

TI * M E S



They're here!

**We provide complete
evidence to convince
the sceptics that they
HAVE arrived!!!**

see "News and Reviews".

Also, more on the MOBB and S&T Bulletin Board

T.I. User Group U.K.
Quarterly Newsletter



They
ARE
REAL!!!



**ISSUE 49
SUMMER 1995**

R.T.

Cover drawn with YAPP
running from Editor/Assembler



User Group Committee

Chairman	Trevor Stevens	249 Southwell Road East, Rainworth, Notts, NG21 0BN	01623 75
Vice Chairman	Mark Wills	20 Cockayne Road, Hartescott Grange, Shrewsbury, SY1 3QS	01743 35
General Secretary	Richard Twynning	24 Peel Road, Mansfield, Nott, NG19 6HB	01623 27
Treasurer	Alan Rutherford	13 The Circuit, Wilmslow, Cheshire, SK9 6DA	01625 52
TI*MES Editor	Richard Speed	8 Corfe Close, Southwater, Horsham, West Sussex, RH13 7XL	01403 73
Disk Librarian	Stephen Shaw	10 Alstone Road, Stockport, Cheshire, SK4 5AH	
Module Librarian	Francesco Lama	14 Granville Court, Cheney Lane, Oxford OX3 0HJ	01865 72
Hardware	Gary Smith	55 Boundary Road, Newark, Notts, NG24 4AJ	01636 70

Contents

1	From the cutting room floor <i>Hello from the new editor, where to send your submissions and in what format</i>	Richard Speed
2	Rambles	Stephen Shaw
2 - 7	The TI99/8 Computer <i>What might have been...</i>	Charles Good
8 - 10	Transferring Cassette Programs To Disk <i>DIY method, or by using programs from the library</i>	Mike Poskitt
11 - 13	The Original TI Home Computer System <i>In the days before the PEB</i>	Charles Good
14 - 17	TI's Video Controller <i>A multimedia computer over ten years ago</i>	Charles Good
17	TI Emulator Update	Stephen Shaw
18	TI Writer tips <i>A handy quick reference guide</i>	TIUG of Will County
19 - 42	The TEX-Files <i>Thoughts on the BBS, the SCSI card finally arrives and some notes on the AGM</i>	Richard Twynning
43 - 50	The BBS In Action	Richard Twynning
51 - 57	The Art Of Assembly Pt 6	Bruce Harrison
57	The Extended BASIC RND function	Walter Allum
58	Putting It All Together	Jim Peterson
59	Telephone Puzzle Solution	David McCann
60 - 61	Win The National Lottery <i>Improve your odds... and don't forget to give Mike a cut of your winnings</i>	Mike Poskitt
62	For Sale	
63	Help Needed <i>TI questions that need answers</i>	
64	Back Page - Richard Speed	

All contributions for issue 50 must be submitted by September 1st 1995

Disclaimer

All the views expressed by the contributors of this magazine are strictly their own, and do not represent those of the committee. Contrary opinions are very welcome and errors will be corrected on request.

Hello, good evening and welcome to the Summer 1995 issue of TI*MES from your new editor... I'll apologise in advance for the late arrival of this issue, but due to the confusion surrounding the editorship I thought it best to extend the deadline by a month in case any members or contributors had sent to submissions to the wrong person. I think we're up to steam now, but if you've written to Gary and not got anything back, please write again to me because your original letter may have got lost in the transfer.

Our esteemed General Secretary has written about the AGM elsewhere in this issue, so I'll spare you the report here except to tell you that I am Richard Speed, and it is to me that you should send any submissions. I welcome all TI related articles and I accept submissions in just about any form:

<i>Printed sheets</i>	Just print your articles and listings out and send them to me.
<i>TI99 disk</i>	SSSD only please.
<i>PC disk</i>	I'll accept articles on PC disks in WordPerfect, Word, Wordstar or just straight ASCII text.
<i>Hand written</i>	If you don't have access to a printer, I'll be happy to type up your submission for you. Please make your writing legible though!
<i>Email</i>	I can take submissions (ASCII text only please) via Email. Mailboxes you can use to contact me on are: SEG (using CIX) 100023,74 (using CompuServe) SEG@CIX.COMPULINK.CO.UK (via Internet)

If you don't have access to Email you can write to me (enclosing an SAE) at:

8 Corfe Close
Southwater
Horsham
West Sussex
RH13 7XL

You can also call me at home on 01403 [REDACTED] if you have any questions (non technical please, I probably know less than anyone!), please call between 7pm and 9pm on weekdays.

This is *your* magazine, please feel free to make any comments to me about content; what you would like to see and what you don't like to see in the magazine. Or better still, write an article yourself! You've got until September the first - so lets see some new faces in the next issue!

Rambles - Stephen Shaw

RAMBLES by Stephen Shaw

Hello there!!!

Very little from me this issue - the late release of the last issue, together with continued domestic difficulties means I have to rely upon previously stored materials!

An apology if anything is reprinted- over the last 18 months or so I have completely lost track of the gap between material submitted and material printed. This continued with the last issue- I was assured everything submitted was printed, but it wasn't! and my continuing request for unused pages to be returned was again ignored. This makes life very difficult here!!!! Hopefully things will begin to make better sense from now onwards...!

A big welcome to our new editor, who will be learning the ins and outs of Funnelweb and all the other little tricks of putting a magazine together.

I will begin my section with articles which appear to have been dropped from the last issue...

- and as mentioned before, if you don't like the print quality or print formatting, volunteers are required who can do better! My time is limited and my equipment very old....

=====

THE TI 99/8 COMPUTER
notes by Charles Good
Lima Ohio User Group Oct 1991

We have all heard of the legendary never released 99/8 computer. Probably the best published description of this computer to date are found in the December 1984 issue of MICROPENDIUM, where it is stated that the suggested retail price of the 99/8 would have been about \$600. Photos of this computer in the article show the nameplate "Texas Instruments Computer 99/8" just above the left side of the keyboard.

Several working 99/8s are known by me to be in private hands, and I have seen some on display, but not operating, at a couple of TI shows. I have, however, until recently never really been clear what this machine could do. Certainly the best way to judge the capabilities of a computer is to use it. I have never had the opportunity play with a 99/8, but I have done the next best thing.

I have obtained a copy of the "Final Draft" of the 99/8 user documentation "GETTING STARTED" (dated 8/30/83 with the code number "0811P") and "PROGRAMMER'S GUIDE" (the p-System part dated 8/22/83 and the rest dated 9/15/83; with the code numbers "1511P, formally 2183L", "2117L", and "1248P"). These books give an excellent description of the capabilities of the 99/8 computer and interesting insights about the very very rare Hexbus Disk Drive/Controller peripheral.

I know of no working examples of the Hexbus Disk Drive/Controller peripheral. Sources tell me that it was made for TI in Germany by Mechatronics. In my opinion, after reading these books, the greatest shortcomings of the 99/8 are the lack of an 80 column display and the inability to generate unlimited speech using its internal speech capabilities coupled with the TEII module.

The TI99/8 Computer - Charles Good

"Press any key to proceed to the next screen. This is the main selection list or the main "menu".

TEXAS INSTRUMENTS
COMPUTER

PRESS

[A] FOR TI EXTENDED BASIC II
[B] FOR P-SYSTEM
[C] FOR SET SPEED

If you have a Solid State Cartridge plugged into the slot at the top of the computer console, the name of that program usually appears as selection D on this screen.

"TI Extended BASIC II--This option enables you to access the programming language built into the Computer 99/8. When you select TI Extended BASIC II, the computer automatically executes at full speed.

"SET SPEED--This option enables you to change the speed at which the computer "runs" programs contained in certain preprogrammed software packages. When the computer is turned on, it is automatically set to run at 99/4A speed. If this option is selected, this screen will appear:

SET SPEED
[A] SLOW
[B] 99/4A SPEED FOR GAMES
[C] FULL SPEED

"To proceed to the next section, "A tour of the keyboard," select TI Extended BASIC II by pressing the A key.

TI EXTENDED BASIC II ready
62720 Bytes free

"DISK DRIVE/CONTROLLER 5102-A HexBus peripheral that uses 5 1/4 inch diskettes, finds files quickly, and allows either sequential or random file access. With its double-sided, double density capabilities, the Disk Drive/Controller can store up to 320K of information on one diskette.

"Disk Drive 5202--Up to three additional disk drives can be used with the Disk Drive/Controller (CG note: up to a total of 4 DSDD drives. Since the Hexbus has only 4 data lines data transfer is in nibbles rather than bytes. I suspect that the data transfer rate between Hexbus Disk Drive/Controller and the 99/8 or other computer would be significantly SLOWER than that obtained with the TI disk controller and a 99/4A.)

"FEATURES OF TI EXTENDED BASIC II:

"DATA TYPES--TI Extended BASIC II allows both integer and real data types for numeric variables. Because the integer data type uses less storage space and requires less processing time, using this data type when applicable (for example, with FOR NEXT loops) facilitates faster program execution. To be stored as an integer a number must be a whole number within the range -32768 to 32767.

"SIX DIFFERENT GRAPHIC MODES--These are accessed using CALL GRAPHICS(x). A wide variety of display techniques and applications, including dividing graphics and text, are available with the six graphics modes:

"PATTERN MODE--The default mode when using TI Extended BASIC II is selected. Characters are in 24 rows and 32 columns with each character defined by an 8x8 pixel configuration.

The TI99/8 Computer - Charles Good

"TEXT MODE--Characters are in 24 rows and 40 columns, with each character defined by an 8x6 pixel configuration.

"SPLIT SCREEN MODE: TEXT HIGH--The screen is split into two portions: the text portion (the top one third of the screen) and a graphics portion. Text is 8 rows by 32 columns; the graphics portion is 128 pixels by 256 pixels.

"SPLIT SCREEN MODE: TEXT LOW-- The same as Split Screen: Text High except that the text is at the bottom one third of the screen, with graphics at the top.

"HIGH RESOLUTION MODE--A full screen graphics mode with 192x256 pixels.

"MULTICOLOR MODE--Rather than in characters or pixels, the screen displays in blocks, with 48 x 64 blocks.

"Screen Margins--CALL MARGINS enables you to redefine the screen margins and thus define a text "window" on the screen.

"FREESPACE(0) returns the amount of unused memory space.

"VALHEX returns the decimal (base 10) value of a hexadecimal (base 16) number.

"TERCHAR returns the key code of the key combination (such as ENTER, PROC'D, BACK, BEGIN) that terminated the last INPUT, LINPUT, or ACCEPT statement.

"ASSEMBLY LANGUAGE SUPPORT--TMS9995 (of which TMS9900 code is a subset) assembly language subprograms may be loaded and run. The subprograms INIT, LOAD, LINK, PEEK, PEEKV, and POKEV are used to access assembly language subprograms. Relocatable assembly language programs created for TI BASIC will execute correctly, although NUMREF may return an integer value instead of a floating point value.

In addition, a string reference error may occur with STRREF because STRREF (in TI BASIC) only allows strings up to 255 characters. All POKES and PEEKS as used in previous BASICs fail in TI Extended BASIC II. The INIT subprogram with no parameter allocates 8K bytes of memory for assembly language subprograms. If a parameter is specified, more or less (up to 24366 bytes) may be allocated. Examples:

CALL INIT allocates 8K bytes of memory space.

CALL INIT(200) allocates 200 bytes of memory space.

CALL INIT(0) releases all memory previously allocated.

"DISPLAY VARIABLE 80 FORMAT--Files in DV80 format, created by the LIST command or a text editing or word processing program, may be loaded with the OLD command. (CG Note: Does this mean that you can just display DV80 on screen, or can you OLD and RUN BASIC code created with a word processor? I don't know.)

"KEY CHIRP--TI Extended BASIC II enables you to turn on an audible "chirp" that sounds whenever a key is pressed. Enter the following command to turn on the key chirp: CALL LOAD(VALHEX("84BD"),1). Enter the following command to turn the key chirp off: CALL LOAD(VALHEX("84BD"),0).

"STRING LENGTH--BASIC II permits strings up to 4090 characters in length. TI BASIC and TI Extended BASIC permit strings of up to (only) 255 characters in length.

"RECORD LENGTH--TI Extended BASIC II allows diskette data files created on the Drive/Controller to have VARIABLE records with lengths up to 4090 bytes. TI BASIC and TI Extended BASIC allow VARIABLE records to be only 254 bytes long. Diskette data files with FIXED length records are limited to 255 bytes (same as TI BASIC). TI Extended BASIC II allows cassette data files to have FIXED length records up to 4032 bytes. TI BASIC and TI Extended BASIC limit cassette data files to 192 bytes or less.

"SOFTWARE CARTRIDGES--CALL to routines contained in a plug in cartridge are not accessible in TI Extended BASIC II. Thus, programs that use the Personal Record Keeping cartridge will not execute properly. Text to Speech cannot be accessed from TI Extended BASIC II with the Terminal Emulator II cartridge. Otherwise the Terminal Emulator II cartridge functions normally.

"RESERVED WORDS--The following reserved words are additions to the TI Extended BASIC reserved word list: ALPHA, FREESPACE, INTEGER, LALPHA, REAL, TERMCHAR, and VALHEX.

"CALL DRAW, CALL DRAWTO, CALL FILL, and CALL DCOLOR--These are graphics subprograms which enable you to plot graphics and add color to them on the screen (in graphics mode and in the graphics portion of a split screen). CALL DRAW and CALL DRAWTO draw or erase lines between specified pixels, thus making elaborate figures or drawings possible. (The screen is comprised of a grid of 256x192 individual pixels.) CALL FILL colors the area surrounding a specified pixel. CALL DCOLOR sets the graphics colors that are used by CALL DRAW, CALL DRAWTO, CALL FILL, CALL HCHAR, and CALL VCHAR.

"Some HexBus peripherals can be accessed by using the general format for file specification described earlier. (This is the method we are all familiar with in BASIC file access with the 99/4A.) Hexbus peripherals that may be accessed with this alternate method of addressing are the Disk Drive/Controller, the RS232, and the HexBus Modem.

```
Alternate Device name: Hexbus Number
-----|-----
DSK.DISKNAME.....: 100.DISKNAME
DSK1.....: 101
DSK2.....: 102
DSK3.....: 103
DSK4.....: 104
RS232.....: 20
```

"HEXBUS SUBCOMMANDS--The CA (catalog) command cannot be used with the Hexbus Disk Drive/Controller peripheral. (CG note: CA is supposed to generate a catalog of programs stored by the 99/8 on a wafertape. On my 99/2 computer CA does generate a catalog of files on a Wafertape or Mechatronic Quickdisk, and is supposed to work similarly with TI's never released Hexbus Interface for the 99/4A.)

"The available characters (ASCII 0-255) and character sets (32 character sets numbered 0-31) in Pattern Mode are listed.

-----continued-----

The TI99/8 Computer - Charles Good

"The P-code interpreter, which is built into your computer, enables you to execute existing p-System programs. To develop your own programs you must have a TI disk system (sold separately). Also necessary are some or all of the following TI products (sold separately):

1. p-System Editor/Filer/Utilities.
2. UCSD Pascal Compiler.
3. p-System Assembler/Linker.

"Note: These products are designed specifically for use with the 99/8. Software designed for use with the TI 99/4A Home Computer may not work when used with the 99/8.

"With the p-System you can execute high-level language programs such as UCSD Pascal and TI PILOT. UCSD Pascal is compiled and TI PILOT is interpreted to an intermediate language called "P-code" or "pseudo-code." The p-system interprets the P-code and instructs the computer to execute the appropriate machine language instructions.

"The built in MINI FILER program in the Operating System (unit #14) file named SYSTEM.TI.FILER provides many of the file management capabilities of the p-System Filer program. (MINI FILER is in ROM. The p-System Filer program is extra cost disk software.) Use the "G"o command from the system command level to access the MINI FILER.

"When you enter the MINI FILER, a menu of "Special p-System Commands" is displayed. To select a menu option, press the letter that precedes it.

- A. Run a program
- B. Copy a disk
- C. Copy a file
- D. Delete a file.
- E. List files on a disk.
- F. Format a new disk.
- G. Clear a disk directory.
- H. Combine free disk space.
- I. Change name.
- Q. Return to standard p-System menu.
- ? Help
- S. Set single disk system.(Toggles to "multiple" disk)

"APPENDIX D: ASSEMBLY LANGUAGE SUPPORT ROUTINES

"The TI Computer 99/8 provides several utilities that enable you to access special capabilities of the computer through TMS9900 assembly language. With these utilities, you can change the values of the VDP chip, access the DSR for peripheral devices, scan the keyboard, link a program to GPL routines, and link to the Editor/Assembler loader. Remember that these can only be used in TI assembly language programs.

"The following list gives each of the utilities predefined in the REF/DEF table: VSBW VNBW VSRR VNBR KSCAN GPLLNK XMLLNK DSRLNK LOADER.

"The TI Computer 99/8 has more utilities available through the Editor/Assembler than did the TI 99/4A Home Computer. As a result, the XMLLINK tables have been changed, so old assembly language programs may need to be updated

NOTE 1: A 99/8 and peripherals can be seen on a Lima videotape available from your group disk librarian.

NOTE 2: Many of the XBII additional and amended commands were used in the Myarc version of XB, which required a special memory card as the language occupied 128k.

Transferring Programs from Cassette to Disk by Mike Poskitt

I was sifting through my fairly large collection of TI-related memorabilia the other day, when I discovered some cassettes containing long forgotten TI Basic programs that I had omitted to transfer to disk when I expanded my TI-99/4a way back in 1988.

