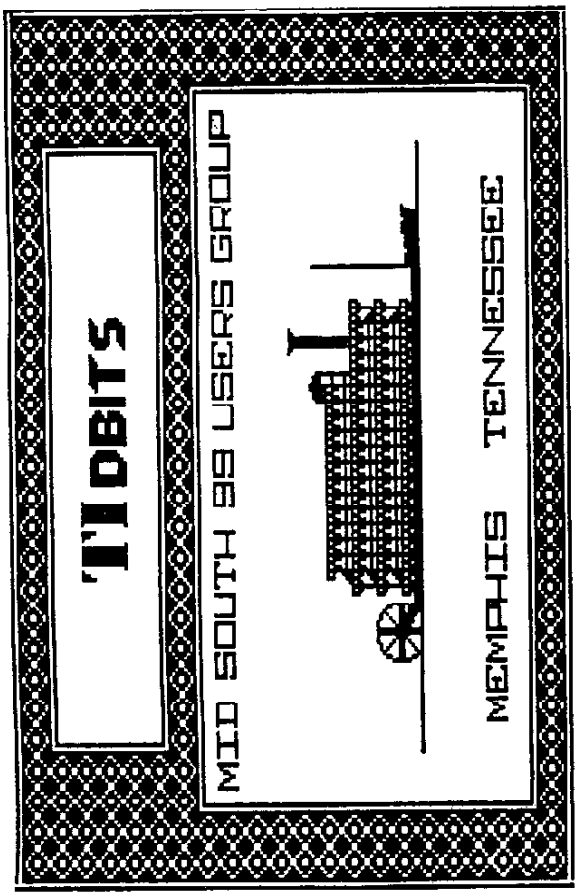
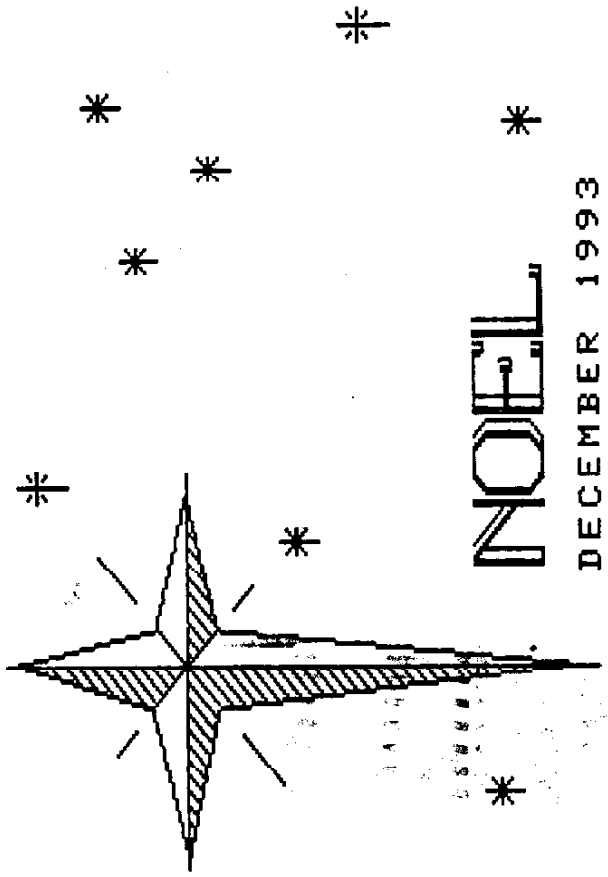




Mid-South 99 Users Group
P. O. Box 38522
Germantown, Tn. 38183-0822

MEMPHIS, TN
PM
DEC 9
1993
381

Greetings
29
USA

FIRST CLASS MAIL

UG 2/86
DALLAS TI USER GROUP
P.O. BOX 29863
DALLAS, TX 75229

75229-0863

Newsletter of the MID-SOUTH 99 USERS GROUP-Vol 11,#12-DEC 1993

TIDBITS

OFFICERS

Gary Cox	PRESIDENT	901-358-0667
Richard Hiller	VICE-PRESIDENT	901-794-9945
Richard Mann	SECRETARY	901-682-4195
Mac Swope	TREASURER	901-363-3880
Jim Saemenes	Technical Support	901-476-7011
Jim Saemenes	Disk Librarian	901-476-7011
Pierre Lamontagne	CO-Librarian	901-386-1513
Gary Cox	Program Chairman	901-358-0667
Mac Swope	Chairman - Equipment	901-363-3880
Marshal Ellis	Editor - TIDBITS Newsletter	901-327-2506
Marshal Ellis	Editor-Technical Interface	901-327-2506
Beery Miller	9640 NEWS BBS Sysop	901-368-0112

DEC. 1993 INDEX

PRESIDENT'S BIT

By Gary W. Cox

This month will be our regular Christmas party/meeting. The group will be providing the drinks, cups, plates, knives and forks and everyone else needs to bring things to eat such as chips, dip, cookies, cake, sandwiches or just whatever will be good for a party. This is usually our biggest meeting of the year so everyone please try to make it to this one!

Nominations for the offices of President, Vice-President, Secretary and Treasurer were made last month with the results as follows:

President - Mac Swope Vice-President - Richard Hiller

Secretary - Marshal Ellis Treasurer - Gary Cox

The election will be held at this months meeting.

Again please try to make this months meeting, we still have some software to demonstrate which was obtained at the Chicago TI Faire!

