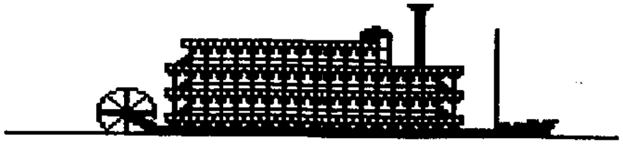
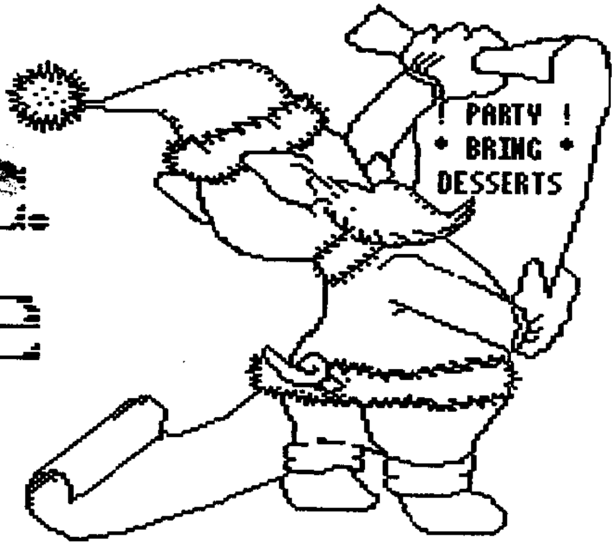
TIDBITS

MID SOUTH 99 USERS GROUP



MEMPHIS TENNESSEE

DEC. 1992



Mid-South 99 Users Group
P. O. Box 38522
Germantown, Tn. 3818

FIRST CLASS

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

Newsletter of the MID-SOUTH 99 USERS GROUP-Vol 10,#12-DEC 1992

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. 1992 INDEX

PRESIDENT'S BIT

--- by Gary W. Cox

Nominations were taken last month for the offices of President, Vice- President, Secretary and Treasurer and the results are as

President - Gary Cox

Vice-President - Richard Hiller

Secretary - Richard Mann

Treasurer - Mac Swope

You may also write in a vote for the above offices. The rest of the officers are appointed by the President.

At this months meeting is our usual big CHRISTMAS PARTY where we will be having a FREE BBQ dinner for everyone plus a demonstration or two. Everyone please try to attend as this is the meeting that really determines how many people are still interested in supporting their group. The desert will be up to you so please bring some sort of desert to the meeting such as cookies, cake etc... So the BBQ is on us and the desert is on you!

C ya at this months meeting and happy holidays...

THE ALTMAN FAIRWARE LIST

from the SouthWest Tucson 99 newsletter, Sep., 1992

The 1992 release of the Altman Fairware List is now available on disk or pre-printed. The new list contains over 4ØØ fairware titles and several indexes. The pre-printed copy contains 35 pages. The list is available from the SouthWest Ninety-Niners for $2 in either form.

Please send your request to:
SouthWest Ninety-Niners
P. O. Box 17831
Tucson, Az. 85711

Announcements of new fairware may be sent to the above address. Please include a short description of the program(s) and the author's address. The information provided will be added to the Altman Fairware List and made available to other TI / 964Ø users.

TI FILES

------------------------------------ by Earl Raguse

from the Portland PUNN newsletter, August 1992

(This article was written by Earl Raguse of the Orange County TI User Group. It was edited for Word Play by Charles Bull.)

It does not seem possible to do any serious programming without access to files, so I have decided that they are important enough to spend an article talking about them. The TI manual is OK, but you must do a lot of digging, experimenting and re-reading if you are going to get command of the situation.

To begin with there are a large number of types of files. Files are either SEQUENTIAL or RELATIVE. The difference is that the records of sequential files must be written and read sequentially, but records in relative files can be written and read in random order.

TI has decreed that all RELATIVE files must have a FIXED record length. SEQUENTIAL files may have FIXED or VARIABLE record length. With FIXED length records, all records must be of the same length, if they are not, they will be padded with spaces.

This all makes sense, when you think about how the disk controller would have to go about finding a random record, if all records were not the same relative length. For instance if you asked to see record #31 in a FIXED RELATIVE file of record length 2Ø, the controller knows the record #31 is 31x2Ø bytes into the files. If the records were variable in length that would not be feasible.

