CLEVELAND AREA TI99-4A
USER GROUPS NEWSLETTER

OCTOBER 1988

We are doing something in the next few months I do not normally like to do and that is, dedicate the newsletter to one major topic, in this case TI-BASE. When this is done, you are always disappointed when the newsletter comes and if you are not interested in the spotlighted subject, you feel it a complete waste. However, TI-BASE is the database for which we have been begging for years. If you have not seen it, perhaps these tutorials will give you the info you need to get it, and if you have it, it will help you get started. Please bear with us. Because we feel this is such a major contribution to keeping the TI going, we have committed the space.

Glenn Bernasek has donated to the library a disk of programs that he has either written from scratch or others from magazines that he has enhanced. I have not had a chance to get it integrated in the library to tell you what disk # it will be, but hope you will look for it in the next update. Hopefully it will give you ideas on how to take other people's programs and customize them. Among these are Glenn's original "TI-SHORT SHEET" mini spreadsheet and we will have a review of it next month. Les Kee also gave me a program last month that he had revised from the old 99er magazine. Let's keep these coming!

Steve Bagsted gave me a disk with multiplan templates for tracking a stamp collection, net worth, high scores, etc. I would like to have enough to make up a full disk and if anyone else has any for bowling, golf handicaps, baseball, etc., please donate them. I am working on 1040 tax returns that you can print directly from Multiplan. Now that you can get Multiplan for less than $25, you should. If all those formulas scare you, that is why we are trying to build templates so that all you have to do is drop in the figures and the program will do the rest!

I must spotlight Paul Newmeyer's contribution of the FORTH language library. I saw the FORTH disks when they first came in and know what a mish mash they were. You had to load the FORTH language, then figure out which options

individual programs needed; some had errors in them; if everthing didn't go right, you locked up, had to reboot and start all over. Pretty soon you were ready to head for the FIFTH in the cupboard instead of FORTH in the PEB. Paul must have a good connection with the man upstairs because he has had the patience to take this mish mash and assemble what I believe to be the finest FORTH library anywhere in the TI world. You can take ANY of his LEGO FORTH disks and load them through option 3 of your E/A loader by typing "DSK1.FORTH". A menu will come up for you to select the program. After the program has completed, you will see instructions on the bottom of the screen to bring you back to the main menu. The only thing that might seem strange to a non-FORTH user, is an occasional message which says "type FORGET____ (whatever the Forth programmer has used) to save memory". FORTH is built on a stack and each time a new program is loaded, it simply stacks the executable words on top of those already there unless told to "FORGET" them. Thus, probably not, but conceivably, you can run out of memory if you run a lot of programs consecutively without "FORGETTING" some of the words. Otherwise, you can run these disks as easily as tho they were Basic, XBASIC, Assembly, whatever. Paul has disks of graphics, utilities, games, etc. Watch for these LEGO FORTH disks as Paul is continually adding more.

CONTENTS

EXECUTIVE NOTES - TI-CHIPS
Mary Phillips, Secretary

The September meeting brought thirty members and friends of TI-Chips to the North Royalton library. A full two hours was spent discussing business and enjoying many demonstrations.

Some disturbing news was related by Tom and Judy Thalner. They have been forced to temporarily (we hope) go out of business because of legal action brought against them by some of their neighbors. The Thalners have been of invaluable help to both Cleveland groups by their support and news of new products and services for the TI. Tom and Judy--please know that TI-Chips is behind you 200%, and we hope your situation can be happily resolved.

John Parken has been busy organizing and printing postcards for stores and ads for publications just to let people know we're alive. TI-Chips membership has been offered and accepted nationally for several months now. A more vigorous postcard mailing will take place within the next few months.

President Glenn Bernasek called for members to serve on a committee to research a 1989 Cleveland area conference for September, 1989. This committee will contact vendors and locate a place to hold the conference. If vendors are not available, then no conference will be held.

Glenn proudly reported he had rewritten his Short Sheet program in Extended Basic. It is now available nationally as Fairware. Keep up the great work, Glenn!!

Mark McCauley demonstrated two programs from the disk library. "Zodiac Wheel of Fortune", from disk NN6, provides astrological information when provided a user's birthdate. The program draws from an extensive database for the information. Mark's second program, "Graphic Editor", from disk XX4, prints designs which can be used for lables. The designs can be made from scratch and completely edited.

Russ Shimandle demonstrated some "just-for-fun" programs. They were "Duck Leader", written in Basic, and "Orbital Defender", a game originally found in Home Computer Magazine. Russ's last program was "Will Writer", which allows the user to compose a will (which should, of course, be checked for accuracy by a lawyer).

Les Kee showed how "Display At", and "Accept at", could be used in Extended Basic. He demonstrated a program which is used to keep newspaper collection records.

The monthly raffle was held. The winner, Russ Shimandle, chose ten disks loaded with new programs from the library.

TI-Chips had a call last month from a concerned mother, Mary Laforet. He daughter, Kimberly, has cerebral palsy and relies on her TI to do her school work. Anyone interested in helping this delightful young lady should call her mom at 734-7864. This is the North Olmsted area. Thanks!

Next month's meeting will include more demonstrations. Matt Andel will show graphics instances--dinosaurs. Les Kee will have more Extended Basic information. Ed Kennelly will show some utility programs. Hope you can be there.

ASK C.T.
By C.T. TIBS - Cleveland, Ohio

It seems as though the mysterious world of printer communication is still confounding some of us WITH GOOD REASON. I am still getting reports that many 99'ers are having difficulty getting their printers to respond at all, let-a-lone have it perform tabbing, line spacing, special fonts and the like.

Well, there's no place like the beginning to start. Let's run a printer SELF TEST by turning on the printer and pressing (and holding down) the <SPACE BAR> while tuning on the TI-99/4A. The printer should start printing out its character sets. If this doesn't happen, check to see if ALL plugs are tightly in place. If they are, then I would suspect an electrical or compatability problem with the TI/PRINTER interface (connection). See G.W. Bernasek's article "PARAPRINT PROBLEMS" as appeared in the March, 1988 newsletter.

If the SELF TEST does perform, then try OPENING the TI to the printer and print a simple message. This can be done by typing and <ENTERING> in the following:

```
OPEN #1:"PIO" or "AXION" or "PRINTER"
        or "RS232" or "RS232C" or "Whatever
        your printer is called" (Make sure the
        name is in QUOTES.) <ENTER>
```

(If you get an error message, look up the meaning in either your printer manual or the TI-99/4A User Manual, and try to correct what the message says is wrong.)

If there is no error message; then proceed...

```
PRINT #1:"HELLO PRINTER!" <ENTER>
```

The printer should print out, "HELLO PRINTER!" If not, make sure the message is in QUOTES. If the message is printed; then proceed...

```
CLOSE #1 <ENTER>
```

REMEMBER: YOU MUST "CLOSE" ALL OPENED CHANNELS TO PERIPHERAL DEVICES AFTER THE JOB IS COMPLETED, or the next time you want to use that channel or device, it won't respond! The "Keyboard" CLOSE or a programmed cLOSE is the BEST way to accomplish this.

Most of the time our printer problems can be easily solved with a DEEP BREATH, a little reading and by "Thinking it Through".