HOW TO ACCESS / FILES

by Jerry Keisler
L.A. 99ers, Nov., 1987
from the pages of the WILL County UG newsletter, Oct.1992

I have been asked by several members how to run some of the programs on our DOM. The following is a list of what you mayt find in your disk directory and how to run it.

If your disk has a LOAD file, it may run all the files on the disk regardless of type.

PROGRAM FILES (PG)

There are several options for running these files.

EXTENDED BASIC (XB)

Will load and run automatically when you select XB and the disk is in drive #1, if it has a LOAD file. Or can be run by typing:

RUN (enter) or RUN "DSK1.name" (enter)

If the program loads correctly but you get a BAD VALUE error when it runs, you need to load the progran into Basic (no CHARS above 143 are allowed in XB). If the program file is more than 45 sectors and won't load, you have to open up more memory. Do this by typing:

CALL FILES(1) (enter) NEW (enter)
OLD DSKn.name (enter) RUN (enter)

BASIC

Programs need to be loaded by typing:

OLD DSKn.name (enter) RUN (enter)

Most Basic programs will load and run in XB but not visa versa. If you get a FOR-NEXT error in line xxx and when you edit the line you get a lot of nonsense, the program is written in XB The same is true if the sectors are greater than 45. More

space is needed in the conputer. See CALL FILES above. If you still get a memory full and tried XB then most likely it can be run on tape (OLD C31) without the expansion box turned on.

EDITOR ASSEMBLER (EA)

If a program file will not load and run in Basic or XB and gives an I/O ERROR 50, it may be an assembly language program. These can be run using BOOT, the Editor Assembler module option #5 with DSKn.name or FUNNELWEB's RUN option of Disk Review. These program files are listed in consecutive order such as MASS, MAST, MASU or UTIL1, UTIL2, UTIL3. The files will normally have 33 sectors.

GRAMULATOR

These require a GROM simulator card or box. The files contain 34 sectors and have the same name with the numbers 1 through 66 attached to files 2 through 8. Follow your GROM simulator instructions to load and run.

OTHER PROGRAM FILES

Some specialized program fles can only be loaded from a special module such as ADVENTURE (54 sectors), PRESONAL RECORD KEEPING, STATISTICS, TUNNELS OF DOOM (52 sectors).

DIS VAR 80 FILES (DV80)

These are usually text or documentation files (DOCS, READ ME, ETC.). They are instructions on how to run programs on the disk. Read or print them using TI Writer, FUNNELWEB, BOOT or the (V, T or P) option of several disk managers.

DIS VAR 163 FILES (DV163)

This is an XB file in MERGE format. It can be merged into a program already in computer memory. Type:

MERGE DSKn.name (enter)

You must do this even if no program is in memory. To save a file in MERGE format, type:

SAVE DSKn.name, MERGE in XB only.

DIS FIX 80 FILES (DF80)

Use EDITOR ASSEMBLY MODULE, LAR or FUNNELWEB to load these. Some files will auto load and/or auto start. Use LOAD and RUN option #3. Type:

DSKn.name (enter)

If the program does not run but asks for a second file name, you must do #1 and #2 or just #2 below.

#1. If there are multi files for the program, type:
DSKn.name (enter) for each file.

#2. Then press enter. If the program does not start, enter the program name. The progam name could be START, BEGIN, GAME, LOAD, RUN, FIRST, etc. FUNNELWEB will give you a list of names found in the program.

DIS FIX 128 FILES (DF128)

These are usually ARCHIVED files. You must unarchive these files before you can run them. Use ARC303 or ARC303G.

INT VAR 254 FILES (IV254)

These files usually have more than 45 sectors and are Extended Basic requiring memory expansion. They do not require CALLS FILES(1). Basic cannot be used. The same commands are used such as RUN or OLD DSKn.name. The programs are usually so long that they cannot be saved to tape. (SAVE CS1)

DATA FILES

Files such as INT FIX 108, INT VAR 128, INT VAR 64 and some program files are data files that can be used by a program on the disk. They will not RUN and should be left on the disk with the other programs.

QUICK REFERENCE

TYPE	SIZE	TRY ---- ---- ----------
PG	----	XB, Basic, EA CART
PG	33	EA #5, BOOT, FW
PG	34	GROM SIMULATOR
PG	52	TUNNELS OF DOOM
PG	54	ADVENTURE
DV	80	TI-WRITER FIC
DV	63	XB MEREGE
DF80	--	EA #3
DF128	--	ARCHIVER
IV254	--	XB
IV245	--	XB
ANY	--	DATA

SOME OTHER FILES

ENDS WITH	TYPE	DESCRIPTION
		TI-BASE
/P	IF255	PROGRAM
/H	DV80	HELP
/C	DV80	COMMAND OR
/C	IF40	COMMAND
/D	IF	DATA BASE DATA
/S	IF255	DATA BASE STRUCTURE
		TI ARTIST
_C	25PG	PICTURES COLOR
_P	25PG	PICTURES PATTERN
_F	DV80	CHARACTER FONT
_S	18DV80	SLIDES
_I	DV80	INSTANCE
_V	DF12	VECTOR
_M	DV254	MOVIE

TRANSFERRING FILES

by Dick Ohi
West Penn 99"ers
from the pages of:Los Angeles 99ers

BETWEEN THE TI-99/4A AND IBM CLONES

TOPICS

You will need the program "PC Transfer" and "PC Transfer Utilities" for the TI-99/4A and Geneve 9640 by Mike Dodd. "PC Transfer" requires a CORCOMP or MYARC disk controller and two double sided disk drives. These programs are distributed by 9640 News & Beery Miller. the following was tested using a TI-99/4A and a Gateway 2000 IBM clone running Quattro Pro SE. These procedures should work with any IBM spreadsheet that will import files in the Symbolic Link Format.