.....Ah yes, I remember it well. I was staying with family in Weymouth in the Summer of 1988, and feeling flush having just pocketed my pay cheque. Armed with a heavily laden wallet (the likes of which has never been repeated), I drove the 40 miles to Honiton where I eagerly descended upon the home of Mr. PARCO himself - Francis Parrish - in a leafy country lane on the outskirts of the town. His garage proved to be a veritable Aladdin's cave of TI-99/4A hardware, software, magazines and books. I greedily snapped up lots of bargains, including a full expansion system with it's mind-boggling 32K of extra RAM, for a mere £400.00. It was second-hand of course but hey, it was built like a tank (fish tank?) and it has functioned faultlessly ever since. Anyway I digress.....

Yes, finding the aforementioned cassettes encouraged me to search out some utilities previously acquired from Stephen Shaw's massive Disk Library, in order to make the task of transferring the programs easier. Having extracted these utilities, I thought it worthwhile telling you about them, if only to encourage some more activity on the Disk Library front!

Now, for those who haven't actually tried it before, you may be forgiven for thinking that to transfer a TI Basic or Extended Basic program you simply type OLD CS1, then SAVE DSK1.xxxx (where xxxx is the name of the program). Well yes, this may work, but attaching a disk system to your console results in a disk buffer claiming some of the available RAM, leaving less for your program. Unfortunately the TI's cassette utilities only use the console's VDP RAM and cannot access the extra 32K RAM in the expansion box. So if your program uses most of the available RAM, you may be able to load it from cassette into an unexpanded system, but not into a disk based system.

In actual fact, the exact amount of space required by the disk system depends on the number of files which are allowed to be open at any given time. The default number is 3, and the disk system then uses 2088 bytes of your precious RAM! You can use the CALL FILES command to change the number of files that may be open, bearing in mind that the greater the number of allowable open files, the more memory the disk system uses. (The amount of RAM available is reduced by 534 bytes when a disk drive is attached, plus 518 bytes for each allowable opened file).

The CALL FILES command (it must not be used as a statement in a TI Basic program) is used as follows :

CALL FILES(number)
NEW

(Where "number" is any number from 1 to 9)

Transferring Cassette Programs To Disk - Mike Poskitt

Using the command CALL FILES(1) may release sufficient memory, otherwise the options available to you are DIY methods or utilities designed for the job. Let us start with the DIY option:

1. Disconnect the disk drive system by switching on your console without first powering up the expansion box. This rids your VDP RAM of that nasty, intrusive, nay downright parasitic, slimy little RAM-eating disk buffer.
2. Load your program from cassette in the usual way.
3. If there are lots of obviously unnecessary lines of code -- I'm thinking of REMark statements in particular -- then remove them, save your program to cassette and then reload with the disk drive system powered up. Still not enough memory? Then take the following steps:
4. You will need to split the program into bits. So, load it from cassette again with the disk drive disconnected.
5. Start deleting lines from the beginning of the program and then save to tape A. Now reload the original unadulterated program and delete lines from the end. Then save to tape B.
6. Now reconnect the disk system and load tape A; save to disk.
7. Load tape B and save to disk in MERGE format.
8. Load A from disk, then merge in B and you have a complete program which can be saved/loaded to/from disk.
9. If you are lucky enough to possess Triton's Super Extended Basic Module, this process is a lot easier. You simply block delete the first half of the program and save to disk, then block delete the second half of the program and save, then merge the two halves.
PHEW!!!!

.....or you can blow the cobwebs off your wallet, eke out a few measly quid, and send them to Stephen Shaw in exchange for some useful utilities.

Let us look at some of the utilities that I have found particularly useful. First on the agenda is a program called DSKBUF. It works like this:

```
RUN "DSKx.DSKBUF/EXB"  
CALL LINK("SAVE")      (this will save the disk buffer)  
OLD CSI                 (load the program now)  
CALL LINK("LOAD")     (this will restore the disk buffer)  
SAVE DSKx.PROGRAM     (for PROGRAM insert the name of your program)
```

OK, so now that you have your program safely installed on disk, you may wish to convert your TI Basic program so that it runs in Extended Basic --- particularly useful if you want to autoloading a menu so that a program can be selected by a simple key press (anyone with an expanded system will know that if TI Extended Basic is selected, it searches for a program named LOAD on the disk in Drive 1. If such a program exists, it is placed in memory and run); or perhaps you want to add extra code to the program and therefore need to take advantage of the extra memory available to TI Extended Basic programs (remember that TI Basic can only make use of the 16K VDP RAM in the TI-99/4A console).

Transferring Cassette Programs To Disk - Mike Poskitt

This is where the utility `VDPUTIL2/M` comes in handy. A few straightforward steps and the job is done:

1. Load your TI Basic program.
2. Make sure there are no lines numbered 1 to 5 (RESequence if necessary).
3. Type `MERGE DSKx.VDPUTIL2/M`
4. Wait for the drive to stop before removing disk as it is accessed several times!
5. Insert a new line at the beginning of the program with `CALL VDPUTIL2`.
6. Save new program to disk.

Aha, I hear you say, but what about all those Mini Memory programs that I have on cassette? Again, the Disk Library has a utility to transfer them to disk, `MM>DISK/OB`. This utility requires that you have a Mini Memory Module, 32K RAM and a Disk system (this of course begs the question: If you don't have the aforementioned, why would you want to use this utility?!).

What this machine code program does is to place itself in the 32K RAM. It then saves the contents of the Mini Memory 4K RAM (whatever they are) to disk, in memory image format ("program"). The program can then be used to load the memory image disk file back into the Mini Memory (the Mini Memory may contain a basic program, machine code program, or data).

This is what you have to do:

1. Place the information to be saved to disk into the Mini Memory module.
2. In command mode, type in: `CALL LOAD(28706,160,0,255,224)`
3. In command mode, type in: `CALL LOAD("DSK1.MM>DISK/OB")`
4. Now save the contents to disk by typing: `CALL LINK("SAVE","DSK1.MYPROG")`
(Where MYPROG may be any suitable disk file name).
5. To recall from disk, in command mode type in:
`CALL INIT`
`CALL LOAD("DSK1.MM>DISK/OB")`
`CALL LINK("LOAD","DSK1.MYPROG")`

.....and the Mini Memory will contain the data/program previously saved!

So, in summary, the utilities I have discussed are as follows:

`DSKBUF`.....for transferring TI Basic/Extended Basic programs from cassette to disk

`VDPUTIL/M`.....for converting TI Basic programs to TI Extended Basic.

`MM>DISK/OB`.....for transferring Mini-Memory programs from cassette to disk.

I hope this is of interest to some of you --- as already mentioned (no apologies for repeating myself), all the above utilities can be obtained from Stephen Shaw, so drop him a line and make his day! While you're at it, why not request some of the other excellent disks available from the library. I'm sure many of you don't know what you're missing. Come to think of it, I haven't used the Disk Library in a long while.....

FCTN-QUIT

The Original TI Home Computer System - Charles Good

THE ORIGINAL TI HOME COMPUTER SYSTEM antiques described by Charles Good Lima Ohio User Group

The "original" TI Home Computer system, released to the public in 1979 and 1980, consisted of the 99/4 computer (without the "A") and a series of stand alone peripherals that plug directly into the side of the 99/4 (or 99/4A) console, or into the side of the previous peripheral (hence the unofficial descriptive term "freight train peripheral"). Each of these freight train peripherals except the speech synthesizer has a base that measures 17x26cm (a bit larger than 6.5x10 inches), a separate power supply rated at 23 watts, and its own separate power cord. I recently purchased a "4" (just to play with) and was later given many of the freight train peripherals. After using these devices for awhile I realize how fortunate we are to have the "4A" and its peripheral expansion box.

Components of the "original" TI home computer system are listed below, together with their official TI part numbers and some prices mostly quoted from an ad by CBM INC of Lexington KY published on page 12 of the first edition (May/June 1981) of 99er Magazine. These CBM INC prices are probably below TI's official list price. These peripherals are not the same as those designed to fit in the PE Box. PE box peripherals all have "PHA12xx" part numbers and are described in official TI publications as "cards".

- TI 99/4 console, (PHC004C): \$499
- RF (TV) modulator; in 1980 this was an extra cost item, (PHA2100): \$41
- Solid State Speech Synthesizer; the same one most of us still use, (PHP1500): \$122
- 32K RAM memory expansion, (PHP2200): \$325
- RS232 Accessories Peripheral, (PHP1700): \$183
- Solid State Thermal Printer, (PHP1900): \$325
- Disk Drive Controller, (PHP1800): \$243
- Disk Memory Drive, (PHP1850): \$399
- P Code Peripheral, (PHP 2400): \$399.95
- Video Controller, (PHP2300): \$699.95

Prices of the last two items are official list prices quoted from TI's suggested retail price list dated June - December 1982 (1049705-1).

You can connect a maximum of three peripherals in series to the right side of the computer. If present, the speech synthesizer has to be first and the 32K second. A "typical" freight train minimum expansion system (99/4 with modulator, 32K, thermal printer, controller and one drive) would be almost four feet wide and cost \$1832. Bringing the system up to the maximum of three 888D drives and buying all the other freight train peripherals would bring the cost up to a total of \$4035. Wow! And you can only have simultaneous use of 3 peripherals.

In this article I will describe what I know about these freight train peripherals. I have hands on experience with the Thermal Printer, 32K, and Disk Controller. I will not discuss the Speech Synthesizer since the 1979/80 device is the same one we are all familiar with. In a separate article I will describe my experiences with the 99/4.

The Original TI Home Computer System - Charles Good

--32K EXPANSION MEMORY: This functions exactly like the equivalent PE box card. These days you can, for about \$10, buy a 32K RAM chip that measures about 1x3cm and draws very little current. It amazes me that TI's original 32K was so bulky and required a 23 watt power supply. A 12 inch black and white TV only draws 29 watts. But I guess if you compare a 1955 room sized UNIVAC computer in memory, watts of power consumption, and bulk, the vintage 1979 TI 32K looks pretty good.

--RS232: This stand alone box offers only one RS232 port and no parallel port. The PE Box RS232 card allows connection of TWO RS232 (serial) devices (with a special Y cable) AND one parallel device all to the same card. The PE Box card is obviously superior to the stand alone peripheral.

--DISK DRIVE CONTROLLER: This device used the original DISK MANAGER module (the DMI), and can control up to three SSSD stand alone drives. Double sided is not available with the freight train disk controller. The main difference between the DMI and DMII modules is that the "I" has no provision for double sided disk initialization. A TI stand alone drive plugs directly into the back of the freight train Disk Controller without the need for any special adapter cable other than the cable that comes with the stand alone drive.

Other drives plug into the cable of DSK1 using a small adapter board. A special cable that comes with the PE box controller card is needed to plug a stand alone TI drive to the back of the controller card for use as DSK2 or DSK3.

An interesting feature of the freight train controller in combination with the DMI module is that they do not recognize the "whole disk protected" byte >10 of sector zero. With the TI PE box controller and the DM2 module, if this byte is set for a value of >50 you cannot copy the disk with the DM2.

--THERMAL PRINTER: This is printer device "TP", and was sold to TI users at a time when cheap dot matrix or daisy wheel printers cost \$600+. The 1982 list price for the 99/4A's official dot matrix printer was \$750. The TP uses 3.5 inch thermal paper, prints 32 characters per line, and like all thermal printers is both quiet and slow. 3.5 inch thermal paper rolls are a non standard size these days.

TP users either have to purchase 10 year old official TI paper from one of the few TI dealers that stock this item, or use a paper cutter to trim 8.5 inch FAX paper rolls down to 3.5 inches. Such 8.5 inch wide FAX rolls are commonly available these days from many stores including SEARS, KMART, and WALMART. On the title page of the TP manual it says that the TP "prints a copy of a TI BASIC program or the screen displays from certain Command Modules."

And that is about it! A few modules, such as MUSIC MAKER, allow screen dumps with the TP. You can specify output to the TP with the DM1, DM2, PRK, Statistics, LOGD2, and maybe a few other modules. You can't use the TP with TI Writer, Funnelweb, the EA module, or Microsoft Multiplan. From BASIC you can LIST a program to the TP, a common application.

You can also OPEN a file to the TP using any of these file attributes: SEQUENTIAL or RELATIVE, DISPLAY or INTERNAL, OUTPUT or APPEND, FIXED or VARIABLE. I can't imagine what use RELATIVE, INTERNAL, or APPEND have in OPENing a printer file. When opened in INTERNAL, the printer prints a meaningless graphic of the internal representation of each ASCII character. The maximum length of a VARIABLE TP attribute is 32.

The Original TI Home Computer System - Charles Good

All printed characters of the TP's built in character set are on a 5x7 dot grid. The TP has a unique graphic set for ASCII 0-31 and the usual alpha/numeric characters for ASCII 32-127. Each printed dot of a character is printed only once and individual dots can be seen with the naked eye. There is no way to make extra dense high quality characters. Emphasized, double strike, and "NLQ" is not available.

The user can also, using an 8x8 dot grid, redefine ASCII chars 32-159 in BASIC using CALL CHAR, and then directly print any of these redefined chars to the TP with the appropriate keyboard keypress as in PRINT #1:"(" where (has been redefined, or with PRINT #1:CHR\$(xxx). This is a neat trick! It is much harder to print redefined characters with other kinds of dot matrix printers.

--VIDEO CONTROLLER: A photograph and brief description of this peripheral appears as part of an article on page 53 of Volume 1, No. 4 of 99er Magazine (Nov/Dec 1981). The photograph shows a box identical to that of the stand alone 32K or disk controller, with a cable coming out of the right side where the "pass through" expansion bus is found on other stand alone peripherals. The article describes the video controller as allowing "computer controlled interactive video with VCR's and Video Disk Players", whatever that means.

As evidenced by the videos we created from the formal presentations at the 1990 Lima MUG Conference, it is possible without this device to mix human voice, computer audio and video output, and video camera footage on the same video tape. Such mixing of various audio and video sources was done by us manually however, not under computer control. An extra cost cable (\$99.95 for each of the three available cables in the June - December 1982 TI price list) is needed to interface the video controller to a Sony or Panasonic VCR or a Pioneer video disk player.

I really don't understand the need for computer control of a video disk player. If I remember correctly, 1980 video disks resembled phonograph records in that you could only PLAY them from the beginning, not record onto them.

[update- this peripheral enabled you from Basic to GOTO a particular spot on the tape, and to operate all the usual video commands, ef FF, REW, PLAY etc. It was possible for a Basic program to interact with the user and for a particular bit of videotape selected from a screen menu or by the program to be played. The VCR could also be used as a storage device in the same way as a cassette player, but OLD VC etc and programs and data could be mixed with video pictures on the same tape.]

--P CODE PERIPHERAL: My June - December 1982 TI price list states that this device is "available only until replaced by peripheral card", with such a card "available in second quarter 1982." The freight train P Code peripheral is apparently exactly equivalent to the PE Box P Code card.

There you have it folks, the original TI Home Computer expansion "system". Now you know why the expansion port on the 99/4 and 99/4A is on the SIDE of the console, rather than on the back where it should have been placed. You can only use three of these freight train peripherals at once, and they take up huge amounts of desktop space. Arn't you glad we now have the PE Box!

I want to acknowledge the generous gift of Mr. E.T. Breer of the St. Louis Missouri User Group who gave me several of the freight train peripherals described in this review.

TI's Video Controller - Charles Good

TI'S "VIDEO CONTROLLER": YEARS AHEAD OF ITS TIME
described by Charles Good Lima Ohio User Group

HISTORICAL BACKGROUND:

"MULTIMEDIA" is the hot concept described today in many computer magazines. Today the term usually refers to combining CD ROM text and graphics and digitized sound files and graphic images from various other sources in a managed sight/sound presentation all under computer control. In 1981 CD ROM did not exist, but VCR's and video disk players playing sight/sound disks the size of LP phonographic records did.

At the May 1981 Consumer Electronics Show TI exhibited a side car peripheral called the VIDEO CONTROLLER designed to mix VCR and video disk sound and audio with 99/4A sound speech and screen displays, all under the control of a running TI BASIC or Extended Basic program.

This first showing of the VIDEO CONTROLLER is described in words and photos in 99er Magazine v1 #2 (July/Aug 1981). The "Video Controller" Bill Cosby video tape we have available in the Lima UG library came from this 1981 show. This video shows a cigar smoking Bill asking people to press the number 1,2, or 3 key on the computer next to him in order to see specific video demos of TI software located at specific places on the video tape.

The November 30 and December 7, 1981 issues of INFOWORLD contain announcements about the release of the Video Controller side car peripheral and associated Course Designer software. This side car VIDEO CONTROLLER peripheral was at the Jan 1982 Las Vegas CES show, the same show that TI used to introduce the Peripheral Expansion Box to the world. Vol.1 #4 of 99er MAGAZINE has a good photo and article about the VIDEO CONTROLLER at this show. List price in 1982 was \$699.95 with one set of cables, plus \$99.95 if you needed another set of cables to hook the VIDEO CONTROLLER to a different kind of VCR or video disk player, plus \$199.95 for the "Course Designer Authoring Package". That's a whopping \$1000!

The Course Designer Authoring Package is a two disk Extended Basic package which includes TI's Text-To-Speech. It is designed to aid in using the VIDEO CONTROLLER for Computer Aided Instruction but can also be used to develop CAI lessons that don't use the VIDEO CONTROLLER. CDAP was reviewed in 99er MAGAZINE v1 #6. I have this very rare software and may review it in a future article. The CDAP subprograms are dated in REM statements as late as 5/12/82, so the CDAP could not have been available in late 1981 as TI's publicity states. Thanks to Bill Gaskill, I also have a copy of an official TI brochure dated 1982 showing the VIDEO CONTROLLER side car and a fancy looking video disk player. The brochure suggests business training applications for the VIDEO CONTROLLER.