If you have any comments and/or questions, please send them to:

C.T. TIBS
13246 HARPER ROAD
STRONGSVILLE, OH 44136

We had another small meeting this month, about two dozen members showed up. The demo by Chuck Poulin was interesting, and the members must have found BOOT/V4 interesting also, 'cause they grabbed nine of the ten copies I had ready for the meeting, when the demo was over. There was a lot of activity after the meeting because the Thalners announced that they were going out of business, and were reducing prices on their stock. I can't say for sure, but my guess is that some of the members at this meeting found a bargain or two to take home. As for my current obsession TI-Base, 6 members have already picked up copies of the program and I believe there are still 6 names on the list of requestees. At this rate we should have some interesting comments and questions on TI-Base in a couple months. The TI now has the three major pieces of software that a computer must have to serve the needs of its users. Those items are, a wordprocessor, a spread sheet, and a database management system. Don't forget that the TI started out with some educational software and some silly little kids games, and those cartridges sold for forty, fifty, and sixty bucks each. "Talk about the bad old days, that was it." Here's one last item of interest. Bud Mills is coming out with a Gram Kracker type item probably called the P-Gram. It will have 72K of memory, reside on a board in the PE-Box and be software driven. The kit will cost approximately $150 and for an extra $20 Bud says he will soon have a clock kit to go on the same board. Bud has been a very important supplier of TI hardware for several years, and I hope he continues in the future.

THE DISK LIBRARY

If you have any problems with disks, or with programs on disks that you have received from the library, please let us know about it. We have about 4,000 programs in the library and we do not have the manpower to check the software for possible problems. If you encounter a problem, we need the disk name and the program name. This must be in written form. We'd like our members to enjoy the full benefit of this great library, so help us out with this problem.

NEW SPECIAL INTEREST GROUP

Bernard Zuckerman's new group is off the ground. Berny has temporarily named the group, "The Novice SIG". The group has some members and the systems those members own range from a console and tape deck, to complete systems with disk drives. That means belonging to this group could be helpful to you, no matter what your status is in the TI community. Berny lives in South Euclid and his phone number is 381-4088. If you're a new member or just need some help, give him a call and join the SIG.

THE NEXT NORTHCOAST MEETING

At the next meeting Paul Newmeyer will demonstrate PLUS. This is a TI-Writer or FunnelWeb enhancement disk. It allows you to produce elaborate documents with ease, because of its transliterate and control code implementation. Even if you have no interest in PLUS, the demo will probably have some great tips on how to use FunnelWeb, and I think Paul is going to bring his Grand Ram Card.

See you all at the next meeting. **Marty**

TI-BASE - INSCEBOT Inc.
P. O. Box 291610
Pt. Orange, FL 32029

IMPORTANT TIPS

NorthCoast 99'ers — Sept. 16, 1988
Late information By Martin A. Smoley

In Tutorial 2, I said that you can use FunnelWeb to enter Command Files and place comments in columns 41 through 80 which would not affect TIB. I must temporarily retract that statement. I am finding that for some reason characters on that side of the page cause TIB to issue the error message "no data base in use". This is not a constant problem, but seems to be affected by certain commands on the left side of the page. So, for now it is best to not type any characters from column 40 to 80 in Command Files. Also, concerning error messages, the message "no data base in use", seems to pop up for a multitude of errors. If you get this message and you don't think that's your problem, check for missing quotation marks, or for improper local variable statements. One reason I use FunnelWeb for CFs is so I can print out hard copies. I have a terrible time finding little mistakes like missing quotation marks on the screen. Here's something you need to remember. If you leave TALK ON you will see all the CF lines scroll up the screen. Keep an eye on the line numbers at the far-left. If you see an asterisk at the beginning of that number, it means that line was not executed. Remembering this could help you find the point where an error originated. Here's an unrelated tip. It seems that 2 is the smallest numeric variable you can create. An example would be "LOCAL X N 2 0". TIB apparently wants space available for a sign, such as "-1", or "-9". I'll admit that I don't know why many of these programming problems occur. I've only been working with TI-Base for two and a half months and I'm doing a lot of (learn as you go) programming. Sorting apparently needs some discussion. I have placed automatic sorts in many of the CFs. This is no problem when you have 5 names in the file, but when you have 100 names it takes a lot of time. With TIB a sort is saved. If you sort a file by zipcode the next time you use that file it will still be sorted by zipcode. This means you only have to resort when you add more names or edit the zipcode field. I'd like to grab this chance to ask for your help. **HELP!!!** I need questions from TI-Base users. If you have a problem with TIB, or something weird is going on with the program, write down some notes and send them to me, Martin A. Smoley, 6149 Bryson Drive, Mentor, Ohio, 44060. I cannot answer the letters individually, but this information will be used in future tutorials. I realize that the tutorials are wordy and complicated at this point, but when I cover almost everything in the manual the tutorials will switch to more programming and less retoric. You should also read the TIB manual and the TIB supplied tutor a couple of times. You should also create small command files with no specific purpose except to learn how something works. You need to use TIB to learn it. FYI: Here is something to think about. TI-Base is in many ways identical to one of those big database programs for the big machines, but scaled down to fit our small machine. But! you can buy a complete TI99/4A system, with disk drives, and TI-Base for less money than it would cost for the database software alone for an IBM compatable machine.

Good Luck. Marty.

TI—BASE - From INSCEBOT
TUTORIAL 2 By Martin Smoley
NorthCoast 99'ers — Sept. 1, 1988
Copyright 1988 By Martin A. Smoley

I am reserving the copyright on this material, but I will allow the copying of this material by anyone under the following conditions. (1) It must be copied in its entirety with no changes. (2) If it is retyped, credit must be given to myself and the NorthCoast 99ers, as above. (3) The last major condition is that there may not be any profit directly involved in the copying or transfer of this material. In other words, Clubs can use it in their newsletters and you can give a copy to your friend as long as its free.

First some of Marty's shorthand from last month. The letters TIB will refer to TI-Base. MT: will signify the beginning of some text which should be considered Marty's Theory. Marty's Theory should not be taken as fact, but as my interpretation of an item. FYI: designates text that is For Your Information. FE will stand for For Example. DP will stand for Dot Prompt. <E> means press ENTER. (FEL) means Further Explanation Later, and last for now is "›", the greater than sign. I will use "›" when program segments are displayed at the left of every line. The position immediately to the right of the "›" will be column one. Take the example ›12345. You should think of the number 1 as column one. The › does not exist. It is for reference only, the same as when you type in an XBasic program, at the head of each line you see › but it is not part of the program. **ALSO!** In Tutorial 2, I have listed some Command Files with line numbers instead of "›" in the left most column. This is to allow for explanation of specific lines only. Line numbers are not used in Command Files, but from now on you will have to use FunnelWeb or the E/A Editor to create the Command Files, and this will be easier on me. Since we're on the subject I might as well fill you in. The editor which comes with TIB is not bad. By editor I am referring to the part of TIB you use to write and save Command Files. However, in TIB version 1.02 when you enter about 33 lines you run out of memory space. If you want or need to use the TIB editor you could produce a bunch of Command Files that run each other and get the job done quite well. I prefer to have the luxury of writing larger files if needed. I also prefer the use of embedded control codes as printer commands, which at this piont are not available in the TIB Editor. There are two more reasons to contemplate an outside editor. The first is that the Command Processor that runs your Command Files truncates or chops off all lines at 40 characters. This means you can set tabs at 40 columns and after typing commands on the right half of the page you can tab over past 40 and type in comments. TIB will never see the comment so they won't interfer with the program logic or slow the speed down. Last, I print out lots of hard copies to check my work. It's hard to print files created by the TIB Editor in Int/Fix 40 Format.

