

NEWS DIGEST

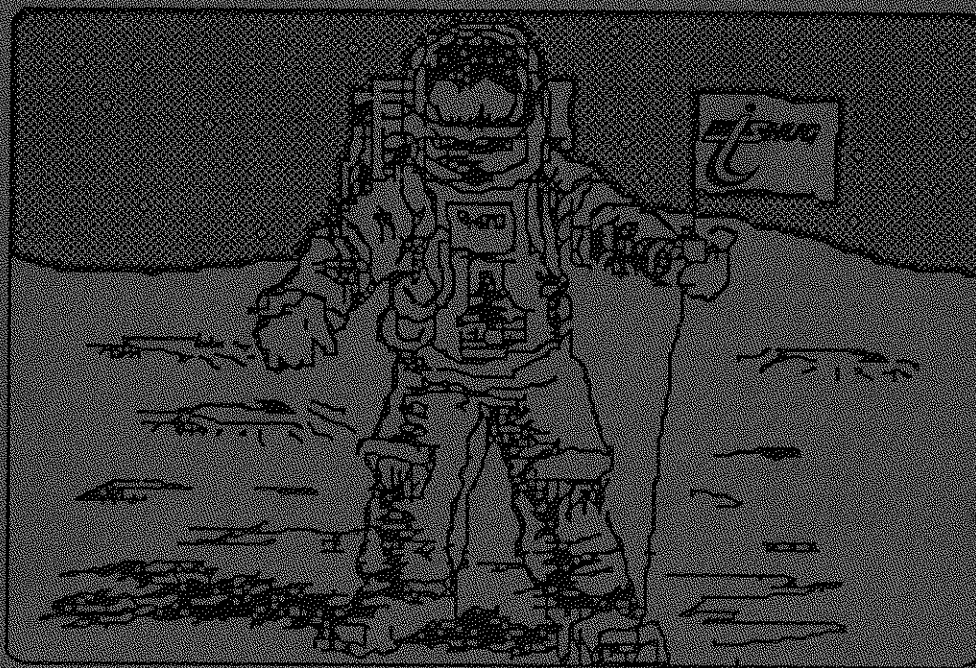
Focusing on the TI99/4A Home Computer

Volume 11, Number 1

January/February, 1992

Registered by Australia Post - Publication No. NBH5933

The TI99/4A and TISHUG



Reaching New Heights!

Sydney, New South Wales, Australia

\$3

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

TiSHUG News Digest

February 1992

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 Trott (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) 456 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 Trott (042) 29 6629

TI-Fair 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

Mawarra

Geoff Trott (042) 29 6629

Liverpool

Larry Saunders (02) 644 7377

Northern Suburbs

Dennis Norman (02) 452 3920

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 February Meeting will start at 2.00 pm on 1st of February at Ryde Infant School, Tucker Street, Ryde.

Printed by

The University of Wollongong
Printery Services

Index

Title	Description	Author	Page No.
Assembly class for 1992	Software hints	Mudie, Ross	2
Author index 1991	General interest		14
Beginning Forth #12	Software hints	Haguse, Earl	29
Control of CSI	Software hints	Hart, Ed	12
Editor's comment	General interest	Relyea, Bob	1
Extended BASIC tips #14	Software hints	Swedlow, Jim	24
Fractal Graphics with TML	Software hints	Shaw, Stephen	21
GIFviewer and hard disk	Software hints	Trott, Geoff	7
Graph paper with TI-writer	Software hints		3
Hints, tips and answers	Software hints	Shaw, Stephen	9
Hollywood HiLinux part 2	Games review	Scorpio	13
Link-It #21	Software hints	Mudie, Ross	25
Newsletter update	General interest	Relyea, Bob	30
Number base converter	Software hints	Relyea, Bob	30
PEBox extension cable	Hardware review	Takach, Ben	2
Printer graphics with TML	Software hints	Shaw, Stephen	18
Printing line numbers with TIW	Software hints	Relyea, Bob	8
Publication index for TI-base	Software review	Relyea, Bob	23
Regional group reports	General interest		31
Screen dump routine	Software hints	Hoddie, J. Peter	26
Secretary's notebook	Club news	Phillips, Terry	2
Sorting, part 3	Software hints	Brubaker, Ron	19
Subject index 1991	General interest		16
Tech time	Disk interface	Trott, Geoff	5
TI-bits #13	Software hints	Swedlow, Jim	11
TI99/4A world news	General interest	Peterson, Jim	6
TiSHUG shop report	Club news	Harrison, Percy	3
TiSHUG software column	Club software	Schreiber, Rolf	4
Treasurer's report	Club news	Trott, Geoff	8
Younger set	Programs	Maker, Vincent	10

TIM 80 column cards

We are still waiting for the TIM cards ordered last year to arrive. They are being assembled and tested by Gary Bowser and his father in Canada, five at a time, so it may take a while for the original large order to be completed. Until delivery of the original order, Gary has extended the offer on the price of \$A165 so if you wish to take advantage of this, contact Percy.

Editor's Comment

by Bob Relyea

We're off- a new year! I hope that this year is as profitable and fun as last year was. The TIM's are on order and I am looking forward to getting my system upgraded to 80 columns. That new accelerator chip sounds great, as well. I managed to get a Wang monitor this past year, a lot of nice software and Geoff managed to get some EPROMS burned for me for my Mini-PE Hardisk with the Pusselweb gear on it- magic! My system is fast getting to where I want it.

From the editing point of view, I could use some volunteers from time to time to type up some articles from overseas newsletters. Any volunteers? See me at one of the meetings. Have a great year computing the TI way.

Secretary's Notebook

by Terry Phillips

Welcome back to another year of TI99/4A activities. I hope the New Year is going well for all members and that you can all make it to at least some of the meetings that will be held this year. The full range of activities that will be covered this year is yet to be finalised but if you have any particular themes you would like to see on the meeting agendas have a talk with one of the directors.

As you all know the last meeting we had was back in December (sure seems a long time ago) and it was our AGM. It was one of the quickest insofar as the formalities go that I can remember. There was an average turn-up of about 40 members present and once the meeting was over all enjoyed a great BBQ and party fare, particularly a great cake that was included in the goodies brought along by Russell.

Since my last column two new members have joined us and it is a big welcome to:

Al Lawrence from Warners Bay - Al was a former member some years back and has been involved with the now defunct Hunter Valley Group. Al brought along his system and gave a great demo of 80 column format at the December meeting.

Phillip Williams from Buttaba also up the Hunter Valley.

Jens Buche, a member from Victoria, has written to say that reluctantly he will not be renewing his membership owing to the fact that he has to concentrate on other business interests. His basic system is up for sale and it includes a lot of books, tapes and some modules. Jens is asking \$200 plus delivery. Anyone interested can get it touch with him through me.

Russell Wehham's son, Warren, is off to Japan for 12 months as an exchange student in February. From all at TISHUG good luck Warren and we all hope you enjoy yourself.

All members are reminded that from the 1st January membership fees have increased to \$35 per year. The increase has been necessary due to rising costs particularly associated with production and mailing of the Newsdigest. To offset this, the current Bulletin Board access fee of \$5 has been abolished and all members can now access the BBS, provided they have the necessary hardware, eg a modem. Just see Ross Mudie to get a user number and a password.

In the past couple of months a number of Interstate and Overseas newsdigests have been received and there are some good articles in them. Of particular mention is Spirit of 99 (Central Ohio). This is now the best produced one we get from overseas so be sure you check it out in the publications library. TI99/4A stalwart, Jim Peterson always has an interesting article or two in it. If you are not aware, your club also still obtains copies of Micropendium on a monthly basis and these are also in the library for you to borrow. Remember there is no fee for publications library borrowings.

That's all for this month. See you at the February meeting. ○

Assembly Class for 1992

by Ross Mudie

I am available to conduct an assembly class again in 1992, if members require the class. Dick Warburton has advised that our meeting place will be open on the morning of the first Saturday of each month. I propose to hold the class from 10am to 1pm, once per month. This will allow time for a lunch break before the monthly meeting.

continued on page 8

PEBox Extension

Test driving the Tex-Comp extension cable
by Ben Takach

Tex Comp is not only the best known outlet for the TI-99 hardware and software, but it is also the longest standing and most reliable. Jerry Price started Tex-Comp exactly 10 years ago. Northridge is a long way from downtown L.A., however a visit is well worth the trouble. One can always pick up some hard-to-get items, some of which are not even listed in the 60 page catalogue.

The heavy, flat cable with its bulky interface adaptor is, without a doubt, the weakest link of the TI-99 system. The card edge connector was not designed for unsupported external duty. It is easily dislodged - generally 30 seconds before an intended file save operation - followed mostly by a string of colourful language while staring at the rigid display of the last half hours' creation on the screen.

There are many solutions to eliminate this problem. One of these is Tex-Comp's offer of an extension cable for US\$19.95. The cable is a well made 2'(approx.600mm) long 44 way flat cable terminated by male and female insulation displacement type edge connectors. The edge connectors are high quality components and the strips are gold plated.

The flat cable is a combination of flat ribbon and twisted pair configuration to minimise crosstalk. 50mm of flat section is followed by 145mm of twisted pair section throughout the length of the cable. A soft transparent plastic film is fused to the finished cable. It is very flexible and very professional. Its trade name is Spectra-strip.

I have connected it to several systems for demonstration purposes without any apparent adverse effects.

Alas, the long term experience on my own system has been less satisfactory. There seems to be no adverse effect when the system is used with the plug-in modules. It performs OK with TI-Writer, Extended Basic and a number of game modules. Funnel-Writer was the first to indicate instability. Frequent high pitched whistles from the speaker made me turn the volume control to minimum. One could listen to the disk drives during save and load operations through the monitor, not unlike the characteristic sounds during cassette load operation.

This in itself did not bother me unduly, however the frequent rigor mortis during word processing did! Loading Myarc's hard and floppy disk control program MDM became more difficult. It often did not load at all, or stopped half way leaving me with an unattractive purple screen. As a matter of fact, Myarc's disk controller software is a "fine weather" program at the best of times, thus at first I did not suspect the extension cable.

Fastterm behaved in the most unusual fashion. It would load normally, accept the log-on file, and when ready to go to modem it would fill the screen with an endless stream of random characters, change the screen colour, or the foreground colour. This was the last straw. I am currently back to the drawing board to devise some alternative method to anchor the bulky TI interface to the console.

To sum up, the extension cable is a useful and convenient device to make the console more flexible for a standard system using mainly TI cards in the PE box. Highly expanded systems may develop some undesirable side effects when the extension cable is installed. On the other hand, I have not experimented with the location of the extension cable. It may well be, that my problems were to some extent due to the proximity of the extension lead to the hard and external floppy disk drive harnesses, as well as to the data cables of my two printers.

continued on page 4

TISHUG Shop with Percy Harrison

Welcome to 1992. I hope that you all had a great christmas with lots of terrific goodies and that this year will bring happiness and prosperity to each one of you.

As usual, I have been kept quite busy since our December meeting with orders for both software and hardware, so much so that we are down to our last PE Box and system so if any of our members who only have the basic console wishes to expand their system then they would be well advised to get their order in early as the PE Box and its ancilliary equipment is becoming as scarce as hens teeth. At our last meeting your directors decided to waver the fee for membership to the Bulletin Board so now all members can now have access to our bulletin board at no extra cost providing that they have a Modem and obtain a password from Ross Mudie. At the moment I have two only 300 baud Modem Cards, at \$80.00 each, that slot into the PE Box, ideal for those in the metropolitan area who cannot afford the luxury of the more expensive 1200 or 2400 baud units.

The list of Commercial Software programs available from the shop is rapidly growing. However, some are in very short supply as only one or two copies are received in a batch from America. If you want a program that is not in stock please let me know and we will order it in for you. Please note that all Freeware programs released by the club since its inception are still available and with the promised transfer of the Master Disks up to the shop, orders for back copies of our software disks will be handled much more rapidly.

PRICE LIST.

5.25 in. DSDD Disks (Boxes of ten)	\$6
5.25 in. HD Disks (Boxes of ten)	\$10
3.5 in. DSDD Disks (Boxes of ten)	\$10
5.25 in. DSDD Half Height Drive (New)	\$65
12 Volt AC Transformer	\$4
13 Volt Arlec Transformer	\$12
8.5, 17 Volt Transformer	\$25
60 VA Transformer	\$20
MFC Printed Circuit Board	\$30
MFC Kit (Disk Controller)	\$103
Music Kit with PCB	\$65
32K Memory PC Board	\$7
Eprom Ram PC Board	\$45
Eprom Ramdisk Basic Kit	\$35
Funnelweb Eprom Set (3 Eproms)	\$36
TI Artist Eprom Set (2 Eproms)	\$24
32K Static Ram IC (62256)	\$10
8K Static Ram IC (6264LP)	\$5
Exchange Console	\$30
ROS Version 8.14	\$12
Peripheral Expansion Box	\$150
TI Disk Control Card	\$60
TI 32K Memory Card	\$40
Multifunction Card	\$220
RS232/PIO Card	\$100
Modem PE Card (300 Bd)	\$80
PE Ramdisk (256K)	\$200
Printer (Serial)	\$120

NOTE: ROS 8.14 must be purchased with & first Eprom Set.

COMMERCIAL SOFTWARE

Artoons SSSD	\$12
BABA Brewery Beer Labels	\$10
Character Set & Graphic Design Cataloguer....	\$6
Character Set & Graphic Design I	\$12
Character Set & Graphic Design II	\$10
Character Set & Graphic Design III	\$14
Disk Utilities (Memorial Edition)-DSDD	\$11
Disk Utilities (Memorial Edition)-SSSD	\$12
Display Master	\$15

Genial Traveler (SSSD)	\$6
Legends (2 Disk Set)	\$30
McPaint (5 Disk Set)-DSDD	\$10
McPaint (10 Disk Set)-DSDD	\$20
Microdex I (SSSD)	\$16
Microdex II (SSSD)	\$11
Nuts and Bolts #1 (DSDD)	\$6
Nuts and Bolts #1 (SSSD)	\$7
Page Pro 99 version 1.6	\$28
Page Pro Utilities	\$17
Page Pro Applications #1	\$2
Page Pro Line Fonts	\$9
Page Pro Medical Clipart-DSDD	\$10
Page Pro Medical Clipart-DSDD	\$13
Page Pro Templates Vol1-SSSD	\$8
Page Pro Templates Vol3-SSSD	\$8
Picasso Publisher Version 2.0	\$14
Picasso Publisher Support Disks	\$6
Picasso Applications Disk	\$2
Pix Pro	\$22
Rockrunner (SSSD)	\$15
Screen Preview	\$20
Smart Connect	\$15
Spell It! (DSDD version)	\$24
Spell It! (SSSD version)	\$27
Star Trek (Calender)	\$14
The Missing Link (TML)	\$28
The Missing Link Companion Disk	\$2
The Ring Companion	\$12
TI Artist Plus	\$25
TI Sort SSSD	\$15
Tris Module	\$25
Typewriter Module	\$25

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.

Graph Paper with TI-Writer

Author Unknown, OH USA

[This article was retyped from the October, 1991 issue of the Spirit of 99 newsletter.]

Need graph paper in a hurry? Here's a program to do it.

```

100 ! GRAPH
110 E$=CHR$(27)
120 A$=RPT$(CHR$(128),228)
130 B$=RPT$(CHR$(255)&SEG$(A$,1,6),8)
140 B$=RPT$(B$&CHR$(255),4)
150 A$=E$&"K"&CHR$(228)&CHR$(0)&A$
160 B$=E$&"K"&CHR$(228)&CHR$(0)&B$
170 OPEN #1:"PIO.CR"
180 FOR I=1 TO 11
190 PRINT #1:E$;"@";E$;"3";CHR$(24)
200 FOR J=1 TO 8
210 PRINT #1:B$;B$;CHR$(10)
220 NEXT J
230 PRINT #1:A$;A$;E$;"3";CHR$(2)
240 NEXT I
250 PRINT #1:RPT$(CHR$(13)&CHR(10),9)
260 PRINT #1:E$;"@"
270 CLOSE #1
    
```

TISHUG Software

Column by Rolf Schreiber

I am handing over a large part of the software library to Percy Harrison. This will speed things up when Percy receives mail order requests for library software. I will continue to concentrate on expanding our library holdings by procuring any software currently not in our library.

Software Releases for December 1991

DISK A465, Lute music from Harrison Software.
DISK A466, the disk-of-the-month from CONNI, described on page 5 of the December TND.

DISK A467, The Singing TI. This disk has been put together by Irwin Hott, a well known blind TI99/4A User in the USA. A speech synthesizer is required to run these programs. The only time I have ever heard my TI99/4A sing to me was when I ran the programs on this disk!

TCC-4, the fourth disk in Jim Peterson's Tigercub Collection. Crammed full of useful programs and utilities.

TC-1131, a collection of printer utilities for the Star Gemini dot matrix printer, from Tigercub Software.

Software Releases for January/February 1992

DISK A354 is Microdex 99 from Bill Gaskill. This program is used with TI-Base.

DISK A464 is called Il Pastor Fido, music by Vivaldi, from Harrison Software.

TCC-5, the fifth disk in Jim Peterson's Tigercub Collection. Crammed full of useful programs and utilities.

Commercial Software

No news yet of the TI-Artist Plus and TI-Base software ordered from Insoebot last year. Also we have not yet received the items still outstanding from the last lot of software ordered from Asgard Software.

We have a new word processor from Harrison Software, for those people who cannot come to grips with Funnelweb. The package comes on three SSSD disks and only costs \$10.00. The documentation files take up two of the disks, and print out to about 50 pages of text.

About the only good news concerning software has been from Notung Software (Ken Gilliland). A package arrived from Ken shortly after our last meeting. Since we are getting a much better software deal from Ken than from Asgard, we will be able to reduce our prices on Notung software from now on. The following items should be available at the next meeting:

Fonts & Borders I	\$8.00
Fonts & Borders II	\$8.00
Fonts & Borders III	\$10.00
Son of the Disk of Dinosaurs	\$12.00
TI-Casino (SSSD)	\$16.00
FilmLib V3.0 (TI-Base template)	\$8.00
Word Processor (Harrison Software)	\$10.00

continued from page 2

Finally, one has to admit that the TI design team did build a remarkable computer, which had less teething problems than many others released since. They had good reason to specify the heavily shielded interface cable. No doubt many tests were conducted with different cables and the one chosen was not by accident.

continued from page 7

to allow repeating if the key is held down. Finally, when a return is made from a picture the program requires the disk to be reread for files although the information is still in memory, which is a pain if there are a large number of files on the disk. I hope to be able to solve all these little annoyances as well as the one destroying sector 1 on the disk.

continued from page 25

END	MOV B @BEEPOF,@SOUND	Ensure sound is turned off
	MOV R0,@>837C	Clear the status byte
	LWPI >83E0	Reload the GPL Work Space
	MOV @SAVRTN,R11	Get the return address
	B *R11	Return to calling program
END		

continued from page 22

```

110 CALL INIT
120 CALL LOAD(16368,79,70,70,32,32,32,36,252)
130 CALL LOAD(16376,79,78,32,32,32,36,244)
140 CALL LOAD(8194,37,4,63,240)
150 CALL LOAD(9460,2,12,0,45,29,0,4,91,2,12,0,45,30,0,4,
91,203,78)

```

Once the program is run, control of the remote is accomplished by the command CALL LINK("OFF") to turn off the command and CALL LINK("ON") to turn on the control. I would suggest trying this first while watching the cassette spindle, with the cassette on, to see how it operates. Once you get a feel for it, you can write a program in Extended Basic to perform the timing.

continued from page 28

```

** VDP write to register
*
VWTRN MOV *R13,R1      Get register number and value
        MOV B @1(R13),@VDPWA Write out value
        ORI R1,>8000      Set for register write
        MOV B R1,@VDPWA   Write out register number
        RTWP              Return to calling program
*
** Set up to write to VDP
*
WVDPWA LI R1,>4000
        JMP WVDPAD
*
** Set up to read VDP
*
WVDPRA CLR R1
*
** Write VDP address
*
WVDPAD MOV *R13,R2      Get VDP address
        MOV B @R2LB,@VDPWA Write low byte of address
        SOC R1,R2        Properly adjust VDP write bit
        MOV B R2,@VDPWA Write high byte of address
        MOV @2(R13),R1   Get CPU RAM address
        MOV @4(R13),R2   Get byte count
        RT               Return to calling routine
END

```

For Sale

The following items are surplus to my needs. The items and the accompanying prices are listed below:

**** GREAT WAY TO EXPAND YOUR SYSTEM! ****

* 1 x RS232 Board & Disk Controller Card (Mini System) professionally installed into an attractive metal case which is approximately the size of a console (and less than 5cm high!) so the console easily sits on top and is connected by a short cable to the IO port. The metal case has a 3 1/2" drive fitted and facing the front for easy access. There is a auxiliary cable connected to the disk drive for optional operation of a second drive at a future time, if required. \$195

* 1 x Custom (black & silver) console in excellent condition which looks nice atop the metal case. PRICE- \$45

* Cartridges- Video Chess, Mind Challengers and Parsec (all with manuals). \$10 or all three- \$25

* Games Books, TI Manuals & joysticks.
PRICE- \$10, the lot
(or free with purchase of Mini-system)

* A box of cassettes (15, with games & programs on most of them). PRICE- \$15

* A box of 3 1/2" disks (10 disks) with software on some.
PRICE- \$8.00
(or free with purchase of Mini System)

Price for the lot- \$250.00
Minisystem & Console only- \$220.00
Ring Bob on (046) 571-253

Techo Time

with Geoff Trott

Disk Managers and Interlace

I published two articles on interlace on disks last year and the effects of interlace on the time of operations on disks. There were also some comparisons made between various disk manager programs. Well, I received a letter from Pierre Garouche which contained some interesting information about the differences in disk managers. Here is most of that letter (edited a bit).

