



NEWS DIGEST

Focusing on the TI99/4A Home Computer

Volume 11, Number 2

March, 1992

Registered by Australia Post - Publication No. NBH5933

The TI-Faire Committee are hard at work.



See if you can help. Contact Dick on 918 8132

Sydney, New South Wales, Australia

\$3

We have changed our postal address. From now on please use:
PO Box 1089, Strawberry Hills NSW 2012.

March 1992

Index

All correspondence to:

P.O. Box 1089
Strawberry Hills, NSW 2012
Australia

The Board

Co-ordinator

Dick Warburton (02) 918 8132

Secretary

Terry Phillips (02) 797 6313

Treasurer

Geoff Trout (042) 29 6629

Directors

Rolf Schreiber (042) 84 2980

Russell Welham (043) 92 4000

Sub-committees

News Digest Editor

Bob Relyea (046) 57 1253

BBS Sysop

Ross Mudie (02) 456 2122

BBS telephone number (02) 436 4606

Merchandising

Percy Harrison (02) 808 3181

Publications Library

Russell Welham (043) 92 4000

Software library

Rolf Schreiber (042) 84 2980

Technical co-ordinator

Geoff Trout (042) 29 6629

TI-Faire co-ordinator

Dick Warburton (02) 918 8132

Regional Group Contacts

Central Coast

Russell Welham (043) 92 4000

Coffs Harbour

Kevin Cox (066) 53 2649

Glebe

Mike Slattery (02) 692 8162

Hunter Valley

Geoff Phillips (049) 42 8176

Illawarra

Geoff Trout (042) 29 6629

Liverpool

Larry Saunders (02) 644 7377

Northern Suburbs

Dennis Norman (02) 452 3920

Sutherland

Peter Young (02) 528 3775

Membership and Subscriptions

Annual Family Dues \$35.00
Associate membership \$10.00
Overseas Annual Dues A\$65.00
Overseas Surface Mail Dues A\$50.00

TisHUG Sydney Meeting

The March Meeting will start at 2.00 pm on 7th of March at Ryde Infant School, Tucker Street, Ryde. The Assembly Class will run from 10.00 am to 1.00 pm for those wishing to learn.

Printed by
The University of Wollongong
Printery Services.

Title	Description	Author	Page No.
Amateur radio log	Data base	Mudie, Ross	11
Assembly class for 1992	Software hints	Mudie, Ross	18
Beginning Forth #13	Software hints	Raguse, Earl	21
Co-ordinator's report	Club news	Warburton, Dick	2
Communicators	BBS information	Mudie, Ross	13
Decoding EPROM files part 3	Software hints	Takach, Ben	22
Editor's comment	General interest	Relyea, Bob	1
Extended BASIC tips #15	Software hints	Swedlow, Jim	16
GIF file transfer	Software hints	Schreiber, Rolf	19
GIF mania	Software review	Ruggieri, Alf	7
Notes on DSCAN, DSR scanner	Software review	Mudie, Ross	3
Page pro 99	Software review	Johnson, Ed	19
Page pro quick reference guide	Software review	Johnson, Ed	20
PP99 high res picture pointing	Software review	Johnson, Ed	20
Regional group reports	General interest		23
Secretary's notebook	Club news	Phillips, Terry	3
Tech time	Formatting problems	Trout, Geoff	5
TI-Bits #14	Software hints	Swedlow, Jim	16
TI99/4A world news	General interest	Peterson, Jim	17
TisHUG shop report	Club news	Harrison, Percy	4
TisHUG software column	Club software	Schreiber, Rolf	5
Treasurer's report	Club news	Trout, Geoff	2
Vapourware, slowware or no ware	General interest	Peterson, Jim	14
Word count	Program	Swedlow, Jim	6

Attention Newsletter Editors

All material appearing in this issue of the TND may be freely reproduced in other Newsletters provided that both the source and author are acknowledged. Articles are also available on disk as TI Writer files for a nominal charge.

Minor Corrections for February TND

Page 25, bottom right corner and page 28, bottom right hand corner should both have "continued on page 4" rather than "continued on page 2"

Editor's Comment

by Bob Relyea

I was late arriving at the February meeting as I had to take my mother to the international airport on the way. I enjoyed seeing everybody again at the meeting, although there were not heaps of people there by the time that I arrived at 3.40 pm. I enjoyed seeing my old friend Colin McKay there whom I had not seen in over a year since he went west to Condoholm to teach. He and his wife Cathy used to attend our meetings regularly and are still members of TisHUG. We taught at the same school in Campbelltown for four years and were/are avid TI enthusiasts. It is at that time of the year again that we are looking for local articles to print in the TND. Remember, any article from any member is most welcome. It is an encouragement to us to hear from somebody new no matter how 'simple' the article is as well as hearing from our old and established members who regularly contribute.

Co-ordinator's Report

by Dick Warburton

Why use a computer? Why have one? Do computers actually save money? Do they solve problems, or do they really create them? Do they make offices more efficient? Do they save paper? Do they solve all our money handling or sales transaction problems? I certainly think that they have had a profound impact in our society, and things will never be the same again, but I am dubious that computers have made our life better in every area they have been applied. How does your friendly teller machine treat you? Have you ever been frustrated by a computer? Are your bills always accurate? They should be, they are mostly done by computer. After all, our great Social Security Department relies on them for their faultless distribution of the nation's wealth. We can sleep easy, knowing that our privacy is in safe hands jealously guarded in Government computers.

Computers have made our life easier with cars. We do not have to worry any more about messy things like carburettors and old fashioned ignition systems. We can now rest easy because all we need is lots of money to tow our car to the nearest service centre equipped with the latest computer diagnostics and servicing facilities and leave it to the experts to ensure that we enjoy our motoring to the full without worrying ourselves about trivial things like what is wrong with the car. All we need for this computerised motoring utopia is a belief that computers are infallible and that the rising costs of motoring are worth every cent. We do not really have to do anything for ourselves. The computers are there to serve us, provided we can pay. It is interesting that 50 dollars worth of components can become worth 1200 dollars when they are made into a car computer.

Are computers making our life simpler or more complicated? Certainly they can take the drudgery out of repetitive manufacturing processes. They can reduce the complexity of tasks. Witness the changing role of the check out chick. Has her job really improved because of the computer. Does she have to think? I wonder whether the computer is actually making many people dependent on the computer and not on themselves. If computer dependency becomes too strong, people will lose skills and abilities and the judgement to know whether the computer is actually doing its job correctly. I recently came across a case at university where a computer in a science course consistently gave the wrong answers to important calculations. Students who had other answers were marked wrong. Because the computer produced the result it was deemed correct.

Will computers ever take over the world? Will they actually develop artificial intelligence which can rival the functioning of the human brain? What would happen if we programmed smart computers to repair themselves and to reproduce? What will happen when the largest and fastest computers are linked together? Is it possible that we will be able to make a hybrid computer human being, by connecting computing power to a human brain? With the incredible speed of advances in the computing world, each new advanced generation of computer is used to design and develop the next generation, at an ever increasing pace. I suspect that computers of the future will have the capacities to duplicate and even better most of the abilities of mankind.

Have you ever thought that computers are getting more and more like human beings. Certainly they are duplicating many characteristics which we consider are peculiarly human. Computers, like people, communicate freely with each other. Some actually talk or even sing. Strangely enough, communication problems are now developing. Why? Well just like the tower of Babel, different languages and dialects are springing up. Most educated computers can speak ASCII to each other, but they have difficulty with more graphic languages. Like human beings, computers have been told to "Go FORTH and Multiplan". Judging by the numbers being sold they have been very successful. Computers are procreating at an

ever increasing rate, partly because they have beaten human beings in the race to produce clones. Computers are able to "C" and TALK. My computer sings, plays musical instruments and has BASIC skills in numbers. Like people, computers get sick, or grow old or even die. Fortunately they can be resurrected with cryogenic techniques and brain transplants, or by using the famous Mudie Method.

Computers get hungry: hungry for power. They are the ultimate consumers; paper, ribbons, information, time and money.

Sometimes computers get neurotic. They react inappropriately and deliver strange messages on the screen. Sometimes they get very confused (with conflicting CRU addresses) and shut down to repair themselves and save us the embarrassing sight of them disintegrating on screen. Sometimes they get angry when they are overloaded and we become aware when we see the smoke before they really blow up. Computers are also subject to viral infections. The disease can be terminal. However with modern information transfusions they usually swiftly recover in skilled hands. Computers suffer blackouts, have memory lapses at times, and are subject to stress when overloaded.

While it seems that computers are becoming more like people, it may be that people are growing more like computers. Do we need to worry about computers? Will they take over the world? Well TI99/4A users do not have to worry about their machines. They may be capricious and frustrating, but definitely not a world threat. TI99/4A users develop a symbiotic relationship with their machines. It takes two to tango. TI99/4A users interact with their computers and are not in danger of becoming mindless computer morons. A TI99/4A is a tool, a powerful mind tool. It does exactly what it is told to do, faithfully and carefully. It does not argue or plot, it simply responds to its master's commands. TI99/4A users, because they interact more with the machine at all levels, are more likely to understand how and why computers act as they do, in fact become competent computer psychologists. It is hard for a TI99/4A user to grow completely dependent on either the software or the hardware. TI99/4A users have to fend for themselves. Computers are taking over the world, but I suspect that TI99/4A users are in a much better position to understand what is really happening than the legions of dependent computer addicts out there in every walk of life.

Dale Carnegie said that "if you can't beat them, you join them". Well in line with this we are definitely holding our TI-Faire in the last week of November 1992. At this time it seems likely that it will be held in the hall at Ashfield Boys' High School, near Ashfield Station. The location will suit most people, because of parking facilities, transport, availability of accommodation, shops food etc. Do not forget the date, the last weekend in November. Exams will all be over, the weather should be good and we will have a really great time. Ask yourself in the words of John Kennedy (paraphrased a little) not what the TI-Faire can do for you, but what you can do for the TI-Faire. If you have any ideas see me or any member of the committee. Join the TI-Faire group and give us a hand. I'll see you at the next meeting. O

Treasurer's Report

by Geoff Trott

We appear to have lost most of the overseas members who joined just over a year ago after our excellent publicity in MICROpendium. I am sure that they and all that remain, received excellent value for their money over the year. It would be nice to hear from more of our members with any comments or suggestions about how we can improve our service to you all.

Income for January	\$1281.43
Payment for January	\$241.18
Excess of income over expenses for February	\$1040.25

O

Secretary's Notebook

by Terry Phillips

The first meeting for the year is over and it was quite a good turnout; somewhere between 40 and 50 members and visitors braved the warm weather to come along and see what was on offer. Without doubt, the best thing I saw was the demonstration by Geoff of the TIM card. Those of you who have one of these on order are in for a treat when you get in installed and up and running. If you have not ordered one, get in and see Percy at the shop. The price is still holding at \$165 and it looks worth every cent of that!

New members continue to filter in and this month we have two to welcome:

Brian Bowers of Beecroft, and
Greg Oaks of Gross Vale

Hope you can both make it to some meetings this year.

Those present at the February meeting will recall Dick advising that it is hoped that the Faire to be held in November will be located at the hall of Ashfield Boys' High School. This venue, should it eventuate, will be ideally situated for the majority of members attending. Ashfield is a centrally located suburb with excellent rail and bus services. In addition there is a very large car park just over the road from the high school, primarily for the use of Western Suburbs Leagues club members, plus large local council and shopping mall car parks all within a couple of minutes walking distance. The best part of all these is that they offer free parking. Those interested in feeding the inner man will find the suburb caters for just about any type of food imaginable. The local club offers snack bar type facilities where a meal can be purchased for \$2 (usually 3 courses), while the local hotel has an excellent snack bar selling fast type foods for \$2 to \$3. As we get nearer the event and this venue firms to a reality I will give you more information of what is available. I happen to know this suburb fairly well having lived here for over 20 years.

The months of March and April will be very important months this year for the club. In those months the majority of memberships become due for renewal with April having the majority at just over 100. (The date of expiry of your membership is shown on the mailing label on the envelope in which your newsdigest is mailed). From my reading of overseas newsdigests the trend is to a rapidly declining membership base with several groups now ceasing to formally function. Latest to cease is the Channel 99 Users Group, established 1983, of Hamilton Ontario Canada. Our group has been exchanging software and newsdigests with this group for as long as I can recall. Anyway the point of this is for you to give some serious thought about your club and make a commitment now to continue with your membership. I am available at all meetings to accept membership fees. If you cannot get to a meeting write out your cheque and post it now.

Remember membership is now \$35 per year, with no additional fees for the Bulletin Board. If you wish to be able to access the BBS, contact the Sysop, Ross Mudie, for a user number and password.

See you at the March meeting. O

continued from page 6

```
170 A=X+1 :: X=POS(A$, " ", A):: IF X THEN C=C-(X>A)::  
GOTO 170 ELSE C=C-(A<LEN(A$))  
180 IF EOF(1) THEN CLOSE #1 :: DISPLAY AT(23,1):"FILE HAS  
ABOUT";C;"WORDS" :: GOTO 200 :: ELSE 190  
190 CALL KEY(3,K,S):: IF S<>0 THEN 190 ELSE PRINT C ::  
GOTO 160  
200 DISPLAY AT(24,1):"MORE? PRESS SPACE, ELSE ANY"  
210 CALL KEY(3,K,S):: IF S=0 THEN 210 ELSE IF K=32 THEN  
110 ELSE RUN "DSKI.DIR" O
```

Notes on DSCAN, the DSR Scanner

by Ross Mudie

This program was written with various intentions in mind. First, and foremost, is curiosity. Most people want to know what is inside of their computer. So do I. As a matter of fact, I want to know EVERYTHING! There was not a program that catered to my needs, so I created it; hence DSCAN. Second, is diagnostics. What can one do when one's system does not work correctly and it is not convenient to pull out all the expansion cards to pin down the culprit? DSCAN can help find conflicts. And thirdly, for programming. Someone wants to write an assembly program that calls subprogram >O110 (read/write sector) in the disk controller card, but wants to call the subprogram directly instead of using slow DSRLNK. DSCAN can retrieve that information for you.

Using DSCAN is quite simple. To load the program, type RUN "DSK1.DSCAN/XB" from Extended Basic (any variety should work), DSK1.DSCAN from Editor/Assembler option 5 or TI Writer option 3, or DSK1.DSCAN/O from Editor/Assembler option 3 or Mini Memory option 1. (The version of DSCAN on TEXPAC BBS is assembly implanted in extended basic. It is named DSCAN_XB, thus if saved using the name in line 100 type RUN "DSKx.DSCAN_XB" where x is the drive number in which the program resides).

You will then be presented with a menu and information area. The status area will have information concerning the current DSR. The information is the CRU address, the version number, power-up address, interrupt address, and what range of addresses contain RAM (if any). The menu choices range 1 though 7. Let's have a look at each.

1. Execute power-up routine. This will initialize the device at the current CRU address as if the computer was reset to the title screen. This works with many devices, but some devices would rather take over the system or mangle RAM here and there. Be cautious and knowledgeable with this function.

2. Execute interrupt routine. This will execute the interrupt routine at the current CRU address (if present). This is also a tricky function, but most devices will immediately give control back to the program.

3. Display DSR List. This will give a page listing of all the device names (if any) and their addresses present at the current CRU address.

4. Display Subprogram List. This will give a page listing of all the subprograms (if any) available to the card from DSRLNK or BASIC along and their addresses present at the current CRU address.

5. Next CRU Address. This will step to the next highest CRU address in your system and display its information at the top of the screen.

6. Previous CRU Address. This will step to the next lowest CRU address in your system and display its information at the top of the screen.

7. Exit Program. There is no turning back on this one. Once you hit 7, it is bye-bye back to the operating system.

The status line at the bottom of the screen will read one of five things at any given time. "Device power-up" if the device has a power-up routine and 1 is pressed. "Device interrupt" if the device has an interrupt routine and 2 is pressed. "First DSR" if you are pressing 6 and you are at the top of your DSR list. "Last DSR" if you are pressing 5 and you are at the bottom of your DSR list. And lastly, it may display nothing to signify that everything is peachy-keen.

BBS users can obtain a copy of this program from the Bulletin Board. O

TISHUG Shop with Percy Harrison

Whilst our February meeting was reasonably well attended, despite the fact that the magazine had not been posted out, the sale of official club software disks and commercial disks was somewhat disappointing as only two freeware disks and one commercial disk was sold on the day.