It is possible for the records to carry an identifying marker of some sort, and the controller could search for the marker sequentially in a linear search. I have previously talked about how poor linear searches are. It is possible to find a specific rcord in a SEQUENTIAL file by reading and tossing the records untill one gets to the record one wants. This is in fact a linear search.

Ok, so now we have SEQUENTIAL or RELATIVE, and FIXED or VARIABLE files. Each of those can be in INTERNAL or DISPLAY format. Essentially the difference is that INTERNAL is in binary, and DISPLAY is in ASCII format. So now we already have 8 possible file structures. To top this all off, each can be opened for file operations as INPUT or OUTPUT, and the latter can have either UPDATE or APPEND modes. UPDATE overwrites, ad APPEND just adds to the existing file. I will have a program to illustrate that later.

So far we have 1Ø possible file parameters that may be used up to four at a time. That is a lot of different types of files, even if we don't count differing record length. What's a poor programmer to do, go crazy? Well, not exactly, if taken one thing at a time its not all that bad.

There are lots of subtleties, in deciding the most efficient file type for any given set of data and its intended use. Generally, unless a FIXED length record is required for random access, I like DISPLAY VARIABLE 8Ø files so I can read them with TIW/FW. I have found that they take up no more space than any other type of file.

I have an unfinished program called CONVERT, to convert any type file to any other type file. Clearly, one cannot easily convert a FIXED 255 file to a VARIABLE 5 file, but I know of no reason to attempt it, unless one discovered that the FIXED 255 file was made up of 5 character records. Clearly, in that instance, the wrong file type was chosen, and conversion could reduce the file size by a factor of 51.

The major difficulty that I encountered with the program is in specifying the OPEN statement. I had intended that the user could select the parameters from a menu, and the program could then build up a string of the file parameters to use in the OPEN statement. Well the problem is the file name and device name are strings, but the parameters are neither strings n or numbers (ie. neither fish nor fowl), they must be actually typed into the statement. For example:

```
150 OPEN #3:"DSK" & DN$ & "." & FILENAME$ APPEND,
INTERNAL, FIXED LENG
```

This is the way one has to do it. Note that DN$ is the drive number string. It is not possible to have the parameters above loaded into strings, and replace the above with the equivalent strings. Note that it is possible to use a variable LENG for the record length. My initial attempts to do so didn't work, and I don't know why, but it does work.

TI says that if OUTPUT parameters are not specified, the default will be DISPLAY VARIABLE 8Ø, UPDATE. That is fine, just as long as a file does not already exist. If the file exists, you MUST open it EXACTLY as it was opened originally, or you will get an error message.

How did I manage that in CONVERT? Well, I used menu selection, as originally intended, but instead of building up a string for the open statement, I wrote all possible different open statements, and used logic based on the user response to the menu to select the correct statement. When at first I didn't seem to be allowed to use a variable for length, I made all files of length 8Ø, and in the instructions, told the user how to modify the program statements if they did not want length 8Ø. Now, I find that a variable does work. I don't know what I did wrong in the first place.

Considering all the different file formats that are used by various programmers, you are rather forced to conclude that the choice is VERY important. I have not found that to be so. INTERNAL files are supposed to be faster than DISPLAY files, if you are not printing the data to the screen or printer. Logicaly, this would seem to be so, I have never made a definitive test, but, I bet it doesn't make much actual difference.

Making a simple file conversion is easy. You recall the little program I published a couple of months ago to convert a DV/8Ø file to DF/16. All it amounts to is opening an OUTPUT file, in order to read it. You can't for instance read a string into a numerical variable, and visa versa.

File reading is usually done with a loop with a test for some condition to end. It is usually wise to test EOF before trying to read else you will surely get an error sooner or later. If you test the status of the condition EOF(7) and it is true, you must branch to a CLOSE statement, and exit the reading loop.

You have already seen an exanple of an OPEN statement for the OUTPUT file. The writing of it is as simple as PRINTN #3: A$(X), or whatever.

TI-Bits Number 16

by Jim Swedlow

from the TI-Sydney-Home Computer Group, June, 1992

[This article originally appeared in the User Group of Orange Coutny, California ROM]

HOW YOUR TI SAVES TEXT FILES

In our TI world, most text is saved in Display Variable 8Ø (DV8Ø) files. This is what the TI Writer Editor uses. What is DV8Ø?

