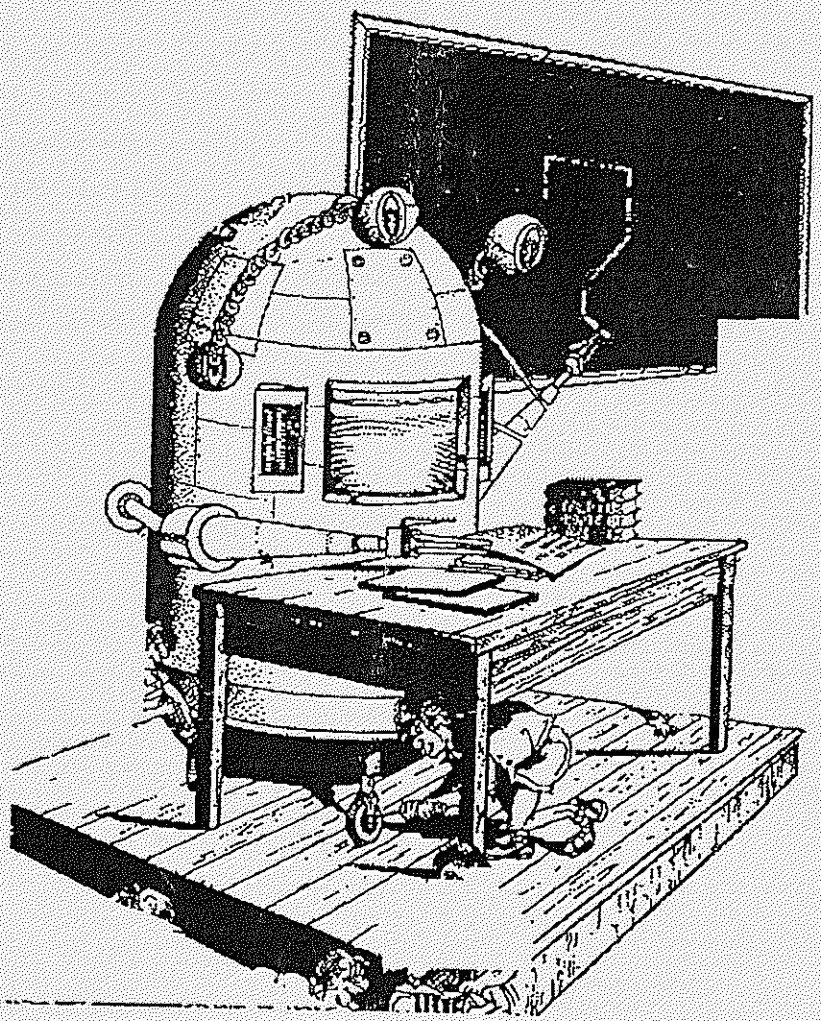
TIsHUG NEWS DIGEST

Focusing on the TI99/4A Home Computer

Volume 12, Number 6 July, 1993

Registered by Australia Post - Publication No. NBH5933



Sydney, New South Wales, Australia $3

TIsHUG (Australia) Ltd.
A.C.N. 003 374 383

TIsHUG News Digest

All correspondence to:
C/o 3 Storey St.
Ryde 2112 Australia

TIsHUG Sydney Meeting

The July Meeting will start at 2.00 pm on the 3rd July 1993 at Ryde Infants School, Tucker Street, Ryde.