I would appreciate feedback from all members as to their opinion of the programs we are making available from the shop and suggestions as to what type of programs you think we should concentrate on or whether you consider the current mix satisfactory. Your opinions are quite important to us to ensure that we do not waste our time and money putting together disks of programs that are of no or little interest to our members. It would be greatly appreciated if you would take a few minutes of your time to write down and send me your thoughts on this matter, even if it is only to confirm that you are happy with the current situation. My address is 3 Storey Street, Ryde NSW 2112.

This month I have decided not to repeat the list of hardware that is available, as you can look back to the previous months lists, but instead I am listing all of the disks that are readily available from the shop at the time of writing this, all other club disks previously withdrawn from sale are available, at the standard club price, against firm orders. There will be a slight delay in getting these off to you as I have not yet received the library from Rolf.

PRICE LIST

CLUB SOFTWARE

A145 Adventures (Scott Adams) DSSD\$2.00
A184 Basic Builder SSSD\$2.00
A212 G Language SSSD\$2.00
A214 Plus Vers 1.0 SSSD\$2.00
A240 Star Vers 1.1 SSSD\$2.00
A249 Animation Demo's (2 Disks) SSSD\$4.00
A261 Assembly Language Games SSSD\$2.00
A338 Maze of Grog SSSD\$2.00
A354 Microdex 99 Vers 4.2 SSSD\$2.00
A380 Super-Cataloger SSSD\$2.00
A382 Boot (40 Column Vers) SSSD\$2.00
A385 Cartoon Kit SSSD\$2.00
A386 Boot (Hard Disk Vers) SSSD\$2.00
A399 Nuclear 99 SSSD\$2.00
A401 Pix Vers 1.2 SSSD\$2.00
A405 1000 Words SSSD\$2.00
A406 Hyphenator SSSD\$2.00
A407 Baseball Logos #1 SSSD\$2.00
A408 Baseball Logos #2 SSSD\$2.00
A430 Configuring Funnelweb SSSD\$2.00
A436 Hotbug SSSD\$2.00
A437 Nasty and Segregation SSSD\$2.00
A438 More Assembly Games SSSD\$2.00
A439 Multiplan Exercises DSSD\$2.00
A445 TI Base Tut/H Disk #8 SSSD\$2.00
A446 TI Base Tut/I Disk #9 SSSD\$2.00
A448 Tips Vers 1.7 SSSD\$2.00
A448A Tips Graphics #1 SSSD\$2.00
A448B Grips (Tips Companion) SSSD\$2.00
A449 Rockrunner Demo SSSD\$2.00
A450 Funnelweb 4.40 DSSD\$2.00
A450A Funnelweb 4.40 (3 Disks) SSSD\$4.00
A451 Multiplan Vers 4.02 SSSD\$2.00
A453 The Nutcracker Suite SSSD\$2.00
A456 Remembrance Disk (Music) SSSD\$2.00
A462 Rediskit SSSD\$2.00
A463 TI-Exam SSSD\$2.00
A464 Il Pastor Fido VivaIdi DSSD\$2.00
A465 Lute Music SSSD\$2.00
A466 Best of DOM #5 DSSD\$2.00
A467 The Singing TI SSSD\$2.00
A468 Speech #1 SSSD\$2.00

TCC1 Tigercub Collection #1 SSSD\$2.00
TCC2 Tigercub Collection #2 SSSD\$2.00

TCC3 Tigercub Collection #3 SSSD\$2.00
TCC4 Tigercub Collection #4 SSSD\$2.00
TCC5 Tigercub Collection #5 SSSD\$2.00
TC1122 Screen Fonts-Peterson DSSD\$2.00
TC1131 Gemini Printer Utilities SSSD\$2.00
TC1211 TI Artist Pictures #1 SSSD\$2.00
TC1212 TI Artist Pictures #2 SSSD\$2.00
TC1213 TI Artist Pictures #3 SSSD\$2.00
TC1219 R Kazmer's Xmas Card SSSD\$2.00
TC1220-1229 Tips (10 Disks) DSSD\$20.00

COMMERCIAL SOFTWARE

Artoons SSSD\$12
BABA Brewery Beer Labels SSSD\$10
Character Set & Graphic Design Cataloguer SSSD\$6
Character Set & Graphic Design I SSSD\$12
Character Set & Graphic Design II SSSD\$10
Character Set & Graphic Design III SSSD\$14
Disk Utilities (Memorial Edition) DSSD\$11
Disk Utilities (Memorial Edition) SSSD\$12
Display Master SSSD\$15
FilmLib Vers 3.0 (TI-Base) SSSD\$8
Genial Traveler SSSD\$6
Legends (2 Disk Set) SSSD\$30
McPaint (5 Disk Set)-DSSD\$10
McPaint (10 Disk Set)-DSSD\$20
Microdex I SSSD\$16
Microdex II SSSD\$11
Nuts and Bolts #1 DSSD\$6
Nuts and Bolts #1 SSSD\$7
Page Pro 99 version 1.6 SSSD\$28
Page Pro Utilities SSSD\$17
Page Pro Applications #1 SSSD\$2
Page Pro Line Fonts SSSD\$9
Page Pro Medical Clipart-DSSD\$10
Page Pro Medical Clipart-DSSD\$13
Page Pro Templates Vol1-SSSD\$8
Page Pro Templates Vol3-SSSD\$8
Picasso Publisher Version 2.0 SSSD\$14
Picasso Publisher Support Disk SSSD\$6
Picasso Applications Disk DSSD\$2
Pix Pro SSSD\$22
Rockrunner SSSD\$15
Screen Preview SSSD\$20
Smart Connect SSSD\$15
Spell It! (DSSD version)\$24
Spell It! (SSSD version)\$27
Star Trek (Calender) DSSD\$14
The Missing Link (TML) SSSD\$28
The Missing Link Companion Disk SSSD\$2
The Ring Companion SSSD\$12
TI Artist Plus SSSD\$25
TIA Fonts and Borders I SSSD\$8
TIA Fonts and Borders II SSSD\$8
TIA Fonts and Borders III SSSD\$10
TI Casino SSSD\$16
TI Sort SSSD\$15
Tris Module\$25
Typewriter Module\$25
Word Processor Harrison Software SSSD\$10
X Basher SSSD\$15
XB : Bug SSSD\$22

Packaging and postage charges:

	Surface	Airmail
1 to 2 Disks	----- \$1.90	1.90
3 to 9 Disks	----- \$2.90	\$3.60
10 to 20 Disks	----- \$3.90	\$4.80
TI Artist Plus	----- \$3.00	\$3.70
Display Master	----- \$3.00	\$3.70
TI Base	----- \$3.00	\$3.70
TI Sort	----- \$3.00	\$3.70
5.25 inch half-height drive (1.25 Kg)	-----	refer to your local post office

Bye for now.

TISHUG Software

Column by Roif Schreiber

There has been no news yet about the TIM 80 column cards that some of you ordered in October last year. I have been informed by Garry Christensen, through whom the order was organised, that Gary Bowser is assembling the units himself and then shipping them out in lots of five units. Please do not forget that OPA is just a one man operation, like so many other software and hardware vendors still supplying the TI99/4A market and it takes time to assemble and test a large number of circuit boards. In fact, Gary Bowser is such a keen supporter of the TI99/4A community that he has indicated to Garry Christensen his intension to attend our TI-Faire in November, all the way from Canada. How is that for support!!

The TIM cards will be supplied in the same order in which payment was received. If anyone missed out on ordering a TIM card last year, Gary Bowser of OPA is prepared to accept further orders (no minimum number) at the same discount price we advertised last year. Please get in touch with Percy for more details.

I would like to pass on the news that we have made a significant breakthrough in the process of converting GIF graphics from the PC format into the TI99/4A format. The GIF (Graphics Interchange Format) standard was developed by Compuserve in the USA as a means of transferring high resolution graphics (either colour or monochrome) by modem or direct RS232 connection between different computers. At present many thousands of GIF graphics and pictures are available for downloading from many BBSs around Australia and overseas.

This vast source of graphics is now available to TI99/4A Users. For those of us who already have an 80 column card, these pictures can be immediately displayed on the screen in up to 256 colours by a suitable GIF viewer, or used in a suitable 80 column graphics program. For the vast majority of TI99/4A Users who only have a standard 40 column display, this source of new graphics will also be available through the use of GIFMANIA, a commercial graphics viewer/conversion program which will shortly be on sale in the TISHUG shop.

Be forewarned that the conversion of the high resolution graphics (which are typically 640x480 pixels in size, in up to 256 colours), when displayed on a 40 column screen using only 13K of video memory, results in low resolution graphics of 258x192 pixels in 16 colours. However, the results are still recognisable and quite acceptable, if the original pictures did not have too many colours, or contained too much fine detail.

The converted picture can be saved to a disk file in TI-Artist format, either as a colour or monochrome image, where it is then available for further manipulation to anyone with a standard TI99/4A and either TI-Artist, GraphX, Page Pro, etc.

Software Releases for March 1992.

DISK A245 is called TELCO V2.0, a terminal emulator program by Charles Earl from Canada. He is the same programmer who worked on PRESS, the new word processor which was never released and HOTBUG, a sophisticated debugger which we recently released as disk A438. Telco is capable of XMODEM file transfers and is currently being used to transfer GIF files (high resolution graphics) from a PC to a TI99/4A, using a direct RS232 connection between the two computers.

DISK A468 was released last month, but was inadvertently left out of my column. The disk is called SPEECH #1 and has been compiled by Irwin Hott, a member of the Central Ohio Ninety Niner User Group (CONN1). Irwin is totally blind, but this has not stopped him from being the sysop of his club's BBS. He runs his

TI99/4A and the BBS through the use of a speech synthesizer. His efforts are a tribute to what can be achieved through the use of speech and a TI99/4A. All the programs require a speech synthesizer and I believe, the Text-to-Speech software. Dick Warburton claims that there are some problems in getting the programs to work with the speech synthesizer running out of Extended BASIC, but I cannot verify this since I do not currently have speech capabilities with my system.

DISK A469 is the second of four disks devoted entirely to speech. The programs require a speech synthesizer and the unlimited speech capabilities of the Text-to-Speech software package released by Texas Instruments.

DISK A472 is a TI-Writer Supplement disk compiled by the Chicago User Group several years ago. It is a collection of utilities for use with DIS/VAR 80 text files generated by TI-Writer and Funnelweb. There is also a printed manual which may be made available if there is sufficient demand.

DISK A473 is DM 1000 V5.0, the latest upgrade by Jack Mathis to the current version that has been around for the last 3-4 years. The new version corrects some of the bugs that were resident in V3.5 of this very popular disk manager. Two of the documentation files were missing from the distribution disk we received from the USA, but we expect to rectify this shortly.

DISK A474 is GIF Pictures #1, a DSDD disk full of assorted picture files in GIF format. An 80 column card and a suitable GIF viewer or graphics program are required to utilise this disk. The pictures may also be viewable on a standard TI99/4A by using GIFMANIA, a commercial program now available from the shop. ○

continued from page 13

The BBS allows you to communicate with other members and it is a simple way to distribute material of common interest.

If you are a TISHUG member and would like to start using TISHUG's TEXPAC BBS, then contact Ross Mudie on (02) 456 2122, (7pm to 9pm weekdays or weekends), to make the necessary arrangements.

The BBS number is (02) 456 4606; data rates 2400/1200 1200-75/300 bauds; 8 data bits; No Parity; 1 stop bit. ○

continued from page 20

Load picture - <CTRL-L>
Print page - <CTRL-P>
Quit from program - <CTRL-=>
Reset program - <CTRL-R>
Save page - <CTRL-F>
Screen colour changing - <CTRL-C>
Turn on/off pictures - <CTRL-O>

<CTRL-A> - Load fonts/lines
<CTRL-C> - Go through screen colour options
<CTRL-D> - Change cursor movement to right
<CTRL-E> - Change cursor movement to up
<CTRL-F> - Load/save pages Import/Export text
<CTRL-K> - Kill picture at cursor location
<CTRL-L> - Load picture at
<CTRL-O> - Toggle pictures on/off
<CTRL-P> - Print page
<CTRL-R> - Reset program
<CTRL-S> - Change cursor movement to left
<CTRL-U> - Erase all pictures
<CTRL-X> - Change cursor movement to down
<CTRL-=> - Quit program
<CTRL-8> - Turn on/off line drawing mode
<CTRL-9> - Turn on/off large fonts
<FCTN-4> - Break printout
<FCTN-9> - Back out of menu ○

Techo Time with Geoff Trott

Formatting problems

I was asked by Tony Bell to check out his Myarc Floppy Disk Controller card as it seemed to be doing strange things at times. The strangest was a change in colours of the screen when loading TI-Base from his RAMdisk but other programs would not load, the Myarc disk manager would not format disks and disks gave wrong file names full of strange characters. I had a look at the card and could not find anything wrong at all. Tony brought along his disk manager and RAMdisk and the disk manager did not work but TI-Base loaded fine on my system. The programs that did not work were not supposed to work with a Myarc disk controller. The problem with the Disk Manager turned out to be fixed by a new copy of the program. To get to the point of this story, I need to concentrate on the formatting of disks with different controller cards and different Disk Managers.

TI had a very well defined standard for how the information should be stored on disk which was unfortunately not the same as IBM chose for the PC. This information is stored mainly on sector 0 of every floppy disk. This sector contains the disk or volume name (10 bytes, bytes 0 to 9), the name "DSK" (bytes 13 to 15) which tells the disk controller that the disk is formatted, the total number of sectors on the disk (bytes 10 and 11, maximum of 65535), the number of sides (byte 18, 1 or 2), the density of recording (byte 19, 1 for single, 2 for double), the number of tracks per side (byte 17), the number of sectors per track (byte 12), the protection byte (byte 16, 32 for not protected and 80 for protected) and the bit map from byte 58 to byte 255. It may be expected that the disk controllers need to use all this information and should at least check it all to see that it agrees with their expected mode of operation. This is not so for all disk controllers, however, as only the Myarc disk controllers actually check and use all these values. The other disk controllers check some of these and ignore others which can and does lead to problems when swapping disks between systems with different disk controllers.

There are no problems associated with single density formatting. This is always done at 9 sectors per track. There can be any number of tracks up to 40 used, or if you have the higher density drives and a controller card which can handle it, there can be up to 80 tracks on each side of the disk. The 80 track drives just put an extra track between each of the normal 40 track positions. The standard TI Disk Controller will only handle single density format and up to 40 tracks, but it should be also possible, by changing the DSR program to allow 80 track drives to be used. It will not even recognise a double density disk at all. This means that the maximum storage capacity for a TI controller card is 180 Kbytes per disk (2 sides by 40 tracks per side by 9 sectors per track by 256 bytes per sector = 184,320), which could be expanded to 360 Kbytes per disk with a software change and 80 track drives.

The CorComp (AT and MiniPE as well) disk controller can handle double density recording. Double density recording allows more sectors to be recorded on each track and in the case of the CorComp controller 18 sectors are recorded on each track. In the case of double density controllers, they read in the first track when first accessing a disk and recognise whether the disk is single or double density from the bit pattern they receive. They have to be able to do this or there would be a "catch 22" situation of having to read the information on sector 0 in order to know how to interpret the data correctly on the disk. The CorComp disk controller does not check the number of sectors per track on sector 0 but just assumes that if the data is recorded in double density then it is done at 18 sectors per track. The maximum storage capacity for a CorComp disk controller card is 360 Kbytes (2x40x18x256=368,640). Like the TI controller card, the

CorComp disk controller card could be modified in its software to handle 80 track drives, which would make its capacity 720 Kbytes.

The Myarc Disk Controller cards (floppy controller and hard controller) can handle double density recording and also 80 track drives. However the problems arise because Myarc decided to provide for 16 sectors per track as well as 18 sectors per track. I guess the main problem is that the default format for double density used by the Myarc cards is 16 sectors per track. I presume that Myarc thought that 16 sectors per track would give a much faster track to track access than the 18 sector per track recording. This means that the Myarc controllers must inspect the sectors per byte value in sector 0 in order to find where a particular sector is on the disk. It also means that disks formatted by a Myarc disk controller card at 16 sectors per track are not readable by a CorComp disk controller past the first track. Since the first track contains most of the contents of the directory of the disk, things may seem to be fine until you try to read a file which is not on the first track. The Myarc disk managers give the option of 16 or 18 sectors per track when formatting a disk but all the other disk managers do not and expect the disk will be formatted to 18 sectors per track. The Funnelweb version 4.40 DiskReview recognises that there is a problem so there is a patch to be made in the file if you have a Myarc Disk Controller without an 80 track DSR. This patch is at offset >0A of the first file (DR) and consists of 2 bytes which should be set to >FFFF for a Myarc 40 track floppy disk controller. This will cause DiskReview to use a special 18 sector per track format routine, otherwise it will use the default 16 sector per track but put 18 sectors per track in sector 0. This means that the disk will not validate without error.