Display means that the file is saved using ASCII characters. If you use a disk sector editor to look at a DV8Ø file, you will see that the text looks just about the same as it was written. Internal files are not always as easy to read (but that is another column).

A file is made up of records. In a DV8Ø file, ech record contains one line of text as it apperas in the Editor, Variable means that each record is only as long as the text line.

Consider these two lines of text:

TI
99/4A

When this is saved on disk, there will be two records in the file. The first recird is "TI" and the second is "99/4A". each record is preceded by the number of characters in the record in hex. Hex FF is used to mark the end of the file. The file would look like this:

Hex Ø2 54 49 Ø5 39 39 2F 34 41 FF
ASC T I 9 9 / 4

In a Sisplay Fixed 8Ø (DF8Ø) file, each record still contains one text line but is exactly 8Ø characters long. Your 4A pads each record by adding the required number of spaces to the end of each text line.

WHY YOU NEED TO KNOW

HOW THE REST OF THE WORLD SAVES TEXT FILES

As a loyal TI user you may not think that you need to know how others (MS-DOS, CPM, etc) save text files. If you do any work with modems, however, you do.

The reason is that you may down load a text file and find that it is Display Fixed 127 (DF128). Why? There is a standard protocol in the TI world for transferring files using XMODEM. It was designed by Paul Charlton (creator of Fast Term). The first record is NOT the first line of text. Instead, it is the disk header sector (which describes the file in a manner than can be read by the disk controller).

If the first record os not the header, however, your modem program (Telco, Fast Term, Mass Transfer, etc) assumes that you are talking to a non TI system and will save the file as DF128.

The reason, then that yu need to know how other systems save text files is that you may get one.

HOW THE REST OF THE WORLD SAVES TEXT FILES

The short answer is DF128. But there is more. Unlike a DF8Ø file, there is no padding. Instead, all of the text lines are run together. The end of each text line is marked with a carriage return <CR or CHR$(13)> and a line feed <LF or CHR$(1Ø)>.

One record may have one, two or more text lines, each ending with a CR and LF. If there is not enough room left in a record to bring it to 128 characters. Ther rest of the text lines starts the next record - followed by a CR and LF. The end of the file is marked with CHR$(26), which in the IBM and CPM worlds is <CTRL Z>. Remember our sample text?

TI
99/4A

Since it is well under 128 characters, the file will only contain one record:

Hex 54 49 ØD ØA 39 39 2F 34 41 ØD ØA 1A
Asc T I 9 9 / 4 A

Hex ØD ØA is a CR and LF. Hex 1A is the end of file marker CHR$(26).

CONVERTING FILES

There are a number of programs that convert files from DF128 to DV8Ø or from DV8Ø to DF128. Some of the assembly ones are quite fast. There should be a program in this issue called CONVERT. It does those conversions and twopo others.

A little background. Sometimes you may look at a file and notice that there are no CR's. If you reformat such a document, everything will be jumbled into one big paragraph. TI Writer stops reformatting when it hits a CR. FUNNELWEB stops when it hits a CR or a blank line. Either way, the document is a mess.

CONVERT, when converting a DF128 file to DV8Ø, can add a CR to blank lines, to the end of paragraphs and to lines that start with a period (Fomatter commands). This takes a little longer but it makes the file much easier to edit. Also, CONVERT can add CR's to DV8Ø files that lack them.

NOTE TO OTHER USER GROUPS

I have now written 2Ø XB columns and 2Ø in the TI BITS series. From time to time, other user groups have published some of my work in their news letters. If anyone wants a complete set, please send me two (2) DSSD disks or four (4) SSSD disks, a return mailer and return postage. My address is 73Ø1 Kirby Way, Stanton, Ca 9Ø68Ø.

