BELOW YOU WILL FIND A PROGRAM THAT WON'T TAKE TOO LONG TO KEY IN WITH 32K AND EXTENDED BASIC. I CAN'T GET IT TO WORK ON MY CONSOLE, BUT I BELIEVE THAT IT IS NOT THE PROGRAM, BUT THE VERSION OF OPERATING SYSTEM IN MY CONSOLE. IT IS A MACHINE CODE PROGRAM LOADED WITH XBASIC, AND RUNS IN LOW MEMORY. MR. MIKE STANFILL OF THE DALLAS TI U.G. WROTE IT, AND I VERY CAREFULLY KEYED IT IN. AFTER YOU TYPE 'RUN', THEN DO A 'CALL LINK("S")'. ON MY CONSOLE, THE CURSOR MOVES DOWN FOUR LINES, NAD THEN THE CONSOLE LOCKS UP. LET ME KNOW IF YOU GET OTHER RESULTS. I'LL NOT TELL YOU WHAT IT DOES, BECAUSE I'LL KNOW IT WORKS FOR YOU IF YOU TELL ME WHAT IT DOES! NOTE: IF YOU DO GET IT TO WORK, YOU CAN CHANGE THE "CALL LINK("S') TO ANY SIX ASCII CHARS. FROM P. 196 IN YOUR EXTENDED BASIC MANUAL. GO TO LINE 3, THE SIX GROUPS OF NUMBERS, 83, 32,32,32,32,32 ARE S. . . . . . CHANGE THEM TO WHATEVER YOU DESIRE.

```
1 CALL INIT
2 !*************************     * BALL PARK-A TINYGRAM *    *   BY MIKE STANFILL
   *    *     DALLAS TI UG       *    *************************
3 CALL LOAD(16376,83,32,32,32,32,32,36,246,"",8194,37,114,63,248)
4 CALL LOAD(9460,4,91,2,0,16,0,2,1,27,44,2,2,0,70,4,32,332,36,200,0,131,204)
5 CALL LOAD(9482,216,32,37,42,131,206,248,32,37,42,131,253,4,199,3,0,0,2,3,0,0,0
)
6 CALL LOAD(9504,152,7,131,206,22,1,16,231,4,91,1,0,3,140,26,144,36,3,134,13,144
,18)
7 CALL LOAD(9526,3,142,15,144,18,3,141,17,144,18,3,131,21,144,18,3,141,17,144,54
,3,141)
8 CALL LOAD(9548,23,144,54,3,140,26,144,36,3,134,13,144,18,3,142,15,144,18,3,141
,17,144)
9 CALL LOAD(9570,18,3,131,21,144,18,3,141,17,141,54,3,159,191,223,0,68,73)
```

LEHIGH 99'ER COMPUTER GROUP
P.O. Box 4437 * 1501 Lehigh St.
Allentown, PA 18103

JOHN GUION
11923 QUINCEY LANE
DALLAS, TX. * 75230

President	Ira Lieberman	820-6332
Vice-pres	Brad Snyder	826-4032
Secretary	Brad Snyder	826-4032
Treasurer	Brad Snyder	826-4032

Vol. VI, No. 7 November, 1988
Ex-Editor Jack Zawedluk 821-1013

LEHIGH 99'ER COMPUTER GROUP

Next meeting: 7:30 PM, Monday
December 19,1988

Conference Room A-D, Second Floor
Sacred Heart Hospital
4th and Chew Streets
Allentown, Pennsylvania

VP-COLUMN

Hello all!, I know it's been a while since a newsletter has appeared from our group, but our newsletter editor Jack Zawedluk has resigned from his post a few month ago, and no one seems willing to take over for him. If we can't get anyone to take over the job permanently, we will no longer publish a newsletter. This will be very unfortunate, as the other clubs that we exchange newsletters vith will no longer know that we exist and will stop sending us their newsletters. Many of which are filled with valuable information. Jack has been sorting our library of newsletters, so get to him if you want to borrov any. There is a $5.00 deposit required on borrowed newsletters.

We are going to try to have another mailing of newsletters this year yet. Our bulk mailing permit is paid up for this year, and stamps must be bought in rolls of 500. This gives us enough for two mailings, that's why there will be another mailing yet.

While I'm still on the subject of newsletters yet, let's try to get some orginal material in our (probably) last few newsletters. If you have anything to say about some program, or maybe some tips on using hardware/ software, or ideas that would help the club, etc., even if it's just a few lines, fire up Funnelweb, Ti-writer, or just jot it down on paper and get it to me. Also, if someone with a printer with adjustable tractors is willing to print out the labels for the next mailing, let me know. (Try running labels through without a tractor on one side, I did). I have the labels and the Ti-writer files of the other clubs that we mail to. (Thanks Jerry for maintaining that list!)

PLEASE, if anyone could be our newsletter editor, let us know!! Jack has promised to help out whoever would take over, and I will also help. But we need someone to volunteer.

At the last meeting, Ira (president) could not attend, so I attempted to fill his shoes. We did get some interest in holding classes on assembly language programming, so we are going to meet at Jack's house and try to learn something. I will be sharing what I know on the subject. If you are interested in attending, give me or Jack a call. John Geisinger has placed an announcement for our club on the cable tv channel in his area - thanks John! We also received an idea to demo a bulletin board session at a meeting. Since we don't have access to a phone line at the meetings, the session would have to be video taped at home. Is anyone interested in doing this? You just need a VCR to record a session (not a camera). We had a disk of the month for sale. We are buying more blank disks to sell to club members at $3.50 for ten.

At the next meeting I will try to give a demo of Batch-It, a new program from Asgard Software. This program runs 'behind' other programs and can control them. Batch-It and whatever other program you are running appear to run at the same

CONTINUED ON PAGE 4

THE GENEVE IS HERE , FINALLY -

Part B
by Jerry Boyer

Hi everybody. It's been a while since I wrote anything about the Geneve cause first there wasn't too much to write about this summer and second I didn't have my Geneve for 2 months. It all started when I ordered the HARD DISK CONTROLLER CARD from MYARC. I received the card in just a couple days right from MYARC which was good. But after many hours of frustration, I still coudn't format or use my new 40 meg. hard drive. After several phone calls to Myarc, they admitted that I didn't have the correct software to operate the hard disk and that I must have messed something else up on the Geneve cause now it won't even work the regular disk drives. They had me send the complete Geneve system and all cards that I was using back to their New Jersey address. A month went by and I received the system back with a note stating that I shorted out a databuss when I was installing or removing the card when I was moving the card from one system to another. They stated that a small electric charge must have shorted it out by touching the card against the P.E.box case. They said that I should have waited 5 full minutes before moving the cards and then to be very careful. But what they didn't do was to send me the correct software and the addendum to set up the hard drives. I finally got Lou Phillips on the phone and he said to send in my old software disks and he would copy the new programs onto them. This I did and it was another month till I got them back. Mr. Phillips was very patient with me about this whole affair and he treated me quite well, in fact he didn't charge me for repairing the cards that I messed up. This seemed pretty decent of him. Now I got this thing up and running and it's just great and extremely fast. It took some time to get it just right.

The first thing you have to do is to open up the HFCC and set the dip switches. There are two sets of switches. The 4 switch set is for the CRU address of the card. It has to be set for CRU>1000 by putting all 4 switches ON. With the present software the card will not support floppy drives, so you have to keep your regular disk controller card in the box with the HFCC. On the 8 switch set leave all the switches set to off. These are to control floppies when the software works in the near future. Connect up the two cables, a 20 pin and a 34 pin, to the card and to the hard disk. Make sure that #1 on the card matches #1 on the drive on both cables. Now is the time to turn on the power. You should power the hard drive first to give it a chance to come up to speed before you turn on the P.E. box. Next go to GPL and load the MYARC DISK MANABER V program so you can format the hard drive. On the first menu go to UTILITY and select FORMAT. The program will ask you to choose hard or floppy, select hard. The next screen asks for the info on your hard drive, such as volume name (usually HD1); sectors per trk (usually 32); number of heads and number of cylinders (this differs with different hard drives). After you input this info I suggest that you accept the defaults that the program gives you, at least until you are really familar with your hard drive. I tried different set-ups that people said would work but all that they did was to totally confuse me and make the hard drive develope errors, so I went back to the default settings and everything worked the way it should. The hard drive takes about 2 beers to format (about 15 min. for a 40 meg drive). You should then copy SYSTEM/SYS to the first level (where you should put all of your most used programs) then set up a subdirectory named "DSK1" and put the "LOAD/SYS" file in it. Now the hard disk will boot up the system. Well, that's it,BYE.

NEWS FROM THE OCTOBER 16 CENTRAL PA. FAIR

NEW SOFTWARE

TRIAD By Wayne Stith

Triad is combination 40-column text editor, Disk manager, and Terminal emulator all rolled into one and residing in memory at the same time. The 40 column editor has keypresses that are consistent with TI-Writer but only in fixed mode. It currently has a 13K buffer. The Disk manager does all the typical things such as, format, copy, rename, sweep, and copy, delete, protect/unprotect, files. It can even View text files.

The Terminal emulator runs at 3, 12, or 24-Hundred baud with even, odd or no parity from port 1 or 2 in full or half duplex. Will up/download in Xmoden with CRC error/checking as default, checksum available. also many other features (to many to mention here)

PRESS By Charles Earl

This one looks like the best word processor yet. Press allows you to create single files as large as your disk space, allows you to type in columns on the screen, include pictures in your text (and on the screen). What you see is what you get, (bold text is bold on the screen underlined is underlined). It even has a 120,000 word spell checker built-in, user defined macros and pull down menus. It is very fast entirely in assembly, it will use any extra memory it can find, (Mini-Memory supercart, even a Horizom Ram-disk)

TI-BASE

I've been reading reviews on this one and it looks like the answer to my and maybe your Data base problems. This is condensed from a review taken from a CPUG Harrisburg newsletter. If you read some of the newsletters we receive you would know more about this program and many others.

This program comes on two disks. The first is the actual program disk while the second is a TUTOR disk.

The (very well documented) manual instructs you to load the TUTOR disk. The TUTOR runs about 20 minutes or so depending on how much you pause.

TI-BASE is a very complex, inter-relational data-base. You can open up to 5 different data files at one time and move from one to the other with the SELECT command. You can create a COMMAND file with a DV80 text editor. You may have up to 17 fields, and 255 characters in each field. The optimun storage will allow 16,129 records in ond database. You can sort on any field, find any item, you have 12 mathematical functions including SQR, LOG, SIN, and ATAN, 5 BOOLEESAN and several LOGICAL commands. You can move data from one slot to another at your will, edit by record or paging.

This is just a few of the newer software items comming out for the orphaned TI, ve sure have come a long way, the TI is up there with the best it's not just a computer for games.

Next, some new Hardware.

NEWHARDWARE

There was a storehouse of new and old software at the show, I do not remember it all but here goes!

HORIZON PLUS

The HORIZON Ramdisk has been around for quite a while but now it's sporting a few new features. Now using the larger memory chips you can have a ONE MEG or even larger ramdisk. You do not have to start that big you can even add memory One Chip At A Time.

Another feature is the PHOENIX MOD , this lets the GENEVE use all the features of a ramdisk and even has a way to use the 16 bit bus for memory expantion.

Another new product from HORIZON is the P-GRAM CARD. This card you install in your Expansion Box and works similar to the GRAM-KACKER. The card has 72k of battery backed RAM, 40k GRAM for GROM emulating, 16k for Module RAM bank-switched, and 16k of DSR RAM also bank switched. It also has a optional CLOCK.