The other potential problem for owners of Myarc disk controllers is the bug in DM1000 version 3.5. This causes disks to be initialised at 18 sectors per track but 16 to be stored in the sector 0 sector per track byte. This can be checked by looking at byte 12 of sector 0 of any disk initialised by DM1000 as double density and this should be >12 and not >10. This error will cause no problems for owners of CorComp disk controllers which do not look at the contents of this byte. To fix the bug in DM1000, enter DiskReview and catalog the disk with DM1000 on it. Move the cursor to the first DM1000 file and then press "I" for inspect. Press "2" or "F" to select a search in a file followed by hexadecimal search. Change the wild card to FF and then type 100002D0005A. Leave the trailing FFs as they are ignored in the search and when the pattern is found, change the first byte from >10 to >12. There should only be one occurrence of the pattern. Write the sector back to the disk and then the bug will not be a problem for you or anyone else.

Word Count

by Jim Swedlow, CA USA
Modified by Earl Raguse, CA USA

Word Count was originally written by Jim Swedlow and later modified by Earl Raguse. This tight compact program shows each line of your DIS/VAR 80 file and then gives a running total of the words as each line is displayed.

```
100 ! SAVE DSK1.WORDCNT/ER
110 DISPLAY AT(2,1)ERASE ALL:"THIS PROGRAM COUNTS WORDS
IN A DISPLAY VARIABLE 80 FILE"
120 DISPLAY AT(5,1):"IT ALSO DISPLAYS THE FILE AS IT
COUNTS, THE NUMBERS AT THE ENDS OF A LINE ARE THE COUNT
THUS FAR."
130 DISPLAY AT(10,1):"THE ORIGINAL COUNT PROGRAM WAS
TAKEN FROM JIM SWEDLOW'S COLUMN IN THE JULY 87 ROM.
THIS MODIFIED VERSION IS BY EARL RAGUSE 11/89"
140 DISPLAY AT(16,11):"SPACE PAUSES, RELEASE GOES"
150 DISPLAY AT(19,1):"FILE:DSK" :: ACCEPT AT(19,9):A$ ::
OPEN #1:"DSK"&A$,INPUT
160 X=0 :: LINPUT #1:A$ :: PRINT A$;:: IF ASC(A$)<32 OR
ASC(A$)=46 OR ASC(A$)>127 THEN 180
```

continued on page 3

GIF Mania or Gateway into Fantastic Graphics

A review by Alf Ruggeri

The first part of the title identifies a GIF (graphics interchange format) file viewer with conversion to TI Artist environment program. The program was created by Barry Boone of Archiver fame and is marketed by Texaments. The alternate title, my own, identifies subjectively, the main facility that the program provides.

I was motivated to volunteer for this review for the sole purpose of determining the possibility of finding a graphic gateway between the world of the PC and that of our humble beast. In three words, yes it exists!, thank you Barry. In this article I propose to inform our TISHUG community on the marvelous features of GIF Mania but before I do, I would like to acquaint the reader with a somewhat involuted anecdote. Call it a quest, for a graphic gateway between the TI99/4A and the PC. It was frustrating at the time and fraught with many bleary hours of expectation in front of the keyboard, yet at times humorous in hindsight.

PRELUDE TO THE QUEST

Highest on my wish list of desirable features for the TI99/4A, since I started working with graphics about three years ago, has been an image digitization or scanning facility. This feature may eventually come about if the continuing development of such exotic hardware improvements as the TIM board is any type of indication. But in its absence an alternate solution has been sought i.e., a method of transporting scanned graphics from a relatives's PC to the TI99/4A.

TRYING TO PICK A LOCK

After many unsuccessful attempts at converting PC PCX files to TI99/4A graphics readability, using the CorComp TI-IBM module and Mike Dodd's PC-Transfer program for transportation of ASCII character files between the two realms, I took up the subject with Arto Heino at a regional meeting. Arto suggested that I should try to transport a PC GIF file with PC-Transfer and then use GIF Mania to view the result. The transported GIF file format, was as anticipated, in DV80 format. After all PC-Transfer is only intended to transport text files. I reconfigured the DV80 to DF128 (the same file format as the sample files that came with GIF Mania) first by using Larry Saunders's Writer Ease program and ultimately with a program of my own design.

My program:

- (a) Read in the DV80 file.
- (b) Segmented the 80 byte strings into suitable lengths, as determined by an algorithm that I worked out.
- (c) Concatenated the segments into 128 byte strings, using the same algorithm.
- (d) Saved the concatenated strings as a file in DF128 FORMAT.

When the converted file was presented to GIF Mania, the initial result was most encouraging. The GIF file header was accepted. The GIF file display program prompts were activated and the first few lines of graphics were displayed. Unfortunately that is where the encouragement abruptly stopped. The problem, simply stated, was that both the CorComp module and PC-Transfer only transported less than 10% of the PC file. The remainder, presumably the ASCII characters below 32 and a large number of characters above 32 were ignored. So much for yet another attempt at disk to disk conversion.

A DILEMMA

The clouds of gloom very rapidly set in. A dilemma! How was I going to meaningfully promote the use of a program, that for all intents and purposes represented an excellent graphic viewer, with a few, yet none the less, very striking pictures. A greater supply

of GIF files in the TI99/4A realm had to be made available otherwise the program had little use. And certainly of no use at all for my expectation of transporting scanned images.

DEEP THOUGHT

I re-read the TEXAMENTS documentation. Apparently the concept of the GIF format, was originally developed for bulletin board interchange of graphics files by computers of different manufacture, including the TI99/4A. Unfortunately the download protocol for the TI99/4A was not mentioned. Presumably this is common knowledge in the USA where possibly more use of PC oriented bulletin boards is made by TI99/4A users. I immediately wrote to Barry Boone, to obtain details of acceptable bulletin board GIF file download protocol. I also asked him to indicate a possible source of TI99/4A realm GIF files in the USA, for the purpose of import. Realizing that perhaps a considerable delay might occur, before the requested details were forthcoming and place a meaningful review of GIF mania on hold, I decided to pursue BBS download on my own.

I was rather hesitant. Perhaps closer to the truth, apprehensive, to dive in at the deep end of downloading a file from a PC bulletin board. Not because of any naive sense of trespass. But the fact that I had read in so much different RLE file downloading documentation, which emphatically indicated that only certain transfer protocols were able to receive and store the graphic file, without corrupting the contents. For the life of me I could not remember the recommended protocol, or indeed, the place where I had safely stored the relevant documentation.

I carefully refreshed my memory, with the TELCO documentation for any mention of GIF file download, that I may have not considered important, when I first read that document several years earlier. The only reference was that the XMODEM protocol stored the downloaded file in DF128, the same format as the GIF Mania sample files. A further reference specified ANSI terminal emulation as being applicable for a graphics driver with IBM V3.64. The significance of this reference was meaningless to me, but as it applied to graphics and IBM, the origin of GIF files, I decided to incorporate it.

TILTING AT WINDMILLS

Armed more with determination than confidence, I sallied forth with my battle scarred 300 baud modem into the Quixotic windmills of PC bulletin boards. By luck more than design I managed to gain access into the GIF file download section of a limited visitor access BBS. However when I requested a download, my trusty lance broke in half. The BBS would not download GIF files at less than 1200 baud. I very gingerly backed out, signed off and went back to ponder the problem.

SUCCESS AT LAST

At the time a 1200 baud modem was unavailable, so with a very patient Steven Carr with his 300 baud modem, parked at one end of a phone line and myself at the other end, Steven was able to send through his XT PC a small GIF file. The result was momentous to say the least. I am sure I could hear the ghost of Alexander Graham Bell sniggering to himself in the background noise of my PE box fan. The received file was readily accepted by GIF Mania.

The success of the transfer was marred only by my upload of one of the GIF Mania sample files. The file was successfully downloaded by Steven, but for some inexplicable reason TELCO decided to add its own header to the file i.e. "TI FILE" followed by half a dozen control codes. This additional header virtually corrupted Steven's file. His XT PC GIF file viewer refused to display the file.

In a more recent experimental GIF file transfer with Steven, in order to deliver an uncorrupted file, I used the XMODEM features of MASS Transfer and FAST-TERM,

uploading of a file was aborted on both counts. I do not know whether this problem arose because neither of these two modem programs support the ANSI terminal emulation. I would appreciate feedback from any source on this problem.

As a result of my successful download via Steven, I have ventured back to the windmills for a return joust. This time with a borrowed lance, a 1200 baud modem. The download went through without any splinters.

A SHORT DIGRESSION

I thank for your patience in reading this far the saga of Alf Ruggeri's quest for the graphic gateway. The review proper will follow after a short digression. The reader may think that my quest may have been a little frivolous if not blatantly masochistic. After all why not be done with it and buy a darn PC.

I certainly appreciate the awesome capabilities of PC's. My professional pastime, the one that puts bread on the table, involves the daily use of a PC equivalent to an enhanced IBM 486 instrument. The Computer Aided Design and Manufacture software, that gives life to the beast, would only a few years ago, have been considered as just another perennial promise from Tomorrowland.

The simple truth is neither the necessary cash flow, or enough genuine needs have materialized up to the present time, to justify the acquisition of a PC for home use. Of course ultimately I will upgrade from the TI99/4A. But not while new software and technical improvements still continue to issue forth, coupled with the prevailing spirit of comradeship at the local user group and global levels of the TI99/4A community.

AT LAST! THE REVIEW

As mentioned earlier, the term GIF signifies Graphic Interchange Format. It is a universal graphics file format that was originally developed by COMPUERVE (reputedly the largest on-line information service in the USA) so that users of different types of computers could exchange graphic files. At the time of GIF Mania's release, 1990, TEXAMENTS indicated that over 100,000 GIF files were in existence throughout the world. It would be difficult to estimate the current number of GIF files, as I am aware of several PC graphics programs that have a GIF output option as standard.

MINIMUM HARDWARE REQUIREMENTS

- (a) Extended BASIC or Editor Assembler module.
- (b) 32K extended memory.
- (c) One disk drive.

It is not necessary to use an 80 column card with this program. As a matter of fact I do not know if it will work with that card.

LOADING GIF Mania

GIF Mania may autoloading from Extended BASIC, or run using the RUN "DSKn.LOAD" command. Alternatively it can be loaded by Option 5 of Editor Assembler and input DSKn.GM.

PROGRAM FEATURES

When the program is loaded, a menu screen of five options is displayed:

- 1-Load GIF/TI ARTIST
- 2-Save TI ARTIST
- 3-View picture buffer
- 4-Disk catalog
- Q-Exit program

The significance of each option is self explanatory with the required option activated by the key entry of 1, 2, 3, 4, Q.

OPTION 1

This option provides a display of a GIF file or a

TI Artist picture. IF a GIF file is loaded, that image's original number of colours and horizontal and vertical pixel size will be indicated. Five sequential prompts will be presented:

(1) Color Select (I/D/G/M)

- "I" indicates intensity.
- "D" indicates deviation.
- "G" indicates grey scale.
- "M" indicates monochrome.

Images with up to 32 original colours are well reproduced by "I" or "D". For most of these images, selecting "I" will produce the best results. However there are definitely occasions where the use of "D", will highlight adjoining colour area inflections, not apparent with "I". If an image contains 256 original colours, the resulting display via "I" or "D" will be quite splotchy. This result should not come as a surprise, as the TI99/4A is only capable of supplying 16 colours. Selecting "G" can improve the displayed image by a reasonably effective black-grey-white tonal substitution for the original colour palette.

On selecting "G" or "M", a Brightness adjustment prompt, ranging from 1 to 255 will be presented. The numerical input will be used to produce an adjustable differentiation between black and grey or white. Varying the input value will greatly affect the final image quality. If the input value is too high, the image will partly vanish.

(2) Black Line Mode? (Y/N)

Selecting "Y" will produce an image black line boundary.

(3) Condense? (Y/N)

Selecting "Y" will proportionally reduce the image originally generated on a PC EGA OR VGA monitor, so that it will maintain an aspect ratio relatively free of boundary degradation. Alternatively a window of 256 by 192 pixels with its origin at the image top L.H.S. will be displayed.

(4) Left Shift: and (5) Up Shift:

will allow the 256 by 192 pixel viewing window to be positioned on various portions of the image, providing "N" was entered for the Condense prompt.

OPTION 2

This option will allow the user to convert the GIF image, that is currently displayed, to TI Artist format. If the picture is to be printed out via TI Artist, or is to be further converted to Page Pro for example, it must first be displayed in monochrome.

OPTION 3

The image can be turned off by any key entry e.g. conversion to TI Artist, the image can be redisplayed if desired by this option.

OPTION 4 is self explanatory.

OPTION 5 is self explanatory.

COMMENTS

In considering the limitations of our vintage computer and what this program has done for it: transportation of high resolution images created on sophisticated PC's and for me, a graphic gateway, it is almost ungrateful to ask for more. Yet a TI Artist to GIF file conversion feature would have made it a true two-way gateway.

NULL MODEM GIF FILE TRANSFER

After the successful download through Steven Carr's XT PC and then the PC BBS, I contacted Rolf Schreiber. Rolf and I had discussed possible GIF file conversion at length and I was very keen to let him know the results. Rolf listened attentively to my news and then completely

floored me. He told me that he and Geoff Trott had connected an improvised null modem cable between Rolf's 386 PC and his GENEVE. The result, successful one-way transfer of GIF files at 9600 baud between the PC and the GENEVE. I say one-way, because when Rolf tried to send a file from the GENEVE to the PC, using TELCO, he encountered the same irritating problem that Steven Carr experienced with my initial upload i.e. a corrupting file header of "TI FILE" plus half a dozen control codes. Any helpful suggestions?

Rolf intends to put together a very descriptive article on his configuration in the very near future. At the first available opportunity, I intend to try out Rolf's configuration for myself. It would appear that the elusive graphic gateway has opened up in a most spectacular way.

GIF FILES AND OUR BBS

Whilst designing my disk to disk DV80 to DF128 conversion program, it occurred to me that should I eventually succeed in generating TI99/4A graphic readable GIF files, they could be loaded up on TEXPAC and perhaps serve as an incentive for more member usage of our BBS. When the GIF-modem penny eventually dropped into place and later Rolf's null modem news, I immediately designed a DF128 to DV80 conversion program to enable upload of the GIF files as DV80 text files to our BBS. My original DV80 to DF128 program could be used to reconstitute the files on download.

At this point, I realized that perhaps our BBS might not be transparent to all the ASCII characters contained in the files. I contacted Ross Mudie and sure enough, the backslash and carriage return characters would not get through, as their use was reserved for normal BBS software operation. An XMODEM facility for our BBS is out of the question as it would involve complete redesign of the BBS software. Ross suggested that if the DF128 files could be formed into memory image format they could quite readily be passed through TEXPAC. I spent the week before the February TISHUG Sydney meeting trying to juggle the contents of a file into memory image program but without much luck. Ross volunteered to inspect the files and is currently trying to find a solution.

If Ross's venture works out, it will be only a small step before users of TEXPAC will have the availability of downloadable GIF files. I have already supplied Rolf with 72 GIF files for conversion, courtesy of my hardware systems manager from my work environment. I would encourage those members of TISHUG who own PC's and have access to GIF files to do the same. GIF Mania is by no means the only GIF viewer available. Rolf has informed me that another program by Achim Liese will soon be available through the shop. This program unlike GIF Mania will only work with an 80 column card. This hardware prerequisite should not prove to be a problem. Several members already possess the Mechatronics or Dijit cards, besides those who have the TIM board on order.

A CONCLUDING THOUGHT

At the risk of being branded a prude, or assuming a censorship role, because some of the GIF images border on voyeurism and might be found offensive by some members of our TISHUG community, I would hope that discretion will be exercised, in the selection of GIF image material offered for general download. ○

continued from page 14

to distribute the Mike Dodd programs which were being offered through JP Software, but is not responsible for previous unfilled orders.

More recently, I have read a complaint that Baker Software, which formerly advertised game programs in Micropendium, has cashed checks without shipping merchandise or responding to inquiries. However, I

wrote to them and they assured me they were still in business.