Begin by running Multiplan on your TI. Load the file you wish to transfer to the IBM.

1. Press T(ransfer), O(ptions), S(ymbolic), ENTER. this sets all transfer operations to the symbolic link format.

2. Press T(ransfer), S(ave) and type in a new file name so that you do not overwrite the original file, or you may want to save the new file to another diskette. (Recommended)

3. If you want to transfer more than one file you have to reset the normal mode for Transfer operations. Press T(ransfer, O(ptions), N(ormal), ENTER. Load the next file to be converted and repeat steps 1 and 2 above.

4. When you have saved all the files you wish to transfer, exit Multiplan and insert the Extended Basic cartridge in the console. Insert the PC Transfer diskette in drive 1 and select Extended Basic, PC Transfer will auto load from DSK1.

5. Using the prompts on screen, select a DOS drive and a TI drive.

6. At the "Conversion File Name" prompt type: DSK1.SYLK and press ENTER.

7. You may now remove the PC Transfer diskette from the drive. Insert the disk with your TI files in the designated TI drive, and either a blank diskette or a DOS formatted diskette in the designated DOS drive.

Note: It is possible to format a DOS diskette with "PC Transfer" but it is a very slow process. I recommend using a formatted 360k DOS diskette.

8. Load the TI disk catalog.

9. Select the files to be transferred by pressing C when the cursor is next to the file name. Use the space bar or down arrow to move down through the list of files, the up arrow allows you to move back up the list.

10. When all files hve been selected, press E to execute the procedure.

11. You will be asked for a DOS file name for each file to be transferred. type in a file name using up to eight characters plus a period and SLK (e.g. MYFILE.SLK) and press ENTER. The file.

12. When all the selected TI files have a DOS file name entered the transfer procedure will begin.

Take the DOS diskette to your PC clone, load your spreadsheet program and either open or import the file from the diskette. All data and formulas should be transfferred to the DOS spreadsheet. You may have to make some changes on some cells as to how the data is displayed.

SPEEDING UP FILE HANDLING IN BASIC

by John Wineke
MADAREA 99ER NEWS
Jan.-Feb. 1990

It is often convenient to write a Basic program to handle specific filing tasks. For instance I have a Mail List Porgram to handle our church membership and newsletter names and addresses. When you have to read in a large number of files from Basic it gets to be a rather long process. By experimentation I have found several ways to speed things up a bit, and I offer them for your perusal.

The program I wrote currently has 92 names with 6 fields per name for such things as address, phone #, etc. I could have used an existing data base program, but preferrred the freedon of having up to 80 characters per field and not having to set any predefined limits on myself. This way, if an address was particularly long; no problem! And if a field was blank, no memory was wasted other than that required to dimension for a string.

Hint #1: Use Internal, Variable 80 files rather than Display/ Variable 80. this allows you to input numerical data as well as strings, and is also faster. I cut file handling time 23% by doing this. I know that in the manual they lead you to believe otherwise, but seeing is believing.

Hint #2: Put all the input and output program loops in one line. By cutting 3 program lines to one using multistatement lines my files loaded 5% faster. (See listing below)

Hint #3: Cut out extraneous program lines. (Bells and whistles) By removing a DISPLAY AT statement which showed each name as it was brought off disk I saved 9 seconds (15%) when inputting my 552 fields.

Hing #4: Use a RAM disk or hard disk. Actually they don't save as much when using Basic as you might suppose, because much of your time is Basic overhead, but it did save me 12 seconds (20%).

Hint #5: Use a Basic Compiler. Now this last idea is really impractical, but I'm throwing it out to show the effect it would have if it were practical. It is impractical because when you dimension your program (mine uses 100,7) the compiler assumes you are going to use the whole dimension up with whatever you set your string limit at. My Ryte Data (Peter Kull) compiler requires you to set a max string limit, and allow no more than 64 characters. Well, when you do that you have no room left for your program.

Anyway, I cut down the size of my dimension and compiled the program and used porportioninc to come up with the time saved. I figure that the whole file would have input in 30 seconds (Of a RAM disk). By not using the above techniques the file took 78 seconds.