TI listed the side car peripheral as PHP2300 in its Jan-June and June-Dec 1982 retail price lists, stating "The Video Controller is intended for industrial and commercial use, it is not intended for use in the home". This statement means that the peripheral DIDN'T have FCC Class B (home use) certification. Some dealers in 1982 advertised the side car VIDEO CONTROLLER (at below list). The earliest example I can find of this is a \$539.95 price quoted in an ad on the inside front cover of 99er MAGAZINE v1#4 (early 1982). Later TI developed the VIDEO CONTROLLER as a PE Box card, PHP1290. This card is listed for "\$399.95 (Pending FCC Certification)" in TI's last official 99/4A price list dated June-Dec 1983.

TI sold few or no side car video controllers and definitely didn't sell any video controller cards. The card and probably also the side car peripheral are, I believe, "Never Released Peripherals". Also, in spite of being listed as PHD5068 (\$199.95) in TI's last official price list, TI apparently never sold its Course Designer Authoring Package software. CDAP is another "Never Released" product.

Why was the VIDEO CONTROLLER and associated software never released? I suspect the answer has something to do with price, and the initials "FCC". The peripheral was (after the impact printer) the second most expensive 99/4A item in TI's price lists, and connected to VCR's or video disk players costing (in 1982/83) \$1000+. That's a lot of money for the typical "Home" computer owner. Since TI specifically states in my June-Dec 1982 price list that the VIDEO CONTROLLER is for "industrial and commercial use", and since TI's last catalog says "pending FCC certification", I suspect that because of radio frequency interference TI never did obtain FCC permission to sell the VIDEO CONTROLLER. Evidence discussed below supports this hypothesis.

DESCRIPTION OF THE VIDEO CONTROLLER PE BOX CARD:

Thanks to the generosity of Charles Stringer and Mike Wright I have an actual VIDEO CONTROLLER card, its 1982 user guide, and a circuit diagram of the card sitting in front of me as I write this. The card comes in a TI clam shell with an official looking printed label that says "Video Controller Model No. PHP1290". The serial number space on the label is blank and hand written are the words "Qual Unit Not for sale".

Sticking out the back of the card is a 26 pin edge flat edge connector and a female mini phono jack like those on a TI cassette program recorder. Once removed from the clam shell you can see lots of chips that say El Salvador, Malaysia, and Korea. Most of these chips have the TI logo, but none say U.S.A. I can see why many TI products are labeled "Assembled in USA with domestic and foreign parts"!

The important chips seem to be a PAL1216CN/8237 and an AM18145CDZ/1501392-19. My circuit diagram identifies the AMI chip as a "TMS4732 4k x 8". My circuit board has "VIDEO CONT. 1050217-2" engraved on it, apparently a TI part number. My schematics indicate that the "Formal Release" product number would have been 1050218.

These schematics from TI's consumer products group have several signatures dated between June and August 1982. Of great significance is the "1-3-84" date of the signature immediately below the words "Final Release" (over two months AFTER TI left the Home Computer market), and the fact that the "FCC APVD" box lacks a date or signature.

My unpublished preliminary PHP1290 doc says the card has FCC class A (commercial, not home use) certification, but the lack of a signature on the schematic suggests that even this low level certification was not achieved.

In addition to the RF modulator or video cable we normally use, 5 other cables are needed to hook the VIDEO CONTROLLER to a VCR. One cable goes from the card's edge connector to the VCR's remote control. Other cables go from the VCR, the monitor, and the console's audio/video out jack to a "relay box". This box, under control of the VIDEO CONTROLLER, switch the monitor back and forth between computer audio/video and VCR or video disk audio/video. Unfortunately I don't have a set of cables or the relay box, so I can't make my VIDEO CONTROLLER card do its tricks.

TI's Video Controller - Charles Good

When I put the card into my PE box, the Horizon Ramdisk config program tells me that the card has a CRU address of 1C00. From BASIC command mode I can enter OPEN #1:"VC.H",INTERNAL without getting an error message. The docs say this means my card is installed properly.

CAPABILITIES OF THE VIDEO CONTROLLER:

Although I can't test my VIDEO CONTROLLER because I lack the proper cables, my documentation tells me what I should be able to do. What follows is based largely on this documentation.

The VIDEO CONTROLLER hooks up to some 1/2 inch (VHS or Beta) or 3/4 inch ("professional" size) VCRs or a Pioneer video disk player. TI provides a list of 1983 machines known to be compatible, but some other VCRs of that era, not on TI's list, are probably compatible. Even if I had a proper cable set I can't today go out and buy a VCR to use with my VIDEO CONTROLLER card. You need a VCR with a WIRED remote control jack and an audio dub input jack. This is not the same as the "audio in" on the back of most VCR's. Audio dub allows you to add audio to prerecorded video without erasing the video. Such VCRs were sold in retail stores in the early 80's for about \$1000. Most were top loaders. I once owned one and now wish I still had it. The OSU Lima Campus still has a couple of these machines. Few modern VCRs sold for home use have audio dub, and WIRED remotes are unheard of these days.

The VIDEO CONTROLLER allows you to use a VCR as a mass storage device, almost exactly as one would use CS1. "SAVE VC" saves a BASIC (either BASIC) program to video tape starting at the beginning of the tape. "SAVE VCA" saves a program starting at the current tape position. "OLD VC" automatically moves the video tape to the beginning and OLDS a program from there. "OLD VCA" attempts to load a program starting from the current tape position.

You can also store data files on video tape by first OPEN #2:"VC",INTERNAL and PRINT #2:"DUB" to open the VCR dub channel, and then PRINT #1:"DATANAME",FIXED to send computer data to a previously OPENed data file stored on video tape. Just as with cassette tapes, record length must be fixed at 64, 128, or 192, and APPEND, VARIABLE, and RELATIVE are not allowed. You can PRINT, INPUT, and LINPOT to and from such video tape files. Of course you can't use a video disk as mass storage since video disks (like CD ROMs) are read only media.

Another type of OPENed file allows the 99/4A to control the video unit. First you OPEN #1:"VC",INTERNAL and then you PRINT #1 the commands that control the VCR. The following commands are available:

PRINT #1:"ONRL" sends video tape (or video disk) audio and video to the monitor.

PRINT #1:"OFFRL" turns off the relay box and sends computer audio and video to the monitor.

PRINT #1:"INIT" marks the start of the tape. I don't know if this means the current tape position as "start" or whether the tape rewinds to its beginning.

PRINT #1:"GOTO",LOCATION-NUMBER forwards or reverses the tape to a specific location. Each number is 16/30 of a second of tape time on VHS systems.

The following PRINT #1:"COMMAND"'s do the same thing as pushing buttons on the front control panel of the VCR: STOP, PLAY, FWD, REW, REC, and PAUSE. With a video disk player, commands are available to display specific still image frames or chapters. A chapter is a large group of frames. Viewing a chapter is similar to playing a specific track on a modern audio CD.

The VIDEO CONTROLLER's capabilities were all designed to allow interactive computer/video training. These lessons could consist of computer segments with computer text/speech/graphics, video segments, test questions (multiple choice or T/F), and branch points depending on the answers to the questions.

Multimedia! A modern example- at Michigan's interstate highway tourist information centers just across the Indiana border you can walk up to a computer terminal displaying a multi color Michigan map with numbers on the map, press a number on the keyboard, and see a short computer AND VIDEOTAPE segment showing the neat tourist stuff at that location. You are then returned to the Michigan map where you can press another location number. Michigan could have done this in 1982 if the 99/4A VIDEO CONTROLLER had been available. This device really was years ahead of its time.

=====

...Rambles..

Stephen Shaw writes:

TI EMULATOR 5.01...

My last letter from Mr Swartz indicated that he was no longer distributing Version 5.01, that he was not accepting registrations, and that he had ceased support. I would LIKE to see Vn 6, but it may be that PC99 has achieved monopoly status here! If I hear of Vn 6, I will let you know as soon as possible.

Meanwhile, I have well over 50 modules RUNNING on TI Emulator, including Triton Super Extended Basic, Multiplan, TI Base, Funnelweb, and some favorite XB programs such as Diablo and Spontaneous Reaction. All driven by a nice simple menu system. Neat. And the whole package takes up less disk space than one major Windows game!

=====

Mike Poskitt will have left our shores by the time you read this, for that other far distant England, sometimes known as New Zealand, and sometimes as Aotorea. Our wishes go with him. Mike is an original TI owner - my first contact with him that I can identify was way back in May 1983 - before TI pulled the plug on us.

=====

To whoever placed the group mention in Computer Shopper, (was it Alasdair?), well done. Keep it up- the magazine indicated it was happy to have further info.

=====

TI Writer Tips - TI Users Group of Will County

TI WRITER

The following handy TI-WRITER commands are reprinted from the 99'er News published by the TI Users Group of Will County, Romeoville, Ill. This puts the most used commands on one page for handy access at your computer.

```

=====
EDITOR COMMAND {FCTN:CTRL: EDITOR COMMAND {FCTN:CTRL: EDITOR COMMAND {FCTN:CTRL:
-----
Back tab          :      : T !Ins. Blank line : 8 : 0 !Quit          : = :
Beginning/line   :      : V !Insert character: 2 : G !Reformat       : | :2orR
Command/escape   : 9 : C !Last paragraph  : |6orH!Right arrow : D : D
Delete character : 1 : F !Left arrow     : 5 : S !Roll down      : 4 : A
Del. end of line :      : K !Left margin rel.: |Y !Roll up        : 6 : B
Delete line      : 3 : N !New page       : |9orP!Screen color : | : 3
Line #'s(on/off) : 0 :      !New paragraph : |8orM!Tab          : 7 : I
Down arrow       : X : A !Next paragraph  : |4orJ!Up arrow     : E : E
Duplicate line   :      : 5 !Next window     : 5 :      !Word tab      : |7orW
Home cursor      :      : L !Oops!           : |lorZ!Word wrap/fixed : | : 0
-----
Load files: LF (enter) DSK1.FILENAME (load entire file)
            LF (enter) 3 DSK1.FILENAME (merges filename with data in memory
            after line 3)
            LF (enter) 3 1 10 DSK1.FILENAME (lines 1 thru 10 of filename are
            merged after line 3 in memory)
            LF (enter) 1 10 DSK1.FILENAME (loads lines 1 thru 10 of filename)
-----
Save files: SF (enter) DSK1.FILENAME (save entire file)
            SF (enter) 1 10 DSK1.FILENAME (save lines 1 thru 10)
-----
Print Files:PF (enter) PIO (prints control characters and line numbers)
            PF (enter) C PIO (prints with no control characters)
            PF (enter) L PIO (prints 74 characters with line numbers)
            PF (enter) F PIO (prints fixed 80 format)
            PF (enter) 1 10 PIO (prints lines 1 thru 10)
NOTE: The above assumes PIO, DSK1.FILENAME, and RS232 are also valid!
      To cancel the print command press FCTN 4.
-----
Delete file:DF (enter) DSK1.FILENAME
-----
Setting Margins and Tabs: (16 tabs maximum)
      L - Left margin      R - Right margin      I - Indent      T - Tab
      Use ENTER to execute or COMMAND/ESCAPE to terminate command.
-----
Recover Edit: RE (enter) Y or N
-----
Line move: M (enter) 2 6 10 (moves lines 2 thru 6 after line 10)
          M (enter) 2 2 10 (moves line 2 after line 10)
-----
Copy:      same as move except use C instead of M.
-----
Find String: FS (enter) /string/ (will look for string in entire file)
            FS (enter) 1 15 /string/ (will look for string in lines 2 thru 15)
-----
Delete:      D (enter) 10 15 (deletes lines 10 thru 15 in memory)
-----

```

Richard
Twynning.

The TEX-Files

Dear TI'ers,

At the moment, my trusty organizer and I are in the Newton building at Nottingham Trent University. "When" I am it's the 22nd of March, and I'm in room 303, which is the main computer room in the Newton building. It's exactly 11:16am according to my organizer, and I thought I would start my article to kill time, while I wait for my Sun SPARC Station to wake up!!!

I've been trying to download a single page of images from the Internet "X-Files" page since 8:30am!!! Since there are some people out there who are interested in the internet, I will tell you what page it is:

<http://www.rutgers.edu/x-files.html>

The way that Internet pages work is a bit like Page Pro, because the text is downloaded seperately from the images, and the images are inserted into their correct positions in relation to the text, as they are downloaded. When I say "download", it's not at all like downloading things from a bulletin board, because from a bulletin board you would have to select download from the main menu, select a file transfer method (XMODEM etc.), and then choose the file to download.

With the internet, everything is done transparently, and it's a bit like reading an on-line book. There are pages of information that can include pictures, or there are pages that contain only pictures, like the one that I'm ATTEMPTING to download!!!!

I say "attempting" because it's taking ages due to the Notts. Trent University's equipment. My sources tell me that they have only got a single 28800bps ("baud") link. My source, who did my course last year, and is now doing his PhD says that they were supposed to be going "ballistic!" ages ago, if not before the start of this year, but that's our educational system for you, still teaching about IBM and intel.

Right, up-to-date now, and before I do anything else, I think I must give an apology for the lateness of the last issue.

Gary, Trevor, and I, had a meeting on what must have been Friday 3rd March if my memory is working correctly. We completed all of the editing and sorting on that night

All that was left to do was a page from Gary, for which we made space, and we included it on the contents page. This was going to be an article on adding a real-time clock to the 4A console, either directly or via the Zeno board. Unfortunately Gary didn't have enough time to complete this, and instead filled the back page with, amongst other things, an apology for a mis-print in the Winter issue.

UPDATE: General Secretary's log,
STARDATE: 22-05-95:

Trevor has now posted all of the Spring issues.

I am typing this the 24th of May. I know it doesn't leave much time to arrange your plans and attend, but great thank now to those who did manage it.

It was very worrying that the Spring newsletters went out so late, because we were hoping to see the maximum amount of members at the AGM.

Those members who are due to renew this quarter, then PLEASE DO.

O.K. So what should I talk about in this article??? I've got so many things to say, that it's a problem where to start. Today's date is the 24th of June.

Recently, I've started including programs with my articles, and I thought that I might not have time to put one in this quarter, but I prepared the first part of this one before the AGM. I thought I would have time to complete the program at the AGM, but time ran out on me then. However, it's just taken me about five minutes to write the data into a little program! I obtained the data by playing a certain tune on a CASIO keyboard a note at a time, and then making a list of all the notes, as I've got little stickers on each key with the relevant TI CALL SOUND frequencies written on them.

The good thing is that I'm not going to tell you what the tune is, but you're not wrong if you think it's got something to do with the change in my title image this quarter! But, when you've realized what the tune is, *DON'T PHONE, IT'S JUST FOR FUN!!!!*

ALSO, I keyed the program in using TI BASIC (not Extended), but I put the GENEVE on GPL speed 1, which should mean it was running at the same speed as a normal 99/4A, but you will notice that the CALL SOUND's use a positive duration, which means they have to last for that amount of time before the next sound is played. If you use a negative duration, the note can be interrupted by a new CALL SOUND command. If there are no more CALL SOUND's to interrupt the current one, then a duration of -400 would last just as long as 400, which is 0.4 seconds.

While you type in the program, I'll get on with my article!!!!!!

```
1 FOR NOTE=1 TO 51
2 READ N,D
3 CALL SOUND(D,N,0,N+800,10,N-20,15)
4 NEXT NOTE
```

```
5 DATA 294,400
6 DATA 440,390
7 DATA 392,400
8 DATA 440,390
9 DATA 523,410
10 DATA 440,600
11 REM
12 DATA 294,400
13 DATA 440,390
14 DATA 392,400
15 DATA 440,390
16 DATA 587,410
17 DATA 440,600
18 REM
19 DATA 698,400
```

I'll just rush over to my book case and check what I haven't reported on before, that I need to report on! Just a mo!

Mmmm! That's left me more confused! Since I haven't had time to do proper articles lately, I appear to have skipped over certain things that I should have spent more time on!

Things such as the Sandbach workshop, and two visits to the U.K. by Richard J. Simpson from Canada, who is a member of Jim Cavanaugh's user group, which is based in New York State. Richard actually lives right on the edge of the Niagara river, a few

The TEX Files - Richard Twyning

20 DATA 659,390
21 DATA 587,400
22 DATA 523,390
23 DATA 587,400
24 DATA 440,600
25 REM
26 DATA 698,400
27 DATA 659,390
28 DATA 587,400
29 DATA 523,390
30 DATA 659,400
31 DATA 440,600
32 REM
33 DATA 294,400
34 DATA 440,400
35 DATA 392,400
36 DATA 440,400
37 DATA 523,400
38 DATA 440,700
39 REM
40 DATA 294,400
41 DATA 440,400
42 DATA 392,400
43 DATA 440,400
44 DATA 587,400
45 DATA 440,700
46 REM
47 DATA 698,400
48 DATA 659,400
49 DATA 587,400
50 DATA 523,400
51 DATA 659,400
52 DATA 440,600
53 REM
54 DATA 294,400
55 DATA 440,400
56 DATA 392,400
57 DATA 440,400
58 DATA 523,400
59 DATA 440,600
60 REM
61 DATA 587,400
62 DATA 440,400
63 DATA 294,800
64 DATA 294,1000

miles from the falls.

I think he's about 12 miles from Jim Cavanaugh, and of course, has to cross a bridge over the Niagara river to get to Jim's place. He does this once a week for a group meeting in Jim's basement!

Once a month, they have a meeting in a local library near Jim, and they have a turnout that puts our AGM's to shame!

I've actually seen a video of one of these meetings which was brought over by RRic on an NTSC tape.

Mark and I first got in contact with Jim and the North West New York State TI'ers when we found his Bulletin Board phone number listed in an issue of Micropendium. His bulletin board was called AmCan Friends, but now, I can't remember the program that it was running on.

