOND,@SC

*
* 2) Check Status Bit for rollover
*
CLR R0
MOV B @STATUS,R0
JNE READ

*
* 3) Convert back from BCD
*
CLR @FAC+2
CLR @FAC+4
CLR @FAC+6
LI R4,HR
LI R5,6
CLR R0
CLR R1

MPY @D6,R2
S R3,R6
SWPB R6
MOV B R6,@FAC+1
MOV B R6,@R4+
BLWP @NUMAS6
DEC R5
JNE RD1

*
* Return Day and Month strings
*
INC R1
LI R2,BUFFER
CLR R3
MOV B @DW,R3
SWPB R3
DEC R3
SLA R3,1
MOV @WEEK(R3),R4
MOV @WEEK+2(R3),R5
BL @SUBR
INC R1
CLR R3
MOV B @MO,R3
SWPB R3
DEC R3
SLA R3,1
MOV @MNTN(R3),R4
DD @MNTN+2(R3),R5
BL @SUBR

*
* Return to XBASIC
*
RETURN LWPI >83E0
MOV @SAVRTN,R11
BL @R11

*
* SUBROUTINE
*
SUBR S R4,R5
SWPB R5
MOV B R5,@BUFFER
LI R6,1
SWPB R5
DLP MOV B @R4+,@BUFFER(R6)
INC R6
DEC R5
JNE DLP
BLWP @STRAS6
B @R11
END

```

## REVIEWS

THE 9900 IS HERE !

OR

Have you weighed your PE Box lately?

By John Clulow - New Horizons

What weighs 35 pounds, sounds like a vacuum cleaner, and has a 225 square inch footprint? Since receiving a CorComp 9900 Micro-Expansion System for Beta testing, I can see the potential for a whole new direction in humor: Peripheral E-X-P-A-N-S-I-O-N Box jokes.

When you **HAVE** to live with something, you learn to overlook some of the negatives (ask any married man). And the TI Peripheral Expansion Box is no exception. Sure it was better than the "freight train" approach (remember?), but did we really need all that bulk? The constant drone of the fan did have some utility -- "Sorry, honey, I couldn't hear you calling me...", -- but that only worked once or twice.

As an avid 99/4A user, I had learned to live with the PE Box. I didn't love it, but I didn't hate it either. But nothing lasts forever.... Last week I received a CorComp 9900 Micro-Expansion System for Beta Testing.

What is the 9900 MES? Well, take a PE Box with a 32K Memory Card, RS-232 Card (two serial and one parallel port), Double-Sided Double-Density disk controller (supporting up to four drives)---take all of that and put it in a 5" x 5" square box weighing only 1 lb., 10 oz. and you have the 9900 MES. The MES has a footprint only slightly larger than the "flex-cable" connector! Incredible... And guess what? **NO FAN!**

Although the sales of this unit are obviously aimed at those 99/4A console users thinking of upgrading to disk, I believe that many users who currently have a TI disk controller and want to upgrade to the CorComp DS/DD card will choose the MES route instead -- a card costs around \$170.00 and then you don't have a use for the old TI controller. For only \$160.00 more, a user can have a complete system with the weight size, and noise advantages, and use the old TI PEB as a backup.

Double-Sided, Double-Density disk drives can be had at bargain prices, and I believe this will really spark sales for the new CorComp products. Full-height MPI drives can be had for \$100, half-height TEAC's for \$115 and 2/3 size REMEX drives are being sold at only \$59 (\$180 for two drives with power supply and case).

So far, the 9900 MES has flawlessly run all software tried -- TI Writer, Editor Assembler, X-Basic, Mini Memory, and Terminal Emulator II to name a few. But, the toughest test is yet to come. By the time you have read this, the 9900 MES will be running our Bulletin Board -- 24 hours a day, seven days each week. We'll keep you advised of our results.

### BON VIVANT!

(or How To Handle a Hungry Computer)

By Darrell Ingold

This offering of computerized recipe filing comes to us from Yashu Software in Dallas, Texas (see address below). BUT WAIT! This is more than just a recipe file; it can easily be, by design, a file for anything. One of the first things I noticed about the program was its built-in versatility. By selecting "Helpful Aids" you can quickly and easily change the names of the 10 different file categories as well as the keyword Recipe. For example: instead of appetizers, soups, etc, one could label them furniture, appliances, etc.. In addition, each place that the word Recipe appears one could substitute Record. For those of you with no programming experience, the instructions clearly tell how to even change the disk on which the data is stored and the default printer configuration.

Speaking of instructions, the set that accompany this program are quite extensive; 7 full sized pages of instruction plus 6 pages of examples. The information is well organized and easy to follow, even for the novice. All instructions are full sized print on full sized paper...fits well into a binder.

The program as it is formatted, allows the user to enter recipes into the file as follows; under a selected category enter recipe name (max 12 letters), Upper Data Space for ingredients (5 lines) and Lower Data Space for cooking instructions (10 lines). Being a firefighter by trade, I immediately set about to enter all my firehouse recipes into the program. Typing recipes isn't the easiest thing in the world for the inexperienced typist I soon discovered. However, once entered they can be utilized in several ways. Before leaving home on the shift I am to cook, I can quickly get a printout of my selected menu recipes to take to work with me. Very impressive to the station crew but it does little for my cooking ability. In addition to the printout, the program will add, change or display recipes alphabetically in total or by category. The menus are easy to work with and the program responds quickly. About 81 recipes can be put on the same SS/SD disk with the program or 100 if a separate disk is used.

Yashu Software encourages the user to make back-up copies but has a liberal replacement policy which is detailed in the instructions. I also understand that they have another home utility program out call the Big Black Book which will store up to 480 names/addresses plus notes on each, and will soon be marketing The Organizer, an inventory program. I have not yet reviewed either of these however, as of yet. Bon Vivant sells for \$19.95 and requires disk, 32K & XB...printer optional. There is a package deal available for Bon Vivant/Big Black Book for \$29.95. Contact Yashu Software, 6730 E. Northwest Highway, Dallas, Texas 75231.

### CORCOMP CONTROLLER CARD

By Darrell Ingold

I have been using CorComp's new controller card for over a month now and going back to the old TI controller would be like trading in my automobile for a pogo stick! I will have to admit, however, that I had some reservations at first. The screen was the wrong color and the words were different. There was this new item on the CRT named "Disk Manager". It didn't used to be there! Then the beautiful full color TI Rainbow didn't automatically appear either. You could order it to appear, but it didn't just pop up like the old days. In spite of all these obvious drawbacks I pressed on.