Now I'd like you to make a correction in the database we created for Tutorial 1. The problem is in the "XP" field of the database, "TNAMES". As it was displayed in SCREEN FOUR the XP dates were "Month-Year", (02-88), etc. This configuration does not sort to a desirable conclusion in a character field (FEL). In order to get what we want out of a "SORT ON command we need the year first and the month second, ie. "Year-Month", or (88-02). Since we only have five names in TNAMES you can edit the file and change them. I have placed a printout of TNAMES at the bottom of this page for your convenience. You are, of course, going to have to learn something along the way. Let's say that you are really trying to learn TI-Base and you were working frantically on something when this newsletter arrived. Reading to this point you want to start editing immediately. In order to get going you must CLOSE your present file, point TIB at disk 3 (which is where you have the database named TNAMES), un-SORT the file, and you don't like the present screen colors. If you had the little program that's listed below, you could type DO EDTN and TIB would do the rest. So let's make one. Fire up FunnelWeb and press 1 for EDITOR. When you get into the editor press (CTRL O) to get out of word wrap mode. You should then see a hollow cursor. At that point you can type in the Command File, EDTN. When this is done save it to disk under the name EDTN/C, and print out a hard copy which you can compare against the listing below. Remember, you don't type in the line numbers, and any line with an asterisk in the first column is a comment line.

```
0001    * Command File to EDIT TNAMES
0002    *       PROGRAM NAME = EDTN
0003    *       SAVED AS   EDTN/C
0004    *
0005    CLOSE ALL
0006    SET DATDISK=DSK3.
0007    USE TNAMES
0008    SORT OFF
0009    TOP
0010    COLOR WHITE DARK-BLUE
0011    EDIT
0012    CLOSE ALL
0013    RETURN
```

Let's attack this little CF (CF = Command File). Lines 1 through 4 can be anything you need to refresh your memory about this program. Line 5 is a good idea for every CF you own. This line has saved me many times. If there aren't any Dbs (Db = Database) open, then 5 will do nothing. Line 6 is not really needed and you can leave it out. I do change drives on occasion with this statement, but you should remember to change it back at the end of the CF with a similar line. The reason its here is to demonstrate that the CLOSE ALL should come at the very beginning of the CF before you do something like line 6 and confuse the system. Line 7 will open TNAMES on drive 3

TNAMES

REC LN	FN	MI SA	CT	ST ZP PH	XP GP ID
0000 Vivannovitch	Elexxie	I. 111 E. 98th. St.	Cleveland	OH 91023 541-5415	88-05 NOCO 0712881
0001 Aardvark	Grant	E. 9995 State Rt. 84	Geneva	OH 44014 1-465-9876	88-02 NOCO 0717851
0002 Whitman	Raymond (Slim)	A. 2574 East 254th.	Eastlake	OH 44094 951-2345	88-09 NOCO 0921861
0003 Jones	Quincy	W. 37285 Burgandy Laine	Mentor-on-the-Lake	OH 44060 257-1029	88-08 NOCO 0820871
0004 Smoley	Martin	A. 6149 Bryson Drive	Mentor	OH 44060 257-1661	89-02 NOCO 0713831

4

as per line 6 or where ever DATDISK is located if line 6 is omitted. You can also use "USE DSKx.TNAMES" where x is any drive number, including a HORIZON Ramdisk No. 6, "which I use". SORT OFF will un-SORT the file and TOP will point TIB at the first record in the Db, as BOTTOM would point TIB at the last record. Line 10 is all you need to change the foreground and background colors. See page 4-2 of the manual for colors available.   **AND NOW THE EDITOR!**
Line 11, EDIT, will put you in edit mode using whatever Db file is open. In this case TNAMES. While you are in EDIT you can use arrow keys or enter to move around. You can then type over any item you want to change. At this time it is the XP field. "This is important!". You can also use FCTN 6 to page up, or enter, or FCTN 5 to page down to the next record. This could cause a problem as the changes you have made will not always be saved. If you make any changes you should always use FCTN 8 to register, or save, your changes and move to the next record. If you are on the last record in the file, you should still press FCTN 8. This will not end the editing session and you will remain in the last record. You can then press FCTN 6 to page up, or FCTN 9 to leave the editor. In this case you would be returned to line 12 of our CF and TNAMES would be closed. RETURN will end the program and take you back to the DP. NT: If I am editing a file without a program, I close the file with CLOSE or CLOSE ALL as soon as I have finished. This allows TIB to update all of the records. One last idea on this CF. I either SET TALK ON in the first line of this CF or more often it is already on when I run EDTN. This will allow me to read lines 1 through 7 on the screen while the Db is being un-SORTed. I can then see if this is actually the program I wanted, with the right Db, and that I have changed the location of the DATDISK. The beauty of little CF programs like this is that you can build on them and add things you realize you want as you go along. The CF will not forget any of the details from one day to the next like I do. Also, once you have the first one done you can copy it to a new name, (COPY DSK2.EDTN/C DSK2.NEWED/C). You can then use MODIFY COMMAND NEWED to edit this new CF to handle another Db, or do whatever you wish. It's much easier than typing a completely new CF from scratch.
                    IMPORTANT TIP!
I have discovered that a CF created with FunnelWeb in DV/80 format can be copied or edited by TIB and the DV/80 format will not be changed. Therefore, if you create the CF below with FunnelWeb and save it to the name BLNK/C on your DATDISK, you can then copy it to a new name, re-edit it, and save it with MODIFY COMMAND, and it will remain a DV/80 file.

```
>SET TALK ON
>*
>*      Command File BLNK
>*
>*      Save as BLNK/C
>*
>*  Use as a seed file for DV/80
>*
>*  Copy to a name of your choice
>*    and type over this stuff.
>*
>RETURN
```

This means you will then be able to print the CF with FunnelWeb for a hard copy. I try to make sure that all my hardcopies have the program name and pertinent comments at the top. Then if I'm writing a new CF I can look over these hardcopies and then merge chunks of previously written material with FunnelWeb LF Merge capabilities.

Let's get started on this months project. We need another Db to try some new routines. Create TNTST2 using the instructions below. Some of this is a repeat so skip over the parts you know and get right to the data entry. If this doesn't look slightly familiar you should refer back to Tutorial number one for more help.

```
>CLOSE ALL <E>
>CLEAR <E>
>CREATE TNTST2 <E>
```

When the CREATE screen comes up enter the following fields, and when you enter the 0 (zero) in the last column of field 4 press FCYN 8 and wait for TIB to create the file for you.

arrows to move, enter to advance

FIELD	DESCRIPTOR	TYPE	WIDTH	DEC
1	TDATE	D	8	
2	NUM1	N	7	2
3	NUM2	N	7	2
4	ID	N	7	0

After pressing FCTN 8 TIB will ask if you want to enter data now. Answer yes and enter the data supplied below. Take your time, there are a lot of numbers here and you may get confused.