Cont. Page 4

Continued frome page 1

time! You can use Batch-It to customize an application program. I have written a batch file to make Rapid Copy do multiple copies just by pressing one key, complete with prompts when to change the copy disk, and an escape to return to normal Rapid Copy operation. I have not been able to get Batch-It to run behind Funnelweb yet, but I have talked to Chris Bobbit (Asgard) about it, so hopefully it is possible to work with Funnelweb because that could be very usefull. (It does run behind Ba-Writer).

As a word of encouragement: always remember that the Commodore 64 has an error message 'FORMULA TOO COMPLEX', meaning that you might have too many parentheses in a formula! Try to do that on our TI's.

Until next time .... later
Brad

Continued from page 3

DISK CONTROLLER UPGRADE KIT
By John Guion

This upgrade gives you some of the features of the Myarc or Corcom disk controllers at a lower cost buy requires changing chips in the TI card.

Some of the upgrades are, up to 4 drives, lower case names, faster head step and double side, but NOT Double Density.

Another kit from John is the RS232 UPGRADE KIT.

This kit reuires replaceing one chip. It lets you run sofrware writen for the TI thermal printer and output to PIO or RS232 in 60, 32, or 28 columns. The TP option even allows the user to write one-line screen dumps for Xbasic programs with graphics. "SIO" Either serial port may be set up to respoind to SIO. Plus other features.

A third kit from John is MULTI-MODE.

This kit is for owners of Triton's Super Extended Basic module. With this (no soldering required) chip change you get, Editor/Assembler, Disk Manager III, and TI-writer In addition to Super Extended Basic in one cartridge. You select any of these with just a couple of key presses.

Well thats enough for this month.

Jack Zawediuk

New Languages for the TI-99/4A

We have some new languages to tell you about. This excerpt comes from a download from CompuServe and originally appeared in the Northwest Ohio 99'ers News.

New Languages for the TI-99/4A
by Ted E. Bear, Pipedreams, Inc.

FIFTH-

This is a precise mathenatical language in which data types refer to quantities. The data types range:

from
CC,
OUNCE,
SHOT and
JIGGER
to
FIFTH (hence, the name of the language),
LITER,
MAGNUN and
BLOTTO.

Commands refer to ingredients such as:

CHABLIS,
CABERNET,
GIN,
VERMOUTH,
VODKA,
SCOTCH,
BOURBON,
COORS,
BUD and
WHAT_EVER IS_AROUND.

Rumor has it that this is the 99/4A owner's favorite of all the 99/4A languages and it has even been dumped into GRAMKRACKERS. You get

President Ira Lieberman
820-6332
Vice-pres Brad Snyder
826-4092
Secretary Lynne Weber
967-2131
Treasurer Brad Snyder
826-4092

Vol. VI, No. 8
Winter 1988

Ex-Editor Jack Zawediuk
821-1043

Next meeting: 7:30 PM,
January 16,1989
Conference Rooms A-D,
Sacred Heart Hospital
4th and Chew Streets
Allentown, Pennsylvania

LEHIGH 99'ER COMPUTER GROUP

V.P. COLUMN

Hello. I wasn't able to attend the November meeting, and neither was Ira, so Jack Zawediuk took over for the night. He has offered to help people set up their Funnelweb disks with their favorite utilities installed on the Funnelweb menus.

D.O.M.'S

Yes we still have some 'Disk Of the Month' disks available for $1.00 each. The current D.O.M.'s are sold for $1.50 a piece. Here is a list of the available back DOM's:

DISK NAME : JCT/INTRO

- ARCHIVER II by Barry Boone. A program to compress and archive disk files. Mainly used for telecommunication, but can also be used to save disk space when backing up disks.
- FRED. A graphics maze type game.
- FUCAT. A cataloger that idetifies what cartridge should be used to load a file.
- WOODSTOCK. A graphics cartoon-like Christmas card by Ray Kazmer.

DISK NAME : PLUS!SSSD1 & PLUS!SSSD2

This 'flippy' disk contains 57 files of use with TI-WRITER or equivalent. There are many templates to automatically set up TI-WRITER for letters, reports, to use various printer commands, etc. There are also programs to print out banners, calendars, gothic letters, print in multiple columns, etc. Included on the disk is extensive docs. There is also the 'MAX-RLE' program for viewing pictures.

DISKNAME : MUSIC

All files on this disk are outstanding music / graphics programs to run in extended basic.

- 12THSTREET
- BEER/POLKA
- LIDA/ROSE
- PENN/POLKA
- BOJANGLES
- HARRIGAN
- SHOW/ME
- THOSE/WERE

DISKNAME : FAST-TERM+

Includes the complete FAST-TERM program, with modifications by J. Peter Hoddie. This version has more disk managing functions.

There are also help files on the disk for telecomunication & TI-WRITER. And last a demo that shows the logo's for other computers and then destroys them.

DISKNAME : MISC.

The files on this disk are compressed and archived, so you will need ARCHIVERII to use this disk.

- !BREAKTHRU. The best version of the old arcade game of the same name that I have seen.
- !DIM. Disk Information Manager. Good disk cataloger that also stores additional information, that you enter, right on the disk. So when you catalog it again, you can see a description of the files on the disk. (or whatever you entered)
- !JPH/GAMES. Several XB and assembly games.
- !PRINTDV80. Prints out all display variable 80 files on a disk automatically.
- !TI-KEYS. A keyboard macro program. Press one key and have it enter whatever you have it programmed to enter.

UPCOMING DEMOS

In January, I will demo loading and saving programs to cassette for our new members with cassette only.

In February I will show Batch-It. A program that can control other programs. It can be used to add macros to other programs, or help screens, or make demos of program that run themselves without user intervention.

Any and all suggestions on what you want to see demoed are welcome! Just let me know.

ASSEMBLY MEETINGS

We are holding classes on assembly language programming once or twice a month at Jack's house. They are starting at 7:00 and ending about 9:00 - 9:30. If you are interested in joining, give me a call to find out when the next meeting is scheduled. This is a good time to join in, we are at a very basic level right now. You should have an Editor/Assembler manual for reference. To assemble and run programs you need the Editor/Assembler cartridge, or Funnelweb, or the Mini-Memory and the Line By Line assembler.

Later, Brad

MINUTES

Jack Zawediuk opened with a reading of the Treasures report. There was one new member and two renewals. New blank disks are in at $4.00 for ten, with lables.

Jerry Boyer reported on new aquisitions to the disk library; PR Base Ver.2, TI Rinner Ver.2, TRI Artist Slide Show, a Calendar (Appointments, ect) Program, Certificate 99 and Genial Traveler. The disk of the month was Music, available for $1.50, at the meetings.

Mike Bellew turned over the cassette library to Lynne Weber since he will not be attending meetings for a while due to other commitments. Which brought up the fact that we DO HAVE a large cassette libray for thoes of you who do not have a disk system.

Jack talked about Funnelweb and setting up disks for members with the programs they use most. Also discussed was the rudiments of a membership drive "Lets find those people with a TI 99/4A sitting in the closet and bring them out!"

Learning to Program in assembly language meetings have begun at Jacks house, anyone interested get in touch with him (821-1043). They're really only at ground zero-so don't be afraid to join, it's a learning experience for all.

A list of upcoming computer shows and fairs was made available, till you receive this news letter most will be over contact Jack for updates or pick up a copy of Computer Shopper at the newsstand look under coming events for shows.

A demonstration of a RAM DISK and time for questions and discussion among members closed out the meeting.

Lynne Weber

PERMANENT SCREEN COLOR CHANGE
Found in the TI*MES

```
100 CALL CLEAR
110 S=2 :: F=16 ! your choice
120 C=16*(F-1)+(S-1)
130 CALL INIT :: CALL LOAD(9
984,C,C,C,C,C,C,C,C,2,0,7,15
+0,4,32,32)
140 CALL LOAD(9999,40,2,0,0,
0,2,1,39,0,2,2,0,0,4,32,32,3
6,2,0,0,0,4)
150 CALL LOAD(10021,32,32,36
,2,0,8,16,4,32,32,36,2,0,8,2
4,4,32,32,36,4,91)
```

---CONTINUED PAGE 4

MY GENEALOGY ON MY TI

by John M. Geisinger,CDP
Lehigh 99'er Computer Group (Pa)

Our newsletter editor has been encouraging me to write an article on my genealogy efforts but I always say "I'll do it when I'm done." At the rate I'm going I'll never be done, so this is just a progress report.

My purpose in writing this is to stimulate your interest in genealogy and to use your TI for your records. As the proverbial pastor says "Do as I say, not as I do", don't necessarily do as I do, you may find a better way.

I have had an interest in genealogy for many years and have so many records I don't know what I have. I wanted to use my TI to help me know what records I have and where they are. Rather than having a well defined system and goal, my system has just "evolved" and I expect it will change as my knowledge of TI, files, data bases, equipment configuration and goals change.

There are two basic forms for collecting genealogical data. First is the "Pedigree Chart" which starts with one person, 2 parents, 4 grandparents etc. Once filled in I seldom refer to it. The second is the "Family Group Sheet". This contains data for each family unit or marriage. A second marriage requires a second FGS.

I have so many (1000+) family group sheets with many having the same given names that I don't know what I really have. This, along with my illegible penmanship, is why I went to my TI for help. There may be "accepted" numbering systems, some using letters to denote generation, some number each individual.

I was most fortunate to find a local county history and biography set of books which had my paternal immigrant ancestor, born in 1701, down to my dad listed. From this I made up a FGS on each descendant on plain notebook paper. On the left side I list the male info and on the right the female info, starting with their names, their parents names, dates, and personal data on each. Below this I listed their children, in sequence, with dates. Aside this I enter the child's spouse, if married.

In my system I number each FGS, not each individual. The immigrant is FGS #1. Each new FGS is given the next consecutive number, regardless of generation or relationship. Each chart contains three types of numbers: the FGS#; aside the father or mother their parent's FGS#; and for each married child his FGS#. It is these three numbers that tie the records together.

My genealogy class instructor stressed to always research from the known to the unknown, but I often research from the unknown to the unknown. Instead of gathering information on only known descendants, I am collecting any information on any Geisinger/ Geissinger/ Gisinger I can find. I am going both up and down the trees. This complicates both my manual and computer systems.

Many times the information is insufficient to definitely identify the person. In these cases I make up a FGS and number it "UN" for unnumbered or "NI" for not indexed, if I did not enter it in my manual index yet. If the information spans 3 generations and I can not tie it to the immigrant an alphabetic series such as A1, A2, A3... is used. When I can tie the A1 to the immigrant I renumber it to a numeric number. I file my sheets in numeric sequence in loose leaf binders.

Since I have so many duplicate names (often the same person) I want to print lists of all the Geis(ss)ingers in alphabetic sequence with a major sort on first name and a minor sort on date of birth.

To do this I needed a database which contains a multi-level sort on non-adjacent fields. The only such data base in our library at that time was Easy Data. As I become familiar with newer and better data bases I plan to write quick dirty programs to convert my files to a better data base.

THE GENEVE IS HERE , FINALLY .