If a vendor wants to go out of business, it would take only a few minutes to scribble a postcard to John Koloen, who would certainly publish it in Micropendium. If they are operating from a post office box, they need only close the box to have all further orders returned to the sender. If they are using a home address, as many are, they can mark the letters "no longer at this address" and drop them in a mailbox.

If a programmer has written and marketed a useful applications or utility program unlike any other, thereby discouraging anyone else from writing a similar program, and no longer wishes to sell it, I do think that he has a moral, although not legal, obligation to license it to another vendor or to release it as fairware (why bother!) or public domain. I have been the TI world's most outspoken opponent of piracy, but I would find it hard to criticize anyone who copies, or allows to be copied, such a program which has been left in limbo.

In fact, if I find myself in need of a program marketed by someone who is notoriously slow and unreliable, I may just obtain a copy from someone and then mail the author his asking price, thereby saving him the cost of a disk and postage and saving myself a great deal of frustration. ○

continued from page 16

LENGTH OF SUBSTITUTE FOR <C>	AVERAGE RUN TIME (seconds)
1	7.74
2	7.72
3	7.76
4	7.90
5	8.04
6	8.12
7	8.23
8	8.34
9	8.46
10	8.56
11	8.62
12	8.71
13	8.83
14	8.94
15	9.02

As you can see, variables up to three characters in length ran in about the same time. Once the length was longer, however, each additional character in the variable name increased the run time by about one tenth of a second for 1000 executions or .1 millisecond for one.

I also ran this one:

```
10 C=0 :: FOR I=1 TO 1000 :: A=0 :: NEXT I
```

The average run time was 7.06 seconds. There is a cost when substituting variables for numbers. TANSTAAPL (there is no such thing as a free lunch)!!

DID YOU KNOW?

You can have more than one item for a VALIDATE. For example,

```
10 ACCEPT AT(5,1)VALIDATE(DIGIT,"Q") BEEP:F$
```

Will accept the ten digits and the letter Q. It does!

Enjoy. ○

GIF File Transfer

From PC to TI99/4A

by Rolf Schreiber

As mentioned in my software column this month, we have made a major breakthrough in converting GIF files from the IBM PC format to the TI99/4A format. By "we" I mean Geoff Trott and myself, with Geoff supplying all of the know how and some of the hardware. The conversion was achieved by using a direct RS232 connection between a 386SX IBM clone and my Geneve (Myarc 9640 computer) running in TI99/4A emulation mode. I could have used my TI99/4A instead of the Geneve, but it does not have 80 column capability at this stage, and I wanted to be able to view the converted pictures.

Geoff worked out the details of directly connecting the two computers, using his RS232 breakout box. He found the connection to be very simple, being basically only a straight through, pin for pin, null modem connection. We used Telco V1.3 on the Geneve and NetComm V3.0 on the PC clone to establish a software link between the two computers. We used Xmodem transfer, since we knew that this mode was ideally suited to transmitting the 128 byte blocks which make up the DIS/FIX 128 file format used by all TI99/4A GIF files.

On our first attempt we used a baud rate of 1200, with 8 bits, no parity and one stop bit as the parameters. The transfer worked very well, except that it took about 23 minutes to transfer the 111,360 byte file. The resultant TI99/4A GIF file took up 437 sectors on disk, being 640x480 pixels in size, in 256 colours.

On our next attempt we used 9600 baud, which is the highest baud rate that an unmodified TI RS232 card is capable of. The increase in speed of transfer was very noticeable, being about 3 times faster than before. It is not possible to achieve the full 9600 baud transfer rate, since the TI99/4A is much slower at processing data than a 386SX and also needs to save the information to a disk file quite frequently. Both these processes are the limiting factors with respect to the rate of transfer.

The "size" of a GIF file (as expressed by the number of bytes that the file occupies in memory) remains unchanged when it is transferred from a PC to a TI99/4A. On a PC clone, file size is always shown in bytes, while on the TI99/4A, size is usually measured by the number of sectors that the file occupies on disk.

To calculate the number of sectors that a converted GIF file will occupy on a TI99/4A formatted disk, do the following calculations on the PC version of the file:

- 1) Divide the number of bytes the file occupies (use the "dir" command in DOS) by 256.
- 2) If the result is not a whole number, then round it up to the next whole number.
- 3) Add one more for the directory entry sector.

If you need to know the size of a TI99/4A GIF file in bytes, it will be necessary to multiply the number of sectors, less one (ie, subtract the directory entry sector) by 256. For example, a 358 sector GIF file is actually only 357 sectors long, since one sector is used for the file directory entry. This means that the maximum length of this file would be 91,392 bytes (if the last sector were completely filled).

The following table shows the maximum size of a single GIF file that can be stored on the various TI99/4A disk sizes, expressed in a variety of ways.

Disk Format	Sectors	Maximum File Size	
		KBytes	Bytes
SSSD	357	89.25	91392
DSSD/SSDD	717	179.25	183552
DSDD	1437	359.25	367872

For those of us using 80 track drives, the above maximum file sizes are doubled. On a hard disk system the practical upper limit would be many times larger than the current maximum file size.

Telco writes to disk whenever it has received 64 blocks of information (of 128 bytes each). Since each disk write operation only transferred 32 sectors and took nearly 16 seconds on the Geneve, a significant reduction in time should be achievable if the file is written out to a RAMdisk, instead of a physical drive. The time to transmit 64 blocks of data and to process the information took 22 seconds. This adds up to 38 seconds for each 8192 bytes transferred, or approximately 2150 baud, far less than the selected 9600 baud rate.

The length of time required to transfer a file depends on the baud rate, the size of the file and the number of errors encountered during transmission. If we select 9600 baud and assume no errors in transmission, then the transfer time can be estimated by the following formula:

$$T = B/13,000 \text{ or}$$

$$T = N/102$$

where T = length of transfer, in minutes

B = size of PC GIF file, in bytes and

N = number of blocks to be transferred, if known

These two formulae are really equivalent, since each block is 128 bytes and 128x102 is effectively equal to 13,000.

The maximum size of GIF files currently available is 1024x768 pixels, up to 256 colours. Such a picture would occupy a maximum of 768Kb of video memory, with typical examples (that I have personally viewed) ranging about 639Kb to 736Kb in size. These files could be transferred to a TI99/4A, but a hard disk would be required to store files larger than 719Kb. Also, a TI99/4A fitted with an 80 column card would only be able to display a "window" of the whole picture. In non-interlace mode, this window displays 256x212 pixels in 256 colours, while in interlace mode the portion of the picture displayed increases to 512x424 pixels in 16 colours. However, when displaying these very large GIF files in the latter mode, the colours are frequently distorted.

After Geoff left my place, I played around with transferring a wide variety of different GIF files, and I did not have any problems with a single transfer. The largest file which I transferred was a 654,237 byte whopper, which took about 50.3 minutes, and occupied 2557 sectors on an 80 track DSDD disk.

Both colour or monochrome graphics can be transferred from a PC to a TI99/4A, as long as the file is in GIF format. Since it is possible to convert a wide variety of different PC graphics formats into GIF files, there is theoretically no limit to the amount of graphics data, from clip art to high resolution colour or monochrome artwork, that can now be readily made available in the TI99/4A format. ○

continued from page 21

that STR1 contains all three words, you can really forget the other two. Try it, 6 4 .ST3 types COWBOY, then 3 7 .ST3 types BOY, and 7 0 .ST3 types HOLYCOW. Get the idea? We did all that without special string handling words.

Forth may seem a little complex, but in XBASIC all the above would probably require ACCEPT AT, SEG\$, LEN, DISPLAY AT, string concatenation &, DATA, READ, FOR-NEXT loops and CALL CLEAR at the very least.

In the next BFORTH I will present some special string words, like GLEN, SEG\$, POS and CENTER.

CU next time; May the Forth be with U.

continued on page 23

Amateur Radio Log

A Data Base Program for the T199/4A, by Ross Mudie

1. INTRODUCTION

Amateur Radio operators who communicate with other amateurs all over the world, often need a data base to record the callsign of stations with whom they have communicated. When hearing a CQ, (a general call for others who may wish to communicate), the data base may be accessed and the date of a possible earlier contact may be quickly established. TISHUG member, Peter Schelling VK2CPS, asked if I could write such a program, which is described in this article. The program is written in TI Extended Basic with extensive links to assembly for speed. When the data is stored on a RAM disk, better speed is achieved during saving and loading of files. Others may have similar applications and some of the ideas that I present in this article and in the program may have application for other data base programming tasks.

The data items stored (with field sizes) are Callsign (6), Name (10), Locality (12), Date (8), QSL Sent (1) and QSL Received (1). (The QSL is a post card which provides written confirmation of the communication. Many Amateur Radio operators collect these cards and just keeping track of the exchange of cards can be a massive task).

The Brief Log program reads either Corcomp or MBP clocks to obtain the date for a default in the program. When the program first runs it displays the date from the clock. If a clock is not installed in the computer then the program prompts for dd/mm/yy, which may be entered from the keyboard.

After prompting with the default storage device file name (eg, DSK1 or HD, etc, which may also be changed from the keyboard), the program goes to the main prompt. Either the CHECK or NEW ENTRY mode may be used to allow the program to ask for the callsign on which the search is based. Amateur callsigns are alpha-numeric with 5 or 6 characters. In most cases they consist of two letters which indicate the country, then a numeral which indicates the state, followed by the letters which denote the individual station. In the case of my own Amateur Radio callsign VK2ZRQ, the VK denotes Australia, 2 is the state of New South Wales and ZRQ is the individual station identification.

The program uses the prefix, the state numeral and in some cases part of the suffix, for the file name. The determining factor for the number of characters in each file name is the number of callsigns which will need to be placed in each file. The physical limitations of the computer become a controlling factor on the size of any file in memory. In this program around 310 data base entries per file is the practical limit, dictated by the available space in the VDP RAM which is used as a load/ save buffer. In combination with a RAM Disk, loading and display of any callsign entered in the data base should be possible in less than 3 seconds. A large file (about 300 entries) will take approximately 15 seconds to load from a floppy disk. Search and display time once the file is loaded will be about a second.

2. FILE NAMES

To determine the file name to be used, an assembly routine named FILENA is accessed to search for callsign prefixes which were previously nominated as likely to require more than one file name per country. The data used to model this program contained 427 entries for the Netherlands (Peter's home land). There were 154 for PA0, 1 for PA1, 12 for PA2 and 260 for PA3. FILENA was provided with data to break the PA0 entries up into 3 files (PA0A to PA0H, PA0J to PA0P and PA0R to PA0Z). PA1 and PA2 were simply handled as 3 digit file names, whilst PA3 was broken into PA3A/PA3B, PA3C/PA3D and PA3E/PA3F. To allow for possible future growth of

callsign allocations from PA3G to PA3Z, a single file name was allocated for this block. Each user can customise the data in the FILENA program to suit the individual needs. If a file grows too big, additional file names can be introduced and the existing data file split up by the computer. When the number of data entries for a particular country are small, ie the file is unlikely to grow to 300 for some time, then no special entry in FILENA is needed. The file name that FILENA will allocate in this case will be the first two letters of the callsign entered in the data base.

When a partial callsign is entered, if there is enough information to provide a valid file name, then the file will be loaded. Once a file is loaded it may be viewed using the V option which shows screens of 6 entries at a time. If the partial callsign is too short then the program goes back to the main prompt.

3. THE MAIN PROMPT

The main prompt allows a single letter key press to provide the required action. The prompt string is:

[C]heck, [N]ewEntry, [U]pdate, [V]iew, [P]rint,
[S]ave, [D]ate, [R]emove, [L]og, [H]elp, [E]nd

Choice of a particular function is achieved by pressing the key corresponding to the first letter of the required function.

CHECK - Allows a callsign to be entered. The program then loads the appropriate file if required and displays the data for that callsign. If no data exists it just shows the callsign entered and the position in the file that would be allocated if entry to the data base is desired. This prevents automatic entry into the data base which occurs if the NEW ENTRY option is used when the required function is just to check if a particular station has been previously contacted. To make a data base entry from the CHECK option, just press U for UPDATE and enter the required data.

NEW ENTRY - Allows a callsign to be entered. The program then loads the appropriate file if required and displays the data for that callsign. If no data exists it shows the callsign entered and the position in the file that has been allocated. It then prompts for the data base entries of NAME, LOCATION, DATE, QSL SENT and QSL RECEIVED. If the callsign already had data entered, NEW ENTRY does not automatically accept update, it just shows the existing data. Press U for UPDATE and enter any required data changes. The action of NEW ENTRY is opposite to that of CHECK.

UPDATE - Update can be used from CHECK or NEW ENTRY modes. It allows new data to be entered from CHECK mode or existing data to be updated (modified) in either CHECK or NEW ENTRY mode. When ever any data is changed, the SAVEFLAG is set to ensure automatic saving of a modified file entry. (Automatic saving will occur before any new file is loaded or before the program ends with the END option).

After NEW ENTRY or UPDATE, the entry fields are cleared on the screen, but below the entry fields the "Last Entry was..." is shown. This is intended for when you return from an interruption and say, "Now, where was I?"

VIEW - View allows the file in memory to be inspected without modification. It Views the entries, showing the position in the data base (0 is the first number). In View mode 6 entries are shown in each screen, Fctn E (Up arrow) allows scrolling up in blocks of 6 entries, Fctn 9 (BACK) provides the ESCAPE to the main prompt and any other key pressed will scroll down the file in blocks of 6 entries.

PRINT - Print allows the file in memory to be printed on an attached printer. The default printer file name is PIO. This is in the extended basic part of the program and may be easily modified. To terminate a Print, just hold any key to escape to the main prompt.

SAVE - Save allows a file to be saved on demand. It is intended to be used part way through an entry session in one file, if you are worried about losing all your hard work to a power glitch or lock up. **SAVE** resets the **SAVEFLAG**, thus if a new file is loaded with **CHECK** or **NEW ENTRY** after **SAVE** and before any other data is changed then the current file is not saved again. This can allow appreciably faster acquisition of a new file when the data base is located on floppy disk.

DATE - Date allows the date in the computer to be manually changed. When the date entry is blank in the **NEW ENTRY** or **UPDATE** mode this date will be automatically entered. The ability to change the default date allows a block of new entries to be entered on a date after the actual contacts. During **NEW ENTRY** and **UPDATE**, the date can be individually changed for each log entry. Pressing enter on the date shown on the screen will store that date.

REMOVE - Remove allows single entries to be removed from the data base. To use **REMOVE**, first find the position number of the data to be removed. This can be done by using **CHECK** (the data base entry number is shown in brackets after the callsign) or search the file using the **VIEW** mode. After **REMOVE** has been used, **View** is automatically called to show the data base from one entry before the data removed to the 5 entries after the item removed. You can move around in **VIEW** with the UP arrow (Fctn E), most other keys for down. To exit the **View** mode inside **REMOVE**, press **BACK** (Fctn 9). The next prompt allows another number to be entered for removal, or by just pressing **ENTER** with no number, it exits to the main prompt. Use of **REMOVE** sets the **SAVEFLAG** ensuring that changed data is saved automatically before another file is loaded or the program is ended.

LOG - Log allows the disk used for data storage to be changed. Typical entries for Log are **DSK1** or **DSK2** for floppy disks or **HD** for an EPROM ROS RAM disk. The default value for the Log disk is stored in the extended basic program variable **DD\$** which is near the start of the program. It is possible to load a file from one disk drive, then change the specification and save the file on another disk drive. The current device and file name are shown whilst using the **CHECK**, **NEW ENTRY** or **UPDATE** modes.

HELP - Help provides a single screen of brief descriptions of each key operation.

END - End checks the **SAVEFLAG** and saves the data file if necessary, before the program is terminated. Both **CLEAR** (Fctn 4) and **QUIT** (Fctn =) are disabled early in the extended basic program, making the only ways out of the program the use of **END** or the power switch. There are no files held open whilst data is being viewed or updated.

The screen saver times out in approximately 4 minutes when the program is at the main prompt without any key activity, and anywhere else that the single key press utility **KEY** is used. When a time out occurs the computer will **BEEP**. When the screen saver times out from other multi-character prompts the normal extended basic time out of 11 minutes applies (without a **BEEP**). To restore operation after a screen timeout, use **Fctn S** or **Fctn D**, (left and right arrows), or **ENTER** to reset without changing anything. Almost any other key will also provide the reset but that key press may register.