In order to properly test the new card it became quite obvious that there was full justification for the purchase of two new half height Teac 55B's. After all I explained to my wife, how else can I make sure it will handle double-sided, double-density, multidrive operation? After confiding that she didn't give a 'teac', I concluded that she concurred with my decision. That was fortunate since I had already purchased the new hardware. Approximately 12 hours, 2 manuals and a dozen phone calls later I had successfully integrated all the new hardware. The biggest single problem I had (in retrospect) was determining the correct strapping on the two Teac drives. The only thing left, once this was accomplished, was to set the head step time with the switches on the inside of the controller card. There are four screws holding the shell together. With those removed, the case separates, exposing, among other things, a set of dip switches which control the head step time. Page 5 of the instructions neatly show the proper settings. Experimentation shows that the Teacs will function on 6ms but do better on 3ms (quicker/smooth). It is about this time that you notice that there are four little washers now scattered around in the box and/or the floor. Don't despair. Take four toothpicks and stick them through the board and into the bottom of the shell. Now drop the washers, one each over the toothpicks. Carefully place the top half of the shell over the protruding toothpicks and clamp the sides together. Remove the toothpicks one at a time and replace with the screws. Incidentally, be sure that the heat sink (little metal gizmo) stays in place under the one corner. The most observant of us might remark that this is perhaps not the easiest way to set dip switches. There is a reason for this however: 1. to test one's manual dexterity and 2. to save you, the consumer \$\$\$\$. This is something that can be appreciated by all of us. Compare the price of this card with the price of TI's when it first came out.

Having done this...and it really isn't as hard as it sounds, I proceeded to initialize my first ever DS/DD disk. Imagine the thrill of watching the 'verifying sectors' count zoom right on by 360...720....all the way to 1440! I don't when I've had so much excitement.

Tune in next month for the next exciting chapter in the ongoing saga of the CorComp Controller Card!

## HOMEWORK HELPER

By Darrell Ingold

This cartridge/disk program from Navarone is designed to help organize the student's class assignments and facilitate the typing/printing of book reports and other written assignments. It is organized into various segments which allow the user to define his hardware set-up (printer, disk drives, etc), student's name and address as well as the name and address of the school. It can then be saved on disk and the pertinent information on names etc. will appear on any document printed under the "Book Report" or "Typewriter" function. It provides a nice, clean format that any teacher would appreciate.

The word processing aspect of the program is certainly not up to the standards of TI-Writer but it does not claim to be. The screen scrolls as the student types but there is no overlap between screens so words are split. For example, the word "coming" might show "com" on the right hand edge of the screen as it reaches the edge of the window and then when the screen windows right it becomes blank and the "com" has disappeared leaving no reference for the typist. This can be confusing, especially if the student is not a very good typist. It is something that, I'm sure, one could become accustomed to, especially if he was not already using a more sophisticated word processor. I do like the fact that the program allows the user to pick his own background, foreground and highlight colors. All of the set-up, names etc. can be saved and become default for the user. Done in this way it is easy for several students to use the program and just keep their own disks. There is a spelling checker that will help the user find the correct spelling of words, even if he does not know any more than the first few letters. The checker will also proof the spelling of an entire document. This is done at a fairly fast rate but when a word is found to be 'not in the dictionary' the student must make note of it and go back later to edit it. Also the 20500 word dictionary is obviously not complete but serves most of the vocabulary of the highschooler, and perhaps even some college students. Having not reviewed the entire vocabulary of the dictionary, I would not want to say it was incomplete but some more common words such as "interesting" did not show up. Keep in mind however, that this program should only be an aid to the student. While the so-called shortcomings mentioned above may be inconvenient for the adult user, they are just right for the student, who needs to "make note of the mis-spellings" by hand before going back to change them. Remember that he is still learning and is not using this program for a living income.

Homework Helper would go a long way toward organizing the typical school student and make his workload seem easier...just because he can use his computer to do it. The module with the 20,500 word spelling Checker is available for your local computer supplier carrying Navarone Products for a List Price of \$49.95. You may contact the firm directly, if you desire, at the address listed below:

NAVARONE INDUSTRIES  
510 LAWRENCE EXPWY #800  
SUNNYVALE, CA 94086  
(408) 866-8579

## HINTS 'N TIPS

### FEAR OF FILING

By Mike Wilcouth - New Horizons

Businesses often spend hundreds of dollars for information management software. In fact, many computer systems are dedicated exclusively to file maintenance. It's not surprising when you consider that all a computer really does is manipulate data. Even your \$49.00 TI can be used to create and search a file of respectable size. How many times have you had someone ask: "What do you use your computer for?", or "What would you do with one?"? Next time say: "get organized!". The program in this article is a generic data base manager. You have to enter the data, of course, following a few simple rules:

1. Each record must be on a numbered line between 171 and 32766.
2. Each line number must be followed by the word DATA.
3. The line number should be included in the record so you can easily make changes. For example: 200 Data 200 etc. Just don't RESequence a file or the line number will not match.
4. No commas can be included in a record. Commas separate records.
5. Record length is limited to a maximum line length (4 screen lines).

This program isn't glamorous, but it is memory-efficient and more RAM is available for your file.