Part 9
by Jerry Boyer

Hi everybody. This month I'll continue on setting up a hard dr. Last mo. we formatted the hard dr. and put the SYSTEM/SYS file in the first level or ROOT directory and we put LOAD/SYS in a subdirectory called DSK1, so that we could get the GENEVE to autoload from the hard dr. With the software that is available, the AUTOEXEC file must be on a disk in the physical drive #1, this just wouldn't do. What I did was to copy the SYSTEM/SYS to a new formatted disk and I got out a disk sector editor program. I then set it to ASCII mode and went to the search utility and I told the program to search for: "DSK1.AUTOEXEC" and when it got it I soon discovered that it appeared twice on the same sector. I then went to the ALTER utility and I changed the DSK1. to WDS1. at both places. I then saved the sector back to the disk and then resaved the complete file back to the hard drive. Now I loaded my AUTOEXEC file that I created with the M-DOS command "COPY CON AUTOEXEC" into the hard disk at the root level. This is my AUTOEXEC file, you can set yours up to suit yourself.

```
ECHO OFF
TIMODE
RAMDISK 90
ASSIGN E=DSK5:
SPOOL 16
COPY C:Z E:
E:
ECHO
ECHO MENU
ECHO ====
ECHO
ECHO ==============================
ECHO A.  M-DOS
ECHO B.  EXT. BAS.
ECHO C.  EDIT. ASSM.
ECHO D.  DISK MANAGER
ECHO E.  COPY DISKS
ECHO F.  FORMAT DISK2
ECHO G.  G.P.L. MODE
ECHO ==============================
ECHO <type "Z", space, selection>
```

What I did was to first set up a TI MODE with a 90k. RAMDISK as DSK5. & a printer spooler of 16k. Then it copied my "Z" macro file from DSK3 to DSK5 which is the RAMDISK. I did this to speed up the whole thing. The AUTOEXEC file now displays a menu that I only have to press a Z,(space),and my selection letter. What I put in my AUTOEXEC file was what I used the most often. The new posibilities this brings up are end less. This is my "Z" macro file, I also created with "COPY CON Z" in M-DOS. "AUTOEXEC & Z" work together

```
IF %1==A ECHO M-DOS    GO AHEAD
IF %1==B WDS1.GPL WDS1.XB
IF %1==C WDS1.GPL WDS1.EA
IF %1==D WDS1.GPL WDS1.MDMV
IF %1==E DISKCOPY A: B:
IF %1==F FORMAT B: /V
IF %1==G WDS1.GPL
```

I keep my "Z" file in DSK3 which is an 80 track DS/DD 3 1/2" diskdrive (a fast and a quiet drive). These 2 files (AUTOEXEC & Z) are changed a couple of times a week as I think up new things for them to do. The way I have it now seems to work out well for me. I'm just trying to get you to see what's possible so you'll be able to set yours up your way.

I then transfered the: GPL files; Extended Basic files; Editor Assm. files; Disk Manager V files to the Root Directory Level. They load up much faster from the First level. Each level can have 127 files and 114 subdirectories. I have a sub directory named DSK1 for all of the programs that must load from Disk Drive 1 and a sub-directory named DSK for the ones that look for a disk name instead of a number. In DSK, I have subs named TIMP for my Multiplan files and MYWORD for my Myword program files and a sub that I called MOVIES for the movie files that I create with the Myword prog. To get to my MOVIELIST, I have to type WDS1.DSK.MYWORD.MOVIES. Well my page is full for this month. Bye.

THE NEW FEATURES OF FUNNELWEB v4.1
by Charles Good Lima Ohio User Group

FUNNELWEB is probably the most significant software ever for the 99/4a. After booting FUNNELWEB v4.1 from XBASIC (you can boot FWB from any assembly language loader, but the XBASIC module is the best way) you can do all of the following without changing modules:

1. With a single keypress you can load from a selection of user created menus almost any software ever written for the 99/4A. If the software you want to load isn't configured into one of your user created software menus, you can call up a disk directory anywhere within FWB, mark the file name of software seen in the directory, and then load that software.

2. Do worD processing with a much improved version of TI-Writer.

3. Create assembly source code and then assemble it as you would with the E/A module.

4. Manage disks with a modified version of DM1000 which is supplied with the FWB package. Pre-configured menu entry points for other common disk managers are also provided.

5. View and edit disk sectors with a modified version or DISK PATCH, also sometimes known as DISKO.

This review will describe the changes and additions in v4.1 as compared to v4.0. Although this description is based on the May 30, 1988 release which says "Memorial Day" on the XBASIC title screen, the review should be valid for all subsequent releases of v4.1.

Enhanced CENTRAL MENU capabilities.

Each central menu now has 8 items, and items 4-7 are completely configurable to load any kind of assembly language file. This includes autostarting D/F80 source code and assembly PROGRAM files. In previous versions of FWB the central menus could only load PROGRAM files and only a limited number of central menu slots were configurable.

The TI-Writer menu reads as follows:

```
1  EDITOR
2  FORMATTER
3  DISK UTILITIES
4  MODEM
5  DATA BASE
6  DM1000
7  DSKU
8  USER LIST
```

As noted above, items 4-7 can be configured to suit the user. MODEM is an entry point for terminal emulation software such as FAST TERM or TELCO. DSKU refers to John Birdwell's "DISK UTILTITIES." This fairware disk manager/sector editor is so good that some former users of DM1000, myself included, have switched to DSKU for most disk management uses. DSKU is not provided as part of the FWB package, but can be obtained directly from John Birdwell or from most user group libraries.

Item 3 in the above TIW central menu leads to a specially created user list menu in which disk management software is grouped together. The DISK UTILITIES menu reads as follows:

```
1  DM1000
2  DSKU
3  MYARC DM
4  DPATCH
5  SCREAMER
6  TRACKER
7  ARCHIVER
8  CONFIGURE
9  <CTR ROM>
```

DPATCH is the modified sector editor DISKO which is provided as part of the FWB package. SCREAMER is a good entry point for an ultra fast whole disk copier such as REDISKIT or TURBO COPY. TRACKER can be used to load one of the various "copies anything including protected disks" track copiers. Will McGovern, one of the FWB authors, has written a fairware track copier called TRACKER that is one of the few (maybe the only) that works with a Myarc disk controller. Send him a few bucks in Australia and he will send it to you, or look in your user group library. ARCHIVER will load the latest version of Barry

Boone's archiving/compressing program. This archiving software is not part of the FWB package. CONFIGURE boots the FWB configuration files CF/CG. Items 1-8 in the above DISK UTILITIES user list menu can be altered with CF/CG to boot any assembly D/F80 (autostarting or not) or PROGRAM files.

The Edit/Assm central looks this way as configured on the FWB distribution disk:

1 EDITOR
2 ASSEMBLER
3 LOADERS
4 C-COMPILER
5 DISK PATCH
6 LINEHUNTER
7 ..
8 RESET

Item 4 loads the latest v4 release of c99. LOADERS, unchanged from FWB v4.0, leads to a menu for loading assembly D/F80 or PROGRAM files that aren't already configured into one of the FWB user lists. LINEHUNTER is new to v4.1. It is an assembly programming utility that prints on the screen any specified line of assembly D/V80 source code. You can also type the name of a label, and LINEHUNTER will display lines that have that label.

THE CONFIGURATION PROGRAM, FILES CF/CG:

This has been totally redone for v4.1 and MUST be used to do any configuring of the various user lists. It is no longer possible to directly edit FWB's XBASIC LOAD program to alter the XBASIC user list because there is very little XBASIC code in LOAD. There are only a few XBASIC line numbers in LOAD and the rest of LOAD is all in assembly.

CONFIGURE is much easier to use in v4.1 than it was in v4.0. CF/CG has a tree structure which allows you to quickly get to any part of the configuration without redoing the entire configuration process. The configuration program is very professional looking with sound effects, overlaping menu windows that pop into view, and help screens that are available at various points in the configuration process by pressing "?". Obviously much effort went into the preparation of the new v4.1 configuration files. The authors note that CF/CG was condensed from over 500 sectors of source code.

Basically what you do is load a configuration data file, alter the configuration, resave the altered data file to disk, and then while it is still in memory install the new configuration data into the FWB LOAD and UTIL1 files. If you later obtain a more recent release of FWB v4.1 you can configure the more recent release simply by loading your old configuration data file and installing this data into the newer LOAD and UTIL1 files. This is REALLY EASY! I had access to a pre-release beta testing edition of FWB v4.1 and was able to use the beta testing edition's configuration data file to configure my "Memorial Day" v4.1 in about 30 seconds. Hopefully the FWB authors will maintain this configuration data file system in all future upgrades of FWB and allow this easy transfer of v4.1 configuration information into all future FWB updates. Unfortunately, configuration information from v4.0 cannot be transferred directly to v4.1.

The important universal keys to remember in the configuration routine are <ENTER> to advance to the next window, and BACK to return to the previous part of the program (often the previous window). AID will get you a disk directory from most places in the program, and "?" brings up the help screens. When a window is displayed, you press the first letter of the text line in the window to perform the function indicated. If the window says ______

```
 ^|Load|
 ^|Edit|
  |Save|
```

then you press "L", "E", or "S". Sometimes "N" and "B" are used to move the cursor up and down within a window when the functions "Next" and "Back" are displayed in the previous window. When more than one window is

visible at the same time, the active window is indicated by a fat (2 pixel wide) border. The borders of the windows get alternately fat and skinny depending upon which is the currently active window.

The early windows in the configuration process are as follows:

```
______________________________________________
SysInfo      Quit         Install | FIRST

                         __LOADING WINDOW___
______  _________        ___________________
|Load|  |Loading|        |Boot Tracking OFF|
|Edit|  |Devices|        |TI-Writer side 1 |
|Save|  |Colors |        |Edit/Assm side 1 |
SECOND  |Menu   |        |Working Drive  2 |
WINDOW  |XB List|
        |UL List|
                  _______________
    THIRD         |Edtr Printer|
    WINDOW        |Fmtr Printer|DEVICE
                  |Object File |WINDOW
                  |Work^File   |
                  |Program     |

                         ______
                         |Edit|
                         |Next|COLOR
                         |Back|WINDOW
                         |Xchg|
                         |Redo|
                         |View|

                    ___________________
                    |TI-Writer^side|  MENU
                    |Edit/Assm^side| WINDOW
```

You start out by pressing S(ysinfo) to display the second window, and then press L(oad) to load the configuration data file, following the prompts for loading. The name of this file on the FWB distribution disk is SYSCON, but you can use any name. You can create different FWB configurations on different configuration data files each with a different file name. After loading the configuration file, you press E(dit) change the the configuration data and display the third window. From here you bring up fourth series of windows where much of the configuration actually occurs.

LOADING WINDOW: Boot Tracking toggles between ON and OFF by pressing "B". Usually it is left ON unless FWB is installed on a ramdisk. In that case, the authors suggest it is sometimes best to leave boot tracking OFF. If the files loaded by FWB's central menus are in different drives (or ramdisks) these drive numbers are specified next to "TI-Writer side" and "Edit/Assm side". The "Working Drive" is the default drive number that appears after "DSK" if the mailbox workfile name is empty when you LF from the editor or use the Formatter.

DEVICES WINDOW: Printer names are self explanatory. "Object File" name is the default that appears on the screen next to DSKx. when you select LOADERS from the Edit/Assm central menu and try to load a D/F80 assembly file. I have "Object File" name configured as "DF/80FILE" to remind me that only this type of file can be loaded from certain parts of LOADERS. "Work File" is the default file name used the first time you LF. It is best to leave this blank as it is on the distribution disk. If "Work File" is left blank, you can exit FWB, go through the title screen and do something in BASIC without turning off the PE box, return to FWB and find the previous workfile name still there when you LF. "Program" is the default displayed on the screen when you use the LOADERS menu to load assembly PROGRAM files. The ability to set "Object File" and "Program" defaults is a feature new to FWB v4.1.

COLOR WINDOW: Here you can edit and view your choice of 10 color combinations. You can alternate between two sets of such combinations. The one listed at the top of the list is the combination that appears first when you boot FWB.

MENU WINDOW: This choice allows you to configure items 4-7 in each of the central menus. You can configure autostarting D/F80 assembly files or assembly PROGRAM files. Each file name is limited to two characters and should either be on the boot disk in the either one of the two central menu drive numbers specified above in the LOADING MENU. Press BACK to exit this part of the configuration process.