REC	TDATE	NUM1	NUM2	ID
0000	03/16/88	100.11	100.22	0712881
0001	02/29/88	200.11	200.22	0713831
0002	08/27/88	300.11	300.22	0717851
0003	03/03/88	400.11	400.22	0820871
0004	12/30/87	500.11	500.22	0921861
0005	06/06/88	600.11	600.22	0717851
0006	04/22/88	700.11	700.22	0921861
0007	01/21/88	800.11	800.22	0713831
0008	05/12/88	900.11	900.22	0820871
0009	06/17/88	1000.11	1000.22	0713831
0010	03/01/88	1100.11	1100.22	0921861
0011	08/03/88	1200.11	1200.22	0713831

I double spaced the data above to make it as clear as possible. If you make any mistakes, this is a good time to convert EDTN. Type COPY DSK2.EDTN/C DSK2.EDTST2/C <E>. After copying it use Modify Command to change lines 1 through 4, and change line 7 to USE TNTST2. Press FCTN 8 to save and you're done.   **Continued Next Page.**

The CF on this page may look complicated, but its not. We will go through it together, and I will try to explain the important parts. I hope you have read Tutorial 1 so I can skim over the routine parts and concentrate on the rest. Remember don't enter the line numbers. Lines 1 through 9 are strictly housekeeping except for CLOSE ALL, and from now on I will consider it housekeeping. CLOSE ALL should be part of every MAIN CF. By MAIN I'm referring CF that may run other CFs, but is not itself run by a previous CF. This CF runs NUMTST2 as you can see in line 48. You would not want to close all the files in NUMTST2 it would bomb the program. I intend to have my data disk in drive 2, line 9 clears the screen and changing the screen colors has no real value. The WRITE statements from 11 through 19 are to demonstrate user prompts. The lines I have included are not important, but it will give you some idea of ROW-COLUMN display. Line 20 is the beginning of the real stuff. USE TNAMES opens that Db which it expects

```
01   * Command File TNTST2
02   *
03      SET TALK OFF
04      SET RECNUM OFF
05      SET HEADING OFF
06      SET LINE=80
07   CLOSE ALL
08      SET DATDISK=DSK2.
09   CLEAR
10   COLOR WHITE DARK-RED
11   WRITE 2,8," TI-Base Demonstration to"
12   WRITE 4,8,"open two Databases at one"
13   WRITE 6,6,"time and find data in File #2"
14   WRITE 8,6,"which is related to an ID No."
15   WRITE 10,7,"in File #1. With some very"
16   WRITE 12,7,"simple math implementation."
17   WRITE 14,9,"************************"
18   WRITE 16,9,"    Running:  TNTST2      "
19   WRITE 18,9,"************************"
20   USE TNAMES
21      LOCAL CDATE D 8
22      WRITE 20,2," Enter the Date   MM/DD/YY"
23      WRITE 21,2," Within Quotes"
24      READ 21,18,CDATE
25      WRITE 22,5,"Current Date: ",CDATE
26   SORT ON ID
27   CLOSE
28   USE TNTST2
29      SORT ON TDATE
30   CLOSE
31   SELECT 2
32   USE TNTST2
33   TOP
34   SELECT 1
35   USE TNAMES
36   TOP
37         LOCAL BLNK C 4
38         REPLACE BLNK WITH "bE bG"
39         PRINT BLNK
40   LOCAL TESTID N 7 0
41   LOCAL TEMP C 60
42      WHILE .NOT. (EOF)
43      REPLACE TEMP WITH "bE bG" | TRIM(LN) ;
44            | " " | TRIM(FN) | " " | MI ;
45            | " " | ID
46         PRINT    TEMP
47            REPLACE TESTID WITH 1.ID
48            DO DSK2.NUMTST2
49            COLOR WHITE DARK-BLUE
50      WRITE 17,9,"  Running: > TNTST2 <     "
51         SELECT 1
52         MOVE
53      ENDWHILE
54   CLOSE ALL
55      SET RECNUM ON
56      SET HEADING ON
57      SET TALK ON
58   RETURN
```

to find on drive 2. After line 8, TIB will expect to find all CFs and Dbs on drive 2, and I will not waste space bringing it up again. There is a three line cluster which is important. The lines are 20, 26, and 27. Their purpose is to open, sort and close a file. This is identical to lines 28, 29, and 30. However! lines 21 through 25 are of interest. Line 21 initializes the LOCAL variable named CDATE, which is a D (date type) entry with a length of 8 characters. A variable is a place to store some type of information. In this case it will be the Current DATE (CDATE) which you will type in when asked. Lines 22 and 23 will ask you to enter the date and 24 will place the cursor on the screen one space after "Within Quotes", and wait for your input. NOTE: with Version 1.02 all Characters, or Dates, which are characters, must be input enclosed in quotation marks, "09/01/88". Line 25 will write the message "Current Date:" and display whatever you type in for CDATE. FYI: I have initialized CDATE close to its use for your benefit. I will continue this procedure in this program, but from then on variables should be initialized at the beginning of a CF. This little chunk (LNs 21-25) was stuck in here because TIB likes to have a Db open before you READ to a variable. AND! line 25 WRITEs to screen line 22. You will notice that screen line 22 doesn't scroll like the rest. You can put a message there and it will stay put until a CLEAR or another WRITE 22,x removes it. Some of these things will be apparent when you run this program, or DO TNTST2. Line 31 leads us into a very complicated and confusing area. I will try to cover it as thoroughly as possible. I will re-analyze it many times in the future, "It's that important."

Think of a Lazy Suzan, or a rotatable table. This table has 5 areas on it with low partitions between each area. You can take from one to five file folders which are filled with sheets of paper and place one in each of the five areas. You must stand in one spot, but you can SELECT one of the five areas to be positioned directly in front of you. The area SELECTed, (1-5), is the one where you can do the most work, but you can see over the partitions to do limited things with the information in the files which are not directly in front of you. If you can grasp this concept and visualize the 5 different slots, or areas, you're going to catch on fast. Remembering, or keeping track of what can be done in non-SELECTed slots is a challange. NOW! line 31 and beyond. Line 31 SELECTs slot #2, 32 opens the Db named TNTST2 in slot #2 and 33 points TIB at the first record in the file. Line 34 SELECTs slot #1, 35 opens the Db named TNAMES in slot #1 and 36 points TIB at the first record in that file. Remember that both of these Dbs were previously SORTed to our specifications. Well, we have done it. At this point we have open two databases at the same time. TNTST2 is open in slot #2 and TNAMES is open in slot #1, and if we don't count all the junk I put in to add flash to the program, we did it with about 12 lines of code. I told you that we'd get through this somehow. If you examine and keep track of this stuff one peice at a time, you'll get the hang of it sooner than you think.

Continued Next Page.

NOTE: Although there is a CF on this page we will be discussing the CF on the previous page until we reach line 48. Don't get confused. We just left line 36. Lines 37, 38 and 39 make up a small group.

