





# TISHUG NEWS DIGEST

TiSHUG (Australia) Ltd

TiSHUG News Digest

ISSN 0819-1984

TiSHUG News Digest

May 1988

## Index

	Title	Description	Author	Page #
All correspondence to:  PO Box 214 Redfern, NSW 2016 Australia	Advanced diagnostic command file	Utilities	Paine, John	21
	Archiving	General interest	Arnold, Tom	4
	Book review	Technical Drive		30
	CALL LOAD("CS1")	Software hints	Meldrum, George	5
	Co-ordinator's report	General interest	Buttner, Chris	2
	Communicators	BBS information	Mudie, Ross	7
	DBM file recovery	Utilities	O'Brien, John	22
	DIS/VAR 80 transfers	Forth screens		26
	Forth column	Forth forum <1>	Smyth, George L	25
	Forth words	Forth screens	Volk, J	26
<u>DIRECTORS:</u>  Co-ordinator: Chris Buttner .Tel.(02) 871 7753 Secretary: Terry Phillips Tel.(02) 797 6313 Treasurer: Percy Harrison Tel.(02) 808 3181	FORTRAN	General interest	Harris, D.N	28
	From the bulletin board	Mail to all		7
	Games information	General interest	Brown, Robert	11
	Games information	General interest	Judd, Stephen	11
	Hitch hiker's guide to galaxy 2	Adventure hints	Brown, Robert	13
	Hitch hiker's guide to galaxy 2	Adventure hints	Judd, Stephen	13
	Letter to the editor	General interest	Christensen, Garry L	2
	Moving from cassette to disk	Software hints		24
	Planner calendar	General interest		15
	Potpourri	General interest	Arnold, Tom	12
<u>SUB-COMMITTEES:</u>  Publication: Geoff Trott ...Tel.(042) 29 6629 BBS sysop: Ross Mudie ...Tel.(02) 456 2122 Merchandising: Bob Bunbury ...Tel.(02) 601 8521 Printed Library: Warren Welham .Tel.(043) 92 4000 Software Library: Terry Phillips Tel.(02) 797 6313 Technical: John Paine ....Tel.(02) 625 6318	Program to type in	General loader	Ryan, Chris	17
	Program to type in	Hangman		14
	Programs on the BBS	Club software	Brown, Robert	11
	Programs on the BBS	Club software	Judd, Stephen	11
	Publications library report	Club library	Welham, Warren	10
	Putting it all together #5	Software hints	Peterson, Jim	18
	QS-Writer bug	Word processing	Hare, Tom	22
	RAMcard tip	Back to the menu	Paine, John	5
	Realist's computer glossary	General interest	Tim, Tiny	3
	Regional group news	General interest		31
<u>REGIONAL GROUP CONTACTS:</u>  CARLINGFORD Chris Buttner .Tel.(02) 871 7753 CENTRAL COAST Russell Welham Tel.(043) 92 4000 COFFS HARBOUR Kerr Wells ....Tel.(066) 55 1487 GLEBE Mike Slattery .Tel.(02) 692 0559 ILLAWARRA Bob Montgomery Tel.(042) 28 6463 LIVERPOOL Larry Saunders.Tel.(02) 644 7377 NORTHERN SUBURBS Dennis Norman .Tel.(02) 452 3920 SUTHERLAND Peter Young ...Tel.(02) 528 8775	RGB monitor interface	Hardware construction	Schubert, Peter	9
	Secretary's notebook	Club news	Phillips, Terry	3
	System RAM addresses	Software hints	Mudie, Ross	23
	They're off	General interest	Trott, Geoff	1
	Tips from the PHOENIX TIBBS	Software hints		18
	TiSHUG shop	For sale	Bunbury, Bob	6
	TiSHUG software column	Club software	Phillips, Terry	4
	TK-Writer loader modifications	Word processing		8
	Transfer between Forth and BASIC	Software hints	Tietz, Leon	27
	Unpacking Forth words	Forth screens	Dicko	26
VIATEL Younger set Younger set	VIATEL	General interest	Nielsen, R	8
	Younger set	Programs, advent. hints	Maker, Vincent	19
	Younger set	Programs, advent. hints	Mudie, Peter	19

## They're off

by Geoff Trott

### MEMBERSHIP AND SUBSCRIPTIONS:

Joining Fee .....	\$ 8.00
Annual Family Dues .....	\$25.00
Dues Overseas Airmail .....	US\$30.00
or .UK	22.00
or AUS\$	50.00
Publications Library .....	\$ 5.00
Texpac BBS .....	\$ 5.00
BBS Membership:	
Other TI User Group	
Members .....	\$10.00
Public Access .....	\$25.00

### TiSHUG Sydney Meeting

The next meeting will be at 2pm on 7th May at Woodstock Community Centre, Church Street, Burwood.

Note: Be sure not to read this space so that you will not know about all the non bargains not on offer!

Printed by  
The University of Wollongong  
Printery.

I was hoping that you would all get a bit of a laugh, or perhaps just a smile, from my April fool's joke last month. Unfortunately it was not possible to get Australia Post to deliver on Good Friday, so there were some reactions on the day before. It did not seem to amuse the directors very much, although I would like to thank Terry for not being upset by all the extra phone calls. I am sorry if anyone was inconvenienced by the joke, but I am more sorry for those who could not see the joke for what it was. Perhaps I should run a competition for an April fool's joke for next year. The prize would be to see your joke in action, and entries would close on January 1 1989. We had a good one last year, but only the committee had the benefit of that one.

\*\*\*\*\*

As you will read on the front cover, this will be your last TND if your subscription ran out at the end of April. I hope you enjoy this issue so much that it inspires you to renew. We have provided a useful calendar for you, and can promise that there are many more interesting and informative articles to come over the next months. Next month will see the final episode in The Hitch Hiker's Guide to the Galaxy. We are planning a series of Forth articles, and hope to give more on Pascal and c99. Of course you could write and ask for articles on any subject that interests you. Why not give TiSHUG one more year and write and tell the directors (and me) what you want out of the club for the year. No one is perfect, but we all think we are, unless someone tells us otherwise.

continued on page 29

## Co-ordinator's Report

by Chris Buttner

Firstly, thank you to all those who renewed their subscription to the club. If you have not already done so, I urge you to do so NOW. If not, this is the last issue of the magazine you will receive.

Forthcoming events include our birthday celebration in May, when the results of the software competition will be announced, and the full day tutorials in June. At this stage, topics for the full day tutorial will include BASIC programming for the new members and also those who want to brush up; copying of disks from the club library; a BBS simulation for those who are not yet on the BBS and also those who want to learn how to get around more efficiently, and other program demonstrations. A technical team will be on hand to diagnose faulty consoles and tell you if it is worth while repairing, and hopefully give an estimate of cost.

A member of the Hunter Valley Users Group has provided us with the artwork for his EPROM programmer kit. If you are technically inclined, or into "blowing" your own EPROMS, this may well be for you. It handles anything up to 128K (27128 for the technically inclined). If you want to know more, write to the secretary and your letter will be passed on to one of the technical team. (You will need an Editor/Assembler module to "doctor" for this project). Specially designed circuit boards are now being sold through the shop for those of you wanting to build the Clock designed by John Paine. The cost is a very reasonable \$2. These boards are being manufactured by a club member and I am extremely happy to see this self help attitude developing within the club.

Incidentally, if you missed the earlier runs of RAMdisk cards (or want to venture to another one), there will be a further limited supply available shortly (hopefully by the time you read this). Cost is expected to be the same, \$35. These cards are being offered strictly on a first come first served basis.

Are you thinking of attending the TI Faire during EXPO? A number of members have already expressed a desire to go and are looking for some way to cut the costs. Try getting a group together. If necessary, use the BBS to make contact with other members wanting to make the trip. The offer of billets was made by the Brisbane Users Group, but in fairness to all, if they are going to assist in this way it is up to you to let them know as soon as possible.

If you are a country member intending to come to Sydney for the full day tutorial and would like us to cover some specific topic, please get your "order" in. Time is running out for us to co-opt someone to run the sessions and also provide any necessary background material. ○

## Letters to the Editor

Dear Sir,

For those who do not know me, my name is Garry Christensen. I am the Co-ordinator of the TI Brisbane User Group, and I have the responsibility of organising the 1988 TI-Faire. A lot of work has gone into the Faire and the time has come to let everyone know what is happening. Let me start by saying that one significant reason for hosting a Faire at this time, was that Brisbane is the site for EXPO 88. This offered us a unique opportunity to bring together the TI users from throughout Australia and to give them the opportunity to see the EXPO while they were here.

What is the Faire all about? The Faire provides a central venue for the display of hardware and software that is available for our beloved computer and it gives the users an opportunity to discuss all aspects of the TI99/4A with fellow enthusiasts. With this in mind, we have invited almost all the companies that support the TI computer in the United States to attend and we have invited all the User Groups in Australia to come along also.

Internationally first: Lou Phillips from Myarc will be at the Faire. The Geneva 9640 computers will be in Australia by then, so there will be plenty of them on display. Lou will no doubt be bringing something new for us to see. The other companies cannot afford to send a representative to Australia, but some have been most co-operative. Rave 99 will be sending some demonstration units of their IBM keyboard interface and the memory enhancement card. They may also be sending a sample of the interface card to allow the speech synthesizer to be placed in the PEB. Both Asgard and Texaments will be sending a selection of their software to be demonstrated. All of the demonstration products will be for sale to those who attend the Faire.

The User Groups are invited to set up a booth at the Faire and demonstrate any hardware and software advances that they have made. This part, in my opinion, is the most important part of the Faire. There is some exceptional talent in Australia. Everyone would like to see what has been developed in other States. The User Groups are also invited to sell any of their wares at the Faire. While there is no charge to be levied to set up a booth, we would ask that the Groups selling equipment or software place an advertisement in the special edition newsletter to help us cover some of the costs. Could those members from your group involved in setting up such a booth, please contact me with regard to the advertisement. The costs have been kept to a minimum. While on the subject of the newsletter, I would like to take this opportunity to invite all TI users to contribute an article for inclusion in the newsletter. The subject can be of your own choosing. A review of each of the products demonstrated by your User Group would give those in attendance a permanent reminder of what is available. We want to make this a truly Australian Faire.

We understand that many Groups will not be able to bring their computers with them. I anticipate that TI-BUG will be able to provide a system for each booth. Any visitors who can bring their own would certainly make it easier for us.

For those who have not yet made arrangements for accommodation, I suggest that you do not delay. As you can imagine, EXPO 88 is putting a strain on the hospitality services in Brisbane. Some members of the Brisbane User Group have offered to supply accommodation in their own homes. This number is limited, so if you would like to share the weekend with one of our members, please contact me directly. My telephone number is (07)2841841. A list of motels and caravan parks in the area of the venue of the Faire, was published in the December 1987, TI-BUG newsletter.

The venue for the Faire is the Brisbane College of Advanced Education in the suburb of Carseldine. It will begin at 10am and end around 5pm. A dinner at one of the local restaurants will be held on the Saturday evening. We feel it is best not to organise anything official for the Sunday, as many visitors may like to take in EXPO 88, however I am sure that something can be arranged for those who will be at a loose end. If you have any thoughts on the matter or would like to attend the dinner, please contact me.

Well that about wraps it up. This Faire is aimed directly at the Australian user. We hope that this can be a chance for the TI enthusiasts to see what is being done with their computer in this country and internationally.

Regards  
Garry J Christensen

## For Sale

Brother HR-5 Thermal Printer for \$160  
Included in the price are: TI PIO interface cable; spare tape; spare thermal paper roll (A4 by 30 metres); 240 volt power supply. This unit is little used and is as new. It is surplus to requirements.  
Also for sale; Brand new Editor Assembler for \$50  
Contact Bert Thomas, (047) 54 1535 any time.

## Secretary's Notebook

by Terry Phillips

Only one new member to welcome to the group this month, and he is Joe Bottemanne from Mt Colah. Welcome Joe, and I trust you enjoy your stay with us.

I would like to thank Bill Walker, a member from Hallidays Point, for taking the trouble to send down a disk full of technical files on the TI. I have passed copies of this disk to both Geoff and Ross, and hopefully they will be able to use Bill's contribution. Bill also passed on this bit of information which may be of assistance to members who have, or maybe will encounter the problem.

Recently I constructed a twin disk drive system using information supplied in articles in TND. I encountered quite a bit of trouble with my system. All worked 99% of the time but had a problem with DV/80 files. I did find that the DSR chip in my TI controller card was faulty, and on replacement of the controller card, with one in an all metal casing (the original was plastic), my system would not work at all. After many hours of trouble shooting, the matter was solved by making up a shielding for the cables from PIO interface to the drives and controller. I would like to make the point, shielding is a very critical thing for all connections along with clean connector pins.

Thanks for that bit of information, Bill.

This month marks the 7th anniversary of our User Group, so I guess a little bit of reminiscing would not be out of order.

For those who do not know, the group was formed in May 1981 by Shane Andersen, and a group of willing "volunteers", who had attended a meeting in his flat, in William Street, Kings Cross. This writer happened to be one of those volunteers, having only just, a matter of a couple of weeks before, hired, yes hired, a TI99/4 computer and converted TV set from Canberra Television. The cost at that time for purchase was in the vicinity of \$1500, but CTV would let you hire the system for around \$35 a month. Anyway at this inaugural meeting, the decision was made to form the group and hold regular monthly meetings and the first two of these were in fact held in Shane's flat. The numbers attending soon put paid to holding them at that venue, so a move was made to the small upstairs room at St. Johns Church Hall just up the road in Victoria Street. It was not long before we had again outgrown this hall, so a move was made to the large downstairs area, a spot we occupied for a number of years.

It was during this time that our membership peaked, at just on 1000 fully paid up members. While such a number of members is a good thing in a way, I certainly sympathise with the Secretary from those days, as even now with a membership of 300, the workload is enough to keep you fairly busy.

Some highlights of those peak membership days at St. Johns, were the full day tutorial workshops in BASIC and Extended BASIC tuition, where each participant was encouraged to bring his own system. At one such event, I counted something like 70 TI99/4As, all fired up. What a sight!

Other highlights included games playoffs, where I recollect a young member scoring well over a million points playing Parsec! As you can imagine this took him some considerable time to achieve.

These were also the days when most members had only a tape system and it was not uncommon to sell upwards of 100 tapes at each meeting. Indeed disk sales were somewhat of a rarity. Other popular items were the old 99'er Magazine, name changed to Home Computer Magazine and any books or programming information we could get, someone would buy them.

The move to Burwood came about two years ago, primarily due to complaints the then Committee was receiving about the problems with parking at Kings Cross, its "reputation", and because of a perceived shift in membership to the Western areas of Sydney. It cannot be denied that we are fortunate in having such a beautiful venue as Woodstock in which to hold our meetings.

This was also a time of change. Certainly over the past two years the expertise of our technically minded members has been passed onto the general membership, and I would suggest that most members now have 32K expansion in some form or other, while those with full systems probably support one or more RAMdisks.

When we stop and think, we should all be thankful for the Group. Without it we would be on our own, and would all have probably sold or trashed our TI99/4A a long time ago. Let us hope the years ahead see TISHUG flourish as it has in the past. o

## Realist's Computer Glossary

by Tiny Tim, UK

reprinted from TI\*MES, TI99/4A Users Group UK  
**BASIC:** Computer language used for generating error messages.

**Extended BASIC:** Similar to BASIC, but with more error messages.

**Assembly:** Very complicated method of crashing your system.

**Mini-Memory:** Yes, except for the price

**PRINT:** Program statement that displays information in the wrong screen position.

**DISPLAY AT:** Similar to print but quicker

**ARRAY:** A quick way to lose track of stored tabulated data.

**DIM:** How you feel when trying to sort out multi-dimensional arrays.

**Expanded System:** Refers mainly to TI's projected profits. (Of historical interest only).

**Speech Synthesizer:** Small talking box that teaches you spelling in American.

**Cassette System:** Slow method of storing programs you will probably never use again.

**Disk System:** Faster method of storing programs you will probably never use again.

**END:** Statement you type when you think your program is complete.

**EDIT:** Method of deleting "END" and inserting five more program lines.

**Cursor:** Small blob on the screen that flashes in time with your swear words.

**NEW:** The random keys always found by your young son while you are on the phone.

**NUM:** What you become until you try a softer chair.

**RANDOMIZE:** The word we used to spell with an "S"! OR it shakes up two small foam-rubber dice somewhere in the console.

**Port:** Lets air into the side of the console. Much discussed by hardware buffs.

**CALL SOUND:** Laborious method of generating music with wrong notes.

**RS232:** A complicated technical method of failing to get a printer to work.

**Modem:** A device that fails to connect two computers by telephone.

**CALL CLEAR:** Blanks the screen to get ready for the next error message.

**QUIT:** The key you gently brush with your sleeve after 3 hours typing.

**PRK:** A module that failed to put this Glossary into alphabetical order.

**DEF:** Used in combination with other words by programmer's wife.

**LET:** Reserves space inside the computer, as in "Room to LET".

**INT:** Exasperates computer by removing all those decimal places it sweated over.

**Sprite:** Small graphic that moves across the screen, almost totally out of control.

**PI:** To do with CHIPS or something.

**RAM:** Male sheep with a good memory.

**RUN:** Command that the manual says will get your program going.

**OPEN:** A statement that cuts the computer off from the outside world.

**CLOSE:** Oh, all right then.

Retyped for TEXPAC BBS by John Ryan of TISHUG. o

## TISHUG Software

Column by Terry Phillips

## Archiving

by Tom Arnold, Channel 99ers

New disks added to the Library over the past few weeks include the following:

- A182 - A music disk containing some popular songs from recent times, all programmed by Jim Peterson. If you like well programmed music and the words to sing-a-long with on your screen, then this disk is for you. It has a nice load program, which offers multiple choices. 358 sectors and in Extended Basic.
- A183 - BEAXS, the latest version direct from the author, Paolo Bagnaresi. The disk now contains some documentation files which are interesting reading. The disk is 595 sectors and requires a full system with a minimum of one disk drive. BEAXS, as some of you may know, is an Editor/Assembler substitute, with a few enhancements added by the author. BEAXS, by the way stands for Bagnaresi Editor Assembler for eXtended BASIC System. With this, and a copy of the Editor/Assembler manual, you as good as have the TI Version. According to the author, this version is now Geneve compatible.
- A184 - BASIC BUILDER, also from Paolo, this utility converts "list" files to Extended Basic program format and while there have been several similar utilities around, this one works well and quickly. The disk also includes all source files and documentation, as well as a sample program. The disk occupies 680 sectors and you will need a full system with at least one disk drive.
- A185 - GENEVE DEBUGGER, also from Paolo, this is an enhanced version of the TI DEBUGGER but now usable on the Geneve. All source files are included on the disk which occupies 386 sectors. Full system needed.
- A186 - BA WRITER V1.5, another one from Paolo. This updated version is now Geneve compatible and occupies 720 disk sectors. This utility, as you are probably well aware, is a TI-Writer substitute and requires a full system.
- A187 - MY-ART, this disk will not work on my system, so I presume it is for the Geneve. 267 sectors.

As you can see a lot of the above software was received from Paolo Bagnaresi, who responded in the following manner, following receipt of our donation to him.

Dear TISHUG,

Thank you for your letter. What a surprise inside the letter. I have never expected to receive a donation from so far away. Thank you so much.

Paolo also advised that he has stopped working on his TIBM transfer program due mainly to the number of similar programs on the market. He is also currently not planning on producing any more programs with a commercial aim, due to lack of sales and/or donations. Paolo would welcome letters from members and can accept exchange disks in any format. His address is Via J F Kennedy 17, 20097 San Donato, Milanese, ITALY.

Also from Jim Peterson, accompanying his music disk mentioned above, a big thank you for the donation he received. Jim put it to good use apparently, and purchased a Super XB Module. Unfortunately he is currently having trouble with this and his system in general, and coupled with a recent spate of bad health, has really upset his corresponding program. Hope all works out well soon Jim. Jim also sent a disk full of his series of Tips, which I have passed onto Geoff for publication in upcoming issues.

At the shop for the May meeting will be disks A182, A183, A184 and A186 and on tape the musical disk A182. If you have a need for a specific program from the library, please write, phone or leave a message on the BBS to username SECRETARY or TERRY and I will arrange copying to tape or disk, and if you would like a copy of the complete software library catalog, on either disk or in printed form let me know this too.

More news on the software scene next month. ☐



Archiving, what the hell is that? A little imitating maybe? Well you are going to have to live with it whether you like it or not. However, it really is not all that difficult. To begin with, archiving is the squeezing together of files so that they can be transmitted faster or fit on a disk more compactly. There are many good reasons for doing this. The most important is when using bulletin board services. Time costs money, especially if you are hooked up to The Source or CompuServe. By compacting files we can transfer more data in less time and so save money. (Not applicable to TEXPAC which supports DIS/VAR 80 files only). Another use is for storage on disks. If you send disks to friends by mail you will be able to get more on your disk, thus saving postage. As I am sure you will come across archived files I thought you would like an explanation of how they work.

The first archive program that I saw, was Barry Traver's Archiver, which was written in Extended BASIC, slow but effective. He soon came up with an assembly language version (I think it was Bob Boone who did this, ED) which was much faster. Several other versions are around but the best and most recent is Barry Boone's Archiver II V2.3. If you do not have this one, get it from our library. It has several features the others do not have, including a Catalog option, Archived File Catalog and Return to FunnelWeb. Most importantly, it archives in two stages so that you can make your files smaller still.

The best way to find out how this program works is to take a disk with half a dozen files and programs on it and archive it. Follow these steps:

- 1) Load up the Archiver II V2.3 with the E/A module (or FunnelWeb) and select #5 - DSK1.ARCL.
- 2) Select the option PACK FILES and then name the Disk Drive you wish to pack. Answer "yes" to "packing all files". Send the output to another fresh disk in drive #2. Make sure the disk is blank so that you have room for all the files. Give the packed file a logical name. Usually at this stage the files are given a name, such as, NAME\_ARC, the \_ARC indicating that the files have been packed. The packed files will be in the form DISPLAY FIXED 128.
- 3) Now select the COMPRESS option. Name the output file, such as, NAME\_ARC, the \_ARC indicating compressed files. When you come across these file names you will know you will have to do a two step procedure to unpack your files. Compressed files have the structure INTERNAL FIXED 128. Answer "All" to pack all the files. The file names will automatically be written to your disk. The compress option only asks for the respective disk drives, make sure you have enough space on your disks.
- 4) Now that you have packed the files, try unpacking them. you should end up with the same files you started with.

Handy hint: The intermediate files (the packed files) are not really needed and are used only temporarily. I used the RAMdisk to write to, so that I saved time and do not have to worry about changing disks. For example, if unpacking, I decompress files to the RAMdisk from disk drive #1. Then I unpack from the RAMdisk to disk drive #2.

The space saving using this double packing is quite significant, but varies with the types of files you are unpacking. Program files do not shrink as well as text files. One example I tried went from 352 sectors to 233 sectors, where another of 66 sectors only went to 54 sectors (it was two program files).

So next time you come across a file name with the suffix \_ARC or \_ARK then get out that Archive program and do not forget to send Barry Boon a donation. His address is:

Barry Boone  
Box 1233  
Sands Springs, OK. 74063  
U.S.A.

From the Ottawa TI User's Group Newsletter, March 1988. Retyped by John Ryan of TISHUG for Texpac BBS. ☐

## CALL LOAD("CS1")

by George Meldrum, TISHUG

My friend, Rolf Schreiber, suggested to me one day, a new way to load machine code programs from cassette. Being a disk jockey, Rolf was most familiar with loading assembly code by typing in the command: CALL LOAD("DSK1.filename"). His next step was to replace the disk name with "CS1". Sure enough cassette prompts burst onto the screen with an insatiable urge calling for input.