Printed by
Kwik Kopy Parramatta

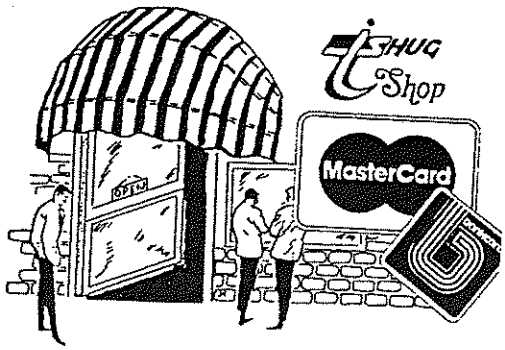
TIsHUG News Digest ISSN 0819-1984

INDEX

Editor's Comments

by Bob Relyea

There are many times at school when I am very happy that I do my computing with a TI. They have all the 'fancy' PCs and Macs and all the problems that go with them. They are forever having an expert come to the school to sort out some expensive problem ranging from a virus to something that is malfunctioning. When you have equipment that sophisticated, any problem can be big bucks! They can do amazing things but I am not sure that the problems that go with them are worth it for the average user. I personally believe that some people's dissatisfaction with and abandonment of the TI can be traced to a common trait of mankind- that the 'grass is greener', rather than to logic. The amount of extra usefulness that the ultra modern machines give us is not worth the vast outlay of cash for most people. Here we have a neat little computer, low in cost, capable of all basic things and ready-made for the person who likes to tinker, program and be inventive. I get so much pleasure out of my machine that I have no plans whatever of shelving it for something else. Sure, the kids at school can learn on these newer machines as preparation for jobs, etc. I am not trying to rubbish progress by any means. Perhaps I could summarise my thoughts by quoting some words of Ross Mudie following a conversation of a few months ago, "we can use the PCs & Macs at work but at home it is our hobby, the TI". Let's keep that spirit alive!



TIsHUG SHOP

with Percy Harrison

The attendance at the June meeting was not too disappointing considering the inclement weather and the fact that the Home Computer show was on at the Sydney Showground. It was indeed pleasing to see two new families turn up with their consoles to get some assistance from our longer standing members on how to interface their console with a cassette recorder and learn the ins and outs of their recently acquired TI's and what they can achieve with them.

You may have noticed that the June issue of TND has been cut from 24 pages to 20 pages due to a shortage of articles available to our editor. We are becoming concerned as to the future of the TI and TI club and feel that unless we can encourage members to become more active in contributing articles to maintain the high standard of our magazine the TI will rapidly fade into extinction and those dedicated TI owners will find themselves left without any technical support or servicing facilities. Members should keep in mind that readers of our magazine encompass a wide range of TI experience, ranging from those who would be considered as being experts in almost every aspect of this orphan computer to those at the other end of the scale who virtually know Zilch about the TI and its capabilities, having only recently acquired their machine. Perhaps we could encourage those at the upper end of the scale who have a very good knowledge of, and many years experience with the TI, to write articles aimed at helping the novice in getting more out of their new acquisition. It does not matter how basic the article might be as there will be someone in our club who will benefit by such articles and in this way we will encourage them to stay with their TI, support the club and thus keep it as a going concern.

I wonder how many of us who read this article will think that other members with more experience will be better equipped to write articles for the magazine and therefore do nothing themselves. let's look at it from a slightly different aspect and ask ourselves what we have achieved from being a member of the TIsHUG club and then convince ourselves how selfish we would be if we sat back and left this work to the stalwart few who consistently contribute material to the TND each month. Browse back through your magazines and you will notice that the same few members from our club contribute the local articles each month.

To those members who have ordered the TIM/SOB Cards, we still are unable to get a despatch date from the supplier, OPA. I did in fact write to OPA some four to six weeks ago expressing our concern at not having received the 15 units that we ordered through the Brisbane Group and politely requested them to respond to my letter advising as to whether they were going to honour their commitment to us and if so to advise us of their anticipated despatch date. Unfortunately OPA did not see fit to respond to my letter, an action which I now believe to be in keeping with their usual performance, as other club magazines that I have received from abroad recently also complain about unanswered correspondence that they have sent to OPA. One would think that having received, and apparently spent, well in excess of Aust $3000 from the Brisbane and Sydney TI User Groups that they would at least have the decency to spend the few minutes that it would take to reply to our letter. Apparently they are only good at spending our dollars and cents and not in spending time to keep us informed.