Their purpose is to initialize the LOCAL BLNK for 4 Characters. Fill it with the control codes that set the printer to Emphasized and Doublestrike, and send the codes to the printer. Entering the control codes can only be done with TIW or FunnelWeb at this time. There'll be some useless repetition concerning control codes. I'll explain later. Line 40 and 41 initialize two more variables to be used in the WHILE loop. TESTID is to hold a Number with the length of 7 and 0 decimal places, and TEMP has been discussed previously. "Let's get into the meaty part." The way I have set this loop up it will continue to do everything from line 42 through line 53 until it reaches the End Of File marker, (EOF), for the Db TNAMES which we have located in slot #1. NOTE: TNAMES is in slot #1, and slot #1 is our currently SELECTed slot because the last slot we worked with was slot #1, in line #34. "I'll keep at this concept as we go along." Line 42 has a simple but important job. It immediately checks to see if we have hit the EOF in whatever file is open, in the slot we are facing. In this case it is slot 1 and the file is TNAMES. WHILE it does .NOT. encounter the (EOF), in that file it proceeds to line 43. If it does encounter the (EOF), it goes directly to the line after the ENDWHILE, which in this case is line 54. Lines 43, 44 and 45 are seen by TIB as one continous line because of the semicolon (;) at the end of lines 42 and 43. So in this line TIB is going to take the 2 control codes directly after WITH and hold them. It will then TRIM the trailing blanks from LN and attach it behind the control codes, and then it will stick 2 spaces behind that. It will TRIM the trailing blank spaces from FN, attach it to our growing string, and then throw in another blank space. It will then tack MI on that followed by 2 more spaces, and last but not least ID. We did not TRIM MI because MI doesn't have any extra blank spaces. We did not TRIM ID because it is of (N)umeric type (a number) and TRIMming is only used on (C)haracter strings. Then TIB takes this whole mess we have put together and sticks it into the variable we call TEMP. NOTE: If you look back at line 41 you'll see we made TEMP with 60 spaces. When you fill up a variable with all kinds of junk, like we just did, you must make sure the variable is big enough to hold it all. In line 46 we PRINT all the junk we just put into TEMP.

```
01    * Command file NUMTST2
02    *
03    CLEAR
04      WRITE 15,9,"***************************"
05      WRITE 17,9,"    Running:  NUMTST2    "
06      WRITE 19,9,"***************************"
07    WAIT 2
08    COLOR WHITE DARK-GREEN
09      WRITE 17,9,"Looking For ID No.",TESTID
10    WAIT 2
11      LOCAL TNUM1 N 10 2
12      LOCAL TNUM2 N 10 2
13      LOCAL STNUM1 N 10 2
14      LOCAL STNUM2 N 10 2
15      LOCAL T C 8
16    REPLACE T WITH "   TOTAL"
17    SELECT 2
18    TOP
19      REPLACE BLNK WITH "&4 "
20      PRINT BLNK
21    WHILE .NOT. (EOF)
22      IF TESTID = ID
23        PRINT TDATE,BLNK,NUM1,BLNK,NUM2 ;
24            BLNK,ID
25        REPLACE STNUM1 WITH TNUM1 + NUM1
26        REPLACE TNUM1 WITH STNUM1
27        REPLACE STNUM2 WITH TNUM2 + NUM2
28        REPLACE TNUM2 WITH STNUM2
29      ENDIF
30      MOVE
31    ENDWHILE
32      REPLACE TEMP WITH "--------------------";
33      ! "------------------------"
34      PRINT TEMP
35    PRINT   CDATE,TNUM1,TNUM2,T
36      PRINT BLNK
37      'REPLACE BLNK WITH "&5 "
38      PRINT BLNK
39    RETURN
```

I'd like to also mention that the junk we filled TEMP with was related to TNAMES, (LN,FN,MI,ID). Using this type of data gathering it is up to us to be sure TIB is pointed at the right slot and that slot contains the Db that holds the information we want. In line 47 we put the same ID number from above into the holding area we named TESTID. The phrase 1.ID is another way to tell TIB that we want the ID number from slot #1. At this time the 1 is FYI only, and doesn't have any real effect on the program except to asure me that I am getting ID from slot #1. TIBs Author uses this form of data gathering in the CF named PROCESS, manual page 5-5.

"Well here comes line 48." When TIB hits line 48 it leaves the CF named TNTST2 with everything exactly as it is and executes the CF named NUMTST2 on Disk 2. So now we start looking at the lines in NUMTST2. Line 3 CLEARs the stuff left on the screen by that other CF. Lines 4, 5 and 6 put up a new message. TIB WAITs 2 seconds, then it changes the COLORs to WHITE on DARK-GREEN. In the same instant it reWRITEs a new message to line 17, followed by the TESTID. This is the Identification Number we brought with us from that other CF. We will use it to find related data in the Db you typed all those numbers into a short time ago. TIB WAITs a couple more seconds just for kicks and we're on our way. Lines 11 through 15 initialize all the variables we will need in this program. We can also use variables from that other CF, but we cannot send these variables (lines 11-15), back there. If we needed to send something back over there, we could put it in one of the variables from that CF (like TEMP) just before we RETURN and then we could use that information when we RETURNed to that other CF. I did not use the names of the 2 CFs in that explanation because it was even more confusing that way. Line 16 places some blank spaces and the word TOTAL into T for later use. In line 17 we SELECT slot #2, which is where TIB holding the Db TNTST2. TOP in line 18 is only to make me feel secure. We should already be at the TOP of the file. Lines 19 and 20 stick the control code for Italics into BLNK and PRINT it. NOTE TWO THINGS: One, we needed a blank line printed anyway, which this gives us, and Two, that BLNK came over from that other CF. "NOW!", in line 21 we have another WHILE loop. The WHILE loop that runs from line 21 through line 31 has the same definition I gave earlier, but we will do different things while we are inside this loop, and it will be looking for the (EOF) for TNTST2 in slot #2.

7

I can't believe I'm on page 5. Well, since I'm using up so much Newsletter space, here is a promo for my sponsor.

Join The NorthCoast 99'ers UG

NorthCoast has 3500 plus programs in it's library and produces this great little Newsletter. You can take full advantage of the club's services by mail, and you will be certain of receiving my wonderful tutorials in the future. The membership cost for someone living in the continental United States is only $15.00. You can send your membership fee to me, Martin A. Smoley, 6149 Bryson Drive, Mentor, Ohio, 44060. Make all checks payable to NorthCoast 99'ers User Group, DO NOT send cash, and I'll expedite your membership personally.