Perhaps we should back pedal a little at this stage. The machine code programs currently supplied for cassette users encompass some form of BASIC program which either has machine code embedded within it, or uses BASIC statements to poke code directly to memory. These systems work well and are quick to use, which may lead you to ask, "why worry about yet another way to load machine code?"

Here I have to admit that the advantages of CALL LOAD("CS1") are subtle while the drawbacks are substantial. The advantages are:

- 1) Ability to load "relocatable" machine code. Should you require several assembly routines, then each can be loaded, one at a time, without problems of one routine overwriting another.
- 2) Ability to load machine code routines without disturbing any BASIC program already in memory. For example you may have a BASIC program in memory and suddenly decide that you would like to include a screen dump routine. Load the screen dump program with CALL LOAD("CS1") and your BASIC program in memory remains untouched.

Having resolved that there is some merit in this new system of loading machine code, we come to the ultimate problem. The TI Assembler program, which produces the machine code in the first place, will do so for disk but not for cassette! Further more, the disk data file produced is a DISPLAY FIXED 80 type, while CALL LOAD("CS1") expects a DISPLAY FIXED 64 file.

More can be learned by looking at the disk file produced by the assembler. By observing the disk object files through the Editor, it becomes apparent that each record, i.e. each line, can be reduced down to 64 characters. The Editor/Assembler Manual states that uncompressed object code consists of records containing up to 71 ASCII characters each. Of the many object files that I have personally viewed none contained lines of more than 69 characters. Most records do in fact contain more than 64 characters but they are easily reduced by deleting the 5 character checksum. Note that a line of code which includes a checksum is readily accepted from cassette. The only reason for eliminating the checksum is to shorten the line to fit within the DIS/FIX 64 boundary.

To remove the checksum, load the object code produced by the assembler into the Editor/Assembler Editor. Move to the right hand side window. On the far right is the sequence number preceded by some blank characters, which in turn are preceded by the object code.

```
...4A9BD3A87F131F      0001
...BA5A47F196F        0002
...BAA127F1BFF        0003
:
```

The last tag character F is needed to indicate the end of the record. The preceding five characters are to do with the checksum. The four characters before the last F form the actual checksum value and before that, the tag character 7 indicates that the checksum follows. So delete the 7 and the four characters following.

```
...4A9BD3A8F          0001
...BA5A4F             0002
...BAA12F             0003
:
```

The last few lines of the file will not need editing as they are always very short. Take note of the last sequence number on the extreme right hand side, so you know how many records there are. Save the edited file to disk as a DISPLAY FIXED 80 type.

The last step is to transfer the newly edited object file to tape, perhaps with the following program.

```
100 REM DATA TRANSFER DISK TO TAPE
110 !
120 DIM C$(*)
130 OPEN #1:"DSK1.filename", INPUT, DISPLAY, FIXED 80
140 FOR I=1 TO *
150 LINPUT #1:A$
160 C$(I)=SEG$(A$,1,64)
170 NEXT I
180 CLOSE #1
190 !
200 OPEN #2:"CS1", OUTPUT, DISPLAY, FIXED 64
210 FOR I=1 TO *
220 PRINT #2:C$(I)
230 NEXT I
240 CLOSE #2
250 END
* Replace this symbol with the number of records in
the file, i.e. as noted by the last sequence number
when editing the file.
At this point you have the finished product which
can be loaded to memory thus:
CALL INIT
CALL LOAD("CS1")
```

To execute the assembly routine you need to know the link name, for example, CALL LINK("SDUMP")

Essential, of course, is the 32K memory expansion and the Extended BASIC module.

I did mention earlier that there were drawbacks to loading machine code by CALL LOAD("CS1"), and so here they are:

- 1) It is slow. It takes one minute to load nine records. A program which has 54 records takes 6 minutes to load while the same routine loads in 30 seconds from a machine code implanted BASIC file.
- 2) If the user presses his cassette play button when told to do so by the screen prompts, then he risks missing the first one or two records. This is a common fault when loading cassette data. You need to wait for the screen to scroll up one line before pressing the play button.
- 3) It is more troublesome to copy a data file. A short program does do the trick, so perhaps I am being a bit lazy. For those who do tape to tape copying, then this is not a problem. ○

## RAMcard Tip

with John Paine

I have a number of useful little Extended BASIC routines and programs sitting on my RAMdisk, which are loaded from the John Johnson Menu loader. The only flaw in this method of loading is that when the BASIC program has finished, I have to either do a CALL MENU or hit the reset button to reload the menu. After again reading the documentation on the MENU Program, the answer to my problems was there in black and white.

DELETE "MENU" or OPEN #1:"MENU" in a running Extended BASIC or BASIC program will immediately reload the MENU program. Use these statements in place of the END statement or in the last line of your application and bingo... Menu.

The following example is the listing of a very simple program that prints documents for me and is well worth having on the RAMdisk as a permanent utility (2 sectors of space) and when used with the print buffer (256K), reloads the menu and allows me to do other things whilst the printer chugs along.

```
Note the last line
100 OPEN #1:"PIO.",OUTPUT
110 INPUT "FILENAME->":A$
120 OPEN #2:A$, DISPLAY, VARIABLE 80, INPUT, SEQUENTIAL
130 IF EOF(2)THEN 200
140 LINPUT #2:A$ :: PRINT #1:A$
150 GOTO 130
200 CLOSE #1 :: CLOSE #2
210 DELETE "MENU"
```

I have now added this statement to every Extended BASIC routine on the RAMdisk as well as some normal disk based files. An example is Ross Mudie's SENDMAIL program for the BBS. ○



# TISHUG NEWS DIGEST

## TISHUG Shop with Bob

G'day. I am Bob Bunbury, the new shop co-ordinator. Services have been pretty bad for the last couple of months, and for that I apologise. Cyril's act is a hard one to follow. With the help of a few good guys at the meetings, and some late night telephone calls to all and sundry, I am just about ready for "business as usual". So please come and see the happy Shop staff at the General Meetings at Woodstock. Or send your mail orders to:

Shop  
PO Box 214  
Redfern  
NSW 2016

or:- BBS, mail to SHOP

or, as a last resort, you can try catching me at home, (02) 601 8521. That is the introduction done! Now to business.

Firstly, we are currently carrying a large stock of back issues of TND/SNDs, and MICROpendiums. Other near dead stocks that YOU (members all) asked us to carry include the Technical manuals, TI-Writer manuals and Editor/Assembler manuals. Popular projects get delayed (or not started) because the directors and the Shop committee have been fried in leftover stocks quite often. Those of us on the "want" lists for RAMdisk pcb/part kits bear witness. Stop moaning Bob! Sell some gear! OK.

Orders are now being taken for yet another run of PE box RAMdisk printed circuit boards. I do have some 6264LP-12 chips, and just two of Robert Peverill's battery packs. Rush me.

PCBs are now being made for the Real-time clock project. Software is also available now. Kits, including parts, will be made up if sufficient interest is shown. PCB is \$2.00, which is required with order. Parts will naturally be extra, if we run to kitting the project. Rush me again.

A couple of P-BUFF kits still remain. They do not have the required Z80 chip, and the memory chips are packaged separately.

PCB, EPROM, CRYSTAL, 8255 \$51.00  
8 x 41256 memory chips \$40.00  
Z80 chip - easy to find at around \$5.00

A power supply, your skill and dexterity, and \$96.00 is all you need to free you from that long wait for your printer to let go of your computer.

Other hardware items are:-

Peter Schubert's Mini-PE items

a. DS/DD controller \$190.00  
b. motherboard plus one option \$ 85.00  
c. 32k, PIO, RS232 option: extra\$ 50.00  
d. Painted box for Mini-PE \$ 35.00  
PE Box cards e. DS/DD controller only \$240.00  
f. Multifunction card \$390.00  
(RS232,PIO,32k,DS/DD controller)

\*\*\*\* Orders will be taken and forwarded on, if we are out of stock on Peter Schubert Hardware!  
Grom Ports temporarily out-of-stock normally \$12.00  
Ivory Console Cases \$2.00

TI joystick handles (and rubbers,actuators,cases) 50c per piece

Plastic Boxes (ex Dick Smith 2505) \$9.50, 2508 \$9.00  
9 V/14-0-14V transformer \$6.50 (NOT \$15.00 as I misquoted at the March meeting.

Modules: CART WRITER, VIATEL1 or 2, DIAGNOSTICS all \$25.00 \*\*\* Grombuster required for 1983 consoles  
And the books:

MICROpendiums \$3.00 June 1986 to March 1988  
TND/SND selected issues \$1.00 Sep 84 to current  
Fun & Games for the TI-99/4A last copy at \$15.00  
Manual, Feb/Mar 85 Tutorial day \$1.00  
Manual, Technical Data \$15.00  
Manual, TI-Writer \$15.00  
Manual, Editor Assembler \$28.00  
Manual, TI LOGO Curriculum Guide (with disk or tape programs. Requires LOGO module \$10.00  
Software

Tapes in stock 87/11  
87/11A  
87/12  
87/12A  
88/01 all \$3.00 each

Disks in stock ROS 7.3 (ramdisk operating system)  
A170 clock routines  
A168  
A169  
RLE PICTURES 1, 2, 3, 4, 5  
88/1  
40 column utilities  
Graphics Labeller  
on demand Funlwriter 4.0  
Disk Aid; all \$5.00 per disk.  
Additional disks SS/SD format \$3.00  
Picasso ver 2.0 Publisher \$20.00

\* Please remember that this is NOT a public domain program. Arto has earned our money AND our congratulations for this program. Updates available for registered purchasers. Otherwise, pay up.

\* Please state your disk format requirements when ordering. And please allow a little extra for postage.

## For Sale For Sale For

### PE box cards for sale

TI RS232 card \$100  
TI Disk controller \$70  
Horizon? RAMdisk 256K, Reliable and fast storage \$250.  
AT DD Controller card \$240  
AT Multifunction card, RS232/32K/Disk controller all on one card, \$s off with trade-ins on old TI cards.  
Modem Card 300 baud cheap BBS access \$??  
Complete Expansion system, TI PE box with RS232 card, 32K, DD Disk drive (360K), DD Control Card, all for \$700.  
Stand-alone VIATEL modem box with extra RS232 printer port. Plugs into side of basic Console, complete with phone plug and power Pluggak. Special Custom Demo unit, looks neat, \$220.  
Peter Schubert (02)358 5602  
P.O.Box 28, Kings Cross, 2011

### For sale

1 complete TI99/4A expanded system.  
Consisting of:-  
Console, modulator, power supply and PE box containing RS232 card, 32K memory expansion card, TI disk controller card with SSSD disk drive. Also with it: a pair of joysticks, Disk Manager 2 module, all original books and manuals plus some extras and the original second disk drive cable.

Price -- \$690

General items

TI Reference Guide, Beginner's BASIC, \$5 each.  
Modules: Extended BASIC with books, \$35; TI-Writer (as new), \$60; Terminal Emulator (new), \$60; Mind Challengers, \$10.

Hardware: 2 dual cassette cables, \$5 each; Second disk drive cable, \$10.

2 Consoles with power supplies, modulators and original books, \$85 each.

Contact by phone (02)699 5137 after 5pm, Monday to Sunday.

### For sale

TI99/4A with books, power supply, VHF modulator in box, \$75.

TI99/4A with books, in box (Console Only), \$50.

Speech Synthesizer, \$50; Personal Record Keeper, \$10; Dragon Mix, \$10; Addition and Subtraction, \$10; Spelling Level 3, \$10; Extended Basic, \$50; Terminal Emulator II, \$50.

Bernard Skeates, (043)69 3644, Kincumber.

### For sale

TI99/4A computer; Peripheral Expansion system, double disk system (1 DS/DD), RS232 card plus Centronics printer interface cable, 32K expanded memory card, disk controller

Over 100 disks (D/S) of games and utilities

Modules including E/A with full documentation

2 TI joysticks + 1 Pointmaster + 1 Commodore joystick with adaptor

Various software manuals

All for \$1500. Must sell quickly. Contact Michael on (043)90 9648 for more information.



## The Communicators

## From the Bulletin Board

Special Interest Group for Users of the TEXPAC BBS.  
by Ross Mudie, 12th April 1988.

### 1. SUB-EDITORS.

In February this year, Sub-editor facilities were established on the BBS. The Directors of TISHUG all have Sub-editor status and their files can appear in the NEWS menu. Watch out for the Directors' files named CO-ORD - Chris Buttner; SECRETARY - Terry Phillips; TREASURER - Percy Harrison; CYRIL - Cyril Bohlsen and MUSICMAN - Russell Welham.

The Sub-editors who now contribute to the material on the BBS are EDITOR - Geoff Trott; GAMES and REQUESTS - Robert Brown and Stephen Judd; LIBRARY - Warren Welham; SHOP - Bob Bunbury; SOFTWARE - Terry Phillips and TECHOTIME - John Paine. These Sub-editors are providing updated or new information, normally on a monthly basis. On the top of each Sub-editor file the BBS provides the details of the date and who uploaded the file. This allows users who are reading the files, to check in the first line if they have previously read the file.

If any other BBS user would like to contribute as a Sub-editor then please contact me. The value of the contributions provided by the Sub-editors to the members of TISHUG is considerable. The benefits to be gained by the whole TI community is totally dependant on the continued involvement of the membership and this is one positive way to communicate your contribution.

Two of our members, John Ryan and Alistair Leslie are helping by retyping selected material from overseas magazines. This material is placed on the BBS and some of the files are reprinted in the TND.

### 2. PROGRAM UPLOAD/DOWNLOAD AREA.

This area, when initially provided in February, was quickly filled to capacity. Since then the area has dwindled in usage. Any user can upload a BASIC or Extended BASIC program into this area to share it with other BBS users. It is preferable to limit programs to less than 46 disk sectors and the size limit is 48 disk sectors. As well as the program, you can provide a file to tell other users about the program. Programs are left in the area for a month to allow others an opportunity to download. This is a really effective way to share your software and programming talents with others.

### 3. REGIONAL GROUP NEWS.

There are 2 areas in the BBS where information about the Regional Groups appear. The MEETINGS file gives details of TISHUG and Regional meetings in brief format. The BULLETIN file is used for bulletins from TISHUG and reports from the Regional Groups. To make this work effectively, information from all the Regional Groups is needed on a regular basis and the groups need to do some forward planning of meeting dates and topics. The editor of the TND also uses the BBS as a source of information for the magazine, which is TISHUG's voice to the whole membership and many other TI99/4A user groups, both in Australia and overseas.

### 4. BBS MEMBERSHIP.

Members of TISHUG can join the BBS for an additional \$5 per annum membership fee. The BBS is not available for use by non-members. To join the BBS, contact Ross Mudie on (02) 456 2122. To use the BBS you will need a TI99/4A, RS232 card, modem and terminal emulator program, which is available on module (TE2) or disk.

### 5. FORGOTTEN PASSWORDS.

From time to time, BBS users who have not been very active, (and some who change their password and then forget), find it very difficult to log on to the BBS. Help is only a phone call away, as this part of the BBS can be accessed by SYSOP, CO-ORD or SECRETARY.

### 6. CONTRIBUTIONS TO THE TND MAGAZINE.

Your contributions for inclusion in the TND magazine can be sent as mail on the BBS to user name EDITOR. The editor regularly collects his mail from the BBS for the magazine and early submission of material make his job a lot easier. The deadline for placing material on the BBS for inclusion in the next TND issue, is the Wednesday following the monthly TISHUG meeting in Sydney (normally on the first Saturday of the month).

MAIL TO : ALL

MAIL FROM : GOWFAR

A lot of people here are using the TI-Writer to write a letter off line, then they upload it to the BBS. Please, I beg you, do not use save file (SF) when you have finished your letter. Use print file (PF), which allows you to print the contents of the TI-Writer buffer either to DISK or PIO or RS232. Print to disk eliminates that annoying 3=GUUUUUU at the bottom of the letter, as I discovered ages ago, using my BBS. Further, if you wish to eliminate all control codes in the file (LF and CR etc.), after you have typed PF and hit [ENTER], type:

C DSKx.filename

where the "x" is the disk drive number. This procedure eliminates all control codes that TI-Writer uses that are NOT able to be deciphered by users of inferior brand computers. Remember that there are some members, here, who have more than one computer and may be using the trashy one instead of the TI sometimes. Any more information on TI-Writer can also be obtained by reading the manual, too. Hope this helps you (and I hope it gets rid of the "U"!!).

Thanks Greg, I have given up trying to tell people. Ross Mudie, SYSOP.

MAIL TO : ALL

MAIL FROM : GOWFAR

Hi to all. I have just received my March 88 MICROpendium and have read up on the Geneve. Shock, horror!! Myarc, according to MICROpendium, are releasing an Advanced BASIC for the machine which will not run any TI99/4A program that has CALL LINK's to assembly programs in it. What I would like to know, from anyone who has any later information is: What is the use of it if most things (such as REQUESTS imbedded progs) will not work? Am I right in what I read? I run a BBS, as most of you know, on the TI. Myarc said it would work with the Geneve and now MICROpendium seems to say it will not. Can anyone clear up this contradiction for me? I was planning on buying one until I read this! Ta!!

MAIL TO : ALL

MAIL FROM : JOHNV

Technically minded people are required to work in a Computer Related environment. Experience is an advantage, however full training will be given to the right people. Interested people should contact John Vandermyer on 697 8666 Ext 439. 9 am to 5 pm Weekdays.

MAIL TO : ALL

MAIL FROM : SMITTY

Anyone fiddling with TELCO? If you are, maybe you could give me a few ideas on linking macros, as I cannot seem to get that aspect to work using the method suggested in the documentation. The program sure seems fine, but needs incorporating 1200/75 75/1200 rates to adapt it to local conditions. I am thinking of a new auto modem and have these in mind to make a choice from.

- 1 Avtec MEGAMODEM
- 2 Maestro SUPERMODEM (v21,22,23)
- 3 BITBLITZER 123E

The Avtec modem I am considering is the MEGAMODEM123. Enough of my ramblings. Your suggestions would be appreciated.

Regards Peter.

MAIL TO : ALL

MAIL FROM : PETESAKE

FOR SALE \* FOR SALE \* FOR SALE

Stand alone CLOCK for MINI-PE owners. It is in a neat grey box complete with batteries. MBP design, will work with MENU, or loader program, disk catalogue, etc. Phone 358 5602 or leave mail. Will have it at the general meeting.

TI DISK CONTROLLER CARD FOR PE BOX

Please contact Peter Schubert on 358 5602 or leave message on the BBS.



## Using VIATEL

by R. Nielsen, TIBUG

Retyped by John Ryan of TISHUG, February 1988.

### What is VIATEL ?

VIATEL is Telecom Australia's national videotex service, combining two way (interactive) communications, with visual display of text and graphics. The technology includes terminals for user access, Telecom's telephone network, the VIATEL computer database and remotely connected external databases. By pressing the appropriate key on a keyboard, a large store of information can be displayed on screen.

The majority of information on VIATEL is supplied by Service Providers (SPs), who are responsible for deciding what they put on VIATEL, how they present it and how often the information is updated. Service Providers range from private sector organisations through to Statutory Authorities and Government Departments.

### How to join VIATEL

If you have not applied to become a registered VIATEL customer, you can do so by obtaining an application form from your nearest Telecom Business Office. Read the "Terms and Conditions" carefully and complete the details required. Return the Application to the TBO or post it to the address shown on the form. In order to be able to access the VIATEL database, the user requires either a VIATEL Keyboard-Adapter for television sets, or a personal computer with a communications modem, suitable software and a telephone line.

In most instances the existing telephone line will suffice. However, in a very small number of cases, VIATEL reception may be unsatisfactory and upgrading of your telephone service to ensure quality data reception may incur additional charges.

### Important numbers

If you a registered VIATEL customer, you will have received a sealed envelope containing your Customer Identity, Personal Password and VIATEL Number. It is important that you distinguish between these numbers.

#### Customer Identity

Every VIATEL customer has a unique 10 digit Customer Identity number which you will need to access VIATEL. It also identifies you to VIATEL for charging purposes, so keep it safe and confidential. Remember, if you let anyone else use your Customer Identity, you will be charged for their use of VIATEL.

#### Personal Password

As an extra security measure, every VIATEL customer has a unique 4 character Personal Password. Your Password can be a combination of numeric or alphabetic characters. (Remember that the system is lower case and upper case sensitive for alphabetic passwords). It is essential that you change your Personal Password regularly. To change your password, key 0# and follow the on screen instructions. If you should lose your Customer Identity number and/or Personal Password you should report this in writing to VIATEL immediately. The address is:

FREEPOST 20  
VIATEL  
G.P.O. Box 188C  
Melbourne VIC 3001.

Never divulge either your Customer Identity or Personal Password to a third party.

#### VIATEL number

This is your 9 digit "Public Address". It is used for mailbox and billing identification. You should quote your VIATEL number in all correspondence to Telecom. Note: You must ensure that your terminal has a Telecom Authorisation number prior to connecting to VIATEL via the National Telephone Network.

#### VIATEL user charges

There are four basic elements to the charges for VIATEL.

1. A VIATEL service subscription fee  
Business \$12.50 per month  
Non Business \$ 3.50 per month

2. The cost of a local call to the VIATEL database from any location in Australia (this charge will be displayed in the top right hand corner of the Welcome to VIATEL Page, and will appear on your account as part of the frame access charges).
3. A charge for time spent connected to the VIATEL Computer Monday to Friday 8am to 6pm (Eastern Standard Time) of 9 cents per minute. At all other times, the charge is 6 cents per minute.
4. A frame access charge where applicable. This charge is set by the provider of the information and is collected by Telecom on behalf of the Service Provider. Frame charges can range from 0 cents to \$4.99. All indices are free. You may check your user charges whilst on line (key #). The screen will indicate your current call and monthly charges.

### Special VIATEL features

#### Response frame

A response frame is a pre-formatted page set up by a Service Provider to enable VIATEL customers to forward messages, order goods and services etc.

#### Mailbox

The mailbox is a facility which allows VIATEL customers to send messages to each other.

#### Viatex

VIATEL users can receive and send Telexes nationally.

#### Gateway

Gateway gives you access to information which is held on an external computer belonging to a Service Provider via the VIATEL network

For TI users wishing to access VIATEL, the club has software available to Members.

## TK-Writer Loader Modifications

Jackson County 99ers

Apparently, when going from the Editor to the Formatter, the LOAD program reloads the AL program, not checking to see if it is still in memory. The resulting wait can be avoided by making the following modifications to the LOAD program. Replace line 100 and add lines 102,104, and 108. Line 100 checks to see if the assembly program is in memory. If not, it jumps to 108 and loads the program. If it is in memory, the REF/DEF table and last free address are loaded, and you can access the assembly code as usual. In line 108 where it says "DSK1.TKWRITER", put in the program name you are using for TK-Writer.

```
100 CALL CLEAR :: CALL INIT :: CALL PEEK(-2043,A,B) ::
    IF A<>84 OR B<>75 THEN 108
102 CALL LOAD(16360, 85, 84, 73, 76, 73, 84, 250, 212,
    70, 79, 82, 77, 65, 84, 250, 132, 69, 68, 73, 84,
    79, 82, 250, 22)
104 CALL LOAD(8196, 63, 232):: GOTO 110
108 CALL LOAD("DSK1.TKWRITER")
```

The TK-Writer loader program written by Tom Knight is a terrific program, freeing everyone from the necessity of using the TI-Writer module. There is one minor problem, bombing out the program by typing in "SD" for Show Directory to catalog a disk. Once you become aware of this problem, you can avoid it. But here is a fix that modifies the EDITAL program so that if you press "SD" by mistake the program does not recognize the request and returns to the command line with no loss of data. The sector that needs to be changed is the third sector of the EDITAL program. The hex values at bytes >14 and >15 are >53 and >44 (ASCII S,D). Change the >53 to >20 (space). Listed below are the printouts of a portion of the sector before and after the change.