XB LIST AND UL LIST CONFIGURATION:

These are both done in a similar manner. First F(etch) the list by pressing "F". Then press N(ext) or B(ack) to select the item to be configured and press E(dit) to change that item. Press <ENTER> to go from menu to menu in the editing process. When asked for the "Secondary" this refers to the drive number specified in the devices window for the E/A central menu files. If you ask for a "Reminder", FWB will display the message INSERT UTILITY DISK when you attempt to boot the configured program from a FWB menu. When XB List or UL List configuration is finished (and S(aved) in the case of UL List), press BACK to return to the third window.

THE FINAL CONFIGURATION STEPS: Press BACK several times to return to the second window and then press S(ave) to save the modified SYSCON configuration data file back to disk for later use. Then press BACK, and from the first window press I(nstall) to install the configuration data into the LOAD and UTIL1 files. Follow the prompts. An alternate name for the UTIL1 file is FW and you can use this name is you want. The alternate name used to be RELOAD in earlier versions of FWB, but this name is too long to use with current Horizon Ramdisk Menu software. It is necessary to save the configuration data to BOTH the LOAD and the FW/UTIL1 files, so cycle through the installation process twice. Then press BACK a couple of times to return to the first window and press Q(uit) to return to FWB. If you exit configuration with Q(uit) you will not immediately see your new configurations. It is necessary to reboot FWB from the beginning for the new configurations to appear on screen.

UL LIST SPECIAL CONSIDERATIONS: Immediately after configuring a USER LIST and before pressing BACK to return to the third window it is necessary to S(ave) the configuration to the USER LIST, since this user list data is NOT saved as part of the configuration data file. When you return to the third window your USER LIST data may be lost. You may create as many USER LISTS as you want, each under different names. These lists can be loaded from each other, or they can be loaded from the central menus. DISK UTILITIES from the TI-Writer central menu is a special user list file named DS, and can be configured from the "UL List" option of the third configuration window. If you come across a more recent release of FWB v4.1 you can use your previously configured user lists (files UL, DS, and any of your own user list files) unmodified with the more recent release. You don't have to configure your user lists all over again. I hope it will be possible to use unmodified v4.1 user lists in future versions of FWB (v4.2 etc) as well. Unfortunately the FWB authors state that v4.0 and earlier user lists are not guaranteed to be compatable with v4.1.

NEW FEATURES IN QUICK DIRECTORY:

You can now mark ANY file in QUICK DIRECTORY, invoked by AID from most places in FWB. If the marked file reads PROGRAM, then its name will appear on screen as the default when you load an assembly language PROGRAM file from items 1-3 of the LOADERS menu. If the marked file is D/F80, then it will show up on screen as the default when you load assembly object code from items 4-7 of the LOADERS menu. The ability to mark files from QUICK DIRECTORY for the LOADERS menu is new to v4.1. Any file may be marked for deletion, and after deletion the sector count and file name list displayed on screen by QUICK DIRECTORY are immediately updated. The ability to delete from QD and immediate updating in both QD and SD (from the editor) are new to v4.1. I consider all the new features described in this paragraph to be very useful.

Other changes in QD include the ability to unmark a workfile name as O(ld) and revert back to the previous workfile.name. In v4.0 you could only do this by using SD from the editor. The N(ext) and B(ack) keys are now used to page through the alphabetical list of file names in QD rather than SHIFT/CTRL as in v4.0. This change makes QD consistant with

other sections of FWB v4.1 since "N" and "B" are commonly used to move foreward or backward, particularly in configuration.

DM1000 CHANGES:

The FWB authors include their own modifications of DM1000 v3.5 as part of the FWB package. (PLEASE NOTE: DM1000 is fairware, and if you use FWB you should not only send a fairware donation to the FWB authors, you should ALSO send a fairware donation to the Ottawa User Group for the use of DM1000.) V3.5 is the last source code sent directly to the FWB authors by the Ottawa UG and this is why the FWB authors have based their modifications on this rather than a later version. FWB co-author Tony McGovern writes me that he beleives his modified v3.5 will do everything that DM1000 v4.0 will do except line by line scrolling with V(iew). Tony has given DM1000 the squeeze job, and the result is that FWB's modified v3.5 files are smaller than the original v3.5 and much smaller than DM1000 v4.0.

The most important feature of FWB's DM1000 is that it formats disks at 18 sectors per track in DD mode with a Myarc disk controller. Bugs in T(ype), P(rint), and C(opy) have been fixed, and you can now use 3 digit printer entry codes. Horizon Ramdisks at high CRU adresses are fully supported except for initialization during Disk Copy.

OTHER FEATURES NEW TO FWB v4.1:

The formatter may now have 4 disk files open at the same time. From the assembler the object file name is passed back to the object file parts of the LOADERS menu and appears as the on screen default for immediate loading.

The keyboard control of DISK PATCH has been augmented to make it consistant with John Birdwell's DISK UTILITIES. The DISK PATCH title screen tells you that you can use the "original" keys (this means the keys that worked with FWB v4.0, which arn't quite the original DISKO keys) and an alternate set of key presses that corresponds to the keys used to control DISK UTILITIES (Ctrl/H for Hex display, Ctrl/N for next sector, Ctrl/W for write to sector, etc). If you are familiar with the keyboard controls of DISK UTILITIES you will have no trouble using FWB's DISK PATCH.

FINAL CONCLUSIONS:

In my opinion everyone who does serious disk based work with the 99/4A should be using FWB. If you don't have it, check your user group's library. User groups, not individuals, may obtain FUNNELWEB v4.1 at no charge by sending a disk and paid return mailer to the Lima User Group, P.O. Box 647, Venedocia OH 45894.

****************Eof************************

COLISTER A TINYGRAM by Ed Machonis

Not just another 28 column lister! "COLISTER has a couple features not available in other programs. First, it prints a blank line between program lines...makes it easier to read. Second, it TABs the output 6 spaces (centering the program in an otherwise 40 Col listing). COLISTER does not require a program be resequenced! And COLISTER will print to either disk or printer. Don't print the listing through the formatter unless you have modified your Formatter to ignore the special format command characters that are also found in programs. To use COLISTER:

```
1 !  *** COLISTER ***
 A Tinygram by Ed Machonis
   99-99ers, Bayside, NY

2 PRINT :"1st LIST your prog
ra to diskThen RUN COLISTER"

3 PRINT :::"INPUT FILENAME?
ex:DSKn.LIST" :: "INPUT F$ ::
 INPUT "OUTPUT FILENMAE? ex:
 PIO OR DSKn.LIST2" :":P$

4 OPEN #1:F$,INPUT :: OPEN #
3:P$,OUTPUT :: ON ERROR 10

5 C=28 :: LINPUT #1:A$ :: IF
 LEN(A$)<80 THEN 8

6 LINPUT #1:B$ :: IF VAL(SEG
$(A$,1,POS(A$," ",2)))<VAL(S
EG$(B$,1,POS(B$," ",2)))THEN
 F=1 :: GOTO 8

7 A$=A$&B$ :: IN LEN(B$)>=80
 THEN 6

8 A=LEN(A$):: L=A/C+.99 :: F
OR I=0 TO L :: PRINT #3:TAB(
6):SEG$(A$,1+I*C,C):: NEXT I
 :: IF EOF(1)AND F=0 THEN CL
OSE #1 :: CLOSE #3 :: END

9 IF F=1 THEN F=0 :: A$=""
 : GOTO 7 ELSE 5

10 ON ERROR 10 :: RETURN 7
```

(Thanx Ed Machonis and the 99 MONITOR).

EXTENDED BASIC - STILL A GOOD CHOICE!

The case for the BASIC Language!

By ART BYERS

Part One: Introduction.

There are new guys in the 99/4A neighborhood. Among them are such stars as FORTRAN, FORTH, PILOT and SMALL C. They have lots of adherants who talk about "Like Basic" (FORTRAN), "Freedom and Exceptionally flexible" (FORTH),"Simplicity" (PILOT), and "Speed and structure" ('c'). They are Compiled languages which means they certainly run much faster than old friend XBasic. SOOOoooo? Why bother with Extended Basic at all? Why not go with the New? The Better? The Faster?

One of the great things about our beloved 99/4A is that even with its limited memory, it CAN support FORTH and C and PILOT. I consider any of the computer languages that will accomplish what is needed to be fine! For me, however, Extended Basic still remains the EASIEST and BEST, most especially when coupled with Assembly Language subroutines that speed up often used important areas.

Let me try to lead you through a discussion of the pros and cons of Extended Basic without "putting down", in the slightest ANY other language for the 99/4A (including Pascal -However Pascal requires a special PEB card and those are hard to find and some early versions have bugs).

Extended Basic has many advantages from a programmer's viewpoint, not the least of which is that it is an interpreted language with a plethora of error deguqqing routines built in. One of the real swift pains in the neck of a compiled language is that if it is compiled containing errors or bugs, these are extremely difficult to find. This does not mean they cannot be found or that good programmers cannot produce error free compiled code. It is just that debugging, adding to, subtracting from, changing code, etc. is much easier with XB. It is a shame that TI chose to make XB a "double" interpreted language by writing it in GPL, TI's "secret" proprietary language, also interpreted, (Which to the best of my knowledge TI has NEVER released and should they have chosen to take legal action, they could make trouble for those who have violated their rights by selling GPL programs, books explaining GPL, etc. and etc.). It would have been better if the interpreter had been written in Assembly a la MYARC's XB. The added speed of MYARC's XB is a big improvement over TI's IBasic. However, The whole subject of execution speed will be covered in more detail in part 3 of this series. It deserves separate discussion because this area is what is most often raised in any and all debates on the merits of TI XB.

One of the biggest advantages of XB is its EASE OF USE AND UNDERSTANDING. BASIC itself was written just for that purpose. BASIC is suppled with such popular computers as Apple, Atari, Commodore, and IBM. This ease of use was most important in bringing better understanding of computers and use of computer languages to large numbers of Americans. For no other reason, the Basic language continues to survive.

As far as the 99/4A goes, another advantage is that the language itself resides outside the RAM areas. It is in ROM and GROM. The cover of the XB manual states that the module contains "32k bytes of preprogrammed memory". Most of the RAM is free. Additionaly, XB accesses, again with simplicity, clarity and ease, the built in ROM routines such as Device Service - printers, cassette, disk drives -, screen access and display, setting up of buffers, graphics and sprites, mathematics, etc. Many of the "new" languages save RAM memory by also accessing these same ROM routines, running at the same speed for all!

Now lets talk about available memory. Because support for Forth and 'c', for examples, must be loaded into the main 32k memory area, they do not have as much memory available as some programmers feel is absolutely necessary. This problem has been solved by using virtual memory - that is disk storage of Forth screens (blocks) or C support routines. XB support resides in console ROM and the module itself, the full 24k upper RAM is available for programs and the 8k low memory for Assembly support routines, and most of VDP RAM for string storage etc. For example, I recently purchased a Disassembler which was written in Forth. The

author plainly stated that because of the memory used by Forth itself plus the program, it was not feasible to dissassembly programs from RAM. It did its dissassembly right off the disk!. Since Basic resides in ROM, a dissassembler written for E/A or MM modules can be written in plain old BASIC, and can disassemble programs that use the 24k upper and 8k lower memory, because it resides in VDP RAM, and not overwrite the program.

Some last points! let us look at what we have to work with. We have a machine designed as a HOME computer. For almost every purpose or use at home, memory and speed available through XB are more than sufficient. We are not tracking satellites, doing high order lengthy math, searching a database the size of the national Social Security register. We have a hundred or so names on our phone list. We do not require massive spread sheets. For our normal practical purposes XB and the 99/4A can suit our needs. In fact I may be accused of HERESY, but I did almost everything with only the XB module and cassette - NO memory expansion or disk!!!