Hint #5: This is not a speed hint, but rather a good piece of advice. USE ERROR TRAPPING! If you don't you are likely to loose your entire file somewhere along the way. Now that would really be a waste of time when you consider all the work it took to type it in. I used the error trap shown in lines 110 and 1720 to 1760 below. You need a complex trap like this with I/O because you usually get two errors instead of one.

```
100 DIM A$(100,7)
110 ON ERROR 1720 :: GOSUB 1110
115 INPUT "DATE?" : DATE$
120 INPUT "1=NEW FILE 2=OLD FILE ": OP1 :: IF OP1>2 THEN
    120 ELSE IF OP1=1 THEN 200
140 INPUT "INPUT FILE NAME? ": Y$ :: OPEN #3: Y$, INPUT,
    INTERNAL, VARIABLE 80
150 CALL CLEAR :: INPUT #3: FYL, A, B$
160 FOR I=1 TO A :: INPUT #3: C$(I), T(I) :: NEXT I
170 FOR I=1 TO FYL :: FOR J=1 TO A
180 INPUT #3: A$(I,J)
190 NEXT J ::DISPLAY AT(24,1): A$(I,1) :: NEXT I :: CLOSE
    #3 :: GOTO 460
195 REM (AND SO ON . . . .)


1720 ON ERROR 1730
1730 CALL ERR(D, S, J, K) :: PRINT :"ERROR" ;D; "IN LINE"
     ;K; :: FOR I=1 TO 3 :: CALL SOUND(900,220,0) ::NEXT I
1740 IF D=130 AND CL=1 THEN CL=0 :: RETURN 470 ELSE IF
     D=130 THEN CL=1 :: CLOSE #3 :: RETURN 470
1750 IF D=109 AND CL=1 THEN OPEN #3: "PIO" :: CLOSE #3 ::
     CL=0
1760 RETURN 470
```

USING THE SCREEN AS A FILE

by Chick DeMarti
Los Angeles 99ers
from TIBUG, Australia
from the pages of the Chicago TIMES newsletter, April '90

Quite often, when working on a prog. that stores data to a file, during the testing, it is necessary to actually write data to one of these files and that data is often formatted in some way special to the program being developed. This method can be quite time and for paper consuming if the data or its format are not being saved exactly the way it is wanted. In TI-Basics, we can have several files opened at one time, each numbered different. We normally start with file #1, then #2, etc. Exam. :

```
OPEN #1 : "PIO" :: OPEN #2: "DSK2. DATALIST", OUTPUT
(etc.)
```

The Extended Basic manual states that you can number files from 0 (zero) to 255 and on page 147 relating to the PRINT

statement if the file number after the PRINT statement is omitted or is 0 (zero), then output goes to the screen.

NOTE: The file to the screen (#0) is always OPEN and does not require an OPEN #0 directive.

Here then is the answer to our needs. At the beginning of the program use a variable for the number of the file being written to. e.g.

```
10 @=0.
```

When you open the file, do it in the normal way such as OPEN #1: "DSK2.DATALIST", etc. but every time you would normally print to the file, use the PRINT #@: A$ as the directive. Even though the file to disk will be opened all info will be printed to the screen. When the program is working just as you want it to, then you can change line 10 to @=1, or whatever file number you opened the file as, and that file will be written to. This system will work perfectly if the information being stored is of the DISPLAY kind. If the files uses the INTERNAL format, only those bytes in the ASCII character range will be seen on the screen.

Well I am out of coffee. See you next month. Chick

JUST A NOTE

"Every child comes with the message that

God is not yet discouraged of man."

< Rabindranath Tagore

MORE ABOUT CONVERTING

by Charles Good
Lima Ohio
Users Group

WORD PERFECT 5.1 FILES
TO TI DV80 FORMAT
from the pages of the Spirit of 99 newsletter, June 1993

In my March 1993 review of Mike Wright's TI CYC I described a method of converting the MS-DOS CYC disk files to TI DV80 format usabel by those who do not have access to Word Perfect 5.1. If you DO HAVE WP5.1 there are several ways to save a file from Word Perfect 5.1 to ASCII format so that the file can be converted with PC TRANSFER to DV80 on a TI disk.

-- Use F5, Save File, and select "Generic".

-- Use F3, and select DOS to save the file.

-- Format the file to your particular printer (the CYC is formatted for laser printer output). Then change the printer name to a disk file and PRINT the file to this disk file. Use the MS-DOS 5.0 text editor to remove the control code junk from the file and resave. The resulting file will be in ASCII format and when you use PC TRANSFER to convert the file to a TI disk the file will print very nicely exactly as it was formatted (in columns for example). This latest article was submitted by Harold hoyt.

WHAT CAN A COMPUTER DO FOR ME?

by Joan Johnson
UGOC ROM

from the pages of the SPIRIT OF 99, July 1993

Before I purchased my TI (short for Texas Instruments), I used to think to myself, "Why do I need a computer?" What would I use it for? All my children had computers, but I put off buying one. I didn't want to spend two thousand dollars for a machine I might never use!

All that changed in July of 1991 when my husband and I attended a "free" computer class at the Orange Senior Center. In a way, that meeting changed our lives.

After learning a few things that the computer could do and finding out how cheaply we could get started with a TI-99/4A computer, we were hooked. Now we can't live without it. Well, I guess we could, but we wouldn't have as much fun.

Here are some of the ways in which we use our computer. Others use their computers for different purposes.

1. WRITING LETTERS . . . The word processing program is great! Anyone can produce a professional looking letter. This article is being produced with the word processing program. When I have it exactly as I want it, I will instruct the computer to print it. If I'm not satisfied, I'll edit it, but I DO NOT have to retype the entire article.

2. MAKING LABELS . . . We no longer handwrite addresses on envelopes. We use the computer to make a label, peel it off,

and slap it on the envelope. The Postal Service appreciates typed addresses because it speeds up their sorting process.

We send labels with our address on them to our friends around the country. All they have to do is peel one off and stick it on an envelope. Saves them from having to look up our address in their rolodex, write out the envelope, etc.

3. SPREADSHEETS . . . It wasn't too long ago that I didn't know what a spreadsheet was. Now we use such a program to track investments, help figure income tax records, sort lists of family names and addresses for genealogy purposes and many other uses.