#### ORIGINAL FORM

```
2D 54 53 48 3E OF 2D 54 52 45 3E 2C OC
2F CA 4D 20 3E 84 2F 42 53 44 3C D4 18
2E 4C 53 46 00 00 2E 8A 4C 46 00 00
```

#### ALTERED FORM

```
2D 54 53 48 3E OF 2D 54 52 45 3E 2C OC
2F CA 4D 20 3E 84 2F 42 20 44 3C D4 18
2E 4C 53 46 00 00 2E 8A 4C 46 00 00
```

This change is not very elegant but it does work, if you are using TK-Writer. You will not be able to Show Directory when you go back to your TI-Writer cartridge after making this change. The decision is up to you. If you are only using TK-Writer, it is an easy decision to make.

## RGB Monitor Interface for the TI99/4A

by Peter Schubert, TisHUG

For those of you who have a RGB colour monitor, or would consider getting one, I have just finished building and testing an RGB interface board that plugs into the standard console and provides outputs for an Analog RGB monitor. It is powered from the 12 volt from the modulator socket, and is simple to build using common transistors (8 off) and resistors. It has adjustable resistors for setting colour saturation and balance. Also included is a simple single chip sound amplifier with volume control. A Printed Circuit Board has been made by club member Leo Gorniak and for details on availability, contact club Director Cyril Bohlsen. This PCB can be mounted into a small plastic box with the 3 inch speaker, or directly inside the Console or Monitor.

This design is based on an idea from a Japanese article by M. Shibuya, circa 1982. It has very good colour purity, and in its present form gives truly exceptional performance, which you will have to see for yourself.

### Circuit diagram

Figure 1 shows the complete circuit of the RGB Board. Some monitors may not handle the 'Y' signal from the console directly as was done in this design, and as it is not buffered you will have to remove the 75 ohm termination from the SYNC input of the monitor, as this load is too high for the console video chip output. Figure 2 shows an optional buffer stage that can be added if needed. It requires a 5 volt supply which must be well regulated, and because it must drive a 75 ohm load will draw more current from the console 12 volt than is desired. This is one reason why I did not use it, because it is the 12 volt regulator in the console that makes the console hot.

### Adjustment procedure

Adjust the following potentiometers for colour saturation on Primary Colours;

- VR1 - GREEN
- VR3 - BLUE
- VR5 - RED

Start with all three potentiometers fully clockwise (minimum colour) and slowly turn each anti-clockwise to get the colour. Adjust VR2 and VR4 for purity of secondary colours, in particular CYAN and MAGENTA. It may be necessary to repeat the adjustment procedure to obtain true white and grey.

Note: If your monitor has a 75 ohm termination on the SYNC input, this must be removed by opening monitor and cutting out this resistor. (Unless Figure 2 buffer amplifier is used).

continued on page 10

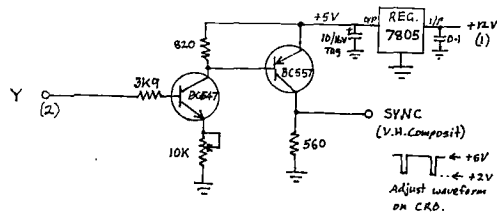
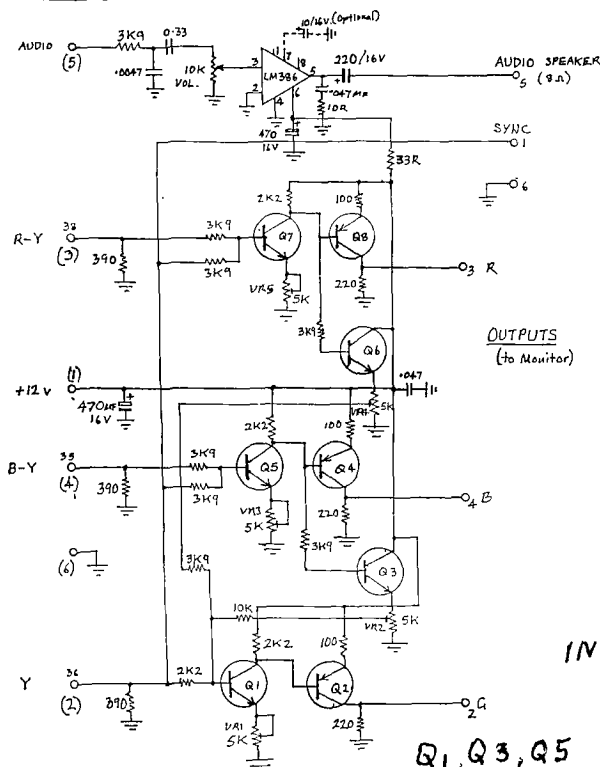


FIG. 2

### TI-99 RGB INTERFACE (Analog)

13-3-88 P. Schubert

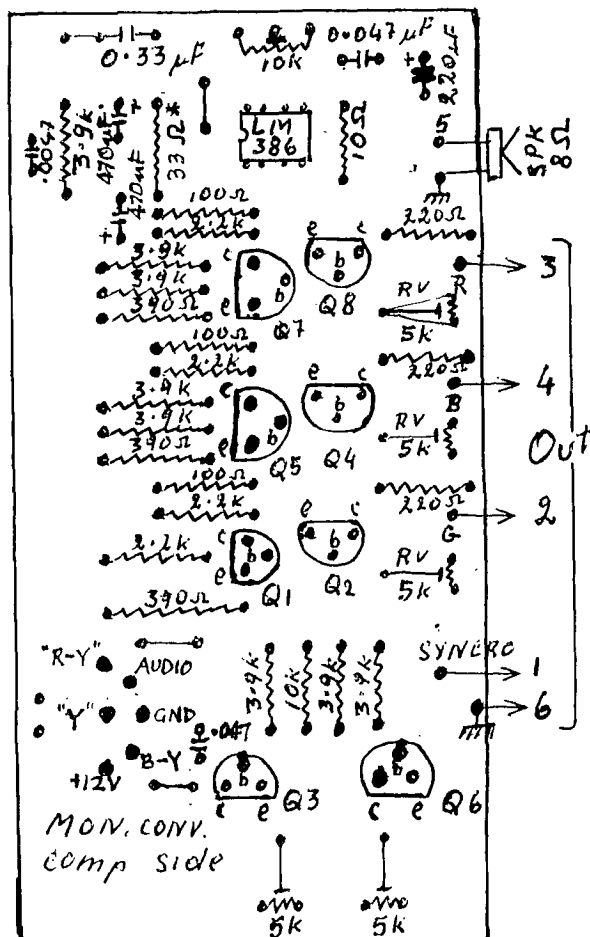
INPUTS (from Console)



OUTPUTS (to Monitor)

Q2, Q4, Q8  
BC 558

Q1, Q3, Q5  
Q6, Q7  
BC 548



continued from page 9

## Publications Library Report

with Warren Welham

Many people have been asking for books on Assembly which are not in the Library. Anybody who has some Assembly books and does not use them all the time, please could you think about donating them to the library to let new people learn Assembly language. In next month's issue, hopefully will be printed a complete library list for books and TNDs for all members to see. Do not forget interstate members or people who cannot get to the main monthly meeting, can join the library and write to me and ask for the books they want using the Codes on this list. (Postage for the books must be paid by the member.)

|New Tnds arrived this month |

|April 1988 (4) |

|Appeal: Anyone who has any TNDs(SND) they do not read anymore, please think about donating them. |

|New Books arrived this month |

Code	Title	Author
00214	HRD+ Ramdisk Construction Guide	Bud Mills
00215	HRD+ Ramdisk Construction Guide	Bud Mills
00216	Texas Instruments Engineering Journal	T.Instrument
00217	" " " "	" "
00218	" " " "	" "
00219	TI Writer Manual	T.Instrument

|New Arrivals of Overseas Publications |

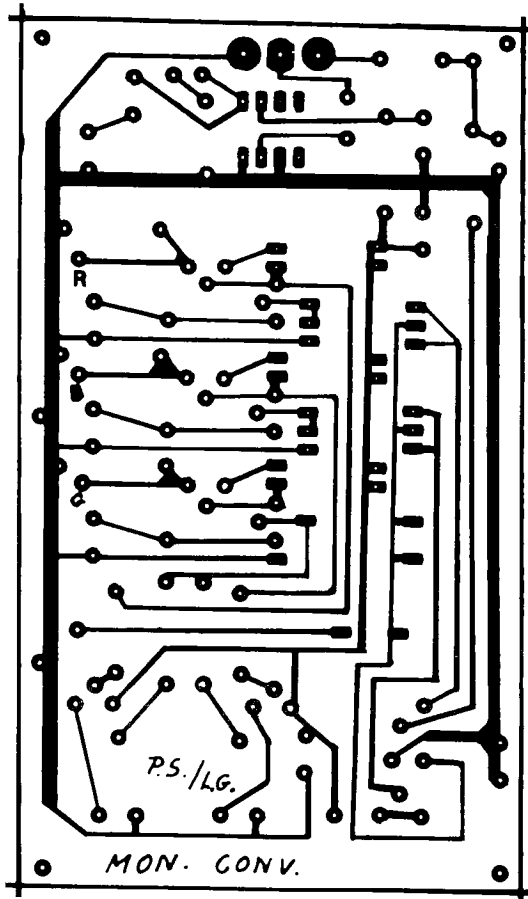
Group	Publications Name	Date
The Ottawa TI-99/4A User Group	Newsletter	Feb 88
Channel 99 Hamilton	Clubline	Nov 87
Channel 99 Hamilton	Clubline	Jan 88
Channel 99 Hamilton	Clubline	Feb 88
Hunter Valley 99ers user group	Hunter Valley News	Feb 88
Micropendium	-	Feb 88
Bluegrass 99 Computer Society.Inc	Bytemonger	Feb 88
T.I.U.P Inc. Perth	Tit Bits	Nov 87
Brisbane User Group	Bug Bytes	Feb 88
Central Ohio Ninety Niners Inc.	Spirit of 99	Mar 88
Pittsburgh Users Group	The Pug Peripheral	Dec 87
Ryte Data	R/D Computing	Mar 87
Ryte Data	R/D Computing	Jun 87
Tacoma 99ers users group	The Tacoma Informer	Feb 88
Northern New Jersey	-	Feb 88

+ 1 Ryte Data 1987 Catalogue.

For those who are wondering why Ryte Data magazine is so far behind the others, it was sent via Manila! o

### Parts List

5	BC547,548	Transistor (NPN) Q1,3,5,6,7
3	BC557,558	Transistor (PNP) Q2,4,8
1	LM386	Sound IC (optional)
1	3 inch SPEAKER	Tandy 40-348 (8 ohm)
5	5KA	Trimpot miniature
1	10KC	Volume control pot.
2	470MF 16V	Electro. cap.
1	220MF 16V	Electro. cap.
1	0.33MF 25V	Ceramic or Greencap
2	0.047MF 25V	Ceramic cap.
1	0.0047MF(4n7)	Ceramic cap.
Resistors all 1/4 watt 5%		
1	10R	
1	33R	
3	100R	
3	220R	
3	390R	
4	2K2	
11	3K9	
1	10K	



continued from page 21

The next routine is one of the standard command files that comes with Advanced Diagnostics and will format a box of disks and name each one individually.

[255][7]Beep[13][7]

Format a Box of Ten Disks as SS/SD in the [253]

[253][253][253][253][7]

Selected Drive [10]

[253][253][253][253][7]

BE PA FD 1 S DISK\_01 [13]

BE PA FD 1 S DISK\_02 [13]

BE PA FD 1 S DISK\_03 [13]

BE PA FD 1 S DISK\_04 [13]

BE PA FD 1 S DISK\_05 [13]

BE PA FD 1 S DISK\_06 [13]

BE PA FD 1 S DISK\_07 [13]

BE PA FD 1 S DISK\_08 [13]

BE PA FD 1 S DISK\_09 [13]

BE PA FD 1 S DISK\_10 [13]

[7][ ]

A variation of this routine allows me to use all four drives to format 4 separate disks as DSD while I make myself a cup of coffee. The last line of my command file restarts the whole procedure again. The format routine is the same speed as any other formatter but the verify is not the fastest I have seen or used, but it is reliable. Who cares anyway, I was out making coffee or mowing the grass or something else at the time.

To format as DSD just replace the 1 S with 2 D and your own disk names instead of the DISK\_01 etc.

The power of these command files is such that you can leave the machine unattended whilst it does mundane tasks for you.

## Games Information

by Robert Brown and Stephen Judd

I would like to take this opportunity to thank all the members who have supported us, and left us problems to solve for them.

### How to Play and Master Slymoids

Start up Slymoids, and in the lower right corner, you will see the numbers 1 to 5, with an asterisk moving next to the numbers. When the asterisk is beneath the level you want, press the fire button or any other key, except the number keys or press FCTN[7] for a demo game. To pause, hit the space bar or P and to resume, press any key.

In this game there are 3 scenarios. The first is on a grassy slope with pretty pictures of castles and caves, although they do not seem to pretty when you kill off all the apes and go to scenario 2. This is the pretty castle. When you finish this you may think you can retire. But no, next is a cave, where rocks can come to life and fall on you. After this you think you must be able to sit back and have a rest, but you find you still have 4 more levels to go, each with 3 more scenarios. Life sure is hard for this cowboy (TI says he is a cowboy, but he looks more like an elephant man).

One thing many people find confusing is the slymoid scanner. If you look in the lower left hand corner you will see a coloured bar, with an arrow showing your location on it. As more slymoids gather in the area, the worse the colour goes. Areas with 0 to 10 creatures have dark green, light green, and yellow. For areas with more than 10 creatures the colours are light red, dark red and black.

There are three types of Slymoid creatures.

- (1) Apey, crawley, black blobby things
- (2) Jelloids
- (3) Spores

These will turn bats, and even rocks into enemies. Anything that moves in this game, shoot it. You do not have anybody else on your side, so if it is near you, blow it away.

Some creatures take a number of blasts to kill them, but do not worry, you have unlimited shots. To kill things high up at the top of the screen pull back and fire. This takes a lot of trouble to decide which is more of a threat.

If spores land or escape off the screen they automatically turn into three new slymoids, and if three slymoids come together they produce another slymoid. Do not be too quick to make a run for it, because they still come back later. Watch the creatures that get at the very edge of the screen, they are fatal if you are making a quick retreat.

Points in this game are awarded for more difficult and long range shots, and if you shoot them before they are on the ground, you also get more points. Do not take points too seriously, as they do not really matter much in this game.

### Downloaded Programs on the BBS

This catalog was written by Chocky and Tictacman, also thanks to Ross Mudie.

Name	Number of	BBS
==Description	==Parts==	Month
=====	=====	=====
4a Flyer	1	No 1/88UL
Flight simulator		
Adventure Module	1	No 3/87
Use this to load adventures		
Aggressor	1	3/87
Shoot the moving aliens		
Ambulance	1	No 1/88UL
Save sick people		
Anteater	1	No 9/87
Bring cubes to nest		
AsTIroids	1	12/87UL
Shoot the AsTIroids		
Bandit	1	No 12/87UL
Multi-screened maze game		

Bandit	1	No	7/87
Multi-screened maze game			
Barrage	2	No	
Destroy the invading aliens			
Berlin	2	Yes	2/88
Escape from East to West			
Beyond Parsec	1	No	1/87
Shoot each other			
Bigfoot	2	Yes	12/87
Go get Bigfoot on the mountain			
Black Hole	1	No	9/87
2 player space game			
Boxing	1	No	12/87UL
Beat up your friends			
Break Thru	1	No	3/87
Break through a wall of bricks			
Buck Rogers	2	No	1/88A
Destroy the mother ship			
Burger Time	2	No	1/88B
Build the burgers			
Carwars	1	No	8/87
Try to avoid the other car			
Cat and Mouse	1	No	9/87
Centipede	1	No	10/87
Shoot the moving centipede			
Chisolm Trail	1 (BIG)	No	12/87
Shoot the moving aliens			
Clone-CC	1	No	12/87UL
Corcomp disk copier			
Clone-TI	1	No	12/87UL
TI disk copier			
Clowns	1	No	1/87
Try to catch balloons			
Congo Bongo	3	No	
Complete the 2 screens			
Cross Fire	1	No	9/87
Shooting grid game			
Cubit	1	No	8/87
Jump from block to block			
Cubit	1	No	12/87
Jump from block to block			
D-Station1	1	No	12/87UL
Shoot the falling aliens			
D-Station2	1	No	1/88UL
Shoot the falling aliens			
Debug	1	No	12/87
Program that did all these			
Defender	1	No	12/87
Save men from aliens			
Dig Dug	2	No	1/88A
Shoot the monsters			
Donkey Kong	2	No	1/88A
Jump barrells and rescue love			
Dragons	1	No	1/88UL
Shoot and avoid the dragons			
Driving Demon	1	No	10/87
Avoid the cars and oil spills			
Driving Demon	1	No	1/87
Avoid the cars and oil spills			
Driving Demon	1	No	7/87
Avoid the cars and oil spills			
Face Race	1	No	12/87
Collect the faces			
Fast Term	2	No	1/88B
Terminal Emulator			
Fathom II	3	Yes	
Find the hidden treasures			
Freddy	2	No	12/87
Find your way out of a maze			
Freddy	2	No	5/87
Find your way out of a maze			
Guardian	1	No	9/87
2 player space game			
Hen House	2	No	2/88
Catch the falling eggs			
Hen Pecked	1	No	9/87
Catch the bird and the eggs			
Hopper	1	No	7/87
Kill the robbers with hay			
Hopper	1	No	12/87UL
Kill the robbers with hay			
Hunt the Wumpus	1	No	1/87
Find the Wumpus			
Hunt the Wumpus	1	No	7/87
Find the Wumpus			

I'm Lost	1	No	6/87	Systex	1	No	3/87
Escape from a maze				See systexDOC file			
Jawbreaker	2	No		TI Runner	3	No	10/87
Collect dots and brush teeth				Escape from the guards			
Jungle Hunt	2	No	2/88	TI Toad	1	No	8/87
Save damsel from pygmies				Like Frogger. Cross the lake			
Kaboom	1	No	1/88UL	TI Invaders	1	No	
Catch the dynamite				Like space invaders			
Lasso	4	Yes	2/88	Tennis	1	Yes	6/87
Round up the cattle etc.				Play tennis. Three levels			
Lines	1	No	6/87	Tombstone City	1	No	12/87
Graphics demo, with lines				Shoot the Morg and Weed			
Mash	3	Yes	1/88B	Vegas	1	No	9/87
Save the wounded soldiers				Poker machine			
Meteor belt	3	Yes		Viatel	1	No	5/87
Shoot each other				This viatel is for RS232/1			
Miner 2049'er	2	No	1/88B	Viatel/2	1	No	6/87
Collect all objects in a time				This viatel is for RS232/2			
Moon Mine	4	Yes	1/88B	Voyager	1	No	1/88UL
Destroy the bad guys				Shoot the moving creatures			
Moon Patrol	2	No	9/87	Word Count	1	No	3/87
Travel from A to Z				Counts words			
Moonsweeper	3	No	2/88	Word Count	1	No	10/87
Shoot the aliens				Counts words			
Pacman	2	No	2/88	Worm Attack	1	No	5/87
Old ATARI maze game				Shoot worms			
Paddle Ball	1	No	1/87				
Hit balls with paddles							
Piano	1	No	9/87				
Electronic piano							
Picnic Paranoia	2	No	1/88A				
Save food from ants							
Pinball	1	No	12/87				
TI pinball machine							
Pole Position	2	No	1/88A				
Racing car game							
Popeye	2	No	1/88A				
Save Olive in 3 screens							
Q*Bert	2	No	1/88A				
Jump from block to block							
Rabbit Trail	1	No	12/87UL				
Hop to the top of the screen							
Rotor Raiders	1	No	1/87				
Keep the maze clean							
Sewer Mania	4	Yes	1/88B				
Two screen maze game. Get bomb							
Shamus	2	No					
Multi-screened maze game							
Shamus	1	No	8/87				
Multi-screened maze game							
Slymoids	2	No	1/88A				
Kill the apes							
Sneggit	1	No	1/87				
Collect eggs							
Sneggit	1	No	7/87				
Collect eggs							
Space Junk	1	No	1/87				
Shoot the space junk							
Space Junk	1	No	12/87UL				
Shoot the space junk							
Spy's Demise	1	No	1/88UL				
Use elevators to climb up							
Star Force	1	No	9/87				
Move around and shoot aliens							
Star Trek	2	Yes	6/87				
Shoot the aliens							
Star Trek	2	Yes	3/87				
Shoot the aliens							
Star Wars	1	No	12/87UL				
Game from movie							
Star Wars	1	No	7/87				
Game from movie							
Star Wars	1	No	1/87				
Game from movie							
Star Wars2	1	No	12/87UL				
Game from 2nd StarWars movie							
Submarine Battle	1	No	12/87				
Shoot planes and mines							
Super Demon Attack	3	Yes	7/87				
Shoot the moving aliens							
Superfly	4	Yes	2/88B				
Destroy the spiders							
Systex	1	No	9/87				
See SystexDOC file							

If users wish to have any of these programs, please leave mail to REQUESTS and we will upload them within a few days.

## Potpourri

by Tom Arnold, Channel 99ers

Retyped from Clubline, Hamilton Canada Users Group Newsletter, February, 1988 by John Ryan of TISHUG for TEXPAC BBS. Edited by Ross Mudie.

Not much in the TI news world at this time. There is a rumor that TI is about to introduce a new home computer to the market. It is supposed to be TI99/4A compatible. I would doubt that there is any truth to this.

Clint Pulley has released his latest release of c99. This is a supplemental disk which you can get from him or from our (Hamilton) library

Rave 99 has released its Memory Enhancement Card system, which allows up to two megabytes of backed-up RAM memory for the TI. Rave 99 also has released its Speech Synthesizer card for the PEB. This allows you to add the speech synthesizer to the PEB, a must for those with the Geneve.

The latest release for the Geneve is 1.0 of DOS, 99.B for the GPL loader and 1.0 for MyWord. No word yet on Advanced BASIC, C-Compiler or the Pascal System. I forgot, there is also a later release of Multiplan but I do not remember if there is a version number involved.

Innovative Programming of Rohnert Park, CA is going out of the TI business. Do not send them any orders.

Ryte Data of Haliburton has not been answering their mail. Be very cautious about sending them any money.

MICROpendium is your best bet on getting news of the TI. Its address is P.O. Box 1343, Round Rock, TX 7B680. Check for prices.

PC-Transfer is a new program that transfers text files from the TI to an IBM via disk. Yes you read right, it allows you to format an IBM disk on your TI and then you can copy your files onto it in IBM format. You must have a Myarc or CorComp disk controller (you need double density) and a TI or Geneve. This program was written by Mike Dodd but I do not know whether it is available yet.

On the subject of mail order, Computer Shopper has many good deals. It also has a TI column by Ron Albright. I ordered a chip upgrade for my Epson called Dots-perfect that has been the best money I have spent in a long time. Computer Shopper is available at most news stands (Canada).

If you order by mail, never, never send cash or cheque. Always order by Visa or Master Card. If they do not deliver, you can have the charges removed from your statement. This is very good protection, and when ordering from the U.S.A., you do not have to worry what the exchange rate is, they will bill you at the current rate.