Dear Geoff

Thanks for your work in editing my last letter for the August issue of the TND. Now I send to TISHUG the programs for direct access to the drive and chip in the Myarc Floppy Disk Controller and I expect to send to you also information and programs for fast cassette transfer. It is not fast just because it is written in assembler but because it leaves out the redundant parts of the original TI code which is in the console at address >1346, after the XML routines.

I have received a letter from Rolf Schreiber. Rolf says that you and he have Geneves. I am very interested, for my son gave me a Geneve and begged me to learn how to write assembler programs for it. Behind his generosity he hopes that I will write programs for him to do high resolution graphics to enable him to drive very fast etching/coating equipment. So on my table there are two computers: a TI99/4A and a Geneve. The Geneve is made of 3 parts: a PEbox with 2 half height (5 1/4") drives, Myarc FDC card with 40 track EPROM and the Geneve card; Fujitsu keyboard with many keys; and a Radio Shack monochrome video display.

I have refused a hard disk, as for me computing must be silent! Only RAMdisks will be used when Bud Mills and the Oasis Pensive Abacutors of Canada reach compatibility with the Geneve. My son has some difficulties on that point.

I used the Geneve assembler of J Paul Chariton version 1.03. When I first tried to write some assembler programs I was not able to assemble more than 5 or 6 instructions in one program! At this time I was using DOS version 1.14. I was only able to write a longer program by cutting it into small pieces and use the COPY directive to link them all at assembly time. This enabled me to assemble longer programs only to quickly find another limit where errors appeared. I told my son that his father is going mad and that he will go fishing to regain his sanity. Some time later he brings me DOS version 0.97H, a beta test version, and all works well. This points to the fact that J Paul Chariton's GenASM version 1.03 runs well with DOS version 0.97H beta test and does not match DOS version 1.14. I am interested to know if you or anyone among TISHUG members have tried to assemble with the Geneve (in Geneve mode, not emulating TI99/4A). What are the DOS version and the author and version of the assembler program used? I would like to have anyone's opinions

I have been interested by the information you have given in the TND of April about recording on floppy disk with interlace and in the June issue about the validate time according to the disk manager and interlace. In 1989 I did some experiments about interlace and time on reading of files. In the June issue in the second column, I see in the table that gives validation times, a strange value for DD05 and DSKU. Was it a Geoff test to see if anyone actually reads the articles or an exact value? If it is only a test value, do not read what I write after this point. If it is not a test, you may read the next lines where I try to explain what I understand from the times I read in your Techo Time in the June issue.

It took me a long time to work out why the value was so high and different from all the other values in the table. After thinking about it for a few days, I suspected it might be the time to deal with the RAMdisks. For me this was easy to verify for I have 4 (four) Horizon RAMdisks in my PEbox; I have already told you that a hard disk is not a friend of mine. When I use DSKUM or DSKU to initialise a floppy disk, I see all the light emitting diodes flash with DSKU or DSKUM and only one with Myarc DM or DM1000 (the McGovern DM1000 v 3.5 modified version for the Myarc FDC). I must explain that the first of my RAMdisks is set at CRU address >1000, the second is set at CRU address >1200, the third is set at CRU address >1600 and the last is set at CRU address >1700 with the RS232 at CRU address >1300. So I conclude that DSKU/DSKUM scans all the CRU addresses from >1200 to >1000 via >2000 at each sector verification and that Myarc DM or DM1000 scans only from >1000 to >1100 for each sector verification.

At this point I decided to do some tests. This is easy, for I have little programs that can clear or restore the header of each RAMdisk (turn to zero or load with >AA81 the address >4000 on the 4 RAMdisks; >AA81 is the value for ROS 8.14). I have made tests with different disk managers; an old version of Myarc DM when hard disks (for home computers) are not yet born, DSKUM v4.2, DSKU v4.2 and DM1000 v3.5 from Funnelweb Farm.

1/ With 4 RAMdisks with >4000 loaded with >AA81

----interlace 5, 18 sectors on each track, double sided, (DSDD) floppy disk 1440 sectors.
-Myarc DM format 35 seconds verify 85 seconds
-DM1000 format 36 seconds verify 76 seconds
-DSKUM format 40 seconds verify 420 seconds and validate option 420 seconds!
-DSKU validate option 415 seconds

----interlace 6, 18 sectors on each track, double sided, (DSDD) floppy disk 1440 sectors.
-DSKUM validate option 113 seconds.
-DSKU validate option 114 seconds.

----interlace 10, 18 sectors on each track, double sided, (DSDD) floppy disk 1440 sectors.
-DSKUM validate option 163 seconds.
-DSKU validate option 162 seconds.

2/ With 4 RAMdisks with >4000 loaded with >0000

-DSKUM validate time: 65 seconds.
-DSKU validate time: 67 seconds.

Some considerations about ROS 8.14. In this version the power-up time is longer than in the previous version. From version 04 to version 7.3 on power up, only one thing happens; the program whose name was the first on the page you configured with CFG (the name frequently used is MENU) is loaded. With version 8.14, the power-up string at >4004 links to a routine and after its execution it links to MENU (or to the program indicated when you configure the RAMdisk). If you know what that first routine at power-up does, I shall be glad to find out.

When power-up is turned off with ROS 8.14, it does not display MENU but does run the power-up routine. That is why I have routines that clear/restore the header when I want a short time for power-up, so that I do not have to extract my RAMdisk out of the PEB. But at the present time, if I knew what that first routine at power-up on the link string does, I may only clear and restore >4004 and have a no longer time of power-up as I had with the preceding versions of ROS.

And to end this letter with a trick.

By software you may change the name MENU (keeping the name length 4) and so have any program start on reset of the console. I use this possibility to have Funnelweb directly on reset and in Funnelweb on the Editor Assembler menu I may run a program that turns the first name into MENU and so return from Funnelweb OS to Horizon OS. You may do a demonstration at a meeting. It is funny to have two OS on our old TI99/4A with

instant loading when the console is switched on. (End of letter)

As you can see, Pierre has found a cause for Disk Utilities to be so much slower when there are RAMdisks in the system. Reading and writing to disks is such a time critical thing that any delay which causes the disk to rotate one more time will cause slow operation. This may be why Disk Utilities is so much slower than the others in all its operations, even when there are no RAMdisks present. However, I would use Disk Utilities in preference to DM1000 as it seems to give fewer errors.

PEBox cards

I have had quite a few cards over the years with a similar problem which I would like to bring to your attention. The problem is that chips attached to the MEMEN signal and the CRUIN signal are no longer working. This is usually a result of the card not being inserted into the socket correctly or a result of the card being knocked while in the PEBox with the power on. The former can be avoided with care and by making sure the power is off when removing and inserting cards. The second cause is more of a problem, particularly for cards which stick out the back of the PEBox. The cause of the problem relates to the position of these signals on the connector.

One end of the connector has the first pair of pins for the +15 volts, the next pair of pins are for the -15 volts with the third pair of pins used for the MEMEN and CRUIN signals. This means that if the board is displaced by one pin spacing in the direction of pushing into the PEBox, +15 volts goes to the -12 volt regulator (destroys it if it is a 3 terminal one), -15 volts goes to those chips connected to the MEMEN signal (74LS244) and those attached to the CRUIN signal (TMS9902, TMS9901). If the card is a fully populated AT disk controller card, it can be quite expensive. Because of this, the sockets used in PEBox replacements like the 2-way adapter, should have very solid stops to make sure the cards cannot be mis-aligned.

TIM board

The TIM boards have not yet arrived, but the one ordered by the club has arrived and been installed in a console along with the SOB board. Both of these boards have more ICs on them than the ones they replace on the motherboard and so draw more current. I noticed that running the setup with a MiniPE disk controller system, also drawing current from the console, that the 5 volt regulator was getting quite hot. I have not measured the difference in current but it may be as much as 100 milliamps extra. This was with a totally open system, not enclosed in a case at all. I gained the feeling that if you are planning to run a system with that configuration, as I am going to do, that an external power supply for the disk controller would be almost mandatory. This would need to supply just over 100 milliamperes regulated to 5 volts. There is provision for a 3 terminal regulator on the disk controller board.

Single Sided 3.5 inch disk drives

The single sided 3.5 inch disk drives sold cheaply by the club some time ago have caused a few problems. First of all they have a row of jumper pins labelled DS 0 to 3 for the device selection and a jumper at the end labelled SW. This last one puts +5 volts on the pull up resistor pack, which seems to have a value of 3.1k ohms. On my test setup, with two 5 1/4 drives with terminator already there, I had to remove this SW jumper. With just two 3.5 drives, they seemed to work with or without the SW jumper in. You may need to experiment with this jumper. The other effect I noticed, was that under certain circumstances the drive seemed to lose track of its stepping motor and was unable to find track zero. The only way to reset it was to turn off the power to the drive, so you might think of putting in a switch to do just that without having to turn off the power to everything.

MiniPE System EPROM RAMdisk

I have installed 4 EPROMs on a MiniPE RAMdisk successfully. The particular RAMdisk has 8 32K RAM memory chips and the 4 27C512 EPROMs giving 512K bytes of memory. It only required the addition of one 74LS08 IC plus sockets and wiring. The worst part is adding the sockets, or in fact soldering anything to the top of the memory chips, as there is virtually no room between these ICs. I removed two banks of memory ICs to give enough room to do the other two in place. I am planning to add 4 EPROMs to my RAMdisk which already has 16 static RAM ICs. This will require adding another 74LS138 at least and perhaps some more logic.

T199/4A World News

by Jim Peterson, Tigercub Software, USA

AUGUST, 1990

The Spring 1990 catalog of All Electronics Corp. lists many brand new original TI parts at remarkably low prices, especially on quantity purchases - TI keyboards for \$3.50, R/F modulators for \$5, heat sinks at 3 for \$1, etc. The address is P.O. Box 567, Van Nuys CA 91408, phone 800-826-5432.

Computer Buyer's Guide is now called Vulcan's Computer Monthly. It contains zillions of ads for computer products, and a monthly TI column by Barry Traver - which was missing in August, probably because he did not get it to them on time. It is still not available on very many news stands, but the annual subscription price of \$12 (delivery in US!) is a steal. However, I pity the mail carrier - if he ever had to deliver that magazine plus Computer Shopper and the MicroCenter catalog on the same day, he would surely have a hernia. I have suggested to Vulcan's that they buy a computer for their bookkeeping department - they double-billed me for my subscription, and then for my ad!

DIJIT is no longer taking orders for their AVPC 80-column card, but will continue to support those they have sold.

Another XBasic programming environment, called Multi-Mode XB, is available from Disk Only Software. It was written by Jean Marleau and, like the others, consists of CALL LINKs to assembly. However, it offers some features not found in any other. For a complete review, see the article by Barry Traver in Vulcan's Computer Monthly for September 1990.

Paul Scheidemantle is publishing the P&A Express, which appears to be a short newsletter with news about new graphics utilities. For a free copy, send a SASE to him at 2762 Lovington, Troy MI 48083.

Corcomp has been taken over by International Diversified Technologies Inc., 2211 E. Winston Rd, Suite G, Anaheim CA 92806, phone (714) 635-1815). They will continue to manufacture and repair Corcomp products. For repairs, call (714) 965-4450 to get a return material authorization. This is a tape, so have your information ready.

continued from page 29

```
SCR #36
0 ( COMP. CODED COSINE - COS : BY RON DAVID / E RAGUSE )
1 HEX
2 CODE COSINE
3
4 0202 , 010E , 0204 , 005A , C1D9 , 8087 , 1501 , 1002 ,
5 61C2 , 1001 , A1C4 , 0647 , 045F ,
6
7 DECIMAL
8
9
10
11
12
13
14
15
```

C U next time; May the FORTH be with U.

GIFviewer and Hard Disk

by Geoff Trott

At the last meeting of the Illawarra Regional Group we had a look at the TIM board installed on a console (with the SOB board also). One of the good things that any 80 column card gives is improved pictures and Rolf had brought along the GIFVIEWER program by Achim Liese and a few disks of GIF pictures. These all were great, except that some were a bit squashed (thin) but they did take some time to load. We tried putting the files on a RAMdisk to see if file transfers slowed it down but this only caused a small decrease in the display time. We also had a Mechatronics 80 column card system (from Jim Ellis) which we used and compared to the TIM system. The TIM system seemed to have brighter colours with more red in particular. We then thought of trying my system which has a Mechatronics 80 column card, Expansion memory on the 16 bit bus (for high speed) and a Myarc Hard and Floppy Disk Controller with a hard disk and 80 track floppy disk. The program ran on this system up to the point where disk accesses started and then bombed out. This meant that we could not run that experiment and rather annoyed me, as here were all these pictures which I could not look at. It was also another example of the HFDC having a bug when using floppy disks. It is getting to the point that I am starting to look at the DSR to see if I can find some of these bugs and fix them.

I digress a little. After the meeting I decided I would have a look at this program to find the problem. The program comes in 3 files. On disassembling these files I found that the first one was the main program and was stored at >A000 to >AEF0. The second one contains subroutines usually provided by the Editor Assembler Cartridge, along with data and text messages and used locations >2900 to >2D9A. The third file contains other subroutines called from the main program and is stored in memory locations >359A to >3BD2. As well as this, the space between the two files in low memory is used for buffers for disk data and for assembling the picture information while the space at the top of low memory (>3BD2 to >4000) is used to save part of VDP memory. The program does not have any instructions and in the process of disassembly and inspection I found out how it operates which may be useful to all, so I will start by outlining the operating instructions.

The program loads as a memory image program and puts up a title screen (you must have an 80 column card to see it) stopping with the cursor just after the DSK on a line with "DSK ." in the middle. At this point you can enter any digit between 1 and 9 or quit (FCTN[=]). If a digit is entered, sector 1 of that disk is read and the sectors listed there are read and the names of all files of type Display Fixed 128 are stored in memory. If no files of this type are found and an error message appears and the program returns to wait for a digit or quit. If files of the correct type are found, the name of the first file in the list, along with the disk number, appears after the DSK on the screen. The program is now waiting for:
Enter - accept the file shown and process it;
X - change file to next one in list;
E - change file to previous one;
BACK (FCTN[9]) - return back to ask for a new disk; and
QUIT (FCTN[=]) - quit to title screen.

When a file is accepted, the first sector of that file is read and the first two bytes are compared with "GI". GIF files start off with "GIF" and if that is found then the information about the data is output (horizontal and vertical resolution and number of colours). If it is not a GIF file, it is assumed to be a Myart file and the information about that is displayed. At this point the program waits for input which can differ for the two types of file. The common input is:
Enter - accept the display data and mode and process it;
and
BACK (FCTN[9]) - return back to ask for a new disk.

For a Myart picture:
Space - toggle the display mode;
QUIT (FCTN[=]) - quit to title screen.

For a GIF picture it is possible to move the picture around the screen using the following inputs:
E - move the picture up one position;
X - move the picture down one position;
S - move the picture left one position; and
D - move the picture right one position.

The limits of movement are 512 increments in each direction and once set these remain until changed. Once the picture is displayed, the program waits for input from the keyboard which can be:
Enter - return back to first screen and ask for a new disk;
BACK (FCTN[9]) - as for Enter; and
QUIT (FCTN[=]) - quit to title screen.

So that is how the program is supposed to work and what I found was the following. When saving the VDP memory to expansion memory, the program tries to store too much in the available space. This does not worry the TI99/4A as there are only DSR routines above >4000 but it would probably cause problems for the Geneve. The problem with the HFDC relates to the file accesses to read the data. The sector accesses work fine to get the file names of the picture files. To read the data from the file selected, the program opens the file using input relative mode. The reading is then done one record at a time. The program then has to use the record number entry in the PAB (bytes 6 and 7 of the PAB, see Editor Assembler manual page 294) to access each record. The program does this by first setting this entry to -1 on open and then incrementing it on each entry to the read subroutine to read each record sequentially. Unfortunately, the HFDC looks at this entry, even on open for input and gives an error return. The program does not test for errors as it has only filenames that it has already found so this means that it bombs out on the read after the open as the file has not been opened. I thought that if I changed the mode from relative to sequential that that entry should be ignored, but no, the error is still there on open. I had to make that entry 0 on open to get no error. Then, even on sequential access, it seemed to need the record number entry to work properly which made things very difficult. Somehow I needed to have that entry to be 0 for the open and reading the first record while incrementing for reading each successive sector. I made some changes to do this without changing any addresses of entry points and now it works reading from floppy disks. However there is still some problem as it tries to write to sector 1 of the selected disk. This destroys the file access table on the floppy so the disk must have a write protect tab on to stop this. I have spent some time reconstructing file access tables on disks!

The HFDC should not look at the record number entry for sequential access and it does not seem sensible for it to look at it when opening a file for input of any sort. No doubt this is an enhancement by the writer of the DSR which then makes it non standard as far as the TI99/4A system is concerned. I have decided to have a look at the DSR when I get some time and have disassembled it as a start. There is still the problem of writing into already used sectors when there are fractured files on a floppy disk.

There are several annoyances with the program that I will have a go at. The first is that the program displays some pictures distorted. The display options are 256 horizontal, 212 vertical with up to 256 colours or 512 horizontal with up to 16 colours and 424 vertical using interlace. The program looks at the horizontal and vertical independantly so that either horizontal or vertical can change on their own. In order to keep the relative shapes correct, either both should change or neither should change. Another change is for moving the picture around the screen. To move 512 positions requires pressing a key 512 times. I will change that

continued on page 4

Printing Line Numbers with TI-Writer, by Bob Relyca

Those of you who use TI-Writer & Funnelweb would have noticed that the line numbers that appear on the very left part of the screen before your text can be toggled on and off by repeated use of Fctn 0. They are handy, for example, for locating certain portions of your text in a hurry. Those of you with the TI-Writer manual would have noticed that on page 77 it specifies how to print the line numbers along with your text if you so desire. The instructions given in the manual are as follows:

"Select the PrintF command. Type L and space once. Then type the devicename. Press ENTER to execute the function."

In terms of a typical 'formula' it would look like this:

L RS232.BA=4800

I have never noticed in the manual that you can also print the line numbers as well as specify the lines to be printed. A typical formula for this would be:

L 2 57 PIO

Notice that there is a space between L and 2 as well as between 2 & 57 and 57 & PIO. I have often made use of this feature when printing out lists for school work.

My main objective in writing this brief article, however, is to outline a method that I have developed for printing an entire text but requiring line numbers to be printed with part of the text. I came across this last year while printing out a list of an entire form of students together with the line numbers. A portion of the result of the printout both with and without line numbers is as follows:

YEAR 10 1992 - CORE AND ELECTIVES
(UPDATED 31/12/91)

NAME	D.O.B.	ROLL	ENG	MAT	SCI	GEO/	HI	BLU	PINK	YELL
ABDY, GREG	24-06-77	B1	LES	M3	S3	HIST	HUM	COM	BAW	
ACHARYA, KUSUM	06-12-76	B3	DAW	M3	S1	GEOG	CST	TD	BAW	
ADAMS, MELISSA	18-10-76	B1	E1	M3	S3	GEOG	COM	HUM	PMG	
ALLEN, ADAM	16-05-76	B1	DAW	M4	S2	HIST	TW	TD	ART	

YEAR 10 1992 - CORE AND ELECTIVES
(UPDATED 31/12/91)

NAME	D.O.B.	ROLL	ENG	MAT	SCI	GEO/	HI	BLU	PINK	YELL
0001										
0002										
0003										
0004										
0005										
0006										
0007	24-06-77	B1	LES	M3	S3	HIST	HUM	COM	BAW	
0008	06-12-76	B3	DAW	M3	S1	GEOG	CST	TD	BAW	
0009	18-10-76	B1	E1	M3	S3	GEOG	COM	HUM	PMG	
0010	16-05-76	B1	DAW	M4	S2	HIST	TW	TD	ART	

The printout with line numbers was not entirely satisfactory for obvious reasons. The lines numbers will always start at the top of the page at the beginning of the text. It was of little use to me to have the class list numbered from the very top as I wanted number one to coincide with the first name on the list. I was determined that I was going to work out a way of doing it without resorting to a 'cut and paste' job. I finally succeeded and the printout that follows is the result.