4. FILE STORAGE METHOD and MEMORY USAGE

Files are stored in Memory Image format. The first two bytes of each file describe the total number of bytes in that file, including the 2 byte length descriptor. Each data base entry uses 38 bytes, thus a file containing 3 entries would be 116 bytes. (3 times 38, plus 2). The CPU RAM space used when a file is in memory commences at hex address >A040 and works up towards the extended basic program and numeric variable tables.

The program stores the data in memory image format.

This is much faster for loading and saving than using **INTERNAL** or **DISPLAY** format files. Extended basic does not provide for files to be stored in **MEMORY IMAGE (MI)** format, thus this facility is provided by a **DSRLNK** in the linked assembly portion of the program. **MI** files (which the disk manager shows as **PROGRAM** format) do not contain waste space in the sectors which can occur with **DISPLAY** or **INTERNAL** files, thus **MI** files are often more compact for the same data. The current **VI.3** software does not contain any compression in the **MI** files.

The **VDP RAM** space which is required to be used for the save **MI** function is also used by the screen, extended basic for scratch pad, pattern descriptors, colour tables, strings, the variable name table and the disk drive buffers. Careful management of the **VDP RAM** is essential to achieve a reliable program. Just before any **SAVE** or **LOAD** a garbage collection is executed from assembly to push the extended basic strings up to one end. The **Pattern descriptors**, **sprite motion table** and **colour table** are then copied over to spare space in the **High RAM**, immediately below the extended basic line **Number table**. During the **LOAD** and **SAVE** functions, the screen is intentionally blanked to prevent a corrupted screen caused by the data through the **Pattern Descriptor table**. When the use of the space is complete, the system information is returned to its place in **VDP RAM** and the extended basic carries on as before.

5. CUSTOMISING THE CALLSIGN PREFIXES FOR FILE NAMES

Callsign files are named using part of the prefix as the file name. If no special file names are specified, the program **FILENA** returns the first two characters of a callsign as the significant characters of the file name. For cases where the data to be stored exceeds 300 items, individual file names can be specified. Customisation requires a straight forward entry in the assembly source file **FILENAS** of the required data and then the source file must be reassembled to create a new object file **FILENAO**.

There are two data lists in file **FILENAS**, one for 3 character file names and the other for 4 character file names. In both cases a **DATA** value must specify how many items are in the list as shown in the following example.

```
CALLS3 DATA 4      Number of items in following list
CALL3A TEXT 'VK3'
      TEXT 'VK4'
      TEXT 'PA1'
      TEXT 'PA2'
```

If an additional 3 letter file name was to be added the data value in **CALLS3** would be changed to 5 and an extra line of **TEXT** and the required 3 characters added inside single quotes.

Four character file names are handled in a slightly different manner. The number of items in the search list are specified in **CALLS4**, similar to the 3 character file names. The **TEXT** for the 4 character file names consists of the "search" partial callsign, followed by the specified file name. Refer to the following example:

```
CALLS4 DATA 78     There are 78 names to be searched.
CALL4A TEXT 'PAOAPAOa'
      TEXT 'PAOBPAOa'
      TEXT 'PAOCPAOa'
      TEXT 'VKZZVK2z'
```

When the first four characters "PAOA" are found, the file name **PAOa** is substituted. Later in the list the text: **TEXT 'PA3BPA3a'** means that the callsigns commencing with **PA3B** will be saved in the file **PA3a**. A lower case character is used in the file name to show when the file name may depart from use of characters in the group of callsigns contained in the file. If an attempt is made, using **TI Disk Manager 2** to catalogue a disk containing file names with lower case characters, then these characters will appear corrupted since **DM2** does not have lower case defined. Use of **DM1000** will overcome this problem.

The Communicators

by Ross Mudie

There is a need to limit the number of file names per disk due to the fact that the TI DOS only allows for 127 file names per disk. Another critical factor is that the VDP RAM buffer is limited to about 310 entries, dependant on what extended basic has stored in strings at the time, which limits use of VDP RAM space for the SAVE buffer.

When 3 or 4 character file names are used, in some cases it is desirable to prevent the possible generation of an unwanted file name from the entry of a partial callsign. PA0 and PA3 are an example of this because these prefixes are both extended to 4 characters. The FILENA program rejects callsign inputs of less than 2 characters in all cases and with suitable entry can also reject 2 and 3 digit callsign entries. (When an entry is rejected, FILENA passes a null string back to the extended basic program and the length variable LE has a value of 0). In the Vn 1.3 software, the callsigns PA, PA0, PA3 and VK2 are all treated as invalid character combinations because they must be extended with additional characters before a decision on file name can be made.

The case of VK is also interesting because VK2 is handled with 4 character file names, VK3 and VK4 are 3 character file names, whilst VK0, VK1, VK5, VK6, VK7, VK8 and VK9 will all be allocated the 2 character file name VK.

The entry of invalid character combinations in FILENA is exactly the same as valid 3 character combinations. In the assembly source file FILENAS, find the part of the invalid lists required, add the combinations as necessary as TEXT in single quotes and change the DATA word at the start of the list to tell the program how many entries are in that part of the list. Resave the source file as DSKn.FILENAS, then reassemble with source file DSKn.FILENAS, object file DSKn.FILENAO, press enter at the list file name and enter R for the options.

6. WHAT HAPPENS WHEN A FILE IN MEMORY GROWS TOO BIG

Due to the variable use of VDP for string space, the amount of VDP space available for the MI save utility varies. If the extended basic program is not changed then it will be possible to store between 316 and 323 entries in a file. The extended basic program links to a SIZE routine in assembly at the main prompt and during the remove routine. Once the data is within 8 entries of full, the screen background colour will change from BLUE to RED as a warning. If the file is too big to save, then a WARNING error message is given. Entries must then be removed from the data base to reduce the file size. It is suggested to remove one entry, then try to save again, repeating until the file is small enough to save. (The removal of too many entries just makes more work when the file is split and the task of re-entry is undertaken. Saving is still possible when the screen is red).

If a callsign file is likely to become too big then the FILENA program should be modified to split the file with two file names, (as detailed in part 5 above), then the FILESPLIT program can be used to create the required new files.

If an incorrect specification is given for a SAVE device then a Warning message will be given, but the data will not be lost. Just fix up the problem with LOG or use a disk which is not full, etc, then save the file again. (Keep a spare initialised disk on hand to allow resolution of this type of problem without loss of data). The warning messages should cover most possibilities.

If the amount of data in FILENA grows too big, it is possible that the free space in the low RAM may be insufficient. This will result in a MEMORY FULL message at the line number containing the CALL LOADS, when the extended basic program is loading the assembly object files. Should this occur, some pruning of the lists in
continued on page 18

All TISHUG members are now entitled to BBS access with no additional membership fee. It is still necessary to be allocated a user number, user name and password. The user name is used as the mailing address, whilst the user number and password allow access for registered users only. This overcomes the disruptive effects of hackers who are hell bent on causing problems.

The equipment that you need to access the BBS is a computer with RS232, 32K memory expansion and a suitable modem connected to a telephone line. The user's computer requires a Terminal Emulator program. The BBS operates at a speed of 2400 bauds. The BBS modem automatically detects the data rate of the calling modem and provides data rates of 2400, 1200 or 300 bauds as necessary for both program and file download. An additional speed of 1200/75 is available for file download.

If you need to buy a modem to get on the BBS, it is a good idea to weigh up the cost of the modem against the expected cost of phone calls. If you are within the local call area of the BBS then a cheaper 300 baud modem will suffice if you are prepared to accept low speed operation. If your phone calls are time charged because of distance from the BBS, then the long term costs may justify a higher speed modem. A one hour session on the BBS at 300 bauds could be completed in about 8 minutes at the top speed of 2400 bauds or about 16 minutes at 1200 bauds.

The BBS is located at Berowra, in the Avalon Beach Outer Zone. Calls from the Central Sydney Telephone Zone, the Dural Outer Zone, part of the Gosford Zone and part of the Wisemans Ferry Zone are local calls which cost 24 cents per call, untimed. (There is a map on page 9 of the Sydney 1991 A-K White Pages telephone directory which provides additional information on the local call area). All other calls to the BBS are time charged, with the cheapest times being daily from 10pm to 8am and Saturday 6pm to Monday 8am.

What benefits are available from the BBS?

The BBS, like anything in the hobby of computing, will give a return for effort expended. Simply if you never use the BBS, it will give you nothing! The major features of the BBS are downloadable TI Memory Image programs (which can include implanted assembly, D/V 80 files and Mail. A different selection of programs and files are placed on the BBS each 2 months by the SYSTEM OPERATOR (SYSOP). Some BBS members contribute Memory Image programs and files which can be uploaded. Contributions for the TISHUG NEWS DIGEST can be sent as mail to the EDITOR, this is a great way to meet that last minute final deadline or to get your information in earlier allowing the Editor time to perform his editing function.

Some of the items on the BBS until the end of March include:

- * A series of programs for amateur radio operators which perform calculations for various antennae, inductance, capacitance, resonance, attenuators, transmission lines and DX propagation.
- * An incredible DSR scanner which may reveal more about your system than you probably knew before.
- * An early series of TI peripheral diagnostics.
- * Lists of computer user groups and BBS's Australia wide. (I never realised that there were so many!)
- * Material for assembly class members.
- * Programs which use The Missing Link software.
- * Clues for hard to play games and lots more.

continued on page 5

Vaporware, Slowware and No Ware at all by Jim Peterson, Tigercub Software, USA

When Texas Instruments was still selling the TI-99/4A, and producing new peripherals and software, they tended to be extremely secretive about what they were working on - thus effectively discouraging any third party developers who would have faced financial disaster if TI came out with the same product.

After Black Friday, all such restraints disappeared. Third party hardware, software and publications began to appear. Others were announced but never did appear, and thus the term "vaporware" was coined, although perhaps it did not originate in the TI world.

One of the foremost early examples of vaporware was the fabled Phoenix computer announced by CorComp, which never did arise from the ashes as did its equally mythological namesake.

And a more recent example was PRESS, the long-awaited program that would make the TI as good as the I-word computer.

Sometimes the vaporous mists did finally blow away to reveal a new and valuable product, such as Myarc's 9640, the Geneve - but then its promised support again became enveloped in the mist.

And so the TI world became very wary of any announcements of new products. I have been contacted a few times by hardware hackers and programmers who envision some great new product and want to know what the market might be, or just want to talk about it. The advice that I give is - do not even tell your mother about it until you are ready to ship!

Almost all new hardware products are still being announced long before they are ready to sell, although nowadays most of them do eventually come on the market. There are many reasons why they announce their product ahead of time, I am sure. Some developers want to test the market reaction before they commit time and money in the shrinking TI market. Some perhaps want to discourage competition, as Texas Instruments did. But most, I believe, just cannot wait to tell the world what they have accomplished, or expect to accomplish. I cannot blame them - they are doing some fantastic things with this long-obsolete computer.

Once the vaporware has become reality, and is actually on the market, all too often the vaporware becomes slowware - the customer sends his order, his check is cashed, and he waits - and waits - and waits!

Again, there are probably many reasons. The person is in most cases working a full-time job or going to college, and marketing his product in his spare time. Perhaps he is swamped with orders - although, after 8 years of trying to sell to the TI community, I find that unlikely!

More likely, he is being swamped with questions and complaints regarding the products he has already shipped. Some of the customers try to call collect, and those who write seldom give enough information. If it is software, some of the complaints are on the level of "it says to push 'any' key - my computer does not have an 'any' key". If it is hardware, it probably requires some technical knowledge to install and any technoklutz - like me - is bound to need some help. And, with so many independent hardware developers, compatibility problems are enormous.

Also, maybe the fellow has become aware of a serious bug that needs fixing, or has almost completed a major improvement, and is trying to find time to take care of that before filling any more orders.

I can sympathize with all of these reasons, and others. Those who are still developing hardware and

software for the TI are doing it largely as a labor of love, and their remarkable knowledge and ingenuity could probably be more profitably directed toward a computer which has an expanding rather than decreasing user base. We owe them our gratitude. HOWEVER! -

ANY VENDOR WHO CANNOT SHIP HIS PRODUCT WITHIN TWO WEEKS OWES HIS CUSTOMER A POSTCARD EXPLAINING THE REASON FOR THE DELAY AND THE ESTIMATED DATE OF SHIPMENT!

And another postcard, with offer of refund, if that date is not met! Postcards cost 19 cents, and take a minute to write - no one is too busy to do that.

The same applies to items sent for repair - if they cannot be repaired in two weeks, or within whatever period is specified in advertising, the customer deserves a postcard!

About this business of cashing checks before shipping merchandise - on a very few occasions I have waited for a check to clear the bank before shipping an unusually large order to an unknown customer. Otherwise, I ship by the next working day, except for my annual 10-day fishing trip to Minnesota. In 8 years of business, I have had two bounced checks, and both of them were made good.

If I was selling a product that cost a considerable amount and required a considerable investment in time and material to produce, I might wait for the check to clear before shipping. The bank can tell you within ten days if a check has cleared, so there is no reason to wait more than two weeks.

Among those who have currently been accused of slowness and/or lack of response to inquiries are Myarc, Bud Mils, Gary Bowser of OPA, Mike Maksimik of Crystal Software, and Pike Creek. Asgard Software was a notorious offender but Chris Bobbitt has now taken on a partner and claims to be giving prompt service.

In fact, this article was inspired by my experience with Crystal Software. I ordered Midi Master on 25 August, and my check was promptly cashed. I sent an inquiry on 29 September. I gave a friend a letter addressed to Mike, authorizing the friend to take delivery for me at the Chicago Faire on 2 November. Mike told him he had just shipped it to me. As of 15 November I have still not received it!

And finally, to the subject of no ware at all - when checks are cashed but no merchandise is received and inquiries are not answered. There have been quite a few incidents of such over the past several years, mostly involving software companies which went out of business without filling some of the orders for which they had received payment - and publishers going out of business without filling or refunding subscriptions.

The most recent examples involve JP Software, and there are many who are wondering whether the item they returned to Myarc for repair will turn out to be very slowware or no ware at all.

Now, J. Peter Hoddie is a fine young man and a programming genius whom I greatly admire. He contributed a great deal to the TI community. Later, he and Paul Chariton set up JP Software to market a few of his products, and some fine products produced by others. Not long after, he secured employment with Apple, and it is reported that he was required to sign a contract which made it impossible for him to continue selling to the TI world.

Somewhere along there, something went badly astray. There seems to be no doubt that checks were cashed, but software was never shipped. I am sure that Peter would never cheat anyone, but it is equally certain that some users did not get what they had paid for. Fortunately, Jerry Coffey has been able to contact Peter and arrange to take over distribution of at least some of his titles, including shipment of those which were purchased but never received. Beery Miller has made arrangements

continued on page 9

TI-Bits Number 14

by Jim Swedlow, CA USA

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

MAKING A DEGREE MARK IN TI WRITER

An owner in Huntsville Texas wrote me and asked if I know how to make TI Writer type a degree sign on a TI Impact Printer (it is a Epson MX80). A degree mark is not one of the standard ASCII characters. Although many newer printers can print it, the MX80 cannot.

The only way I could figure to do it was to combine TI Writer's transliterate command and the MX80's graphics ability. After a bit of experimentation, I hit on this:

```
.TL 91:27,76,7,0,48,72,72,72,48,0,0
```

This redefines the left bracket ({}). The first two characters (27,76) tell the MX80 to invoke graphics. The next two (7,0) tell it that there will be seven graphics characters. The last seven characters define the degree mark.

This is not a perfect solution as, if you right justify, the right margin will be a bit uneven. It should work, however, on most Epson and compatible printers.

SOME THOUGHTS ON WORD PROCESSORS

Of late I have occasion to use a number of word processors on other machines. I learned word processing on TI Writer and I wanted to see how the 4A stood up.

TI Writer is limited by the 4A's design. Eighty columns and a full keyboard make text management (warning: buzz phrase alert!!) much easier. Otherwise, TI Writer fares well.

Just about anything you can do with the big name packages, you can do with TI Writer. Sometimes it is a bit harder, but it can be done. TI Writer is a powerful and flexible tool. It has some abilities, like transliterate, that are superior to other word processors.

The others are slicker because they have much more memory available. They can do things with one or two key strokes that take five or ten with TI Writer - but they can be done on the 4A.

If you are writing a book, it might be worth the cost to move up. But for correspondence, writing this column and similar jobs, TI Writer can do anything you need it to do. And that is a fact.