Enjoy.

```
     1ØØ ! CONVERT 11Ø ! VERSION 1.Ø
     12Ø ! Ø9 Aug 88
     13Ø ! By Jim Swedlow
     14Ø ! Based on XPREP by Carl Walters
     15Ø !
     16Ø DISPLAY ERASE ALL :: CALL SCREEN(5) :: FOR  A=Ø  TO  14
::CALL COLOR(A,16,1) :: NEXT A
     17Ø FOR A=1 TO 4 ::READ T$(A) :: NEXT A
     18Ø  N$= CHR$(13) & CHR$(1Ø) :: Z$=CHR$(26) :: C$= CHR$(13)
:: GOTO 3ØØ
     19Ø    DATA    DF128->DV8Ø    add    CR's,DF128->DV8Ø    no
CR's,DV8Ø->DV8Ø add CR's,DV8Ø->DF128
     2ØØ CALL KEY :: Q$,S,P,K,I$,W$ :: !@P-
     21Ø ! 22Ø ! STRING CHECK SUB
     23Ø !
     24Ø  P=1 :: IF I$=" " OR I$="" THEN I$="" :: RETURN ELSE IF
ASC(I$)=46 THEN RETURN ELSE P=Ø :: RETURN
     25Ø !
     26Ø ! CLOSE FILES AND END
     27Ø !
     28Ø CLOSE #1 :: CLOSE #2 :: DISPLAY AT(19,1)BEEP: "DONE"
     29Ø FOR P=1 TO 1ØØ :: NEXT P
     3ØØ !
     31Ø ! TITLE SCREEN
     32Ø ! 33Ø DISPLAY AT(5,5):"CONVERT Version 1.Ø": : : : :  :
:"Press For"
     34Ø  FOR S=1 TO 4 :: DISPLAY AT(14+S,1):STR$(S); " "; T$(S)
:: NEXT S :: DISPLAY AT(19,1)BEEP: "5 End Program"
     35Ø !
     36Ø ! PICK FUNCTION
     37Ø !
     38Ø CALL KEY(Ø,K,S) :: IF K<49 OR K>53 THEN 38Ø ELSE K=K-48
:: IF K=5 THEN DISPLAY ERASE ALL :: STOP
     39Ø  DISPLAY AT(13,1):"T$(K): :"Input File: DSK": :"Out put
File: DSK": : : :
     4ØØ ACCEPT AT(15,18)BEEP:I$
     41Ø ACCEPT AT(17,18)BEEP:W$
     42Ø !
     43Ø ! OPEN FILES & INIT
     44Ø !
     45Ø DISPLAY AT(19,1):"Working .   .   .   ."
     46Ø IF K>2  THEN  OPEN  #1:"DSK"  &  I$,  INPUT  ELSE  OPEN
```

```
#1:"DSK" & I$, INPUT, FIXED 128
     47Ø  IF K=4 THEN OPEN #2:"DSK" & W$, OUTPUT, FIXED 128 ELSE
OPEN #2:"DSK" & W$, OUTPUT
     48Ø A=1 :: W$="" :: ON K GOTO 72Ø,57Ø,49Ø,65Ø
     49Ø !
     5ØØ ! DV8Ø -> DV8Ø ADD CR's
     51Ø !
     52Ø LINPUT #1:I$ :: GOSUB 21Ø :: IF EOF(1) THEN 55Ø
     53Ø IF A THEN IF P THEN PRINT #2:I$; C$ ::  GOTO  52Ø  ELSE
Q$=I$ :: A=Ø :: GOTO 52Ø
     54Ø  IF  P  THEN PRINT #2:Q$; C$; I$; C$ :: A=1 :: GOTO 52Ø
ELSE PRINT #2:Q$ :: Q$=I$ :: GOTO 52Ø
     55Ø OF A=Ø THEN IF P THEN PRINT #2:Q$; C$ ELSE PRINT #2:Q$
     56Ø PRINT #2:I$;C$;C$ :: GOTO 25Ø 57Ø !
     58Ø ! DF128 -> DV8Ø NO CR's
     59Ø !
     6ØØ LINPUT #1:I$ :: W$=W$ & I$ :: K=1 :: S=LEN(W$)
     61Ø IF SEG$(W$,K,1)=Z$ THEN 25Ø ELSE IF K>S THEN IF  EOF(1)
THEN 25Ø ELSE W$="" :: GOTO 6ØØ
     62Ø  P=POS(W$,N$,K) :: IF P THEN PRINT #2:SEG$(W$,K,P-K) ::
K=P+2 :: GOTO 61Ø
     63Ø P=POS(W$,Z$,K) :: IF P THEN PRINT #2:SEG$(W$,K,P-K)  ::
GOTO 25Ø
     64Ø W$=SEG$(W$,K,255) :: IF EOF(1) THEN PRINT #2:W$ :: GOTO
25Ø ELSE 6ØØ
     65Ø !
     66Ø  !  DV8Ø  ->  DF128  67Ø  !  68Ø  LINPUT  #1:I$  ::  IF
ASC(I$)=128 THEN I$=" "
     69Ø W$=W$ & I$ & N$ :: P=LEN(W$)
     7ØØ   IF  P>128  THEN  PRINT  #2:SEG(W$,1,128)  ::  W$=SEG$
(W$,129,255)
     71Ø IF EOF(1) THEN PRINT #2:W$ & Z$ :: GOTO 25Ø ELSE 68Ø
     72Ø !
     73Ø ! DF128 -> DV8Ø ADD CR's
     74Ø !
     75Ø LINPUT #1:I$ :: W$=W$ & I$ :: K=1 :: S=LEN(W$)
     76Ø IF SEG$(W$,K,1)=Z$ THEN 82Ø ELSE IF K>S THEN IF  EOF(1)
THEN 82Ø ELSE W$="" :: GOTO 75Ø
     77Ø  P=POS(W$,N$,K) :: IF P THEN I$=SEG$(W$,K,P-K) :: K=P+2
ELSE 8ØØ
     78Ø GOSUB 21Ø :: IF A THEN IF P THEN  PRINT  #2:I$;  C$  ::
GOTO 76Ø ELSE Q$=I$ :: A=Ø :: GOTO 76Ø
     79Ø  IF  P THEN PRINT #2:Q 76Ø ELSE PRINT #2:Q$ :: Q$=I$ ::
GOTO 76Ø
     8ØØ P=POS(W$,Z$,K) :: IF P THEN I$=SEG$(W$,K,P-K)  ::  GOTO
82Ø
     81Ø  W$=SEG$(W$,K,255) :: IF EOF(1) THEN I$=W$ ELSE 75Ø 82Ø
IF A=Ø THEN GOSUB 21Ø :: IF P THEN PRINT #2:Q$;  C$  ELSE  PRINT
#2:Q$
     83Ø PRINT #2:I$; C$; C$ :: GOTO 25Ø
```