YEAR 10 1992 - CORE AND ELECTIVES
(UPDATED 31/12/91)

NAME	D.O.B.	ROLL	ENG	MAT	SCI	GEO/	HI	BLU	PINK	YELL
0001	24-06-77	B1	LES	M3	S3	HIST	HUM	COM	BAW	
0002	06-12-76	B3	DAW	M3	S1	GEOG	CST	TD	BAW	
0003	18-10-76	B1	E1	M3	S3	GEOG	COM	HUM	PMG	
0004	16-05-76	B1	DAW	M4	S2	HIST	TW	TD	ART	

What follows are the steps that I follow when I want to start numbers at a point further down the text. First of all I noticed that when the line numbers were printed with the text that the entire text was pushed six spaces to the right. The coincides with what is says on page 77 of the manual under the title 'Printing Line Numbers' - "The maximum line length when printing with line numbers is 74 characters per line. Any characters in columns 75-80 are not printed". Of course 75-80 represent six spaces so this is the reason why. Notice that for the text in my example there are six lines before the student names begin. To get the text printed the way I have shown it above I do the following:

1) While in 'fixed mode' I 'push' each of the first six lines six spaces to the right. [To 'push' a text to the right you (while in fixed mode) insert spaces by using Fctn 2 and hitting the space bar, making sure the cursor is to the left of the text you are pushing to start off with.] I do this because I am going to print these lines without line numbers and the remaining lines with line numbers so I want the first six lines to stay 'lined up' properly with the remaining part of the text.

2) Then I print the first six lines without line numbers by typing in:

1 6 RS232.BA=4800

3) The you use Fctn 3 to delete the first six lines. Now the first line number, 0001, is level with the first name on the student list.

4) Now you print the rest of the text with line numbers by typing in:

L 1 E RS232.BA=4800

If you do not want the entire rest of the text to have line numbers, then substitute E (which means End) with the desired number.

This has helped me to make my form lists look more professional and, incidentally, the TI produces the only class list for an entire form at Airds High School despite there being heaps of other computers around the place. Perhaps they are not so easy to use?

Treasurer's Report

by Geoff Trott

Well as a result of the AGM, everything is back as it was! Here I am giving you all the good news still. Hopefully we will have an excellent year culminating in the Fair in November. It may well be that all the extra work that this will involve may mean that there will be quite a few vacancies for the next AGM. Please do all that you can and then a bit more to help this year. Do not wait to be asked, join in and get the most out of your club. We all know that you only get out of something what you put into it. It can be quite fun really, I think!

Income for November	\$3372.35
Payments in November	\$911.73
Excess of income over expenses for November	\$2460.62
Income for December	\$2433.40
Payments in December	\$2795.54
Excess of expenses over income for December	\$382.14 <input type="checkbox"/>

continued from page 2

If members want an assembly class, then each class member 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).

I propose to start the meeting at 2pm, 1st February, 1992.

Hints, Tips and Answers

TI-Writer

compiled by Stephen Shaw, England

('F' refers to a tip for the FORMATTER while 'E' refers to a tip for the Editor section. Equivalents include Funnelweb.)

1F Outdenting- This is the reverse of "indenting". It will allow the first line of a paragraph to be started further to the left then the remaining lines in the paragraph...

eg - ".LM12;IN7;RM71"

... causes the first line to start at column 7 and subsequent lines to begin at column 12.

2E When using REPLACE STRING you should use word wrap OFF if the document lacks carriage returns or you will end up with one huge paragraph. With word wrap ON, Replace String will reformat from the amended word to the end of the paragraph.

3E This is for use in Funnelweb Text Editor mode (not TI Writer) Want to convert the character case (Upper to Lower; Lower to Upper)?

- i) Upper to Lower - depress CTRL and "." (period)
- ii) Lower to Upper - depress CTRL and ";" (semi-colon)

By keeping the keys depressed the auto-repeat function will take affect and every character the cursor passes over will be converted (you only need hold the ',' or ';' once auto repeat takes over- you can let go of CTRL).

4E To save part of a document insert in front of the filename the first line number you wished saved then either a comma or a space and then the last line number you wished saved.

eg - 5 30 DSK1.MINUTES or 5 30 DSK1.MINUTES - will save the lines 5 to 30, inclusive, onto disk drive #1 under the name MINUTES.

5E To "get rid" of the line numbers on the left of the screen press FCTN 0 (zero). To get them back press FCTN 0 again.

6E If you erase a line in error, press CTRL I (known as "OOPS!") and your line will be restored. Note: This usually only works if you have pressed no other keys after the accidental erasing. There are a few exceptions, however. I have often had it happen that after editing several parts of a line and then making a mistake I tried to recover by using Ctrl I. By doing so I found that "OPPS!" returned the entire line to what it was like before I started editing it!

7E When using the "SearchH" command remember that the search is only from the point that the cursor is located. Therefore to search the total document the cursor must be on line 1 before you go to the Command mode.

8E To backspace beyond the left margin press CTRL Y. This will temporarily disable the left margin. It will also disable the right margin in latest Funnelweb ONLY. In both cases the cursor should be next to the margin.

9F When using the Header or Footer command with the page number it is possible to have NO value printed (such as for the introduction, etc) by using the .PA format command with a value of zero. The page numbering will begin on the following page. A .PA at the end of each page will delay the numbering further.

10E TI-Writer can save a file in other than the normal D/V80 format by using the PF command and either putting a "F" in front of the filename (ie F DSK1.MYFILE) or by putting a "C" in front of the

filename. "F" causes a file to be created in a Display/Fixed 80 format. "C" strips any control characters from the file as it is sent.

11E TI-Writer can be used as a database. Each line must be a record and set up exactly the same. For example if the data was names, addresses and phone numbers then all names must start in the same column; all addresses must start in the same column and all phone numbers must start in the same column. There can be no lines which are blank or which have other type of information on them (ie - titles). Then using the program SORT UTILITY (by D R Romer & J Clulow) you can sort this file. Once sorted, which is done very quickly, titles can be added if you are printing it out.

12E There are CTRL keys equivalents to most FCTN keys, plus a few others. For example to tab to the right you can go FCTN 7 or CTRL W (Funnelweb also has CTRL Z) '; to tab back (to the left) you can go CTRL T (there is no equivalent FCTN key).

13E If you must go to the bottom of your text (and it is very long), instead of paging down simply go to the Command Line and press "S" for "show line" and at the prompt "enter line number" just type "E" and press ENTER. "E" is a valid line number for the last line (end) of a document. This feature is active in all the commands requiring you to enter a line number.

14F If you wish to prevent the form feed at the end of printing when using the Formatter then make the last line of your text ".PL 1". This will suppress the form feed, but note: do not forget to reset PL if you have another document to print.

15F You can string the formatter commands on the same line separated with a semicolon.

eg - LM 10;RM 70;IN +5;FI;AD

There does not have to be a space between the formatter command and the number which follows it. In other words, the commands-

.LM 4 and .LM4 are equivalent.

16F If you are having problems with formatter commands, make sure they are UPPERCASE letters.

17E To avoid a BUFFER FULL notice you just save the files as it gets larger, then use the SD command to see the file size. The BUFFER is usually becoming full at 92 sectors. [due to use of Run Length Encoding in the buffer area, a full buffer may occur at very different file sizes depending on nature of text. Stephen].

18F When using the .CE command you MUST also use the .LM and .RM commands because .CE centres the text between the SET margins, not necessarily the middle of your paper.

19F The Formatter also ensures that you have two spaces after each period. To control this when you are typing such things as "Mrs. E Smith" or "1023 N. Queen Street" then use the " " sign in place of the space after the period.

eg - Mrs. E Smith; 1023 N. Queen Street

20F If you must have a dot in column one of your text, transliterate it. ".TL 124:46" will allow FCTN A to print out a period. The reason for avoiding the dot (.) is because when the text is passed through the formatter the whole line following the dot will be erased. If you are not passing it through the formatter then there will be no problems.

21F To create a file without line feeds yet Formatted, do the following:

- i) Use the FORMATTER to Print the text to disk
- ii) Go back to EDITOR, load the formatted text, and do a Print File (PF) replacing PIO with C DSKn.filename.

22F If you wish to include a program listing in your document instead of retyping it into TI-Writer just LIST the program to the diskette using the following command:

```
LIST "DSKn.filename".
```

This will save the program in DISPLAY/80 format which allows it to be read by TI-Writer. You can now load this file into TI-Writer and place the carriage return character at the end of each program line.

23E If you wish to place a Carriage Return at the end of a line of text (a line without the return will usually occur when you have inserted blank lines in the text and then put text on them) then place the cursor at the end of the text and press CTRL 8. This will place a Carriage Return where you want it and insert a blank line below. If this line is not wanted you can delete it with FCTN 3. Or, alternatively, enter Special Character Mode by pressing Ctrl U, press Shift M and the leave the Special Character Mode by pressing Ctrl U again. This may seem like a lot but when you get used to it you can perform the three-step operation is less than a second!

24E You can get a print out of your file WITH LINE NUMBERS when printing out of the Editor mode by placing an "L" and a space before the printer name in the command instruction.

eg - L PIO

This will eliminate the last 6 characters at the end of each line (#75 to #80) therefore keep your line lengths to a maximum of 74.

25E Did you know that you can type anything you want after a carriage return ON THE SAME LINE and it will not print out; but it will SAVE. This is great for text notes for screen reading.

26E When typing up a document which uses certain long words or phrases frequently then a time saver (and also added insurance against typing errors) is to type the words in shortform or initials (ie - TI-ARTIST could become TIA; Ottawa TI-99/4A Users Group could become OTIUG). When you are finished with the document use the Replace String function (RS) to change the words back to the full spelling (eg - /TIA/TI-ARTIST/). Care must be taken in three forms when using this:

i) the search only begins from the spot that the cursor is on so to do the whole document insure that the cursor is on line 1 before starting.

ii) the search will locate all occurrences of the string. Therefore if the string searched for is "at" it will find word "at" and also the "at" in "cat" and "that", etc. So before telling the machines to "Change all occurrences automatically be sure this program cannot arise. If you are not confident of this it is best to walk through and change each separately as it is found.

iii) as a reformatting will be done wherever a change is made it would be wise to review the document after to be sure that it is still formatted correctly.

27E When using the FIND STRING command you can specify which column range to search.

eg - 5 15 /text/ will look for the string "text" in the columns 5 through to 15 inclusive.

[also available with Replace String- sjs]

28F If your printer does not have a slashed zero and you want to print it out that way use the following Transliteration:

```
.TL 48:48,8,47
```

This will cause the normal zero (48) to be printed; then backspace (8); and then a printing of a slash (47).

29E Two files can be loaded into the Editor (assuming the total size is not too large for memory) by loading in the first file then doing a LF and entering E DSKn.YYY (where n=disk # and YYY=second file). This will load the second file after the end of the first file.

30E If you do not like the windowing when using the 80 column format then set the margins for 0 and 39 and turn off the line numbers (FCTN 0). When you are finished reset the left and right margins to what you desire and reformat each paragraph.

31E You can merge sections of a second file into the current document by the following entry using the LF command:

```
25 7 12 DSK1.YYY
```

This will load lines 7 to 12 (inclusive) from file YYY to the current document after line 25.

32E If you are using FUNNELWEB 4.1 or greater after you have the directory on the screen (using the SD command) you will be able to see how many lines are in a file by marking the file and then requesting (V)iew. The line count will be shown at the bottom of the screen as you read through the document. ○

Jenny's Younger Set

Here is the first 'arrival' for the new year, an interesting program from Vincent Maker, as usual. I am sure there are many out there of the younger set that could contribute something like this. Let's see Vincent have some competition this year. What about a LOGO program from somebody? Anyway, type this program in and let us know how you got on.

```
100 RIGHT=0
110 WRONG=0
120 REM*****
130 REM*QUIZ PROGRAM *
140 REM*DR WHO *
150 REM*FOR MELANIE *
160 REM*BY VINCENT MAKER*
170 REM*****
180 DISPLAY AT(5,7):"1. WHO KILLED TORBIS?"
190 DISPLAY AT(7,7):"A) AGGEDOR B) HEPERSH C) THE
DOCTOR D) KING PELADON"
200 PRINT "PRESS THE MOST APPROPRIATE KEY"
210 CALL KEY(O,K,L):: IF L=0 THEN 210
220 IF K<>65 THEN WRONG=1 ELSE RIGHT=1
230 CALL CLEAR
240 DISPLAY AT(5,7):"WHO KILLED ORTRON?" :: DISPLAY AT
(7,7):"A) SSKEL B) SSORG C) AZAYXR"
250 PRINT "PRESS THE MOST APPROPRIATE KEY"
260 CALL KEY(O,K,L):: IF L=0 THEN 260
270 IF K=65 THEN RIGHT=RIGHT+1 ELSE WRONG=WRONG+1
280 DISPLAY AT(5,7)ERASE ALL:"WHO TRANSPORTED THE FIVE
DOCTORS TO THE DEATH ZONE ?"
290 DISPLAY AT(7,7):"A) BORUSA B) THE CASTELLAN
C) THE CYBERMEN D) THE DALEKS"
300 PRINT "PRESS YOUR GUESS"
310 CALL KEY(O,K,L):: IF L=0 THEN 310 :: IF K=65 THEN
RIGHT=RIGHT+1 ELSE WRONG=WRONG+1
320 IF RIGHT=0 THEN A$="VERY POOR 0/3."
330 IF RIGHT=1 THEN A$="POOR 1/3."
340 IF RIGHT=2 THEN A$="GOOD 2/3."
350 IF RIGHT=3 THEN A$="VERY GOOD 3/3."
360 CALL CLEAR
370 PRINT A$
380 END
```

For Sale

Santron Computone Cassette Recorder
(Suitable for use with TI Computer)
\$30.00 only
Ring Percy on 808 3181

TI-Bits Number 13

by Jim Swedlow, CA USA

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

XMODEM

You may have heard of a transfer protocol called XMODEM and wondered what it is. If you use FAST-TERM or 4A TALK, you probably use it. The following should give you some idea of how it works.

When you communicate with another computer on phone lines through modems, your data must travel through the same voice phone lines that we use every day. Some connections are better than others. Most have noticeable static.

Your brain, a computer whose power has never been equalled, can usually distinguish the 'data' (voice) from the 'noise' (static). It is almost impossible for your computer to make this judgment.

In the early days of data transfer, data was simply sent and the receiving computer had to do as good a job as it could to distinguish between data and noise. In a text, or DV80 file, this was not a major problem. If one character was bad you could easily find the problem and edit it.

With a memory image or Program file, however, one bad byte could render the entire file useless. Although editing is possible, it is very tricky.

In August 1977, Ward Christensen developed an error detection method he called MODEM2. It was also dubbed "Christensen" protocol or XMODEM.

It was very simple. Data is sent in blocks of 128 bytes. XMODEM adds up the values of all the characters in each block and compares that number with a total that is sent by the sending computer. If they do not agree, the receiving computer sends a code to the sending computer and the block is transmitted again.

In 1982, Ward Christensen and Chuck Forsberg released an enhancement called Cyclic Redundancy Checking (CRC). CRC does sequential division on each character in the block resulting in a significant improvement in error detection.

Both protocols continue to be called XMODEM. Although others have been developed, XMODEM is used by all major systems, including Compuserve. (Source: an article in FOGLIGHT)

TI WRITER TIP

Find String (FS) is a powerful tool for finding something in a document. Just hit FCTN 8 and then enter FS. Your TI Writer gives you this prompt:

```
FIND enter /string/ :
```

You enter your string and use the slash as limiters. If you want to find the word "John", you would enter /John/. If you wanted to find John only when it is used as the last word in a sentence, you would enter /John./.

Should the "John" you find not be the one you wanted, you would go back to command mode and enter FS again. You will find /John./ still there. You just press enter and the search resumes.

Let us say, however, that now you want to find the word "Mo". But /John./ is on your screen. You could delete /John./. You could type in Mo but then you would have this:

```
/Mo/n./
```

Need you worry about the text after the second slash? No. Your TI Writer only searches for the information between the first and second slash. It ignores everything to the right of the second slash.

You will have a problem with that if you use Replace String, but that is another story.

THE PAPERLESS OFFICE

One of the things that futurists often project is the paperless office. Everything would be done on computers so paper would virtually disappear.

Not necessarily so. According to an article in a recent issue of 'The Office', the demand for paper has been increasing at the rate of 5% to 8% a year. Growth is expected to continue at that rate.

Cited reasons include the continuing shift from a production to a service economy and the fact that computers generate reams of paper. Also noted were the need to generate hard copies for filing and the proliferation of photocopy machines.

A MULTIPLAN APPLICATION PAYING YOUR BILLS

Home computers were sold for many things, some of which were even possible. One of the big selling points was managing your home finances. That was one of the motives behind my purchase. Well, folks, 'twern't true. A calculator is vastly superior for balancing a check book and, for most of us, the time and trouble required to maintain home records outweigh the benefits.

After many false starts, I developed a Multiplan application that I use for paying my bills. I wanted to automate the manual process of deciding how much goes to whom. That was a pencil, paper and calculator exercise of writing down what I owed, adding it up and then figuring how to dole out what was in my paycheck. Sound familiar?

This application does the following:

- * Remembers your fixed expenses (house note, etc).
- * Has a place for the monthly expenses that vary (utilities, etc).
- * Keeps track of the running balance and the check number for each check.

A sample of the finished product will appear a bit further down in this article. Here are step by step instructions for making it.

With a blank Multiplan screen, press <F> for Format and then <D> for Default. Press <W> for Width and set the default width to 11. Then press <ENTER>. For those of you who are not used to Multiplan, in this and following steps, do not press <ENTER> until I tell you to.

Press <F> for Format and then <C> for Cells. Type in C3:4 and then tab (CTRL 2) twice. Press <\$> and then press <ENTER>.

Move your cursor to R1C4 and press <F> for Format and then <C> for Cells. Press tab <CTRL 2> twice. Press <D> for DEFault and then <ENTER>. Move the cursor to R9C3 and repeat this process.

Push <O> for Options and then <N> (to turn 'recalc' OFF). Now press <ENTER>.

Move your cursor to R1C1, and press <F> for Format and then <W> for Width. Set the column width for column 1 as 6.

Enter the data for the first twelve rows. Be careful that the numbers are entered as values not as alpha characters. For the dollar amounts, do not enter the dollar sign, Multiplan will add it. Use amounts

that fit your situation. It is not necessary to enter the decimal if there are no cents (.00).

1	2	3	4
1		OCTOBER	1987
2			
3	CHECK DUE TO	AMOUNT	BALANCE
4			
5	Balance	\$100.00	
6	Pay Check	\$900.00	
7	Service Chg	\$6.00	
8	Cash	\$20.00	
9	Next Check	1223	
10			
11	MORTGAGE	\$234.44	
12	GAS	\$57.20	

Now we are ready for some formulas:

Location	Formula
R5C4	=R5C3
R6C4	=R5C4+R6C3
R7C4	=R6C4-R7C3
R8C4	=R7C4-R8C3

What have we done? We made a place for your beginning check book balance. We have a cell for the amount of your pay check, your monthly service charge AND for that handy cash advance you got from the electronic teller on the way home. We have entered formulas for updating your check book balance. Finally, we told Multiplan the number of the next blank check in your check book.