Soon after we first began communicating they realized that their program had certain limitations, and decided that they needed to upgrade to something more serious.

They were limited to a certain number of messages in their message base section, and Jim had to keep clearing messages out etc.

This was when Jim said he was upgrading his system to the S&T Bulletin Board, written by Tim Tesch.

After dialling into Jim's board when he'd got S&T installed, I realized that it would be a good idea for us to use it in replace of After Hours, which had gone SAD! For a start, Ed Schaum who wrote After Hours had gone completely IBM'ish!!!

He was running his own After Hours on a sad 386 PC!!! And, he clearly didn't want to give us any support when we tried to get an upgrade version of the TI version of the program, and our only option was to use the version we already had, but it kept causing the group's MODEM to reset itself, and it would keep scrolling round, and wouldn't wait for calls properly. We wrongly thought

that this was a fault of the MODEM at first, but since setting up S&T, we've just realized, that it needs a slightly different cable to be able to tell the MODEM what to do, since it's a VERY high speed MODEM.

Back to Jim Cavanaugh, more on S&T later.

Jim was kind enough to send us loads of disks, with loads of software, and RRic himself bought us more over when he came at the start of this year. I must thank Jim again for sending us all of the disks, and apologize that I haven't sent any in return, but I've been messed about with university all year, learning how to pass a course,

rather than learning about true computing. Our educational system is a complete farce!

I've still got to complete my final year project at the time of writing, which must be finished by August, so I've got about four or five weeks to go! So, hopefully, not long after that is out of the way, I'll be able to sort my life out onto a single Optical Floppy Disk, and free up some standard TI magnetic disks, and send him some things in return. Our main problem is that most of the decent software originates on his side of the pond, so that limits our possibilities, but he better make some disk space to receive my entire collection of MYART and YAPP pictures, if I can recover them from DSDD 80 Track floppy, since their existence on hard disk gets ever more doubtful, the more my ST506/412 drives sit idle.

More about ST506/412 drives later! Stay tuned!

Last October, Gary and I met Richard Simpson in Bath. He was on one of his U.K. excursions with his band of American tourists, and they were due to visit Bath for the day, so we arranged a place to meet him, and before his visit had requested that he and Jim would try and find us three P-Code cards, which he smuggled through customs for us, and brought in a very large bag. They were all boxed, with complete photocopied manuals, for which they had also paid the copyright.

Unfortunately, I haven't had the time to do anything with them, but Mark borrowed mine after Richard's visit earlier this year, and found that the disks were not working.

Mark and Gary had a little bit of a go at trying to sort the problem out at the AGM, since Gary had brought another P-Code card, but even with a new set of disks, it still appeared to be a bit strange.

As I have said, Richard returned to the U.K. around February or March. This year as been so busy, that I can't remember the exact date. I know it was about a week before Mothers Day, because they have it much later in the U.S.A., and he reminded his party of Americans that it was in a weeks time, just to wind them up!!!

I filmed his October 1994 trip, including him showing us his two Royce's Rolls, which is why his name gets shortened to RRic (with two R's!!)

However, my film of this years trip is much longer, and I ran out of tape before I ran out of batteries.

He has a busy schedule whenever he's over here. He's president of the Sherlock Holmes Society, so he was making a presentation to Jeremy Brett who plays Sherlock Holmes in the ITV series. He also has an elderly aunt in a nursing home in Sutton in Craven in Yorkshire, which is somewhere near Skipton. This gave us the chance to meet him this time. He visits her every time he's here, so we promised to pick him up at Leeds railway station, and drive him up to see his aunt. This gave us chance to chat in the car. In October, my dad's car was playing silly sods, so we had to struggle back home with it and had to make the trip in Little Nellie, my trusty Peugeot 205.

For our Leeds trip, my dad's car was in one of its phases where it seems to be working O.K., so he was able to drive us up there, which allowed me to film Richard.

He has loads of information, and told us about some very advanced programs from years ago that I have never seen. We discussed everything, from his usergroup, to the S&T bulletin board.

The TEX Files - Richard Twyning

At that time, we were having problems with the board, but there seemed to be no logical explanation, but all is solved now, and you will read later!

One of the members of Jim's group is Norm Wahl (I hope I've spelled his name correctly.) He's about 82, and they call him "Sparky", because he's a rocket scientist!!!! YES!!! After World War II, when the Allies captured Von Braun and took him back to the states, it was Norm's job to debrief him!!!!

Norm was on the tape that Richard brought over. The first thing on the tape however, was a video he'd made of the Lima show, in Lima Ohio, with an extensive interview with Tim Tesch, personally demonstrating the S&T BBS. Unfortunately, we never get to see Tim's face, as the camera is tightly focussed on the screen all the time, but it was a very effective demo, and on the strength of seeing it, Richard recommended to Jim that they obtain S&T to run their bulletin board with. I think they had a bit of a long wait to receive it, which kept us reassured while we were waiting for it. Tim Tesch is so busy, that it takes a while for him to get around to all of his jobs. He's also at university, and also works part time for Don Walden at Cecure!!! The lucky bar steward!!! If you think that was an expletive in disguise, then you'd be correct!

Also at the Lima show was Cecure's stand, which showed an excellent demo of a Musical Instrument Digital Interface (MIDI) being used to control a large CASIO keyboard, which was in fact the very same model that Gary Smith owns! So, there you go Gary, yet another project for you to consider!

Also on the tape, we get chance to see Larry Tippet, and Harry Brashear

Richard gets into a discussion about ANSI graphics with Tim Tesch, and Tim explains why we can't get them on the TI. Don't be disheartened though, as with all TI projects, it's about as difficult as getting the 9919 sound chip to play digitized sound files!

O.K., since I can't remember explaining ANSI graphics before, I'll do a little bit on it now. O.K. so how do terminal programs work?

There's nothing to it. This will also explain to those people who have the TERMINAL EMULATOR II cartridge and just bought it for the speech capability, like I did when I first got it.

If you ever thought to yourself, I'll try the actual Terminal Emulation option and see what it does. When you get through the options to find that you've got a blank screen with a cursor in the top left, and thought there must be something wrong when you try and type, but nothing happens.

This is perfectly O.K., since it's one of the way that terminal programs work. If you start pressing keys, nothing appears to happen, because the program is not written to display your keypresses to the screen, but just send them straight out of the RS232 card's transmit (TX) line.

O.K., why do they do this? Well, our phone system is so noisy, how do you know if a character has reached the other end correctly? Well, your keypresses are received by the system at the other end, which in the majority of cases will be a Bulletin Board program, and

The TEX Files - Richard Twyning

in our case will be S&T, now that our own Bulletin Board is up and running!

When you dial into a Bulletin Board, it will first transmit a title screen, and then will ask for your name and password. When you type your name, Terminal Emulator II just sends your keypresses to the BBS. The BBS reads and stores your keypresses, but at the same time, it also transmits them back to you, and then Terminal Emulator II displays them on your screen. Therefore, if your keypresses appear correctly on your screen, you know that they have been received by the BBS correctly, because they've gone all the way from your MODEM, through the phone system to the BBS, in and out of the BBS, and back through the phone system to your machine, and onto your screen. If they are O.K. on your screen, then you know that the BBS has received them. If they are scrambled however, then it is highly likely that they are scrambled, and you can retype them.

It's a very sensible way of making sure there are no errors in what you're typing. When the BBS sends back characters, it is known as echoing. However, on TELCO, and other Terminal Emulation programs, it's possible to have local echoing, which means that the program itself prints your characters on the screen at the same time it sends them to the remote system (in this case BBS).

This is not as reliable, and it's much more possible for the BBS to receive scrambled characters, and you wouldn't know about it because the characters haven't made two journeys down the phone line. I must say that this doesn't slow down the communication, because it happens so fast, and it's easily fast enough for you to type.

Right, where was I? Oh yes, ANSI graphics. Well, you know that all characters such as letters and numbers and dollar signs and asterisks have a code to tell them apart, called ASCII (American Standard Code for the Interchange of Information). Besides these displayable characters, there are other codes which are called control characters. Those users with printers will be much more familiar with these. Character 10 for example is the linefeed character, and is placed on the end of every line by the TI Text Formatter, because sometimes it needs to reprint a line without moving the paper up (when overstriking characters). This is why you need to specify a printer file with .LF which turns automatic line feeds off. If you format something to disk, and then reload it into the text editor, you will see that the formatter has put linefeed characters on the end of each line.

Another character that you may know about is character 12, which is Form Feed. If you open a file to the printer as #1, and then do a PRINT #1:CHR\$(12) the printer will move up one sheet of paper if you're using tractor feed paper, or will eject the current sheet, and reload a new sheet if you're using a single sheet feeder.

Character 12 is also relevant for the Bulletin Board though. If the BBS sends you a character 12, it makes your screen clear, so it can clear your screen at the start of each main menu etc.

Other control characters do other things, including character 7 which causes a beep to sound on the receiving end. Try printing this character to your printer, and your printer will also beep!

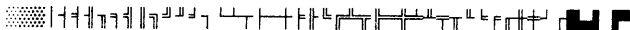
Besides these there are other control characters, which have various uses. You will notice that these control characters are below 32 (space character), but there are other characters above the normal printable characters that finish at 126. Character 127 is the delete character, which not to be confused with the backspace character (which is character 8) but they have more or less the same

use.

On the normal TI system, that's where things more or less stop. In TI BASIC and Extended BASIC we can redefine any characters, including those above 126, but in TI BASIC we can only go as far as about 159 I think, and in Extended BASIC we can only go as far as 143 (because of the memory requirement for sprites in XB).

Since the ASCII character set is eight bit, it allows characters right up to 255. In fact, on the GENEVE with Advanced BASIC from MDOS, and I think even Myarc Extended BASIC II for the 4A and GENEVE will allow you to use characters up to 255.

On PC's however, these characters are defined as standard by the operating system, and they contain what we call ANSI graphics characters, as shown below:



If you're wondering how I managed to get those symbols to appear on TI-Writer, then I will tell you it took NINETY FOUR transliterate commands, which I'm sure was included in Stephen Shaw's excellent snippets of information that he provided on using either TI-Writer, or the Funnelweb editor a few TI*MES issues ago.

Well, where was I? Oh yes. These symbols allow, I'm sure you'll realize, the BBS software programmer, and System Operator (SYSOP) to create fancy menus or even pictures, as you will remember seeing in the Spring magazine.

These are just symbols that are re-defined on the character set from ASCII codes 176 to 223, so by sending them in a certain order, you can make borders for menus, which you can see later in the S&T printouts that I've submitted.

Those users with expanded systems that have access to TELCO can make use of ANSI graphics. Don't worry because my printouts are 80 columns! Even if you've only got a normal 4A with 40 columns, you can receive text in 80 columns and window around on it, just like TI-Writer and Funnelweb editors window their screens. If your printer has IBM ANSI graphics, then you will be able to receive and print the same screens that I've submitted.

But! You still can view ANSI graphics if you ask S&T BBS to only transmit you data in 40 columns. However, that's only with TELCO. Unfortunately, TERMINAL EMULATOR II doesn't have the same defined graphics, so you won't be able to use it on ANSI graphics, but you can ask S&T to supply your menus and title screens without ANSI graphics.

This doesn't mean we're not going to do something for people logging on with TERMINAL EMULATOR II, since TE II will support graphics much better than ANSI, and it will support sound, and speech. The problem is though, that we need to know what codes we need to write into an S&T file, in order to give the desired effect when someone reads it on-line with TE II.

If there is anyone out there who has the information about TE II, we would be very grateful if you would let us know.

Back to ANSI graphics! On TELCO, you can only change the foreground and background colours of the entire screen, but S&T allows you to use ANSI colour graphics, which means that the BBS can also transmit data to change the text foreground and background colour, and when you link this with the ANSI symbols, you can do some amazing things, and generate some amazing menus, and pictures, as you will see with the colour printouts that I've also submitted. When I say colour, it's not possible to include colour in the

<p>TRANSFERS</p> <p>(U)ploads (D)ownloads</p>	<p>AREAS/INFO</p> <p>(A)rticles (B)ulletin (L)ast callers (I)nfo about BBS (O)ther BBS's (V)endor List (W)ant to buy an S&T BBS?</p>
<p>OTHER</p> <p>(C)hat w/SysOp (G)oodbye (\$) Fast Logoff (-) Help!</p>	<p>(E)dit setups (F)eedback (M)essage base (R) Game Doorway (S)ub-boards</p>
<p>*Registered*</p> <p>(c)1995 S&T Soft.</p>	

[b_11...

pm] (? = Menu) Choice :

*Example printout of 40 column
main menu.*

ALT-F10 HELP VT-100 FDX 19200 N81 LOG ON

S&T BBS MAIN MENU

<W>ant to Purchase/Register?	<U>pload File Areas
<I>nformation about S&T BBS	<D>ownload File Areas
ulletin	<M>essage Areas
<O>ther BBS Telephone Numbers	<R> Game Doorway
<V>endors Listing	<S>tory/Humor/Fun Areas
<C>hat with SysOp	<E>dit Your Account
<G>oodbye	<L>ast Few Callers
<\$> Quick Goodbye	<A>rticle Area
<-> Help in using this BBS	<F>eedback / Comment to SysOp

ANSI Colour Menu (80 column)

an] (? = Menu) Choice :

03-110 HELP 07-100 FDX 15200 N01 LOG CLOSED PRY DEF CR

S&T BBS PROTOCOL SELECTION

SINGLE FILE XFER

- (A) View Text File
- (B) Xnoden CRC
- (C) Mass Xfer Ynoden
- (D) 1K-xnoden

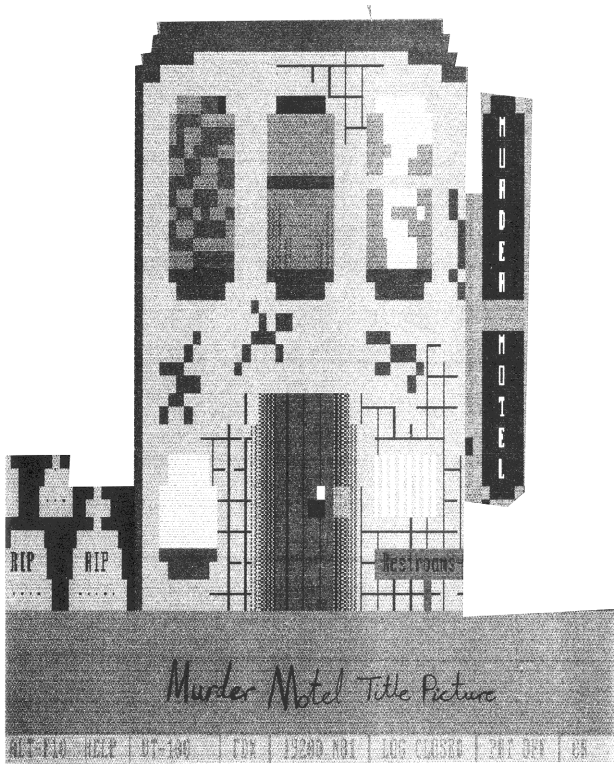
MULTIPLE FILE XFERS

- (E) Mass Xfer Xnoden
- (F) Mass Xfer Ynoden
- (G) Ynoden Hatch
- (H) Ynoden-g

OTHER

- (-) Help with protocols
- (Q) Quit this area

Choice:



magazine, so I've had to print them out in grey scale. Since we haven't got a program yet that does ANSI colour, I had to resort to using my lap top 386 PC running the PC Terminal Emulator called PROCOMM. It does support ANSI graphics symbols, and ANSI colour, but the program itself is nowhere near as good as TELCO. TELCO is the best Terminal Emulator I've ever used, and the fact that it doesn't support ANSI colour, is the only little thing that lets it down.

Before running PROCOMM on the lap top, I ran a program called Screen Thief, which waits in the background until I press CTRL-ALT-T, which makes the program grab whatever's on the current screen, and save it as a GIF image.

I did this when I took the lap top around to Trevor's and connected it directly. I transferred the resulting GIF images over to the GENEVE by copying them onto my organizer, and then brought my organizer upstairs and uploaded them to a TI disk via TELCO.

Once I'd got them on a TI disk, it was an easy job of loading them with YAPP which will read GIF images as long as you've got a Supercart or Superspace Editor/Assembler cartridge.

Then I could easily print them out, with only a slight change to the tables that control the dithering patterns for each colour.

You can see how amazing the ANSI colour screens turn out. Imagine dialling in and seeing them in full colour on your screen.

More about S&T in a while. How long can this article go on for????? Yep, I'm back on my previous form of article writing!!!

Alright, you've been kept in enough suspense over the subject of the front cover. What am I talking about? I suppose you've guessed it's a bit of a play on words, and when I say THEY have arrived, I don't mean grey Reticulans from the star system Zeta 1 & 2

O.K. Some of you will already know what I am talking about. THEY HAVE ARRIVED, THEY DID ARRIVE, THEY WERE, AND ARE.....

SCSI CARDS!!!!

Three Hundred And Ninety Seven days after I obtained the money order for them, and Two Hundred And Twenty Two days after I received my CD-ROM drive, and 33 days after Bud Mills last promised to post the cards, and with Sixty Six days to go until the AGM, the SCSI cards arrived.

I forget what I was doing now, but I remeber the moment distinctly. I was in the lounge one morning, at about 10:15am.

I instinctively rushed to the window when I heard an engine noise, and saw a Parcel Force van heading up the road. I came away

SCSI

Small Computer Systems Interface

Hard and Floppy Disk Controller Card

Copyright 1992 Western Horizon Technologies



HARDWARE INSTALLATION MANUAL
REV "E"

The TEX Files - Richard Twynning

The SCSI card supports co-existence with other hard and floppy disk controller cards. For example, you may already own a Myarc HFDC, it's default address is >1100, therefore you must locate the SCSI card at >1000 or some other unused address.