What is more, when I need a special program written to fill a personal need, I write it, dubug it and am using it in a matter of a few minutes to at most an hour. This is possible because the most frequently used XB GOSUB routines and CALL SUBs are saved on disk as MERGE files ready to be placed into a program, easily and quickly. Many programmers overlook this useful feature of XB.

The following articles in the series offer concrete evidence to backup the ideas expounded above. They are NOT a tutorial in Basic programming. Rather, they will place a point of view before you as food for thought that, hopefully, will lead to a return to some good Basic programming.

The Case for Extended Basic Part 2

By Art Byers

EASE OF PROGRAMMING
EASE OF UNDERSTANDING
Basic Wins Hands down!!

At the start of many Assembly tutorials, authors often give examples of the same program in both Basic and Assembly. Usually, the Basic is one or two lines, but the Assembly is a screen or more. The author then points out, with pride, that the assembly code, when compiled, takes up much less RAM than does the Basic.

I suppose my reaction was not what those authors expected. I saw the programming of one line vs programming more than 20 lines. Then I remembered that in over four years of programming with Extended Basic, I have NEVER run out of memory. Basic knocks Assembly out of the ball park when it comes to ease of programming, and if you write good Basic code, you have enough RAM in the expanded 99/4A for every HOME USE. (REMEMBER my premise is that this is an excellent HOME computer).

There are other very good reasons for having assembly coupled to your Basic programs and I will cover these in another chapter. Now let me offer proof of the pudding with a specific example. The 9900 Assembly code, listed below, converts a decimal number into an Integer number (ie: converts 3.1416 [PI] to 3). The code is from M. S. Morley's book "Fundamentals of TI-99/4A Assembly Language", pages 119 and 120, and is written for the Mini Memory.

What the code does is to read an ASCII encoded multidigit number and replace the decimal point and following digits with blanks. If no decimal point is found, then the number remains unchanged.

```
        AORG >7DB4
        TEXT '3.1416'
        DATA >0D00
M1      DATA >7DB4
        LWPI >70B8
        LI   R0,>2E00
        LI   R1,>2000
        LI   R2,>0D00
        MOV  M1,R3
             M1,R3
             M1,R3
             M1,R3
J1      CB   *R3,R1
        JEQ  J3
        CB   *R3,R2
        JEQ  J3
        CB   *R3+,R0
        JNE  J1
        DEC  R3
J2      MOVB R1,*R3+
        CB   *R3,R1
        JEQ  J3
        CB   *R3,R2
```

```
       JNE  J2
 J3    B    *R11
       END
```

Remember that all this code does not even take the result and print it out on the screen or a printer - You will need five more lines of code to write to the screen and more than double that to run it out to your printer!!

Here is the same thing done in Basic INCLUDING placing it on the screen:

```
PRINT INT(3.1416)
```

Now Come on!! Which would you rather sit down and program?? I rest my case on this point. There are very good reasons for higher languages and one of them is ease of writing source code.

At this point, you may accuse me of false-weighting the scale. You may say that FORTH, PILOT, FORTRAN and 'c' are equally as easy if you want to print the integer of PI to the screen. Yes, that is true - BUT - Basic does it in English, and because PRINT and INT are reserved words represented by tokens, does it with a minimum of memory.

In general you would be correct. Nonetheless, it is in ease of understanding, for the average home computer programmer, and therefore ease of programming, that Basic shows its advantage best.

Part 3 will examine Extended Basic's weakest point and Assembly's (and some other languages') strongest point - Speed of execution!

THE CASE FOR EXTENDED BASIC, Part 3

by Art Byers

Speed of execution, the weak link?

While we could compare Basic to 'c' or FORTRAN, because this chapter is about speed, let's stay pretty much with the acknowledged racing champ, Assembly, for comparison.

To begin, let's admit straight out that where TI Extended Basic fails is in speed of execution. When Monty Schmidt wrote his well known and popular TECHIE BBS program (in Extended Basic), he had to CALL LINK to assembly routines to do such things as reading the pin input/output of the RS232 to send and receive data via a modem. TI's double interpreted Basic simply could not hack it there!! But note, Monty was able to write most of the BBS program in XB.

Let's go back to some fundementals. As in any computer, all languages eventually end up being put into machine language in order to be run. For the 9900, the fastest language and one that can do anything and everything that is possible with the computer is, naturally, Assembly - the instruction set of the chip itself. The compiled languages, 'c' and FORTRAN, (as well as FORTH) allow the use, where the programmer desires, of Assembly language routines. The reason these Assy routines are used is that the compilers do not allow full access to the instruction set, but only use a limited part. This restricts them from equaling true assembly speed, nor is the compiled code written in such a way as to be congruent with the most efficient way of writing Assembly. Where maximum speed is needed, XBasic can also use Assembly routines as shown by the above example. In this respect it is the equal of the compiled languages.

The areas where speed is vital, such as lengthy math calculations, sorting a large data bases, big recursive loops, global searching, etc. are the areas where ALL the languages of the 99/4A most often tie in to Assembly routines.
Where blinding lightning speed is required, Assembly programers will write their own Device Service Routines, often accessing several chips, such as the 9918 (Video display) or the chips in the RS232 card, directly and bypassing the ROM/GROM routines many of which are written in GPL. More often, however, most programmers prefer to access the same ROM/GROM routines to save writing code because they don't require the extra speed.

You may now point out, and with justification, that if I stress the ease of understanding and simplicity of XB, I am ignoring the difficulty of writing Assy routines. Have I just shot my own arguments full of holes??

How can the average XB programmers (as apposed to advanced programmers in general) be expected to write Assembly code.

The answer is that they are not expected to write it, no more than they were expected to write the code in the XB module itself. Today, there are many sources of routines that can be loaded into memory and accessed via "CALL LINK". The programing group led by Barry Traver of Philadelphia has not only put together these routines under the catch name of XXB,(Extended, Extended Basic) but they have devised a way of putting them into memory image (program) form to be loaded simultaneously into memory along with any XB program you have written. Several other Commercial and Fairware authors offer similar material. In the critical areas, XB can use assembly routines to bring its speed to completely acceptable levels.

This business of speed, in general, I consider to be highly over rated. For most purposes Extended basic is more than fast enough. Fortran, c, or Assembly may pop a full screen of text into view in less than half a second - but what matter if it takes a few minutes for the user to read the text. XB can fill a screen with text in less than one and a half seconds. Surely that is fast enough.

When you realize that most programs, no matter what the source language, spend much time WAITING for things to happen - ie: a user to make an entry, a disk to be searched, or a printer to finish printing, you must then consider that more often than not, XB is easily fast enough to accomplish its purposes. In most cases, what you are really talking about is APPARENT speed. Anyone who has searched a commercial data base at $1.00 per minute, using an all machine language terminal program, will tell you that the 30 seconds it takes to dump the text buffer to floppy disk can seem like half an hour, while imagining money pouring out of their pockets into the service's coffers!!

Careful planning and proper XB programming can help to minimize the "waits" in an Extended Basic program, or any other language programming for that matter. However, often these delays are out of control of the program. A printer buffer, a RAM disk or hard disk will materially speed up any language on any computer.

To sum up, I sincerely believe XB, even TI BASIC, is more than adequately fast for practically all the programs the average user will write for a home computer, which is what the 99/4A was meant to be. Where it can't fill the required bill of speed, a link to a faster language is certainly allowable and available.

Part 4 will discuss accessing the support built into the XB module that are often more difficult to duplicate in other languages.

The Case for Extended Basic: Part 4

By Art Byers

Support Built Into The System.

As was mentioned in the introductory article, support for XB resides in the System and does not have to be loaded into RAM. More than half of it is in the plug-in XB module itself -some 32k in four banks of 8k. The console, however, also contains a tremendous amount of powerful support that is used by both TI BASIC and XB.

Part and parcel of this support are 106 commands, functions, and statements, all easily understood as plain English words. In addition some of these have multi sub-commands available. For example, ACCEPT has the following: AT(,,) VALIDATE, SIZE, DIGIT, UALPHA, and NUMERIC. All these "extras" are powerful added options - and, again, I stress, written in plain easily understood English. Others having additional modifying options are CALL COINC, CALL COLOR, CALL DELSPRITE, DISPLAY, CALL DISTANCE, ON BREAK, ON ERROR, RETURN, RUN, and SAVE. That 106 expands to over 125 with the added options.

Additionally, 37 standard error messages are in the interpreter to aid debugging. Floating point and scientific math are built into both the immediate mode as well as the program mode. Math abilities include PI, LOG, SQR, SIN, COSIN,ATN AND TAN. With the use of the powerful DEF, any math function from secant to inverse hyperbolic cotangent can be acessed with one line of code.

Files can be OPENed and CLOSEd in the immediate mode - all types: printers, disk drives, serial port included. Six relational expressions are recognized. (<,>,<=,>=,<>,=). A full range of Boolean logic is on board including IF, THEN, ELSE, OR, NOR, and XOR.

I could go on and on and on and on - There are reserved words such as STEP TO, APPEND etc and etc. I have made my points. There is nothing to be gained by pounding them home further.

All the above take up not one single byte of RAM nor must they be accessed from disk during a program's run or preloaded from disk to enable a program to run.

Now one of the biggest complaints by programmers about the Basic language, in general, is that the frequent use of GOSUB and GOTO make it impossible to follow the flow of long and complicated programs. Also, the complaint is that it leads to unstructured "spaghetti code". TI XB has provision for writing very structured code through the means of user written CALL SUB programms. Pascal and C are known as structured languages, in this respect they have nothing on an XB program that properly uses this outstanding built in feature. By proper use of meaningful names, it becomes very easy to follow program flow. You know immediately what is happening when you read CALL CHIMES or CALL MENU1(CHOICE)

What is just as important is that when you have built up a library of useful CALL SUBprograms, you can save them in MERGE file format and use them over and over in any new programs you write. This saves considerable programing time and effort.

To put it succinctly, This computer was designed, primarily, with BASIC and XBasic as the home user's language. That is why it is so easy to access all the goodies from sprites to sound from simple single lines of basic programming.
*******************EOF*****************

TWO TIPS
by Charles Good LIMO OHIO 99 UG

DISK UTILITIES v4.1 TIP

Printing out a "DISK REPORT" complete with comments takes alot of paper, even if you use compressed print. If the disk has alot of files the DISK REPORT will probably not fit onto the front of the disk envelope. Using "System Setup" and the following special characters will print your disk report in compressed, subscript, double strike, and small line feeds. From the "Printer Setup" submenu type "S" next to "Disk Report" and then enter the following "Special Character" code:
1B0F1B53011B330B

You can permanently enter this special character with a sector editor so that it will always be available for instant use. You need the complete documentation, only available to registered DISK UTILITIES owners to tell you where to do the sector editing. A sample commented Funnelweb v4.1 system disk report is shown below printed with this special character codes. The code above works for epson compatable printers.