Two more formulas:

Location	Formula
R11C4	=R8C4-R11C3
R11C1	=IF(R11C3=0,R9C3-1,R9C3)

More about that IF function later. For the formulas in row 12 we need to move the cursor. Place the cursor on R12C4 and press <=>. Move the cursor to R11C4 and then press <->. Now move the cursor to R12C3 and then press <ENTER>. Your formula should look like this:

R[-1]C-RC[-1]

Move your cursor to R12C1 and press <=>. Type 'IF(' [The single quote mark is used here to show what you type in - in this case letter I, letter F and open parenthesis.] Move the cursor to R12C3 and then type '>0,'. Now move the cursor to R11C1 and type '+1,'. Move the cursor to R11C1 again and then type ')' and then press <ENTER>. The formula should read:

IF(RC[+2]>0,R[-1]C+1,R[-1]C)

This IF function tells Multiplan what check number to put in R12C1. If R12C3 (the amount of the check for that row) is greater than zero (>0 or you are paying that person something), increase the check number for the previous line (R[-1]C or R11C1 in this case) by one, otherwise copy it.

The end is in sight. Figure the highest number of bills you will ever pay and add a few. Lets say your total is 15. Move your cursor to R12C1 and press <C> for copy and then <D> for down. Type in 13 (you already have two) for the number of cells and then tab (CTRL 2) to the 'starting at' field and change it to R12C1:4. Now press <ENTER>.

We need some totals. Now move to R27C1 (use <G> for Go). If your number of bills is other than 15, the location of these total lines will be different. Enter:

R27C2	TOTALS
R27C3	=SUM(R11:25C4)
R27C4	=R25C4

This repeats your final check book balance and totals the amounts you have paid.

The next step is to lock your formulas. Press <L> for Lock and then <F> for formulas. Then press <Y> to confirm.

You must now unlock some cells that you will want to change. Press <L> for Lock and then <C> for Cells. Type in R11:25C2 and then press tab (CTRL 2). Press <U> to Unlock and then <ENTER>.

You should also unlock the month cell. Move your cursor to R1C3 and press <L> for Lock. Press <C> for Cells and then tab (CTRL 2). Now press <U> to Unlock and then <ENTER>.

Now all you have to do is to go back in and enter your numbers and the names of the fine folks you owe money to and how much you owe.

You have built your spreadsheet.

Since automatic recalculation is off, remember to press Recalc (FCTN 8) to update the totals when you change information. You should <T>ransfer <S>ave this sheet before going any further.

About once a week, I update my sheet with the bills that came in the mail (bills and advertisements seems to be most of the Postal Service's offerings). On payday, I enter my check book balance, the next check number and the amount my employer paid me. I can adjust what I pay to make sure I have enough left over to make it until next payday. As Bill Harms says, you can go "what-if'ing" to work out the best solution.

Enjoy.

	1	2	3	4
1			OCTOBER	1987
2				
3	CHECK DUE TO	AMOUNT	BALANCE	
4				
5	Balance	\$100.00	\$100.00	
6	Pay Check	\$900.00	\$1000.00	
7	Service Chg	\$6.00	\$994.00	
8	Cash	\$20.00	\$974.00	
9	Next Check	1223		
10				
11	1223 Mortgage	\$438.21	\$535.79	
12	1224 Gas	\$57.20	\$478.59	
13	1225 Dpt Store	\$40.00	\$438.59	
14	1226 VISA	\$75.00	\$363.59	
15	1227 Gas Company	\$45.73	\$317.86	
16	1228 Disk Land	\$30.00	\$287.86	
17	1228 ----	\$0.00	\$287.86	
18	1228 ----	\$0.00	\$287.86	
19	1228 ----	\$0.00	\$287.86	
20	1228 Put One	\$0.00	\$287.86	
21	1228 Time Only	\$0.00	\$287.86	
22	1228 Bills Here	\$0.00	\$287.86	
23	1228 ----	\$0.00	\$287.86	
24	1228 ----	\$0.00	\$287.86	
25	1228 ---->	\$0.00	\$287.86	
26				
27	TOTAL	\$5013.02	\$287.86	o

Control of CS1

by Ed Hall, CA USA

[This article was retyped from the September, 1991 issue of the LA 99ers Topics.]

Have you ever wondered if there was a way to control the remote line which turns the cassette motor on and off? Well, there is a way and it does not take too much work either. You do need the 32K memory and, for the following program, the Extended Basic Cartridge as well.

continued on page 22

Hollywood Hijinx part 2

by Scorpia, Copyright 1987 USA

Still in the Living Room, take a good look at those three statuettes (actually candles) over the fireplace. Red, white, and blue, how patriotic! Even if they do happen to be Hindu gods. But there is something else about those candles. Each is holding up one hand, either right or left, and each hand displays a different number of fingers. Hmmm. A right hand with three fingers, a left hand with seven (seven???) , and a right hand with five fingers. Gee, you would think that was the combination to a safe or something.

Well, let us not worry about that now. It is off to the Screening Room and environs (that is north and east, for those of you who have not been mapping; and if you have not, shame on you!). And yes, here is another of those punch cards, a yellow one this time. Grab that, then go south to the Projection Room.

Several interesting items in this room. First, take the slide and put it in the slide projector, then turn the projector on and focus it. Now get the film, put that in the film projector, and remove the lens cap from the film projector (all kinds of sneaky little traps here, eh?). Turn on the film projector and look at the screen.

It is a message from Aunt Hildegard! Do pay special attention to it. The song she tells you to play varies with each game, and is chosen randomly. Thus you really have to come here and set everything up to read it. So make note of the song title, then return to the Foyer and from there East to the Parlor.

Along the way, in the Long Hall, you will pass an interesting painting; we will come back to that shortly. Continue to the Parlor, where you see a piano. You also see that all the furniture in the room, except the piano, is bolted to the floor. How strange!

No matter. Play the song title you got from Aunt Hildy, and suddenly a door opens up in the floor! Below is a crawl space, but you do not want to go there just yet. First, push the piano north to the wall. Now go down into the cramped space, and go south.

You notice a dirty pillar supporting the floor here. Get that and drop it. Now go north and back up to the Parlor. Push the piano south twice. Wow! The floor begins to tilt up! Aha! Go down into the crawlspace, and this time, you are able to go north, and get the parking meter from that biting political commentary, "You Cannot Fight City Hall--But You Can Blow It Up!"

When you return to the Parlor, open the piano, and inside is...yep, you guessed it, yet another of those punch cards, this time a violet one. Will it never end? Keep going and you will find out!

You are done with the Parlor, so head along back the way you came, stopping off this time by the painting. Of course you know what is behind it. Right, a safe. And a green punch card, too! Take the card, then examine the safe. It has a combination dial, which can be set to any number 0-10.

Got any ideas here? Seen anything recently that might be a safe combination? How about those candles? Hehehe! So you turn the dial right 3, left 7, and right 5. Click! Now you can open it, and get Mamma Maggio's cheese grater (note: you have to do the safe right the first time, so you may want to save the game first in case of accidents).

You are certainly moving right along here! Speaking of moving, move to the Living Room and drop off your latest acquisitions. Now go north and west to the kitchen, and down into the cellar. Hey, lookee there, a super-hi-tech computer prop!

And it works, too. But you still need one more card (and that is the last one, fortunately). Open the closet door, and step into the shaft. Yep, there is a blue punchcard. Take that and return to the cellar. Of course, it is obvious that you have to put the cards into the computer, but in what order?

Well, those cards are also the same colours as the colours of the rainbow. In the good old days of real education, teachers had a mnemonic device they taught their students for remembering the proper order of the colours. It was (can you guess?) ROYGBIV (now where have you seen that before?).

So, turn on the computer, and insert the cards, one at a time, in the proper order, into the slot. Each time a card goes in, the display of lights on the front changes. You do not have to look at them, tho, until after the last card has been fed in. When you do look at the lights, they spell out a phone number.

Ok, hop up the stairs to the Living Room, and call that number on the telephone. Aunt Hildegard cannot come to the phone right now, but she reminds you to look in the hopper (you do have to make the call or you will not get the treasure). So it is back downstairs, and sure enough, sitting in the card hopper is Uncle Buddy's toupee (there are some who may dispute that this item is a treasure, but we will not go into that here).

Another trip to the Living Room to drop off the toupee, then it is on to the Game Room, and probably the most fun part of the game, the Tokyo model with the Atomic Chihuahua!

This certainly is a very detailed scale model, all right. If you have fooled with this before (or if you have not, save now and do so), you know what each of the buttons does. The object is to get the "dog" safely over to the end of the model, so you can grab the Big Diamond Ring. Never mind saving Tokyo; it can stand to be destroyed one more time.

Ok, let us get started! Push the green button and Chi (for short) rumbles along eastward. But wait...defenders are coming on the scene! Two tiny tanks and two tiny planes! Do not be deceived by their size; given enough time, they can put an end to Chi, and that is the last thing you want!

Push the green button until Chi is near the tanks. Push the black one. Stomp! One less tank to worry about. But the planes are a nuisance, too, so push the white button. Swipe! One plane down! Chi is still taking heavy damage, tho, so push the black one again. Take that, tiny tank! Crunch. And now the white button...whoosh! The enemy has been defeated!

Now move Chi east to the monument. Wait a minute... What is that sneaking out of the bushes? It is a rocket truck! And it just fired a rocket! How dare they???? Hmff! Push the black button and stomp that truck! Hah! The rocket goes wild and wipes out the Ginsu Knife Company!

Now push the blue button, and Chi grasps the Big Diamond Ring. We are getting there! Green button again, until Chi is right up against the dome. Push the red button. Chi breathes flame right on the plastic. Do it again, and a hole begins to form! Once more, and you can get the diamond ring! (You need three bursts of flame, and that is all Chi can manage, so do not use flame against the tanks, planes, or rocket).

Gee, that was fun! (Ok, so I am a kid at heart... are we not all?). It is back to the Living Room to drop off the ring, and prepare for the final two problems, and the end game, which is full of surprises!

Hollywood Hijinx is copyrighted 1987 by Infocore Inc.

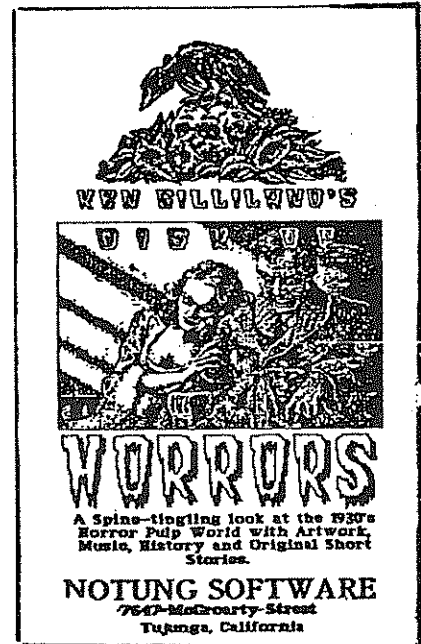
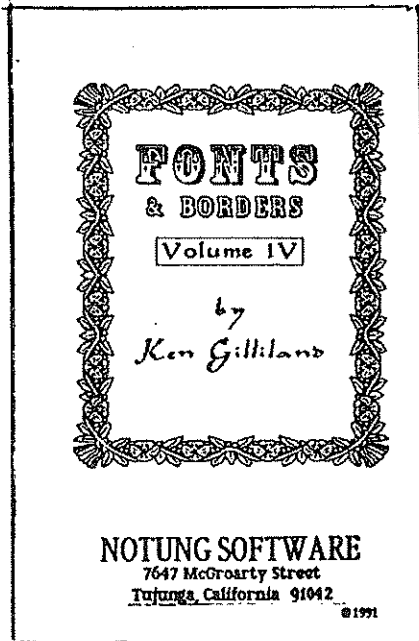
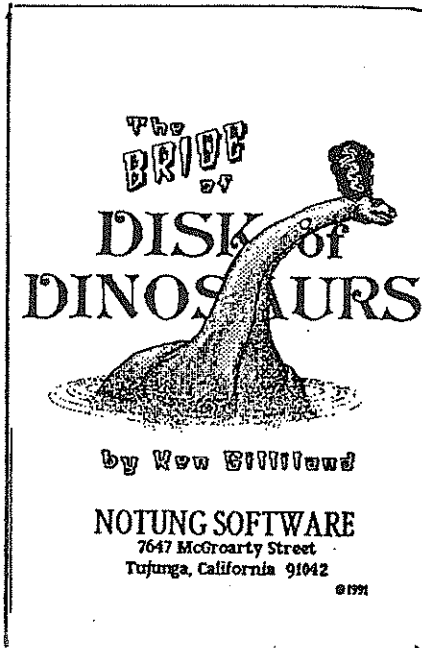
This walkthru is copyrighted 1987 by Scorpia, all rights reserved.

Author Index 1991

Author	Title	Description	Page No	Author	Title	Description	Page No
Adlemann,JC HCM4.03,USA	Program to type in	Missile math	11.18	Machonis,Ed USA	Tips on TIPS	Software review	11.25
Aibright,Ron USA	Insights	General interest	7.02	Maker,Vincent	Younger set	Program	3.20
Amadio,Lou	CSGD software overview	Software review	2.06		Younger set	Program	5.04
	Labels with TI-Writer	Word processing	5.18		Younger set	Program	6.18
	Reviewing club software	General interest	7.16		Younger set	Programs	11.17
	Rock Runner	Software review	6.04		Younger set	Quiz	4.15
	Techo time	RAMdisk disable,EPROMs	1.05		Younger set	Quiz	8.26
	Tris	Software review	7.05		Younger set	Quiz	10.20
	Typewriter 99	Software review	4.04	McCarthy,Chris	Joystick special	Hardware review	10.08
Arnold,Tom Ont,Canada	Archiving	Software review	2.21	McGovern,Tony	Assembly squeezing, part 5	Software hints	1.03
				Miller,Craig CA,USA	GPL disassembly of E/A module	Software hints	4.30
				Mudie,Peter	Clock modifications	Hardware hints	9.12
Ball,Eric	Younger set	Games	2.11	Mudie,Ross	Assembly and trains	Software hints	4.07
Ball,Michael	Letter to editor	Page Pro	9.03		Assembly class	General interest	1.07
Balthrop,William USA	Program to type in	Boolean brain	5.11		Assembly class	Link-It #19	2.07
Brown,Robert	Games information	General interest	7.12		Assembly class	Software hints	7.21
	Games information	General interest	8.10		Assembly class	XB loader	3.04
Brubaker,Ron	Sorting, part 1	Software hints	10.06		Attention: BBS members	Club news	8.05
USA	Sorting, part 2	Software hints	11.22		Communicators	BBS information	2.02
Bull,John H	Contract bridge V3.00	Software review	11.28		Communicators	BBS information	8.14
					CQ...CQ...	General interest	6.17
					Letter to editor	Thank you	3.02
Cell,Dry	Colour monitor fault	Hardware hints	11.30		Link-It #18	Software hints	1.07
	To charge or not to charge	Hardware hints	6.08		Link-It #20	Software hints	9.17
Charlton,Paul VA,USA	Fastterm documentation	Documentation	4.27	Mullins,Ian	BBS competition	General interest	10.02
Christensen,Col TIBUG	Multiple menus for BOOT	Software hints	8.06	Nuvalini,Joe CO,USA	C'est la vie, ma cherie Geneve	General interest	5.02
Christensen,Garry TIBUG	TI Fair, a few thoughts	General interest	11.02	O'Neil,Tim USA	Cartoon documentation	Software review	8.12
				Opheys,Thomas Germany	TI99/4A emulator	General interest	6.02
Dhein,Anne IL,USA	TI-Writer graphics	Word processing	7.09				
Dorais,Lucie Ont,Canada	Rotating TIA instances	Software hints	8.15	Packham,Randy Ont,Canada	Love my computer not user group	General interest	3.06
	Program to type in	Barbie Boutique	6.13	Payne,Harold	Letter to editor	Departing member	7.04
	Root finder	Software hints	2.20	Peterson,Jim OH,USA	Air taxi	Software review	4.20
					Appreciate your programmers!	General interest	1.19
Garoché,Pierre France	Letter to editor	New member	7.04		I like brain games	General interest	5.09
Green,R.A. Ont,Canada	Look at assembler language #1	Software hints	2.22		Putting it all together #7	Software hints	4.11
	Look at assembler language #2	Switches	3.21		Putting it all together #8	Software hints	5.10
	Look at assembler language #3	Calling subroutines	6.14		Reformatting	General interest	10.09
	Look at assembler language #3	Subroutines	4.11		TI99/4A world news	General interest	9.16
	Look at assembler language #4	Entry and exit	5.20		TI99/4A world news	General interest	11.07
	Look at assembler language #6	String searching	6.14		Tips from the tigercub #60	Software hints	1.15
	Look at assembler language #7	Counting	11.31		Tips from the tigercub #61	Software hints	2.09
	Look at CS1	String searching	9.18		Tips from the tigercub #62	Software hints	9.11
	Look at GPLLNK	Software hints	9.18		Why should you learn to program?	General interest	1.09
	Look at GPLLNK	Software hints	11.31		You do not have to have it all	General interest	3.05
	Look at interrupts	Software hints	7.20	Phillips,Terry	AGM agenda, 1991	Club news	10.02
Harms,Bill USA	Multipan machinations	Spreadsheet	8.21		Minutes of 4th AGM, Dec. 1990	Club news	10.04
Harris,Daniel	Analog joystick	Hardware project	10.21		Secretary's notebook	Club news	1.02
	Auto-loading cassette files	Software hints	4.02		Secretary's notebook	Club news	2.03
	Program to type in	Typing	9.14		Secretary's notebook	Club news	3.03
	Terminal emulator II tips	Software hints	9.23		Secretary's notebook	Club news	4.03
Harrison,Percy	TiSHUG shop report	Club news	1.03		Secretary's notebook	Club news	5.03
	TiSHUG shop report	Club news	2.03		Secretary's notebook	Club news	7.03
	TiSHUG shop report	Club news	3.04		Secretary's notebook	Club news	8.03
	TiSHUG shop report	Club news	4.04		Secretary's notebook	Club news	9.03
	TiSHUG shop report	Club news	5.04		Secretary's notebook	Club news	10.03
	TiSHUG shop report	Club news	7.06		Secretary's notebook	Club news	11.03
	TiSHUG shop report	Club news	8.05		TiSHUG BBS	General interest	3.03
	TiSHUG shop report	Club news	9.04	Raguse,Earl CA,USA	Back up copy?	Software hints	6.05
	TiSHUG shop report	Club news	10.05		Beginning Forth #2	Software hints	1.17
	TiSHUG shop report	Club news	11.04		Beginning Forth #3	Software hints	2.16
					Beginning Forth #4	Software hints	3.17
					Beginning Forth #5	Software hints	4.26
Kazmer,Ray CA,USA	Comic show instructions	Software review	8.13		Beginning Forth #6	Software hints	5.21
Krawczyk,Gene ATICC	G, the HIRES language	Software hints	8.07		Beginning Forth #7	Software hints	6.21
					Beginning Forth #8	Software hints	7.23
					Beginning Forth #9	Software hints	8.25
Lawrence,Al	TIM installation	Hardware hints	8.04		Beginning Forth #10	Software hints	9.19
					Beginning Forth #11	Software hints	10.17
					Modern revolution?	General interest	4.17