DV80 TO PROGRAM

-- by Jim Peterson
from the pages of the Spirit of 99 newsletter, Sep., 1992

John "Jeb" Hamilton of the Central Iowa User Group was the first to realize, several years ago, that a DV8Ø listing of a Basic or XBasic program could be converted to a DV163 file and then merged in and run as a program. I no longer have his program in my library, but this is a quick and dirty version of it -

```
100 DISPLAY AT(12,1)ERASE ALL:"Input file? DSK":"":"Output
    file? DSK"
110 ACCEPT AT(12,16):A$ :: ACCEPT AT(14,17):B$
120 OPEN #1:"DSK" & A$, INPUT :: OPEN #2: "DSK" & B$,
    VARIABLE 163, OUTPUT :: LINPUT #1:M$
130 LINPUT #1:M$ :: IF LEN(M$)>78 AND EOF(1) <> 1 THEN
    LINPUT #1:M2$ :: M$=M$ & M2$
140 X=POS(M$," ",1) :: Y=VAL(SEG$(M$,1,X-1))
150 PRINT #2: CHR$(INT(Y/256)) & CHR$(Y-256* INT(Y/256))&
    "!" & SEG$(M$,X+1,255) & CHR$(0)
160 IF EOF(1) <> 1 THEN 130 ELSE CLOSE #1 :: PRINT #2:
    CHR$(255) & CHR$(255) :: CLOSE #2
```

To try that out, key in this useless little program -

```
100 CALL CLEAR
110 FOR J=1 TO 10
120 PRINT J
130 NEXT J
140 END
```

List that to disk by LIST "DSK1.8Ø". Then run the above converter program, answer the input prompt with 1.8Ø and the output prompt with 1.163. After it runs, enter NEW, then MERGE DSK1.163 and then LIST. This is what you should see -

```
100 !CALL CLEAR
110 !FOR J=1 TO 10
120 !PRINT J
130 !NEXT J
140 !END
```

Type 1ØØ and FCTN X to bring line 1ØØ to the screen with the cursor on the "!". Type FCTN 1 to delete the "!" and repeat with FCTN X and FCTN 1 to delete all the others. Then enter RUN and it should do so!

All that the program does is delete the blank first line of the listing, convert each program line number to tokenized format, add a CHR$(Ø) end-of-line marker to each line, move the record to a DV163 file, and add the double CHR$(255) end-of-file marker.

The result is a merge format program composed of REM statements; when you delete the "!" REM indicator, these become progam lines.