## Hitch Hiker's Guide to the Galaxy part 2

by Stephen Judd and Robert Brown

There you are in the dark again. Wait and watch the display, until it no longer says you cannot hear. Then listen, and you will hear the sound of the star drive. Now it gets cute: the program will lie to you, and say there is an exit to port. Do not you believe it! Go Aft instead, and you will be in Entry Bay 2. You can ignore the brochure, if you like. Go Aft again, and you are in the Fore End of the corridor. Here, Ford will find you, and take you up to the bridge, where you meet Zaphod and Trillian.

Actually, you have seen them both before, at a party you attended a short time ago. While you listen to the chatter between Zaphod and Ford, you can begin to drop some items here. You can drop the plotter, screwdriver, gown, and signalling device. By this time, everyone else has gone to the sauna, leaving you alone on the Bridge with Eddie, the shipboard computer. Do not mind Eddie, he is a little over protective, but he is a good sort at heart. In fact, you can pretty much ignore him, as well as Marvin the Paranoid Robot, although Marvin will be important much later on (depressing as that may sound).

OK, time to prepare for some pretty weird happenings! First you will need the spare improbability drive. So, go down, then Aft. Keep going Aft. The program will tell you that the Engine Room is dangerous. It lies! Do not listen to it, just keep going Aft. Eventually, you will get there. Of course, as soon as you are there, you will want to look around. The program will tell you there is nothing to see. That, too, is a lie! Keep looking, and you will find that there are some things to see here, especially the spare drive. Do not worry about the tools for now; you can leave them where they are.

Once you have the drive, go back to the Fore end corridor, then head Port where you will find the Nutrimat (try consulting the Guide about the Nutrimat). Touch the pad, and you will be provided with a delicious (?) cup of advanced tea substitute. Fortunately, you do not have to drink it. Take the cup (ignore the carton, it is useless) and return to the bridge. Drop the cup and the drive. Now, plug the small plug in to the small receptacle, and put the plotter's dangly bit into the tea substitute.

OK, you are about to have some pretty strange experiences, but before you throw the switch, some words of advice and caution. There are five scenarios (all rather short, but all of them important), that have to be completed. They come up in random order, so each one has its own little section of the walk through. The lead in to each of them is that familiar dark area, where you have to wait until one of your senses is working again. You will be in the dark area again when the scenario ends (and you will have to listen for the drive sound), which will then bring you back to the HOG. Also, be aware there are times that you may briefly go back to one of the scenarios you have already completed. You just sort of bounce in and out of those, but you do have to spend time waiting in the dark. I could not find a way around this, so you will just have to live with it. Finally, it is a good idea to save the game after you complete each scenario, just in case.

With that said, it is time, so push the switch!

### The Bugblatter Beast

When you come out of the dark, you find yourself in the Lair of the dreaded Bugblatter Beast. You are here for a purpose, so you will just have to do something about the Beast. Consulting the Guide tells you that Bugblatters are incredibly stupid, which is certainly the case. In the meantime, the Beast is bearing down on you, demanding your name. Do not be shy, introduce yourself, then run like heck East out of the Lair. Pick up one of the sharp stones, and then cover your head with the towel. Old Buggy is so dumb, he thinks that, since you cannot see him, he cannot see you. But, this will not last for very long, so you have to fool him, and quickly. Lucky for you, this is not hard. Carve your name on the Bugblatter's memorial. When he sees the name there, he will think he is already eaten you, which is why he cannot see you (lumb may be an understatement here).

The Beast will then curl up for a nap, leaving you free (after removing the towel!) to re-enter the Lair and then go Southwest. Here you will find the skeleton of some poor soul clutching a Nutrimat Computer Interface Card. Take the card, and just wait around for a while. You will be mistakenly captured as a Bugblatter Beast (talk about insults!), but you will eventually be freed, and have some other adventures along the way, before you find yourself back in the dark again.

When you get back to the HOG, you can drop the asteroid paint chpper and the interface card in the Fore End before going up to the Bridge (you will need the interface later, but there is no need to drag it around with you now). Once on the Bridge, push the switch again, and you will be back in the dark.

### Trillian

The dark ends with something liquidy to the touch. In fact, you find your fingers bathing in a glass of wine. Coming to your senses, you realize that you are now Trillian, and you are at the party where you (she?) met both Arthur and a mysterious man named Phil. Take a good look at Arthur, and you will see he has a huge ball of fluff on his jacket. Just what you want, but your hands are full. Drop the plate you are holding, and get the fluff. Open your handbag and put the fluff in it, then get the plate again (otherwise, the pushy hostess will not leave you alone). Now, all you need to do is wait, trying not to be bored to tears by Arthur's feeble attempts at conversation. Give Phil a look, and shortly he will come over, and take you out to his scooter. As you blast off, everything once again becomes....dark.

### Ford

Now you find yourself standing in a country lane, holding a satchel. The place looks familiar. In fact, it is the lane outside Arthur's home, and this time you seem to be Ford Prefect. Those Vogons will be arriving soon, so there is not much time. Open the satchel, and take the satchel fluff, the towel, and the sub-etha signalling device. Go North, and there you will see Arthur lying in front of the bulldozer. With a certain feeling of deja vu, you offer him the towel. However instead of taking it he asks you about his home. You suddenly realize what is going on (not that it really matters, considering what will shortly happen!). In a moment of magnanimity (or possibly madness), you decide to take Arthur hitchhiking with you. But first, you have to deal with Prosser. Go over to him, and ask him to lie down in front of the bulldozer. He will make a little fuss, but you will manage to persuade him. Now, you and Arthur can hurry over to the pub, and drink some beer (remember to buy peanuts).

Sit there, drinking your beer (no more than three!), meditating on why Arthur is taking the imminent demise of the world so calmly, until the house falls and Arthur goes tearing out. Follow him to the ruins of his home. Drop the satchel, and put the satchel fluff on top of it. Now wait. The Vagon ships will appear, the winds will pick up, and you will start fumbling with the device. Oops! You just dropped it! Fortunately, it rolls over by Arthur, who picks it up and looks at it. Also fortunately, Arthur manages to push the right button, and everything becomes....dark.

That is it for this month. But remember if you have any problems in the above file, please direct them to GAMES, or any other problems that may arise, we will be happy to hear about them. ○

C HOCKY AND T ICTACMAN



# TISHUG NEWS DIGEST

```

100 REM *** HANGMAN ***
110 REM P.S.I. 10/81
120 REM
130 REM
140 REM ***** TI 99/4(A) **
*
150 REM ***** EXTENDED **
*
160 REM ***** BASIC **
*
170 REM
180 REM
190 OPTION BASE 0
200 DIM WORD$(1000)
210 F3=0 :: F4=0
220 FOR I=1 TO 8 :: CALL COL
OR(I,2,15):: NEXT I
230 CALL CLEAR :: CALL SCREE
N(13):: DISPLAY AT(2,11)SIZE
(7):"HANGMAN" :: DISPLAY AT(
3,11)SIZE(7):"-----"
240 DISPLAY AT(12,3):"PRESS:
"
250 DISPLAY AT(11,12):"L TO
LOAD FILE"
260 DISPLAY AT(13,12):"R TO
BUILD FILE" :: DISPLAY AT(4
,08)SIZE(13)BEEP:"WORD DATAB
ASE"
270 DISPLAY AT(19,5)SIZE(22)
:"PRESS BACK TO STOP"
280 DISPLAY AT(23,7)SIZE(15)
:"7229 BYTES USED"
290 CALL KEY(0, KK, SS)
300 IF SS<=0 THEN 290
310 IF KK=76 THEN 1750
320 IF KK=82 THEN 1240
330 IF KK=15 THEN 1190
340 CALL SOUND(200,225,0)::
GOTO 290
350 RANDOMIZE
360 IF LEN(WORD$(0))<1 THEN
100
370 RESTORE
380 CALL CLEAR :: GOSUB 830
390 CALL SCREEN(14)
400 FOR I=1 TO 8 :: CALL COL
OR(I,2,15):: NEXT I
410 DISPLAY AT(2,11)SIZE(7)B
EEP:"HANGMAN"
420 DISPLAY AT(3,11)SIZE(7)B
EEP:"-----"
430 CALL COLOR(9,13,13):: CA
LL COLOR(12,2,15)
440 FOR G=5 TO 6 :: CALL HCH
AR(G,6,96,8):: NEXT G
450 FOR G=7 TO 8 :: CALL VCH
AR(7,G,96,2):: NEXT G
460 FOR G=12 TO 13 :: CALL V
CHAR(7,G,96,12):: NEXT G
470 FOR G=19 TO 24 :: CALL H
CHAR(G,10,96,6):: NEXT G
480 CALL HCHAR(13,19,96,10)
490 CALL HCHAR(22,19,96,11)
500 CALL VCHAR(13,19,96,9)
510 CALL VCHAR(13,29,96,9)
520 CALL CHAR(136,"010101010
1010101"): CALL COLOR(14,4,
2)
530 CALL HCHAR(9,7,121)
540 DISPLAY AT(24,21)SIZE(4)
:"USED" :: DISPLAY AT(2,25)S
IZE(3):"WON"
550 DISPLAY AT(10,15)SIZE(5)
:"TRY:" :: DISPLAY AT(8,25)S
IZE(4):"LOST"
560 DISPLAY AT(4,26)SIZE(3):
F3 :: DISPLAY AT(8,26)SIZE(3
):F4
570 REM PICK RANDOM WORD
580 R=INT(RND*(VAL(WORD$(0))
)): IF R=0 THEN R=1
590 REM DISPLAY BLANKS FOR W
ORD

```

```

600 FOR G=15 TO LEN(WORD$(R)
)+14
610 DISPLAY AT(8,G)SIZE(1)BE
EP:"- "
620 NEXT G
630 REM MAKE A GUESS
640 ROW=15 :: COL=18 :: GUES
SS="" :: Q2=0
650 CALL COLOR(0,2,2)
660 DISPLAY AT(10,19):CHR$(3
0):: CALL KEY(0,K9,S9)
670 IF S9=0 THEN 660 ELSE TR
Y$=CHR$(K9)
680 DISPLAY AT(10,19)SIZE(1)
:TRY$
690 Q2=0
700 FOR I=1 TO LEN(WORD$(R))
710 IF TRY$=SEG$(WORD$(R),I,
1)THEN DISPLAY AT(7,I+14)SIZ
E(1)BEEP:TRY$
720 IF TRY$=SEG$(WORD$(R),I,
1)THEN Q(I)=1 ELSE Q(I)=0
730 Q2=Q(I)+Q2 :: NEXT I
740 IF Q2=0 THEN 760
750 FOR I=1 TO LEN(WORD$(R))
:: Q7=Q7+Q(I):: NEXT I
760 DISPLAY AT(ROW,COL)SIZE(
1):TRY$
770 IF Q7=LEN(WORD$(R))THEN
1080
780 COL=COL+1 :: IF COL=27 T
HEN ROW=ROW+1
790 IF COL=27 THEN COL=18
800 IF ROW=20 THEN ROW=15
810 IF Q2=>1 THEN 660 ELSE Q
2=0
820 GO TO 930
830 CALL CHAR(120,"FF81A581A
5BD81FF")
840 CALL CHAR(121,"181818181
8181818")
850 CALL CHAR(122,"804020100
8040201")
860 CALL CHAR(123,"010204081
0204080")
870 CALL CHAR(124,"181818181
8181818")
880 CALL CHAR(125,"010204081
0204080")
890 CALL CHAR(126,"804020100
8040201")
900 Q6=120
910 CALL SCREEN(4)
920 RETURN
930 IF Q6=120 THEN CALL HCHA
R(10,7,120)
940 IF Q6=121 THEN CALL HCHA
R(11,7,121)
950 IF Q6=122 THEN CALL HCHA
R(11,6,122)
960 IF Q6=123 THEN CALL HCHA
R(11,8,123)
970 IF Q6=124 THEN CALL HCHA
R(12,7,121)
980 IF Q6=125 THEN CALL HCHA
R(13,6,125)
990 IF Q6=126 THEN CALL HCHA
R(13,8,126)
1000 Q6=Q6+1
1010 IF Q6=127 THEN 1030
1020 GOTO 660
1030 DISPLAY AT(1,4)SIZE(32)
:"++++++YOU LOSE++++++"
1040 F4=F4+1
1050 CALL SOUND(1000,300,0):
FOR I=1 TO 100 :: NEXT I
1060 DISPLAY AT(7,15)SIZE(LE
N(WORD$(R)))BEEP:WORD$(R)
1070 GOTO 1110
1080 DISPLAY AT(1,4)SIZE(32)
:"++++++YOU WIN++++++"
1090 F3=F3+1
1100 CALL SOUND(2000,800,0)
1110 FOR I=1 TO 500 :: NEXT
I

```

```

1120 CALL CLEAR
1130 DISPLAY AT(11,5)SIZE(17
)BEEP:"PLAY AGAIN? (Y-N)"
1140 CALL KEY(0,K9,S9)
1150 IF S9=0 THEN 1140
1160 Z$=CHR$(K9)
1170 Q7=0
1180 IF Z$="Y" THEN 350
1190 CALL CLEAR :: FOR QD=1
TO 200 :: NEXT QD
1200 DISPLAY AT(10,4):"END S
SESSION? (Y-N)"
1210 CALL KEY(0, KKK, SSS)
1220 IF SSS=0 THEN 1210 ELSE
IF KKK=78 THEN 350
1230 CALL CLEAR :: END
1240 REM BUILD DISKETTE FILE
FOR HANGMAN
1250 REM 10 LETTERS IN 10 WO
RD BLOCKS
1260 CALL CLEAR :: CALL SCRE
EN(9)
1270 DISPLAY AT(2,10)SIZE(7)
BEEP:"HANGMAN"
1280 DISPLAY AT(3,10)SIZE(7)
:"-----"
1290 DISPLAY AT(7,5)SIZE(19)
:"BUILD DISKETTE FILE"
1300 DISPLAY AT(22,6)SIZE(18
):"ENTER ZERO TO STOP"
1310 DISPLAY AT(20,6)SIZE(18
):"10 LETTERS MAXIMUM"
1320 R=1
1330 DISPLAY AT(10,10)SIZE(1
6):"WORD #":R
1340 DISPLAY AT(16,9):"-----
"
1350 ACCEPT AT(15,9)SIZE(10)
BEEP:WORD$(R)
1360 IF WORD$(R)="0" OR R=10
00 THEN 1380
1370 R=R+1 :: GOTO 1330
1380 DISPLAY AT(14,8)SIZE(14
)BEEP:"END FILE ENTRY"
1390 FOR DD=2 TO 1000 :: NEX
T DD
1400 R=R-1
1410 WORD$(0)=STR$(R)
1420 CALL CLEAR
1430 DISPLAY AT(5,4)SIZE(8):
"PRESS:"
1440 DISPLAY AT(8,4):"1 TO R
ECORD DATA FILE"
1450 DISPLAY AT(11,4):"2
PLAY THE GAME"
1460 DISPLAY AT(14,4):"3
CHANGE A WORD"
1470 CALL KEY(0, KK, SS)
1480 IF SS<=0 THEN 1470
1490 IF KK=49 THEN 1530
1500 IF KK=50 THEN 350
1510 IF KK=51 THEN 1620
1520 CALL SOUND(300,225,0)
1530 REM RECORD FILE
1540 CALL CLEAR
1550 OPEN #1:"DSK1.1266/D",S
EQUENTIAL,DISPLAY ,OUTPUT,FI
XED 120
1560 PRINT #1:WORD$(0)
1570 FOR R=1 TO VAL(WORD$(0)
)
1580 PRINT #1:WORD$(R)
1590 NEXT R
1600 CLOSE #1
1610 GO TO 1420
1620 REM CHANGE A WORD
1630 CALL CLEAR :: CALL SCRE
EN(9)
1640 DISPLAY AT(2,10)SIZE(7)
:"HANGMAN"
1650 DISPLAY AT(3,10)SIZE(7)
:"-----"
1660 DISPLAY AT(6,8)SIZE(13)
:"CHANGE A WORD"

```

continued on page 17

# TIsHUG (Australia) Ltd. Pla

	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	
<b>May 1988</b>	1	2	3	4	5	6	7 TIsHUG meeting Sydney	8	9	10	11	12	13	14	15	16	
				1	2	3	4 TIsHUG meeting Sydney	5	6	7	8	9	10	11	12	13	
<b>June 1988</b>							1	2 TIsHUG meeting Sydney	3	4	5	6	7	8	9	10	
							1	2 TIsHUG meeting Sydney	3	4	5	6	7	8	9	10	11
<b>July 1988</b>							1	2 TIsHUG meeting Sydney	3	4	5	6	7	8	9	10	11
			Mon	Tue	Wed	Thu											
<b>August 1988</b>	1 Bank holiday	2	3	4	5	6 TIsHUG meeting Sydney	7	8	9	10	11	12	13	14	15		
					1	2	3 TIsHUG meeting Sydney	4	5	6	7	8	9	10	11	12	
<b>September 1988</b>					1	2	3 TIsHUG meeting Sydney	4	5	6	7	8	9	10	11	12	
						1	2	3 Labour Day	4	5	6	7	8 TIsHUG meeting Sydney	9	10		
<b>October 1988</b>						1	2	3 Labour Day	4	5	6	7	8 TIsHUG meeting Sydney	9	10		
			Tue	Wed	Thu	Fri											
<b>November 1988</b>			1	2	3	4	5 TIsHUG meeting Sydney	6	7	8	9	10	11	12	13	14	
					1	2	3 TIsHUG meeting Sydney	4	5	6	7	8	9	10	11	12	
<b>December 1988</b>					1	2	3 TIsHUG meeting Sydney	4	5	6	7	8	9	10	11	12	
	Sun	Mon	Tue	Wed													
<b>January 1989</b>	1 New Year's Day	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	
					1	2	3	4 TIsHUG meeting Sydney	5	6	7	8	9	10	11	12	13
<b>February 1989</b>					1	2	3	4 TIsHUG meeting Sydney	5	6	7	8	9	10	11	12	13
					1	2	3	4 TIsHUG meeting Sydney	5	6	7	8	9	10	11	12	13
<b>March 1989</b>					1	2	3	4 TIsHUG meeting Sydney	5	6	7	8	9	10	11	12	13
						1	2	3	4	5	6	7	8	9	10		
<b>April 1989</b>						1	2	3	4	5	6	7	8	9	10		
							1 TIsHUG meeting Sydney ?????	2	3	4	5	6	7	8	9	10	



# ing Calendar 1988 – 1989

Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue								
18	19	20	21	22	23	24	25	26	27	28	29	30	31								
													Wed	Thu							
15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30						
															Fri	Sat	Sun				
13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
17	18	19	20	21	22	23	24	25	26	27	28	29	30	31							
															Thu	Fri					
14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30					
																	Sat	Sun	Mon		
12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31		
16	17	18	19	20	21	22	23	24	25	26	27	28	29	30							
															Thu	Fri	Sat				
14	15	16	17	18	19	20	21	22	23	24	25 Christ- mas Day	26 Boxing Day	27	28	29	30	31 New Year's Eve				
18	19	20	21	22	23	24	25	26 Australia Day	27	28	29	30	31								
15	16	17	18	19	20	21	22	23	24	25	26	27	28								
															Wed	Thu	Fri				
15	16	17	18	19	20	21	22	23	24 Good Friday	25	26 Easter Day	27 Easter Monday	28	29	30	31					
																	Sat	Sun			
12	13	14	15	16	17	18	19	20	21	22	23	24	25 Anzac Day	26	27	28	29	30			

```

100 !AUTOMATIC LOAD AND RUN
PROGRAMME
110 ! BY CHRIS RYAN.
120 !ENABLES THE USER TO SEL
ECT AND RUN ANY ONE OF THE F
IRST 20
130 !PROGRAMMES STORED ON TH
E DISKETTE
140 !IN DRIVE#1 AT THE TOUCH
OF A BUTTON.
150 !REQUIRES TI EXTENDED BA
SIC,DISK DRIVE AND CONTROLLE
R
160 !AND MEMORY EXPANSION UN
IT.
170 !IF PROGRAMME IS SAVED A
S "LOAD",
180 !IT IS RUN AUTOMATICALLY
190 !UPON ENTERING TI EXTEND
ED BASIC.
200 !** INSTRUCTIONS **
210 !THIS PROGRAMME AND THE
PROGRAMMES IT IS TO RUN
220 !MUST HAVE BEEN PREVIOUS
LY SAVED ON DISKETTE. THIS
230 !DISKETTE MUST BE LOCATE
D IN DRIVE #1
240 !FOR THE PROGRAMME TO EX
ECUTE.
250 !TO RUN A PROGRAMME SIMP
LY PRESS THE LETTER
260 !NEXT TO THE PROGRAMME N
AME.
270 !TO DELETE A PROGRAMME P
RESS 1
280 !FOLLOWED BY THE LETTER
NEXT TO THE NAME OF THE
290 !PROGRAMME TO BE DELETED
.THE PROGRAMME THEN RE-BOOTS
THE DISK AND DISPLAYS THE UP
DATED PROGRAMME MENU.
300 !IF DELETE IS CHOSEN MIS
TAKENLY,TO EXIT DELETE MODE
310 !PRESS ANY KEY THAT
320 !DOESN'T CORRESPOND TO A
PROGRAMME NAME
330 !TO EXIT THIS PROGRAMME
PUSH 2
340 ! N.B. WHILE THIS PROGRA
MME WILL RUN ANY EXTENDED BA
SIC PROGRAMME,
350 !SOME PROBLEMS MAYBE EXP
IENCED WHEN TRYING TO RUN BA
SIC PROGRAMMES THAT DO NOT C
ONFORM TO EXTENDED BASIC PAR
AMETERS.
360 !(I AM INDEBTED TO 99'ER
FOR
370 !THE IDEA AND GENERAL FO
RMAT OF
380 !THIS TOTALLY ORIGINAL A
ND MOST
390 !OF ALL PRIZEWORTHY PROG
RAMME)
400 CALL CLEAR :: CALL SCREE
N(16)!CLEAR SCREEN,MAKE BACK
GROUND WHITE
410 OPTION BASE 1 :: DIM PGS
(20)!SET DEFAULT FOR ARRAYS,
DEFINE ARRAY FOR STORING PRO
GRAMME NAMES
420 OPEN #1:"DSK1.",INPUT ,R
ELATIVE,INTERNAL !OPEN INDEX
FILE ON DISK
430 INPUT #1:NS,A,A,C !READ
NAME OF DISK FROM FILE
440 DISPLAY AT(1,1):"DISK ME
NU: DISK=";NS :: R,D=0 !DISP
LAY DISK NAME,SET RECORD COU
NTER AND PROGRAMME NUMBER TO
0
450 DISPLAY AT(2,19):USING "
###":A-C :: DISPLAY AT(2,23)
:"USED" :: DISPLAY AT(3,19):
USING "###":C