Author	Title	Description	Page No	Author	Title	Description	Page No
Rebel,E.P. Netherlands	Speech	Software hints	5.17	Smoley,Martin OH,USA	TI-Base tutorial #14	Data base	9.21
Relyea,Bob	Editor's comment	General interest	1.01	Spirit of 99 OH,USA	TI-Base tutorial #15	Data base	10.19
	Editor's comment	General interest	2.01		Handy tips for the TI99/4A	Software hints	2.11
	Editor's comment	General interest	3.01	Sponchia,Bill USA	Hints, tips and answers	Software hints	11.29
	Editor's comment	General interest	4.01				
	Editor's comment	General interest	5.01	Stith,Wayne VA,USA	Hyphenator version 2.0	Software review	8.11
	Editor's comment	General interest	6.01				
	Editor's comment	General interest	7.01	Swedlow,Jim CA,USA	Extended BASIC tips #5	Software hints	3.08
	Editor's comment	General interest	8.01		Extended BASIC tips #6	Software hints	4.10
	Editor's comment	General interest	9.01		Extended BASIC tips #7	Software hints	5.08
	Editor's comment	General interest	10.01		Extended BASIC tips #8	Software hints	6.06
	Editor's comment	General interest	11.01		Extended BASIC tips #9	Software hints	7.18
	Extended BASIC problem	Software hints	9.15		Extended BASIC tips #10	Software hints	8.19
	Extended BASIC tips	Software hints	9.07		Extended BASIC tips #11	Software hints	9.10
	Extended BASIC tips	Software hints	10.13		Extended BASIC tips #12	Software hints	10.08
	Extended BASIC tips	Software hints	11.08		Extended BASIC tips #13	Software hints	11.27
	Newsletter update	General interest	1.21		TI-Bits #3	Software hints	2.08
	Newsletter update	General interest	2.22		TI-Bits #4	Software hints	3.07
	Newsletter update	General interest	3.12		TI-Bits #5	Software hints	4.09
	Newsletter update	General interest	4.29		TI-Bits #6	Software hints	5.07
	Newsletter update	General interest	6.22		TI-Bits #7	Software hints	6.05
	Newsletter update	General interest	7.26		TI-Bits #8	Software hints	7.17
	Newsletter update	General interest	10.22		TI-Bits #9	Software hints	8.17
	Program to type in	Grade standardiser	7.13		TI-Bits #10	Software hints	9.09
	Program writing	Software hints	7.21		TI-Bits #11	Software hints	10.07
	Transliteration and languages	Word processing	7.10		TI-Bits #12	Software hints	11.23
	Tutorial day, Multiplan, TI-W	Club news	4.02				
	1000 Words, TI-Writer utility	Word processing	7.07	Takach,Ben	EPROM file decoding	Software hints	10.15
Rokke,Norman OH,USA					EPROM file decoding	Software hints	11.33
Ruggeri,Aff	Coming to grips with TIPS	Software review	11.09		Visiting TI99/4A users in USA	General interest	11.05
Ryan,John	Death of a CAD!	General interest	10.13	Tilling,Ashley	Wordwriter-	Software review	1.06
				Trott,Geoff	Letter to editor	Life members	4.02
Scheidemantle,Paul USA	TI-Writer file keeping	Software hints	4.03		Newsletter update	General interest	6.22
Schlesinger,Herbert USA	Multiplan exercises #1	Spreadsheet	2.17		Printing graphics with DMP-105	Software hints	10.11
	Multiplan exercises #2	Spreadsheet	3.15		Printing graphics with DMP-105	Software hints	11.13
	Multiplan exercises #3	Spreadsheet	4.13		Software help	Software hints	9.08
	Multiplan exercises #4	Spreadsheet	5.19		Spell It! review	Software review	1.14
	Multiplan exercises #5	Spreadsheet	6.19		Techo time	Formatting	4.05
	Multiplan exercises #6	Spreadsheet	7.22		Techo time	Formatting	5.05
	Multiplan exercises #7	Spreadsheet	8.20		Techo time	Formatting	6.03
	Multiplan exercises #8	Spreadsheet	9.20		Techo time	RAMdisk for MiniPE	7.19
Schreiber,Rolf	1000 Words exit fix	Software hints	7.16		Techo time	RAMdisk, consoles	2.05
	John Birdwell - in memorium	General interest	7.06		WD1773,EPROMs,consoles		3.22
	TiSHUG software column	Club software	1.04		Treasurer's report	Club news	1.04
	TiSHUG software column	Club software	2.04		Treasurer's report	Club news	2.23
	TiSHUG software column	Club software	3.06		Treasurer's report	Club news	3.07
	TiSHUG software column	Club software	6.04		Treasurer's report	Club news	4.02
	TiSHUG software column	Club software	7.05		Treasurer's report	Club news	5.20
	TiSHUG software column	Club software	8.05		Treasurer's report	Club news	6.20
	TiSHUG software column	Club software	9.05		Treasurer's report	Club news	7.24
	TiSHUG software column	Club software	10.05		Treasurer's report	Club news	8.18
	TiSHUG software column	Club software	11.04		Treasurer's report	Club news	9.03
Scorpio USA	Hollywood Hijinx part 1	Games review	11.15		Treasurer's report	Club news	10.08
	Lurking horror part 1	Adventure hints	5.12		Treasurer's report	Club news	11.03
	Lurking horror part 2	Adventure hints	6.09	Warburton,Dick	Co-ordinators report	General news	1.02
	Lurking horror part 3	Adventure hints	7.11		Co-ordinators report	General news	2.02
	Spellbreaker part 2	Adventure hints	1.20		Co-ordinators report	General news	3.02
	Spellbreaker part 3	Adventure hints	2.12		Co-ordinators report	General news	5.02
	Spellbreaker part 4	Adventure hints	3.19		Co-ordinators report	General news	6.02
	Spellbreaker part 5	Adventure hints	4.15		Co-ordinators report	General news	7.02
Shaw,Stephen England	Fractal graphics	Software hints	6.15		Co-ordinators report	General news	8.02
	Rambles	Software hints	3.11		Co-ordinators report	General news	9.02
	Rambles	Software hints	4.25		Co-ordinators report	General news	10.02
	Rambles	Software hints	5.13	Warren,Bob	G corner	Software hints	8.09
	Rambles	Software hints	11.19	ATICC	G, an introduction	Software review	8.07
	Reviews	Software review	11.16	Williams,Scott USA	Program to type in	Cyber-abacus	4.16
	TI*MES utility catalogue pt 1	Software review	2.13				
	TI*MES utility catalogue pt 2	Software review	4.21	Wynne,Tom WA,USA	DV/80 file converter	Software hints	1.20
	Tips	Software hints	4.12				
	TML graphics programs part 1	Software hints	3.09	Author Unknown			
	TML graphics programs part 2	Software hints	4.08	PA,USA	ASCII files to program files	Software hints	6.07
	TML graphics programs part 3	Software hints	5.14		Author index 1990	General interest	1.11
Smoley,Martin OH,USA	TI-Base tutorial #8	Data base	2.19	TX,USA	GPL, an introduction	Software hints	1.13
	TI-Base tutorial #9	Data base	3.13	USA	Graphing data with Multiplan	Software hints	10.14
	TI-Base tutorial #9 (cont)	Data base	4.18	USA	Program to type in	Great inventions	6.10
	TI-Base tutorial #9, pt 2	Data base	4.19		Program to type in	Itche worm	9.13
	TI-Base tutorial #10	Data base	5.15		Subject Index 1990	General interest	1.10
	TI-Base tutorial #11	Data base	6.18	IL,USA	Telecommunications with TI-W	Software review	9.06
	TI-Base tutorial #12	Data base	7.25				
	TI-Base tutorial #13	Data base	8.23				

New from Notung!



The Third in the Dino-Series is definitely the best! It includes over 25 TI-Artist Artfiles depicting not only solo dinosaurs, but dinosaurs in combat, nesting and even stages of standing up! Text files, a slide show and animation also compliment this package. \$12.

The fourth in a series of high quality TI-Artist Format Fonts and Borders includes the fonts; Coffee Can, Codex, Beton Open and Legend. Also included are 6 new borders. \$7

Also Available: Fonts & Borders II at \$7 +P&H, Fonts & Borders III at \$8 +P&H, or get any 3 Fonts & Borders Volumes for \$20 +P&H!

Venture into some of the most frightening sights ever brought forth to the TI world in celebration of the 1930's Horror Pulp Magazines. There's 30 bone-chilling TI-Artist Artfiles, a Horror Font, Music, some animation (too strong to mention here) plus a History of the Pulp along with 3 original Horror Short Stories by K.Gilliland. Three Disk package for \$12!

Updates to Products...

FILMLIB (Version 3.01) has an enhanced menuing system, better print routines, an enhanced onscreen display plus editing and appending features directly within the menu system. For TI-BASE V3 or above.

TI CASINO (Version 3.00) has 5 separate bank accounts for multiple users complete with passwords. Enhanced Craps play with more messages pertaining to dice totals and winnings. Raised house limits in Blackjack and Baccarat. True Vegas style Baccarat now allows to you play with the House, or for a Tie. Plus, the nightclub, "CLUB NOTUNG", has been added for those who wish to see a show while visiting TI Casino.

UPDATES, as always, are FREE to owners of the Software- just return the ORIGINAL disks and PROPER postage & mailer for its' return.

Package Deals are Real Steals!

DINO-FAMILY PAK... Get the Original DISK of DINOSAURS, plus THE BRIDE and THE SON for.....\$25

) Planning Calendar, 1992 - 1993

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

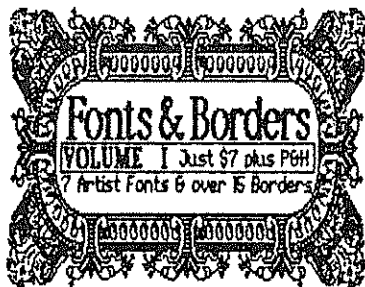


Of course you can have this dance, provided, of course, that you have...

BOOK of the LIST of DINOSAURS



- * 16 all-new Dinosaurs
 - * Educational Information on the life of Dinosaurs
 - * A Dinosaur Quiz Game
 - * An Animated Cartoon
- A two-disk set for only \$10



Also Available: Fonts & Borders II at \$7 +P&H, Fonts & Borders III at \$8 +P&H, or get all 3 Fonts & Borders Volumes for \$20 +P&H!



Tired of saving the universe and slaying dragons with your joystick? Want to go to a place where the action is always hot and your credit is always good? Well then, NOTUNG Software has just the place for you... TI CASINO.

TI Casino is a collection of eight Casino games, all interlinked, so you can play any of the games with the same money. Each game is virtually joystick operated with no confusing keypresses. At the end of your gambling session, take your winnings to the cashier and get a printed check!

ACEY DEUCEY ♦ BLACKJACK ♦ BACCARAT ♦ CRAP TABLES
KENO ♦ DRAW POKER ♦ ROULETTE ♦ SLOT MACHINES

Blackjack includes Doubling down, Splits & Insurance, there's unlimited betting in Roulette and on the Crap Tables, and above all, be sure to dine at the Green Parrot Restaurant while playing Keno!
\$15 through NOTUNG Software

Specify SSSD or DSSD Version, Joysticks, 32k & Ext.Basic Req'd, Printer Opt1

And yes, we are proud to admit the fact that this entire Advertisement was comp

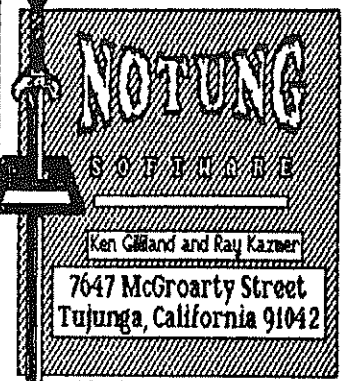


THE RING COMPANION

EXPLORE the fiery world of Richard Wagner's 'Der Ring des Nibelungen' through 14 TI-Artist format pictures of Dwarves, Giants, Heroes and Gods. There's also onscreen text of all aspects of the Ring and an informative and musical look at the Leitmotifs (or musical words) used in the Musical dramas. The two-disk package sells for just \$8 +P&H. A must if you have Ken Gilliland's Wagner fairware disks.

ALSO available from NOTUNG SOFTWARE:
FILMLIB for TI-BASE _____ \$7
CERTIFICATE 99 COMPANION PLUS _____ \$7
STAR TREK NEXT GENERATION CALENDAR
(JULY 90-JUNE 91) _____ \$7

SHIPPING: Please remember to specify Disk Format and include \$1 for the first piece of software and 50 cents for each add'l piece. Ask for our FREE illustrated catalog.



used & printed using a TI99/4a & 9640.

Printer Graphics

by Stephen Shaw, England

We have had a number of programs of screen graphics, but details of printer graphics seem a little scarce, and as the manuals tend to be a little unhelpful, this item is directed in that direction.

The EPSON standard is now covered by almost all "mainstream" printers, and to some extent almost all dot matrix printers sold today are compatible with the commands we shall be discussing first, although the second method may not be so widespread.

Firstly let us look at the way the Epson graphics mode works. On our computer characters are typically defined in an 8 x 8 block, in a manner fairly well known to most TI owners- consult your TI manual if in doubt! The TI defines its characters in two horizontal blocks, so that a straight line character could be defined as "FF000000000000" while a solid 8x8 block would be "FFFFFFFFFFFF FFFF" or a straight vertical line "8080808080808080". The printer has a print head with 9 pins arranged vertically, and characters are defined in a vertical manner to allow faster printing.

On the Epson, characters are nominally 8 dots high, with an option of shifting them down one row to use the ninth (bottom) printing pin. We shall here discuss only the use of the "normal" eight pins, and refer to the eighth pin from the top as the "bottom" pin- eight pin bit image graphics by definition do not use the ninth pin!

The bottom pin is allocated a value of 1, the next one up a value of 2, then 4 then 8 then 16 then 32 then 64 and finally the top pin a value of 128. We add up the values of all the pins we wish to strike in a single column and send this value to the printer- so a straight vertical line would require us to send a value of 255, eg CHR\$(255).

Sending several columns to print a character needs a little rethinking from the TI approach however, as the Epson graphic standard uses a half column. Imagine that the usual TI columns are numbered 1,3,5,7 and so on. These print exactly next to each other. The Epson standard allows us to print dots half-under each other, like this:

TI:	EPSON:
XXX XXX XXX XXX	XXX XXX XXX XXX XXX
XXX XXX XXX XXX	XXX XXX XXX XXX XXX
XXX XXX XXX	XXX XXX XXX XXX
XXX XXX XXX	XXX XXX XXX XXX
XXX XXX	XXX XXX XXX XXX
XXX XXX	XXX XXX XXX XXX

Thus the printer allows characters to be a little less square than the TI forces us to be. Note however that the intermediate half column pins cannot be used as well as an adjacent normal column pin when using quadruple density which is ESC * 3 or double speed double density which is ESC * 2 ! (Thus if we use pin value 4 in column 1 we cannot use pin value 4 in column 2, but we can use pin value 2 or 8 in column 2.).

The "bog standard" Epson 8 bit graphic uses 480 dots along each line and uses the command ESC K... as you will see in the program which follows. Note that an exact equivalent to ESC K is ESC * 0- so instead of:

```
PRINT #1:CHR$(27);"K";.... you could use:
PRINT #1:CHR$(27);"*";CHR$(0);...
```

Generally you will be printing graphics using more than 80 columns of dots so to avoid an auto line feed we open the printer up by adding .CR to its name, and make our program sane the line feeds at the end of each graphic line.

Our program is using a graphic with a defined shape in a 6x6 grid, so we need to tell the computer to move

the line up 6 dots at a time (Line 110).

In line 140 we tell the printer to use the 8 pin bit image mode with 480 dots across, and that after each ESC K sequence we are going to send (218)+(16)=474 defined columns -after which we will tell it to move up a line (line 260). Note than by not using the full width, and by sending say 200 columns of dots, the printer then switched to text mode for the remainder of the line- we can send mixed text and graphics, or even use graphics mode for micro adjustment of character positions.

The program is based on a Truchet tile generation.

```
100 OPEN #1:"PIO.CR"
110 PRINT #1:CHR$(27);"A";CHR$(6);
120 REM lines are 6 dots high
130 RANDOMIZE
140 A$=CHR$(27)&"K"&CHR$(218)&CHR$(1)
150 REM sets up for 474 dot columns to follow.
160 [$=CHR$(8)&CHR$(16)&CHR$(32)&CHR$(65)&CHR$(2)&CHR$(4)
170 ]$=CHR$(8)&CHR$(4)&CHR$(2)&CHR$(65)&CHR$(32)&CHR$(16)
180 PRINT #1
190 PRINT #1:A$;
200 FOR WIDTH=1 TO 79
201 REM 6 x 79 = 474
210 IF RND<.5 THEN 240
220 PRINT #1:[$;
230 GOTO 250
240 PRINT #1:]$;
250 NEXT WIDTH
260 PRINT #1:CHR$(10);CHR$(13);
270 GOTO 190
280 END
```

Please note that this graphics program is written in pure TI BASIC and only requires a printer and appropriate TI->Printer interface to operate. Instead of ESC K try using instead ESC * 0 to ESC * 6 or any other ESC * modes your printer may support. The above program does not use adjacent horizontal dots and so all modes may be used.

The Epson range of printers can be found with "definable fonts" which allows you to redefine the character definitions held in the printer. This is a little more arduous. The TI Basic program below prints about the same pattern as the above program, but instead of using different ESC * modes you may use character modes such as condensed, expanded, Pica, Elite, and so on. You printer must support the commands ESC & as well as ESC %. On the FX80 printer it is necessary to switch OFF the on board print buffer, as this is used to store the new definitions. See your printer manual under "download characters".

```
100 OPEN #1:"PIO.CR"
110 PRINT #1:CHR$(27);"A";CHR$(6);
120 PRINT #1:CHR$(27);"p";CHR$(1);
121 REM proportional mode required to get correct
horizontal spacing for 6x6 graphic.
130 RANDOMIZE
140 A$=CHR$(27)&"&"&CHR$(0)
150 B$=CHR$(0)&CHR$(0)&CHR$(0)&CHR$(0)&CHR$(0)
160 [$=A$&"["&CHR$(6)&CHR$(8)&CHR$(16)&CHR$(32)&CHR$(65)&CHR$(2)&CHR$(4)&CHR$(8)&B$
170 ]$=A$&""]&CHR$(6)&CHR$(8)&CHR$(4)&CHR$(2)&CHR$(65)&CHR$(32)&CHR$(16)&CHR$(8)&B$
180 PRINT #1:[$&]";
181 REM redefines characters [ and ]
190 PRINT #1:CHR$(27);"%" ;CHR$(1);CHR$(0);
191 REM tells printer to use new definition instead of usual character
200 FOR WIDTH=1 TO 131
210 IF RND<.5 THEN 240
220 PRINT #1:"[";
230 GOTO 250
240 PRINT #1:"]";
250 NEXT WIDTH
260 PRINT #1:CHR$(10);CHR$(13);
270 GOTO 200
280 END
```

Have fun. Stephen Shaw 1990

Sorting part 3

by Ron Brubaker, USA

A very common application of sorting is the alphabetisation of names or of words. Alphabetic sorting in BASIC is particularly easy due to the fact that all of the characters utilised by BASIC have numeric equivalents. Thus, BASIC allows alphabetic characters or even strings to be used in comparisons (e.g. "A"<"B" is a valid comparison as is A\$<B\$ where A\$ = "ABC" and B\$ = "BCDE"). Due to this feature of the language it is very easy to modify a program that sorts numbers to sort strings.

Sorting Strings

The following program segment will be used to provide a list of names in the form of a one-dimensional string array.

```
10 REM *** READ IN A LIST OF RANDOMLY ORDERED NAMES ***
20 REM
30 DIM A$(100)
40 READ N
50 FOR I=1 TO N
60 READ A$(I)
70 PRINT A$(I)
80 NEXT I
90 DATA 15
100 DATA "WASHINGTON, GEORGE","JEFFERSON, TOM","FORD,
    GERALD"
110 DATA "KENNEDY, JOHN","FILMORE, MILARD","ARTHUR,
    CHESTER"
120 DATA "ADAMS, JOHN Q","LINCOLN, ABE","ROOSEVELT,
    FRANKLIN"
130 DATA "REAGAN, RONALD","CARTER, JIMMY","WILSON,
    WOODROW"
140 DATA "MONROE, JAMES","ROOSEVELT, THEODORE","ADAMS,
    JOHN"
150 END
```

The output from this program is the following:

```
WASHINGTON, GEORGE
JEFFERSON, TOM
FORD, GERALD
KENNEDY, JOHN
FILMORE, MILARD
ARTHUR, CHESTER
ADAMS, JOHN
LINCOLN, ABE
ROOSEVELT, FRANKLIN
REAGAN, RONALD
CARTER, JIMMY
WILSON, WOODROW
MONROE, JAMES
ROOSEVELT, THEODORE
ADAMS, JOHN
```

The sorting is very similar to the program shown at the end of Part 2.

```
10 REM *** READ IN A LIST OF RANDOMLY ORDERED NAMES ***
20 REM
30 DIM A$(100)
40 READ N
50 FOR I=1 TO N
60 READ A$(I)
70 PRINT A$(I)
80 NEXT I
90 DATA 15
100 DATA "WASHINGTON, GEORGE","JEFFERSON, TOM","FORD,
    GERALD"
110 DATA "KENNEDY, JOHN","FILMORE, MILARD","ARTHUR,
    CHESTER"
120 DATA "ADAMS, JOHN Q","LINCOLN, ABE","ROOSEVELT,
    FRANKLIN"
130 DATA "REAGAN, RONALD","CARTER, JIMMY","WILSON,
    WOODROW"
140 DATA "MONROE, JAMES","ROOSEVELT, THEODORE","ADAMS,
    JOHN"
160 REM ***** IMPROVED BUBBLE SORT *****
170 REM
```

```
180 J=1
190 Q=0
200 FOR I=1 TO N-J
210 IF A$(I)<=A$(I+1)THEN 260
220 Q=1
230 T$=A$(I)
240 A$(I)=A$(I+1)
250 A$(I+1)=T$
260 NEXT I
270 IF Q=0 THEN 310
280 J=J+1
290 GOTO 190
300 REM
310 REM **** ROUTINE TO PRINT SORTED LIST OF NAMES ****
320 REM
330 PRINT
340 FOR I=1 TO N
350 PRINT A$(I)
360 NEXT I
```

The changes consist of replacing all references to the array A with the array A\$ and replacing the variable T with T\$. The only other change is the removal of the trailing semi-colon in the print loop so that the names are listed on the separate lines.