4. TRACK SAVINGS BONDS . . . There is a program that will help you keep track of the current value of your bonds.

5. HOUSEHOLD INVENTORY . . . Using the computer to list your household items by name, date of purchase, purchase price, serial number, etc. is a valuable tool for insurance purposes in the event of theft or fire.

6. CREDIT CARD NUMBERS . . . Making a list of all your credit card numbers including the telephone numbers to call if the card is lost or stolen, is another way to use your computer. Just remember to file your list in a safe place.

7. GAMES . . . Some people think that is all a computer is used for. The fact is that a computer can be a friend to a lonely person. You can play games such as bridge, poker, cribbage, backgammon, chess, you name it, with the human playing against the computer. It can be a lot of company. The uses are unlimited.

8. TELECOMMUNICATIONS . . . Using a modem, you can hook up to other computers and bulliten boards via the telephone lines.

Maybe the above ideas will get you thinking of how you could use a computer. Operating a computer does require some effort on the part of the user, however. Even though some computers are more "user friendly" than others, all require a training period to learn to operate them. MANUALS ARE A MUST! Getting a free computer is no bargain unless you get the manual with it.

Maybe you thought you never came in contact with a computer. Well, think again. Everyone probably drives a car, uses a microwave, VCR, calculator, telephone, bank ATM card. All of these everyday items have microprocessors in them, and they are computers.

Many modern automobiles have a part referred to as a "brain". If the brain goes bad, the car won't run. That is only one of the computers in the car. Air bags use microprocessors and sensors to tell them when to inflate.

We can't go back to the days before computers, we probably wouldn't want to.

CONVERTING MIDI MASTER 99

by Jim Peterson
Spirit of 99, Mar, 93

SNF FILES

A corresponeent, who likes to practice by playing his instruments along with computer music, remarked that he wished MIDI music could be converted to Basic music, because it is so easy to change key in Basic but so difficult with an SNF file. I understand that some of the more expensive keyboards allow you to change key from the panel, but users of the Casio MT-240 do not have that option. Even those who have the better keyboards might like to permanently change some music. Anyway, I wrote this little TRANSPOSER utility.

```
100 DISPLAY AT(3,8)ERASE ALL : "TRANSPOSER": "" : "
FOR USE WITH MIDI MASTER" : "" : "      SNF FILES" :
    "" : "     by Jim Peterson"
110 DISPLAY AT(16,5) : "Initializing. . ."
120 DATA 0C, 0C#, 0D, 0E@, 0E, 0F, 0F#, 0G, 0A@, 0A,
    0B@, 0B
130 DATA 1C, 1C#, 1D, 1E@, 1E, 1F, 1F#, 1G, 1A@, 1A,
    1B@, 1D
140 DATA 2C, 2C#, 2D, 2E@, 2E, 2F, 2F#, 2G, 2A@, 2A,
    2B@, 2B
150 DATA 3C, 3C#, 3D, 3E@, 3E, 3F, 3F#, 3G, 3A@, 3A,
    3B@, 3B
160 DATA 4C, 4C#, 4D, 4E@, 4E, 4F, 4F#, 4G, 4A@, 4A,
    4B@, 4B, 5C
170 DATA 0C, 0D@, 0D, 0D#, 0E, 0E#, 0G@, 0G, 0G#, 0A,
    0A#, 0A#, 0B
180 DATA 1C, 1D@, 1D, 1D#, 1E, 1E#, 1G@, 1G, 1G#, 1A,
    1A#, 1B
190 DATA 2C, 2D@, 2D, 2D#, 2E, 2E#, 2G@, 2G, 2G#, 2A,
    2A#, 2B
200 DATA 3C, 3D@, 3D, 3D#, 3E, 3E#, 3G@, 3G, 3G#, 3A,
    3A#, 3B
210 DATA 4C, 4D@, 4D, 4D#, 4E, 4E#, 4G@, 4G, 4G#, 4A,
    4A#, 4B, 4C, 5C
220 DIM C$(61) :: FOR J=1 TO 61 :: READ A$ :: DISPLAY
    AT(18,12) : A$ :: L=LEN(A$) :: C$(J) = A$ :: V$ =
    V$& C$(J)& RPT$(" ", 3-L) :: NEXT J
230 FOR J=1 TO 61 :: READ A$ :: DISPLAY AT(18, 12):
    A$ :: L = LEN(A$) :: V2$ = V2$& A$& RPT$( " ",
    3-L) :: NEXT J
240 DISPLAY AT(10,1)ERASE ALL :" Input filename?" :
    "DSK" :: ACCEPT AT(11,4) BEEP : IN$ :: OPEN #1 :
    "DSK"& IN$, INPUT
250 DISPLAY AT(13,1): "Output filename?" : "DSK" ::
    ACCEPT AT(14,4) BEEP : OUT$ :: OPEN #2 : "DSK"&
    OUT$, OUTPUT
260 DISPLAY AT(16,1) : "Transpose by how many keys?":
    "( plus or minus )" :: ACCEPT AT(17, 17) VALIDATE
    (NUMERIC) BEEP : TR
270 IF EOF(1) = 1 THEN CLOSE #1 :: CLOSE #2 :: STOP
    ELSE LINPUT #1:M$
```



```
280 IF SEG$(M$, 1, 2) = "17" OR ASC(M$) < 49 OR
    ASC(M$) > 57 THEN PRINT #2: M$ :: GOTO 270
290 P = POS(M$, ",", 1) :: A$=SEG$(M$, 1, P) :: B$ =
    SEG$(M$, P+1, 255) :: P = POS(B$, ",", 1) :: X$=
    SEG$(B$, 1, P-1) :: Z$ = SEG$(B$, P, 255)
300 IF X$ = "R" THEN PRINT #2 : M$ :: GOTO 270
310 X = POS(V$, X$, 1) / 3 :: IF X <> 0 THEN 330
320 X = POS(V2$, X$, 1)/ 3 :: IF X = 0 THEN PRINT #2:
    "1"& M$& " INVALID!" :: PRINT M$& " INVALID!" ::
    GOTO 270
330 IF X+TR > 0 AND X+TR < 62 THEN TR$ = C$(X+ TR) ::
    GOTO 360
340 DISPLAY AT(19,1) : M$ : " transposes out of
    range." : " Want to substitute? (Y/N)" :: ACCEPT
    AT(21,27) VALIDATE ("YN") SIZE(1) BEEP: Q$ :: IF
    Q$ = "N" THEN TR$ = "   " :: GOTO 360
350 DISPLAY AT(24,1): "Sustitute?" :: ACCEPT AT(24,
    13) BEEEP : TR$ :: DISPLAY AT(19,1) :"" : "" : ""
    : "" : "" : ""
360 PRINT #2 : A$& TR$& Z$ :: GOTO 270
```