```
100 INPUT "ENTER SEARCH SPEC. ":A$      140 IF POS(B$,A$,1)=0 THEN 120
110 RESTORE                               150 PRINT B$::
120 READ B$                                160 CALL SOUND(100,1000,0)
130 IF B$="EOF" THEN 100                  170 GOTO 120
                                           32767 DATA EOF
```

When you run this, the computer will ask for a search specification. This can be any

alphanumeric string of any length. When you press ENTER the computer will begin looking through the records in your file. Each time it finds a coincidence of the search spec it will print the record on the screen. When the entire file has been searched, you will be asked for another search spec. If you have an address file, your spec may be a zip code, a last name, or a state, for example. To stop the program press CLEAR (FCTN 4). If you wish the search to be more exclusive, you can require that two or more specs be found. Add these lines:

```
102 INPUT "ENTER SPEC #2 ":C$          142 IF POS(B$,C$,1)=0 THEN 120
```

This article was copied from the December, 1984 issue of the Northwest Ohio 99'ers Newsletter.

TWO COLUMN PROGRAM LOADER - HANDLES FORTY PROGRAMS

The program listed below will list forty programs on the screen in two columns. This is the first two column loader that I personally have seen. What really is needed now is for some industrious soul to combine the reselection feature found in Tigercub's version and add it to the two column loader presented here. The CorComp Disk Controller can store up to 127 files on a disk. Part two of this challenge is to increase the number of files which can be read to 127. Oh yes, the LOAD program listed below only places on the screen actual programs on a disk and ignores all other file types, DIS/VAR as an example. Forward your answers to this challenge to the letterhead address.