"OK, NUMTST2, line 22." When we get to this point TIB is looking at the first record in NUMTST2, which we have SELECTed in slot #2. Therefore, in line 22, IF the value in TESTID matches or is equal to the value in ID, then TIB will execute all the lines between the IF (line 22) and the ENDIF (line 29). Remember, TESTID holds the ID number which matches the LN, FN and MI we just printed in from the Db TNAMES. ID holds the ID number from the current record of the DB TNTST2. I will not follow the program accuretly because TIB will not find a match to make the IF true until the sixth record of this Db. So lets say it finds a match which makes line 22 true. Line 23 prints the information held in TDATE, NUM1, NUM2, and ID under the persons name from TNAMES. Lines 25 and 26 make up an accumulator that keeps a running total of the NUM1 part of any matching records. Similarly lines 27 and 28 keep a running total of the numbers in NUM2 if the ID match is true. Coming from line 28 to line 29, TIB ignores 29 and goes directly to line 30. This line tells TIB to MOVE its pointer to the next record in the file. So we are now looking at the next record in the Db TNTST2. The ENDWHILE in line 31 is not ignored by TIB, and TIB is sent back to line 21 to test the new ID we now have against TESTID which remains the same. This loop goes around and around. Each time it does, it moves to the next record and then checks for (EOF). If its not the End Of File, and it has data to work on it immediately tests to see IF the ID numbers match, etc. When it runs out of data or hits the (EOF), line 21 sends TIB directly to line 32, the first statement after the ENDWHILE. TIB then puts the dashed line into TEMP and prints it. TIB then prints the current date (CDATE), which you entered at the beginning of that other CF, the totals in TNUM1 and TNUM2, and the word TOTAL. In lines 37 and 38 TIB turns Italics off, at the printer. We then RETURN to that other CF named TNTST2. In doing so we throw away all the LOCALs we initialized in this CF. When we land back in the CF named TNTST2 we land on line 49, which changes the screen colors. Line 50 WRITEs this CFs name to the screen over screen line 17, which was left there by that other CF. Line 51 SELECTs slot #1, so we are once again working with TNAMES. Line 51 MOVEs TIBs pointer to the next record, for a new name, and line 53 sends us back to line 42 to start the whole process over

again. These two loops will ratchet through the names in TNAMES one at a time, and for each name in TNAMES, will completely search TNTST2 for any information that is related to that name by comparing ID numbers in TNAMES to ID numbers in TNTST2. It will continue to search until it runs out of names or records, in TNAMES. At that time 42 will send TIB to lin. 54. ALL Dbs will be CLOSEd, things that were turned off will be turned back on and the whole thing is finished. In line 58 you are RETURNed to the Dot Prompt. That just about raps this tutorial up except for a few things I said I'd get back to. I threw around a lot of control codes in this set of CFs. If your using FunnelWeb to produce your CFs, you can carry these ideas back to the LABEL program we did last month. Fire up FunnelWeb and retype the CF called LBLS1/C, but this time name it LBLS2/C. There are only about 32 lines and most of them are very short. Leave out the present line that reads LOCAL BLNK C 1. Next, add lines 37, 38 and 39 from TNTST2. Insert them between the line that says TOP and WHILE .NOT.(EOF). This will cause your printer to print in Emphasized and Doublestrike Mode. If you don't like that, try what I did in line 43. You can concatenate (|) control codes on the front and rear of a character string. There are lots of ways to do it. Before my mind goes completely I'm giving up. I copied the printout from this months stuff below. I'd also like to add that this set of CFs make a nice club demo.

Vivannovitch Elexxie I. 0712881			
03/16/88	100.11	100.22	0712881
————————	————————	————————	————————
09/11/88	100.11	100.22	TOTAL

Smoley Martin A. 0713831			
01/21/88	800.11	800.22	0713831
02/29/88	200.11	200.22	0713831
06/17/88	1000.11	1000.22	0713831
08/03/88	1200.11	1200.22	0713831
————————	————————	————————	————————
09/11/88	3200.44	3200.88	TOTAL

Aardvark Grant E. 0717851			
06/06/88	600.11	600.22	0717851
08/27/88	300.11	300.22	0717851
————————	————————	————————	————————
09/11/88	900.22	900.44	TOTAL

Jones Quincy W. 0820871			
03/03/88	400.11	400.22	0820871
05/12/88	900.11	900.22	0820871
————————	————————	————————	————————
09/11/88	1300.22	1300.44	TOTAL

Whitman Raymond (Slim) A. 0921861			
12/30/87	500.11	500.22	0921861
03/01/88	1100.11	1100.22	0921861
04/22/88	700.11	700.22	092186
————————	————————	————————	————————
09/11/88	2300.33	2300.66	TOTAL

Continued Next Month.

TELEVISION TO MONITOR
(A Simple Cable Conversion)
by Glenn Bernasek, TI-CHIPS, Cleveland, Ohio

Has your old B&W gone "belly-up", and your family (spouse) says no-way to your suggestion to hooking up the TI to the family's color TV? As luck would have it, you pick up a Trading Times, and lo and behold, someone is selling a used monochrome monitor. This is just what you are looking for to replace the dead B&W. However, either the seller can't find the interface cable, or the cable won't fit the output plug in the back of your TI-99/4A.

What to do? Do you try to find a TI monitor cable (fat chance now), replace the incompatable console plug with one that works, or just cut off the 99/4A cable from the Modulator and fit it with two RCA connectors? All of the above will work , but I have a "better" solution to the problem of interfacing the TI with the monitor without sacrificing the potential use of the TV modulator at some future time.

As it turns out, the TI modulator interior is accessible to both the hobbyist and the tinkerer for experimentation. All that has to be done is pop off the top and bottom covers, and the inner secrets of the device are exposed to the world! Therefore, you are able to modify this unit as you desire.

Following is a list of items you will need to complete this project.

1. One (1) RADIO SHACK, "ARCHER" brand, (Low-loss) AUDIO/VIDEO DUB CABLE Cat. No. 15-1538.
2. A low wattage (Pencil) soldering iron, resin core solder and resin flux.
3. A pair of needle-nose pliers, a small screw driver and a pair of tin snips.
4. A SHARP knife or wire strippers and a pair of side cutters.
5. A couple of small (1/2 inch) pieces of #118 guage wire insulation.
6. A well lighted work area and about 15 to 30 minutes of spare time to do this SIMPLE project.

Now that you've got everything together, let's get started. First, cut the Dub Cable in HALF (one half will be used for the VIDEO lead and the other will become the AUDIO lead), and CAREFULLY strip about one inch of insulation from each cut end. Untwist the braided shield, and retwist it into a separate lead. The CAREFULLY strip about 1/2 inch of insulation from the center conductor. At this point, it would be a good idea to pre-wet all four (4) stripped wire ends with a coating of solder. Set the preparred cables inside.

Let's take a look at what we're about to modify. With the small screw driver, pry off the snap covers from both sides of the TI MODULATOR. When you look into the TOP side, you'll see some compartments and where the main cable from the 99/4A enters. If you look closely, you'll notice that the wire connections are NAMED on the circuit board. This way you'll know which is the VIDEO and which is the AUDIO leads. Now turn the Modulator over and look at the solder points that are in the SAME area as the cable wires were.

You're going to LIGHTLY solder one (1) center cable conductor to the VIDEO IN solder point and one (1) center cable conductor to the AUDIO IN solder point. (See the enclosed illustration for an idea of what I'm talking about.)

At this point, slip a piece of 1/2 inch insulation over each re-twisted copper shield lead, and solder these leads to the large area of the circuit board that is connected (soldered) to the Modulator's frame. This is the return GROUND loop for both the VIDEO and AUDIO circuits.

Now cut a SMALL 1/8 inch wide by 3/8 inch long notch into the corner of the bottom cover where the Monitor leads will come out. All it takes is two (2) parallel cuts and a bend-out with the needle-nose pliers of the notch piece.

GENTLY press the soldered connections to the board to allow room for the cover to snap in place (with the leads extending through the notch). Then replace the top cover, making sure the modulator switch is in the PROPER position on the interior switches, by snapping it shut as you did with the bottom. Your TV/MONITOR modulator unit is now ready for service. The best part of this modification is that this unit will now serve equally well on either a TV or computer monitor. By the way, I hope you took the time to mark each lead as to where it goes.