A second, and most likely the most common configuration, is the use of the SCSI card with a FDC, in this case, the FDC occupies >1100 and the SCSI card should occupy >1000 or any other unused address.

Another configuration may be an HFDC and FDC (TI, CorComp, Myarc, etc.). In this case, the FDC will occupy >1100, the HFDC >1000, and the SCSI card may be set to >1200 or any other unused address.

There are too many configurations to list here, if you are unsure about how to set up the CRU address for your system, please call for help.

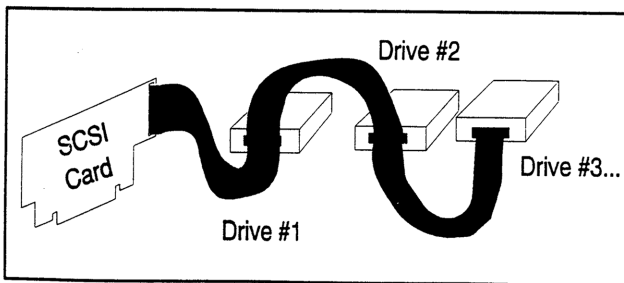
DIP SW2 CONFIGURATION

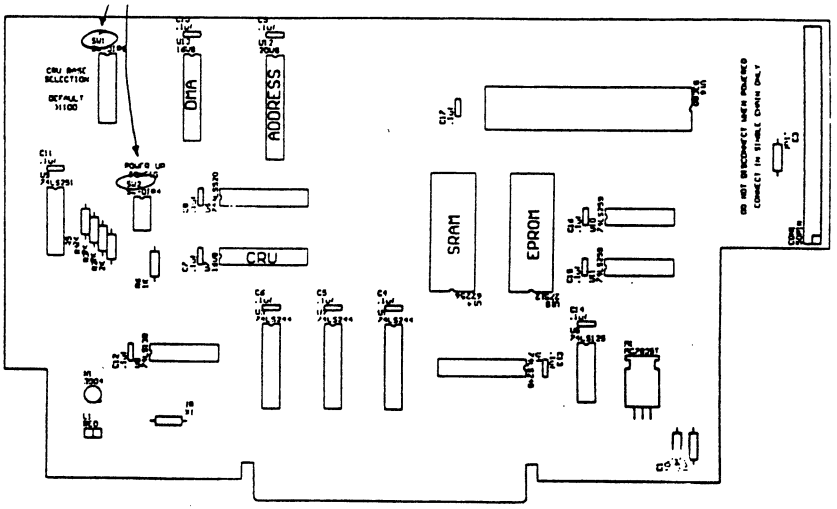
SW2 activates software related functions, the following table explains what these functions are:

FUNCTION	POS	NOTES
SCSI ID SELECT BIT 0	1	Normally ON = ID 7
BIT 1	2	Normally ON
BIT 2	3	Normally ON
System Select	4	ON= Geneve. OFF = TI

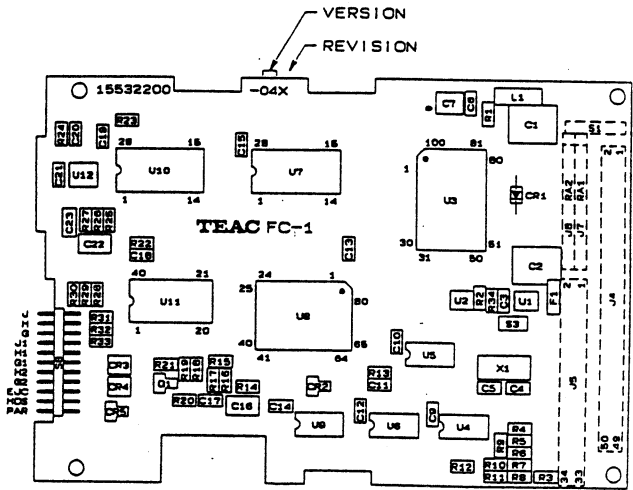
CABLING

Included with the SCSI controller card is one internal/external drive cable. This cable has **3** 50 Pin connectors on it, **one** at one end, and two at the other. To hook up SCSI drives, they must be connected in a single continuous chain. This cable is provided so you can attach up to **2** internal/external devices, or **1** devices and continue the chain off the third connector. If you intend to attach more than **2** devices to the SCSI card, you will require additional cabling, please call or write.





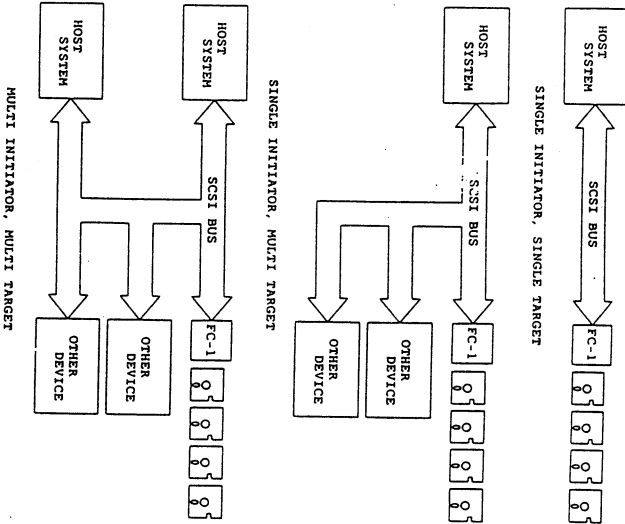
SCSI Card Board Layout



SCSI Floppy Controller

System Configuration

The following system configurations are available with the SCSI interface unit. When more than one target is connected, it is necessary to remove the terminating resistors except for the one on the last target on the bus.

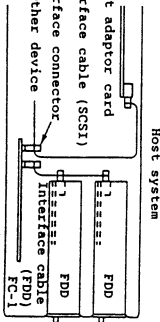


(Fig.5-1) System configuration

Connection to the Host System

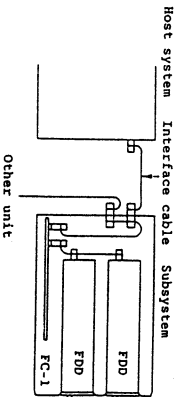
The FC-1 is either incorporated into the host system of the IBM-PC, IBM-PS/2, etc. to connect to the SCSI bus along with other SCSI device or used as a part of a subsystem as shown below.

(a) When the FC-1 is incorporated into the host system using daisychain connection, it is necessary to use an interface cable, the middle part of which is connected to the interface connector. (refer to Fig.5-2)



(Fig.5-2) Daisychain connection within the host system

(b) When the FC-1 is used as a part of a subsystem, run the cable from the FC-1 connector, then install two sets of connectors onto the back panel of the subsystem as shown in Fig.5-3. It is desirable that both of the two connectors are connected.



(Fig.5-3) Daisychain connection when incorporated into subsystem

The TEX Files - Richard Twynning

and told my mum it was a Parcel Force van, but it wasn't coming here, and I sat down again!

A few seconds later, and I thought I could still hear the sound of the engine! I got up again, and I could! It was reversing.

I was almost in shock when the bloke got out and started coming up the drive with a box that is almost identically the same size as the Myarc Hard and Floppy Disk controller box.

I opened it and made sure that we'd got all the bits we asked for! We had both SCSI cards, and both FC-1 floppy controller cards. Unfortunately, we were supposed to find the SCSI cables supplied, but they weren't!!!, but we did get a copy of the new version of MDOS for the SCSI card, along with a disk full of SCSI utilities, including a utility for playing audio CD's!!!

One complaint that I've really got to tell Bud is that he's underselling himself. Once you've heard what the SCSI controlled FC-1 card will do, you HAVE to buy one when you order the SCSI card, and while he said that it's an off-the-shelf item that's manufactured by a third party, he didn't happen to mention that the third party was one of, if not THE best floppy drive manufacturer, TEAC.

It's probably the best floppy controller available today.

It arrived with a little leaflet that gives a very brief technical specification:

1. INTRODUCTION

This is the hardware specifications of the interface unit (hereinafter referred to as FC-1) which controls floppy disk drives (hereinafter referred to as FDD) manufactured by TRAC Corporation via SCSI interface.

For the specifications of the software, refer to "FC-1-11 Software Specification".

2. APPLICATION

The FC-1 is applicable to the following FDD's.

<u>Model</u>	<u>Type</u>	<u>Drives' specifications</u>
FD-35/235F	F	Data capacity 1MB
FD-235HF	HF	" 2/1MB
FD-35/235GF	GF	" 1.6/1MB
FD-235J	JHF (JGF)	" 4/2/1MB (or 4 / 1.6 / 1MB)

- Notes:
1. FD-235 also represents the FD-35/FD-334/FD-335 series.
 2. The data capacity given is one for the MFM unformatted mode.
 3. The track format is IBM format. (excluding drives with data capacity of 4MB.)
 4. FDD signal interface pin number 34 must be DISK CHANGE output.

5. The power supply of the FDD must be switched ON simultaneously with that to the PC-1. Since the PC-1 is overloaded at the power ON and OFF, it is desired that the interval of switching the power OFF/ON is 5 seconds or more.

You will see that I've included a few copies of the documents that come with the SCSI card, including a copy of the cover of the VERY thin SCSI installation manual.

I've also included a copy of the first page of the installation manual which explains a bit about how to set up CRU addresses, and shows a very simplified view of arranging cables and drives.

Don't let this confuse you into thinking it's like the Myarc Hard Disk Controller that will only support three hard disks. The SCSI card will support five hard disks, and DEVICE NUMBER THREE MUST BE THE FC-1 FLOPPY CONTROLLER! The diagram at the bottom of the page is very misleading, because it shows that drive three is a hard disk! DEVICE THREE MUST ALWAYS BE USED BY THE FC-1 FLOPPY CARD, AND, DEVICE SIX MUST ALWAYS BE USED BY A CD-ROM DRIVE!!!!

Since with SCSI, you can attach seven devices (because for some reason device zero is used for internal control), you are then left with FIVE DEVICE ADDRESSES which can be used by hard disks, or as I'm doing, with an Optical Floppy Drive.

There are arguments for and against getting an optical drive, or another removeable media device such as SyQuest. If you get an issue of Computer Shopper, or Personal Computer World, and leaf through, you will see many dealers and mail order companies with full page adverts showing every different type of removeable media, and you will see what I mean by SyQuest.

SyQuest disks are totally magnetic, so if you stick them near a magnetic field you're up **** creek without an outboard engine! You can imagine them as a single plate of a hard disk that you can take out and swap for another one.

The reason for getting an Optical Drive is because I wanted to be extra sure of keeping my TI data protected without ever having to worry about it again, and the reason I wanted something removeable is so I can have it to take to AGM's and workshops.

It's not a good idea taking hard disks to workshops. When running a hard disk, it's a good idea to have a steady table, and not knock them about. Derby is not too bad, but if anyone remembers Cuffley, the tables there might as well have had rubber legs they were so wobbly!

My Optical Drive is the Fujitsu M2512A which is the 230Meg version. I got it for £460 excl. VAT from AG Peripherals in Andover. I've not checked on the prices lately. I think they are around the same, but at the time it was excellent. Some companies were doing the older version for £600, including the con artists (Powermark) that Mitsubishi Finance used to deal with when I was on placement there. They are definitely not a sensible place to go when you want to buy any type of disk drive, hard or floppy.

After the drive, you also need some disks, which are another £150 excl. VAT for a box of five. Yes, it does sound expensive, but that's £30 per disk. Try buying a brand new 230Meg SCSI hard disk for that price, and you would get arrested!!! When you multiply it, that works out at nearly 1.2 Gigabytes for £150.

Yes, with the combined price of the disks, and the drive, you could get a 1.2 Gig SCSI hard disk cheaper, but for another £150

The TEX Files - Richard Twynning

you've got another 1.2 Gig's! So, in that respect, it can be more cost effective to go for an Optical Drive, but then you haven't got the full capacity on line at once. You've only got 230Megabytes at a time, although more expensive drives with a bigger capacity are coming out that use 5.25" disks instead of 3.5" drives, but it doesn't make all that much difference with TI software. One 3.5" 230Meg Optical Disk, if I ever get the damn thing working, will give a capacity greater than any Myarc Hard Disk controller has ever run.

I got two brand new Seagate ST4096's which are full height 5.25" 80Meg ST506/412 drives, and a brand new ST251-1 half height 5.25" 42Meg ST506/412 drive, but while I formatted the 42Meg drive and used it regularly, I've never had one of the 80Meg drives formatted, and formatted the other one once to 60Megs, which was unstable because it was before I'd modified my Write-Precompensation Line and turned it into a Head Select line. After that, I formatted it to 42Megs so I could back up the ST251-1 drive.

It's also a good job I've still got a backup on a box of 80 DSD 80 Track disks (720K each).

It's getting late. It's Sunday the 25th of June at 23:30, and I'm in a bad mood because NASA have called off the Shuttle launch that was going to meet up with the Mir space station, so the program ain't on tonight, and all that the BBC could replace it with is some boring trash about cricket!!!!

I suppose I should do a report on the AGM! You'll have to forgive me for a short report on it, as my time is running short, and one of these days I'll collapse of exhaustion because I've had so many late nights!

The show was surprisingly very well turned out, despite the fact that most people found out about it in such short notice.

Of course, Trevor's opening comments were to apologize for the delay in TI*MES, but I have already explained this at the start of my article.

Things have been relatively quiet. Sorting out and setting up the bulletin board has been the main thing that Trevor has been involved in, and I've been depressed for most of this year. I think Liz Windsor's phrase "Annus Horribillis" would fit it quite well.

Things are looking up now that the final end of my degree is in sight, and I've only got to finish my final year project.

My CAD program WILL run one day. I know what I'm doing with it, and I'm determined to finish it. It's not going to be another PRESS!

Mark has been busy too with his jet setting lifestyle, so he didn't have much to report on.

You'll have to forgive me for not including the groups bank balance, but I've misplaced my copy, and Trevor gave me another copy, and I've misplaced that too!

The balance is around the same as it's been for the last god knows how many years anyway which is around £4400.

Gary was giving up as editor.

Francesco gave his report as module librarian in which he also described a very quiet year.

And, then we were onto the discussion.

The TEX Files - Richard Twynning

The main topic of conversation was about the S&T bulletin board, and the installation of a dedicated line at Trevor's place so that the BBS could be left on without disturbing the peace on Trevor's existing phone line. This was discussed at great length, and was given a unanimous positive vote by all those in attendance.

In the absence of Stephen Shaw, we had to discuss the pricing of the disk library without him. The problem is when ordering LOTS OF DISKS! Last year I ordered around 40 disks, which cost me £62!!!!

This is a bit crazy when considering the value for money you get when buying an Amiga magazine with a 650Meg CD-ROM on the cover!

We decided to have some sort of discount when ordering a lot of disks, and Alan Rutherford was going to speak to Stephen about it. * see note

Next, the group re-elections. Gary was giving up as TI*MES Editor, and Alasdair Bryce was leaving the group altogether, so we needed a new editor, and a new Membership secretary.

After the elections the positions are as follows:

<u>Mr. Chairman</u>	<u>Trevor Stevens</u>
<u>Mr. Vice Chairman and Membership Secretary</u>	<u>Mark Wills</u>
<u>Mr. General Secretary</u>	<u>Richard Twynning</u>
<u>Mr. Treasurer</u>	<u>Alan Rutherford</u>
<u>Mr. Editor</u>	<u>Richard Speed</u>
<u>Mr. Cassette and Module Librarian</u>	<u>Francesco L. Lama</u>
<u>Mr. Disk Librarian</u>	<u>Stephen Shaw</u>
<u>Hardware Specialist</u>	<u>Gary Smith</u>
<u>Programming Advice</u>	<u>Mark Wills again!</u>

* 2022 Note: Now we had a new editor who would print what I submitted I was able to respond fully to RT's comments in Issue 50. After very little that I submitted about the disk library had been printed for some years under the old editor, the new editor printed and sent out the disk library list with Issue 50 - but by then many members had left the group. The disk library was not run for profit, accounts were submitted (but never printed) and initial surplusses were transferred to group central funds. I did not sit there waiting for software to be sent in by donors. I sought it out and paid for disks, and two way postage and encouraged authors directly. This cost funds and as this issue was published the disk library was at a small loss. See Issue 50 for more.... RT wanted everything for nothing - and ultimately as Gen Sec, took it. The valuable disk library was lost. Copying TI disks took a great deal of time and I received nothing from the group. stephen shaw, former disk librarian.

Oh yes, just so none of you feel left out. When Gary gets the time, he's going to fix the MODEMS that I salvaged from Mitsubishi Finance, so we can hire them out to help support the cost of the incoming phone line for the BBS. We had a suggested cost at the AGM of £5 per month, but don't quote me on that. This price was not definite, so keep an eye on TI*MES, and don't neglect the small print!

Don't forget to send your magazine articles to Richard Speed, and all of your membership renewals to Mark Wills.

Right, I'd better conclude now with my bit about the S&T Bulletin board so far, since this article already takes up 221 sectors on disk, and I've only got 20% of my memory left in Myarc MY-WORD which I'm running in normal 4A mode on the GENEVE in GPL mode, from Option 5 of Editor/Assembler.

We voted on having a line installed at Trevor's place for the BBS, and I can report that this has already happened, and I used the Bulletin Board properly for the first time tonight. The phone number is:

01623 

Those who know about MODEMS will need to know that the line settings you need are 8N1, which means Eight Data Bits, No Parity, and One Stop bit. You can dial in with any terminal speed you like, as the MODEM will handle between 300 and 19200 bits per second. If your MODEM is also intelligent, it will negotiate the best line speed, and best COMPRESSION and ERROR CORRECTION method.