There is just one problem. A LISTed program is a DV8Ø file, consisting of records of 8Ø characters or less, but a program line in XBasic can be keyed in up to 14Ø characters long, and can be forced even longer (as I often do!). When such a line is LISTED, it is broken into 8Ø-character records, which confuses the conversion program completely.

Line 13Ø of the conversion program attempts to resolve that problem. If a record is more than 78 characters long (because it could have been an 8Ø-character line ending in a blank, which would become a 79-character record without the blank) it is taken to be most probably the first part of a long program line; another record is read in and tacked onto it.

However, this creates another problem, as you will find out if you LIST the converter program and then try to convert it back to a program - line 14Ø will be tacked onto line 13Ø because line 13Ø is 79 characters long.

The best fix for this is to load the DV8Ø file into Funelweb and print out a hard copy; use a ruler to draw a vertical line after the 78th characters; mark any program line that ends on the 79th or 8Øth characters, delete these characters, save the listing, run it through the converter, merge it in and key those deleted characters back in - still much easier than keying in an entire listing.

After John Hamiton published his discovery, several authors wrote their own version. It was suggested that programs could be written in text format, using the superior editing features of TI-Writer or Editor Assembler, and then converted to program format. Personally I was satisfied with the editing features of Basic and was not about to give up its syntax error-catching capability, so I never tried this method.

However, nowadays several hundred text files of newsletter articles are available on the Clearing House BBS and other newsletters are being circulated on disk. Many of these articles contain program listings, and it would be much easier to extract and convert them than to print them out and key them in.

Later on John Ford wrote a more complex converter called XLATE, which eliminates the need to delete all the "!" by converting the ASCII text file directly to tokenized merge format. It also checks for syntax errors and corrects them or reports them on-screen. If the LISTed program had regularly sequenced line numbers, it will check these to see whether records should be combined, which should greatly improve accuracy - I have not tested it enough to say how foolproof it is.

Blanks at the end of a DV record are dropped, so if the 8Øth character of a long program line is a blank, then the line is broken into two records and then recombined the blank will be missing. For instance, if the blank between FOR and J in FOR

J=1 TO 1Ø happend to be the 8Øth character, it will recombine as FORJ=1 TO 1Ø. This results in a SYNTAX ERROR referencing the line number, which is therefore easy to spot and correct. The same problem can cause the string "John Doe" to become "JohnDoe".

The above conversion programs are intended for listings in 8Ø-column format. However, many of the listings within text articles have been reformatted to 28-column or 4Ø-column width, or listed in those widths with Triton Super Extended Basic.

Fortunately, there is an alternative. Curtis Allen Provance has written a truely remarkable program in assembly, called TEXTLOADER, which will convert a DV8Ø file directly into a program in memory, and will handle the shorter line lengths, although with increased chance of error because the method of detecting new lines is far from foolproof.

I have not tested this procam extensively, but have found only two major problems. The one is with records ending in a droped blank, as described above; these are easy to correct. The other is with split-referenced line numbers. For instance, if a line ends in GOSUB 12Ø :: GOTO 2ØØ and splitting of records turns this into GOSUB 1 and 2Ø GOTO 2ØØ, you will find the line ending with GOSUB 1 and a new line 2Ø :: GOTO 2ØØ at the beginning of the program. Comparison with the original listing makes this easy to correct.

TEXTLOADER loads into memory and remains there, so that you can load other text files by simply typing -
CALL LINK("OLD"<"DSKn.filename"). The file loads and converts rapidly, displaying each line as it does so. Sometimes a line will be reported as a syntax error and omitted, but sometimes it will be omitted without being reported, and sometimes it will not be detected untill you try to run the program. Ocassionally, especially when working with 28-character lines, you will get all sorts of invalid error messages. Apparently the program in memory differs from the screen display, and it may be implssible to debug in such cases.

Other features allow you to merge a converted text file into a program in memory rather than overwriting it, and to read and run a batch file of command type instructions, such as -

```
CALL FILES(1)
NEW
RUN "DSKn.biqprogram"
```

An improved general -purpose menory image loader program is also included.

XLATE is a public domain program, available on my TI-PD disk # 1Ø83. TEXT-LOADER is a fairware program available in my TI-PD disk # 11Ø4.