FUNNELWEB v4.1 TIP

Do you have FWB v4.1 stored on a ramdisk designated other than DSK1. If so, when you call up FWB's DM1000 you may have to wait for physical drive #1 to grind away for awhile before DM1000 appears. This access of drive 1 every time you boot FWB's DM1000 can be eliminated. Use FWB's DISK PATCH, or any other sector editor, to display the first sector of the MG file. Make the display ASCII, and near the beginning of the sector you will see "DSK1.MG". Change this drive number to the ramdisk drive number that actually contains file MG and your problem is solved. Now, when you select DM1000 from FWB v4.1 it boots instantly!
**********************EOF******************

Would you like to hook up an 80 column,
display terminal to your TI and have it work
too? Try hooking one up through the RS-232
Card - instructions are in the RS-232 manual.
Then to access it through Extended Basic, type
in:

```
 LINPUT #1:A$
 PRINT #1:A$
```

GETTING THE MOST FROM YOUR CASSETTE SYSTEM
BY MICKEY SCHMITT
NUMBER 1
GETTING STARTED

BEFORE YOU TRY TO DO ANYTHING WITH A CASSETTE SYSTEM YOU NEED TO START WITH THE RIGHT EQUIPMENT. THERE ARE MANY DIFFERENT MODELS OF STANDARD CASSETTE RECORDERS AVAILABLE WHICH WILL WORK WITH YOUR TI COMPUTER. (BESIDES THE OFFICIAL TI PROGRAM RECORDER). HOWEVER, FOR BEST OPERATION AND ALOT LESS AGGRAVATION, YOU SHOULD USE A CASSETTE RECORDER WITH THE FOLLOWING FEATURES:

1. VOLUME CONTROL: FOR BEST RESULTS THIS SHOULD BE SET BETWEEN MID-RANGE AND MAXIMUM SETTINGS.
2. TONE CONTROL: FOR BEST RESULTS THIS SHOULD ALSO BE SET BETWEEN MID-RANGE AND MAXIMUM SETTINGS.
3. MICROPHONE JACK: THIS JACK IS NEEDED IN ORDER TO RECEIVE INFORMATION FROM YOUR COMPUTER.
4. EARPHONE OR EXTERNAL SPEAKER JACK: THIS JACK IS NEEDED IN ORDER TO SEND INFORMATION TO YOUR COMPUTER.
5. REMOTE JACK: THIS JACK MAKES IT POSSIBLE FOR YOUR COMPUTER TO CONTROL YOUR CASSETTE RECORDER'S DRIVE MOTOR - THUS YOUR TAPE RECORDER WILL RUN BY PRESSING THE 'ENTER' KEY ON YOUR COMPUTER CONSOLE.
6. DIGITAL TAPE COUNTER: THIS IS A VERY IMPORTANT FEATURE AS IT WILL SAVE YOU ALOT OF UNNECESSARY AGGRAVATION. THIS FEATURE ENABLES YOU TO EASILY LOCATE THE CORRECT TAPE POSITION OF YOUR PROGRAM OR DATA FILE. THIS IS ESPECIALLY USEFUL WHEN YOU WANT TO STORE MORE THAN ONE PROGRAM ON THE SAME SIDE OF THE CASSETTE TAPE.

NExT, YOU WILL NEED TO HAVE THE TI CASSETTE INTERFACE CABLE WHICH IS USED TO CONNECT YOUR RECORDER TO YOUR COMPUTER. ALTHOUGH, THIS CABLE COMES WITH THE OFFICIAL TI PROGRAM RECORDER, IT MUST BE PURCHASED SEPARATELY IF YOU ARE USING ANOTHER TYPE OF CASSETTE RECORDER. IF YOU ARE HAVING TROUBLE FINDING THIS CABLE I WOULD SUGGEST TRYING THE COMPUTER BUG (412-882-3374) 5075 CLAIRTON BLVE. PITTSBURGH, PA 15236. THE FOLLOWING INSTRUCTIONS WILL GUIDE YOU THROUGH THE PROCESS OF CONNECTING YOUR CASSETTE RECORDER TO YOUR COMPUTER USING THE TI INTERFACE CABLE:

1. LOCATE THE NINE-PIN PLUG AT ONE END OF THE CASSETTE RECORDER INTERFACE CABLE. INSERT THIS PLUG FIRMLY INTO THE JACK ON THE RIGHT REAR OF THE COMPUTER.
2. LOCATE THE SET OF THREE PLUGS AT THE OTHER END OF THE CABLE. THE WIRES THAT LEAD TO THESE PLUGS ARE COLOR-CODED: RED-WHITE-BLACK.
3. LOCATE THE JACKS LABELED MIC, EAR (OR EXTERNAL SPEAKER) AND REM ON YOUR CASSETTE RECORDER.
4. INSERT THE PLUG WITH THE RED WIRE INTO THE RECORDER'S MICROPHONE JACK (LABELED MIC).
5. INSERT THE PLUG WITH THE WHITE WIRE INTO THE RECORDER'S EARPHONE (OR EXTERNAL SPEAKER) JACK (LABELED EAR).
6. INSERT THE PLUG WITH THE BLACK WIRE INTO THE RECORDER'S REMOTE JACK (LABELED REM).

THAT'S ALL THERE IS TO IT! YOUR CASSETTE SYSTEM IS NOW READY TO GO. NEXT MONTH'S TOPIC WILL BE LOADING AND SAVING PROGRAMS.

(Ed. note: This is the first of a series of articles to be presented monthly in this newsletter. Thanks to West Penn 99er's Club. If you would like to read the other articles in this series (13 to date) before they are published please check out the West Penn newsletters out from our Newsletter Librarian Dick Heim. The 2nd article appears in the Jun 1986 issue).

REMEMBER THIS DATE-TICOFF Saturday, March 18,1989-REMEMBER THIS DATE

GETTING THE MOST FROM YOUR CASSETTE SYSTEM
BY MICKEY SCHMITT
NUMBER 2
LOADING AND SAVING PROGRAMS

WHILE LOADING AND SAVING PROGRAMS WITH THE USE OF A CASSETTE RECORDER IS
NOT A DIFFICULT PROCESS IN ITSELF - READING AND UNDERSTANDING THE
INSTRUCTIONS FOR THE VERY FIRST TIME CAN BE QUITE CONFUSING. WITH THAT
THOUGHT IN MIND, I HAVE TRIED TO KEEP THE INSTRUCTIONS AS SIMPLE AS
POSSIBLE.

INSTRUCTIONS FOR LOADING PROGRAMS:

1. TYPE: OLD CS1
2. THEN: PRESS ENTER
3. FOLLOW THE DIRECTIONS AS THEY APPEAR ON YOUR MONITOR OR TV SCREEN:

    3.1 * REWIND CASSETTE TAPE CS1
    THEN PRESS ENTER

    3.2 * PRESS CASSETTE PLAY CS1
    THEN PRESS ENTER

    3.3 COMPUTER DISPLAYS MESSAGE
    * READING

    3.4 COMPUTER DISPLAYS MESSAGE
    * DATA OK

    3.5 * PRESS CASSETTE STOP CS1
    THEN PRESS ENTER

4. WAIT FOR THE FLASHING CURSOR TO APPEAR IN THE LOWER LEFT-HAND CORNER OF YOUR MONITOR OR TV SCREEN
5. TYPE: RUN
6. THEN: PRESS ENTER

INSTRUCTIONS FOR SAVING PROGRAMS:

1. TYPE: SAVE CS1
2. THEN PRESS ENTER
3. FOLLOW THE DIRECTIONS AS THEY APPEAR ON YOUR MONITOR OR TV SCREEN:

    3.1 * REWIND CASSETTE TAPE CS1
    THEN PRESS ENTER

    3.2 * PRESS CASSETTE RECORD CS1
    THEN PRESS ENTER

    3.3 COMPUTER DISPLAYS MESSAGE
    * RECORDING

    3.4 * PRESS CASSETTE STOP CS1
    THEN PRESS ENTER

4. YOUR PROGRAM IS NOW SAVED - BUT YOU SHOULD GET INTO THE HABIT OF
CHECKING ALL YOUR PROGRAMS TO BE SURE THAT THEY WERE SAVED WITHOUT ERROR.
5. CONTINUE TO FOLLOW THE DIRECTIONS AS THEY APPEAR ON YOUR MONITOR OT TV
SCREEN:

    5.1 COMPUTER DISPLAYS MESSAGE
    * CHECK TAPE (Y OR N)?

    5.2 TYPE: Y

    5.3 THEN PRESS ENTER

    5.4 * REWIND CASSETTE TAPE CS1
    THEN PRESS ENTER

    5.5 * PRESS CASSETTE PLAY CS1
    THEN PRESS ENTER

    5.6 COMPUTER DISPLAYS MESSAGE
    * CHECKING

    5.7 COMPUTER DISPLAYS MESSAGE
    * DATA OK

6. YOUR PROGRAM IS NOW SAVED - SAFELY AND WITHOUT ERROR. THAT'S IT. NEXT
MONTH'S TOPIC WILL BE KEEPING YOUR CASSETTE TAPES AND PROGRAMS ORGANIZED

(Ed. note: This is the 2nd of a series of articles to be presented monthly in this newsletter. Thanks to West Penn 99er's Club.)

REMEMBER THIS DATE-TICOFF Saturday, March 18,1989-REMEMBER THIS DATE

GETTING THE MOST FROM YOUR CASSETTE SYSTEM
BY MICKEY SCHMITT
NUMBER 3
KEEPING YOUR CASSETTE TAPES AND PROGRAMS ORGANIZED
PART I

HOW MANY TIMES HAVE YOU WANTED TO FIND A SPECIFIC PROGRAM THAT YOU HAD BUT...

1. YOU CAN'T REMEMBER WHICH CASSETTE YOU PUT IT ON.
2. OR...YOU CAN REMEMBER WHICH CASSETTE YOU PUT IT ON BUT NOW YOU CAN'T REMEMBER WHETHER YOU PUT IT ON SIDE A OR B.
3. OR...YOU CAN REMEMBER WHETHER YOU PUT IT ON SIDE A OR B BUT NOW YOU CAN'T REMEMBER WHAT THE COUNTER READING WAS FOR THE BEGINNING OF THE PROGRAM.
4. OR...YOU CAN REMEMBER WHAT THE COUNTER READING WAS FOR THE BEGINNING OF THE PROGRAM BUT NOW YOU CAN'T REMEMBER IF THE PROGRAM WAS WRITTEN IN BASIC OR EXTENDED BASIC OR...MAYBE IT WAS THAT YOU NEEDED TEII...OR WAS IT MINI-MEMORY?

IF ALL OF THIS SOUNDS WAY TOO FAMILIAR TO YOU...DON'T PANIC. YOU ARE NOT ALONE! THE SAME SITUATIONS HAVE HAPPENED TO ALL OF US WHO USE A CASSETTE RECORDER - AT LEAST AT ONE POINT OF TIME OR ANOTHER.

```
*****************************************************************
*                THE SOLUTION - GET ORGANIZED!                  *
*         STOP WASTING ALL OF YOUR VALUABLE COMPUTER TIME       *
*                    HUNTING FOR PROGRAMS!                      *
*****************************************************************
```

NOW THAT YOU SEE THE NEED FOR SOME "ORGANIZATION" - LET ME BE ONE OF THE FIRST TO TELL YOU THAT THERE ARE ALOT OF DIFFERENT WAYS IN WHICH TO GO ABOUT ORGANIZING YOUR PROGRAMS. KEEP IN MIND THAT WHILE ONE METHOD MAY SEEM TO WORK THE BEST FOR YOU IT MAY NOT BE THE BEST METHOD FOR SOMEONE ELSE. ONLY YOU KNOW WHAT METHOD WILL BEST MEET YOUR OWN NEEDS!

IF YOU ARE NOT USING ANY SYSTEM RIGHT NOW I WOULD SUGGEST ORGANIZING YOUR PROGRAMS WITH THE USE OF 3 X 5 INDEX CARDS - WITH THE FOLLOWING INFORMATION AS A GUIDELINE:

1. CASSETTE TITLE AND/OR CASSETTE NUMBER
2. CASSETTE SIDE
3. PROGRAM NAME
4. COUNTER READING
5. LANGUAGE USED
6. PERIPHERALS NEEDED
7. PROGRAM DESCRIPTION

THAT SHOULD BE ENOUGH TO GET YOU STARTED AND KEEP YOU QUITE BUSY FOR AWHILE. I KNOW THAT IT ALL SOUNDS LIKE ALOT OF WORK BUT IT WILL BE APPRECIATED IN THE LONG RUN - WHEN YOU NEED TO FIND A SPECIFIC PROGRAM AND YOU DON'T HAVE ALL DAY TO HUNT FOR IT!!!