I have wished that my keyboard had a "silent instrument" that I could temporarily assign voices to, in order to listen to the harmony or bass line by itself. Since it doesn't, I wrote this SPLITTER utility to take an SNF file apart into a seperate file for each track. It writes in a dummy 1, R, S because I find that, for some reason, some single-track files above track 1 will not run otherwise.

```
100 DISPLAY AT(3,1) ERASE ALL : "SNF FILE SPLITTER"
    ::DISPLAY AT(8,1): "How many voices to seperate?"
    :: ACCEPT AT(9,1) VALIDATE(DIGIT) SIZE(1) BEEP:V
110 IF V>8 THEN DISPLAY AT(12,1): "Sorry - 8 is the
    limit!" :: STOP
120 IF V>2 THEN DISPLAY AT(12,1): "Did you do a CALL
    FILES("& STR$(V+ 1)& ")" : "before loading this
    program?" :: ACCEPT AT(14, 1) VALIDATE("YN")
    SIZE(1) : Q$
130 IF Q$ = "N" THEN DISPLAY AT(12,1) :"Enter CALL
    FILES("& STR$(V+ 1)& ")" : "Then enter NEW" :
    "Then load and run program again." :: STOP
140 DISPLAY AT(8,1) ERASE ALL :"Input file?" : "DSK"
    :: ACCEPT AT(9,4) :IN$ :: OPEN #1: "DSK"& IN$,
    INPUT :: LINPUT #1: M$
150 DISPLAY AT(10,1) :"Output drive #?" :: ACCEPT
    AT(10,17) VALIDATE(DIGIT) : D
160 DISPLAY AT(12,1): "Output root filename?" ::
    ACCEPT AT(13,1): R$ :: IF LEN(R$) > 9 THEN DISPLAY
    AT(15,1) : "Limit 9 characters, please" :: GOTO
    160
170 FOR J=1 TO V :: OPEN #J+1 : "DSK"& STR$(D)& ","&
    R$& STR$(J), OUTPUT :: PRINT #J+1: M$: "1, R, S"
    :: NEXT J
180 IF EOF(1) <> 1 THEN 200
190 CLOSE #1 :: FOR J=1 TO V :: CLOSE #J+1 :: NEXT J
    :: STOP
200 LINPUT #1: M$ :: IF SEG$(M$, 1, 2) = "17" THEN
    220
```

```
210 X=ASC(M$) :: IF X>48 AND X<49+V THEN PRINT #X-47:
    M$ :: GOTO 180
220 FOR J=1 TO V :: PRINT #J-1 :M$ :: NEXT J :: GOTO
    180
```