BLACK FRIDAY PLUS EIGHT

It was over eight years ago that TI announced that they were dropping the 4A. October 28, 1983. A date that changed everything for 4A owners.

We moved from the main stream of computing to a cul-de-sac. Software and hardware became scarcer and scarcer. Retailers dropped from many to only a few.

And yet a cul-de-sac is not a bad place to live. In fact, they are preferred. Ours turned out to be pretty good. Eight years later, exciting software continues to appear. We have had quite a few graphics applications programs, with many innovative programs coming out.

There are some signs of strain. User groups report declining membership and money problems. TI owners are slowly moving to other machines (often with three letters).

The end, however, is not upon us. Our 4A still has strong support from retailers, developers, publishers, user groups and owners.

ON DISKS AND DRIVES

A while back the Disk Doctor attended one of our meetings. He had a number of interesting things to say. Since some of you missed it, here are a few of his comments.

o Do not clean your drives until you need to. Your system will tell you when it is time - you will have trouble reading disks.

o When you do clean your drive, use any brand name commercial disk drive cleaner and follow instructions.

o If this fails, you need to have your drive cleaned professionally. If you want to try yourself and you have a double sided drive, be careful with the second read/write head. It is very, very easy to bend the bracket to the point that the head must be re-alinged.

o He has tested the amount of residue left on heads with brand name disks (\$1.00 + each) and the cheepies (\$0.25 or so). He found no difference. This does not mean that they are of equal quality, only that the cheepies are not dirtier.

o He opposes flippies for single side users. His point is that when you flip the disk and it runs backwards in its cover, dirt is loosened and spun into your drive.

o His overall advise is the first rule of engineering: If it is not broke, do not fix it.

SOME MORE THOUGHTS ON BACKING UP DISKS

Over the years I have mentioned the importance of backing up your disks. Simply put, disk drives eat disks. On the first weekend of October, I was working on some letters. This was the weekend where the temperature was well over 100 degrees. I blew both my word processing disk and my data disk.

I had a backup of the word processor, but it was not configured. That night, after it cooled down a bit, it took me about half an hour to recreate a working disk. The data files were simply lost.

The moral? Keep two back ups of your program disks. One of the disk as you received it (the master) and one of your configured working disk (back up working disk). Do not forget to back up your data disks every now and then. This will save you time and aggravation next time your drive gets hungry.

ANOTHER TI-IBM CONNECTION

I have three TI's and an IBM clone. I use all of my computers. As I sit here typing, it is on the good old 48 keys that TI so kindly gave us.

I upgraded my clone to colour and bought a Magnavox RGB Monitor 80. It has two inputs, composite and RGB. On the front there are the usual picture controls and two buttons. One lets you switch from a colour monitor to a green screen. The other changes input mode from RGB to composite.

I wondered how my nice new monitor would work with my TI. I plugged my TI monitor cables into the composite inputs, leaving the clone RGB cable in place. Guess what, the Magnavox works with both the IBM and the TI. At once. I can have both running and switch from one to another with the front switch (RGB to composite).

TI WRITER's INCLUDE FILE

One of TI Writers nicer features is Include File (.IF). It has a few limitations, but it extends TI Writers capabilities.

continued on page 23

XIB tips Number 15

by Jim Swedlow, CA USA

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

RE-MAPPING THE KEYBOARD

You normally see CALL KEY(0,K,S). There are five other values for the first variable, the key unit. They remap your keyboard:

- 0 Keeps the keyboard in the same mode as the last time a CALL KEY was executed. If this is the key unit on the first CALL KEY in a program, you stay in 4A mode.
- 1 Splits the keyboard into two.
- 2 smaller boards. Good for games.
- 3 Remaps the keyboard as a 99/4 (no A). CONTROL and lower case are inactive.
- 4 Pascal mode.
- 5 Remaps to the 99/4A mode with lower case and CONTROL active.

When you use 3, regardless of the position of SHIFT and ALPHA LOCK, all alphas return as upper case. The problem is that this condition continues until you do a CALL KEY with a different key unit. Try this:

```
10 CALL KEY(3,A,B) :: PRINT : "Key Unit is 3"
20 INPUT A$ :: PRINT A$
30 CALL KEY(5,A,B) :: PRINT : "Key Unit is 5"
40 INPUT A$ :: PRINT A$ :: GOTO 10
```

With ALPHA LOCK up, try inputting lower case letters and see what happens.

A key unit of 3 is very useful to make sure that only upper case alphas are caught by the CALL KEY. If you want lower case later in an INPUT or ACCEPT, however, you must remap the keyboard with another CALL KEY.

USER FRIENDLY/USER PROOF

When programming, you want your program to help the user. You also want to keep the user from crashing your program. Remember, the user will do most any fool thing. One area of vulnerability is inputting information thru INPUT and ACCEPT statements.

Lets say you want an integer between 1 and 9. Here are a number of ways you can input that number.

INPUT F can cause two problems. First, any number will be accepted. Second, if your user inputs anything but a number, you get: "WARNING: INPUT ERROR IN nnn TRY AGAIN".

This destroys your screen, scares your user and looks bad too.

ON WARNING can help. If you add ON WARNING NEXT the warning message will be suppressed but any number will be accepted. This coding is probably the best that can be done with INPUT F:

```
10 ON WARNING NEXT
20 INPUT F :: IF F<1 OR F>9 OR F<>INT(F) THEN 20
```

Bad values of F (0, 3.1, etc) will still cause the input prompt to be repeated and mess up your screen but you will get a good value in the end.

ACCEPT AT has a number of features that will help. With SIZE and VALIDATE you can avoid some problems:

```
10 ACCEPT AT(5,10)SIZE(1)VALIDATE(DIGIT)BEEP:F
```

Your user can still goof you up two ways. Zero is acceptable input and inputting a null will do strange things. SIZE(1) means no bigger than one character but it can be smaller!

ACCEPT F\$ will help solve some more problems. Strings will be accepted. This coding is about as user proof as you can get:

```
10 DISPLAY AT(5,1):"1"
20 ACCEPT AT(5,1)SIZE(-1)VALIDATE(DIGIT)BEEP:F$ ::
IF F$="" THEN 20 ELSE F=MAX(VAL(F$),1)
```

By making the SIZE value negative, whatever is on the screen at (5.1) will be the default value if ENTER is pressed.

About the only way I have found to mess this up is to have a non-numeric sitting at (5,1). VALIDATE works ONLY on which key is pressed, it assumes that you know what is on the screen!

MEMORY SAVERS

This is the third and last article on how our TI's store stuff in memory. We have looked at the line number table (October, 1985) and encoding line contents (December, 1985). This month we will draw some conclusions.

A variable name takes only its length in memory. <A> takes one byte while <MASTER@DEVICE> takes 13 bytes and <A\$> takes two bytes.

A number, however, takes the number of characters plus two bytes. For example, the number 2.13 would take six bytes of memory and the number 1 would take three bytes.

Strings also take the number of characters in the string plus two. "He won" takes eight bytes.

Some suggestions:

--Use the shortest possible variable and sub-program names.

--If you use a given number often, assign it to a variable and use that instead.

Figure this way: A=2.13 takes eight bytes (one for A, one for = and six for 2.13). Each time you use A instead of 2.13 you save five bytes. Therefore, after two substitutions you are conserving memory.

--Look for places where you can replace numerics with variables. This line:

```
10 C=0 :: INPUT A(0) :: PRINT A(0)
```

Would take four less bytes if done this way:

```
10 C=0 :: INPUT A(C) :: PRINT A(C)
```

Be sure, however, that you only do this when the variable must be the number you intent it to be!!!!

SPEED

Longer variable names slow program execution. I ran this program with progressively longer variable names substituted for <C>:

```
10 C=0 :: FOR I=1 TO 1000 :: A=C :: NEXT I
```

I ran each three times and averaged the results. Here is what I found:

continued on page 9

TI99/4A World News

by Jim Peterson, Tigercub Software, USA

November ~ December, 1991

Since PC Transfer has long been unavailable, due to the failure of JP Software to fill orders or answer inquiries, it is welcome news that Bruce Harrison has released Smart Connect, a program to transfer files between TI and PC computers. It automatically splits large PC files into increments small enough to be loaded into TI-Writer. The program is available for \$10, which includes shipping and handling, from Harrison Software, 5705 40th Place, Hyattsville MD 20781. Also, the price of their unique Word Processor has been reduced from \$20 to \$14, which includes packaging, posting and mailing.

Jerry Coffey has reached an agreement with J. Peter Hoddie to distribute JP Software programs. He will also work with buyers who have paid for but never received products from JP Software; in these cases he requests that the purchaser send him as much information as possible, including xerox copies of cheques, statements, etc. His address is 9119 Tetterton Ave., Vienna VA 22182.

SNUG has also amended their constitution to include "PC and compatibles". This may mean that their newsletter will soon be of little interest to TI users. I have observed that when a TI group opens its doors to other computers - Sun Coast, Brevard, Broward, etc. - their newsletters soon drop any mention of the TI. Florida used to be a major center of TI activity, but only the Miami group seems to be very active these days. The Daytona group survives only as an informal get-together, and several others have been swallowed up by Big Blue.

Asgard Software has released Thumbnails, to organize, catalog and convert MacPaint pictures; Starbase Raiders, an arcade style game; and Gofer, a utility for user with Page Pro 99. The price of each is \$12.95 plus \$3 shipping and handling.

Don O'Neil's DIGI-PORT is a cable which plugs into the TI, CorComp or Myarc parallel port and allows you to play, through an external amplifier, 8-bit digitized sounds ported over from IBM, Macintosh, Amiga, Atari ST, Sound Blaster or whatever. The software which is supplied will also play the sounds through the TI-99/4A's own sound generator, but only at 5-bit accuracy. The TI with memory expansion alone can only play 10 seconds of digitized sound, but when augmented with additional memory it can play much longer, up to 10 minutes or more. Also supplied in the package are 10 disks in a choice of SS/SD containing short sounds for the TI-99/4A alone, DS/SD containing longer sounds for the 4A with 80-column card, DS/DD with still longer sounds for the MEMEX, RAMBO or Geneve, or DS/QD with even longer sounds. The price is \$39.95 US, from OPA, 432 Jarvis St., Toronto ONT, Canada M4Y-2H3. OPA also apparently offers additional sound disks.

Asgard Software has announced a much improved Version 2.0 of the Asgard Mouse, and much improved software to go with it. There are no compatibility problems with programs written for the old Mouse. Current owners can obtain the new software by returning the mouse disk with \$7.50, which includes postage. They can upgrade to the new Mouse by returning the old version, providing it is serviceable, together with the adapter cable and \$15 plus \$3 for shipping and handling. The Asgard Mouse, with software, can be purchased for \$49.95 plus \$5 for shipping and handling. Asgard has also released the Mouse Development Package, which enables programmers to add mouse support to their programs. The price is \$14.95 plus \$3 for shipping.

Asgard's Reflections has resumed publication with Volume 2 No. 4, but new subscriptions are not being accepted and only two or three more issues are planned. Chris Bobbitt promises that all subscriptions will be satisfied one way or another. Asgard Publishing has now begun publication of Asgard Brief, a shorter publication

devoted to news of upgrades and new products from Asgard Software. User groups can get a free copy by sending a postcard to Asgard Publishing, PO Box 10306, Rockville MD 20849, and copies can be purchased for 25 cents, but it is not available by subscription. The same information is available by download from Delphi and Genie. Asgard Publishing also publishes Page Pro Times for users of Page Pro. Issue 6 was released in October.

Asgard Software is retrenching somewhat, in order to give better service, and will deal primarily in software rather than hardware. Plans for the reintroduction of the Mechatronics 80-column card have been cancelled. Chris reports that since he took on Harry Brashear as a partner, orders are being filled in 3 or 4 days rather than 3 or 4 weeks, and products returned for repair are being completed in 2 to 4 weeks.

Asgard has released 4 new games (TI-Pei, Classic Checkers, Colors and Starbase Raiders), and has made arrangements to offer three Commodore games (War Zone, Living Tomb and Boxsteine), one from Notung (TI Casino) and 6 never-before-released Infocom Adventure titles. The new Asgard Software 1991-1992 Entertainment Catalog is available free from Asgard Software, PO Box 10306, Rockville MD 20849.

Disk Only Software (PO Box 244, Lorton VA 22079) will soon be offering Tony Lewis's "Interface Standard and Design Guide", a compendium of some of the most important references on TI hardware design.

The following report on the Berlin TI Faire is condensed from a report by Jim Fetzner in Reflections and a letter from Alexander Hulpke published in the Lima newsletter.

* One of the features was a new P-System for the TI-99/4A and Geneve, not requiring the P-Code card. It was rewritten from scratch, supports 80-column cards and more disk drives, and runs 3 or 4 times faster. It is still in the process of final debugging and testing.

* The Berlin User Group was selling, for about \$60, a device to install a Speech Synthesizer in the P-Box. Unlike the Rave card, this device contains the ROMs from the TEII cartridge, thereby allowing unlimited text to speech in any programming language and without the memory limitations of the disk-based Text-to-Speech.

* TI Club Leipzig was demonstrating a true video digitizer for the P-box which accepts a video input through standard ports and digitizes it in real time (on the fly!) into a My-Art/YAPP format picture suitable for display on an 80-column card. It uses very expensive new chips and therefore costs about \$350.

* The same group demonstrated modified routines for disk access which allowed direct controller programming, avoiding the TI file system overhead, thereby achieving speed comparable to a hard drive. They also discussed an AT-Bus hard disk, but had not written a DSR for it.

* Another group was demonstrating, and expects to have available soon for about \$100, a TI Hex-Bus interface which allows the TI-99/4A to be used with the many peripherals which Texas Instruments released for the CC-40 hand-held computer.

* Also displayed, and costing about \$14, was a simple chip and motherboard for the Mechatronics 80-column card which expands the number of available colours from 256 to 256 out of 256,000. Alexander Hulpke suggests that it could be adapted for the DIJIT card or the Geneve if an unused 8-byte area could be found to decode.

* Also released was XB3, a complete rewrite of TI Extended Basic which is 16k larger, offers dozens of new commands, is 100% compatible with existing XBasic programs and runs them an average of 2 to 4 times faster! However, it requires a GRAM device. It will soon be released on disk by Asgard Software, and will later be available as a module. I am not sure if this

module consists of 3 ROM banks at >6000 and will therefore run only on the Mechatronics GRAM card, not on the Geneve or Gram-Kracker or other simulators with just two banks.

* New software included a CAD program by Henrik Wedekind, described as very user friendly, for the TI with 80-column device; CRASH by Peter Muys for the Geneve, a stock market program claimed to be equal to very expensive PC programs; and an XBasic compiler for the TI, written in Basic (and not able to compile itself into assembly) which compiles everything but SUBs, DEFs and arrays, but takes hours to do so.

The Lima User Group has booked space at the Ohio University Lima Campus for the next totally free all TI/Geneve show from 4 PM Friday May 15 through 8 PM Saturday May 16 1992.

A couple of user group newsletters have mentioned that Baker Software, which formerly advertised game programs in Micropendium, has cashed checks without sending merchandise, and is not responding to inquiries. However, when I wrote to them they replied promptly and said they were still actively in business.

Texaments (53 Center St., Patchogue NY 11772) has issued a new catalog which is free on request. Their products include TI-Artist PLUS! and supporting programs including 13 Companions, 3 volumes of Starfleet Technical Drawings, Fonts Frames and Fun, Designer Labels, Guidelines, and Artoons; GIF Mania to view GIF pictures on a TI; TI Base Version 3.0; 9 games for the Geneve; TI Sort; C.S.G.D. Graphics; The Missing Link; Sound F/X; Rapid Copy; XB Detective, and Exec. They also offer the Geneve (\$419.95) as well as the Myarc Hard Disk Controller, Floppy Disk Controller and RS232 Card.

Bruce Harrison of Harrison Software has released two new programs. Scud Buster is a unique assembly game based on Patriot missiles shooting down Scuds in the Gulf War. The joystick is used to set an aiming point on the screen, and the fire button launches a missile toward that point to cross the trajectory of the Scud and meet it. There are several degrees of difficulty.

Code Breaker is a cryptogram program in assembly, therefore lightning fast. It contains 380 puzzles and has a feature where you can make your own. The standard level is difficult, the advanced level is for experts only.

Beery Miller (PO Box 752465, Memphis TN 38175-2465) has received permission from Myarc to distribute the P-System for the Geneve, and will provide the disks for \$10. It requires a DS/DD system.