I've got the identical MODEM to the group's MODEM, since I bought them at the same time, and you will see from my submitted printouts of my BBS logs that the MODEMS negotiated a CONNECT speed of 16800 bits per second, whereas my TELCO terminal speed was set to 4800 bits per second.

It makes no difference to the board though if you access it with TE II at 300 baud, with a 300 baud MODEM, as long as you've got 8 data bits, No parity, and 1 stop bit.

I forgot to mention that at the AGM, Francesco brought Richard Sierakowski's entire collection of TI gear. Some of you will know that Richard used to be the official UK distributor of MYARC, and for a lot of small U.S. software distributors, but he's made great contributions to the TI community over the years, and helped me with problems on many occasions.

He made considerable contributions to Peter Brooks' group, and TI-LINES, and attended as many Bloxwich Workshops as I can remember.

I didn't realize that I was going to be put in complete charge of the gear at the show, and in charge of dividing it up and selling it, but Little Nellie was definitely weighted down on the return journey to H.Q.!!!

I will complete a list of the gear remaining, and hopefully find time to include a list in the Autumn issue of TI*MES.

The TEX Files - Richard Twynning

Amongst this equipment was Richard's MYARC Hard And Floppy Disk Controller, and a 65Meg drive, which we managed to obtain for the group to provide a suitable base with which to run the S&T Bulletin Board, since it supports both the MYARC HFDC and now the SCSI card.

The reason we needed the hard disk controller, is because the BBS needs a fast access disk system so that users aren't kept waiting too long, but also, the file uploads and downloads wouldn't work with the TI Floppy Disk Controller, because it uses some VDP memory for managing its drives, and due to the complexity of S&T BBS, it needs just about all of this particular VDP area for its own uses, because it swaps out the memory in this area to disk, and uses it as a buffer for the uploaded or downloaded data from the file currently being transferred. Since the TI controller is using this memory, it gets REALLY confused, and causes the BBS to give an error of TRANSFER FAILED!!! With the hard disk controller we corrected this problem, but unfortunately, we had considerable difficulty trying to format the hard disk.

Gary's monster power supply would not even turn it, so I have lent Trevor my Switch Mode power supply, which was what I always used to run my Seagate ST251-1 half height, but it has no trouble running the 65Meg full height.

I must thank Richard Speed for promising to stick us an old salvaged IBM PC XT power supply in the post.

Trevor would like me to give you his apologies for the lack of "From The Chairman's Chair" in this issue, but he has been busy trying to format the drive. We spent several nights hacking away at it, and because something wasn't quite right with the drive, it kept contaminating the HFDC's 8K cache with crap, which then caused it to overwrite the MYARC Disk Manager 5 floppy disks!

I had a message on Friday night from Trevor that he had managed to format the drive, and discovered that the 34-way ST506/412 address edge connector on the drive was slightly warped (at least I think it was the 34-way address connector, or it could have been the 20-way data connector. It's late and it's not really important!) Anyway, he applied some solder on the ends of the pins to thicken them up so they would make a better connection with the springy contacts in the plug, which is a very good tip, and then he placed the plug on so that it was making contact with the very edge of the connector, and Fanny's your uncle, Bob's your aunt! It worked!!!!

After he gave me the news that it had formatted yesterday morning, he said he was going to work, and would have a go at formatting it all the way yesterday night, and he managed to do it.

The final format is:

32 sectors per cylinder
1024 cylinders
7 heads

$32 * 1024 * 7$ gives a total of 229376.

Multiplying this by our standard 256bytes per sector gives us 58720256bytes, or just over 58.7Megabytes!

Today he phoned again to say that the BBS was fully installed,

The TEX Files - Richard Twynning

and ready for me to try! I dialled in, and debugged one miniscule problem which was an erroneous full stop on a path name which can easily get confusing when you're talking about several levels of paths on hard disk systems.

The BBS caller doesn't have to worry about this though, since it's all handled by the S&T BBS.

Tim Tesch is just working on a colour ANSI terminal program for the 4A and GENEVE, so today I accessed with TELCO using standard ANSI graphics.

Since Trevor had to totally re-install S&T from scratch when putting it on the hard disk, the user account file has to be totally built from scratch, so I had to put myself back on as a new user. This is a VERY good thing though, since I've logged the whole procedure to my printer, and included it, so you can see what's involved.

One quick tip. When the program asks for your address, it only requires your house number and street. You get chance to put your town and post code in seperately! I had to correct mine after I'd entered it.

Trevor will be putting the BBS on at the following times to get an idea of the initial response:

Fridays: 6pm until 10pm

Saturdays: 10am until 10pm

Sundays: 10am until 10pm

And, don't forget the phone number:

01623 [REDACTED]

Oh well, it is VERY late now, and getting earlier! It's 1:01am on the 26th June 1995, so I'd better well and truly end my article, since I've only got 7% of my MYWORD text memory remaining!!!!

All for now
from
Richard Twynning.

OK
ATDT 01623 [REDACTED]
NO DIALTONE
ATDT 01623 [REDACTED]
OK

RING

NO CARRIER
ATDT 01623 [REDACTED]
CONNECT 16800 LAPM COMPRESSED

P.S. Unfortunately, the BBS will not be available during August.

P.P.S. Extreme thanks to our new editor Richard Speed for providing a power supply and another 40 Meg Hard Disk.

The BBS In Action - Richard Twynning

Name : The S&T Software BBS
Baud : 9600 v.32/v.42 bis MNP
Hours: 24 hours a day
SysOp: Tim Tesch

Enable IBM Graphics and/or COLOR ANSI if available. This BBS supports Email SOFTmail, online games, multiple message and file areas, Xmodem CRC through Ymodem-g, multiple MXT x/modem, and more! Support for the TI and Geneve computers is our specialty, but all are welcome!

Last caller was Trevor Stevens #2

[C]heck userlog [N]ewuser [G]oodbye [I]nfo [L]ogon : N

Are you sure [y/N] ? Y

* Please Wait *

The Following Questions MUST Be Answered Truthfully For Full Validation!!

** NOTE: Most Questions Require Answers of THREE Characters _Minimum_

Enter Your First and Last Name:

[Richard-Twynning--]

Enter an Alias (real name for none):

[Bluesman-----]

Enter a 3-8 character password:

~~XXXXXXXXXX~~ ← censored!

Enter your mailing address:

[24-Peel-Road7-Mansfield7--Netts---]

Enter your City and State of residence:

[Mansfield-----]

Enter your ZIP code (5 or 9 number):

[NG19-6HB--]

Enter your area code and phone number:

[01623-

Which Terminal Emulation are you using?

1) Adm3A

The BBS In Action - Richard Twyning

- 2) ANSI
 - 3) VT-100/VT-52
 - 4) Other
- Choice: 22J

WAS YOUR SCREEN JUST CLEARED?

(Y)es, my screen got cleared!
(N)o, my screen did *not* get cleared

? Y

Computer Type:

- <1> Atari
- <2> Apple
- <3> Commodore
- <4> Geneve 9640
- <5> IBM
- <6> TI-99/4A
- <7> TRS-80
- <8> Other

Choice: 4

Display width (4)0 or (8)0 columns ? 8

Display IBM (ansi) Graphics ([Y]/n) : Y

Would you like Color ANSI (y/[N])? : N

To Edit, PRESS:

- (A) Full Name...Richard Twyning
- (B) Alias.....Bluesman
- (C) Password.....XXXXXXXXXX
- (D) Address.....24 Peel Road,
- (E) City/State...Mansfield
- (F) Zip Code.....NG19 6HB
- (G) Telephone...01623
- (H) Screen Clear change
- (I) Computer.....<Geneve>
- (J) Term Width...80 Columns
- (K) IBM Graphics.<On>
- (L) Color ANSI...<Off>

Choice, (S)ave, or ?=menu : S

You will now be given the opportunity to enter a message. Please include

- 1) Where you heard of this BBS
 - 2) Are you a SysOp?
 - 3) If yes, your BBS name/number
 - 4) Any other useful info...
- (Maximum length = 230 characters)

>Am Can Friends Bulletin Board run by James P. Cavanaugh.

The BBS In Action - Richard Twyning

You typed:

Am Can Friends Bulletin Board run by James P. Cavanaugh.

Is this correct ? Y

* Please Wait - creating your account *

* Searching for an ID#

* Preparing message base file pointers

(cQ)=Resumes (cS)=Pause (cX)=Abort

Welcome to Another S&T BBS
Dedicated to the Free Exchange of Fairware & Public Domain Programs & Ideas

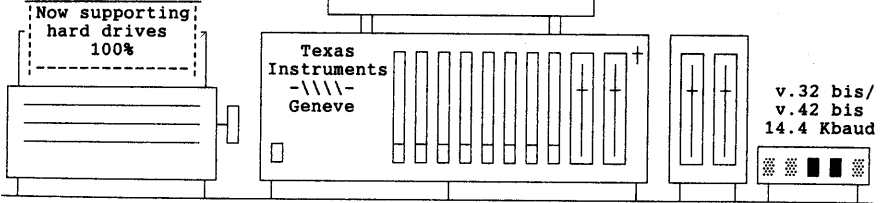
Keeping the TI/Geneve BBS's
competitive without taking
away the joy of operating
one and -without- going to
an IBM compatible system!

Version 01.01.95

(C)1995 S&T SOFTWARE

The <ONLY> software to
support 1K-Xmodem, MXT
multiple files, Ymodem-
Batch and Ymodem-g on
the TI/Geneve!!!!

Now supporting
hard drives
100%



<Press any key>

```
Welcome,           Richard Twyning
City and State     Mansfield
Telephone #        01623 
Expiration Date    *None*
Date last called   pliiii
Last Upload On     *None*
You Uploaded        0 file(s)
You Downloaded     0 file(s)
Messages Written   0
Calls to system    1

System Uploads     3
System Downloads   1
Total bbs calls    7
Last Uploader      Trevor Stevens
```

<Press any key>

The BBS In Action - Richard Twynning

NOTICE: According to the Electronic Communications Privacy Act of 1986 (18 U.S.C. 2510 et seq.), notice is hereby given that there are no facilities provided by this system for sending or receiving truly private electronic mail.

* Message Retrieval *

(A)ll NEW mail, (Y)our mail, (Q)uit : q

Check for new files? [Y/n] n

TRANSFERS

(U)ploads
(D)ownloads

OTHER

(C)hat/page SysOp
(G)oodbye
(\$) Quick Logoff
(-) Help!

INFORMATIONAL

(A)rticles of interest
(B)ulletin
(L)ast few callers
(I)nfo about this BBS
(O)ther BBS numbers
(V)endor Listing
(W)ant to purchase/
register S&T BBS?

AREAS

(M)essage base
(R) Game Doorway
(E)dit your account
(S)ub-boards
(F)eedback to SysOp

Registered v01.01.95

[dldor li pm] (? = Menu) Choice : m

* Please Wait *

* Available Conferences *

- [1] TI FORUM
- [2] TI*MES
- [3] YOU SPEEK OUT
- [4] FOR SALE
- [5] WANTED SECTION
- [6] THE X FILES
- [7] COOKERY RECIPES
- [8] PROGRAMING
- [9] BOOKS
- [10] STARTREK
- [11] YOUR INTERESTS
- [12] GARDENING

Choose a New Area: 2

The BBS In Action - Richard Twyning

OTHER

(C)hat/Page SysOp
(G)oodbye
(M)ain Menu
(\$) quick logoff
(-) help!

RETRIEVAL

(B)rowse the messages
(R)ead messages
(L)ease/post mail

UTILITIES

(A)rea Change
(U)serlist search
(P)ush pointers
(S)etup add/drop
of conferences

You are reading 'TI FORUM' [■■■■■ pm] (? = Menu) Choice : 1

Type the user's number, 0=ALL, or press [ENTER] to search userlog

To-->0

Send to ALL [Y/n] ?y

TI FORUM

p.....ding ' pm

From: Richard Twyning

To : ALL

Subj: Greetings

Enter Message (75 cols/20 lines Max!)

Press (Enter) on blank line for menu.

(-----)
01: Greetings to all in the TI community, and particularly to those trying this
02: Bulletin Board System for the first time. Thanks to Trevor the hard disk
03: seem is now on-line and as you will see in the download area is formatted
04: to around 58.7 million bytes!!!!
05: S&T has taken some sorting out, but now is past the initial stages of
06: debugging. It's given new life to our user group, and will keep the
07: interest going for ages to come. This is just the beginning.
08: -
09: Have fun with this BBS.
10: -
11: All for now from,
12: Richard Twyning,
13: Mr. General Secretary.
14:

<A>bort <C>ontinue <E>dit <L>ist <S>ave <N>ew <V>iew <D>elete <->Help: S

Saving message #2

* Message saved *

The BBS In Action - Richard Twyning

Msg#: 1 *TI FORUM*
a d_ding ' pm (Read 2 times)
From: SYSOP #1
To: ALL
Subj: Your BBS

Welcome to Your BBS run by the TI User Group UK (Sysop Trev Stevens).
I hope you enjoy your trip round the board. REMEMBER start with the Help
screens and READ everything. If you do this you will not go wrong.
*****ENJOY*****

THE MOBB BBS working for the TI User Group and TI*MES

(A)gain, (D)elete, (N)ext, (R)eply, (S)top ? s

* Retrieval Complete *

<Press any key>

TRANSFERS

(U)ploads
(D)ownloads

OTHER

(C)hat/page SysOp
(G)oodbye
(\$) Quick Logoff
(-) Help!

INFORMATIONAL

(A)rticles of interest
(B)ulletin
(L)ast few callers
(I)ntro about this BBS
(O)ther BBS numbers
(V)endor Listing
(W)ant to purchase/
register S&T BBS?

AREAS

(M)essage base
(R) Game Doorway
(E)dit your account
(S)ub-boards
(F)eedback to SysOp

Registered v01.01.95

[.....] pm] (? = Menu) Choice : ?

OTHER

(C)hat/Page SysOp
(G)oodbye
(M)ain Menu
(\$) quick logoff
(-) help!

RETRIEVAL

(B)rowse the messages
(R)ead messages
(L)eave/post mail

UTILITIES

(A)rea Change
(U)serlist search
(P)ush pointers
(S)etup add/drop
of conferences

You are reading 'TI*MES' [`j Compl pm] (? = Menu) Choice : u

The BBS In Action - Richard Twyning

THE MOBB 1995 TIUG Uploads

- [A] TIGAMES1
- [B] TIGAMES2
- [C] TIGAMES3
- [D] GIF PICTURES
- [E] TI EXT-BASIC
- [F] TI UTILS

Select an Area [Enter=exit] : f

* Please Wait *

225226 sectors / 57657856 bytes available

Upload to this area [Y/n] ? y

Filename: [DBASE-ARK]

Please enter a description:

[-----]

>Archived group database file

Almost all terminal emulators will work properly with this BBS. Telex, Qmodem, and other IBM programs have also been tested and work properly. Telco, Mass Transfer, Fasterm, GenTri, and others work fine as well.

You *MUST* use Mass Transfer 4.4 or higher if you want to use Ymodem to upload a DF128 file. Version 4.3 has a bug which has been removed in versions above 4.3. Contact Tim Tesch if you need a TI/Geneve version.

- (X)modem CRC
- (Y) Mass Transfer v4.3 Ymodem
- (Q)uit

Choice: x

Ready to Receive - DBASE-ARK

(c)1995 S&T Soft.

Press CONTROL-'X' to abort transfer
C

* Please Wait *

Successful!!

Upload another file? n

The BBS In Action - Richard Twyning

Your Setup:

(*) Password

(T)erminal Width..80
(I)BM Graphics....On
(S)creen Clear....12
(C)olor ANSI.....Off

(Q)uit/save

Choice: q

* Please Wait *

TRANSFERS

(U)ploads
(D)ownloads

OTHER

(C)hat/page SysOp
(G)oodbye
(\$) Quick Logoff
(-) Help!

INFORMATIONAL

(A)rticles of interest
(B)ulletin
(L)ast few callers
(I)nfo about this BBS
(O)ther BBS numbers
(V)endor Listing
(W)ant to purchase/
register S&T BBS?

AREAS

(M)essage base
(R) Game Doorway
(E)dit your account
(S)ub-boards
(F)eedback to SysOp

Registered v01.01.95

[am] (? am] (? = Menu) Choice : g

Logoff? Are you SURE? [y/N] y



S&T Software copyright (c)1995 by Tim Tesch / Scott Stasiowski

** Thank you for calling! Now giving Ma Bell the boot... **

Thanks for calling THE MOBB 1995 TIUG

Logging you off at: * Thank am

(c)1995 S&T Soft.

Disconnecting...

The Art Of Assembly (Part 6) - Bruce Harrison

The Art of Assembly - Part 6
The Ins and Outs
By Bruce Harrison
Copyright 1991, Harrison Software

Previous parts: Part 3=TI*MES 41. Part 4=TI*MES 42. Part 5=TI*MES 47

As we promised, this part of our series will deal primarily with getting into and out of your Assembly program gracefully. We consider this an important topic, since it can make all the difference when you're writing entire programs in Assembly language. In one book that we used while trying to learn Assembly, a small program example was shown, but there was no way out of the program once it started except the On-Off switch. That shouldn't be.

TI's E/A book gives several ways of returning from programs, but we don't use any of them. Instead, there are two methods that we've used, each of which gets you back where you came into the program from. If you entered from E/A option 3, we'll return you to that screen that says "PRESS ENTER TO CONTINUE" at the bottom. If you entered from XB, we'll send you back to XB with the * READY * and prompt on the screen.