NEXT MONTH I WILL CONTINUE WITH THE TOPIC OF KEEPING YOUR CASSETTE TAPES AND PROGRAMS ORGANIZED USING THE INFORMATION GENERATED BY THE 3 X 5 INDEX CARDS AS A FOUNDATION FOR A PROGRAM WHICH CAN BE SAVED ONTO CASSETTE.

IF YOU NEED ANY HELP IN STARTING TO GET YOUR CASSETTE TAPES AND PROGRAMS ORGANIZED JUST GIVE ME A CALL (412-335-8163) AND I'LL TRY TO HELP.

MICKEY SCHMITT

REMEMBER THIS DATE-TICOFF Saturday, March 18,1989-REMEMBER THIS DATE

To minimize wasted disk space I blocked 4 63 I/F byte records to a sector. The fields are: FGS#; male first, init, last, parent#; female first, init, last, parent#; generation; and birth date.

Many times I do not have a birth date, but one is required for the minor sort. I use an 8 digit number YYYYMMDD and print it as "YYYY.MMDD". If the date is unknown I make an educated guess and span the parent/child about 25 years. Some tombstones have years only. To distinguish which dates are guesses I enter the year YYYY, then MMDD 0000 for a correct year only and 9999 for a guess.

As for listings I list all my files in columnar format, which produces a neat appearing list. I want the flexibility to list any of the records in any format and in any sequence. I keep each file on separate floppies. So far I have only created my master (numeric), alphabetic, and male Geisinger child files.

The first set of printouts were of the complete files. First in numeric sequence, second sorted on the male (last,first) name, third sorted on the female (last,first) name. I then selected only "GEIS" records from all files, combined them, and then sorted them (major on first name, minor on date of birth). As I get more files created I will make larger combined files.

My fear is that I may soon have files which are too large to sort. I assume I would then have to sort the files first, then merge them. If you have a program which can merge files on major and minor non-adjacent fields (if such a thing is possible) I would appreciate knowing about it.

After I had my master file (immigrant descendants) created I wrote an EXB program called FGS/SEARCH. I enter any FGS# and it prints that record and all the ancestors back to the immigrant. If necessary it switches male/female so that all descendants are on the left and the spouses on the right. It then advances for the next FGS#.

I have used TI-WRITER to print descendant charts. The format and size will be limited by the capacity of the printer. On my old 80 column printer I printed descendants of the 5th generation. I summarized 1st to 4th, and detailed 6th to 9th adding spouse and the FGS#. Each generation is indented one inch after the 6th.

I want the ability to change fields, record sizes, file lengths or data bases without re-keying any data. My original system was EXB, 32K, single SSSD disk and 80 column printer.

Should any of you other genealogists care to send me your articles on genealogy or data bases, or Geis(ss)inger family group sheets, or want further details on my system, my address is RD #2-Box 741; Zionsville, Pa. 18092; ph (215) 966-5233.

I hope this will encourage you, my fellow orphaned siblings, to get an interest in genealogy and use your TI. Drink lots of ROOT beer, the preferred drink of genealogists.

CASE_OF_THE_MISSING_DISK_NAME

by John F. Willforth
PUG Peripheral, Pittsburgh PA

Last night I acciently missed typing in the DISK NAME when I began to move files from one disk to another. Everything continued fine, and when I was done, I tried out the disk to see if it worked ok, and all appeared fine. It wasn't until I went to print a label, that I found the disk could not be cataloged, copied or even a name changed!

What could I do? I'll tell you what I did... if you know the disk is DSDD, just get one of the same density and sides and insert it as if you were going to rename it. Do everything you re asked up to the point where the disk manager is about to write the new name to the disk, then pull the matching density disk out of the drive and insert the one that can't be cataloged or re-named, and press the enter key. ZAPO! The previously un-named disk now has a name, and then can be dealt with in the usual manner. There is no easier way!

```
FROM  160 CALL LOAD(-31884,39,0)
PG 2  170 CALL LOAD(-31952,255,231
      ,255,231)
      180 END
```

Set (B)ackground and (F)oreground colors of your choice (line 110)and save as "LOAD" on your disk.

GETTING THE MOST FROM YOUR CASSETTE SYSTEM
BY MICKEY SCHMITT
NUMBER 4
KEEPING YOUR CASSETTE TAPES AND PROGRAMS ORGANIZED
PART II

(Ed. note: This is part 4 in this series. Thanks to Mickey Schmitt and the West Penn 99er's Club).

THIS MONTH I AM CONTINUING WITH THE TOPIC OF KEEPING YOUR CASSETTE TAPES AND PROGRAMS ORGANIZED - USING THE INFORMATION GENERATED BY LAST MONTH'S 3 X 5 INDEX CARDS - AS THE FOUNDATION FOR THE FOLLOWING PROGRAM.

ALTHOUGH THIS PROGRAM WILL WORK AS WRITTEN - YOU ARE ENCOURAGED TO MAKE ANY CHANGES THAT YOU MAY WANT IN ORDER TO MEET YOUR OWN PERSONAL NEEDS. DON'T BE AFRAID TO DO A LITTLE EXPERIMENTING. IT CAN'T HURT AND YOU JUST MAY LEARN A THING OR TWO IN THE PROCESS.

THIS PARTICULAR PROGRAM WAS CREATED WITH THE INTENT OF GIVING YOU THE FOLLOWING OPTIONS: YOU MAY EITHER TYPE IN THE FOLLOWING PROGRAM AS LISTED - FILLING IN THE BLANKS AS THEY APPEAR OR YOU COULD JUST TYPE IN THE INFORMATION THAT WOULD APPEAR IN THE BLANK AREA AND FORGET ABOUT TYPING IN ALL THE "FORMAL TITLES". PERSONALLY I LIKE THE LATTER CHOICE MYSELF AS IT SAVES ALOT OF UNNECESSARY REPETITIVE TYPING AND IT KEEPS MY SCREEN INFORMATION DOWN TO A BARE MINIMUM WHEN I RUN THE PROGRAM.

NEXT MONTH'S TOPIC WILL BE CASSETTE TIPS - TRICKS - AND TIDBITS. IT SHOULD PROVE TO BE QUITE INTERESTING - AS I PASS ALONG WHAT I'VE FOUND OUT THE HARD WAY - AND WHAT I'VE LEARNED FROM MY FELLOW T.I. FRIENDS. I GUARANTEE THAT YOU'LL ENJOY A FEW GOOD LAUGHS - AT MY OWN EXPENSE!

```
100 REM ***************************************************************
110 REM *       PROGRAM LISTING FOR A CASSETTE TAPE CATALOG IN T.I. BASIC     *
120 REM ***************************************************************
130 CALL CLEAR
140 PRINT "CASSETTE TITLE:_______________________________________":::
150 PRINT "CASSETTE NUMBER:__________":::
160 PRINT "CASSETTE SIDE:____________"::::
170 PRINT "CASSETTE READING:_________":::
180 PRINT "LANGUAGE USED:____________":::
190 PRINT "PERIPHERALS NEEDED:___________________________________":::
200 PRINT "PROGRAM NAME:_________________________________________":::
210 PRINT "PROGRAM DESCRIPTION:__________________________________"::::
220 GOSUB 10000
230 CALL CLEAR
240 REM TO CATALOG MORE THAN ONE PROGRAM - FOLLOW THE SAME FORMAT AS USED IN
250 REM LINE NUMBERS 170 - 230. CONTINUE USING THIS SAME FORMAT TILL ALL OF
260 REM YOUR PROGRAMS HAVE BEEN CATALOGED.
270 REM CAUTION: AFTER THE FINAL ENTRY - REMEMBER TO USE AN "END" STATEMENT.
280 REM RIGHT AFTER YOUR FINAL "CALL CLEAR" STATEMENT.
290 REM FOLLOWING THIS FORMAT WILL HELP KEEP ALL OF YOUR PROGRAMMING
300 REM INFORMATION UNIFORM AND EASIER TO FOLLOW ON YOUR MONITOR OR TV SCREEN.
310 END
10000 PRINT "PRESS: ANY KEY TO CONTINUE"
10010 CALL KEY(0,K,S)
10020 IF S/0 THEN 10010
10030 RETURN
```

THANK EUGENE 99/4A for these two items

If FCTN 4 is too much of a stretch sometimes, try this: (hold down at the same time) FCTN Space bar J - it works!

REMEMBER THIS DATE-TICOFF Saturday, March 18,1989-REMEMBER THIS DATE

Financial Planning One Liners

by Tony Falco

Until computers came on the scene, calculations involving compound interest were laborious and complex. In most on the job applications, values were not calculated but rather they were read from tables. Your TI can now make financial calculation easy as illustrated by the four one liners below.

Suppose Auntie Mabel donates $1000 for your newborn son's education. Running program 1 you will find that if you invest it at 8% compunded monthly and leave it for 18 years then you will have earned $4,200.57.

You estimate needing $100,000 (a conservative estimate) for college 18 years hence. Program 2 tells you that at 8% compounded monthly for 18 years you should make a one time deposit of $23,806.27 to have $100,000 when you need it.

When you see that amount you decide that a systematic savings plan would be more practical for you. So you will invest $200 a month at 8% annual interest for 18 years. Program 3 tells you you will have accumulated $96,017.23 by the end of your 18 year ordeal.

You are curious to find the exact monthly deposit needed to yield your $100,000 goal. Program 4 to the rescue. This program says you will need $208.30 per month if you use all the figures above.

More technically speaking. Program 1 computes the future value of a one time investment. Program 2 computes the present value for a one time investment. Program 3 gives values for an annuity. And the last program creates values for a sinking fund.

Of course the hardest part is not computing the values but coming up with the dough.

```
1 CALL CLEAR :: INPUT "Invested:$":P ::
INPUT "Rate:":R :: INPUT "Cpds/Yr:":N ::
 INPUT "Years:":T :: A=P*(1+R/100/N)^(N*
T):: PRINT "FINAL VALUE=$"&STR$(INT(A*10
0+.5)/100):: END

2 CALL CLEAR :: INPUT "Needed:$":A :: IN
PUT "Rate:":R :: INPUT "Cpds/Yr:":N :: I
NPUT "Years:":T :: P=A/(1+R/100/N)^(N*T)
:: PRINT "Deposit=$"&STR$(INT(P*100+.5)/
100):: END

3 CALL CLEAR :: INPUT "Deposit:$":P :: I
NPUT "Rate:":R :: INPUT "Times/Yr:":N ::
 INPUT "Years:":T :: A=100*N*P*((1+R/100
/N)^(N*T)-1)/R :: PRINT "Final=$"&STR$(I
NT(A*100+.5)/100):: END

4 CALL CLEAR :: INPUT "Needed:$":A :: IN
PUT "Rate:":R :: INPUT "Times/Yr:":N ::
INPUT "Years:":T :: P=A*R/((1+R/100/N)^(
N*T)-1)/N/100 :: PRINT "Deposit $"&STR$(
INT(P*100+.5)/100):: END
```

TRACE_SUBROUTINE by Mike Slattery (reprinted courtesy TISHUG)

The next program will "take the TRACE line numbers off the screen and dump them to your printer...type it in at the end of your programand insert a gosub to the line number..."

```
9100 OPEN #1:"PIO"
9110 PR$=""
9120 FOR R=1 TO 24 :: FOR C=3
 TO 28 :: CALL GCHAR(R,C,X)
:: IF X=60 THEN 140 :: IF X=
31 OR X=32 THEN 150 :: IF X=
62 THEN X=32
9130 PR$=PR$&CHR$(X):: CT=CT+
1 :: IF CT>75 AND (X=32 OR X
=31)THEN PRINT #1:PR$ :: PR$
="" :: CT=0
9140 NEXT C
9150 NEXT R
9160 PRINT #1:PR$ :: PR$="" :
: CT=0
9170 CLOSE #1: :: CALL CLEAR
9180 RETURN
```

A GREAT debugging tool ... (hick)

Our Program who art in Memory Hello be Thy Name. Thy Operating System come, Thy Commands be done, at Printer as it is on the Screen. Give us this day our daily Data, and forgive us our I/O Errors as we forgive thoes whose Logic Circuits are faulty. Lead us not into frustration, and deliver us from Power Surges. For Thine is the Algorithm, the Application, and the Solution, looping forever and ever.