```

100 ! *****
110 ! * GENERAL PURPOSE *
120 ! * DISKETTE MENU *
130 ! * BY A. KLUDGE *
140 ! * MODIFIED TO A *
150 ! * 2 COLUMN LOAD *
160 ! * BY PHIL BENNIS *
170 ! * AND BILL TIEP *
180 ! *****
248 ! 250 OPTION BASE 1 :: DIM PG$(40):: CALL CLEAR :: CALL CHARPAT(97,A$):: IF SEG$(A$,7,8)="3B447C44" THEN 280
260 DISPLAY AT(15,2):"ABCDEFGHJKLMNOPQRSTUVWXYZ"
270 FOR I=65 TO 90 :: CALL CHARPAT(I,A$):: B$="0000"&SEG$(A$,1,4)&SEG$(A$,7,4)&SEG$(A$,13,4):: CALL CHAR(I+32,B$):: NEXT I
:: CALL KEY(5,I,1)
280 IMAGE ##
290 D$="1"
300 D$="DSK"&D$&". " :: OPEN #1:D$,INPUT,RELATIVE,INTERNAL :: INPUT #1:N$,A,A,A :: DISPLAY AT(24,1)ERASE ALL:D$@N$ :: I=0
310 FOR X=1 TO 20 :: I=I+1 :: IF I>127 THEN K=X :: GOTO 440
320 INPUT #1:P$,A,B,B
330 IF LEN(P$)=0 THEN 420
340 IF ABS(A)<>5 THEN 320
350 DISPLAY AT(X+2,0):USING 280:X :: DISPLAY AT(X+2,4):P$ :: PG$(X)=P$ :: NEXT X :: IF X<21 THEN 420
360 FOR X=21 TO 40 :: I=I+1 :: IF I>127 THEN K=X :: GOTO 440
370 INPUT #1:P$,A,B,B
380 IF LEN(P$)=0 THEN 410
390 IF ABS(A)<>5 THEN 370
400 DISPLAY AT(X-18,16):USING 280:X :: DISPLAY AT(X-18,19):P$ :: PG$(X)=P$ :: NEXT X
410 DISPLAY AT(X+6,16):USING 280:X :: DISPLAY AT(X+6,19):"TERMINATE" :: DISPLAY AT(1,2):"CHOICE? 1" :: GOTO 430
420 DISPLAY AT(X+2,0):USING 280:X :: DISPLAY AT(X+2,4):"TERMINATE" :: DISPLAY AT(1,2):"CHOICE? 1"
430 ACCEPT AT(1,10)SIZE(-2)VALIDATE(DIGIT):K
440 IF K=X THEN CALL CLEAR :: CLOSE #1 :: END
450 IF K<1 OR K>40 OR LEN(P$(K))=0 THEN 410
460 CLOSE #1
470 CALL INIT :: CALL PEEK(-31952,A,B):: CALL PEEK(A6+B-65534,A,B):: C=A6+ B-65534 :: A$=D$PG$(K):: CALL LOAD(C,LEN(A$))
480 FOR I=1 TO LEN(A$):: CALL LOAD(C+I,ASC(SEG$(A$,I,1))):: NEXT I :: CALL LOAD(C+I,0)
490 RUN "DSKX.1234567890"
    
```

NOTES ON THE TI ADVANCED ASSEMBLY LANGUAGE DEBUGGER  
By Richard J. Bailey - New Hampshire '99'ers

One excellent purchase from your Users Group's Library is the debugger disk. Unfortunately the description of the disk is incorrect. The disk can be used not only with the Editor/Assembler module but also with TI Basic and Extended Basic. You still need to have 32K of added memory (Expansion Memory) and the disk drive to use SUPER-DEBUGGER. The help file on the disk can be printed or viewed on the screen with one of the following short programs. It isn't too practical to view the HELP file on the screen since it is seven (7) printed pages long! If you use the printer version, change the printer number in line 120 to agree with your printer and use drive 1 if you have more than one drive. The help file is basically an instruction manual on disk. With the screen version you can press and hold 'ENTER' to halt the listing.

SCREEN VERSION	PRINTER VERSION
-----	-----
100 CALL CLEAR	100 CALL CLEAR
110 OPEN #1:"DSK1.HELP"	110 OPEN #1:"DSK1.HELP"
120 LINPUT #1:B\$	120 OPEN #2:"RS232.BA=2400.DA=8"
130 PRINT B\$	130 LINPUT #1:B\$
140 CALL KEY(0,KY,ST)	140 PRINT #2:B\$
150 IF KY=13 THEN 140	150 GOTO 110
160 GOTO 120	

Enter and run the following program to see how you could enter the debugger with Basic or Extended Basic. Refer to the help file for more instructions on its use.