The means of entering a program may vary all over the place, from the very simple LWPI WS to a section of code that re-arranges the locations of tables in the VDP ram, and to even more exotic openings. All the openings have that one thing in common, setting the workspace registers to a workspace of our choosing. As we explained earlier our usual choice is to set the workspace at >20BA, which TI set aside for us to use. This can be used even when programs start from XB, so long as the program only returns to XB upon exit. Utility subroutines for use in XB programs via CALL LINK should always have a self contained workspace. We have found, for example, that the NUMASG utility will corrupt the workspace at >20BA. After returning to XB from an Assembly routine that uses NUMASG and the >20BA workspace, the XB program will break with an error.

In today's Source Code (see below) there are two separate programs, with different entry and exit methods used. Program one is of course not complete, since it needs the subroutine CRSIN and its supporting smaller subroutines given in our last article. You can combine that code with this "shell" and assemble it. When you do that combination, you'll have to delete the line of REFs and the equate for STATUS, from the subroutine's code, and the line at label TENSTR from the subroutine's Data Section. The resulting program will serve to demonstrate the subroutine. It will also illustrate the simplest possible entry and exit for your own programs. The entry simply sets the workspace pointer, then goes about its business. The exit uses a trick passed along to us by Harry Wilhelm. We set the workspace pointer back to GPLWS, clear the status byte, then B @>006A.

That exit method will work whether you entered from E/A or Extended Basic. It may not be necessary to clear the STATUS, but the only way to find out in any particular program is to run it and see whether an error is reported when you exit. If no error is reported, then you can omit CLR @STATUS from this exit.

Our normal practice is to leave that line in, just to be on the safe side. We don't like seeing error reports on the screen, and we're too lazy to go look up their meanings in the appropriate book.

The second program uses a slightly more exotic way of entering and leaving. At the opening, it stashes away the value from R11 of whatever workspace the computer was using, then restores that to R11 of the GPL workspace before doing an RT.

Early in our experiences with the TI Assembly language, we discovered that when you enter your program, the computer has essentially performed a BL operation to get into your program, so register 11 contains the return address you can use to exit.

There are exceptions to this when you entered from Extended Basic, and this method from Program 2 will not always work for XB entry. The first method (B @>006A) will always work, provided only that you first load the workspace pointer with the GPL workspace (>83E0).

That brings us to a very minor point, but one that might be important in some of your programming efforts. In our music programs, we discovered that, for some reason we've not discovered, if one does NOT move R11 to someplace on entry, as in Program 2, the sending of bytes directly to the sound generator at >8400 will not work properly.

We have no idea why that's so, or whether other functions might be affected, but in our music programs we use the entry method of Program 2 and the exit method of Program 1. That keeps everything working.

Both Programs are set up to be entered from E/A Option 3. In the first one, we included the code to define the edge character to look like a space, then proceeded to set up for and call our subroutine. Note that there is no screen-clearing operation here. Since this program does NOT auto-start, but requires you to type in the program name START at the PROGRAM NAME prompt, the screen will be cleared and set to light green for you.

After the subroutine has finished, the program takes the string just placed in TEMSTR and displays it a few lines down the screen. It then calls the subroutine again. This is done simply to give you a chance to see that the subroutine did what was intended. Pressing <ENTER> will get you out of the program and back to the E/A prompt PRESS ENTER TO CONTINUE.

The second program is intended for you to use as a small utility. We wrote this originally for our own use, because many times when we were operating with the E/A module in place, we wanted to print a source code file, but wanted a way to set the printer to skip over the perforations while printing. Before we had a RAMDISK, we kept this program on a disk with EDIT1 and the ASSM1, ASSM2 files. Now, we keep it available all the time on a RAMDISK.

This program does nothing fancy. When it loads from Option 3, it auto-starts and runs the part starting at label SKIPIT. This sets our printer to skip over some lines at the bottom of each sheet. That happens very quickly, so you may not even see the light blink on the RS-232 card. You'll also not see anything happen at the printer, since we've opened the file to the printer with the .CR option, so no line feed or carriage return will go to the printer unless we intend to send one.

The Art Of Assembly (Part 6) - Bruce Harrison

The program will do its job and simply return to E/A, which will place you back at the FILE NAME prompt. If all you wanted to do was set up for skip-over, press Function-9 to get out to the main E/A menu. This small program, however, has another entry point called DOUBLE. If you also want double strike printing, press <ENTER> at the FILE NAME prompt, then type in DOUBLE <ENTER> at the PROGRAM NAME prompt. This will send another three characters to your printer, putting it in double strike and sending a harmless carriage return. That carriage return is sent only so that each thing sent to the printer by this program will contain three characters. If the carriage return were not there, the 10 from the previous three character string would still be present in the VDP Ram buffer, would be sent to the printer, and would cause an unwanted line feed to occur.

The escape sequences we've put into this program will work for all models of Epson, Star Micronics, and Panasonic printers. The number of lines to skip (third byte at label PRNBYT) is ten for us, because of the way we usually have our printer's paper loaded. You may want to change that number to something less, say 5 or 6, before assembling the program. If your printer is some other make, such as an Okidata, you may need to change the escape sequences in other ways. I've run into one printer, called the Olivetti ink jet, in which sending a line feed or a carriage return, or both in either order, will always result in both a carriage return and line feed being performed. Most printers have a DIP switch setting to prevent added line feeds, but not the Olivetti.

This program incidentally introduces the new (for these articles) topic of file management. It Opens, Writes to, then Closes a file. As we've noted in the source code's annotation, there are some shortcuts we've taken here which would not generally be used in file operations. This program does, however, work nicely for its intended purpose. In a later article, we'll get deeper into file accesses, and avoid the shortcuts that were used in this program.

We promised some discussion on the ramifications of using Assembly programs that run from Extended Basic. One could nearly write a book on this topic alone. One of the big problems is this business of the character offset (>60) that one must use when operating from XB.

Strangely enough, it is possible to avoid that offset in XB. In our Word Processor, which was originally designed to run only under E/A Option 3, we avoided needing the offset by switching to the text mode and loading our character definitions starting at >800, where they are located normally when using the E/A module. To do that, we had to perform some VWTR operations, so that VDP would know where its tables were located. This operation is performed not by the Word Processor itself, but by the loader program's Assembly portion, embedded in the Extended Basic LOAD program.

Let's digress into that subject just a bit. The actual Word Processing program is stored on the disk as a series of memory-image files. There are two loaders included in the program disk, one named LOAD, which runs from Extended Basic, and one called UTIL1, which is an Option 5 E/A program file.

Both these loaders contain code to put the VDP into the required setup for the TEXT mode, place a PLEASE STAND BY message on the screen, then load in the five memory image files that comprise the actual Word Processor.

The Art Of Assembly (Part 6) - Bruce Harrison

In the Assembly part of the XB LOAD program, we set up to avoid the need
offset by performing the following:

```
LI R4,32*8>800 Location of space character
MOV R4,@>834A Move that value to FAC location
BLWP @GPLLNK Use GPL Linkage
DATA >001B Load "Small Capitals" characters
TEXMO LI R0,>01F0 Setup for text mode
BLWP @VWTR Place VDP in text mode
LI R0,>074E Setup for screen colors
BLWP @VWTR Write screen colors for text mode
LI R0,>0401 Relocate character table to >800
BLWP @VWTR By writing to VDP register 4
MOVB @TEXMO+3,@>83D4 Stash the text mode byte
```

This last operation, putting the byte at TEXMO+3 at >83D4, is necessary because otherwise the computer will go back to graphics mode as soon as any keystroke is accepted.

Of course the LOAD program does a host of other operations, but these are the key ones. The next-to last two lines tell VDP to look for its character definitions at >800, and this allows us to perform reading and writing of screen characters without that nagging offset. Those who've done work involving Assembly and XB will notice that we've done a BLWP @GPLLNK. XB does not supply such a link vector. Our LOAD program supplies one of those, as well as a DSRLNK. The utility vectors (GPLLNK & DSRLNK) we use are those written and published some time ago by Craig Miller.

That leads into another topic, the use of utility vectors and routines. If a single program is to operate in both the E/A and XB environments, one must also overcome the fact that the nice easy REFS provided by E/As Option 3 are not available. If the program was designed for XB, one can arrange to provide the XB utilities when operating under E/A. Conversely, one can design so that the XB version uses the E/A utilities.

In different programs, we've used both these approaches to closing the utility gap between XB and E/A. That's a topic we plan to explore at some length later in this series. For now, we'll just say that on a disk here at Harrison, we have a file called EAUT and a file called XBUT, so we can get the whole set of either into one of our programs.

When exiting from our WP program, we undo the things done on entry. We reset the VDP to graphics mode by putting a byte of >E0 at location >83D4, then LI R0,>01E0, and perform a BLWP @VWTR.

That's important, because E/A expects the screen to be in graphics mode when it resumes control. If you omit doing this, then return to E/A, the message PRESS ENTER TO CONTINUE, instead of being centered at the bottom of the screen, will be moved right so much that the UE of CONTINUE will be on a separate line. No real harm is done by this, but it's annoying to the user, and so we feel it should be avoided.

That's about all we'll cover today. It's a lot to digest for one sitting, anyway. For those who are serious students of Assembly, we recommend trying the two programs in today's source.

In our next article, we'll get into the subject of Loaders, of the sort we mentioned in passing here. For our own programs, we make customized loaders in each case, and take some liberties with the structure of our memory image files (no file headers), so our methods may be controversial, but they do work.

The Art Of Assembly (Part 6) - Bruce Harrison

```
* TWO PROGRAMS
* PROGRAM #1
*
* A DEMO PROGRAM FOR THE SUBROUTINE CRSIN
* (INCLUDED IN PREVIOUS ARTICLE)
* THIS IS IN EFFECT A SHELL THAT ONE CAN USE TO TEST THE CRSIN SUBROUTINE
*
* REQUIRED REFERENCES
  REF KSCAN,VMBW,VMBR,VSBW,VSBR
* DEFINE PROGRAM ENTRY POINT
  DEF START
*
* REQUIRED EQUATES
STATUS EQU >B37C
WS EQU >20BA
GPLWS EQU >B3E0
*
*
START LWPI WS          LOAD WORKSPACE
      LI R0,32*8+>800  SET R0 TO POINT TO SPACE CHARACTER DEFINITION
      LI R1,TEMSTR    POINT R1 AT OUR TEMPORARY STORAGE
      LI R2,8         EIGHT BYTES TO GET
      BLWP @VMBR      GET EIGHT BYTES
      S R2,R0         STEP BACK ONE CHARACTER, TO THE EDGE CHARACTER
      BLWP @VMBW      WRITE EIGHT BYTES
      LI R0,32*9+2    SET R0 FOR ROW 10, COLUMN 3
      LI R4,20        TWENTY CHARACTERS TO ACCEPT
      LI R15,RTNSTK   SET OUR RETURN STACK IN R15
      BL @CRSIN       ACCEPT 20 CHARACTERS STRING
      LI R0,32*14+2   SET FOR ANOTHER SCREEN LOCATION
      MOV R2,R2       CHECK VALUE IN R2
      JEQ SKIP        IF ZERO, JUMP AHEAD
      BLWP @VMBW      ELSE WRITE THE ACCEPTED STRING HERE
SKIP  LI R4,20        RESET FOR 20 CHARACTERS
      BL @CRSIN       RE-ENTER SUBROUTINE
      LWPI GPLWS     LOAD GPL WORKSPACE
      CLR @STATUS    CLEAR THE STATUS
      B @>006A       RETURN TO GPL INTERPRETER
*
* DATA SECTION FOR PROGRAM 1
*
TEMSTR BSS 21
* THE NUMBER IN THIS BSS MUST BE ONE MORE THAN THE LARGEST STRING LENGTH
* EXPECTED IN THE PROGRAM'S EXECUTION
* FOR THIS TEST, IT WAS SET AT 21 FOR A TWENTY CHARACTER INPUT STRING
      EVEN          SET PROGRAM COUNTER TO EVEN LOCATION
RTNSTK BSS 2        RETURN STACK ADDRESS AT AN EVEN LOCATION
      END
* END OF PROGRAM #1
* ~~~~~
* PROGRAM #2
* SETS PRINTER CONNECTED TO PIO PORT
* WILL AUTO-START AND RUN LABEL SKIPIT
* ENTRY AT LABEL DOUBLE WILL SET PRINTER TO DOUBLE STRIKE.
* REQUIRED REFERENCES
  REF VMBW,DSRLNK,VSBW
* DEFINE ENTRY POINTS
  DEF SKIPIT,DOUBLE
* REQUIRED EQUATES
```

```

*
PABPNT EQU >8356      POINTER LOCATION FOR DSRLNK
STATUS EQU >837C      GPL STATUS BYTE LOCATION
PAB EQU >1000         LOCATION FOR PAB IN VDP RAM
PABBUF EQU >1050      BUFFER FOR BYTES TO BE SENT (VDP RAM ADDRESS)
GPLWS EQU >83E0       GPL WORKSPACE
*
* MAIN CODE SECTION FOR PROGRAM 2
*
DOUBLE MOV R11,@SAV11 STASH CURRENT R11 VALUE INTO MEMORY AT LOCATION SAV11
        LWPI >20BA      LOAD USER WORKSPACE
        LI R1,DSBYTE    SET R1 TO POINT TO DOUBLE STRIKE CHARACTERS
        JMP PRNO        THEN JUMP
SKIPIT MOV R11,@SAV11 STASH CURRENT R11 VALUE INTO MEMORY
        LWPI >20BA      LOAD USER WORKSPACE
        LI R1,PRNBYT    SET R1 TO POINT TO SKIP-OVER PERFS CHARACTERS
PRNO    LI R0,PABBUF     SET R0 TO CHARACTER BUFFER LOCATION
        LI R2,3         THREE BYTES TO WRITE TO VDP RAM
        BLWP @VMBW      WRITE BYTES
        LI R0,PAB       SET R0 FOR PERIPHERAL ACCESS BLOCK (PAB)
        LI R1,PAB2DT    POINT R1 AT DATA FOR PAB
        LI R2,16        SIXTEEN BYTES TO WRITE
        BLWP @VMBW      WRITE PAB TO VDP RAM
        AI R0,9         ADD NINE TO POINT TO DESCRIPTOR LENGTH BYTE
        MOV R0,@PABPNT  PLACE THAT VALUE AT >8356
* THE FOLLOWING LINE OPENS THE FILE
        BLWP @DSRLNK    PERFORM LINKAGE TO DEVICE SERVICE ROUTINE
        DATA 8         DATA FOR DSR LINKAGE
        LI R1,>0300      PLACE WRITE OPCODE IN R1
        LI R0,PAB       SET R0 FOR PAB LOCATION
        BLWP @VSBW      WRITE THE "WRITE" OPCODE INTO FIRST BYTE OF PAB IN VDP
        AI R0,9         ADD NINE
        MOV R0,@PABPNT  PLACE AT >8356
        BLWP @DSRLNK    WRITE THE BYTES FROM PABBUF TO PERIPHERAL (PIO PORT)
        DATA 8         REQUIRED DATA FOR DSRLNK
        LI R1,>0100      PLACE CLOSE FILE OPCODE IN R1
        LI R0,PAB       RESET R0 TO PAB
        BLWP @VSBW      WRITE THE CLOSE FILE OPCODE TO PAB
        AI R0,9         ADD NINE
        MOV R0,@PABPNT  MOVE TO >8356
        BLWP @DSRLNK    PERFORM CLOSE FILE OPERATION
        DATA 8         REQUIRED DATA
        LWPI GPLWS      LOAD GPL WORKSPACE
        MOV @SAV11,R11  PUT RETURN ADDRESS BACK AT R11 OF GPL WORKSPACE
        CLR @>STATUS    CLEAR STATUS
        RT              RETURN (BRANCH TO ADDRESS IN R11)

```

The Art Of Assembly (Part 6) - Bruce Harrison

```
*
* DATA SECTION FOR PROGRAM 2
*
SAV11 DATA 0          PLACE TO SAVE R11 AT ENTRY
* FOLLOWING TWO LINES ARE THE REQUIRED DATA FOR A PERIPHERAL ACCESS BLOCK
* TO OPEN A D/V 80 FILE TO THE PIO PORT WITH THE .CR OPTION
* THE NUMBER >5003 IS PRELOADED SO AS TO PRINT ONLY THREE BYTES - THIS IS A
* SHORTCUT METHOD, NOT FOR GENERAL USE
*
PAB2DT DATA >0012,>1050,>5003,>0000,>0006
          TEXT 'PIO.CR'
DSBYTE BYTE 27,71,13   BYTES FOR DOUBLE STRIKE, PLUS A CARRIAGE RETURN
PRNBYT BYTE 27,78,10   BYTES TO SKIP OVER PERFS ON PRINTER
* NOTE - THE LAST BYTE ABOVE, WHICH WE SET AT 10, GIVES THE NUMBER OF
* LINES TO SKIP - THAT NORMALLY RANGES FROM ABOUT 4 TO 10

* NOTE - THE LAST BYTE ABOVE, WHICH WE SET AT 10, GIVES THE NUMBER OF
* LINES TO SKIP - THAT NORMALLY RANGES FROM ABOUT 4 TO 10
* WE USE 10 BECAUSE WE NORMALLY START OUR PRINTER WITH THE TOP EDGE OF
* THE SHEET JUST ABOVE THE PAPER BAIL TO PUT A BUILT-IN TOP MARGIN ON
* EACH SHEET, THUS MUST MAKE THE NUMBER HERE LARGER
*
          END SKIPIT
* END OF PROGRAM #2 - PLACING THE LABEL SKIPIT AFTER THE END DIRECTIVE
* MAKES THE PROGRAM RUN IMMEDIATELY AFTER LOADING FROM OPTION-3 OF E/A
*
```

[These articles by Bruce are available on disk from the group disk library, they occupy in total 7 ssd disks, or ask for a particular article by quoting the number at the start].

=====

EXTENDED BASIC

XB RND

Walter Allum

RND

Rather to my surprise, some experiments of mine seem to have identified the congruence underlying XB RND:

$$X(N+1) = 14389820420821 * X(N) + 21132486540519 \pmod{10^{14}}$$

where $X()$ is an integer. The random fractions delivered are $X()/10^{14}$. The coefficients all conform to the standards set out in various authorities (e.g. Knuth "Art of Computer Programming, Vol.2; Addison-Wesley 1981). Strictly, we should submit the routine to, say, spectral analysis to verify its statistical performance. But, the signs are good. The expected cycle length is 10^{14} , enough for over 200000 years of continuous use!

Maybe some readers will ask "Why this continuing fuss about RND?". I would reply that, if your only interest was to set initial positions and govern the motions of "invaders", then forget it. But, there are many more serious uses. Knuth names: simulation of real world processes; sampling of situations too varied to examine exhaustively; solving complicated numerical problems; testing computer algorithms; decision making. With these in mind, does it make sense to crow about the superiority of our TIs over mere IBMs while uncritically accepting a facility whose specification nobody seems to know?

Putting It All Together - Jim Peterson

PUTTING IT ALL TOGETHER No. 5
by Jim Peterson

The hard part of learning to program is not in learning what the various commands do - it is in learning to put them together to do what you want them to do!

Key in this little program and run it to see what it does, then read the explanation of how it does it.