```

```

460 DISPLAY AT(3,23):"FREE"
!DISPLAY AMOUNT OF SPACE ON
DISK
470 FOR X=1 TO 20 !BEGIN LOO
P FOR 20 PROGRAMES
480 R=R+1 !INCREMENT RECORD
COUNTER
490 IF R>127 THEN 560 !CHECK
HAVEN'T READ PAST LAST RECO
RD ON FILE
500 INPUT #1:PS,A,B,B !READ
PROGRAM(OR FILE)NAME FROM DI
SK
510 IF LEN(PS)=0 THEN 560 !C
HECK NAME EXISTS I.E. HAVEN'
T READ PAST END OF FILE
520 IF ABS(A)<>5 THEN 480 !C
HECK THAT NAME READ IS IN ME
MORY IMAGE FORMAT I.E. IT IS
ACTUALLY A PROGRAMME
530 DISPLAY AT(X+2,1)SIZE(14
):CHR$(91);CHR$(64+X);CHR$(9
3);" ";PS !DISPLAY LEFT BRAC
E+CORRESPONDING LETTER+RIGHT
BRACE+PROGRAMME NAME
540 D=D+1 :: PGS(D)=PS !INCR
EMENT PROGRAMME NUMBER,ASSIG
N NAME OF PROGRAMME TO CORRE
SPONDING ELEMENT OF ARRAY
550 NEXT X !LOOP BACK FOR NE
W PROGRAMME NAME
560 CLOSE #1 !CLOSE THE INDE
X FILE
570 BS="PRESS LETTER TO RUN
OR, 1=DELETE 2=EXIT....." !D
EFINE BS
580 DISPLAY AT(24,1):SEGS(BS
,1,28):: BS=SEGS(BS,2,43)&SE
GS(BS,1,1):: FOR J=1 TO 10 :
: NEXT J !ROUTINE TO MAKE BS
SCROLL ACROSS BOTTOM OF
590 REM SCREEN FROM RIGHT TO
LEFT
600 CALL KEY(0,KEY,STATUS)::
IF STATUS=0 OR KEY<49 OR(KE
Y>51 AND KEY<65)OR KEY>84 TH
EN 580 !DETECTS KEY PRESSED
AND CHECKS THAT IT IS
610 REM EITHER 1,2 OR "A" TH
RU "T"
620 IF KEY>50 THEN 720 !IF K
EY IS LETTER BRANCH TO RUN R
OUTINE
630 IF KEY>49 THEN 710 !KEY=
2 BRANCH TO END OF PROGRAMME
640 DISPLAY AT(24,1):"WHICH
PROGRAM ?" :: FOR Q=1 TO 400
:: NEXT Q !ASK WHICH PROGRA
MME TO BE DELETED,KILL SOME
TIME
650 CALL KEY(0,KEY,STATUS)::
IF STATUS=0 THEN 650 !KEY P
RESS ROUTINE
660 IF KEY<65 OR KEY>84 THEN
570 !CHECK THAT KEY IS A LE
TTER IF NOT BRANCH TO SCROLL
ROUTINE
670 IF LEN(PGS(KEY-64))=0 TH
EN 570 !CHECKS THAT LETTER P
RESSED CORRESPONDS TO A PROG
RAMME NAME IF NOT BRANCH TO
SCROLL ROUTINE
680 DISPLAY AT(24,1):"DELETE
";PGS(KEY-64)!DISPLAY NAME
OF PROGRAMME TO BE DELETED
690 DELETE "DSK1."&PGS(KEY-6
4):: RUN "DSK1.LOAD"!DELETE
THE PROGRAMME,RUN LOAD PROG
RAMME FROM DISK TO DISPLAY N
EW MENU(ASSUME THIS PROGRAMM
E
700 REM ALREADY SAVED ON DSK
1.
710 END !EXIT PROGRAMME

```

```

720 IF LEN(PGS(KEY-64))=0 TH
EN 570 !CHECKS LETTER PRESSE
D CORRESPONDS TO A PROGRAMME
NAME
730 DISPLAY AT(24,1):"RUN ";
PGS(KEY-64)!DISPLAY NAME OF
PROGRAMME TO BE RUN
740 ! RUN ROUTINE FROM 99'ER
MAGAZINE BY A.KLUDGE. READS
FROM ADDRESS -31952 THE ADRE
SS OF THE HIGHEST LINE NUMBE
R IN THE LINE NUMBER TABLE
750 !THEN USING THIS ADDRESS
READS FROM THE LINE NUMBER T
ABLE THE ACTUAL ADDRESS OF TH
E HIGHEST NUMBER PROGRAMME L
INE(THE RUN STATEMENT)
760 REM THEN POKES THE NAME
OF PROGRAMME TO BE RUN ON DS
K1 INTO THE "HOLE" CREATED B
Y THE STRING "0123456789"IN
THE RUN STATEMENT
770 REM THEN ADJUSTS THE LEN
GTH BIGHT OF THE LINE. EXCUT
ES THE RUN STATEMENT
780 CALL INIT :: CALL PEEK(-
31952,A,B):: CALL PEEK(A*256
+B-65534,A,B):: C=A*256+B-65
534 :: AS="DSK1."&PGS(KEY-64
):: CALL LOAD(C,LEN(AS))
790 FOR I=1 TO LEN(AS):: CAL
L LOAD(C+I,ASC(SEGS(AS,I,1)
)):: NEXT I :: CALL LOAD(C+I,
0)
800 RUN "DSK1.1234567890"

```



*"Tell me Parkhouse, just who is this Pacman who is supposed to have eaten your homework?"*

continued from page 14

```

1670 DISPLAY AT(10,7)SIZE(12
):"ENTER WORD #"
1680 ACCEPT AT(10,20)SIZE(3)
BEEP:R
1690 DISPLAY AT(12,10):WORD$(
R)
1700 ACCEPT AT(13,10)SIZE(10
)BEEP:WORD$(R)
1710 DISPLAY AT(15,1)SIZE(26
):"CHANGE ANOTHER WORD (Y-N
)?"
1720 ACCEPT AT(15,28)SIZE(1)
BEEP:Q2$
1730 IF Q2$="N" THEN 1420 EL
SE 1620
1740 REM LOAD DISK FILE INTO
MEMORY
1750 CALL CLEAR
1760 CALL SCREEN(9)
1770 OPEN #1:"DSK1.1266/D",S
EQUENTIAL,DISPLAY ,INPUT ,R
IXED 120
1780 INPUT #1:WORD$(0)
1790 C8=VAL(WORD$(0))
1800 FOR R=1 TO C8
1810 INPUT #1:WORD$(R)
1820 NEXT R
1830 CLOSE #1
1840 GOTO 350
1850 END

```



# TISHUG NEWS DIGEST

## Putting it All Together part 5

by Jim Peterson, Tigercub Software

Always clear the screen of garbage first.  
 This colors the screen color 16 (white).  
 All 14 of the XBasic character sets are set to 3 (dark blue) on 1 (transparent), therefore dark blue on white.  
 If you intend to ever let anyone else have your program, please put in some instructions! There are too many good public domain programs that are worthless for lack of instructions!  
 Dummy CALL KEY mode 3 makes all input come out in upper case. Arrays of more than 10 must be DIM'ed.  
 Number of inputs indefinite so use GOTO back to line 180 instead of FOR NEXT loop. X is incremented by 1 each time and is counter for inputs.  
 Input of END leaves loop. We do not want to count END as a category, so subtract 1 from 1

A\$ is the first letter of the category name.

If it is not in the string of code letters already selected, add it to that string; and if less than 17 categories have been selected, go back for another. But if code letter has been already used, ask for another code letter.

Limit it to one character.

If the next choice has already been selected, blink the prompt on and off to ask for another, otherwise add it to the code string, and add it in front of the category name (it will be taken off in the next step).

If 17 selected, line 280 is jumped to from 180 or 230, or 270  
 Start with row 2 unless X counted more than 11 categories, in which case X>11 has a relational value of -1; 2+(-1) = row 1. Display 1st character of each category within (), followed by 2nd through last characters.

Increment row by 2, minus 1 if more than 11  
 And continue printing all categories.  
 Display at 2 rows below last category.

Display at 4 rows below last category.

Display at bottom of screen

Accept at row of "category", column after category and length of code string, one character validated as being in the code string

is its position in the code string; this will be the subscript number of the array to keep the totals in.

Accept the amount, validated numeric.

Add the amount to the array subscript category.

And multiply Y by 2 or 1 to display the subtotal on the same row as the category name.

And go back for the next one.

```

100 CALL CLEAR
110 CALL SCREEN(16)
120 FOR SET=1 TO 14
130 CALL COLOR(SET,5,1)
140 NEXT SET
150 DISPLAY AT(3,4)ERASE ALL
:"TIGERCUB ADDER-UPPER": : "T
o add up several categories"
:"at one time.": "Input cat
egories - END when": "finishe
d"
160 CALL KEY(3,K,5)
170 DIM C$(22),T(24)
180 X=X+1
190 DISPLAY AT(12,1):"Catego
ry #";STR$(X)
200 ACCEPT AT(12,13):C$(X)
210 IF C$(X)="END" THEN X=X-
1 :: GOTO 280
220 A$=SEG$(C$(X),1,1)
230 IF POS(F$,A$,1)=0 THEN F
$=F$&A$ :: IF X<17 THEN 180
ELSE 280
240 DISPLAY AT(15,1):"Code 1
etter ";A$;" already": "used.
": "Pick another code letter"
250 ACCEPT AT(17,26)SIZE(1):
A$
260 CALL HCHAR(15,1,32,96)
270 IF POS(F$,A$,1)<>0 THEN
240 ELSE F$=F$&A$ :: C$(X)=A
$&C$(X):: IF X<17 THEN 180
280 CALL CLEAR
290 R=2+(X>8)
300 FOR J=1 TO X
310 DISPLAY AT(R,1): "(";SEG$(
C$(J),1,1);";";SEG$(C$(J),2
,255)
320 R=R+2+(X>8)
330 NEXT J
340 DISPLAY AT(R+2,1): "Categ
ory ";F$
350 DISPLAY AT(R+4,1): "Amoun
t"
360 DISPLAY AT(24,1): "Use mi
nus value to subtract"
370 ACCEPT AT(R+2,11+LEN(F$)
)SIZE(1)VALIDATE(F$):Z$
380 Y=POS(F$,Z$,1)
390 ACCEPT AT(R+4,8)VALIDATE
(NUMERIC):A
400 T(Y)=T(Y)+A
410 DISPLAY AT(Y*(2+(X>8)),1
0):T(Y)
420 GOTO 370

```

Here are some interesting tips from the PRACTICE  
TIBBS (713) 537-0741

Want to see how accurate your TI is?  
Try this routine.

```

100 N=1.0000001
110 FOR C=1 TO 27
120 M=N^2
130 N=M
140 NEXT C
150 PRINT "ANSWER= ";M
160 PRINT 100*((674530.470741-N)/N); "% ERROR"
170 END

```

Tired of that cyan colour screen while programming? Try this Extended BASIC immediate statement.

```

FOR I=0 TO 9 :: CALL COLOR(I,16,1) GOTO NEXT I :: CALL
SCREEN(14) :: ACCEPT AT(1,1):A

```

Press ENTER. Press FCTN 4 (CLEAR). VOILA

The foreground colour can be changed by changing the CALL COLOR number and the number in CALL SCREEN.

Tired of using the same old CALL CLEAR command to clear the screen? Try this command instead.

```

10 CALL HCHAR(1,1,32,768)

```

It will clear the screen by sweeping from top to bottom.

Now try this.

```

10 CALL VCHAR(1,1,32,768)

```

This one clears the screen by sweeping from left to right.

Want a sharper display with your black and white TV? Add this line at the start of your program:  
CALL SCREEN(15)

This will disable the colour generating circuit in the computer and remove the vertical lines often seen on black and white TVs. It also increases the sharpness of the characters.

Tired of accidentally hitting the FCTN QUIT when aiming for the SHIFT + keys? If you have 32K and Extended BASIC, disable the QUIT key by entering this routine upon entering Extended BASIC:  
CALL INIT :: CALL LOAD(-31806,16)

If you accidentally enter OLD CS1 when you mean to enter SAVE CS1, do not panic. All is not lost, type SHIFT E and hit ENTER. You will get an I/O error but do not worry. The program will still be in memory and you will get a second chance at saving it.

## Jenny's Younger Set

Hi to all the younger set and those who like adventures. Well Vincent has sent in more for us to amuse ourselves with. I am sorry it takes so long for your contributions to appear, Vincent, they do not get lost usually. You have repeated yourself a bit but I have put everything you sent into the magazine. Keep up the good work. What about some of you others trying to expand on Vincent's good work. I am sure that additions and improvements could be made to the programs Vincent sends us. I will see if I can get a prize to give to the best article which explains how a program can be improved and gives an example. It is a good way to learn to program.

Dear Jenny,

Here are a program and some drawings for you.

```

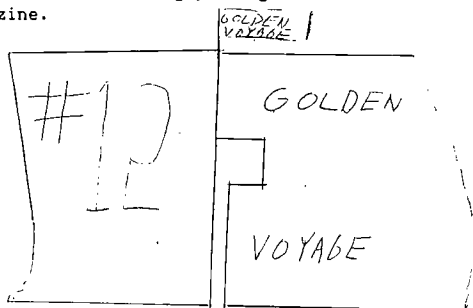
5 REM SPECIFICALLY DESIGNED FOR TEST MATCHES
10 CALL CLEAR
50 DISPLAY AT(5,3):"CRICKET ANALYSER"
100 A=0
110 B=0
120 REM DONE BY VINCENT MAKER
130 INPUT "TEAM A (WHO BATS FIRST) ":A$
140 INPUT "TEAM B ":B$
150 INPUT "TEAM A RUNS - ":A
160 INPUT "TEAM A WICKETS IN HAND ":C
170 INPUT "TEAM B RUNS - ":B
180 INPUT "OVERS BOWLED ":OVER
190 CALL CLEAR
200 IF A=D THEN RESULTS="TIE"
210 INPUT "WAS THE MATCH FINISHED?(Y/N) ":Z$
220 IF Z$="N" THEN RESULTS="DRAW"
230 IF B>A THEN RESULTS=B$&" WINS BY "&STR$(C)&
" WICKETS"
240 IF A>B THEN RESULTS=A$&" WINS BY "&STR$(A-B)&
" RUNS"
250 PRINT RESULTS: : "HAVE FUN PLAYING CRICKET!!!"
260 END
    
```



A sign is needed in the shooting gallery (if you do not want to get shot, that is).  
 Cannot get wrench? Stop the thing, then climb to greater height and be observant.  
 Guard shoots you? They do not like too much noise.  
 That is all,  
 Vincent Maker

Dear Jenny,

Here is a drawing you might want to use in your magazine.



HINTS TO SOME  
 PROBLEMS YOU MIGHT  
 FIND...

VINCENT MAKER

Merchants a problem? They want to sell you something, so you have to pay for it.

Statue a problem? You can use the stairs for this (Clue: push him..).

Tablets need to go in the fountain.

The Chalice is in the Altar.

There is something in the jungle if you have your shovel.

Dear Jenny,

I found this program in a year 10 maths book that I used last year in year 10. I thought that it might come in handy for the column. This is the best I can remember it.

```

100 REM GUESSING GAME
110 N=3
120 INPUT "WHAT IS THE MYSTERY NUMBER? ":A
130 IF A=N THEN 160
140 PRINT "WRONG"
150 GOTO 220
160 PRINT "RIGHT! ANOTHER GO? "
170 PRINT "ANSWER YES OR NO"
180 INPUT B$
190 IF B$="YES" THEN 100
200 IF B$="NO" THEN 220
210 GOTO 170
220 END
    
```

For anyone planning to change this program for a random number, line 110 needs to be changed to (say you wanted it to be from 1 to 10),

```
110 N=INT(RND)+1
```

Then, if you wanted a different random number each go, you would need to include,

```
101 RANDOMIZE
```

or for just the first go the line would have to be,

```
99 RANDOMIZE
```

Anyway, you can make the changes you prefer.

Are you running any more software competitions? Is my Yahtzee score of 300 the best score in the Younger Set Hall of Fame?

Also I would like to set up a Scott Adams Adventure Advisory Service. Do not ask me about Adventures #1 or #13, but for all the rest I will do the best I can to help you. Please write to me at:

Crocodile Jones  
 29 Palmeston Avenue  
 Winston Hills, NSW, 2153

Yours faithfully,  
 Vincent Maker

Thank you Vincent, for all of that. I am not planning any more software competitions unless I am sure that there are more than 2 who would go in them. I could not find Yahtzee mentioned in the Hall of Fame, so I guess you will have the highest score. Thank you for offering an Adventure Advisory service. Hopefully you will send the questions and answers to me for inclusion in this column, as I am sure that there are a lot of adventurers like me who have great trouble getting to the end of the games.

On another subject, I received the following set of LOGO procedures from Peter Mudie. They should provide a good introduction for any of the younger set who want an interesting way to start trying out LOGO. I can recommend LOGO as a fun language.

I have sent these procedures, which when they are all put together, will make a series of patterns. To start it all, you enter START and it will start. When saving to disc or tape you use option 3 which is BOTH 1 AND 2. There are twenty five patterns in different foreground and background colours. I hope you will include these in the next TND. More logo later, Peter Mudie, age 11.

## PROCEDURES

```

TO START
TI 95 SC 5 TT SC 15 CB 5 MAIN W 30 TI 95 SC 1 TT CB 7
CS
PRINT "GOODBYE!"
END
TO TI :NO
TELL TILE :NO
END
TO TI
TELL TURTLE
END
TO MAIN
CS TT HT H
REPEAT 36 [RT 10 BOX ]
W 30 CS H CB 10 TI 95 SC 10 TT HT
REPEAT 36 [RT 10 TRI ]
W 30 CS H HT CB 6 TI 95 SC 6 TT HT
REPEAT 36 [RT 10 CIR ]
W 30 CS H CB 2 TI 95 SC 2 TT HT SC 15
REPEAT 36 [RT 10 TRII ]
W 30 CS H CB 8 TI 95 SC 8 TT HT SC 15
REPEAT 36 [RT 10 OCT ]
W 30 CS H CB 1 TI 95 SC 1 TT HT SC 15
REPEAT 36 [RT 10 DES ]
W 30 CS H CB 13 TI 95 SC 13 TT HT SC 15 LT 3
REPEAT 2 [MAKE "C 0 RT 180 REPEAT 9 [MAKE "C :C + 1
CIRC :C ] ]
W 30 CS H CB 14 TI 95 SC 14 TT HT SC 4
REPEAT 4 [MAKE "C 0 RT 90 REPEAT 8 [MAKE "C :C + 1 CIRC
:C ] ]
W 30 CS H CB 14 TI 95 SC 14 TT HT SC 1
REPEAT 4 [SQ 1 SQ 2 SQ 3 SQ 4 SQ 5 SQ 6 SQ 7 LT 90 ]
W 30 CS H CB 14 TI 95 SC 14 TT HT SC 8 PU FD 10 RT 45
PD
REPEAT 4 [SQ 1 SQ 2 SQ 3 SQ 4 SQ 5 SQ 6 SQ 7 RT 90 ]
W 30 CS H CB 2 TI 95 SC 2 TT HT SC 3 PU FD 10 RT 45 PD
REPEAT 4 [MAKE "C 0 RT 90 REPEAT 8 [MAKE "C :C + 1 CIRC
:C ] ]
W 30 CS H CB 9 TI 95 SC 9 TT HT SC 6
REPEAT 4 [RT 90 WHEEL ]
RT 45
REPEAT 4 [RT 90 WHEEL ]
W 30 CS H CB 12 TI 95 SC 12 TT HT SC 15
REPEAT 4 [RT 90 WHEEL ]
W 30 CS H CB 2 TI 95 SC 2 TT HT SC 9 RT 45
REPEAT 4 [RT 90 WHEEL ]
W 30 CS H CB 2 TI 95 SC 2 TT HT SC 1
STAR
W 30 CS H CB 7 TI 95 SC 7 TT HT SC 1
TES
W 30 CS H CB 7 TI 95 SC 7 TT HT SC 15
REPEAT 36 [RT 10 WHEEL ]
W 30 CS H CB 3 TI 95 SC 3 TT HT SC 1
STAR RT 5 STAR
W 30 CS H CB 10 TI 95 SC 10 TT HT SC 9
REPEAT 36 [LT 10 WHEEL ]
W 30 CS H CB 14 TI 95 SC 14 TT HT SC 7
STAR RT 2 STAR RT 2 STAR
W 30 CS H CB 4 TI 95 SC 4 TT HT SC 15
REPEAT 4 [WHEEL RT 90 ]
SC 7 RT 45
REPEAT 4 [WHEEL RT 90 ]
W 30 CS H CB 14 TI 95 SC 14 TT HT SC 2
REPEAT 4 [TR 8 RT 90 ]
W 30 CS H CB 2 TI 95 SC 2 TT HT SC 15
REPEAT 4 [TR 1 TR 2 TR 3 TR 4 TR 5 TR 6 TR 7 TR 8 TR 9
RT 90 ]
W 30 CS H CB 13 TI 95 SC 13 TT HT SC 15
REPEAT 8 [TR 1 TR 2 TR 3 TR 4 TR 5 TR 6 TR 7 TR 8 TR 9
RT 45 ]
W 30 CS H CB 9 TI 95 SC 9 TT HT SC 15
REPEAT 36 [TR 1 TR 2 TR 3 TR 4 TR 5 TR 6 TR 7 TR 8 RT 1
0 ]
END
TO W :NO
WAIT :NO
END
TO H
HOME
END
TO TRI
REPEAT 3 [RT 120 FD 30 ]
END

```

```

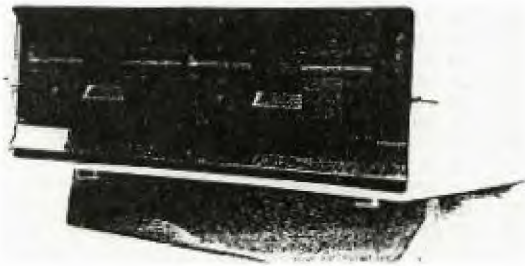
TO CIR
REPEAT 36 [RT 10 FD 4 ]
END
TO .
REF . 6 [RT 60 FD 10 ]
END
TO OCT
REPEAT 8 [FD 10 RT 45 ]
END
TO DES
REPEAT 10 [RT 36 FD 10 ]
END
TO CIRC :SIZE
REPEAT 18 [RT 20 FD :SIZE ]
END
TO SQ :SIZE
MAKE "SIZE :SIZE * 5
REPEAT 4 [RT 90 FD :SIZE ]
END
TO WHEEL
REPEAT 2 [RT 90 FD 40 RT 90 FD 15 ]
END
TO STAR
REPEAT 36 [RT 10 FD 40 BK 40 ]
END
TO ---
REI 4 [RT 45 FD 20 BK 40 FD 20 ]
END
TO TR :SIZE
MAKE "SIZE :SIZE * 5
REPEAT 3 [FD :SIZE RT 120 ]
END
TO TRR
FD 20 RT 90 FD 20 RT 133 FD 29
END
TO
REF . 4 [RT 90 FD 30 ]
END

```

Whew, Peter, that is quite a program. Thank you very much for it, and I hope you have time for some more. I can see that the Younger Set will become popular with every one if we keep up this standard! ○

## For Sale

### Dual disk drives in case



MDS, 59A Boronia Street, Kensington, NSW, 2033, are offering the "MDS Junior" to TI users only, as a special. It consists of 2 525A MPI FDD's disk drives in a case with power supply, wired to go and a 34 way cable, warrantee 60 days for \$299. The disk drives on their own for \$85 each. Also available, half height drives for \$120. Telephone (02)662-4884. Geoff Woods Electronics (02)427-1676, have Sakata dot matrix printers, SP-1200 Plus for \$399.



## Advanced Diagnostic Command Files

by John Paine, TIsHUG

The following files are samples of what you can do with the Command File option of the Miller's Graphics Advanced Diagnostics(c). Command files allow you to set up batch type files to run a number of repetitive functions and/or diagnostic and system maintenance routines.