CHRISTMAS GREETINGS WITH A T. I. FLAIR

by Wayne Garrison (MDCRCC-STL99ers)

from the St. Louis BRIDGE newsletter, Dec., 1991

For those of us who work everyday at the keyboard of a computer terminal or typewriter then go home and enjoy a few peaceful hours with out TI-99/4A, or whatever our daily task may be, it isn't hard to get the feeling of Christmas Love sitting at the console of this little machine. You see, for some of us, this little orphaned machine is a lot more than just a computer. Sit in front of it for a few minutes when the house is still and you are alone. Let your mind wander back a few years ago when you first got this little machine. Remember the joy and excitement that filled your heart when you first took it from the box and set it up.

I see many things when I sit in front of mine. I get visions of my wife, God bless her soul, running around from store to store weary from putting in a long day at the office trying to find out all the facts she can about the computers that were on the market at that time. She researched them all to see which one of these "Home Computers" would be the best. Software quality, expandability, capability, features such as it's 16 bit CPU, where it was made, these and other facts were on her list of items in her research. Then came the moment of excitement on Christmas when she watched the expression on my face as I opened the gayly wrapped package. Now I see myself expanding my T.I. with all the peripherals and writing cute little Christmas programs for my son, who was very small then. I recall the gleam in his eyes when he saw Snoopy appear as if by magic on the monitor screen wearing a stocking cap. He thought it was amazing that it could play Jinge Bells and all his favorite Christmas songs. My mind drifts forward and I see the tiny little fingers of small hands pressing the computer keys as he used a software module which is teaching him his nimbers, colors, how to read, and how to spell. I'll never forget the puzzled look in his face when he heard it speak his name. He had never imagined that a machine could talk. A few years have gone by in my thoughts. My son is now in elementary school. He has benefitted immensely from this little computer and is now writing programs of his own. Once again I remember the family all gathered around my computer desk amazed at how it could draw Christmas pictures and play Christmas songs. Now I've returned to the present. My son will soon be moving up to high school on the honor role and will again turn to his TI-99/4A to do term papers and budget his earnings so he can buy his first car. As for myself, I've just finished running off our Christmas card mailng list to review it for any possible address changes. Tomorrow night I'll run off the mailing labels and return address labels. This is the only machine in my life that gives me these warm feelings. The others are just machines. Althcugh this is not the first computer I ever used and certainly not the last, it is the first one to earn a place in my home for it's many uses. I'm sure many of you have similar thoughts. If so, take a moment to reminisce. If not, use your TI; with the family. It has a way of growing on you.

On a lighter note, I was working on an article the other day and I had the stereo on playing the Mannheim Steamroller Chritmas tapes, and if you aren't familiar with them, they have produced some ot the most beautiful Christmas music you would ever want to hear. If you've heard them, you know what I mean. Anyway, as these songs were filling the house with Christmas cheer and I was beginning to get into the spirit, I started putting words with the music I was hearing. Well what I came up with wasn't to be found in any songbook. I chose to jot them down and here they are;

The TI/994A Christmas Songbook

Deck the Halls

Deck the files with Christmas greetings,
tab, tab, tab, tab, tab, tab-tab, tab, tab.
Keyboard takes an awful beating,
tab, tab, tab, tab, tab, tab-tab, tab, tab.

Jam we now our matrix printer,
tab, tab, tab, tab, tab, tab-tab, tab, tab.
Hope to make it through the winter,
tab, tab, tab, tab, tab, tab-tab, tab, tab.

New Disk Drive (White christmas)

I'm dreaming of a new disk drive
One that is double density.
I have bigger programs, and lots of filenames,
I need, more capacity.

I'm dreaming of a new disk drive,
Like all my friends have shown to me.
But for now I'll try to get by,
And I'll use more floppies woe is me.

Away with a T.I. (Away in a Manger)

Way back in a closet my T.I. was found.
Left there by some idiot till I came around.

I asked all my neighbors to find me T.I.s'.
One told me of one up the street, I should try.

The guy said he got it at the grocery store.
He got it to come on but couldn't do more.

I offered him 2Ø and took it and ran.
Now I use it daily how happy I am.

Silent T.I.

Dusty T.I.
Rusty am I
Not much time
Off the line

Big noisy fan, doesn't spin anymore.
Modules tucked 'way, in the drawer where they're stored.

Printer is silent and lifeless
Monitor has a blank stare.

God Rest Ye, Merry 99s

God rest ye, merry 99s,
Fret nothing you display.
For new software and good support,
Is always on it's way.

To keep you out of dirty closets,
Cursorless to lay.