The resulting list is:

```
ADAMS, JOHN
ADAMS, JOHN Q.
ARTHUR, CHESTER
CARTER, JIMMY
FILMORE, MILARD
FORD, GERALD
JEFFERSON, TOM
KENNEDY, JOHN
LINCOLN, ABE
MONROE, JAMES
REAGAN, RONALD
ROOSEVELT, FRANKLIN
ROOSEVELT, THEODORE
WASHINGTON, GEORGE
WILSON, WOODROW
```

Not a bad job for such a simple program!

Some Difficulties With String Sorts

Occasionally string sorts produce unexpected results. For example, although it is possible to sort numeric information that is in the form of strings, the results may not always be anticipated. For example, if the data in the above program is modified as follows:

```
110 DATA "2","9","7","57"
120 DATA "229","11","41","99","4"
130 DATA "89","199","50","220","1"
```

N.B. Lines 100 and 140 must be deleted and also line 70 if you want to eliminate the printing of the unsorted random numbers.

The results will be:

```
1 100 11 199 2 200 220 4 41 50 57 7 89 9 99
```

The difficulty may not be immediately apparent. However, if you note that the first digit is strictly in order and that the digits that follow are in order of the second digit, etc. the problem becomes apparent. Consider the following data statements:

```
100 DATA " 2"," 9","100"," 57"
120 DATA "220"," 11"," 41"," 99"," 4"
130 DATA " 89","199"," 50","200"," 1"
```

The resulting output is:

```
1 2 4 7 9 11 41 50 57 89 99 100 199 200 220
```

The lesson to be learned from the above is that string representations of numerics must be right justified to ensure that they are sorted properly. Either leading spaces or zeros may be used to accomplish

this. In the case of alpha strings it is normally necessary that the strings be left justified in order to ensure proper sorting. In the case of mixed alpha and numeric strings, which are common for computer file names, precautions should be taken to pad the numeric portions of the name with leading zeros (e.g. SORT01, SORT02, SORT10).

A second type of problem can arise if one attempts to sort mixed upper and lower case alphabetic characters. (Refer user handbook for the numeric equivalents of the ASCII character set). When BASIC is used to sort ASCII coded characters it does so on the basis of these numeric equivalents. Thus all upper case letters come before their lower case equivalents. Note also that the most common punctuation marks and all numbers come before the upper case alphabet. Care must be exercised to avoid situations where this inherent order of the string characters will dictate sorting in a different order than is desired.

The Use Of Pointers For Indirect Sorting

The sorting routines shown so far have actually rearranged the data in the data array variable. Although this can be the fastest method in the case of a simple numeric sort the time required to move lengthy strings about tends to slow down a direct string sort. In addition, it may not be desirable to actually rearrange the data. Since BASIC does permit string data and numeric data to be stored in the same array and there are some extra difficulties associated with sorting string representations of numbers, it is often more convenient to sort by an indirect method using pointers.

A pointer is simply a numeric array of the same length as the array that is to be sorted. It must be initialised by assigning each element a value to the index of the element:

(i.e. P(1)=1, P(2)=2, ----> P(N)=N)

This is accomplished in line 90 of the following program. In line 190 the numeric array to be sorted is referenced indirectly using P(I) in the place of the simple index loop index I. Remember, initially P(I)=I. However, in lines 200-230 which normally swap elements that are out of order, the values in the pointer array are swapped instead.

```

10 REM **** GENERATION OF A LIST OF RANDOM NUMBERS ****
20 REM
30 DIM A(100),P(100)
40 PRINT "HOW MANY NUMBERS DO YOU WANT?"
50 INPUT N
60 PRINT
70 FOR I=1 TO N
80 A(I)=INT(100*RND)+1
90 P(I)=I
100 PRINT A(I);
110 NEXT I
120 PRINT
130 REM
140 REM ***** BUBBLE SORT WITH POINTER *****
150 REM
160 J=1
170 Q=0
180 FOR I=1 TO N-J
190 IF A(P(I))<=A(P(I+1))THEN 240
200 Q=1
210 T=P(I)
220 P(I)=P(I+1)
230 P(I+1)=T
240 NEXT I
250 IF Q=0 THEN 290
260 J=J+1
270 GOTO 170
280 REM
290 REM ***** ROUTINE TO PRINT LIST USING POINTER *****
300 REM
310 PRINT
320 PRINT " I A(I) P(I) A(P(I))"

```

```

330 IMAGE ### ### ### ###
340 PRINT "=====
350 FOR I=1 TO N
360 PRINT USING 330:I,A(I),P(I),A(P(I))
370 NEXT I

```

The output routine was also modified to show the values of the index, the numeric array, and the sorted results. Note that the letter is achieved by referencing the numeric array using the pointer array.

HOW MANY NUMBERS DO YOU WANT? 15

64 48 2 79 36 5 66 71 100 24 14 67 57 13 34

I	A(I)	P(I)	A(P(I))
1	64	3	2
2	48	6	5
3	2	14	13
4	79	11	14
5	36	10	24
6	5	15	34
7	66	5	36
8	71	2	48
9	100	13	57
10	24	1	64
11	14	7	66
12	67	12	67
13	57	8	71
14	13	4	79
15	34	9	100

Examine the pointer array carefully. Note that the first element has a value of 3, the smallest number in the numeric array, and the last element has a value of 9, the largest number in the numeric array. Thus, the values found in the pointer array after sorting point to the values in the original array in the order that they must be printed to obtain a sorted list. Pretty neat, huh?

Next month we will look at how this type of sorting can be applied to string arrays. ○

continued from page 23

command files written for PI assume that both the program and data files share the same disk". There is no problem about storage space as the program files do not take up that much space. The problem is that the file that you are creating automatically is stored under the _DBASE1 file that I have mentioned elsewhere. So it becomes obvious that if you want to create a second entirely new database then as far as I have been able to work out you have to do one of the following:

1. Use a disk manager to transfer the first _DBASE1 records to another disk and change the name. If you want to use that database again you will have to rename it to _DBASE1 and put it on a disk containing the program files, or
2. Make several copies of the original disk up (one for each new index database). If anybody out there knows of an easier method then please enlighten me!

I am aware of the _DBASE2 XFER feature which allows you to transfer data from one file to another which could be "used for subfile creation, cleaning up the order of a file for faster data access, making the backup of the original file, or whatever other use you can think of." There is no provision for a _DBASE3, etc. This feature of XFER cannot be accessed from any menu in the PI system. It is accessed from the dot(.) prompt by typing DO XFER.

All in all I found it to be a useful piece of software for those who are making a Index of some sort. The only awkward part that I found was the storing of multiple (different) index databases. ○

Fractal Graphics

with The Missing Link (TML)

by Stephen Shaw, England

Here is a truly fractal program, which, using The Missing Link and Extended Basic ONLY takes about 12 hours or so to plot a graphic of 120 x 200 pixels!

The total graphic is vertically reflected about 0, so if 0 is at screen centre you can plot top and bottom parts at the same time to half plotting time...

This program is all about "attraction basins" which I do not understand, and while the program can probably be modified for other functions, I do not know how! What I know is that it plots a truly fractal shape which we can zoom in on.

```
100 ! TO DRAW AN ATTRACTION BASIN USING NEWTON
    FORMULA ON:
110 ! F(z)=z^(-3)-1
120 !
130 !
140 ! J C TOPHAM ! Fractal Report 13 ! Feb 1991
150 ! for ti+tml by S Shaw march 1991
200 WIDE=242 :: HEIGHT=190
210 !
220 !
230 CALL LINK("CLEAR")
239 ! actual pixels plotted:
240 ACROSS=200 :: DOWN=120
246 ! area to be plotted
247 ! whole image is within X=-2 to +2
248 ! Y= -1.75 to +1.75
249 ! Y is vertical and is reflected about 0.
250 XMIN=-1.1 :: XMAX=0.9
260 YMIN=-1.70 :: YMAX=0.30
270 !
280 !
290 RY(1)=-SQR(3)/2
300 RY(2)=SQR(3)/2
310 RY(3)=0
320 !
330 DX=(XMAX-XMIN)/ACROSS
340 DY=(YMAX-YMIN)/DOWN
350 XPOS=INT((WIDE-ACROSS)/2)
360 YPOS=INT((HEIGHT-DOWN)/2)
370 !
380 FOR YP=0 TO DOWN
390 FOR XP=0 TO ACROSS
400 CALL LINK("PRINT",181,12,STR$(YP)&" ":"&STR$(XP))
    410 YN=YMIN+YP*DY
    420 XN=XMIN+XP*DX
    430 FOR ITER=1 TO 30
    440 GOSUB 650
    450 XM=(A*C+B*D)/9
    460 YM=(B*C-A*D)/9
    470 IF XM*XM+YM*YM>1000 THEN 620
    480 !
    490 !
    500 IF ABS(YN-YM)>1E-2 THEN 590
    510 IF ABS(XN-XM)>1E-2 THEN 580
    520 FOR I=1 TO 3
    530 IF ABS(RY(I)-YM)>1E-2 THEN 560
    540 CALL LINK("PRINT",1,200,STR$(ITER)&" "): IF
        ITER/2<>INT(ITER/2) THEN ITER=30 :: GOTO 570
    550 CALL LINK("PIXEL",YPOS+DOWN-YP,XPOS+XP): ITER=300
    560 !
    570 NEXT I
    580 !
    590 !
    600 XN=XM :: YN=YM
    610 NEXT ITER
    620 NEXT XP
    630 NEXT YP
    640 END
    650 ! INVERT Z3
    655 X2=XN*XM :: X3=X2*XM :: X4=X2*X2 :: Y2=YN*YM ::
        Y3=Y2*YM :: Y4=Y2*Y2
    660 A=4*XN-X4+6*X2*Y2-Y4
    670 B=4*YN-4*X3*YM+4*XN*Y3
    680 C=3 :: D=0
    690 RETURN
```

MANDELBROT PLOT

This is EXCEEDINGLY SLOW if you want a detailed plot! However, a minimum detailed plot over a limited screen area is not TOO slow! Maximum detail and you could be looking at SEVERAL DAYS to complete a screenful.

THIS PROGRAM DOES WORK! and cries out for some machine code! Some means of saving the picture, to disk or printer, would be required- that is perhaps the hardest bit.

User should input area to be plotted, and possibly be given a choice of what method is to be used for deciding if a pixel is to be plotted or not- see comments in listing! - and the degree of inner detail required!

```
100 ! MANDELBROT PLOTTER
110 ! VERY SLOW!
120 ! FOR EX BAS + THE MISSING LINK
130 !
140 CALL LINK("CLEAR")
150 !
160 ! DESIGN LIES IN AREA
170 ! PMIN -2.25 PMAX 0.75
180 ! QMIN -1.50 QMAX +1.50
190 !
200 ! use smaller area for more detail- ensure
    there IS detail there though!
210 !
220 ! Concentrate on areas very close to central
    creature!
230 !
240 A=240 :: B=180 ! maximum values
250 !
260 ! A=width plotted B=height plotted in pixels
270 !
280 ! M= number of iterations per point. Needs to be
    enough to reach CMAX at centre of beast.
290 ! 100 is often used but lower values can usually be
    safely used.
300 !
310 !
320 A=160 ! A=width
330 B=150 ! B=height in pixels
340 M=71 ! MAX ITERATIONS
350 ! P=REAL Q=IMAG
360 PMIN=-.250 ! DEFINE AREA
370 PMAX=-.20 ! TO BE
380 QMIN=-0.83 ! DRAWN
390 QMAX=-0.79 !
400 !
410 ! CMAX must be a power of 2, eg 2,4,8,16,32,64,128..
420 ! The higher the value the more detail close to the
    beast.
430 ! 16 gives minimum acceptable detail, 32 is
    reasonable for an odd/even test, while 64 and
    higher give maybe too much detail.
440 ! If using 64 or higher you must use a logarithmic
    scale with greater gaps as K gets larger... eg
450 ! 64...pixel on. 53 to 63...pixel off
    K= 44 to 52...pixel on K= 36 to 43...pixel off
    K= 29 to 35...pixel on...
460 !
470 ! or use IF INT(K^.8)/2=INT(INT(K^.8)/2)
    THEN pixel on... or some other fractional power
480 ! note that processing time really does get longer
    as CMAX increases!!!!!!
490 !
500 !
510 CMAX=64 ! DETAIL, 8,16,32,64,128,256 ETC
520 !
530 DP=(PMAX-PMIN)/(A-1)
540 DQ=(QMAX-QMIN)/(B-1)
550 FOR NP=1 TO A
560 FOR NQ=1 TO B
570 P=PMIN+NP*DP :: Q=QMIN+NQ*DQ :: K,X,Y=0
580 ! LOOP
590 !
600 !
610 XN=X*X-Y*Y+P
620 Y=2*X*Y+Q
```

```

630 X=XN :: K=K+1
640 !
650 IF X*X+Y*Y>M THEN 690
660 IF K=CMAK THEN 690
670 GOTO 580
680 !
690 IF K/2=INT(K/2)THEN CALL LINK("PIXEL",NQ,NP)! SEE
    COMMENTS ON CMAK ABOVE!
700 !
710 CALL LINK("PRINT",170,200,STR$(NQ)&STR$(K)&" ")
720 NEXT NQ
730 NEXT NP
740 CALL LINK("SAVEP","DSK2.PIC2")
750 GOTO 750

```

THE GOLDEN RATIO

```

1 ! THE GOLDEN RATIO - 1:1.618...
2 ! removing a square from a rectangle with sides in
    this ratio leaves a smaller rectangle which also has
    sides in the Golden Ratio
3 ! -points dividing sides lie on a logarithmic spiral
    which can be found in shells, and in art works by
    da Vinci, Dali- even in
4 ! the Parthenon.
5 ! Fractal in nature- it keeps getting smaller or
    larger depending on which way you go...
6 ! program written by Ashley Tilling for JBM103
    converted by S Shaw for TML.
7 !
8 ! Due to our not having square pixels the rectangle
    on screen is NOT in the golden ratio as viewed.
    Ah well...
9 !
99 ! remove ! from line 100 for use with JBM103.
100 ! call load(-31890,56,0) :: CALL LOAD(-31964,56,0)
110 AX=12 :: AY=8
120 CL=200 ! side length
130 CALL LINK("CLEAR")!for jbm103 CALL LINK("SCR2") also.
140 RL=INT(CL/1.618)
150 FOR I=0 TO 1 :: FOR J=0 TO 1
160 CALL LINK("LINE",AX+RL*I,AY+CL*I,AX+RL*J,AY-CL*(J=0))
162 ! CALL LINK("LIGNE",16,AX+RL*I,AY+CL*I,AX+RL*J,AY-
    CL*(J=0))
170 NEXT J :: NEXT I
180 FOR K=1 TO 3
190 M=CL-RL :: N=INT(CL*.236):: P=INT(CL*.146)
195 CALL LINE(AX,AY,AX,AY)
200 CALL LINE(RL,M,1,M)
210 CALL LINE(N,1,N,M)
220 CALL LINE(1,N,N,N)
230 CALL LINE(P,M,P,N)
240 AX=AX+P :: AY=AY+N :: CL=P :: RL=INT(CL/1.618)
250 NEXT K
260 GOTO 260
270 SUB LINE(A,B,C,D)
280 REM ! LINES for JBM103 INSTEAD of CALL LINK(
    "LINE".. and CALL LINK("PIXEL"...
290 IF (A=C)*(B=D)THEN AX=A :: AY=B :: SUBEXIT
300 CALL LINK("LINE",AX+A,AY+B,AX+C,AY+D)
310 ! CALL LINK("LIGNE",16,AX+A,AY+B,AX+X,AY+D)
320 IF B=D THEN 350
330 L=ABS(B-D)-1
340 ST=-PI*(B>D):: FI=PI/2+PI*(D>B):: GOTO 370
350 L=ABS(A-C)
360 ST=-PI/2-PI*(A>C):: FI=PI*(C>A)
370 IF L<3 THEN SUBEXIT
380 FOR J=ST TO FI STEP -1/L
390 Y=AY+D+INT(.5+L*SIN(J))
400 X=AX+C+INT(.5+L*COS(J))
410 CALL LINK("PIXEL",X,Y)
420 ! CALL LINK("POINT",16,X,Y)
430 NEXT J
440 SUBEND
450 END

```

POLYNOMIAL EQUATIONS...

```

1 ! Newtons method for solving polynomial equations.
2 ! eg find z when 4Z^6-2Z^3+z-1=0
3 ! sorry about the input format, you just need to enter
    the multiplicand for ea ch power of z and its sign
    if negative (6z or -6z)

```

```

4 ! enter the default zero if that power is not in the
    equation.
5 ! one numeric answer will usually be given but some
    formulae may have more than one answer- try varying
    the seed
6 ! in line 290, value of X
7 ! to produce different answers.
8 ! a few equations MAY not be sovable with this
    program.
9 ! amend input method if using multipliers over +99 or
    under -9 or if you wish to use higher powers (remember
    to DIM the array).
10 !
11 !
100 CALL CLEAR :: V$="0123456789-+"
110 DISPLAY AT(1,1):"NEWTONS METHOD TO SOLVE Polynomial
    Equations"
120 DISPLAY AT(3,4):" Dr M Ecker 1987.S Shaw for
    TI 1991"
130 DISPLAY AT(6,1):"Your equation is in the form 5Z^5
    +0Z^4-2Z^3 +0Z^2-2Z+9=0"
140 DISPLAY AT(12,1):" 0 Z^9 0 Z^8 0 Z^7 0 Z^6"
150 DISPLAY AT(13,1):" 0 Z^5 0 Z^4 0 Z^3 0 Z^2"
160 DISPLAY AT(14,1):" 0 Z 0 = 0"
170 ACCEPT AT(12,2)SIZE(-2)VALIDATE(V$):A(9)
180 ACCEPT AT(12,8)SIZE(-2)VALIDATE(V$):A(8)
190 ACCEPT AT(12,14)SIZE(-2)VALIDATE(V$):A(7)
200 ACCEPT AT(12,20)SIZE(-2)VALIDATE(V$):A(6)
210 ACCEPT AT(13,2)SIZE(-2)VALIDATE(V$):A(5)
220 ACCEPT AT(13,8)SIZE(-2)VALIDATE(V$):A(4)
230 ACCEPT AT(13,14)SIZE(-2)VALIDATE(V$):A(3)
240 ACCEPT AT(13,20)SIZE(-2)VALIDATE(V$):A(2)
250 ACCEPT AT(14,2)SIZE(-2)VALIDATE(V$):A(1)
260 ACCEPT AT(14,6)SIZE(-2)VALIDATE(V$):A(0)
270 FOR T=0 TO 9 :: IF A(T)>0 THEN N=T
280 NEXT T
290 TLD,TL=.00000000001 :: X=.800 ! INITIAL GUESS
300 FOR K=0 TO N
310 Y=Y+A(K)*X^K
320 NEXT K
330 IF ABS(Y-0)<TL THEN 410
340 FOR K=1 TO N
350 YD=Y+K*A(K)*X^(K-1)
360 NEXT K
370 IF YD=0 THEN X=X+.01 :: GOTO 400
380 X=X-Y/YD
390 DISPLAY AT(22,1):"Next iterate:":X; " "
400 Y,YD=0 :: GOTO 300
410 DISPLAY AT(20,1):"***DONE***":<any key for
    another>"
420 DISPLAY AT(16,3):"Z=";X
430 CALL HCHAR(18,6,42,6)
440 CALL KEY(5,P,B)
450 IF B=1 THEN RUN
460 CALL HCHAR(18,6,32,6)
470 CALL HCHAR(18,6,45,6)
480 GOTO 430
491 ! Recreational & Educational Computing
492 ! 909 Violet Terrace
493 ! Clarks Summit PA
494 ! USA 18411
495 ! 1991 sub Europe US$36 or
496 ! send $12 up for sample copies
500 END

```

I gather this method is best for orders up to a power of 6 but better methods are available for higher orders... so I am told...

continued on page 12

First, let's talk about why you want to control this line for anything else. What about using the computer to control other devices? With a bit of circuitry and the following code, a program could be written to turn lights on and off or maybe a stereo. Your TI could be at work while you are away.