Col Christensen of the Brisbane Group has not only written numerous letters to OPA but has spent considerable Time and expense trying to communicate with them by phone but apparently they will not reply to either the recorded messages that he has left on their answering service or his letters. I will continue to write to this company until such time as we either receive an acceptable answer or receive the cards - hopefully the latter.

Finally we plan, in the very near future, to start stocking freeware for those members who also own an IBM compatible computer with the view that it will encourage them to remain in and support the club.

CLUB SOFTWARE DISKS

G001	Module Games #1 SSSD	$2
G002	Module Games #2 SSSD	$2
U003	Utilities Disk #1 SSSD	$2
PP004	Page Pro Pics #1 SSSD	$2
AT005	TI Artist Pics #1 SSSD	$2
G006	Module Games #3 SSSD	$2
G007	Module Games #4 SSSD	$2
G008	Games #5 SSSD	$2
G009	Games #6 SSSD	$2
AV010	Adventure Games #1 SSSD	$2
PP011	Page Pro Pics #2 SSSD	$2
G012	Games Disk #7 SSSD	$2
G013	Games Disk #8 SSSD	$2
AV014	Adventure Games Disk #2 SSSD	$2
PP015	Page Pro Pics #3 SSSD	$2
U016	Utilities Disk #2 SSSD	$2
G017	Games Disk #9 SSSD	$2
AT018	TIA Char/Fonts Disk#2 SSSD	$2
PP019	PP Pictures Disk #4 SSSD	$2
PP020	PP Templates Disk #1 SSSD	$2
PP021	PP Templates Disk #2 SSSD	$2
G022	Games Disk #10 SSSD	$2
G023	Games Disk #11 SSSD	$2
PP024	PP Templates Disk #3 SSSD	$2
PP025	PP Templates Disk #4 SSSD	$2
G026	Games Disk #12 SSSD	$2
G027	Games Disk #13 SSSD	$2
AT028	TI Artist Fonts Disk #3 SSSD	$2
AT029	TIA Pictures Disk #4 SSSD	$2
AT030	TI Artist Fonts Disk #5 SSSD	$2
G031	Games Disk #14 SSSD	$2
A386	Boot (Hard Disk Vers) SSSD	$2
A401	Pix Version 1.2 SSSD	$2
A430	Configuring Funnelweb SSSD	$2
A432	Mini-Mem Utility V1 SSSD	$2
A448	Tips Vers 1.7 SSSD	$2
A448A	Tips Graphics #1 SSSD	$2
A448B	Grips (Tips Companion) SSSD	$2
A450	Funnelweb 4.40 DSDD	$2
A450A	Funnelweb 4.40 (3 Disks) SSSD	$4
A472	TI Writer Supplement SSSD	$2
A473	DM 1000 Version 5.0 SSSD	$2
A482	Horizon Utilities SSSD	$2
A489	Fontart #1 SSSD $	$2
A490	Fontart #2 and #3 DSSD	$2
A506	Memory Manager SSSD	$2
A507	Implanting SSSD	$2
A508	Booklet SSSD	$2

```
TCC1  Tigercub Collection #1 SSSD  .............$2
TCC2  Tigercub Collection #2 SSSD  .............$2
TCC3  Tigercub Collection #3 SSSD  .............$2
TCC4  Tigercub Collection #4 SSSD  .............$2
TCC5  Tigercub Collection #5 SSSD  .............$2
TCC6  Tigercub Collection #6 SSSD  .............$2
TCC7  Tigercub Collection #7 SSSD  .............$2
TCC8  Tigercub Collection #8 SSSD  .............$2
TCC9  Tigercub Collection #9 SSSD  .............$2
TCC10 Tigercub Collection #10 SSSD  ............$2
TCC11 Tigercub Collection #11 SSSD  ............$2

TC820 Health and the Human Body SSSD  ..........$2
TC830 Physics SSSD  ............................$2
TC850 Chemistry SSSD  ..........................$2
TC860 Astronomy Disk #1 SSSD  ..................$2
TC890 Teacher's Helper SSSD  ...................$2
TC911 Display Calculator SSSD  .................$2
TC990 Sports (Requires XB) SSSD  ...............$2
TC1015 Word Processing Utilities SSSD  .........$2
TC1102 Sorts,Scrambles,Searches SSSD  ..........$2
TC1119 Hardware Utilities #1 SSSD  .............$2
TC1120 Sound Effects SSSD  .....................$2
TC1122 Screen Fonts-Peterson DSDD  .............$2
TC1131 Gemini Printer Utilities SSSD  ..........$2
TC1145 Telecommunications SSSD  ................$2
TC1210 Graphics Printing SSSD  .................$2
TC1211 TI Artist Pictures #1 SSSD  .............$2
TC1212 TI Artist Pictures #2 SSSD  .............$2
TC1213 TI Artist Pictures #3 SSSD  .............$2
TC1219 R Kazmer's Xmas Card SSSD  ..............$2
```