And finally, someone asked about converting MIDI SNF files to Extended Basic music! I don't think that is a very worthwhile project, but I had to see if it could be done. This routine expects the tracks to be in 1-2-3 sequence and ignores any tracks above 3. In XBasic all voices must have the same duration, so I have used the duration in track 1. You can adjust the speed by changing the value of D in line 1 of the completed program, and you can adjust the volume of each voice in line 2 - something you can't do in MIDI! The program is not very efficient and perhaps not foolproof, but it doesn't seem worthwhile spending too much time on.

```
100 DISPLAY AT(12,1) ERASE ALL : "SNF filename?" :
    "DSK" :: ACCEPT AT(13,4) :IN$ :: OPEN #1: "DSK"&
    IN$, OUTPUT
110 DISPLAY AT(15,1): "Output filename?" : "DSK" ::
    ACCEPT AT(16,4) :OUT$ :: OPEN #2 :"DSK"& OUT$,
    VARIABLE 163 :: X=1
120 PRINT #2: CHR$(0)& CHR$(X)& "D"& CHR$(190)&
    CHR$(20)& CHR$(3)& "200"& CHR$(0) :: X=X+1
130 PRINT #2 : CHR$(0)& CHR$(X)& "V1"& CHR$(190)&
    CHR$(200)& CHR$(1)& "1"& CHR$(130)& "V2"&
    CHR$(190)& CHR$(200)& CHR$(1)& "5"& CHR$(130)&
    "V3"& CHR$(190)& CHR$(200)& CHR$(1)& "9"& CHR$(0)
    :: X=X+1
140 IF EOF(1) <> 1 THEN LINPUT #1 : M$ :: GOTO 160
150 PRINT #2 : CHR$(255)& CHR$(255) :: CLOSE #1 ::
    CLOSE #2 :: DISPLAY AT(12,1) ERASE ALL: "Enter
    NEW" : "" : "Then enter MERGE DSK"& OUT$ :: STOP
160 IF ASC(M$) < 49 OR ASC(M$) > 51 OR SEG$(M$, 1, 2)
    = "17" THEN 140
170 IF ASC(M$) <> 49 THEN 140
180 GOSUB 250 :: GOSUB 410 :: C$=CHR$(0)& CHR$(X)&
    CHR$(157)& CHR$(200)& CHR$(5)& "SOUND" & CHR$
    (183)
190 DU$ = STR$(D) :: C$ = C$& "D"& CHR$(195)& CHR$
    (200)& CHR$(LEN(DU$))& DU$& CHR$(179)
200 NU$ = STR$(N) :: C$ = C$& CHR$(200)& CHR$(LEN
    (NU$))& NU$& CHR$(179)& "V1"
210 LINPUT #1: M$ :: IF ASC(M$) <> 50 THEN PRINT #2:
    C$& CHR$(182)& C$& CHR$(0) :: C$="" :: X=X+1 ::
    GOTO 140
220 GOSUB 250 :: NU$=STR$(N) :: C$ = C$& CHR$(179)&
    CHR$(200)& CHR$(LEN(NU))& NU$& CHR$(179)& "V2"
230 LINPUT #1: M$ :: IF ASC(M$) <> 51 THEN PRINT #2 :
    C$& CHR$(182)& C$& CHR$(0) :: C$ = "" :: X=X+1
    ::GOTO 140
240 GOSUB 250 :: NU$=STR$(N) :: PRINT #2: C$& CHR$
    (179)& CHR$(200)& CHR$(LEN(NU$))& NU$& CHR$(179)&
    "V3"& CHR$(182)& CHR$(0) :: C$ = "" :: X=X+1 ::
    GOTO 140
250 T=ASC(SEG$(M$, 3, 1)) -48 :: P=POS(M$,",", 3) ::
    N$=SEG$(M$, 4, P-4) :: D$=SEG$(M$, P+1, 255)
```



```
260 IF N$="A" THEN N=110
270 IF N$="A#" OR N$="B@" THEN N=117
280 IF N$="B" THEN N=123
290 IF N$="C" THEN 131
300 IF N$="C#" OR N$="D@" THEN N=139
310 IF N$="D" THEN N=147
320 IF N$="D#" OR N$="E@" THEN N=156
330 IF N$="E" THEN N=165
340 IF N$="F" THEN N=175
350 IF N$="F#" OR N$="G@" THEN N=185
360 IF N$="G" THEN N=196
370 IF N$="G#" OR N$="A@" THEN N=208
380 IF N$="A" THEN N=220
390 FOR J=1 TO T :: N=N+N :: NEXT J
400 RETURN
410 IF D$="S" THEN D=1
420 IF D$="E" THEN D=2
430 IF D$=".E" THEN D=3
440 IF D$="Q" THN D=4
450 IF D$=".Q" THEN D=6
460 IF D$="..Q" THEN D=7 470 IF D$="H" THEN D=8
480 IF D$=".H" THEN D=12
490 IF D$="..H" THEN D=14
500 IF D$="W" THEN D=16
510 IF D$=".W" THEN D=24
520 RETURN
                                        END
```

TI WRITER TO TI BASE

---------- by Jerry Keisler

from the pages of the LA TOPICS newsletter, August 1993

I wrote an article on converting TI Writer to TI Base files, which appeared in Micropendium, when TI Base was in version 1. I am still recieving phone calls about this conversion process. TI Base is now in version 3.02 and can easlily do the conversion. Version 2.0 has the command ".CONVERT FROMFILE TOFILE GO", which is found on page 3-27 of the version 2 manual. This could have the form of ".CONVERT DSK1.fienam DSK2.filename GO". If you only have one drive, leave "GO" off and TI Base will prompt you for the correct disk.

---------- TI WRITER FILE ----------

The TI Wrtier file can take seveal forms, but it can not exceed 80 columns. Print the file to printer and identify the fields you want to convert to TI Base.

One of my TI Writer files called BUDGET, has a date field, money field, initial field, and description field with sizes of 8, 9, 3, and 60 characters.

```
DATE      MONEY INI DESCRIPTION
--------  ------ -- -----------
01/06/93   45.39 FP ----
02/23/93  184.33 FP ----
03/04/93  -23.45 JP ----
```

This file is a list of operating expenses and who paid

what.

---------- TI BASE CONVERSION ---------- I know the field names and thier size so to convert to TI Base, load TI Base and enter the command:

".CONVERT DSK1.BUDGET DSK2.BUDGET2 GO"

TI Base wants to know the record information and shows the structure input screen: I completed the structure input screen as folows:

NAME	TYPE	SIZE	DEC
DATE	D	8	
MONEY	N	9	2
INI	C	3	
DESC	C	60	

Press Function 8 and TI Base will read the BUDGET file and crate a BUDGET2/F file using the structure information provided. Each record in BUDGET2/D contains 80 characters.