The following set-up-a-link program will allow user control of the cassette remote control for CSI. By doing so, an Extended Basic program can be used to control external items other than the cassette recorder. This program has to be loaded and run in Extended Basic with the 32K memory expansion operating.

continued on page 4

Publications Index

for TI-Base, version 1.0
by Bill Gaskill, USA, reviewed by Bob Relyea

Last year, I ordered through the post via my mother who lives in America, a copy of TI Publications Index as I make good use of the TI-Base software and I wanted to have all of the supporting software that I could find. The price for the software at the time was \$14.95 US and my mother found their service to be prompt (hear that Larry Saunders?). Textaments is located in the town of Patchogue on Long Island, New York State, USA. I remember visiting this place back when I was at university as I was a member of the University Men's Glee Club and we were on Winter tour. The lower part of NYS and Long Island was the itinerary for the tour that year. Most of you who know me would realise that this would surely have been before the days of the TI- and you are right! The little secret is going to be that I am not going to tell you how many years before.

Anyway, I have had a chance to use the software enough to be able to give you an idea of its worth. You must have TI-Base in order to make use of the Publications Index (hereinafter called PI). The package comes with an explanation booklet and a single disk (SS/SD). If you have a double disk drive set-up then you can make your Datdisk DSK2 and put the PI disk into drive 2 to eliminate swapping disks all the time. It is very easy to load, just boot up TIB and at the first dot prompt after the date type in DO MENU and it loads up for you. The software is designed to allow a person to enter information about various publications to enable an 'index' to be published or printed out. For those of you familiar with the subject and author index that is produced each year by our club (by Geoff) then you will know what I am talking about. Upon loading the PI the main menu (Systems Menu) appears in the form given below:

	Date
A	to Add new data
C	Change a record
F	Find a record
H	Help screens
K	Key search
L	List all records
P	Print a record
R	Print a report
U	Utilities Menu
X	eXit the program

The operation of most of the above is obvious and there is also a 'status line' at the very bottom of the screen in inverse video which shows, among other things, the file that is activated.

To familiarise yourself with the software and what it can do PI has a DEMO file that can be loaded by entering the > at the System Menu prompt. You can reactivate the main PI access file by entering <. Before we go any further with this let me explain what types of information can be entered into the PI main access file (called _DBASE1). The file structure built into the software is the following:

SUBJECT: stores the topic or title of an article.
SOURCE: the name of the book, magazine, etc.
TYPE: i.e. tutorial, review, etc.
DATE: any format can be used here.
PAGE: stores the page number of the article.
AUTHOR: reverse format recommended, such as Harrison,Percy with no spaces.
KEY: stores a two character numeric identifier that describes the article in a broad sense, such as,

O1 Article
O2 Editorial, etc

There are seven key numbers in all.
IDX: this is a system for an easier creation of subfiles of data from the main _DBASE1 file in alphabetic sections.

Now, this same structure could be created by somebody using the normal TIB software but a lot of the features of the PI, such as the 'custom' input screen would not be readily available, of course.

To help myself get familiar with the PI I loaded the Demo file and started making my way around. Let's have a look at some of the features from the main menu. Find(ing) a record and Print(ing) a record are handled practically the same way. In both cases you are asked to enter up to 8 characters of the record from any one of Subject, Type, Date, Author, etc so the PI can find it and do the rest. The only error I found here is that it asks you to enter up to 8 characters of one of those fields. This is misleading because if you do not enter at least 8 it will not Find or Print the record. Entering any more than 8 will be of no benefit. So, if you ask the PI Demo file to Find all the files with the Author COX,GARY (notice- no spaces) then it will Find all the records with that author and output it to screen. Doing an identical thing with Print will output it to the printer. A little programming error that I found was in the case of Print record, with the cursor being placed in an incorrect position- right on the last 'R' of ENTER. It does not affect the operation of the subroutine, however, it just looks silly.

If you choose 'U' from the main menu to get the Utility menu you will quickly find the following menu appear:

	Date
B	for Bottom of file
C	Catalog a disk
D	Display structure
F	File record delete
H	Help screens
M	Modify set-up
P	Print command files
R	Remove deleted data
S	Sort the file
T	Top of the file
X	eXit the program

From here you are able to observe and make your way around your database and manipulate it in various ways, such as with the Sort routine. Changes can be made in the structure and disks can be catalogued. Quite a handy little set-up. If you go for the disk catalogue you get the following information:

PUBINDEX	Total=	Free=	
NAME	TYPE	SECTORS	REC-SIZE

There is provision for cataloguing disk drives up to DSK4 as well as WDS1.TIB & WDS1.PI if you are using a hard drive.

As I started to mention before, when you choose the APPEND MODE to add new data (which is also used to start a new data base (i.e. _DBASE1) then you are given a choice of screens:

1. a Custom Input Screen, or
2. the Standard TIB append screen

Either one is as useful as the other, so I feel it depends on what you like to look at. I prefer, for aesthetic purposes, the Custom screen which appears to be set out a little more professionally.

The only awkward thing that I have so far discovered about the PI software is the storing of information. As it states in the literature, "The

continued on page 20

XB tips Number 14

by Jim Swedlow, CA USA

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

PRODUCT REVIEW: TI-WRITER INSTRUCTIONS AND HINTS

This is crammed with ideas, hints, suggestions and encouragement. You should print it thru both the Text Formatter and the Text Editor. That way you can see how things were done. Like, for example, printing in compressed type with a line width of 130 characters.

I would only quibble with one suggestion: that you start your file on line 0001. I leave that line blank as it is lost if you do a Recover Edit.

While it is aimed at the beginner, I would rate this as quite valuable to any TI-Writer user. This program is released as FREeware. If you find it useful, send a few dollars to the author, Dick Altman. He earned it as this must have taken quite a bit of time and effort to write.

TOKENS AND LINE CONTENTS

In the October, 1985 ROM, I covered the line number table. Briefly (if you missed it), it is a list, in RAM, of a program's line numbers and the memory locations of the line contents.

This month, we will look at how the line contents are stored. Lets take a fairly simple line:

```
IF I=2 OR A=B THEN PRINT "HI" :: GOTO 120
```

If you dumped this from RAM, it would look like this (the numbers are in HEX):

```
16 84 49 BE C8 01 32 BA 41 BE 42 B0
9C C7 02 48 49 82 86 C9 00 78 00
```

What does this mean? The translation:

HEX Dec Meaning

```
16 *22 This line has 22 bytes
84 132 Token for IF
49 73 ASCII for I
```

(For the rest I will omit 'Token for' and 'ASCII for')

```
BE 190 =
C8 200 flag, unquoted value next
01 1 value is one byte long
32 50 2
BA 186 OR
41 65 A
BE 190 =
42 66 B
B0 176 THEN
9C 156 PRINT
C7 199 flag, quoted value next
02 2 value is two bytes long
48 72 H
49 73 I
82 130 :: (line separator)
86 134 GOTO
C9 201 flag, line number next
00 0 High byte of line #
78 120 Low byte of line #
00 0 End of Line marker
```

Tokens do not get you on a bus but do store TI Basic and XB commands as one byte each. See the May, 1985 ROM for a complete list of tokens and ASCII values for our TI's. This same system is used when a program is saved on disk.

DISK*LABEL

In the September, 1985 ROM, I published a utility that would scan a disk's directory and then print the contents on a 1x3 sticky label. The commands were specific to the GEMINI 10X. Several of you had problems running it on other printers.

This month's program is a revision of DISK*LABEL. The printer commands are now at the beginning of the program with full annotation (see lines 180 to 230). You can easily make it work on any printer that prints superscripts.

I have made a number of changes in the program. It should be harder to crash now. Also, it will always read the first 23 files on a disk (even if there are not that many files on the disk). This avoids a problem some protection schemes cause. For example, this version works with Advanced Diagnostics.

I recommend that you NOT put the label directly on the disk. I have tried two different types, and both tend to peel. Very bad news if one comes off inside your drive!! Put the labels you got with your disks on the disk and these on the sleeve.

Enjoy.

```
100 ! DISK*LABEL
110 ! VERSION XB.2.1
120 ! BY JIM SWEDLOW
130 ! BASED ON A PROGRAM BY P.C.B. AND W.A.R.
140 ! 09 NOV 85
150 !
160 B,@=1 :: CALL CLEAR :: DIM F$(144),T$(144),Y$(4)::
FOR I=0 TO 14 :: CALL COLOR(I,16,@):: NEXT I :: CALL
SCREEN(5)
170 GOTO 180 :: A,C,D,J,C$,D$,E$,DS$,CN$,SS$,LF$,IN$ ::
!@P-
180 E$=CHR$(27)! ESCAPE
190 DS$=E$&"G" ! DOUBLE STRIKE
200 CN$=CHR$(15)! CONDENSED
210 SS$=E$&"S"&CHR$(0)! SUPERSCRIPT
220 LF$=E$&"3"&CHR$(12)! CHANGE LINE FEED TO 12/144 INCH
230 IN$=E$&"@" ! INITIALIZE PRINTER
240 Y$(@)="DF" :: Y$(2)="DV" :: Y$(3)="IF" :: Y$(4)="IV"
:: D$="DSK1." :: OPEN #2:"PIO"
250 DISPLAY AT(7,10):"DISK*LABEL": :: "Check the
position of the labels before starting."
260 DISPLAY AT(14,6):"Labels/Disk: 1": " Drive:
DSK1": " Comment:"
270 DISPLAY AT(20,@):"ENTER <P>rint <C>hange or": "
<Q>uit
P"
280 ACCEPT AT(21,20)VALIDATE("CQPcpq")SIZE(-@)BEEP:E$ ::
IF E$="" THEN 280 ELSE I=ASC(E$):: E$=CHR$(I+32*(I>81))
290 IF E$="Q" THEN CALL CLEAR :: CLOSE #2 :: STOP ELSE I
F E$="P" THEN 330 ELSE DISPLAY AT(20,@) ::
300 ACCEPT AT(14,20)SIZE(-2)VALIDATE(DIGIT)BEEP:E$ :: IF
E$="" THEN 300 ELSE B=MAX(VAL(E$),@)
310 ACCEPT AT(15,20)SIZE(-@)VALIDATE("12")BEEP:E$ :: IF
E$="" THEN 310 ELSE D$="DSK"&E$&"."
320 ACCEPT AT(17,6)BEEP:C$ :: GOTO 270
330 C=0 :: DISPLAY AT(20,@): "Initializing": :: OPEN #@
:D$,INPUT,RELATIVE,INTERNAL :: INPUT #@:F$(C),I,I,I ::
T$(C)="FREE "&STR$(I)
340 DISPLAY AT(22,@):F$(C);"";T$(C):: IF C=127 THEN 37
0 ELSE INPUT #@:F$(C+@),I,J,J
350 IF F$(C+@)="" THEN IF C>23 THEN 370 ELSE C=C+@ :: GO
TO 340
360 I=ABS(I):: C=C+@ :: IF I=5 THEN T$(C)="Prog" :: GOTO
340 ELSE T$(C)=Y$(I)&STR$(J):: GOTO 340
370 CLOSE #@ :: DISPLAY AT(20,@):"Printing": :: FOR
A=@ TO B :: J=0 :: D=8
380 E$="" :: PRINT #2:DS$;CN$;F$(0);E$;C$;E$;T$(0);SS
$;LF$:
390 FOR I=J+@ TO J+D :: PRINT #2:F$(I);TAB(12);T$(I);TAB
(18);F$(I+D);TAB(29);T$(I+D);TAB(35);F$(I+2*D);TAB(46);T
$(I+2*D):: NEXT I
400 J=J+24-6*(D=10):: IF C>J THEN D=10 :: PRINT #2: ::
:: GOTO 390 ELSE PRINT #2:IN$
410 NEXT A :: FOR A=@ TO C :: T$(A),F$(A)="" :: NEXT A :
: GOTO 270
```

Link-It #21

by Ross Mudie

This linked assembly program partially emulates CALL KEY whilst providing a cursor and beep. The program is accessed by CALL LINK("KEY",Row,Column,ReturnVar\$) in your Extended Basic program. The program, as included below, accepts upper case and returns only strings of one character. If a numeric result is required, it is easily proved to be numeric and converted to numeric in your Extended Basic program. The program will pick up a character from the cursor position on the screen and if just the ENTER key is pressed then the character from the screen position is returned to the Extended Basic program.

The assembly program is extensively documented, thus a lengthy description is not warranted.

A short Extended Basic program is included to try out the assembly program.

```

100 ! SAVE DSK1.LOAD
110 DISPLAY AT(10,1)ERASE ALL:"LOADING ASSEMBLY"
120 CALL INIT
130 CALL LOAD("DSK1.KEY")
140 CALL LINK("KEY",20,4,K$)
150 DISPLAY AT(22,4):K$
155 FOR D=1 TO 200 :: NEXT D
160 GOTO 140

* Source file = KEYS      Object file = KEY

      IDT 'KEYmudie'
      DEF KEY

NUMREF EQU >200C      Assembly subprogram entry points
STRASG EQU >2010
KSCAN EQU >201C
XMLLNK EQU >2018
VSBW EQU >2020
VSBR EQU >2028
ERR EQU >2034

FAC EQU >834A          System Equates
SOUND EQU >8400

CFI EQU >12B8          XMLLNK Equate

SAVRTN BSS 2           Memory reservations
WS BSS >20

B3 BYTE 3             Byte size constant assignments
ENTER BYTE 13
FF BYTE >FF

CURBUF BSS 1          Place to store character off screen
CSRTIM DATA 300      Cursor Timer
CSFLAG DATA 0        Cursor Flag
CSRDC DATA 2         Cursor down counter

ONE DATA 1           Word size constant assignments
ROWMAX DATA 24
COLMAX DATA 28

BUFFER DATA >0100    Buffer for STRASG

BEEPT DATA >8005     Sound data for 1400 Hz TGI
BEEPLE DATA 200      Beep length
BEEPDC DATA 0         Beep down counter
BEEPON BYTE >92       TGI on
BEEPOF BYTE >9F       TGI off

ERRBV LI RO,>1E00     Bad value      Error handling
BLWP @ERR

SUBRTN BLWP @NUMREF   Get numeric value from x/b
BLWP @XMLLNK
DATA CFI             Convert floating point to Integer
C @FAC,@ONE         Is value less than 1?
JLT ERBV
RT

```

```

* CALL LINK("KEY",Row,Column,Return_String$)

KEY  MOV R11,@SAVRTN      Save the return address
     LWPI WS              Load register workspace address

     CLR RO                Element zero
     LI R1,2              Argument 2 Column
     BL @SUBRTN           Get column value
     C @FAC,@COLMAX      Is the column value too big?
     JGT ERBV             If yes, give error
     INC @FAC             Add 2 for 28 columns then take
                         * 1 off for first column zero.
                         Put column in R4

     MOV @FAC,R4

     DEC R1                Argument 1 Row
     BL @SUBRTN           Get row value
     C @FAC,@ROWMAX      Is the row value too big?
     JGT ERBV             If yes, give error
     MOV @FAC,RO          Put row value in RO
     DEC RO               Subtract 1 for zero based values
     SLA RO,5             Multiply by 32 (32 chars per row)
     A R4,RO              Add columns value to rows value
                         * (RO gives VSBW & VSBR position on screen)

     BLWP @VSBR           Read character off screen
     MOV R1,@CURBUF      Store it in Cursor buffer

     MOV @ONE,@CSRDC     Value in cursor loop counter
     CLR @CSFLAG         To ensure cursor on early

BEEP MOV @BEEPLE,@BEEPDC Routine to start BEEP
     MOV @BEEPT,@SOUND   First byte to sound chip
     SWPB @BEEPT         Swap bytes at sound data
     MOV @BEEPT,@SOUND   Second byte to sound chip
     SWPB @BEEPT         Swap sound data back to normal
     MOV @BEEPON,@SOUND  Turn sound chip on

SLOOPI MOV @BEEPDC,@BEEPDC Is beep finished yet?
     JEQ BEPEND          If yes, then leave routine
     DEC @BEEPDC         Count down beep counter
     C @BEEPDC,@ONE     When down counter=1 turn off
     JNE BEPEND         Jump, except when ending sound
     MOV @BEEPOF,@SOUND Turn sound off

BEPEND

CURSOR DEC @CSRDC        Decrement Down counter
     JNE CSREND          Jump end if not zero
     MOV @CSFLAG,@CSFLAG Test Cursor flag
     JEQ CSRON          If zero, JUMP to turn the cursor on

CHREST CLR @CSFLAG       Make flag 0
     MOV @CURBUF,R1      Get saved screen character
     BLWP @VSBW          Write char back to screen
     JMP CSREN

CSRON MOV @ONE,@CSFLAG   Set cursor flag
     LI R1,>7E00         Cursor with >60 added
     BLWP @VSBW          Put cursor on screen
CSREN MOV @CSRTIM,@CSRDC Timing value in down counter
CSREND

     MOV @B3,@>8374      Instruct KSCAN, UPPER CASE
     BLWP @KSCAN         Scan keyboard
     CB @>8375,@FF       Test if any key pressed
     JEQ SLOOPI         Loop around if no key pressed

     CB @>8375,@ENTER   Test if key pressed is ENTER
     JNE SHOKEY         Jump if not ENTER
     MOV @CURBUF,R1     If ENTER, replace the byte 13
     AI R1,->6000       with the character which was
     MOV R1,@>8375      stored in CURBUF without hex 60

SHOKEY MOV @>8375,R1     Take the byte from the key press
     AI R1,>6000         Add hex 60 to the left byte in R1
     BLWP @VSBW         Write character on the x/b screen

     MOV @>8375,@BUFFER+1 Put key char in BUFFER
     CLR RO              Element zero for x/b variable
     LI R1,3             Third argument in x/b link
     LI R2,BUFFER        Point to buffer for STRASG
     BLWP @STRASG        Send the key press string to x/b

```

continued on page 2

Screen Dump Routine

by J.Peter Hoddie, for the Boston Computer Society

This routine is based on a routine by John Clulow from his New Horizons User Group.

This routine will produce a full graphics dump to an Epson printer in under one minute. The dump is done in double density graphics mode and so produces a very clear image. The routine is set to run in Extended BASIC but can be easily modified to run in other environments.

To load the program type "CALL INIT :: CALL LOAD("DSK1.DUMPJPH)". The program is Re-locatable so if you have another assembly language program in memory you will not lose it if you load the DUMP routine.

The routine comes set up to dump to the "PIO" device but if you have your printer hooked up to a different port this device name may be modified as follows. Type "CALL LINK("INIT","RS232.BA=4800.CR")" or whatever your device name may be. However the device name must end in ".CR" or the DUMP routine will lock up.

To dump a screen you must execute the statement "CALL LINK("DUMP")" and when the routine is done executing it will return you to your program.

There is a second way to obtain a screen dump but it is rather messy in that after the screen dump is complete you must turn your computer off and on again to continue. It has one major advantage: It will dump the screen at the push of button. This is accomplished by using the user directed ISR. To initialize this mode type "CALL LINK("START")". To dump a screen now hold down Function 5 and the screen will be dumped unfortunately once the routine is done your computer will lock. I am still working on this problem but for now this is the best I can do.

Finally: If you go over the code to this program you will find that it is rather inefficient and could probably be speed up quite a bit more, I leave that for you to work on and would be interested to see the results. And as usual this DUMP routine does not do sprites . . . has anyone seen one that does? It is possible !

* * *

```
** Block 1 **
32000 OPEN #1:"PIO",OUTPUT,VARIABLE 255 :: PRINT #1:CHR$(
(27);"A";CHR$(8)
32001 Y1=0
** Block 2 **
32002 Y1=Y1+1 :: Q$="" :: T$="" :: S$=""
32003 FOR Z=3 TO 29 :: CALL GCHAR(Y1,Z,G):: Q$=Q$&CHR$(
(G):: NEXT Z
32004 FOR Z=1 TO LEN(Q$)
32005 G=ASC(SEG$(Q$,Z,1))
32006 CALL CHARPAT(G,A$)
32007 CALL DUMP(A$,S$)
32008 T$=T$&S$
32009 NEXT Z
32010 PRINT #1:CHR$(27);"K";CHR$(LEN(Q$)*8);CHR$(0);T$
32011 GOTO 32002
** Block 3 **
32012 SUB DUMP(C$,S$)
32013 IF C$=RPT$("0",16)THEN S$=RPT$(CHR$(0),8)::
SUBEXIT
32014 S$=""
32015 B$="000000010010001101000101011001111000100110101
0111100110111101111"
32016 H$="0123456789ABCDEF"
** Block 4 **
32017 FOR X=1 TO 8 :: FOR Y=1 TO 2
32018 P=POS(H$,SEG$(C$, (X-1)*2+Y,1),1)
32019 Z$=SEG$(B$, (P-1)*4+1,4)
32020 FOR Z=1 TO 4 :: G(X,(Y-1)*4+Z)=ASC(SEG$(Z$,Z,1))
-48
32021 NEXT Z
32022 NEXT Y :: NEXT X
```

```
** Block 5 **
32023 FOR Y=1 TO 8 :: T=0 :: FOR X=1 TO 8
32024 T=G(X,Y)*2 (8-X)+T
32025 NEXT X :: S$=S$&CHR$(T):: NEXT Y
32026 SUBEND
```

Block 1: Initialization. Opens printer for output and sets up printer to use 8/72 of an inch line spacing.