Bye for now.

********** END OF ARTICLE **********

TIsHUG Software

by Larry Saunders

Disk AT028

This disk is full of TI-Artists Fonts convertered by ALF.

```
Used= 315 Free= 43
PCSET2_F     22 d 80 PCSET3_F     17 d 80
PCSET4_F     19 d 80 PCSET_F       5 d 80
PSET_F       23 d 80 RCSET1_F     11 d 80
RCSET2_F     11 d 80 RCSET3_F     12 d 80
RCSET4_F     12 d 80 RCSET5_F     12 d 80
RCSET6_F     11 d 80 RCSET7_F     11 d 80
SBLOCK_F     28 d 80 SHADOW2_F    55 d 80
SHELBY_F     36 d 80 STENCIL_F    30 d 80
```

Disk AT029

This disk has TI-Artist pictures, two fonts of which I did not have room to fit on AT028. There is also a TI-Artist picture display program that can be merged with other programs. Note: It must have the program merged into it, NOT the other way around! The T/MENU is the stripped down version of the program. It displays the pictures in full colour, and is ideal for using in programs to display a title screen designed using TI-Artist etc.

```
Used= 334 Free= 24
BI-PLANE_P   25 Prog CLOCK#1_P    25 Prog
CLOWN#3_P    25 Prog CLOWN#4_P    25 Prog
CLOWN#5_P    25 Prog CLOWN#6_P    25 Prog
LOAD         13 Prog MENU5_C      25 Prog
MENU5_P      25 Prog STRIPE2_F    48 d 80
T/MENU        8 Prog TRAIN_P      25 Prog
VETICA_F     40 d 80
```

Disk AT030

This is a collection of TI-Artist fonts that I had not seen before I was putting together this disk.

```
Used= 345 Free= 13
BBLOCK_F     21 d 80 BLOCK_F      47 d 80
CARTOON_F     8 d 80 HEART_F      28 d 80
LBLAKTXT_F   35 d 80 LVETICA1_F   35 d 80
MIRROR_F     40 d 80 RICH_F       46 d 80
UBLAKTXT_F   43 d 80 UVETICA1_F   42 d 80
```

Disk G031- Games Disk

Cannon Ball Run, a type of Donkey Kong.
Frogger, the best version of this classic game on the TI-994A.
Night Mission, you are in command of a rescue helecoptor, your mission is as many men on the ground (one at a time) without getting destroyed by the tank. This is a very hard game to play.
Solitaire, This is the updated version of Chainlink (assemble language) the old version was in Extended Basic. This is a game that you must use your brain.

```
Used= 346 Free= 12
CANONBL1XB   35*Prog CANONBL2XB   29*Prog
FROG         33*Prog FROH         26*Prog
LOAD          5 Prog LOAD/N       14 Prog
NMJOY        39 Prog NMKEY        40 Prog
ROOT         28 Prog SOLITAIRE    31*Prog
```

********** END OF ARTICLE **********

Word Processing Part 7

By Col Christensen

One aspect of word processing that needs more coverage, I feel, is the use of printer control code sequences to manipulate the various printer functions. These code sequences range in length from just one character to a great number. The most common sequences, however, are from one to three characters.