".USE DSK2.BUDGET2" will open the new TI Base file.

".RECOVER" will build the sturcture file, BUDGET2/S. The TI Writer file is now in TI Base format.

---------- RULES OF CONVERSION ---------- 1) You can not change the order of the input file (TI Writer file).

2) You can convert less than the full recored. All conversions start from the first character in a record (or a TI Writer line). The end of a record or TI Writer line will be truncated. If only DATE, MONEY and INI in the structure then DESC would be truncated. The TI Base records would be only 20 characters wide.

3) You can create a TI Base fie that is larger than the TI Writer file. The extra fields will be blank. If DATE, MONEY, INI, DESC, BUDITEM and ACCOUNT are used, in the structure with sizes of 8, 9, 3, 60, 30 and 40, the new TI Base sturcture will have 150 characters.

This allows you to manually fill in extra data. If you use size 8, 9, 3, 30 and 40, you would have part of DESC in the BUDITEM and ACCOUNT fields.

4) Conversion is not limited to TI Writer files. Any ordered file can be converted to a TI Base fie.

---------- HOW TO SUM ---------- Using TI Base version 2 and the above TI Writer file, which contains 100 to 300 records, I can SUM all of the MONEY field. I also do a conditional sum on the MONEY field using the INI field.

This is done as follows:

Load TI Base.

".CONVERT DSK1.BUDGET DSK2.BUD GO"

NAME	TYPE	SIZE	DEC
DATE	D	8	
MONEY	N	9	2
INI	C	3	

Press Function and 8.

".USE DSK2.BUD"

".RECOVER"

".SUM MONEY" or

".SUM MONEY ;FOR (INI = " FP")" and

".SUM MONEY ;FOR (INI = " JP')"

The total amount of MONEY or the amounts of MONEY for the conditions shown will be diaplayed to screen. This saves a lot of time on the calculator.

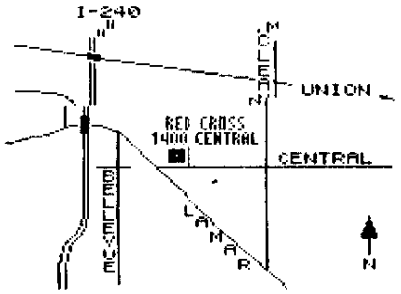
JUST A NOTE

```
                *
               * *
              * * *
             MS99UG
            SENDS TO
           YOU AND YO
          URS A MERRY
         CHRISTMAS AND
        A HAPPY NEW YEAR.
       MAY YOU BE RICHLY
     BLESSED WITH HEALTH
   AND HAPPINESS IN THE YE
  ARS AHEAD.BE EVER THOUGHT
 FULL TO YOUR FELLOW MAN
IN    WORD     AND     DEED.
            JOY
           PEACE
          AND HIS
          LOVE BE
          WITH US
          ALWAYS.
```

```
/------------------------------\
|                              |
|  CHRISTMAS GATHERING - PARTY |
|                              |
|  DECEMBER   MEETING  : 7:00  |
|                              |
|  BRING SOME FOOD  -  ENJOY   |
|                              |
\------------------------------/
```

*<|:-) *<|:-) *<|:-)

. *<|:-) *<|:-)



LOCATION MAP

WORKSHOP : to be announced.

PROGRAM BIT - third Thursday

DEC 16th , 1993

MEETING: 7:00pm - Red Cross Building - 1400 Central.

7:00pm - Doors Open - Christmas Party

8:00pm - Demonstration of programs obtained at Chicago TI Faire.

11:00pm - Meeting ends.

NOTICE

Information contained in TIdbits is accurate and true to the best of our knowledge. Viewpoints and opinions expressed in TIdbits are not necessarily that of the Mid-South 99'ers. We welcome any opinions/corrections from our readers. Articles may be reprinted elsewhere as long as credit is given to the author and newsletter.

GROUP INFO

Visitors and potential members may receive 2 free issues of TIdbits while they decide if they wish to join (no obligation) On the top of your label is a code. A Y means you are a member, N means 2 free list, UG means user group and S means a business. Beside the Y is a date, one year from that date your dues are due. A dollar sign ($) on the label will indicate that your dues are due. The library is open only to members. Library list is $1. Mail order disk library access is $2 for the first disk and $1 for each additional disk - max of 5 disks per month. Order by disk number only. At meetings, library access is FREE if you exchange your disk for ours or $1 per disk for our disks. Send all mail order library requests to librarian's address! Send dues and correspondence to group address.

CALENDAR

MEETINGS: DEC. 16, (3rd Thursday!)
WORKSHOPS: TO BE ANNOUNCED

24HR TI BULLETIN BOARD

The 9640 NEWS BBS 300/1200/2400/4800/7200/9600/12000/14400
Hayes. 901-360-0112

GROUP MAILING ADDRESS

Mid-South 99 Users Group
P.O. Box 38522
Germantown, Tn. 38183-0522

LIBRARY ADDRESS

Jim Saemenes
46 Higgins Road
Brighton, Tn., 38011

MEMBERSHIP APPLICATION

NAME _____ | | $18.00 FAMILY
ADDRESS _____
CITY _____ ST ZIP
PHONE(___) - :INTERESTS _____
_____
EQUIPMENT,ETC. _____
_____
Detach and mail with check payable to: Mid-South 99 Users Group, P.O. Box 38522, Germantown, Tn, 38183-0522.