Block 2: Screen scan. Line 32003 puts the horizontal screen line (Y1) into the string Q\$. 32004 to 32009 then convert this to output for the printer by getting the character codes for each character and passing them to the subprogram DUMP. Line 32010 outputs the line of graphics data to the printer with the ESCAPE K sequence followed by two bytes describing the length of the data.

BLOCK 3: Initialization of the DUMP subprogram. C\$ is the character pattern passed to DUMP and S\$ is the printer data returned by DUMP. Line 32013 checks to see if the character is a space. If it is then DUMP gives it the printer code immediately and returns to save time. Line 32015 contains the Binary number system from 0 to 15 stored in a string. Line 32016 is the Hexadecimal system from 0 to 15.

Block 4: These lines unpack the character pattern (C\$) into the array G(,) as a series of ones and zeros.

Block 5: These lines rotate the array formed in Block 4 into data acceptable by the printer and pack the data back into a string (S\$) to be returned by DUMP.

To use this program, simply MERGE it into your program and add the statement "GOSUB 32000" when you want a screen dump. The problem is that the program will take about 20 minutes to run so be prepared to wait.

```
DEF DUMP,START,INIT
*
NUMREF EQU >200C {
STRREF EQU >2014 {
ERR EQU >2034 {
ERCODE EQU >1E1C {
WS BSS 32
S1 BSS 2
IN BSS 8
DO BSS 512
LINE DATA 0 JPH DATA HOLDER
SAVRTN DATA 0
MK DATA >001F
PD DATA >0012,>1E00,>FF00,>0000
SP DATA >0006
TEXT 'PIO.CR'
EVEN
CR DATA >0D0A
E1 DATA >1B4C,>0002
E2 DATA >001B,>4108
D6 DATA >4019
*****
* GET PRINTER SPEC
*
INIT MOV R11,@SAVRTN
LI RO,>0017
MOV RO,@SP 23 BYTES MAXIMUM
CLR RO
LI R1,1 SECOND PARAMETER
LI R2,SP+1
BLWP @STRREF GET PRINTER SPEC.
MOV @SAVRTN,R11
RT
START
LI R1,INTER
MOV R1,@>83C4
RT
INTER
MOV R11,@SAVRTN
LIMI 0
LWPI WS
```

```

LI RO,>0E00
CB RO,>8375
JNE EXIT

SETO RO
MOVB RO,>8375 CLEAR BYTE OF KEY PRESSED
JMP DUMPO

EXIT LWPI >83E0
MOV @SAVRTN,R11
RT

*****

QUIT B @CLOSE

*****

DUMP MOV R11,@SAVRTN
LIMI 0

DUMPO LWPI WS

CLR R8
CLR R9 STARTING SCREEN POSITION
CLR R0
LI R9,-1 JPH WAS HERE!
MOV R9,@LINE

*
* SET UP PAB
*
NEXTL LI RO,>1D00
LI R1,PD
MOV @SP,R2
AI R2,10 NO OF PAB BYTES TO MOVE
BLWP @VMBW WRITE PAB TO VDP RAM
LI R6,>1D09 POINT TO DEVICE NAME LENGTH
MOV R6,>8356 DSRLNK TO OPEN PRINTER
BLWP @DSRLNK
DATA 8
JNE GO
MOVB @S1,>9C02
SWPB @S1
MOVB @S1,>9C02
LI RO,ERCODE
SWPB RO
BLWP @ERR
JNE GO
BLWP @ER CHECK FOR PRINTER SPEC ERROR
RETURN I/O ERROR

GO INC @LINE
MOV @LINE,R9
CI R9,24 JPH WAS HERE
JEQ QUIT
CLR R8
CLR RO

SLA R9,5
MOVB @>9802,@S1
SWPB @S1
MOVB @>9802,@S1
SWPB @S1
DEC @S1

LI RO,>1D00
LI R1,>0300
BLWP @VSBW PUT WRITE OP CODE IN PAB
LI RO,>1D05
LI R1,>0400
BLWP @VSBW PUT LENGTH OF 4 IN PAB
LI RO,>1E00
LI R1,E2 PUT CODE FOR CARRIAGE RTN &
LI R2,4 8/72" VERTICAL LINE SPACING
BLWP @VMBW IN DATA BUFFER.
MOV R6,>8356 POINT TO DEVICE NAME LENGTH
BLWP @DSRLNK DSRLNK-CHANGE VERT SPACING
DATA 8
LO MOV R9,RO
BLWP @VSBW PUT BYTE OF SCREEN RAM IN R1
SRL R1,8 SHIFT TO LSB OF R1
AI R1,-128 ADJUST FOR BASIC
SLA R1,3 *8
AI R1,1024 PTRN ADDR=1024+(CHAR#-32)*8

MOV R1,RO
LI R1,IN
LI R2,8
BLWP @VMBR
LI R5,128
LI R6,128
CLR R3
CLR R4
CLR R7
MOVB @IN(3),R7
SWPB R7
C R7,R5
JLT LI
A R6,R4
S R5,R7
SWPB R7
MOVB R7,@IN(3)
INC R3
SRA R6,1
JGT L2
SWPB R4
MOVB R4,@DO(8)
INC R8
MOVB R4,@DO(8)
INC R8
SRA R5,1
JGT L3
INC R9
CZC @MK,R9
JNE L0
LI R3,4
LI RO,>1D05
LI R1,>0400
BLWP @VSBW
LI RO,>1E00
LI R1,E1
LI R2,4
BLWP @VMBW
LI R6,>1D09
MOV R6,>8356
BLWP @DSRLNK
DATA 8
LI R4,DO
LI R2,128
MOV R4,R1
LI RO,>1E00
BLWP @VMBW
LI RO,>1D05
LI R1,>8000
BLWP @VSBW
MOV R6,>8356
BLWP @DSRLNK
DATA 8
AI R4,128
DEC R3
JNE L5
LI RO,>1D05
LI R1,>0200
BLWP @VSBW
LI RO,>1E00
LI R1,CR
LI R2,2
BLWP @VMBW
MOV R6,>8356
BLWP @DSRLNK
DATA 8
LI RO,>1D00
LI R1,>0100
BLWP @VSBW
MOV R6,>8356
BLWP @DSRLNK
DATA 8
LI RO,4
LI R1,20000
NOP
DEC R1
JNE DEL2
DEC RO
JNE DEL1
MOVB @S1,>9C02
SWPB @S1
MOVB @S1,>9C02
CLR RO

PUT PATTERN INTO IN
R5 = BIT#
R6 = BYTE#
R3 = OFFSET FOR IN
R4 FOR BUILDING NEXT CHAR

R7 HOLDS BYTE BEING DECODED
PUT BYTE IN LSB OF R7
IS BIT ON?
NO
YES, TURN OUTPUT BIT ON
TURN OFF INPUT BIT
PUT BYTE IN MSB OF R7
REWRITE TO IN
POINT TO NEXT BYTE
/2
DO NEXT BYTE IF MORE
PUT OUTPUT BYTE IN MSB OF R4

STORE AT DO
POINT TO NEXT BYTE OF DO
/2
CONSTRUCT NEXT OUTPUT BYTE
NEXT SCREEN POS
EOL?
NO, NEXT POSITION
COUNTER
ONLY ESC K WRITE

PUT LENGTH OF 4 IN PAB
VDP BUFFER

PUT ESC K SEQ IN DATA BUFF

POINT TO DEVICE NAME LENGTH
DSR TO WRITE ESC K SEQUENCE

START OF CPU GRAPHICS BUFFER
QUARTER OF GRAPHICS STRING

VDP ADDR
PUT DO IN DATA BUFFER

128 BYTES

POINT TO DEVICE NAME LENGTH
DSR TO OUTPUT 8 CHARS

POINT TO START OF NEXT QUARTER

DO IT AGAIN FOR LAST HALF
OUTPUT CR/LF

PUT LENGTH OF 2 IN PAB

DSRLNK TO OUTPUT CR/LF

PUT CLOSE IN PAB

DELAY

```



```

MOV B @NEXTL
CLOSE LWPI >83E0
MOV @SAVRTN,R11
RT

SCLN EQU >8354
SCNAME EQU >8356
CRULST EQU >83D0
SADDR EQU >83D2
GPLWS EQU >83E0
FLGPTR DATA 0
SVGPRT DATA 0
SAVCRU DATA 0
SAVENT DATA 0
SAVLEN DATA 0
SAVPAB DATA 0
SAVVER DATA 0
NAMBUF DATA 0,0,0,0,0
DLNKWS DATA 0,0,0,0,0
TYPE DATA 0,0,0,0,0,0,0,0,0,0
C100 DATA 100
H20 EQU $
H2000 DATA >2000
DECMAL TEXT '.'
HAA BYTE >AA
DSRLNK DATA DLNKWS,DLENTR

JMPLNK B @LNKERR

DLENTR MOV *R14+,R5
SZCB @H20,R15
MOV @SCNAME,RO
MOV RO,R9
AI R9,-8
BLWP @VSBR
MOVB R1,R3
SRL R3,8
SETO R4
LI R2,NAMBUF
LNK$LP INC RO
INC R4
C R4,R3
JEQ LNK$LN
BLWP @VSBR
MOVB R1,*R2+
CB R1,@DECMAL
JNE LNK$LP
LNK$LN MOV R4,R4
JEQ JMPLNK
CI R4,7
JGT JMPLNK
CLR @CRULST
MOV R4,@SCLN
MOV R4,@SAVLEN
INC R4
A R4,@SCNAME
MOV @SCNAME,@SAVPAB
SR0M LWPI GPLWS
CLR R1
LI R12,>0F00
NOR0M MOV R12,R12
JEQ NOOFF
SBZ 0
NOOFF AI R12,>0100
CLR @CRULST
CI R12,>2000
JEQ NODSR
MOV R12,@CRULST
SBO 0
LI R2,>4000
CB *R2,@HAA
JNE NOR0M
A @TYPE,R2
JMP SGO2
SGO MOV @SADDR,R2
SBO 0
SGO2 MOV *R2,R2
JEQ NOR0M
MOV R2,@SADDR
INCT R2
MOV *R2+,R9
MOVB @SCLN+1,R5

```

```

JEQ NAME2
CB R5,*R2+
JNE SGO
SRL R5,8
LI R6,NAMBUF
NAME1 CB *R6+,*R2+
JNE SGO
DEC R5
JNE NAME1
NAME2 INC R1
MOV R1,@SAVVER
MOV R9,@SAVENT
MOV R12,@SAVCRU
BL *R9
JMP SGO
SBZ 0
LWPI DLNKWS
MOV R9,RO
BLWP @VSBR
SRL R1,13
JNE IOERR
RTWP

NODSR LWPI DLNKWS
LNKERR CLR R1
IOERR SWPB R1
MOVB R1,*R13
SOCP @H20,R15
RTWP

```

```

VDP RD EQU >8800 VDP read data address
VDP WD EQU >8C00 VDP write data address
VDP WA EQU >8C02 VDP write address address

```

*** General utility workspace registers (Overlaps next WS)

```

UTILWS DATA 0,0
BYTE 0
R2LB BYTE 0
BSS 28

```

* Utility Vectors

```

*
VSBW DATA UTILWS,VSBWEN VDP single byte write
VMBW DATA UTILWS,VMBWEN VDP multiple byte write
VSBR DATA UTILWS,VSBREN VDP single byte read
VMBR DATA UTILWS,VMBREN VDP multiple byte read
VWTR DATA UTILWS,VWTREN VDP write to register

```

=====

* VDP UTILITIES

** VDP single byte write

```

*
VSBWEN BL @WVDPWA Write out address
MOVB @2(R13),@VDPWD Write data
RTWP Return to calling program

```

** VDP multiple byte write

```

*
VMBWEN BL @WVDPWA Write out address
VWIMOR MOV *R1+,@VDPWD Write a byte
DEC R2 Decrement byte count
JNE VWIMOR More to write?
RTWP Return to calling Program

```

** VDP single byte read

```

*
VSBREN BL @WVDPRA Write out address
MOVB @VDP RD,@2(R13) Read data
RTWP Return to calling program

```

** VDP multiple byte read

```

*
VMBREN BL @WVDPRA Write out address
VRDMOR MOV *R1+,@2VDP RD Read a byte
DEC R2 Decrement byte count
JNE VRDMOR More to read?
RTWP Return to calling program

```

continued on page 2

Beginning Forth - part 12

by Earl Raguse, UGOC, CA USA

FASTER GRAPHICS

Last time I promised to show you how to speed up the graphics. I do not think you will be disappointed. Very shortly after I had published last months screens in the Orange County UG ROM, a couple of years ago, a fellow member, and astute programmer, by the name of Ron David gave me a disk, with little comment, except to look at it. I was not only amazed at his clever graphics, but at their speed. What he had apparently done was to avoid the slowness of Floating Point Arithmetic FPA, by writing his own SIN and COS functions which were very fast because he used integer arithmetic, as I will explain later.

I was so impressed with the speed that I modified some of my graphics programs to use them (see Screens 38-42); What a difference. I found one does pay a price though; is not that always the way life is? No free lunch. The price is a restriction to integer numbers less than 32768. You say, "Big deal, I never use numbers that big. I have trouble counting that high. "Maybe so, but you do use numbers like 2.7 which is not allowed because it is not an integer. The difference between 2567 and 2568 does not seem significant, but I have difficulty considering 1.1 and 1.9 to be equal numbers. But they are equal when using integer arithmetic. Also 32768 is not such a big number, it is less than 181 * 182 and they certainly are not big.

Anyway, when I finally figured out how to use Ron's sine-cosine functions and put them into my graphics in place of Forth's floating point arithmetic, all hell broke loose. The weird results were eventually traced to the fact that I was multiplying two numbers, one of which was my loop counter, the other was an incremented parameter. I had never really thought about how big that product got to be.

Actually since index I counts the number of dots to make up the displayed figure it could easily get to be 400 or more. This counter (I in the loop) is sometimes used as the angle and I did have sense enough to know that I had to multiply this by 57.3 (actually 573 10 */) to convert from radians to degrees, because Ron's program works with degrees instead of radians as does FPA. Neither of these seem like big numbers, but their product is larger than integer arithmetic will tolerate.

I will not try to explain all that Ron does in the attached Screens 34-36, mostly because I do not completely understand it myself. But in a nutshell, he has used Forth's capability to define a new defining word SINETABLE, using the compiling word <BUILDS...DOES> which allows him to compile the table of sine values into a word called SINX. SINETABLE has the two digit integer approximations of the actual sine values multiplied by 200.

The values in the table are for integer degrees only. This may irk the precision bent of some people who are used to seeing sine-cosine values printed in tables with 8-10 decimal places and a resolution of at least .001 degrees. Our CRT displays do not require this accuracy. I spent many years using a slide rule which did not do much better than two digit resolution, and I never heard of any engineering project which failed because of that. Evidently Ron realized that this unwarranted precision was sabotaging the potential speed of the computer.

I modified Screen 36 and wrote Screen 37 to practice with his SIN and COS. I wrote the words .SIN and .COS to examine and verify what the values were. Since Ron stores the values as 200 times the actual value, we must divide by 200. The reason he does this is that sine and cosine values are less than one and hence cannot be stored as integers. multiplying and dividing by 200 yields a little better precision than would 100.

The 200 value requires some mental gymnastics. If one divides the stored values by 200, you just get zero, because the result is less than one. Therefore I multiplied them by 100 first so that at least the correct digits could be recognized even if the decimal place is shifted; which is noted by printing "/100" after the values. To use .COS and .SIN, enter an angle, in degrees, then .SIN or .COS. Compare the result with tabulated values or use your scientific calculator. I wrote the word TEST to give a tabular demonstration.

To make the use more straight forward, I wrote *SIN and *COS. To use these words you must insure that the stack contains both a length value and an angle before invoking *SIN or *COS. The * prefix serves both to make unique words and to remind one that the result is a product of function times a length. The length and the angle in degrees must be on the stack with angle on top.

Ron also wrote a word for taking square roots to avoid the slow speed of floating point. I could not understand how his definition worked so I wrote my own. (see *SQRT, Screen 37). My version has the advantage of providing a better value when dealing with numbers which are not perfect squares. His word of course was faster, but Ron's SQRT, for example, the root of 63 as 7 while mine yields 8 which is closer to the correct value.

Enough talk, lets get to Screens 38-42, the revised graphics are much the same as in BFORTH #11, Screens 43-47, but certain changes were made to account for integer arithmetic. The word DONUT has been added, which would have been impractical with the slower plotting speed. I urge you to enter these screens and try them.

SCR #33

```
0 ( STACK MANIPULATORS EGR 12/27/87)
1 : PICK ( n1 -- n2 ) 2 * SP@ + @ ;
2 : ROLL ( nk ... n1 k -- nk-1 ... n1 nk )
3   DUP 1 = IF DROP ELSE DUP 1 DO SWAP
4   R> R> ROT >R >R >R LOOP 1 DO
5   R> R> R> ROT ROT >R >R SWAP LOOP THEN ;
6 : NIP ( n1 n2 -- n2 ) SWAP DROP ;
7 : TUCK ( n1 n2 -- n2 n1 n2 ) SWAP OVER ;
8 : 2DUP ( n1 n2 -- n1 n2 n1 n2 ) OVER OVER ;
9 : 2DROP ( n1 n2 -- ) DROP DROP ;
10 : 2SWAP ( n1 n2 n3 n4 -- n3 n4 n1 n2 )
11   ROT >R ROT R> ;
12 : 2OVER ( n1 n2 n3 -- n1 n2 n3 n1 n2 )
13   >R 2DUP R> ROT ROT ;
14 : 2ROVER ( n1 n2 n3 n4 -- n3 n4 n1 n2 n3 n4 )
15   2DUP >R >R 2SWAP R> R> ;
```

SCR #34

```
0 ( BYTE SINETABLE-SINX/200 RON DAVID)
1 FORGET IT
2 : SINETABLE <BUILDS 0 C, DOES> + C@ ;
3
4 91 SINETABLE SINX
5 3 C, 7 C, 10 C, 14 C, 17 C, 21 C, 24 C, 28 C, 31 C, 35 C,
6 38 C, 42 C, 45 C, 48 C, 52 C, 55 C, 58 C, 62 C, 65 C, 68 C,
7 72 C, 75 C, 78 C, 81 C, 85 C, 88 C, 91 C, 94 C, 97 C, 100 C,
8 103 C, 106 C, 109 C, 112 C, 115 C, 118 C, 120 C, 123 C, 126 C,
9 129 C, 131 C, 134 C, 136 C, 139 C, 142 C, 144 C, 146 C, 149 C,
10 151 C, 153 C, 155 C, 158 C, 160 C, 162 C, 164 C, 166 C, 168 C,
11 170 C, 171 C, 173 C, 175 C, 177 C, 178 C, 180 C, 181 C, 183 C,
12 184 C, 185 C, 187 C, 188 C, 189 C, 190 C, 191 C, 192 C, 193 C,
13 194 C, 195 C, 196 C, 196 C, 197 C, 198 C, 198 C, 199 C, 199 C,
14 199 C, 200 C, 200 C, 200 C, 200 C, 200 C,
15 -->
```

SCR #35

```
0 ( COMP. CODED SIGSIN - SIN : BY RON DAVID )
1 HEX
2 CODE SIGSIN
3
4 04C6 , 0201 , 0168 , 0203 , 00B4 , C1B9 , 0287 , 0168 ,
5 1501 , 1001 , 3DB1 , 0287 , 010E , 1501 , 1002 , 61C1 ,
6 100D , 0287 , 00B4 , 1501 , 1003 , 61C3 , 0507 , 1006 ,
7 0287 , 005A , 1501 , 1002 , 61C3 , 0747 , C647 , 045F ,
8
9 DECIMAL
10 : SIN SIGSIN DUP 0< IF ABS ' SINX + C@ MINUS ( SINX AD.)
11   ELSE ' SINX + C@ THEN ; ( +(0-9) )
12
13
14
15 -->
```

continued on page 6