If you find that the Video input jack on the monitor is a BNC type connection (it has ears on it) instead of an RCA type jack (as both plugs on your new leads are), then all you have to do is pick up an RCA/BNC adapter at RADIO SHACK and you're in business.

This is an easy project that works, and the modification won't interfere with the normal operations of the TI Modulator.

A message from the Delphi TI NET forum.

5844 9-JUN-01:18: News
RE: A look at the CW99'ers mailbox (Re: Msg 5665)
From: SMICKELSON To: ARTBYERS

The Z80 Simulator seems it will come out as a card, for both the TI-99/4A as well as the Geneve. It will have the latest Z80 upgrade CPU, which will permit a clock speed of 16 MHz, enabling software to run 3 to 4 times as fast as it origionally did on the ADAm. The Nintendo software, if ever made available, will be through a royalty arrangement with the manufactures of same.

Back to the Z80 card, though it,(i.e. the z80 simulator software), runs on both the TI (in a Maximem) or on the the Geneve, it runs better on the Geneve, though I understand still slower than on the ADAM, which brings the necessity of having a Z80 (I think a Z8000)CPU with 64K RAM. The Z80 card will also hav e an assembler utility, to enable the user to write/program their own CPM software. Gary Bowser, maker of the card, thinks the unit can run MSX software in the Geneve. The TI or Geneve will control the I/O to disk, joysticks, screen, while the software will use the hardware environment provided by the card, to operate a speed close to origional. Gary has made one such card on a proto board.

T.V. VHF LEADS

TI MODULATOR (BOTTOM SIDE)
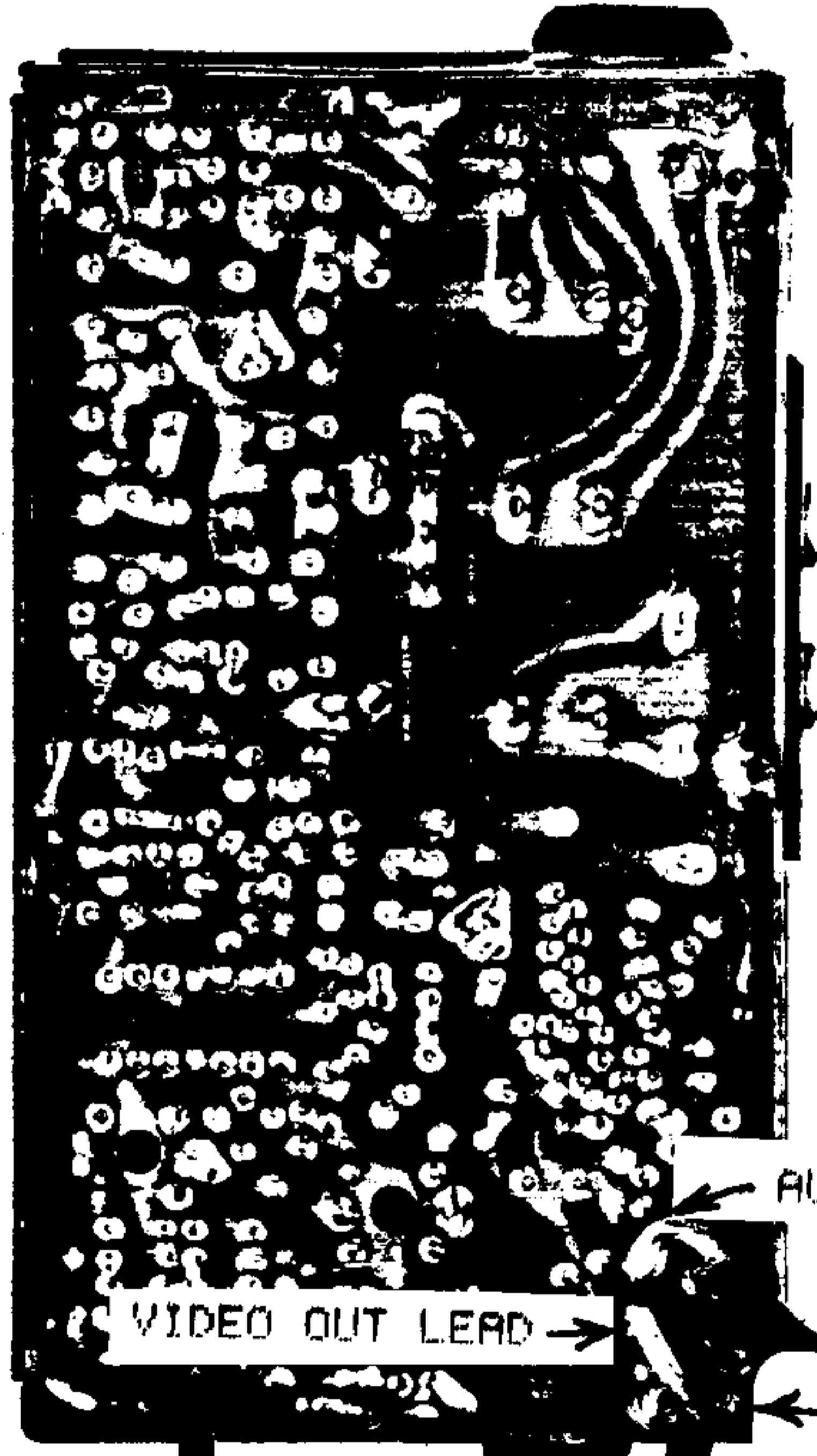
AUDIO OUT LEAD

VIDEO OUT LEAD →

COMMON GROUND POINT

MAIN CABLE FROM THE TI-99/4A.

USING TI WRITER INCLUDE FILE AND
TRANSLITERATE COMMANDS IN THE PLUS!
ENVIRONMENT - PART2
By Mark Armstrong
Bluegrass 99 Computer Society - August, 1988

ED NOTE: Last issue we printed Mark's tutorial on the transliterate command as background for using the PLUS! diskette from Jack Sughrue. We also printed an overview of the utilities on that disk. Rather than "reinvent the wheel", we are using Mark's very fine review and tutorial of the TI Writer files and transliterate commands on that disk.

I. Summary of Part One

PLUS! is a fairware disk which is a companion disk to TI Writer word processing. It rests on three concepts: First, the creation of an Include File which contains Formatter Commands; Second, the use of Transliteration Commands in the Formatter Command file so that the user is able to invoke complex formatter commands with only a few key strokes; and Third, the use of the Special Character mode of the TI-Writer so all keyboard characters remain available. The purpose of this Part Two is to discuss specific implementation of these three concepts.