```

100 CALL CLEAR
110 CALL INIT
120 INPUT "PUT DEBUGGER DISK IN DRIVE 1,
    PRESS ENTER ";DUMMY$ 130 PRINT 140 PRINT "PRESS 'N' WHEN THE LOADING STOPS.
ENTER PRINTER NAME THEN 'DA000 A140' TO DUMP TO THE PRINTER." 150 CALL LOAD("DSK1.SBUG")
160 CALL LINK("SBUG")
    
```

TEX-BUG TIDBITS  
By Darrell Ingold  
Tex-Bug President

This is the summary of all that's new, even if it's not new. For starters we have a brand new year and a new opportunity to commit ourselves to taking greater advantage of our computers to keep track of all those important figures for our tax records, inventory etc.

Secondly, we have a new publication coming to each of us as part of our regular membership dues; THE NATIONAL 99ER! (see attached newsletter). In a recent board meeting held in the back office of an Oildale fire station, your Board of Directors elected to make the official Tex-Bug newsletter this one page attachment to the NATIONAL 99ER. Since this is a national publication, it is (as you may have noticed) mailed out of Little Rock, Arkansas along with everyone else's across the country; BUT yours and yours alone will contain this dazzling display of local TEX-BUG data for your digestion. This decision will not in any way affect the amount of your dues now nor at renewal time. We felt that the members would benefit greatly from the information in the NATIONAL 99ER and that since the normal private subscription rate for the 99ER is \$12.00 annually, it would increase the value of your membership in TEX-BUG as well.

Next item of note is the change of meeting dates. The meetings, incidently, are still your best source of what's going on in the TEX-BUG world. They will now be on the **SECOND THURSDAY** of each month beginning promptly (if I'm officiating) at 7:00 p.m. (1900 hours for you military types). Refreshments will be served and meeting concluded by 8:30 p.m.. Each meeting will include tickets for a doorprize(s), a short business meeting (mostly informational) and some sort of demonstration. The librarian will be available for members to purchase software from the club library. New programs are presently being sought on exchange with other clubs and we hope to soon have an even greater variety for you. The meeting schedule for the first 6 months of 1985 will be:

JANUARY 10, 1985	APRIL 11, 1985
FEBRUARY 21, 1985	MAY 9, 1985
MARCH 14, 1985	JUNE 13, 1985
THERE WILL BE NO MEETINGS DURING JULY - AUGUST, 1985	

Although this letter will not likely reach you prior to the meeting in January, the planned demonstration will be the use of the Personal Record Keeping module and the ability of the Personal Report Generator module to manipulate the PRK data. Next month will bring a complete listing of the Winter/Spring meeting demonstrations. Also watch the Californian's column "Computalk" by Ken Durham each Monday for other updates and info on TEX-BUG.

If it is not already apparent, I will be your club President for the new year. You will be hearing from me from time to time but it is **MOST IMPERATIVE THAT I HEAR FROM YOU** also from time to time on your needs as club members. My home phone is 399-8686. Thank you.

One last thought: there will be a series of 5 classes on programming in TI BASIC starting on Tuesday, February 26, 1985, from 7:00 p.m. to 9:00 p.m.. Cost of the class will be announced shortly.

**SUBSCRIPTION FORM**  
FOR "THE NATIONAL NINETY-NINER" NEWSLETTER  
PUBLISHED BY THE 99'ERS ASSOCIATION

NAME: \_\_\_\_\_ DATE: \_\_\_\_\_

ADDRESS: \_\_\_\_\_

CITY: \_\_\_\_\_ STATE \_\_\_\_\_ ZIP \_\_\_\_\_

MAIL CK./M.O. FOR \$12.00 TO:

THE 99'ERS ASSOCIATION  
3535 SO. H ST., #93  
BAKERSFIELD, CALIF. 93304

ATTN: LUCI VEITH

12/84

**THE NATIONAL 99'ER**  
3535 SOUTH H STREET #93  
BAKERSFIELD, CA 93304



BILL PECHNIK  
1467 CARMi DR.  
PENTICTON, BC  
CANADA V2A4R9

(133)