The first file will allow you to format the second side of a single sided disk without even looking at or damaging the original side. The only thing you have to do is update the bitmap on sector 0.

```
[253][7]
Will format side 2 of DSX2 [253]
from SS format to DS. [253][253][7]
After formatting, sector 0 [253][253]
must be edited for DS disk [253][253][253][7]
PA [13][7]
SD 2 [13][7]
WT 40 S WT 41 S WT 42 S WT 43 S WT 44 S WT 45 S [13]
WT 46 S WT 47 S WT 48 S WT 49 S WT 50 S WT 51 S [13]
WT 52 S WT 53 S WT 54 S WT 55 S WT 56 S WT 57 S [13]
WT 58 S WT 59 S WT 60 S WT 61 S WT 62 S WT 63 S [13]
WT 64 S WT 65 S WT 66 S WT 67 S WT 68 S WT 69 S [13]
WT 70 S WT 71 S WT 72 S WT 73 S WT 74 S WT 75 S [13]
WT 76 S WT 77 S WT 78 S WT 79 S [13][7] BE [13][7]
```

All Done Now [253][253][253][253][7][7]

As you can see, that is not a difficult file to understand if you have ever read the manual. The WT command tells the disk controller to write to a track number, for example 40, where track 0 to 39 appear on the first side of the disk and tracks 40 to 79 appear on the second side. The S indicates single density, the [13] is a carriage return and [ ] is a space, if needed. For those of you that only have a single disk drive, just change the SD 2 (Select Drive) command to SD 1.

Another useful routine copies a Myarc 16 Sector (DSDB Format) disk to 18 Sector format that can be read and written to by a CorComp or AT controller card, by using the CorComp or AT controller alone.

This is a long command file and is given next.

```
[7]BEEP[13][7]
Place Myarc disk in drive 1 and press a key
[253][253][253][253][253][253][253][253][253][253]
[253][7]PA[13][7]BEEP[13][7]
Place CC disk in drive 2 and press a key
[253][253][253][253][253][253][253][253]
[253][253][7]PA[13][7]
SD 1 CR 0 16 SD 2 CW 0 16 [13]
SD 1 CR 18 16 SD 2 CW 16 16 [13]
SD 1 CR 36 16 SD 2 CW 32 16 [13]
SD 1 CR 54 16 SD 2 CW 48 16 [13]
SD 1 CR 72 16 SD 2 CW 64 16 [13]
SD 1 CR 90 16 SD 2 CW 80 16 [13]
SD 1 CR 108 16 SD 2 CW 96 16 [13]
SD 1 CR 126 16 SD 2 CW 112 16 [13]
SD 1 CR 144 16 SD 2 CW 128 16 [13]
SD 1 CR 162 16 SD 2 CW 144 16 [13]
SD 1 CR 180 16 SD 2 CW 160 16 [13]
SD 1 CR 198 16 SD 2 CW 176 16 [13]
SD 1 CR 216 16 SD 2 CW 192 16 [13]
SD 1 CR 234 16 SD 2 CW 208 16 [13]
SD 1 CR 252 16 SD 2 CW 224 16 [13]
SD 1 CR 270 16 SD 2 CW 240 16 [13]
SD 1 CR 288 16 SD 2 CW 256 16 [13]
SD 1 CR 306 16 SD 2 CW 272 16 [13]
SD 1 CR 324 16 SD 2 CW 288 16 [13]
SD 1 CR 342 16 SD 2 CW 304 16 [13]
SD 1 CR 360 16 SD 2 CW 320 16 [13]
SD 1 CR 378 16 SD 2 CW 336 16 [13]
SD 1 CR 396 16 SD 2 CW 352 16 [13]
SD 1 CR 414 16 SD 2 CW 368 16 [13]
SD 1 CR 432 16 SD 2 CW 384 16 [13]
SD 1 CR 450 16 SD 2 CW 400 16 [13]
SD 1 CR 468 16 SD 2 CW 416 16 [13]
SD 1 CR 486 16 SD 2 CW 432 16 [13]
SD 1 CR 504 16 SD 2 CW 448 16 [13]
SD 1 CR 522 16 SD 2 CW 464 16 [13]
SD 1 CR 540 16 SD 2 CW 480 16 [13]
SD 1 CR 558 16 SD 2 CW 496 16 [13]
SD 1 CR 576 16 SD 2 CW 512 16 [13]
SD 1 CR 594 16 SD 2 CW 528 16 [13]
```

```
SD 1 CR 612 16 SD 2 CW 544 16 [13]
SD 1 CR 630 16 SD 2 CW 560 16 [13]
SD 1 CR 648 16 SD 2 CW 576 16 [13]
SD 1 CR 666 16 SD 2 CW 592 16 [13]
SD 1 CR 684 16 SD 2 CW 608 16 [13]
SD 1 CR 702 16 SD 2 CW 624 16 [13]
SD 1 CR 720 16 SD 2 CW 640 16 [13]
SD 1 CR 738 16 SD 2 CW 656 16 [13]
SD 1 CR 756 16 SD 2 CW 672 16 [13]
SD 1 CR 774 16 SD 2 CW 688 16 [13]
SD 1 CR 792 16 SD 2 CW 704 16 [13]
SD 1 CR 810 16 SD 2 CW 720 16 [13]
SD 1 CR 828 16 SD 2 CW 736 16 [13]
SD 1 CR 846 16 SD 2 CW 752 16 [13]
SD 1 CR 864 16 SD 2 CW 768 16 [13]
SD 1 CR 882 16 SD 2 CW 784 16 [13]
SD 1 CR 900 16 SD 2 CW 800 16 [13]
SD 1 CR 918 16 SD 2 CW 816 16 [13]
SD 1 CR 936 16 SD 2 CW 832 16 [13]
SD 1 CR 954 16 SD 2 CW 848 16 [13]
SD 1 CR 972 16 SD 2 CW 864 16 [13]
SD 1 CR 990 16 SD 2 CW 880 16 [13]
SD 1 CR 1008 16 SD 2 CW 896 16 [13]
SD 1 CR 1026 16 SD 2 CW 912 16 [13]
SD 1 CR 1044 16 SD 2 CW 928 16 [13]
SD 1 CR 1062 16 SD 2 CW 944 16 [13]
SD 1 CR 1080 16 SD 2 CW 960 16 [13]
SD 1 CR 1098 16 SD 2 CW 976 16 [13]
SD 1 CR 1116 16 SD 2 CW 992 16 [13]
SD 1 CR 1134 16 SD 2 CW 1008 16 [13]
SD 1 CR 1152 16 SD 2 CW 1024 16 [13]
SD 1 CR 1170 16 SD 2 CW 1040 16 [13]
SD 1 CR 1188 16 SD 2 CW 1056 16 [13]
SD 1 CR 1206 16 SD 2 CW 1072 16 [13]
SD 1 CR 1224 16 SD 2 CW 1088 16 [13]
SD 1 CR 1242 16 SD 2 CW 1104 16 [13]
SD 1 CR 1260 16 SD 2 CW 1120 16 [13]
SD 1 CR 1278 16 SD 2 CW 1136 16 [13]
SD 1 CR 1296 16 SD 2 CW 1152 16 [13]
SD 1 CR 1314 16 SD 2 CW 1168 16 [13]
SD 1 CR 1332 16 SD 2 CW 1184 16 [13]
SD 1 CR 1350 16 SD 2 CW 1200 16 [13]
SD 1 CR 1368 16 SD 2 CW 1216 16 [13]
SD 1 CR 1386 16 SD 2 CW 1232 16 [13]
SD 1 CR 1404 16 SD 2 CW 1248 16 [13]
SD 1 CR 1422 16 SD 2 CW 1264 16 [13]
SD 1 CR 1440 16 SD 2 CW 1280 16 [13]
SD 1 CR 1458 16 SD 2 CW 1296 16 [13]
SD 1 CR 1476 16 SD 2 CW 1312 16 [13]
SD 1 CR 1494 16 SD 2 CW 1328 16 [13]
SD 1 CR 1512 16 SD 2 CW 1344 16 [13]
SD 1 CR 1530 16 SD 2 CW 1360 16 [13]
SD 1 CR 1548 16 SD 2 CW 1376 16 [13]
SD 1 CR 1566 16 SD 2 CW 1392 16 [13]
SD 1 CR 1584 16 SD 2 CW 1408 16 [13]
SD 1 CR 1602 16 SD 2 CW 1424 16 [13]
```

```
[7]BEEP[13]Conversion complete
[253][253][253][253][253][253][253][253][253][253]
[7]PA[13][7][ ]
```

As you may notice, this routine actually tries to read more sectors than available on the Myarc disk when formatted as 16 sectors per track (a total of 1280 sectors as against 1440). The extra lines of the conversion are needed because the CorComp controller only reads 16 sectors and then needs to skip 2 sectors. Sector 17 and 18 which do not appear on the Myarc disk but which are expected by the CorComp card are just ignored.

Note: The above file will run in my version of Advanced Diagnostics but my not run in the unprotected funny versions floating around. If the file does not load properly, you may have to split it in half and run it as two separate command files i.e. one command file per side of disk.

The above file should be a mandatory addition to any Club Secretary or librarian who deals with incoming disks from overseas. I was prompted to write this routine after receiving a disk full of GramKarte files and being unable to read or copy same (Thanks Ben!!!).

continued on page 10

## Navarone DBM File Recovery

by John O'Brien, TISHUG

This article is primarily for the benefit of users of the Navarone Database Manager. However the techniques used, I think, could be applied to any disk based filing system which has suffered data loss by failing to close a file correctly.

Over the past nine months I have been, for my own convenience, maintaining business records on my TI because of a lack of computer capacity at work. This sometimes can mean a typing session of anything from one to two hours if I happen to let things pile up. It therefore does not require a great imagination to picture the horror after say, more than an hour's work and maybe fifty records entered with only a few records to go, when out of the blue LOCKUP!, and no way out but to start all over again.

Whilst this is not a regular occurrence, because I generally clean on a weekly basis all module and port contacts, there is always the possibility of bumping the module or resting a hand on the widget board, which will occasionally cause the problem. When it happened the last time, (Easter Saturday), it was a wet day and with nothing much to do I decided to try to save all that work.

First I loaded a track copier and copied the disk. Next I loaded Disk Utilities and did a String Search for the last Key number that I knew was readable by the Navarone. When found, I noted the sector number and then went to Edit Sector and called up the sector. The next move was to read the next sector and there it all was, sector after sector of data. In fact it was all there except for the last couple of records (which would have been in the text buffer when the lockup occurred), there but not accessible.

After some thought, off I went on a search for a set of DV/BO files that I had downloaded from the BBS a couple of years ago and printed them out, namely, TI Disk Management by Bruce Caron. Armed with these files and a bit of knowledge pieced together at many long regional meetings, I began several hours of work, finally emerging with every file on the disk accessible to the Navarone.

So what one may say? Several hours work to save doing a couple of hours? The answer is that I had learned how to do something different. As a further proof of the procedure, I continued adding the remaining ten records to the file (five sectors) and then left the file with the widget reset button and fifteen minutes later ALL records were accessible.

Below I have attempted to set out the method I used. It should work for you. Anyway it is more fun than doing all that typing twice..

### Procedure

- (1) First make a CLONE of the disk to be adjusted, using a Track copier which will give an identical copy. Remove the original disk or open its drive door.
- (2) Load a disk utility program and dump to printer a Disk Report, and Sector Dumps of sector "0" and the File Directory sector of the file concerned (in this instance sector "3").
- (3) Select Sector Edit and call up the end sector noted in the disk report. Then manually advance, counting the sectors until a blank sector is encountered. Record the last USED sector number and how many new sectors were there. At this stage establish also how many new records are on these sectors.
- (4) Next, load E/A's Debug program and using the "H" command subtract the start sector from the end sector, add the number of new sectors to the total number of sectors in the file, and add the number of new records to the total number of records in the file (take the last two "old" figures from the #3 sector dump). All calculations must be in Hex.
- (5) Enter the figures obtained onto a copy of the chart below, working from the bottom up. The top line when completed is the new Block Link.
- (6) Reload the disk utility program and call up the Directory Sector ("3"). Change the numbers marked (\*) below, to the new ones calculated and write the sector back to disk.

- (7) Still using the utility program "DELETE" the corrected file from the disk and then use the "RECOVER FILE" option to replace it. This will update the bitmap on sector "0". (compare against sector dump!).
- (8) Next, reload the database and test for the missing records. They should all be accessible. DO NOT WRITE to the disk at this stage. If all records access OK, then exit the database.
- (9) Load DM1000 and "FILE" copy the recovered file back to its original disk. This will sort out any problems which may have been caused in the Directory Link Map (sector "1") by deleting and recovering the file.

### Disk Report

Diskname DBM\_DATA Total Sectors 718 Free Sectors 620

Filename	File Type	Size	FDR	Start	End	Pro	Comment
RDATA3	D/F	121	98 0003	002B	008B	No	Testfile

### Part dump of directory sector

M & T UTILITYWARE \*\*\*\*\* DISK + AID \*\*\*\*\* PRINT SECTOR SECTOR:0003

ADR- 1 2 3 4 5 6 7 8 9 A B C

00- 52 44 41 54 41 33 20 20 20 20 00 00

0C- 00 02 00 61 00 79 C2 00 00 00 00 00

18- 00 00 00 00 2B 00 06 00 00 00 00 00

### Breakup of above sector dump

+-----+

Filename

00- 52 44 41 54 41 33 20 20 20 20 00 00

Record length = 121

Records per sector: |

0C- 00 02 00 61 00 79 C2 00 00 00 00 00

\* Total # of records in file = 194  
( see note below )

\* Total # of sectors in file = 97

18- 00 00 00 00 2B 00 06 00 00 00 00 00

+-----+

### \* 3 Byte BLOCK LINK.

(Comprising 3 Start Sector nybbles (SS1,2,3). and 3 Record Offset nybbles (RO1,2,3)). All bytes marked (\*) must be adjusted for missing records to be accessed.  
SS2 SS1 - RO1 SS3 - RO3 RO2  
> 2 B > 0 0 0  
SS3 SS2 SS1 - - - RO3 RO2 RO1  
> 0 2 B > 0 0 0  
Start :02B # Sect :000 End :008

### Number of records in file.

This is a two byte field, so if more than one byte is used they will be reversed. For example if a file contains 423 records (01A7), the sector would read as >A701.

## QS-Writer Bug

by Tom Hare, PUG Peripheral

For those of you that have chosen to load up TI-Writer from XB with QS-Writer by Quality 99 Software and have the 99/4A Auto Spell Checker by Dragonslayer ASC, there is a bug in QS-Writer. This is that when the Dragonslayer spell checker is loaded from the utility option, there is not any control of lower case letters, regardless of whether your alpha lock key is up or down. That could cause some problems in editing misspelled words in your document. The fix is if you have the E/A cartridge, you choose option 5 RUN PROGRAM FILE and when prompted for FILE NAME? enter DSK1.UTIL1. Then you have upper and lower case.



# TISHUG NEWS DIGEST

## Uses of System RAM Addresses by the TI99/4A

by Ross Mudie, TIsHUG

This listing has been developed from information in INTERN, by Heiner Martin and the TI Editor Assembler manual.

The listing is not considered complete nor is it guaranteed free of errors. It is placed on TEXPAC BBS as it may be of assistance to other linked assembly programmers. The use of this document is to show at a glance memory usage for the routines and the equates for routines that I have commonly used. The use of FAC, ARG and VSPTR by the mathematical routines is not fully documented.

Your comments and additional information would be appreciated.

Decimal HEX	Usage.
-32000 >8300	Link ARGID 1,2
-31998 >8302	Link ARGID 3,4
-31996 >8304	Link ARGID 5,6
-31994 >8306	Link ARGID 7,8
-31992 >8308	Link ARGID 9,10
-31990 >830A	Link ARGID 11,12
-31988 >830C	Link ARGID 13,14: GETSTR
-31986 >830E	Link ARGID 15,16
-31984 >8310	Basic VSPTR when LINK is used.
-31982 >8312b	Link Argument quantity.
-31980 >8314	
-31978 >8316	
-31976 >8318	Pointer to start VRAM space used for strings (String space), Top
-31974 >831A	Pointer to end of VRAM space used for strings, Bottom. GETSTR
-31972 >831C	Tempory pointer to string (GPLLNK) GETSTR
-31970 >831E	Step for NUM mode
-31968 >8320	Pointer to start of screen input
-31966 >8322	Error code for assembler routines to GPL routines.
-31964 >8324	Pointer to start of Basic value stack and of character definitions -31962 >8326 Return address for the return of Basic Assembler routines to GPL
-31960 >8328	Address for jump table in GROM for the Basic commands.
-31958 >832A	Pointer to end of screen input ( may be located outside screen)
-31956 >832C	Pointer to text in a Basic line.
-31954 >832E	Pointer to line in-line list just executed.
-31952 >8330	Pointer to start of line # list (Highest line number)
-31950 >8332	Pointer to end of line # list (Lowest line number)
-31948 >8334	Pointer to next data element
-31946 >8336	Pointer to line in line list for next data element.
-31944 >8338	Pointer to actual position in Crunch buffer.
-31942 >833A	B=Pointer to free space used by PABs? x/b=Sub prog list pointer
-31940 >833C	Pointer to first entry in PAB list
-31938 >833E	Pointer to first entry in symbol list (variable list)
-31936 >8340	Pointer to free space which can be used by symbol list.
-31934 >8342	Executed character
-31933 >8343	Value of option base
-31932 >8344b	B=Run flag
-31931 >8345b	X/B=Run Flag : B0 xxxx xxxx B7 ; B0=Protect B1=On Brk Next B2= B3=Trace B4= B5=On Warn Stop B6=On Warn Next B7=RUN
-31930 >8346	Start line number
-31928 >8348	
	NN D X G LC C C CC
	UU S M E OH N S FI
Used in	MM R L T AR S N IF
	RA L L S DS
	ES N N T
	FG K K R
-31926 >834A	FAC +0,1 XX X X X XX
-31924 >834C	FAC +2,3 XX X X X XX

-31922 >834E	FAC +4,5 XX X X XX
-31920 >8350	FAC +6,7 XX X X XX
-31918 >8352	FAC +8,9
-31916 >8354	FAC +10,11 X X X Xb
-31914 >8356	FAC +12,13 X X X X
-31912 >8358	FAC +14,15 X
-31910 >835A	FAC +16,17
-31908 >835C	ARG +18,19
-31906 >835E	ARG +20,21
-31904 >8360	ARG +22,23
-31902 >8362	ARG +24,25
-31900 >8364	ARG +26,27
-31898 >8366	ARG +28,29
-31896 >8368	ARG +30,31
-31894 >836A	ARG +32,33
-31892 >836C	ARG +34,35 >836D-CDSR
-31890 >836E	VSPTR - Pointer to upper end of basic value stack
-31888 >8370	Pointer to end of usable VDP RAM, ie, start of disk buffers
-31886 >8372b	LSB of data stack pointer
-31885 >8373b	LSB Sub Stack pointer
-31884 >8374b	Keyboard number to scan (0-5)
-31883 >8375b	Key code from scan
-31882 >8376b	Joy Stick Y from Scan
-31881 >8377b	Joy Stick X from Scan
-31880 >8378b	Random Number generator
-31879 >8379b	VDP Interrupt Timer
-31878 >837Ab	Number of sprites in motion
-31877 >837Bb	VDP Status Byte
-31876 >837Cb	GPL Status Byte
-31875 >837Db	VDP Character buffer
-31874 >837Eb	VDP Current Screen row
-31873 >837Fb	VDP Current Screen column
-31872 >8380	
-31870 >8382	
-31868 >8384	X/B only, Highest usable address in expansion memory.
-31866 >8386	X/B only, End of free space in expansion memory.
-31864 >8388	Various flags B0 xxxx xxxx B7 : B0= B1= B2= B3=Trace B4= B5= B6= B7=
-31863 >8389	GROM/VDP Flag (0=VDP)
-31862 >838A	Subroutine and Data stack for GPL
-31860 >838C	as above
-31858 >838E	as above
-31856 >8390	as above
-31854 >8392	as above
-31852 >8394	as above
-31850 >8396	as above
-31848 >8398	as above
-31846 >839A	as above
-31844 >839C	as above
-31842 >839E	as above
-31840 >83A0	as above
-31838 >83A2	as above
-31836 >83A4	as above
-31834 >83A6	as above
-31832 >83A8	as above
-31830 >83AA	as above
-31828 >83AC	as above
-31826 >83AE	as above
-31824 >83B0	as above
-31822 >83B2	as above
-31820 >83B4	as above
-31818 >83B6	as above
-31816 >83B8	as above
-31814 >83BA	as above
-31812 >83BC	as above
-31810 >83BE	as above
-31808 >83C0	Random number seed
-31806 >83C2	TI99/4A Flag to control interrupt routine. Disable if 1 : B0= all, B1=Auto Sprite, B2=Sound, B3=QUIT, B4= , B5= , B6= , B7=
-31804 >83C4	User defined interrupt routine
-31802 >83C6	
-31800 >83C8	
-31798 >83CA	Console keyboard debounce
-31796 >83CC	Sound list pointer



```

-31794 >83CE Sound byte number
-31792 >83D0 } CDSR
-31790 >83D2 } Search pointers for GROM and ROM search
-31788 >83D4 Current value in VDP Reg 1 (mode)
-31786 >83D6 Screen timeout counter
-31784 >83D8 Return address saved by scan
-31782 >83DA Player? number used by scan
-31780 >83DC
-31778 >83DE
-31776 >83E0 GPL Work space Registers 0 XMLLNK
-31774 >83E2 Register 1 XMLLNK
-31772 >83E4 Register 2 XMLLNK
-31770 >83E6 Register 3 XMLLNK
-31768 >83E8 Register 4 XMLLNK
-31766 >83EA Register 5 XMLLNK
-31764 >83EC Register 6 XMLLNK
-31762 >83EE Register 7 XMLLNK
-31760 >83F0 Register 8 XMLLNK
-31758 >83F2 Register 9 XMLLNK
-31756 >83F4 Register 10 XMLLNK
-31754 >83F6 Register 11 XMLLNK
-31752 >83F8 Register 12 XMLLNK
-31750 >83FA Register 13 XMLLNK
-31748 >83FC Register 14 XMLLNK
-31746 >83FE Register 15 XMLLNK
-31744 >8400 Sound
-30720 >8800 VDP RD VDP Read Data register
-30718 >8802 VDP STA VDP ram read STAtus register
-29696 >8C00 VDP WD VDP Write Data register
-29694 >8C02 VDP WA VDP Write Address register
-28672 >9000 SPCHRD Speech Read Data
-27648 >9400 SPCHWT Speech Write
-26624 >9800 GRMRD GROM Read Data register
-26622 >9802 GRMRA Grom Read Address register
-25600 >9C00 GRMWD GROM Write Data register
-25598 >9C02 GRMWA GROM Write Address register

```

### GPLLNK data

```

DECIMAL HEX
20 >0014 Convert Number to String (STR)
22 >0016 Large Capitals (8 pixel)
24 >0018 Small Capitals (7 pixel)
32 >0020 Power up to master screen
34 >0022 Greatest Integer Function (INT)
36 >0024 Involution, Raises number to a power
38 >0026 Square Root (SQR)
40 >0028 Exponent Routine (EXP)
42 >002A Natural Logarithm routine (LOG)
44 >002C Cosine Routine (COS)
46 >002E Sine Routine (SIN)
48 >0030 Tangent Routine (TAN)
50 >0032 Arctangent Routine (ATN)
52 >0034 BEEP
54 >0036 HONK
56 >0038 Get unsecured string space
60 >003B Bit reversal routine
62 >003D Cassette DSR
74 >004A Lower case Tiny capitals (6 pixel)
BASIC >51A9 Garbage Collection
BASIC >56CD Scroll

```

### XMLLNK data

```

BASIC and EXTENDED
ASSEMBLY BASIC
>0600 >0D80 Floating Point Addition (FADD)
>0700 >0D74 Floating point Subtraction (FSUB)
>0800 >0E88 Floating point Multiplication (FMUL)
>0900 >0FF4 Floating point division (FDIV)
>0A00 >0D3A Floating point compare (FCOM)
>0B00 >0D84 Value Stack addition (SADD)
>0C00 >0D74 Value Stack subtraction (SSUB)
>0D00 >0E8C Value Stack Multiplication (SMUL)
>0E00 >0FF8 Value stack division (SDIV)
>0F00 >0D3A Value Stack Compare (SCOMP)
>1000 >11AE Convert string to number (CSN)
>1200 >12B8 Convert Floating Point to Integer
(CFI)
>2300 >20 Convert Integer to Floating Point
(CIF)
>00 Garbage Collection
>02 Get unsecured string space
>06 Convert Number to string (CNS)
>26 Scroll screen

```

### EQUATES and POINTERS

```

EXTENDED :IC shown * (BLWP @>xxxx)
>000E SCAN Scan keyboard (Use GPLWS and BL @SCAN)
>2002* First Free Low RAM Address (FFLA) 8194
>2004* Last free Low RAM Address (LFLA) 8196
>2008* NUMASG NUMBER ASsiGn Numeric to x/b
>200C* NUMREF NUMBER REFErence Numeric from x/b
>2010* STRASG STRing ASsiGn String to x/b
>2014* STRREF STRing REFErence String from x/b
>2018* XMLLNK Utility ROM routines
>201C* KSCAN Keyboard SCAN
>2020* VSBW Video Single Byte Write
>2024* VMBW Video Multiple Byte Write
>2028* VSBR Video Single Byte Read
>202C* VMBR Video Multiple Byte Read
>2030* VWTR Video Write To Register
>2034* ERR ERRor handling utility to x/b
>8300 PAD
>834A FAC Floating Point Accumulator
>83E0 GPLWS GPL Work Space registers
>8400 SOUND
>8800 VDP RD VDP Read Data register
>8802 VDP STA VDP ram read STAtus register
>8C00 VDP WD VDP Write Data register
>8C02 VDP WA VDP Write Address register
>9000 SPCHRD Speech Read Data
>9400 SPCHWT Speech Write
>9800 GRMRD GROM Read Data register
>9802 GRMRA Grom Read Address register
>9C00 GRMWD GROM Write Data register
>9C02 GRMWA GROM Write Address register

```

### CALLS in BASIC

```

CALL LOAD(-31860,96,41) Return to E/A Editor from
console basic after lock up.
CALL PEEK(-28672,A) :: IF A=96 THEN Speech is
connected.