DELPHI has announced the establishment of Custom Forums. Any subscriber can open a Forum for a set-up fee of \$39.95 and a monthly fee of \$15, and then has complete control over who can post and read messages on his forum. DELPHI offers up to 20 hours per month of evening/weekend access via Tymnet for \$20 per month.

For several years, Texas Instruments conducted a mail survey of TI user groups each year, and provided an address listing on request. I recently inquired whether they had made such a survey in 1991. They did not reply to the question, but sent me a copy of the listing they had distributed in 1990, which was out-of-date even then.

I have repeatedly urged user groups to register with FOG, so that they would be listed in Computer Shopper and Computer Monthly. Fewer and fewer have kept up their registration. Now I hear that FOG is charging \$25 for new registrations and \$15 for renewals. ○

continued from page 13

FILENA may be required or possible AORging of some of the data to spare space in the high RAM may be required. Always keep a master copy of the original program before modifying source files to meet individual needs. ○

Assembly Class for 1992

by Ross Mudie

A small number of members have indicated interest in continuing with the assembly class from last year. The class will be held from 10am to 1pm prior to the meeting on the first Saturday of each month, at the Ryde Infants school. The subject material for the March class was distributed at the February meeting and is also on the BBS. See the files KEYS (similar to Link-It21) and FILENAS).

The proposed subject for April is "Using DSRLNK". There will be NO assembly class in May since I have another commitment.

If members are serious about learning assembly then they must be prepared to put in some time with assembly between classes. At least 4 hours each week as a guide, and to turn up at the agreed time to start for the monthly class. (These were problems in 1991).

There have been some requests for a class using Mini-Mem for assembly. Since I do not have this module it is not practical for me to run a class especially on this subject. Once assembly in general is understood, the specifics of Mini-Mem can be understood. Regardless of which operating environment is used, (Mini-Mem, Extended Basic, E/A Basic or pure assembly), an editor assembler manual is essential. Class members should bring their editor assembler manual to each class since it is the reference manual for the subject.

Any TISHUG members who want BBS access should call me, (week day evenings 7pm to 9pm preferred or weekends) on (02) 456 2122. All TISHUG members are now entitled to BBS access at no extra membership cost. ○

For Sale

The following items are surplus to my current needs and are for sale: Phone (042) 842980

Mechatronic standalone 32K memory expansion with 96K RAMdisk and PIO printer interface. No modification to the console is necessary, as the unit just plugs into the side of the TI99/4A. Complete with power supply, manual and printer cable, \$100.

UHF modulator, needs five pin DIN plug, \$5.00.

Navarone Widget in excellent condition. Saves wear and tear on your module port, \$20.

Stand alone power supply, suitable for powering an external half height disk drive, \$30.

Slimline mono cassette tape recorder, suitable for use with the TI99/4A or for recording lectures or documentaries. Runs from batteries or mains power. Comes with leads, manual and has a remote control plug. Only \$12.

AT Multi-Function Card, with double density disk controller (Geoff Trott modification fitted) and PIO printer port, fully tested and guaranteed to work. Plugs straight into the PEB or any 1 or 2 slot expansion system. Comes with comprehensive manual and software, \$190.

Stand alone single slot expansion system (DCC, 32K and PIO not included) with power supply and half height DSDD disk drive, housed in an attractive metal case. Fits the MFC or any controller card in a clamshell. Selling for the cost of parts only, \$145.

Internal PEB modem (Peter Schubert design) with 300, 1200 and 1200/75 baud. Fully tested and comes with software and a digital phone, \$95.

TI99/4A console with 32K memory expansion built in, \$60.

Speech Synthesizer, with manual and demo software, \$30.

Page Pro 99

by Ed Johnson, USA

Introduction

Page Pro 99 has a long and varied history. The program was begun almost two years ago by Chris Bobbitt and Ed Johnson. In its original conception it was simply a modified version of TI-Writer that would allow you to generate a form on the screen with a limited number of line patterns. Originally, we decided to call the program "Form Maker 99" (you may have seen it advertised under this name). The program worked fine, but we were not satisfied.

The program was not capable of doing things that form programs for other computers could do. So Ed and Chris thought of a lot of things they both wanted to see in the program, in retrospect things which redefined the concept they originally had. Ed started from scratch in assembly and what you see before you is the result of nearly two years of development and "fine tuning". PRO PAGE 99 is a program that, to put it simply, allows you to create full page forms. With Page Pro 99, you can type in any of four different directions using a small or large text font, draw lines with line font and you can include up to 28 pictures of any size anywhere on the page. You can "import" a TI-Writer text file or "export" a page as a text file for use in other programs. You can even print out the page in different dot densities (ranging from a "rough draft" single density to reproduction quality quadruple-density).

Due to the vast changes in the program over its course of development, the name "Form Maker 99" just did not seem to do justice to its capabilities. At the recommendation of a very good friend (thanks Denny), it was decided to change the name to "Page Pro 99". We hope that you will agree that the performance of this program lives up to its new name.

Page Pro 99 will, in effect, let you do much of what a dedicated desktop publisher will do, much simpler and faster. Page Pro 99 is the only program of its type that is entirely "what-you-see-is-what-you-get". What ever text, lines, or pictures you place on the screen will appear exactly as they will on your paper. Unlike other desktop publishing programs, it is easy to create impressive looking letterheads, signs, charts, maps, graphs and even forms.

However, we are not going to call PAGE PRO 99 a true "desktop publisher". Why? Because Page Pro 99 will only let you have three different fonts on the page at once and only 28 pictures. It really is not fair to call it a desktop publisher, even though it does more in this area than most other programs that call themselves such. We simply do not want to raise everyone's expectations of what it will do. We will not object if you would like to use Page Pro 99 as a desktop publisher, or even if you would like to call it one. We simply will not do so. We use it to make more than just forms, and think that you will too!

Loading Instructions

Before doing anything else, make a backup copy of the program and use the backup copy.

Editor Assembler Module

Select Editor Assembler option #6 - Load Program File.

Press <enter> for the default name "UTIL1" (even though no filename appears. This loader will look for DSK1.UTIL1). The program will then load and run.

TI-Writer Module

Select option #3 - Utility.

Press <enter> for the default name that appears on the screen (DSK1.UTIL1).

The program will then load and run.

Extended BASIC module

With the program disk in drive #1, the program will load and run automatically after selecting Extended BASIC from the menu screen.

RAMdisks

Will load from any RAMdrives, Loading with UTIL1. Note that the program runs the best when using the files, pictures, etc. from a RAMdrive.

Program Concepts

All of the functions of Pro Page 99 are in one program. When the program firsts loads and runs, a title screen will appear in the upper 3/4 of the screen. This upper 3/4 area is the "page window" to a much larger area, which is the entire page.

What you see is 12 lines by 31 columns of the total 66 lines by 60 columns of the page. As you move around the page, the window will adjust itself to display the part of the page you are on. The only way to view the whole page is to print it out.

The 66 line by 60 column page of Page Pro 99 is almost the same size as a standard typewritten page (a standard text page is 66 by 80). When printed out, it will fill an 8 1/2 by 11 inch page from perforation to perforation, with a small margin on the left and right. Proper paper alignment is very important.

The bottom 1/4 of the screen is used for displaying the "status" line, prompts and for getting the keyboard input for file names etc. The status line keeps you up to date on the current cursor position on the page, the direction the cursor will move as you type and whether or not the picture display is "ON or OFF" (more on this later!).

Getting Started

Once you understand how the program represents a page, you can actually get started in creating your own layout for a page. Once you are familiar with the cursor controls and how they work, you can easily begin typing in your text.

Note that as you type, when reaching the end of a line the cursor automatically wraps down to the next line. However, any word you may have been typing at the righthand side of the page will be split in two. Page Pro 99 does not support "word wrap"; in other words, it will not automatically keep words connected between lines. If this editing limitation is cumbersome to you Page Pro 99 will let you use any text editor that saves files in Display Variable 80 format to generate the bulk of your text and then you can "IMPORT" it into Pro Page 99 to put the "finishing touches" to it.

Cursor control Keys

Arrow Keys - <FCTN-E>, <FCTN-S>, <FCTN-D>, <FCTN-X>

These keys will simply move around your page without disturbing whatever you have on a page.

Page Down - <FCTN-4>

Move the cursor down the page 12 lines at a time until it reaches the last line (line 66).

Page Up - <FCTN-6>

Moves the cursor up the page 12 lines at a time until it reaches the first line (line 1).

Window left and right - <FCTN-5>

Moves the cursor forward or back 30 columns to the window opposite of the one you are currently in.

Cursor Auto-movement direction - <CTRL-E>, <CTRL-S>, <CTRL-D>, <CTRL-X>

These keys control which direction the cursor will move after pressing a key when editing your text or drawing lines. The arrow in the lower right-hand corner of the screen indicates the direction you will move as you type. Page Pro 99 will always treat the cursor movement as a left to right in terms of where the cursor will end up after pressing <enter> key or when it wraps to the next line. That is, if you are typing "backwards" (right to left), the cursor will move up one line after <enter> is pressed or if you "wrap" left of column #1. Similarly, if you are typing down the page, you will end up the top of the page one column to the left after pressing <enter> or if you wrap off the bottom of the page. On the surface this may seem trivial or too confusing but it will make entering text with a sideways or upside-down fonts much easier (anyone thinking "greeting cards" yet? We were!). Spend some time getting used to typing in different directions. You will soon be a real "pro" getting around with Page Pro 99!

Line Commands:

Line drawing mode is entered by pressing <CTRL-8>. A menu of the available line parts that can be drawn appears in the status window with the corresponding character which when pressed will cause that line part to be drawn.

Edit Keys:

- <FCTN-E> - move cursor up
- <FCTN-X> - move cursor down
- <FCTN-S> - move cursor left
- <FCTN-D> - move cursor right
- <FCTN-1> - delete character
- <FCTN-2> - insert character
- <FCTN-3> - delete line
- <FCTN-4> - down one screen
- <FCTN-5> - next screen across
- <FCTN-6> - up one screen
- <FCTN-8> - insert line

Editing Keys

Delete Character, <FCTN-1>, is only active when typing small text or line characters. Characters to the right of the cursor move one position to the left. Large text and pictures are not shifted, nor are any characters to the right of large text or a picture affected.

Page Pro 99 High Resolution Picture Print Utility v2.0 Copyright 1989, by Ed Johnson

This rather simple utility was written to fill a need that many Page Pro 99 users have: to get higher resolution output than is currently available with Page Pro 99.

I think you will find this utility very useful in its current form. It will read any size Page Pro 99 format picture file and print it out in 4 different combinations of density and width. The program is simple to use, is reasonably fast and gives excellent results.

In addition, you can selectively overstrike the output. If you choose not to overstrike, the output is that same as from Page Pro. With overstrike, the paper is fed 1/216" (or 1/180" for 24 pin printers) and the same line is printed again. With overstrike output, printing takes twice as long, but the quality is much improved.

The program will handle any size Page Pro picture, up to and including a full page. You can clip a page as a picture and then use this utility to print it out in higher resolution than is currently available with Page Pro itself.

I have provided no options to squeeze the picture vertically (yet!), so I recommend that you leave a blank line at the top and bottom if you want to print a whole page that you have clipped.

After the program is run, you are greeted with a title screen that describes the program. Press any key to advance to the next screen. This screen has the fairware notice and will prompt you for several choices for printer output.

First, select the printer mode; 8 pin or 24 pin. 24 pin printers will handle both options. The exciting part of the new 24 pin mode is that it supports full 24 pin output, not just emulating an 8 pin mode! The output is also set up so that a full 66 lines will fit on an 11" page. What I had to do to accomplish this was to convert the normal 8 pin graphics data (one byte) into 3 bytes of graphic data for the 24 pin mode. In doing this conversion, I expanded (or reduced - depends on your point of view) the data to use 20 of the 24 pins and set the line spacing accordingly. No matter your point of view, the output fits on the page and the quality is excellent!

The next three selections are for determining density, width, and overstrike.

You can get 8 different "resolutions" of output with this version. The 4 combinations of density and size below can each be printed with or without overstrike. Those with 24 pin printers actually have 16 combinations since there is also a vertical size difference between selecting 8 and 24 pin mode.

Single selections)	Density/Normal	Width	(Use Single/Normal)
Single selections)	Density/Double	Width	(Use Single/Wide)
Double selections)	Density/Half	Width	(Use Double/Normal)
Double selections)	Density/Normal	Width	(Use Double/Wide)

To quit the program, enter "0" for printer selection, or a null string when prompted for the picture file name.

Page Pro Quick Reference Guide

Line Characters:

Line drawing mode is entered by pressing <CTRL-8>. A menu of the available line parts that can be drawn appears in the status window with the corresponding character which when pressed will cause that line part to be drawn.

Edit Keys:

- Delete character - <FCTN-1>
- Delete line - <FCTN-3>
- Insert character - <FCTN-2>
- Insert line - <FCTN-8>
- Move cursor up - <FCTN-E>
- Move cursor down - <FCTN-X>
- Move cursor left - <FCTN-S>
- Move cursor right - <FCTN-D>
- Next screen across - <FCTN-5>
- Scroll down one screen - <FCTN-4>
- Scroll up one screen - <FCTN-6>

Command Keys:

- Back out of menu - <FCTN-9>
- BREAK printing - <FCTN-4>
- Change cursor movement to up - <CTRL-E>
- Change cursor movement to down - <CTRL-X>
- Change cursor movement to left - <CTRL-S>
- Change cursor movement to right - <CTRL-D>
- Erase all pictures - <CTRL-U>
- Erase page - <CTRL-R>
- Erase individual picture - <CTRL-K>
- Export/Import text - <CTRL-F>
- Large fonts on/off - <CTRL-9>
- Line mode on/off - <CTRL-8>
- Load fonts/lines - <CTRL-A>
- Load page - <CTRL-F>

continued on page 5

Beginning Forth - part 13

by Earl Raguse, UGOC, CA USA

FILES AND STRING HANDLING

The Useful Forth Words of BFORTH #4 offered some words like GSTR\$, GNUMB, PP, CSTR\$, etc which were specifically for string handling, but no explanation or examples, this will be rectified in this article. GNUMB is for getting numbers as a string from the keyboard, then converting them to numbers to put on the stack after you had been prompted by a string to enter them. CSTR\$ will be used in some future column, and will permit comparing strings for "greater than", "equal to", or "less than", thus permitting us to test and change the order of file entries, in other words, SORTING. This article will use some of the above words so please refer to BFORTH #4, which includes most of my UFW's.

DISPLAY VARIABLE FILES

However to make any real use of this capability one should be able to create and read files in Forth, which we have not covered yet. But files are not of much value unless they have some data in them. I have previously discussed, EDITing a screen as a way of entering file data for an application. Some time I will also show how to rewrite screens under program control. One reason for establishing file handling with Forth is so one can manipulate the data in a file, like sorting it for example. It would seem logical to use DV80 (Display Variable 80) format so that files could be more easily exchanged with other users and languages.

There are several Forth programs for converting a Forth screen to a DV80 file and vice versa. One of these was written by J W Vincent. It is shown on Screens #50-#52. The screen line numbers do not move because they are not a part of the actual screen. They are the result of a template provided by EDITOR. Notice that screens displayed by 64SUPPORT do not have line numbers either. All the screens you see in these columns have been transferred by this program to DV80 files and edited with TI Writer Replace String to restore the line numbers. That in itself is a neat trick, but since this is not a TIW article, we will not get into it.

One does not do that, however, if one intends to read the file with Forth which will automatically restore the line numbers when converted back to screen format. A common reason for putting Forth screens into DV80 format is to up or download them from one Forth user to another via a modem.

Next time I will provide examples of what it takes to permit one to Open, Close, Read and /or Write to DV80 files in memory or on disk. After that I hope to make some constructive use of them.

Because the Forth EDITOR is far more convenient to use than entering text in the immediate mode, especially if you have installed the auto-repeat cursor, than is the limited editing capability of Forth in the immediate or execution mode; it seems appropriate to write new files on screens and then transfer them to DV80 files on disk for subsequent reading by Forth for whatever purpose. Of course, one could use Basic or TI Writer to create the DV80 files on disk also.

BACK TO STRINGS

Before getting too carried away with handling of files I want to cover a little more about manipulation of the strings they contain, numbers are not prohibited, but they generally will be treated as strings.