Return

loaded faster that way.

DOGO-

Developed at MIOT (Massachusetts Institute of Obedience Training). DOGO heralds a new era of computer literate pets.

DOGO commands include:

SIT,
HEEL,
STAY,
PLAY_DEAD and
ROLL_OVER.

An innovative feature of DOGO is the "puppy" graphics. This is a small Cocker Spaniel that occasionally leaves "deposits" as it travels across the screen.

VALGOL-

From its modest beginnings in Southern California's San Fernando Valley, VALGOL is enjoying a dramatic surge in popularity across the country. It has been adopted by many of the more youthful 99/4A programmers.

VALGOL commands include:

REALLY,
LIKE,
WELL and
Y*KNOW.

Variables are assigned with the =LIKE and =TOTALLY operators.

Other operators include the California Booleans, AX and NOWAY.

Repetitions of code are handled in FOR - SURE loops.

Here is a sample program.

```
LIKE, Y*KNOW (I MEAN) START
   IF PIZZA =LIKE BITCHEN AND
   GUY      =LIKE TUBULAR AND
   VALLEY GIRL =LIKE GRODY**MAX
THEN
   FOR I =LIKE 1 TO OH*MAYBE 100
     DO*WAH - (DITTY**2)
     BARF(I) =TOTALLY GROSS OUT
   SURE
     LIKE BAG THIS PROGRAM
     REALLY
     LIKE TOTALLY (Y*KNOW)
     IN*SURE
     GOTO*THE MALL
```

VALGOL is characterized by its unfriendly error messages. For example, when the user makes a syntax error, the interpreter displays the message:

GAG ME WITH A SPOON!

REAGAN-

This language was developed in California, but is now widely used in Washington D.C. It is the current subset of the international bureaucratic language known as DOUBLESPEAK.

Commands include:

REVENUE_ENHANCEMENT,
CAP_WEINBERGER,
CABINET,
CHOP_WOOD and
SCENARIO.

WATT and BURFORD have been removed from the commands while there is a current effort to add MEESE.

The Operating System used is NEW_RIGHT and the designated memory is THE_RANCH. The compile SCENARIO is a compile with NANCY, followed by a link with BONZO, resulting is a SNOOZE. Program bugs, called COMMIES, are removed with the GRANADA command. A program written in REAGAN commences with a LANDSLIDE and terminates with SENILITY.

Worth_Repeating_Department

TI WRITER HINT

C.R. McDonald of TICO TOPICS

My wife does not like the windowing when using the 80 column format, so I have experimented and found that by using the 40 column format, without line numbers and then reformatting each paragraph after preparing the ducument, there is no need for the window to see what you wrote in the previous paragraph. This is probably not new to anyone else, but I have never seen it as a suggestion.

****

Here three short! routines I found in the Newsletters I thought you might find interesting.

```
90 ! WEIRD SPRITE DEMO
100 CALL INIT :: CALL CLEAR
:: CALL MAGNIFY(4):: FOR B=1
 TO 10 :: FOR A=100 TO 255 :
: PRINT A :: CALL LOAD(-3197
5,-A):: PRINT CHR$(A);CHR$(9
);:: NEXT A :: NEXT B :: GOT
O 100 -
```

..IMPACT-99.. T.I. Happenings

SHAZAM

by Jack Sughrue
Box 459
E Douglas MA 01516

ADVENTURING

Marathons of the Mind

From tiny little word puzzles called *riddles* to elaborate puzzles called *mystery novels* there is really great intellectual growth. This growth is nurtured by fun: the fun of solving puzzles.

In recent years problem-solving puzzles written or developed on massive scales have become the intellectual pastime for a large number of young and old people throughout the world.

On the one hand such brainstorming, role-playing activities as the *Dungeon and Dragon* style puzzle/games have become extremely popular. These situation games require intense researching and reading skills which are only now beginning to astound teachers. How is it possible that a child who can't read *SEE PUFF RUN*. can not only read very technical books on mythology but apply that learning to problem-solving situations?

On the other hand we have - *computers!* - the greatest friend a teacher (particular those of us harried by fragmentation) ever had.

Besides being electronic flashcards and word-processors and educational development tools (i.e., Beginning Grammar, Reading Rally, Dragonmix, Logo II, et al), the computer can serve as an excellent reading/writing/problem-solving tool. This tool gets students *involved*. This tool is known, jargon-wise, as interactive fiction. But, to the novice, it is simply identified as text adventures: No graphics except those in the mind.

Most educators who have been using these games in the classrooms with any regularity have agreed that different approaches to the game (henceforth known as the adventure) are varied and effective. Most teachers prefer a group of three to five youngsters on the adventure at one time and that there should be almost no teacher intervention once the programs have been explained and the classroom structure has been established.

A good place to start for most computers would be Scott Adam's *ADVENTURES*. Pirate Adventure (the text version) is probably best to start with. With our 99/4A, of course, we might even start with simpler adventures like *FOUR VERBS* which aren't available for other computers. Others in that series include *RIVER RIDER*, *FUNHOUSE*, *HAUNTED HOUSE*, and *STONE AGE*.

Once the game is loaded there is a certain structure usually followed by most adventure programmers: At the top of the screen the scene is described in a manner similar to this: *You are in a small dark room. There is a square patch of light in the distance. Shrill, high-pitched noises can be heard above your head.*

Directions: S N W

Then there is a flashing cursor.

The players must decide what to do. They all assume the role of the one character. Everything is perceived from that person's perspective.

Now the character may be directed to go Up or South or West or take an INventory or LIGHT CANDLE (only if he/she has one and matches to light it with) and LOOK ROOM. (Most commands, as you can see, are given in abbreviated form in adventures. This helps the game move faster. Usually commands require only the first or the first three letters: N or INV. To perform specific actions (or reactions) usually requires a two-word situation: verb/noun. CLIMB TREE, GO HOUSE, PICK NOSE, TAKE KNIFE, EAT SOUP.

Very high-level games, such as those created by Infocom, allow you to write complex multiple commands in sentence form. Some of their games (the ZORK series, Hitchhiker's Guide to the Galaxy, Suspended) may take adults months. Although they are marvelous they probably would not be suited to classroom use. (Many elementary school children, alone or with their families DO these incredible adventures.)

Meanwhile, back in the classroom, if you have one or two computers going with adventures, you will note an enthusiastic brainstorming at a very profound level. This is an exciting thing to watch. This kind of experience can be easily translated into other kinds of reading, writing, problem-solving, brainstorming experiences.

Depending on the amount of time you allow a group to participate will depend on how many days it takes to complete the adventure. It is an unusually rewarding experience for the youngsters and one which motivates the next group ready to start their adventure (the same or another; it doesn't matter). A beautiful side effect is the sharing that goes on.

There may be individuals who can attempt a solo adventure for some good educational reason, too. The children will volunteer to work on the program during lunch and recess.

Adventuring turns on kids. And it's impossible to keep a turned-on kid from learning. (An excellent classroom environment is provided for computer-use also.)

##

There are other kinds of adventuring, too. The D&D-style slash & hack appeals to most youngsters. There are two that stand out for the TI: *TUNNELS OF DOOM* and the extraordinary one, *LEGENDS*. TOD is superb because you can get TOD Editor, which lets you create your own adventures; and *LEGENDS* is great because it is, simply, the *best of the D&D adventures around. The graphics are* superb, the options are excellent, the potential for growth is built in, and the "real world" geography is unmatched by any other similar game.

##

Now, for your own adventuring. There are an incredible number of adventures for the TI, depending upon what one considers an adventure.

If it's an all-text thing, then Tunnels of Doom and the great LEGENDS would not be. Thus, some adventures have graphics. But are all-text programs adventures? If so, one would have to include ELIZA (which all but seven people in the world have played) as an adventure. It certainly is an adventure of the mind, though no goals are set and no end results. Maybe psychological wanderings shouldn't be adventures.

Do adventures have to take place in the Middle Ages (or pseudo Middle Ages)? If so, then some of the very best adventures from INFOCOM (Hitchhiker's Guide to the Galaxy, Suspended, Witness, etc.) wouldn't be. But are. One Tunnels of Doom game takes place in K-Mart.

What then are adventures? Well, I think they must allow the user to make decisions that can change the course of the game. That's is player decisions. I think there has to be reactions to those decisions that are logical (even in a nonsensical fantasy a certain kind of logic must exist). I think there has to be a pre-determined (early in the game) mission. A goal. There has to be a set of tasks successfully completed to reach that goal. There has to be punishment of some kind for failure (or failure to take risks). There must be reward for achievement of the goal.

Basically, that's it. But there's still a problem. If you have Art Auction or Car Dealer or Lemonade or Gone Fishing or any number of small programs like these that have you make decisions (usually built upon other decisions), most of the above would apply: text, decision making, reaction to decision, goals (to achieve so much money or fish), punishment (bankruptcy or drowning), reward for successful achievement. Most throw in a thwarting agent, too. (Storm in Gone Fishing, other bidders in Art Auction, rain in Lemonade.

So are these adventures? Certainly more than ELIZA (which I consider a form of adventure), but they are not what many adventure players consider adventures.

There must be more - er - exotic environments: space, jungles, funhouse, battlefields (Watch it! The game Civil War might be considered an adventure.), Wonderland); exotic times (the 30's, prehistoric times, the future); exotic characters (pirates, bugblatter beasts, K-Mart customers); exotic situations (saving the world, getting a cup of tea, understanding a computer poet, finding treasure).

Which brings us to the mazes which are adventures (Zork being the most famous) and mazes (Cat and Mouse) which are not; treasures which are adventure goals (Dungeons) from treasures which are goals but not adventures (Blackbeard's Treasure).

The criteria for "true" adventures, if there is such a thing, is vast and not always something which people agree on.

All this, as you probably figured, is leading up to something.

Mickey Schmitt, the world's leading expert on TI Adventuring, is putting together for publication the ENTIRE adventure world on the TI. Every adventure ever written for the TI - Commercial, Fairware, and Public Domain. This is a vast undertaking.

If any readers have any Fairware or Public Domain adventures, please gather them on a disk or two (preferably with a TI WRITER file to go along with it, giving the title, whether it requires the XB, ADVENTURE MODULE, TUNNELS OF DOOM, E/A, or just plain BASIC), and a short description (in case I have the game under a different title or want to categorize them by theme) and mail them to me. I will see that Mickey gets them. I will also send you back an equivalent number of disks with suitable PD and/or Fairware games for your efforts in this monumental task.

Games released by Scott Adams, Asgard, Infocom, Steve Davis, A.S.D.& E., John Granros are all copyrighted materials and may not be copied and sent. They have been purchased for this compilation.

(Jack Sughrue, Box 459, E.Douglas MA 01516)