```

## Moving Large Programs from Cassette to Disk

Author unknown

If any of you have been frustrated by the inability to load a large Extended BASIC cassette program onto disk and being able to save it from disk, read on. The problem is created by the disk system using about 3K of console memory to run itself, whether or not you have the 32K memory expansion card.

As some of you may know, you can free up approximately 2K of memory by typing and entering (one at a time) CALL FILES (1), and NEW and then loading your program.

Any Extended BASIC programs (in a disk based system) that are too big for the disk are not in program format but in INT/VAR 254 format. This makes use of the 32K memory. In order that you do not have to go through the CALL FILES routine every time you wish to use the program, do this routine once.

1. Type CALL FILES(1) <ENTER>
2. Type NEW <ENTER>
3. Load the program from cassette (or disk)
4. Type SAVE DSK1.program-name,MERGE <ENTER>
6. Go back to the title screen (to cancel CALL FILES(1) )
7. Re-enter Extended BASIC mode.
8. Type MERGE DSK1.program-name <ENTER>
9. After a lengthy wait for the above to finish, SAVE your program in the normal way. i.e. SAVE DSK1.program-name <ENTER>

The program is now on disk in INT/VAR format and will run in the normal way from disk.

The CALL FILES(1) routine will work with BASIC programs that are too big for the 16K memory console as well. If the BASIC program lends itself to RUNNING in Extended BASIC mode, (no problems with the graphics codes etc.) then load and save it in Extended BASIC, as described above. Just do not try to run the Extended BASIC saved version in BASIC.

## The Forth Column

Forth Forum <1>

by George L. Smyth

I would be the first person perhaps, to admit a lack of expertise with Forth, but I do realize a real need for a column concerning this language TI has placed in the public domain. There is a need for a tutorial for those just starting to learn the language. Just as there are numerous books that concern themselves with BASIC, few cover TI's version of this language exclusively. The same situation faces the prospective Forth programmer. TI's version of Forth is not examined in any book I have seen. Leo Brodie's excellent tutorial, 'Starting Forth' is as close as one can come and, by the way, is very highly recommended. The basic advantage here is that TI found fit to comb the book to find differences and mention them in one of the appendices. In my attempts to learn the language, I found certain sections short on explanation and other parts unnecessarily verbose. Of course this is merely my viewpoint, but I am sure that there are others who share my feelings on this. It certainly could not hurt to explore some of the niceties TI put into their system. There is also a need for a forum which will allow Forth programmers of all levels to exchange ideas. Questions come into my head by the minute it seems, and I find it unfair to rely on one or two individuals to find a solution to my personal programming dilemmas. Although I may have emphasized a certain portion of the language in my studies, I may have fallen short in other aspects, such as file processing. Another person may be in an opposite situation and in this respect we can use our strengths to shore up each others' weaknesses. A beginner can save the combined information for a later date of study and acquire a more rounded education in the language.

The purpose of this series of articles is to fill this gap. I plan to offer a tutorial to the beginning Forth programmer and to open the column to anyone with questions, words, suggestions, criticisms, or just plain talk concerning the Forth language. The more participation, the more informative we can be to each other. This column should run about once a month as long as I think anyone cares. With all that out of the way, let us make a Forth system disk that will be more convenient to use than the current configuration.

### A new Forth System disk

Use whatever means available to backup your Forth system disk and place the original in an inconvenient place. If I had a dollar for each time I blew out a system disk trying to do something I did not fully understand, I would have retired to Black's Beach long ago. This backup disk will now be used to configure several system disks that will be more convenient to use than the original. Upon booting Forth you are presented with an array of options which, when loaded, place additional words into Forth's dictionary. This, quite obviously, eats up valuable memory. The bottom line, however, is that beginners to Forth programming (or any language for that matter) are not going to write long, extensive programs. So for now, we will not worry about memory limitations. Here is a method for devising a system disk, for those of us just starting, which will boot and load a majority of the TI Forth vocabulary we may require. Start off by making another copy of the backup disk you just made. You will be overwriting all of the screens on this disk so if you foul up somewhere along the line you may have to start all over again. The backup made, boot the new system disk. When control is finally returned to your terminal, enter the following options:

-EDITOR -BSAVE -COPY -TRACE -GRAPH -VDPMODES

Your disk will kick on and off for about 2.5 minutes so be patient. It may say at one point " : isn't unique ". That is alright. I did not consider loading the 64 column editor because I feel that it is inappropriate for TV sets, it is a memory glutton, is error prone, and a general nuisance to sane people with normal eyesight. At least to start off, let us stay with the 40 column editor.

If the parameters your printer requires are 'RS232.BA=9600', you need not concern yourself with

this paragraph. If not, enter '72 EDIT'. Your disk should whirr and bingo, screen 72 will appear on your monitor with the cursor flashing at column 1, row 1. The parameters for the printer are found on line 4 between the quote marks. Use your arrow keys to move the cursor to this area to change this. Make sure that you leave a space between 'F-D' and your printer parameters. If you use the FCTN 2 key to insert information be sure that you do not scroll the 'OPN 3' word off the end of the line. If you do, be sure to type it back in! Once the printer parameters are correct, press FCTN 9 to place the cursor underneath the screen. Now type 'FLUSH' and press enter. The revision you just made is now being written back to the disk.

To test things out try entering 'SWCH printer should print the words in the quotes (note the space between the ' . ' and the text), and give a carriage return. If you have problems, try the procedure again, this time comparing the words on the screen with the printout found in Appendix I of your manual.

If everything is okay, enter -PRINT and let this section load into your memory. We now have enough words in memory to satisfy us for a while so let us put them back on our Forth disk in binary format. Enter 'TASK 20 BSAVE (the ' by the way is FCTN(0)). The computer is now overwriting screen 20 and successive screens with the words you saved when you entered the options. Screens 20 to 29 should be filled up with "garbage" now.

We will be using screens 30 to 89 to program on, so let us get rid of the information on these screens. Enter ': X 90 30 DO I CLEAR LOOP ;'. You should receive an "ok" at the end of your definition as the computer accepts this into memory. Now enter 'X' and wait for the disk drive to finish turning (it should be a few minutes).

We are almost done. Enter '3 EDIT'. This entire screen will have to be revised exactly as shown below (OK OK, you can enter your own name in place of mine).

```
Line numbers
0 ( Load screen )
1 BASE->R HEX
2 9 A GOTOXY ." Smyth's Forth loading!"
3 14 BLOAD 10 83C2 C! DECIMAL
4
5 0 DISK_LO ! 1 VDPME !
6 180 DISK_HI ! 71 7 VWTR
7
8 : MT EMPTY-BUFFERS ;
9 : SIZE SP HERE - . ;
10 : PAGE 0 0 GOTOXY CLS ;
11
12 PAGE
13 SIZE ." Bytes free"
14
15 R->BASE
```

Now, double and triple check the screen for accuracy. If you made a mistake here you may have to start all over, as your system may not boot properly if you goofed. Press FCTN(9) to leave the screen and type 'FLUSH' to write this screen back onto your disk. Entering 'COLD' will reboot the system and load in all those options you spent so much time waiting on, in just a few seconds. Also the colours on the screen will be more pleasing to your eyes (if not we will check on a fix next month). You should see.

7538 Bytes free

TI Forth

Finally you have a system disk which will load in the most commonly used words in just a few seconds. In a later installment I will go back over the procedure in case you want to customize a loading procedure for some specific purpose. I will also explain what is on the boot screen (#3) in a bit more detail. Next month I hope to begin a tutorial, starting from the first words the Forth programmer should be familiar with and on a monthly basis, continuing with additional words until we run out. With over 460 words TI Forth offers, I think that it may be a long time before we run out of words to learn.

If you have any questions, words, criticisms, comments, suggestions, etc., please contact me either on the TIBBS boards (434-0117 or 927-7079) or write to me:

George L. Smyth  
3017 Sylvan Dr.  
Falls Church, VA.  
22042

## Unpacking Forth Words

by Dicko, USA

The following group of Forth words will define a method to "dis-Forth" an existing Forth word. What I mean by that is the given word will be broken down into its component words (provided it is not an Assembly language primitive). I have stored these words -> in the order given - on screens 31 and 32. When I need to add them to the dictionary, I type "decimal 31 load".

To invoke the dis-Forth process, choose a word and type in " ' word TRACK". The apostrophe is required. Its Forth def- inition is TICK, and it find the word in the dictionary.

( ROUTINE TO TRACK BACK THROUGH A FORTH WORD'S COMPONENTS )

```
BASE->R HEX
: -COMPARE ( COMPARE STRINGS )
  OVER OVER + SWAP DO DROP 2+
  DUP 2- I - DUP
  IF DUP ABS / LEAVE THEN
  2 +LOOP SWAP DROP ;
: DISPWORD ( CFA --- )
  2+ NFA COUNT IF AND
  OVER OVER ' LIT NFA COUNT
  DROP -COMPARE ROT ROT
  CR OA SPACES TYPE 0= IF 2+
  DUP CR OA SPACES U. ENDIF ;
: DISPLIT
  CR OA SPACES U. ;
R->BASE -->
```

( CONTINUATION FROM 31 "TRACK" )

```
BASE->R HEX
: NEXTCFA ( ADDR1 --- ADDR1+2 ADDR2 )
  2+ DUP
DUP A446 ( ;S ) = IF R> DROP ENDIF
DUP 045F ( A/L RT ) = IF R> DROP
ENDIF DUP A030 ( Dict bot )
U< IF DISPLIT ELSE
DUP D200 ( Dict top )
U< IF DISPWORD ELSE
DISPLIT ENDIF ENDIF ;
: TRACK ( PFA --- )
  ( Invoked ' word TRACK )
  2- BEGIN NEXTCFA ?TERMINAL
  ( CLEAR to kill ) UNTIL ; R->BASE
```

Note that if you get caught in a loop, the ?TERMINAL UNTIL combination allows you to get out by pressing FCTN 4.

Happy Forthing. Dicko 71016,1553



"Have you anything just a ... .. jallible?"

## Forth Words by J. Volk, USA

```
( Moire Pattern ---- 22 Feb. 84 J. Volk )
( Load These Options First -GRAPH -VDFMODES ) GRAPHICS2
( X LOAD TO RUN >>> THEN 'GO' )
HEX : DC D RND 2+ 10 * 1+ DCOLOR ! ;
DECIMAL 128 CONSTANT YPOS 128 CONSTANT XPOS 132 VARIABLE X4
: QUAD1 128 0 DO DC YPOS DUP XPOS DUP ROT DOT SWAP I DUP -4 X4
+! X4 DUP ROT DDT SWAP LINE 4 +LOOP ;
: QUAD2 190 128 DO DC YPOS DUP XPOS DUP ROT DDT SWAP I DUP 4 X4
+! X4 DUP RDT DOT SWAP LINE 4 +LOOP ;
: QUAD3 128 190 DO DC YPOS DUP XPOS DUP ROT DOT SWAP I DUP 4 X4
+! X4 DUP ROT DOT SWAP LINE -4 +LOOP ;
: QUAD4 0 128 DO DC YPOS DUP XPOS DUP ROT DOT SWAP I DUP -4 X4 +
! X4 DUP ROT DOT SWAP LINE -4 +LOOP ;
: DESTRUCT ." TYPE 'FORGET DC' TO CONSERVE MEMORY " ;
: GO BEGIN 129 X4 ! QUAD1 0 X4 ! QUAD2 190 X4 ! QUAD3 255 X4 ! Q
UAD4 ?KEY UNTIL TEXT DESTRUCT ;
```

( MY MOST USED WORDS )

```
( LOAD -SYNONYMS FIRST )
: MYLOAD -GRAPH -VDFMODES ; ( Will load these options )
: AT GDTXY ; ( same as 'Display at' )
: TOP CLS 0 0 AT ; ( same as Brodie's 'PAGE' )
: RANDOM RND 1+ . ; ( n RANDOM >>> gives random number )
: PICK ( Leave copy of n1-th number on top of stack )
  ( n1 --- n2 )
  2 * SP + ;
: ROLL ( rotate nth number to top of stack ) ( n --- n )
  DUP 1 = IF DROP ELSE DUP 1 DO SWAP R> R> ROT >R >R LOOP
  1 DO R> R> ROT ROT >R >R SWAP LOOP THEN ;
: TEST BEGIN ." HELLO THERE" 2 SPACES ?TERMINAL UNTIL ; ( FCTN 4
TO END )
: SGN DUP IF DUP 0< IF -1 ELSE 1 ENDIF ELSE 0 ENDIF ;
: WORK BLOCK DROP UPDATE ; ( My word to update a Forth screen )
```

( A Word to copy Forth disks-Single Drive 5/16/84 J. Volk )

```
( Load Screen #N1 and -COPY then RUN )
0 VARIABLE COPYSCR 0 DISK_LO !
: MES1 COPYSCR 88 > IF CLS ABORT ENDIF TOP 2 11 AT ." INSERT M
ASTER DISK " KEY DROP ; (PRINT MESSAGE AND KEY PRESS )
: COPY1 5 0 DO COPYSCR WORK 2 20 AT ." SCR# " COPYSCR ? 1 COP
YSCR +! LDOP ; ( DO THE WORK AND LET US KNOW-GET NEXT SCREEN )
: COPY2 2 11 AT ." INSERT COPY DISK-ANY KEY " KEY DROP ;
( COPY 5 SCREENS AND PRINT MESSAGE )
: GETIT BEGIN MES1 COPY1 COPY2 FLUSH COPYSCR 89 = UNTIL ;
( RUNS ABOVE WORDS )
: MESO TOP 2 11 AT ." INITIALIZE FORTH DISK ? (Y/N) " ;
: MSG TOP 2 11 AT ." INSERT COPY DISK " KEY DROP ;
: RUN MESO KEY 89 = IF MSG 0 FORMAT-DISK DISK-HEAD ENDIF GETIT ;
( ROUTINE TO INITIALIZE DISK )
```

## Forth Words DIS/VAR 80 Transfers

```
( BUILD SCREEN FROM VAR 80 FILE. )BASE->R HEX
0 VARIABLE MY-BUF 50 ALLOT PABS 30 + MY-BUF 1900 FILE DISK-80
DECIMAL
: DISK-TO-SCREEN ( scr# --- )
  DISK-80 SET-PAB DSPLY SQNTL VRBL
  80 REC-LEN INPT F-D" DSK1.F-SCR"
  CR ." INSERT TI-WRITER DISK." CR ." F-SCR WILL BE READ. "
  KEY DRDP
  DISK-80 OPN
  DUP BLOCK
  16 0 00
  RD DROP
  DUP MY-BUF SWAP 64 CMOVE
  64 +
  LOOP DROP BLOCK DROP UPDATE CLSE CR ." COMPLETE" ;
R->BASE
```

```
( BUILD VAR. 80 FILE FRDM SCREEN ) BASE->R HEX
0 VARIABLE MY-BUF 50 ALLOT PABS 30 + MY-BUF 1900 FILE DISK-80
DECIMAL
: SCREEN-TO-DISK ( scr# --- )
  BLOCK
  DISK-80 SET-PAB DSPLY SQNTL VRBL
  80 REC-LEN OUTPT F-D" DSK1.F-SCR"
  CR ." INSERT TI-WRITER DISK." CR ." F-SCR WILL BE WRITTEN. "
  KEY DROP
  DISK-80 OPN
  16 0 00
  DUP MY-BUF 64 CMOVE
  64 WRT
  64 +
  LOOP DROP CLSE CR ." COMPLETE" ;
R->BASE
```

## Data Transfer between TI Forth Screens and Extended BASIC files

A tutorial by Leon Tietz  
(an Author of TI Forth)

This file originally appeared in the TINEWS section on TEXNET. This is a tutorial written by the author of TI Forth on methods of data transfer between TI Forth and Extended BASIC. The file does not contain any programs, but contains enough information on how the Forth disk is structured that a programmer can use it as a major reference source.

### Overview

If you are a TI Forth programmer or a FORTi music system user, you may be frustrated by the difficulty of sharing your latest creations with other TI99/4A users with similar tastes. This problem is brought about by a characteristic of Forth that dates from its inception: that disks are used without a file structure and that the programmer is responsible for tracking his disk contents.

This article shows some of the characteristics of how the TI99/4A disk system uses the diskette and how Forth's approach maps onto this scheme. This article will not actually present any code, however, after understanding what is presented here, anyone with a programming bent should be able to write an Extended BASIC program to move data between the two formats and the Forth hackers should have an opportunity to try the Forth hooks that have been provided in TI Forth to do file I/O. The programs to perform these functions will be easier for those of you with at least two disk drives, never the less, consider it a challenge to write the transfer program to use a single drive and to request the user to swap disks at the necessary times.

A quick and dirty solution should not be more than about 1 screen of BASIC or Forth. More elegant solutions with error recovery etc., will undoubtedly be somewhat longer. The solution presented here is based on the Single Sided, Single Density (SSSD) disk format. Those with other formats can either use SSSD for this job or figure out the equivalent structures for your disks, since either you or someone you got Forth from has converted it from SSSD and I do not know how you chose to convert it.

### Layout Basics

A TI99/4A disk has a total of 360 sectors on each diskette. These are arranged with 9 sectors per track and a total of 40 tracks (9 times 40 is 360). Sectors 0 through 8 are on the track nearest the outside of the diskette, 351 through 359 are on the track nearest the inside of the diskette. Each sector contains a total of 256 bytes of data. Forth uses the disk as a collection of 90 SCREENS, numbered 0 through 89, each with 1024 bytes of data. The following correspondence is ALWAYS true: SCREEN 0 is sectors 0 through 3; SCREEN 1 is sectors 4 through 7; ... SCREEN 89 is sectors 356 through 359.

Although Forth disks do not require a file structure, your TI Forth system has a file structure on the disk so that the Editor/Assembler can find TI Forth and so that the Disk Manager can be used to Backup the diskette. The Standard TI Forth diskette has 3 files:

```
FORTH      6 DIS/FIX 80
FORTHSAVE 39 PROGRAM
SYS-SCRNS 313 DIS/PIX 128
```

The file FORTH is a small boot program that reads the file FORTHSAVE and uses it to load the various parts of the TI Forth system into the proper memory areas. The remainder of the disk is included in SYS-SCRNS. The FORTi disk layout is basically the same, except the files are named FORTI, FORTISAVE and SYS-SCRNS. Part of the SYS-SCRNS file is devoted to FORTi's own code and bookkeeping.

Included with TI Forth is also a utility called DISK-HEAD. This utility is designed to take any formatted disk and by writing a carefully crafted SCREEN 0 to the disk, making the disk look like one large file (named SCREENS). SCREEN 0 thus becomes dedicated to disk file overhead, but the remainder of the disk can be used as Forth screens while retaining Disk Manager and BASIC compatibility. Figures 1 and 2 describe these disk layouts in detail, and should be consulted as you read on.

### The Details

Sectors 0 and 1 are always used for the file system overhead on the TI99/4A (that is why the Disk Manager says you have 358 sectors available, not 360). An additional sector of overhead is allocated for each file placed on the disk. These are normally (space permitting) allocated starting with sector 2 for the first file, sector 3 for the second file etc. The actual data portion of the file is stored beginning in the first empty sector found, starting with sector 34. If no empty sectors remain between 34 and 359, then the sectors between 1 and 34 are used for data.

The information in sector 1 (a list of pointers to other sectors) always keeps the files in alphabetical order. The names of the files on the TI Forth disk have been chosen with this ordering in mind, so that when they are copied with BACKUP, they will always be placed on the new diskette in this same order. The file FORTH first with its header sector in Sector 2 and data in Sectors 34 through 38 (for a total of 6 sectors). FORTHSAVE next with its header sector in Sector 3 and data in Sectors 39 through 76 (for a total of 39 sectors). SYS-SCRNS (the one we are really interested in) is last with its header sector in Sector 4 and its data in Sectors 77 through 359 and Sectors 5 through 33 (for a total of 313 sectors).

Notice that the Forth screens that are usually of interest (2-7 and 20-89) are contained entirely within this file.

Figure 1.