Forth does not have number variables and string variables as BASIC and other high level languages do. It is up to the user to know what he/she puts into a variable. You may name a variable so that you know what you originally intended however. If you should try to multiply two strings, even if they are numerical, you

are doomed to failure unless you ask Forth to convert them to numbers before treating them like numbers. The UFW GNUMB gets a number from the keyboard, using QUERY, as a string and uses INTERPRET to convert the string to a number on the stack which can then be used in arithmetic operations.

Forth characterizes a string as an Address and a Count. Screen #37 defines some variables STR1 thru STR3. A variable is a word defined by VARIABLE and is simply an address in memory with two bytes reserved. The word ALLOT then insures that a specified additional space is allotted in memory for the variable. They are then initialized with BLANKS and a string by INIT. The Forth way to enter a string at an address is !" (store quote). If you do not fill the variable with blanks first, you can expect garbage. Notice that I initialized only the first 20 bytes of STR1 thru STR3 because that was more than I was actually using. I could have made the ALLOT smaller but I was thinking of defining general purpose strings. INIT uses (!") to load strings HOLY, COW, and BOY into these variables. As you will subsequently discover, a variable can contain more than one string.

The word TT, abbreviation for -TRAILING TYPE, for the "I hate to type crowd", is used to type any string given an address and a count on the stack; the -TRAILING eliminates trailing blanks and spaces. The word .ST (print string) uses INIT to initialize the strings, then 3 CMOVE moves 3 characters from address STR2 to address STR1 offset by 4 byte positions. TT then types the first 20 characters at address STR1 which now contains the string "HOLYCOW" which has been concatenated from two strings. Not quite as simple as BASIC but not too tough. The word .ST1 then demonstrates this a little more by moving 3 bytes of STR3 to an offset of 7 bytes after start of STR1. The printed result is "HOLYCOWBOY".

The word .ST2 shows that after STR1 contains three strings you can extract and print the middle one (or any one for that matter).

Not very awesome, but try entering INIT between executing both the words .ST and .ST1. I hope you see that you would get "HOLY BOY". Spacing of words is adjusted by controlling the offset from

```
DSK #37
0 ( STRING PRACTICE EGR 1 12 87 )
1 FG IT : IT ; CLS
2 0 VARIABLE STR1 30 ALLOT
3 0 VARIABLE STR2 30 ALLOT
4 0 VARIABLE STR3 30 ALLOT
5 : LOC CLS 10 12 AT ;
6 : INIT STR1 DUP 20 BLANKS !" HOLY"
7   STR2 DUP 20 BLANKS !" COW"
8   STR3 DUP 20 BLANKS !" BOY" ;
9 : TT -TRAILING TYPE ;
10 : .ST INIT STR2 STR1 4 + 3 CMOVE STR1 20 TT ;
11 : .ST1 STR3 STR1 7 + 3 CMOVE STR1 20 TT ;
12 : .ST2 STR1 4 + 3 TYP ;
13 : .PRMPT ." INPUT #CHAR TO PR" CR GNUMB ." OFFSET " CR GNUMB ;
14 : .ST3 .PRMPT STR1 + SWAP LOC TT ;
15 CR CR ." TRY <.ST>,<.ST1>,<.ST2>,<.ST3> "
```

```
SCR #50
0 ( TO/FROM VAR 80 FILES 1/3 JWVincent 8/15/84)
1 BASE->R DECIMAL 16 SYSTEM 0 0 GOTOXY FORGET IT : IT ;
2 ." FORTH Screens to/from V80 files " CR
3 ." By J.W.Vincent " CR
4 ." These screens will read or write TI "
5 ." variable 80 files to or from TI-FORTH "
6 ." screens. If DISK_HI equals DISK_SIZE "
7 ." one drive will be used. When using one "
8 ." drive, begin with FORTH loaded, you "
9 ." will be prompted when to load each "
10 ." disk. If multiple drives are used put "
11 ." FORTH in #1 and the V80 files disk in "
12 ." #2. V80 files read/written must/will "
13 ." be named SCRnxxx where xxx is the "
14 ." screen number. When reading a V80 file "
15 ." EOF will cause a disk error, after " ->
```

the beginning address of STR1.

The word .ST3 is a little more universal, it expects, on the stack, a count to type and an offset from address STR1. If you execute words .ST and .ST1 so

continued on page 10

Decoding EPROM files

File Handling Techniques part 3

by Ben Takach

It is appropriate to interrupt our program design task and make a short excursion into the strange world of number systems. It will be a very short trip, so do not get bored!

We count and think in the decimal system. The PC's, PLC's and other digital electronic devices live in the binary system. An easy human readable system is needed to make the necessary communication between man and beast somewhat less confusing than rows upon rows of noughts and ones.

The bridge is the BCD code and the hexadecimal number system. Now why did we have to go to a hexadecimal system to manipulate data in a binary format? You may well ask. Elementary my friends! The base of the hexadecimal system is 16, 2 to the power of 4 is also 16. The highest value single digit in the hex system is F, which is the highest value one can reach in the binary system using 4 binary digits (bits) ie, 1111. In other words, we can use the hexadecimal system as a convenient short hand to express 4 bits by a single hex digit. The BCD code on the other hand is a "look up" table and a convenient means of simplifying the interface between the real world and the microprocessor environment.

Generally the same conventions are used in computer or PLC programs as in the decimal system, ie, the place value increases from right to left.

The names associated with these confusing digits and codes in the computer jargon are:

1 word = 2 bytes = 4 nibbles = 16 bits (16 noughts and ones) 1 byte = 2 nibbles = 8 bits

The rightmost digit is called LSB = Least Significant Bit;

The leftmost digit is called MSB = Most Significant Bit.

(LSB may also be called the right most digit) (MSB may also be called the left most digit)

2	B	4	F	Hex word (example)
3	2	1	0	Digit number (decimal or hexadecimal)
13 14 13 12	11 10 09 08	07 06 05 04	03 02 01 00	Bit number, 0 to 15
0 0 1 0 1 0 1 1 1 0 1 0 1 1 1 1	Content in binary			
MSB				LSB

Counting in HEX always start with 0!

BCD (BINARY CODED DECIMAL) is easily converted to decimal or hexadecimal, because each four bits of a binary number is numerically equivalent to one digit of a hexadecimal number. For example the BCD number 0101 (111 010) (111) converted to hex will be: 0101=5, 1111=7 thus the expression is 5F5F.

The decimal equivalent is:

$$\begin{aligned}
 5 * 16^3 &= 20480 && \text{Where } 5_{16} = 5_{10} \text{ and} \\
 + 15 * 16^2 &= 3840 && F_{16} = 15_{10} \\
 + 5 * 16^1 &= 80 \\
 + 15 * 16^0 &= 15 \\
 \hline
 5F5F_{16} &= 24415_{10}
 \end{aligned}$$

BCD is also easily converted to decimal, because likewise each four bits of the binary number will be numerically equivalent to one decimal digit. For example the BCD number 01010010101001 converted to decimal will be:

0101 = 5 and 1001 = 9, thus the decimal number is 5959

Note, this is not the same as 01010010101001 converted to hexadecimal! If the expression 5959 is a hexadecimal number then its decimal equivalent would be:

$$\begin{aligned}
 5 * 16^3 &= 20480 \\
 + 9 * 16^2 &= 2304 \\
 + 5 * 16^1 &= 80 \\
 + 9 * 16^0 &= 9 \\
 \hline
 5959_{16} &= 22862_{10}
 \end{aligned}$$

Decimal digit	Hex digit	Binary code	Notes
0	0	0000	
1	1	0001	
2	2	0010	
3	3	0011	
4	4	0100	
5	5	0101	
6	6	0110	
7	7	0111	
8	8	1000	
9	9	1001	
10	A	1010	These binary codes not allowed for BCD numbers
11	B	1011	
12	C	1100	Any 4 bit combination numerically greater than 9 cannot be used
13	D	1101	
14	E	1110	
15	F	1111	

Binary should not be confused with BCD, BCD is a code, binary is a number system. The relationship may be better understood by the study of the decimal conversion table.

Decimal Conversion Table

Decimal	BCD	Hex	Binary
0	0000 0000	00	0000 0000
1	0000 0001	01	0000 0001
2	0000 0010	02	0000 0010
3	0000 0011	03	0000 0011
4	0000 0100	04	0000 0100
5	0000 0101	05	0000 0101
6	0000 0110	06	0000 0110
7	0000 0111	07	0000 0111
8	0000 1000	08	0000 1000
9	0000 1001	09	0000 1001
10	0001 0000	0A	0001 0000
11	0001 0001	0B	0001 0001
12	0001 0010	0C	0001 0010
13	0001 0011	0D	0001 0011
14	0001 0100	0E	0001 0100
15	0001 0101	0F	0001 0101
16	0001 0110	10	0001 0110
17	0001 0111	11	0001 0111
18	0001 1000	12	0001 1000
19	0001 1001	13	0001 1001
20	0010 0000	14	0010 0000
21	0010 0001	15	0010 0001
22	0010 0010	16	0010 0010
23	0010 0011	17	0010 0011
24	0010 0100	18	0010 0100
25	0010 0101	19	0010 0101
26	0010 0110	1A	0010 0110
27	0010 0111	1B	0010 0111
28	0010 1000	1C	0010 1000
29	0010 1001	1D	0010 1001
30	0011 0000	1E	0011 0000
31	0011 0001	1F	0011 0001
32	0011 0010	20	0011 0010

Regional Group Reports

Meeting Summary For March

Banana Coast	08/03/92	Sawtell
Central Coast	14/03/92	Saratoga
Glebe	12/03/92	Glebe
Hunter Valley	14/03/92	Woolaroo
Illawarra	16/03/92	Keiraville
Liverpool	15/03/92	
Northern Suburbs	26/03/92	
Sutherland	20/03/92	Jannali

BANANA COAST Regional Group (Coffee Harbour Environs)

We never miss meeting at Kerri Harrison's residence 13 Scarba St. Coffee Harbour, 2 pm second Sunday of the month. Visitors are most welcome. Contact Kerri 32 3136, Kevin 51 2649, Rex 51 2485 or John 54 1451.

CENTRAL COAST Regional Group

Regular meetings are normally held on the second Saturday of each month, 6.30pm at the home of John Goulton, 34 Nimosa Ave., Saratoga, (043) 69 3990. Contact Russell Welham (043392 4000).

GLEBE Regional Group

Regular meetings are normally on the Thursday evening following the first Saturday of the month, at 8pm at 101 Arundel St. Glebe. Contact Mike Giverny, (02) 692 8167.

HUNTER VALLEY Regional Group

All welcome. Please contact Geoff Phillips on (049) 428 176 for details and the exact date for future meetings.

ILLAWARRA Regional Group

Regular meetings are normally held on the second Monday of each month after the TISHUG Sydney meeting, except January, at 7.30pm, at the home of Geoff & Heather Trell, 20 Robsons Road, Keiraville. A variety of activities accompany our meetings, including Word Processing, Spreadsheets and hardware repairs. Contact Lou Amadio on (042) 28 4966 for more information.

LIVERPOOL Regional Group

Regular meeting date is the Friday following the TISHUG Sydney meeting at 7.30 pm. Contact Larry Saunders (02) 6447377 (home) or (02) 7598441 (work) for more information.

NORTHERN SUBURBS Regional Group

Regular meetings are held on the fourth Thursday of the month. If you want any information please ring Dennis Norman on (02)452 3920, or Dick Warburton on (02) 918 6132.

Come and join in our fun. Dick Warburton.

SUTHERLAND REGIONAL REPORT

Regular meetings are held on the third Friday of each month at the home of Peter Young, 51 Jannali Avenue, Jannali at 7.30pm.
Peter young
Regional Co-ordinator

TISHUG in Sydney

Monthly meetings start promptly at 2pm (except for full day tutorials) on the first Saturday of the month that is not part of a long weekend. They are held at the RYDE INFANTS SCHOOL, Tucker Street (Post Office end), Ryde. Regular items include news from the directors, the publications library, the shop, and demonstrations of monthly software.

MARCH MEETING - 7TH MARCH

Any new software and hardware that has arrived from abroad will be reviewed as well as another possible look at the TIM card and what it will do. There should be games set up for anybody interested. Come along for a relaxing time of fellowship to take in any new ideas in the TI world.

The cut-off dates for submitting articles to the Editor for the TND via the BBS or otherwise are:

April	15 March
May	12 April

These dates are all Sundays and there is no guarantee that they will make the magazine unless they are uploaded by 0500pm, at the latest. Long articles should be to hand before the above dates to ensure there is time to edit them.

continued from page 10

```

001 FILE TRANSFER to /dev V80 2/3 (Merrisa)
002 which the FORTH files should be loaded
003 and the FLISH command executed.
004 The word target is
005 1 02 WRITE-V80 (output dev)
006 2 1 READ-V80 (in2 and scr)
007 00 CLOAD STAT B4 CLOAD (output)
008 VARIABLE OF 0 VARIABLE PF 1 VARIABLE BUFF 76 1217
009 PARS 0 10 + RDR 0144 FILE V80 V80 SET-048 V80 50 PSC-LEN
010 P-D' DEAN.SCHWAB
011 DRIVE-NO DISK HL 0 DSK SIZE 0 16 0 ELSE 1 THEN IF 1
012 V80-DEK DF 0 IF ELSE 01 LOAD V80 DSK1 CR KEY DROP THEN
013 FTH-DEK DF 0 IF ELSE 01 LOAD FORTH DSK1 CR KEY DROP THEN
014 FILE-NAME DRYTP-NO PARS 0 DF 0 49 0 OVER 25 1 V80V OVER 100
015 AND 48 1 ROT 25 1 V80V 10 /MOD 48 1 PARS 0 30 1 V80V 48 1 PARS
016 0 31 1 V80V V80-DEK

```

```

001 FILE TRANSFER to /dev V80 3/3 (Innocent)
002 C-LEN DUP 12 + 44 0 10 1 OVER OVER - CR 01 + IF 1 03 + IF
003 DROP THEN ELSE LEAVE THEN LEAF SWAP DROP 04 SWAP -
004 SWITS-16 16 2 DO DUP DSK1 04 ONKEY 1 LEAF V80 01 + LOOP DROP
005 ROTV80 CASE 1 05 1 0500 1 OF SWAP 1 1000 1 OF SWAP ROT 1
006 1000 4 OF SWAP ROT 22 ROT 10 SWAP 4 0000 00000 0 10
007 WRITE-16 LOOP
008 SCF-16 OVER OVER 4 1 + IF DUP 1 + OVER 1 ELSE 0 THEN IF 1
009 1 ROT NOT DO 1 BLOCK SWAP 1 + LOOP
010 WRITE-V80 10 SWAP DUP 1 050-30 10 FTH-NAME DROP OPN BEGIN
011 V80-V80 PF 0 WHILE FTH-DEK 1 + SCF-16 V80-DEK REPEAT ELSE
012 READ 4 0216 OVER 1 + ROT DO DUP 1024 01480 16 0 DO 16 DUP 04
013 1 + IF DROP 04 THEN 1 IF ELSE OVER 1 11000 01 SWAP 2 1 THEN
014 OVER DSK1 SWAP ROT ONKEY 04 + LOOP 4 1 LOOP DROP FTH-DEK FLISH
015 READ-V80 FTH-NAME OPN BEGIN DUP READ 4 1 + V80-DEK AGAIN
016 0200-010000 1->11000

```

continued from page 15

TI writer cannot work on large files. No books in one file here. As you reach the size limit, the time it takes to load and save files increases markedly. Include File to the rescue.

Suppose you have written two chapters of your next book. You named your files CHAPTER1 and CHAPTER2 (very original). At the end of Chapter 1 (the very last line), add this:

```
.IF DSK1.CHAPTER2
```

Name CHAPTER1 for the formatter and it will print both chapters. All the formatting commands you set for Chapter 1 will be used when Chapter 2 is printed, so you do not have to restate the margins and such.

Oh, you finish Chapter 3. No problem. At the end of Chapter 1, add another line:

```
.IF DSK1.CHAPTER3
```

You cannot do this at the end of Chapter 2, as you can not chain these commands. Also note that you must specify the drive number (DSK1 in this case).

I prefer to make a master file (called CHAPTER0) with all of the .IF commands:

```
.IF DSK1.CHAPTER1
.IF DSK1.CHAPTER2
.IF DSK1.CHAPTER3
```

Before (not after) your .IF lines, put in your format, header and footer instructions. Now you have all of your format commands in one place that is easy to find and edit.