O-oh tidings of upgrades to come.
Upgrades to come,
O-oh tidings of upgrades to come.

We Wish You a Merry Backup

We wish you a merry backup,
We wish you a merry backup,
We wish you a merry backup,
And a happy filename.

Good sectors to you,
Whatever you do.
Good sectors for backup
And a happy filename.

Enter Here (Jingle Bells)

Dashing through spreadsheets
With good old Multiplan.
Deadline's comiung soon
Tired though I am.

Fields are all filled in
Calculations right.
What fun it is to run this stuff
In the middle of the night.

Enter here, enter there,
Checkout the results.
The boss is angry it ain't done,
He's full of bad insults.

Enter here, enter there
No time for delay.

I've got to have this finished
To be off on Christmas Day.

Well I hope these songs have lifted you spirits a little. They were fun to put together and are surely the right thing for the person that walks and talks computers. I want to take this opportunity to send from our home to yours, wishes for a very merry Christmas and a Joyous New Year!

Happy Computing!

* SEASON'S GEETINGS *

* MEILLEURS VOEX *

* GOD JUL OCH GOTT NYTT AR *

* MANUAI LE ASO MALOLO *

* TILLYKKE *

* ALOHA PUMEHANA *

* FELICIDADES *

* BOAS FESTAS *

* FROEH WEIHNACHTEN UND
EIN ERFOLGREICHES NUES HAHR*

* PRETTIGE KERSTADGEN EN
EEN VOORSPOEDIG NIEUJAAR *

* GESEENDE KERSFEES EN'N
VOORSPOEDIGE NEWEJAAR *

* TREVLIG HELG *

* BUONE FESTE *

* SEASON'S GEETINGS *

QUOTES FROM THE PAST

-------------------by Gene Bohot
from the pages of the Milwaukee newsletter, Feb, 1992

[Ed: Just one of the goodies on the disk sent me by Gene. I remember reading these quotes in the Pomona Newsletter, and thinking, "How in the world dod Gene get the time to do all this reading?" , , , , ,]

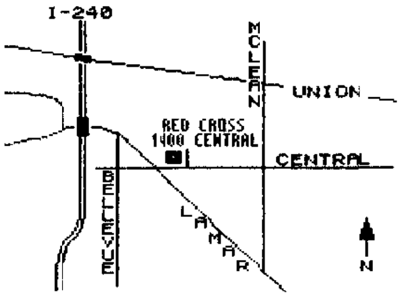
"While theoretically and technically television may be feasible, commercially and financially I consider it an impossibility, a development of which we need waste little time dreaming."
Lee De Forest, "Father of the Radio," 1926

"The evergy produced by the breaking down of the atom is a very poor kind of thing. Anyone who expects a source of power from the transformations of these atoms is talking moonshine."
Ernest Rutherford, physcist, 1930

"Landing and moving around the moon offers so many serious problems for human beings that it may take another 200 years to lick them."
Science Digest, Aug. 1948

"In my opinion such a thing is impossible for many years rocket shot from one continent to another carrying an atomic bomb and so directed a to be a precise weapon ... I think we can leave that out of our thinking."
Dr. Vannevar Bush, 1945

Prediction is very difficult, especially about the future."



LOCATION MAP

WORKSHOP : to be announced.

PROGRAM BIT - third Thursday

DECEMBER 17th , 1992

MEETING: 7:ØØpm - Red Cross Building - 14ØØ Central.

7:ØØpm - Doors Open

7:15pm - General announcements, elections.

7:3Øpm - BEQ Dinner and CHRISTMAS PARTY.

8:3Øpm - Special Demonstration.

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, M means 2 free list, UG means user group and $ 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 17 JAN. 21, (3rd Thursday!)
WORKSHOPS: TO BE ANNOUNCED

24HR TI BULLETIN BOARD

The 964Ø NEWS BBS 3ØØ/12ØØ/24ØØ/48ØØ/72ØØ/96ØØ/12ØØØ/144ØØ
Hayes. 9Ø1-368-Ø112

GROUP MAILING ADDRESS

Mid-South 99 Users Group
P.O. Box 38522
Germantown, Tn. 38183-Ø522

LIBRARY ADDRESS

Jim Saemenes
46 Higgins Road
Brighton, Tn., 38Ø11

MEMBERSHIP APPLICATION

NAME ________________________________ □ $18.ØØ 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-Ø522.