----Bootable Forth Disk----		
Forth Screen#	Disk Sector#	BASIC Record#
	0	xxx
0	1	xxx
	2	xxx
	3	xxx
	4	xxx
1	5	566,567
	6	568,569
	7	570,571
	8	572,573
2	9	574,575
	10	576,577
	11	578,579
~~~~~		
	31	618,619
	32	620,621
8	33	622,623
	34	xxx
	35	xxx
~~~~~		
	75	xxx
	76	xxx
19	77	0,1
	78	2,3
	79	4,5
	80	6,7
20	81	8,9
	82	10,11
	83	12,13
	84	14,15
21	85	16,17
	86	18,19
	87	20,21
~~~~~		
	351	548,549
	352	550,551
88	353	552,553
	354	554,555
	355	556,557
	356	558,559
89	357	560,561
	358	562,563
	359	564,565

end of disk

xxx = cannot be read from BASIC

## FORTRAN by D.N. Harris, TisHUG

Figure 2.

Forth Screen#	Disk Sector#	Disk-- BASIC Record#
	0	xxx
0	1	xxx
	2	xxx
	3	652,653
	4	654,655
1	5	656,657
	6	658,659
	7	660,661
	8	662,663
2	9	664,665
	10	666,667
	11	668,669

	31	708,709
	32	710,711
8	33	712,713
	34	0,1
	35	2,3
	36	4,5
9	37	6,7
	38	8,9
	39	10,11

	351	634,635
	352	636,637
88	353	638,639
	354	640,641
	355	642,643
	356	644,645
89	357	646,647
	358	648,649
	359	650,651

end of disk

xxx = cannot be read from BASIC

Each Forth Screen is composed of 8 of the 128 byte DIS/FIX 128 records (for a total of 1024 bytes). These records can easily be read or written by opening the file as a RELATIVE file and by using LINPUT #n, REC r: X\$ to read and PRINT #n, REC r: X\$ to write, where n is a file number designator and r is the record of interest. Thus reading or writing 8 records will move one screen. For screens from 20 to 89 the first of these 8 records is  $8*(SCREEN\#-19)-2$ . For screens 2 through 7 the first of the 8 records for a screen is given by  $8*(SCREEN\#+69)+4$ .

To read or write SCREENS 20 through 22 you would have to read or write each of the records from 6 through 29. Figure 2 shows the DISK-HEAD diskette arrangement. There is, however, only one file on such a diskette and all SCREENS from 1 through 89 are accessible. Notice, however that screen 8 will require some special attention and that you can easily derive the formulas for automatic screen# to record# conversion yourself.

To read or write SCREENS 2 through 7 you would have read or write each of the records from 654 through 709. Also remember that before you can successfully execute in DISK-HEAD Forth you must perform 0 DISK\_LO ! first in order to gain access to write to SCREEN 0.

### Conclusion

The information presented here should be adequate to allow you to transfer data between TI Forth and other applications on the TI99/4A. This should facilitate the exchange of Forth programs via TEXPAC and in just using your TI99/4A to the fullest extent. Even if you are not a Forth programmer (yet), a few of the disk structure basics were presented and can provide you the basis for further exploration. Note that since Forth can read or write anywhere on the disk, it makes an excellent vehicle for exploration. ○

```

C CONVERTS CENTIGRADE TO FAHRENHEIT
20 FORMAT (F15.5)
30 FORMAT (' ENTER CENTIGRADE WITH A DECIMAL POINT
  1 EVEN IF WHOLE')
40 WRITE (6,30)
50 CENT = 0.0
60 READ (5,20) CENT
80 FAHR = (1.8 * CENT) + 32.0
100 FORMAT(' FAHRENHEIT IS')
112 WRITE (6,100)
120 WRITE (6,20) FAHR
130 FORMAT (' ENTER 999.99 TO STOP')
140 WRITE (6,130)
145 FORMAT (F20.5)
C ***OPTION TO STOP*****
150 IF (CENT .EQ. 999.99) GO TO 270
155 FAHR2 = 0.0
C **FAHRENHEIT TO CENTIGRADE**
160 FORMAT (' ENTER FAHRENHEIT WITH A DECIMAL POINT
  1 EVEN IF WHOLE ')
170 WRITE (6,160)
180 READ (5,145) FAHR2
190 FORMAT (' CENTIGRADE IS ')
200 WRITE (6, 190)
210 CENT2 = (FAHR2 - 32.0) / 1.8
220 WRITE (6,145) CENT2
230 GO TO 40
270 END

```

This program is an interactive one. It will surprise many acquainted with the card punch system, that FORTRAN can be made interactive. The reader, by studying the above, can see that the program takes prompts encapsulated in FORMAT statements to be written to the screen, device 6, and accepts input from the keyboard, device 5. It is possible to write the program to access files, and to write to files. Output could be directed to a printer, or to a robot with a speech synthesizer, or the program could make a decision and then send instructions to a machine, according to nested FORMAT statements. FORTRAN is the language of real-time applications. It gets a bit creaky to have to write the prompt string as a FORMAT, and the numerical data must be specified as to length and where the decimal point goes, both for input and output. Extended BASIC users will remember the PRINT USING option. In FORTRAN there is no option. All numbers which go in and all numbers which come out must be allocated your guess of the size of the number, exactly. FORMAT(F20.5) allocates 20 spaces, one for the decimal point, one for the sign + or -, leaving 18 spaces, 5 of which are for the decimal part, leaving 13 spaces. Most temperatures have less than 13 places, assuming one wants to run a thermo-nuclear reactor and control it, without blowing up a continent, which would be a typical FORTRAN application involving temperatures. The F means it is a real number. If the output had to be handled as an integer, one could specify it with an I and say how much, for example allowing for a maximum of 13 spaces it would be FORMAT(I13), or one might want it in exponential form, say FORMAT(E13.5) meaning a five digit mantissa with 13 total characters. There is a lot more of this interesting language to learn.

Some readers probably already have thoughts of useful or intriguing applications, and a yen to get hold of a FORTRAN module. Ideally, a cartridge for the FORTRAN language should be available with instructions, but sooner or later I hope to find time to write an emulator in BASIC and somebody will hopefully do a compiler in Assembly, preferably one that could be run in Extended BASIC using 32K expansion, which most of us now have in one form or another.

There are string handling capabilities in FORTRAN which I will deal with subsequently, the least messy of which is the FORMAT method, in which the string goes into brackets as shown. This is equivalent to the BASIC statement;

```
A$="comment.....advice"
```

but FORTRAN takes the line number of the FORMAT statement and does what it is told, say WRITE(device number, line number) and the appropriate comment then goes to the appropriate device. I will have to hunt around and see if FORTRAN does anything like a DATA statement. Enough for now. ○

Continued from page 1

## Oops!

Thanks to George Meldrum of the Illawarra regional group for pointing out the following mistakes in the March and April TNDs. Unfortunately, I think I probably introduced two of them with careless attention to the position of end of paragraph, when reformatting to our column width. The other one is an improvement to what was published, as well as correcting an error. My apologies for the mistakes and profound thanks to George for pointing them out, and for providing us with the improvement. Here they are.

### March page 10

Extended BASIC Save Utility, half way down the column should be:

\*Start of "SAVE" program  
SAVE LIM1 0 missing label

### April page 7

George points out that the poke 'CALL LOAD(-31952, 255, 0, 255, 0)' wastes memory space. A better solution is (without the errors):

```
100 CALL INIT :: CALL LOAD(-31952, 255, 231, 255, 231)
```

This also is better than the line printed in the magazine, as this was missing one parameter and had an extra comma.

### April page 8

In the article on File Processing, near the bottom of the first column, there is a program in which one data statement does not start on a new line like it should. George also says that the data should have commas inserted in the center of each statement. It should have looked like:

```
180 CLOSE #1
190 DATA THIS IS A TEST OF THE SCREEN,AND SPEECH FILES
    ON THE
200 DATA TEXAS INSTRUMENTS 99/4A HOME,COMPUTER. IT
    SHOULD HELP
210 DATA TO DEMONSTRATE HOW ALL INPUT,AND OUTPUT IS
    TREATED,AS A FILE BY THE COMPUTER
```

Then there is a silly mistake in the program of the next column. The line at fault should read:

```
130 INPUT #2:X$
without the word FILE
```

My only excuse was that we were hurrying to produce a rather large TND before Easter intervened. I shall try to be more careful in future. Please, if anyone else finds what appears to be an error, please tell me and I will inform the rest of the readers as soon as possible.

Another error which crept in occurred in the drawings for the circuit of the real time clock from John Paine. The output of the first 74LS138, which connects to the second 74LS138, should be from pin 11, not 14. So all would be constructors, please note that very important change to the circuit in the March TND (page 5), and check your printed circuit board as well, to make sure that it is correct.

\*\*\*\*\*

There have been quite a few newsletters received this month. The first one I will look at is CIM99 from Club Informatique Montreal, Canada. As the name might imply, most of the contents are in French, which makes it a bit difficult for me. There are 3 issues starting with January 1988. This has a colour change program similar to the one in our last issue. There is an introduction to GPL by Jean Penne, an introduction to assembler by R. Felton and an article on using PRbase by Michel Johnson (all in French). One article in English is part 7 of Printer Capabilities by Lou Borrelli. Some clues are also given for an adventure game called Deadline (Infocom adventure). In the Febvier issue (interestingly, in a calendar of events which appears in each issue, the abbreviation for Tuesday appears as Thu, while that for Thursday is Thr) part 8 of Printer capabilities appears, and the third part of the PRbase article. There is also a listing of a c99 program called Spider Bop by John Behnke. There is part one of the clues for the adventure game Suspended (also an Infocom adventure as is Hitch Hiker's Guide to the Galaxy), and an article in French on the Geneve. In the March issue, part 7 of the Assembler course is continued as is part 4 on PRbase and part 2 on Suspended in both French and English. There is an assembly program for a clock in BASIC using Minimemory, a BASIC program for prime numbers, and various other tips by J.L. Mouquot, all in French (any

volunteers for translations?). There is also another article on the Geneve.

The March issue of Bytemonger from Lexington Kentucky, has an article on Fractals by Don MacClellan, another on speed comparisons for counting to 99999 and showing it on the screen and an article on constructing a switch to allow two printers to be connected to one PIO port.

The Northern NJ 99er's Users Group March issue has a program for producing a calendar, with the output also there. There is a program to test the TI 32K memory expansion and more by John Willforth on RAMdisks.

In the March edition of the Ottawa group newsletter there are the usual columns on the Geneve, Fast extended BASIC by Lucie Dorais, and BASIC by Steven Shaw, along with Questions and Answers by Bill Sponchia. Bill gives the clues on how to send command codes to the printer from Multiplan, and why raising the frequency of the crystal driving the processor causes problems with the RS232 ports. David Caron continues with part 4 of Expansion port interfacing looking at the TMS9901 chip and Lucie talks about music. Hal Tonkin also has a VDP tutorial and there is a games column which looks at backgammon and mastermind. In the April issue, David Caron, Lucie Dorais and Steven Shaw continue with their columns with more on the TMS9901, PRINT USING and relational operators.

The PUG Peripheral from Pittsburgh of December 1987 has Tips for beginners #7 by Frank Zic about printers, Tips from the Tigercub #42 by Jim Peterson, Getting the most from your cassette system #9 by Mickey Schmitt and other articles on Multiplan by Audrey Bucher and RAMdisks by John Willforth. In the January issue the series by Mickey Schmitt, John Willforth, Audrey Bucher, Frank Zic and Jim Peterson are continued and a series by Stan Katzman on TI-Writer is started.

Spirit of 99 from Central Ohio for December 1987 has some Tigercub growls and Tips from the Tigercub #45 from Jim Peterson with #46 in January and #47 in the March issues. Similarly, the TI-Writer series from Stan Katzman goes from part 3 in December to part 6 in March. A column by Jack Sughrue of Massachusetts called Impact-99 about general interest items appeared in December, January and March. In January there are articles on Multiplan by Bill Harms and Shuffling by Jim Peterson. Bill gives the clues for getting data from Extended BASIC to Multiplan, while Jim gives some different ways to program shuffling. There is information to build your own Joystick adapter, on the GRAMulator, and an article on Genealogy by Dick Beery. March has #5 of Mickey Schmitt's series on Getting the most from your cassette system, an article on connecting to a VCR by John Parkins, another on making your own envelopes with the addressee printed on it and a review of Data Base programs by Bill Goskill.

The Tacoma Informer for March has information on Cataloguing Library and the Companion disk for cataloguing all your disks (up to 123 at a time) by Marty Kroll and a program for generating row and column display of information by George Tilley.

TI-Sig Newsletter from San Diego for February has a hint on cleaning the print head of a dot matrix printer, and tips on installing your Extended BASIC module inside the console. In March there is mainly talk about people, plus some information about the Octopus display, and also about a proposed data base of all information ever written about the TI99/4A.

Bug Bytes from Brisbane for March has an article by Ralph Nielsen on synchronisation pulse separation from composite video signal. There is a game in Forth by Jim Zagielski, an article on the Geneve by Garry Christensen, a program by Harold Pierson for sector comparison in BASIC and Assembly and a program to print the overlays for the keyboard.

The March Hunter Valley 99'ers News has the usual informative roundup by Joe Wright, John Paine's article on the real time clock, one of Jack Sughrue Impact-99 columns, Random Bytes with Bob Carmany on Extended BASIC tips and BASIC class notes with Paul Mulvaney. For those into graphics, there is a good article on compatibility between all those packages by Don MacClellan, and a humorous article on an American's reaction to his first Jar of Vegemite by Harry Brashear.

MICROpendium for February has the results of a survey of 753 TI99/4A owners, of whom 103 were female. There are some interesting statistics there, although how they apply to our situation would be difficult to guess without running our own survey. However 9.4% did not have disk drives. There are two hints in the Feedback column for Extended BASIC, one for proper generation of random numbers in a program which does an auto load start (CALL PEEK(-31880, A, B) :: CALL INIT :: CALL LOAD(-31808, A, B)) and the other which allows a program to erase itself without erasing the screen or affecting assembly routines in low memory expansion (CALL LOAD(-31952, 255, 231, 255, 231)). There are programs on Trigonometry functions, cleaning the console, Nut-z game, Printing multiple columns continued from January, Cryptograms in c99, Bud Mills talking about hardware and the horizon RAMdisk and of, course, the articles on the Geneva. In the User Notes there are articles on turning off the CorComp screen when running a cartridge, on running the Tom Freeman checksum program, on label printing, a calculator program, extending the widget on a cable, installing XBII+ inside the console and the letter envelope design program from Richard Bailey. If you are interested in any of these items, then they are, or will be, in the library. \*\*\*\*\*

It always happens that when time is short, things go wrong more often. This is just a rewording of Murphy's Law, I guess, but this month we have had more than a fair share of problems. I was printing out some articles and I left the printer running, only to find when I returned, that the two ribbons had both been broken (black and red) and the daisy wheel was in about 16 pieces. I tried to get a new daisy wheel but Brother in Sydney were out of stock of daisy wheels of that type. They ordered from Sydney a wheel of another font of the right type for me. That did not arrive before the meeting, so I asked Chris to try to get one from Sydney. Meanwhile the wheel I had ordered came in, so I started printing again, only to find that there was a problem with the printer and the wheel was not being rotated to the correct place in time. This was undoubtedly why I had the problems in the first place, but it meant that I have had to print this issue with my own printer which has a smaller font. Actually, the font on the new daisy wheel is also rather small, so those with trouble reading the small print will be in trouble until we can return to the original font. I hope you all appreciated the slightly larger printing in the TND last month, as the reduction was not as much as it had been in previous months.

I have been slaving over TI-Writer for many hours now over the last few months and have found what appear to be bugs. This month I had several files become corrupted in that one sector of one file turned up in the middle of another file. The files were very fragmented and usually quite large. I also found that if I held down the CTRL key (instead of the SHIFT key) and pressed c followed by e, then the program would lock up with the cursor not flashing on the command line. Another thing which happens frequently to me, is that the line number command, FCTN[0], changes to perform the Next Window command, FCTN[5]. The only way to show line numbers is to press Escape, FCTN[9], followed by E to return to the editor. One thing I have done to make my life easier, is to make the formatter cycle around so that I can format any number of files without having to reload the formatter. To do this, you need a disk editor like Disk Fixer or the one I used, Disk Utilities. In the first file, FORM1, use Replace String to change the 2 occurrences of the hexadecimal address >B0F0, to >B704. Then do the same for the 1 occurrence of >B0F0 in the second file, FORM2 (or FORM1 and FORM2, if that is what you are using). To exit from the formatter, you must press FCTN[=] twice. \*\*\*\*\*

I promised to acknowledge all letters received, either in this column, or by printing them in full, where appropriate. I have received 3 letters from the directors, 2 this month and 1 the previous month. I would have hoped that at least one of these would have contained a message of encouragement or praise even, but no. There was no good news in any of them, except that they are allowing us to buy a paper cutter to speed up the cutting and pasting part of the job. Oh well, back to the keyboard!

continued on page 31

## Book Review

Technical Drive, by Monty Schmidt

Reviewer unknown, TIBUG

Technical Drive is a product of the recent revival of interest in the TI computer. It was written by Monty Schmidt and published by RYTE Data in Canada. It is not available in Australia so those interested will have to send overseas. The cost was US\$17.00 plus postage. I sent over US\$20.00, more about addresses later.

The book is 160 pages long and is supplied as punched and loose-leaf ready to be put into a ring binder. This has certainly reduced costs (no binding) but does detract from the image when you first open the envelope.

Regardless of the presentation, the important part is the information inside. This book is aimed at the experienced assembly programmer and those who want to know more about the functioning of some of the peripheral cards.

Technical Drive begins with a brief description of the CRU and its addressing layout, then lists the source code for a DSRLNK routine that can reside in the Mini-Memory.

Half a page introduces the CorComp clock card before a completely commented source code is provided. This is followed by several assembly routines to access the clock so that it can be used in assembly programmes.

Next comes the TI RS232 card. Once again a short introduction precedes the commented listing of the source code. A demonstration programme that writes to and reads from the RS232 directly is also listed.

The TI Disk Controller card is covered next. This section begins with a discussion of the layout of the disk and the meaning of the data stored in sectors 0, 1 and in the directory. The Disk Controller has 7 low level routines that can be accessed directly through assembly:

- Read/write to an individual sector
- Format a disk
- Modify file protection
- Rename a file
- Transfer direct to input file
- Transfer direct to output file
- Buffer allocation

Each of these routines is discussed and a listing of a routine to access each function is given. I found this the most interesting part of the book. After a couple of pages of description, the source code for the disk controller card is given.

I found that there were parts of the book that were interesting but I did not sit down and read it from cover to cover. You may want to read the entire dis-assembled source code for the disk controller, I do not.

Monty is to be congratulated on his work in deciphering the DSR's for the various peripherals. He has produced a book that stands alone as a source of information. Technical Drive functions well as a reference book. Go to it when you need information but do not expect to be able to memorise it all. What is the use when there is such a good book to look it up in.

This is not a book for novices. Some experience in assembly is necessary to make any use of it at all.

Where do you get it? RYTE Data have ceased to publish their newsletter and are not answering letters or the telephone. I would be dubious about sending any money to them. The best place to enquire is the author. The copyright is in his name so he should be able to help. His address is:

Monty Schmidt  
525 Wingra St  
Madison, WI, 53715  
U.S.A.

From TI BUG, (Brisbane) Jan/Feb 1988. Retyped by John Ryan of TISHUG for TEXPAC BBS.

## Regional Group News

### Meeting summary.

Banana Coast	8/5/88	Sawtell
Carlingford	18/5/88	???
Central Coast	14/5/88	Toukley
Glebe	12/5/88	Glebe
Illawarra	16/5/88	Keiraville
Liverpool	13/5/88	Campbelltown
Northern Suburbs	26/5/88	Belrose
Sutherland	20/5/88	???

### BANANA COAST Regional Group (Coffs Harbour area)

The March meeting of the Banana Coast Regional Group was held at the Sawtell Tennis Club with 10 members attending. This will now be our permanent meeting place as the committee has granted us permission to have our meetings on the 2nd Sunday in the month in their rooms.

One member who could not attend was Larry Wilmot of Bellingen, whom we believe has suffered a stroke. We wish him a speedy recovery and hope to see him at our meetings in the future, as his guile and story telling is sadly missed.

Great interest was shown as Keir Wells conducted a BASIC class for members, as it is a fact that programming techniques are at a low ebb and much can be learned by the more inexperienced owners of the TI.

It is hoped that the new Directors can do a good job and that one aspect of the group can be improved, delay in shop orders. We wish the Directors all the best in the forthcoming year and if there is any thing we can do, let us know. We intend to embark on an advertising campaign in this area to try and increase the membership.

For information on meetings of the Banana Coast group, contact Keir Wells at 9 Tamarind Drive, Bellingen, telephone (066) 55 1487.

### CARLINGFORD Regional Group.

Regular meetings are usually on the third Wednesday of each month at 7.30pm. Contact Chris Buttner, 79 Jenkins Rd, Carlingford, (02) 871 7753, for more information.

### CENTRAL COAST Regional Group.

Meetings are normally held on the second Saturday of each month (not April) at 6.30pm at Toukley Tennis Club hall, Header St, Toukley.  
Contact Russell Welham (043) 92 4000

### GLEBE Regional Group.

Regular meetings are normally on the Thursday evening following the first Saturday of the month, at 8pm at 43 Boyce St, Glebe. Contact Mike Slattery, (02) 692 0559.

The activities at these regional meetings are rather informal and include looking at new hardware, hardware repairs, looking at new software and having a general chat.

### ILLAWARRA Regional Group.

Regular meetings are normally on the third Monday of each month, except January, at 7.30pm, Keiraville Public School, Gipps Rd, Keiraville, opposite the Keiraville shopping centre. Contact Bob Montgomery on (042) 28 6463 for more information.

### LIVERPOOL Regional Group

Regular meeting date is the Friday following the TISHUG Sydney meeting at 7.30pm. Contact Larry Saunders (02) 644 7377 for more information.

### Meetings coming up.

May 13th 1988, at Stan MacPuckle's house, 15 Richmond Crescent, Campbelltown, (046) 25 6157  
June 17th 1988, at Hans Zecevic's house, 33 Malinya Crescent, Moorebank, 600 8716

Note, if you find it hard to get me at home due to the hours that I work, I will put on the BBS my work phone number, when I find out what it is.

### NORTHERN SUBURBS Regional Group.

Hello, just a reminder from the Northern Beach's Regional Group that we meet monthly on the fourth Thursday of the month at various locations. We have a lot of fun trying different things. We welcome any visitors. Our next meetings will be held as follows:-  
Thursday 26th May at Chris's home, 17 Lowanna Street, Belrose. Topic: Using printers effectively.  
Thursday 23rd June at Craig Sheehan's home, 21 Suzanne Rd, Mona Vale. Topic: Using PRbase.

Future meetings will be advertised at least two months ahead. See you at the next meeting.

Regular meetings are on the fourth Thursday of each month. Contact Dennis Norman on (02) 452 3920 or Dick Warburton on (02) 918 8132 for further information.

### SUTHERLAND Regional Group.

The Sutherland Regional Group has maintained a consistent but small number of regular members since its formation approximately 18 months ago. With the assistance of Terry Phillips I will again be contacting local members who are not attending Regional meetings, with the view to increasing our local participation.

In recent months the group has been concentrating on the use of TI-Writer and become more familiar with its many features. I, in particular, have derived great enjoyment from this "new" software, in producing numerous letters and other correspondence.

Of particular interest has been the hardware developments initiated by John Paine and Peter Schubert, especially the RAMdisk project. The time clock appears to be the next task on the horizon.

The TISHUG Shop continues to be an invaluable source of new and updated software and many thanks to the Merchandising sub-committee for their efforts.

Any persons interested in joining the Sutherland Group on the third Friday of each month, are more than welcome. The format of the meetings are very informal as more often than not the conversation digresses onto matters purely social rather than related to computerisation. The supper is not bad either.

Meetings are held on the third Friday of each month. Group co-ordinator is Peter Young, (02) 528 8775. BBS Contact is Gary Wilson, user name VK2YGW on this BBS.

### TISHUG in Sydney

Regular meetings are normally at 2pm on the first Saturday the month, except January and possibly other months with public holidays on that weekend, at the Woodstock Community Centre, Church Street, Burwood.

All TISHUG Regional groups are invited to submit items for this department. Send details as system mail to SYSOP.

continued from page 30

One of our members in the news is Les Andrews (DOBELL on the BBS), who appeared in 'Money Watch'. This is a magazine for those who contribute to a service for electronic information about the stock exchange. Les started his interest in the stock market just before the crash, but is hanging in there trying to recoup his losses. He is using his TI99/4A with a modem to keep abreast of the market. He has some software to help, but would be very pleased to hear of any other programs which help with the analysis of trends and the management of a share portfolio. Good luck to you Les. Anyone who could help Les, or has a similar interest, can usually find him at the Sydney meeting (medium height, black hair with a bushy moustache), or else try the BBS.

I would also like to thank Peter Schubert for upgrading my AT controller card to the latest version. I have not had any problems this month with tracks becoming not formatted. Apart from problems with TI-Writer (or DM1000), I have had no disk problems. If you only have a single sided drive as DSK1, then you will need a single sided version of the AT system disk, as the double sided version will just not work, for