II. PLUS! Files

PLUS! contains nine files labeled C1 through C9. These files contain a series of formatter commands. Following are the formatter commands contained in the C9 file:

Line 0. `.FI;AD;LM 3;RM 74;IN +2
Line 1. ^A .TL 1:32,27,52   Italic ON
Line 2. ^B .TL 2:27,53,32   Italic Off
Line 3. ^C .TL 3:32,27,83,0   Superscript On
Line 4. ^D .TL 4:32,27,83,1   Subscript On
Line 5. ^E .TL 5:27,84,32   Script Mode Off
Line 6. ^F .TL 6:27,15   Condensed ON
Line 7. ^G .TL 7:18   Condensed Off
Line 8. ^H .TL 8:27,87,1   Double Wide On
Line 9. ^I .TL 9:27,87,0   Double Wide Off
Line 10. ^J .TL 10:32,27,51,11   Line Feed Adjustment
Line 11. ^K .TL 11:32,27,50   Line Feed Adjustment
Line 12. ^L .TL 12:7,32   Sound Printer Bell
Line 13. ^M .TL 14:8,32   Backspace
Line 14. ^N .TL 15:32,27,71   Double Strike On
Line 15. ^O .TL 16:27,72,32   Double Strike Off
Line 16. ^P .TL 17:32,27,69   Emphasized Mode On
Line 17. ^Q .TL 18:27,70,32   Emphasized Mode Off
Line 18. ^R .TL 19:32,37,66,2   Elite Spacing On
Line 19. ^S .TL 20:27,20,27,66,1,32 Elite Spacing Off
Line 20. ^T .TL 21:32,37,45,1   Underline on
Line 21. ^U .TL 22:27,45,0,32   Underline Off
Line 22. ^W .TL 23,32,27,69,27,71,27,45,1
         Doublestrike, Emphasized, Underline On
Line 23. ^X .TL 24:;27,45,0,27,72,27,70,27,32
         Doublestrike, Emphasized, Underline Off
Line 24. ^Y .TL 25:32,80,76,85,83,33 Graphic definition
Line 25. ^Z .TL 26:27,32,173,174,173,174 Graphic
         definition
Line 26. ^1 .TL 0:27,64   Master Reset

A look at the first six lines of formatter code reveals how the transliteration codes work. Line 0 in this C9 formatter file merely sets the left and right margins at 3 and 74, paragraph indent at 2 characters to the right of the left margin, and invokes the fill and adjust commands.

The expression, " A", in Line 1 is shorthand for the key strokes Control U, Shift A. These key strokes are entered in the calling document to invoke the particular effect in C9 file. Control U accesses the Special Character Mode of TI-Writer which, in turn, allows access to ASCII codes 0 through 26. Thus, in line 1, ASCII codes 32, 27 and 52 are transliterated to ASCII code 1. ASCII code 32 is a space, ASCII code 27 alerts the printer that the next code received is a printer code. ASCII 52 is a printer command to turn on Italic Print Mode. Thus, in the calling file, control U, Shift A will cause the printer to print a space and print subsequent text in Italics.

In line 2, ASCII codes 27,53 and 32 are transliterated into ASCII code 2. ASCII code 2 is accessed through Control U, Shift B. When ^B is entered in the calling file, the printer will exit to the Italic mode (ASCII codes 27 and 53) and print a space (ASCII code 32).

In line 3,ASCII codes 32, 27, 83, and 0 are transliterated into ASCII code 3. Entering C in the calling file will cause the printer to print a space (ASCII code 32) and to enter its superscript mode (ASCII code 27,83,0)

In line 4, ASCII codes 32, 27, 83 and 1 are transliterated into ASCII code 3. Entering D in the calling file will cause the printer to print a space (ASCII code 32) and to enter its subscript mode (ASCII code 27, 83, 1)

In line 5, ASCII codes 27, 84 and 32 are transliterated into ASCII code 5. Entering E in the calling file will cause the printer to exit the script mode.

Each subsequent line can be analyzed in the same manner to determine its effect. Several important points need to be made. Notice in line 22 that several print modes have been combined. Notice also that PLUS! is not limited to control over the print quality modes. In lines 24 and 25 ^Y and ^Z a graphic has been designed and transliterated. In this regard, PLUS! graphics have been designed using Gemini graphics and do not work particularly well on Epson printers. Some graphics will print; however, Epson users will be disappointed with the graphics in files 61, 62 and 63. Finally, note that an attempt is made in line 12 to transliterate ^L. In fact ^L is defined by TI-Writer as a form feed which will take precedence over the transliteration. ^J is also defined as a linefeed; however, ^J will cause the line spacing to adjust to fit the script mode invoked in lines 3 and 4. Accordingly, the user is cautioned not to use ^L unless a form feed is required.

The formatter command files C1 through C9 follow the same outline:
         ^A Italic On
         ^B Italic Off
         ^C Superscript On
         ^D Subscript On
         ^E Script Mode Off
         ^F Condensed On
         ^G Condensed Off
         ^H Double Wide On
         ^I Double Wide Off
         ^J Adjusts line to Script Mode
         ^K Backspace
         ^L Formfeed
         ^N Double Strike On
         ^O Double Strike Off
         ^P Emphasized On
         ^Q Emphasized Off
         ^R Elite On

˙S Elite Off
˙U Underline On
˙V Underline Off
˙W Varies according to file
˙X Varies according to file
˙Z Varies according to file
˙1 Master Print Reset

Some personal experimentation is required to determine which style is most suitable for the user's requirements. C1 does not alter print style but allows the user to enter various modes. C2 switches the print mode to condensed and allows condensed Italic, condensed underline et cetera. C3 sets cpl to 12. C4 sets 1 inch margins, 10 character indent and emphasized printing. C5 sets an 80-column page. C6 is left open for user definition. C7 redefines words and phrases. C8 prints text in superscript. C9 is discussed above. Files G1, G2 and G3 set up graphic definitions. Files L1, L2 and L3 set up different letterheads.

III. Access to PLUS! Files

When the user enters the text editor of the TI-Writer, the first line should Include File C1 through C9 as required by the user. After this line is entered, text is entered as usual. It is at this point that ˙A through ˙Z is entered to change print type styles. This file then becomes the calling file. The calling file is saved. When the calling file is printed through the formatter, the transliterations are performed and the text is printed in the style set by the transliteration.

PLUS! thoughtfully provides letterhead styles in formatter files L1, L2 and L3. These letterheads are also accessed by simply entering an Include File DSKn.L1. After the line is entered, the required text is entered with ˙A through CZ as required. Again, it is important for the user to familiarize himself with the various letterhead styles provided.

Conclusion

PLUS! is an excellent companion to TI-Writer Word Processor. Although the average user will not need 9 different styles of print and three different letterheads, nonetheless, software access to printer control gives the user necessary control over document production. One use which is not stressed in the PLUS! documentation is the transliteration of graphic designs. It is possible for the user to prepare graphic designs far in advance of printing and to access these graphic designs as the text requires. Using PLUS! in this manner would seem to give the TI-99/4A user very inexpensive, albeit somewhat crude, desktop publishing capability.

The PLUS! disk contains numerous programs in addition to the PLUS! transliteration files. These files consist of a Banner program, screen dump, label program, multicolumn printer and calendar. These utility files are probably worth the fairware price of this diskette. PLUS! is heavily documented and following the documents will enable the user to access the capability of PLUS! The author of PLUS! is Jack Sughrue, P.O. Box 459, East Douglas, MA 01516.

** NEWSLETTER SUBSCRIPTION ONLY **

CLEVELAND AREA TI-994/A USER
GROUPS NEWSLETTER
C/O DEANNA SHERIDAN
20311 LAKE ROAD
ROCKY RIVER, OH 44116

CHECK YOUR SUBSCRIPTION. THIS MAY
BE YOUR LAST NEWSLETTER!

TIME DATED MATERIAL!!!!!