```
100 DISPLAY AT(3,1)ERASE AL
L:"SPELLIT" 'by Jim Peterson
110 DATA HIPPOPOTANUS,CRITIQ
UE,KHAKI,IRIDESCENT,ARCHAIC,
PNEUMONIA
```

```
120 !add as many DATA statem
ents as you want
```

```
130 FOR CH=97 TO 122 :: CALL
CHARPAT(CH-32,CH$):: CALL C
HAR(CH,CH$):: NEXT CH :: CAL
L COLOR(9,8,2,10,8,2,11,8,2,
12,8,2)
```

```
140 DATA END
```

```
150 READ M$ :: T=100 :: IF M
$="END" THEN CALL CLEAR :: S
TOP
```

```
160 GOSUB 230 :: ACCEPT AT(1
2,1)SIZE(-28)BEEP:Q$
```

```
170 IF Q$=M$ THEN CALL SOUND
(100,392,5):: CALL SOUND(200
,523,5):: DISPLAY AT(12,1):"
" :: GOTO 150
```

```
180 FOR J=1 TO LEN(Q$):: IF
SEG$(Q$,J,1)=SEG$(M$,J,1)THE
N 210
```

```
190 DISPLAY AT(12,J):CHR$(AS
C(SEG$(Q$,J,1))+32);
```

```
200 T=T+50 :: IF LEN(Q$)=LEN
(M$)THEN GOSUB 230 :: GOTO 2
10 ELSE DISPLAY AT(12,J+1):"
" :: T=T-50 :: J=LEN(Q$)
```

```
210 NEXT J
```

```
220 T=T+50 :: GOTO 160
```

```
230 DISPLAY AT(10,1):M$ :: F
OR D=1 TO T :: NEXT D :: DIS
PLAY AT(10,1):"" :: RETURN
```

Line 100 erases all the trash from the screen and prints the title centered on line 3. The screen is 28 characters wide. SPELLIT contains 7 characters. 28 minus 7 divided by 2 is 10.5, so center the title at column 11. Put in as many lines of words in DATA as you want.

The lower case characters "a" through "z" are ASCII 97 to 122. The upper case are just 32 below that, ASCII 65 to 90. CALL CHARPAT to get the hex pattern identifier of each upper case letter in CH\$, then CALL CHAR to reidentify the corresponding lower case letter to that pattern. The lower case letters are in sets 9 to 12, so color them in the reverse of the normal screen, cyan on black.

The dummy data END in line 140, and the statement in line 150, causes the program to stop without crashing when it runs out of words, regardless of how many you put in. Line 150 reads each word from DATA one after another and sets the initial time to display it at 100 milliseconds.

Line 160 jumps to line 230 to display the word at line 10 column 1, wait for the set time, then erase it by displaying a null string which erases the line. Then it signals with a beep and cursor that it is waiting for your spelling, Q\$, at line 12 column 1.

Line 170 checks whether your spelling is the same as the word M\$. If so, it sounds two notes, displays a null string to erase the word and goes back for the next.

If not correct, line 180 starts a loop for the number of letters, LEN(Q\$), in your

spelling and compares each letter with the letter in the same position in M\$. If the same, it jumps to 210 to check the next letter. But if incorrect, line 190 displays at that point the character of the ASCII 32 higher, which is the same letter in inverted colors.

Line 200 increments the flashing time by 50, then checks to see if the word you spelled is the same length as the correct word. If so, it goes to 230 to flash the correct word, then continues checking letters. When finished, line 220 increments flash time and sends you back to try again. The -28 size in the ACCEPT statement prevents the misspelled word from being erased.

If your spelling has a different number of letters, the first error probably has caused all subsequent letters to be in the wrong position. They would all be marked as wrong, so a null string is displayed to erase the rest of the word. Then you are prompted to try spelling the word again.

Telephone Puzzle Solution - David McCann

Telephone Puzzle Solution, by David McCann

Here's a program in TI Basic to solve the puzzle set in issue 48: it runs in about half the time of Stephen's solution. In Extended Basic it could be speeded up by using multi-statement lines.

```
100 LET ADJ$ = "36987412684248621478963"
110 FOR N = 333 TO 999 STEP 2
120 LET N$ = STR$(N * N)
130 IF POS(N$, "0", 1) + POS(N$, "5", 1) THEN 190
140 FOR P = 1 TO 5
150 IF POS(N$, SEG$(N$, P, 1), P + 1) THEN 190
160 IF POS(ADJ$, SEG$(N$, P, 2), 1) THEN 190
170 NEXT P
180 PRINT "NUMBER IS "; N$
190 NEXT N
```

To get a six-digit square, we must square the numbers from 333 to 999. Half these will be even, but instead of discarding them after they've been calculated, we avoid calculating them in the first place by including "STEP 2" in line 110. Line 130 skips the numbers 0 (not on the keypad) and 5 (adjacent to every other number). The inner FOR loop checks each digit in turn to see if it is duplicated (line 150) and each pair of digits to see if they are adjacent (line 160). The latter test uses the string ADJ\$ which lists all the adjacent numbers on the pad other than 5. If we go round the sides of the keypad, the adjacent numbers are

1	2	3	1 2 3 6 9 8 7 4 1;
4	5	6	going round the diagonals, we have 2 6 8 4 2.
7	8	9	Adding the two and removing the duplicated "1 2" at the beginning gives "3 6 9 8 7 4 1 2 6 8 4 2. But as well as checking for the sequence 3 6, we must also check for 6 3, so we repeat the string backwards to give ADJ\$.

One warning: this program exemplifies bad practice! It keeps setting up a loop and then jumping out of it without telling the computer that it's not needed any more. The correct way out of a loop (in a language without an EXIT command) is

```
150 ... THEN LET P = 5 :: GOTO 170
```

Of course we would then have to avoid printing every number in line 180 by adding

```
145 LET OK = 1
150 ... THEN LET P = 5 :: OK = 0 :: GOTO 170
...
180 IF OK = 1 THEN PRINT ....
```

What a bother — to program you have to know when to break the rules!

Win The National Lottery! - Mike Poskitt

```
100 REM      Programmed by
110 REM
120 REM      M.G.Poskitt
130 REM
140 REM      25/12/94
150 REM
160 REM      for the
170 REM
180 REM TI User Group (UK)
190 REM
200 REM
210 OPTION BASE 1
220 DIM P(30)
230 DIM D(49)
240 CALL CLEAR
250 CALL SCREEN(11)
260 PRINT "THE NATIONAL LOTTERY RANDOM": "NUMBER GENERATOR."
270 PRINT : : :
280 INPUT "Draw date(dd/mm/yy) ":DATE$
290 INPUT "How many lines?(1-5) ":LINES
300 IF LINES>5 THEN 290
310 CALL CLEAR
320 PRINT " Office Staff Syndicate": " National Lottery Numbers": " Draw Da
:DATE$: :
330 FOR Z=1 TO LINES
340 RANDOMIZE
350 FOR I=1 TO 6
360 X=INT(49*RND)+1
370 IF D(X)=1 THEN 360
380 D(X)=1
390 NEXT I
400 N=0
410 FOR I=1 TO 49
420 IF D(I)=0 THEN 470
430 PRINT I;
440 N=N+1
450 P(N+6*(Z-1))=I
460 D(I)=0
470 NEXT I
480 PRINT : : :
490 NEXT Z
500 PRINT "1>PRINT 2>MORE LINES 3>QUIT"
510 CALL KEY(O,K,S)
520 IF S=0 THEN 510
530 IF K=49 THEN 570
540 IF K=50 THEN 290
550 IF K=51 THEN 700
560 GOTO 510
570 OPEN £1:"PIO",OUTPUT
580 PRINT £1:"Office Staff Syndicate"
590 PRINT £1:"National Lottery Numbers"
600 PRINT £1:"Saturday ";DATE$: :
610 FOR Z=1 TO LINES
620 FOR I=1 TO 6
630 PRINT £1:P(I+6*(Z-1));
640 NEXT I
650 PRINT £1: : :
```

£ = #

£ = #

Win The National Lottery! - Mike Poskitt

```
660 NEXT Z
670 PRINT £1: : :
680 CLOSE £1
690 GOTO 510
700 Q$="quit?---are you sure?(y/n)"
710 FOR Q=1 TO 26
720 H$=SEG$(Q$,Q,1)
730 CALL HCHAR(24,Q+2,ASC(H$),1)
740 NEXT Q
750 CALL KEY(0,K,S)
760 IF S=0 THEN 750
770 IF K=89 THEN B10
780 IF K=121 THEN B10
790 CALL HCHAR(24,1,32,32)
800 GOTO 510
810 CALL CLEAR
820 PRINT "GOOD LUCK!!!!": : : : : :
```

£ = #

840 END

- The algorithms used in devising this program are seriously advanced. After careful consideration of previous winning sequences, they have been totally ignored, and numbers are picked at random, just like the real thing!
- Bear in mind that this program is for free. There are many commercial offerings available for other lesser computers, but before parting with your money, ask yourself this question: If they work, surely the authors could well afford to give their programs away?.....
- My program runs in both TI Basic and TI Extended Basic, though probably not at the same time. When typing it in, for "£" type "#" (no, I categorically deny being too lazy to flick my printer's DIP switches. I deliberately left this "bug" in the printout because most other listings seem to have it and I wanted to be consistent and avoid confusion. Honest.)

- Lines 350-390 Generates 6 different numbers from the range 1 to 49.
- Lines 410-470 Prints the 6 numbers to the screen in ascending order.
- Line 450 Stores each number for subsequent printing.
- Line 570 Puts the parallel printer on standby ready for printing the (winning?) numbers. Change the PIO if your output device is different.
- Lines 610-660 Prints out a hard copy of your numbers.
-
- **GOOD LUCK!!!!**
-
- **RCTN-QUIT**

FOR SALE

Malcolm Berry has the following equipment for sale:

Hardware

- TI-99/4A Console
- 2 TI Joysticks
- Peripheral Expansion Box
- Disk Controller Card
- RS232 Card
- Internal 5 1/4" Disk Drive
- Voice Synthesiser

Cassette Software

- Starter Pack 1
- Starter Pack 2
- Games Writer Pack 1
- Games Writer Pack 2
- TI Teach Yourself Extended BASIC
- List File (Database program)

Books

- TI Extended BASIC
- Getting Started with the Texas TI99/4A
- The Texas Program Book
- Learning to use the TI99/4A Computer

Everything includes original manuals and is in good condition. Malcolm also has all issues of TI*MES from number 3 onwards which he is willing to part with.

The hardware is being sold as a complete item for £100, and offers will be accepted for the cassette software or books.

Please contact the editor, Richard Speed, on 01403 730836 for further information.

Help Needed

I have a question from a member regarding a Tandy TRS printer - the CGP 115. Apparently there was a review of this printer back in issue 7, so if anyone can send me a copy of issue 7 for copying I'd be grateful (it'll be returned of course!).

The member is also looking for a technical manual for this printer; my own enquiries have turned up the Tandy technical desk in the US who apparently still stock all technical manuals for Tandy printers (even ones not made for over ten years). The number of this technical hotline is 800 442 2425. Since it is a US freephone number you may have some problems getting through, but the operator should be able to assist.

In the meantime, do any other members have any information or manuals for this printer?

Please contact the editor.

APPEAL FOR INFORMATION

Some members may be using, or have used, Assembler in conjunction with XB in the following circumstances:

- 1) The XB cartridge is left plugged in to the console throughout so that the Assembler cannot be derived from the official E/A package or Mini-Memory, but comes via Funnelweb
- 2) Assembler routines are called from XB and require transfer of data both ways

I can manage so long as the Assembler routines are self-contained as regards initial data and can arrange their own printout but I cannot while (2) applies. As members will know, the transfer business within the E/A scheme is facilitated by a utility BSCSUP but this does not come with Funnelweb. However, I got a copy from our Library. It comes as an object module without the associated source. When I try to load it via XB, an "Unrecognized Character" message comes up from CALL LOAD

The XB Manual says this means a bad field in an object file accessed by LOAD. This is clearly due to BSCSUP and I have my suspicions about one field 3 03AO UTLTAB which seems to arise from using REF which, if I am not mistaken, is not supported by XB.

All this surmising and going down blind alleys could be cut out for me if someone who has done it all will tell me how. I will, of course, gladly refund postage and replace/return disks etc. sent to me with evidence.

Walter Allum
8 Newbridge Road
Tiptree
Colchester
Essex CO5 OHS

Welcome to the back page... a bit like the Twilight Zone really, except a lot cheaper to produce and lacks the nee nah nee nah tune.

I hope you'll all be patient with this, my first issue as Editor - obviously there'll be some mistakes and the page format I've chose to use makes it a lot harder to correct them, but I hope will make it easier to find articles a give a clearer layout. It seemed such a good idea back in June to pre-print page numbers and descriptions and then cut and paste articles in... doesn't seem so good now after spending twelve hours painstakingly trimming articles to fit!

I hope you'll all be using the BBS - if you do, and begin using the conferences you should know about Smileys... :-) Turn the magazine sideways and look at the list below, and you'll see what I mean:

- : -) Happy Smiley
- ; -) Winking Smiley
- : - (Sad
- : - 0 Surprised (The SCSI card turned up!)
- * : -) Silly haircut
- 8 -) Smiley wearing glasses
- : - / Undecided

The variations are endless! :-)

Shame on you if you didn't go to the AGM, there was a live demo of

the BBS and most interestingly, the new 80 column card from Germany - an easy way to eyestrain if you use a television with your TI like I do!

The Evil Laughter award goes to Gary Smith - "You don't know what you've let yourself in for" he cackled as he passed over the TI*MES box of bumpf and the editorial PEB (that was then dismantled shortly after in the search for a power supply).

There was also a demonstration of immense self control from our chairman and general secretary when they discovered that there a wasn't a powerful enough PSU in the building to start the 65Mb hard disk up, three dismantled systems later they gave up leaving the air a pale blue colour and the BBS running on two floppy disks.

Anyway, that's the end of this issue - last date for submissions for the Autumn issue is September 1st, please remember to send them to me, Richard Speed, and not Gary. Oh and be sure to try the BBS, it all works a treat!\$""!###

NO CARRIER

State of the Groups Finances - Alan Rutherford

INCOME and EXPENDITURE Statement up to 1st April 1995

Income	93	94	95
Subscriptions	\$1,465.00	\$1,164.50	\$967.50
Interest	\$36.01	\$15.61	\$14.08
Sales	\$136.60	\$0.00	\$47.25
Total	\$1,638.41	\$1,180.11	\$1,028.83
Expenditure			
TT*MES	\$1,072.44	\$999.45	\$524.57
Room Hire	\$70.00	\$40.00	\$80.00
Expenses	\$57.74	\$8.66	\$14.90
Equipment	\$150.00	\$305.50	\$614.16
Doc Library	\$0.00	\$40.00	
Total	\$1,350.18	\$1,193.61	\$1,233.63
Balance	\$4,267.48	\$4,253.98	\$4,049.18

Outstanding

Room Hire - workshop	\$47.00
TT*MES /48	\$170.44
Hardware so.e	\$97.00
Expenses	\$6.91
Donation	\$13.50
	-\$113.85

INCOME

Subscriptions	27-Apr	\$131.50
	8-Jun	\$62.00
	8-Jun	\$387.50
	9-Aug	\$166.50
	5-Nov	\$175.00
	11-Jan	\$95.00
		\$967.50

Interest	Jun-94	\$3.46
	Sep-94	\$3.61
	Dec-94	\$3.52
	Mar-95	\$3.49
		\$14.08

Sales	J. Murphy - console	\$30.00
	AGM - Raffle	\$17.25
		\$47.25

EXPENDITURE

TT*MES 100 copies	(56) Apr	\$140.98	\$18.86	\$159.84
	(60) Jul	\$151.05	\$30.22	\$181.27
	(60) Oct	\$151.05	\$39.41	\$190.46
		voucher		
	(48) Jan	\$134.52	\$35.92	
				\$524.57

EQUIPMENT	Full System - P. Walker	\$400.00
	Software for BBS	\$29.66
	Modern for BBS	\$184.50
		\$614.16

ROOM HIRE/ EXPENSES	AGM 1994 - Room hire	\$50.00
	Workshop - Post/Phone	\$30.00
		\$14.90
		\$94.90

T.I. User Group U.K. 
Quarterly Newsletter

 **ISSUE 49**
SUMMER 1995