When the printer receives a valid single control character, the character is removed from the text string and the text string is acted upon according to the function that it controls. In a sequence, the first control character will usually be the character 27 and it will be followed by one or more other characters. The control character 27, usually referred to as ESCape, will be removed as well as a predetermined number of other characters. The number depends on what the second character is.

As a matter of interest, The control characters from 0 to 31 in value were named in the days of early systems of electronic text communication and became more evident in the days of teletype transmissions. Some of the mnemonics that generally depict the function of the characters are:

ASCII	MNE	Meaning
2	STX	Start of text.
3	ETX	End of text
6	ACK	Acknowledge
7	BEL	Ding a ling
8	BS	Back space
12	FF	Form feed
14	SO	Shift out
20	DC4	Device control 4
27	ESC	Escape

THE PRINTER CONTROL CODES

Printer control sequences can be grouped into the following categories:

1. Text mode settings
2. Print positioning
3. Page formatting
4. Graphic bit imaging
5. Downloading characters
6. Printer status

For each of these groups the printer functions and control code sequences are given as both mnemonics and their equivalent characters.

TEXT MODE SETTINGS

Function	Mnemonic	Characters
Expanded printing for one line	SO	14
Cancel expanded	DC4	20
Begin condensed mode	SI	15
Cancel condensed	DC2	18
Begin Pica size	DC2	18
Begin Elite size	ESC :	27 58
or	ESC M	27 77
Begin Italics	ESC 4	27 52
Upright print	ESC 5	27 53
Begin emphasized	ESC E	27 70
Cancel emphasized	ESC F	27 71
Begin underlining	ESC - 1	27 45 1
Stop underlining	ESC - 0	27 45 0
Begin dble strike or near letter quality	ESC G	27 72
Cancel above	ESC H	27 73
Begin subscript	ESC S 1	27 83 1
Begin superscript	ESC S 0	27 83 0
Cancel either	ESC T	27 84
Begin expanded print	ESC W 1	27 87 1
Cancel expanded	ESC W 0	27 87 0
Begin overlining	ESC _ 1	27 95 1
Cancel overlining	ESC _ 0	27 95 0

123456789012345678901234567890123456789012345
EXPANDED IN WIDTH
PICA STYLE 10 CHARACTERS PER INCH
LITE SIZE IS 12 CHARACTERS PER INCH
CONDENSED SIZE IS 17 CHARACTERS PER INCH
EMPHASIZED IS DARKER AND BROADER
NEAR LETTER QUALITY IS TOPS
Overlining Underlining both
ITALICS PICA nd ELITE and CONDENSED
SUBSCRIPT SUPERSCRIPT SUBSCRIPT SUPERSCRIPT

Some of the different text modes can be combined as you have probably noticed in the examples above. For convenience, those that will combine are more easily seen if the codes are put into sub-groups. You can use any code from one of the sub-groups with one from any or each of the others but don't be disappointed if some discrepancies occur. Your printer manual might list some of the restrictions and which ones have priority over others.

1 Expanded
2 Pica, Elite, Condensed
3 Subscript, Superscript
4 Italics, Upright
5 Emphasized or not
6 Underlined
7 Overlined
8 Doublestrike (NLQ) or not

PRINT POSITIONING

Function	Mnemonic	Characters
Line feed	LF	10
Form feed	FF	12
Carriage return	CR	13
Set perforation skip	ESC N n	27 78 n
Cancel perf skip	ESC O	27 79
Set margins	ESC X m n	27 88 m n

The perforation skip refers to the space left unprinted at the perforation of continuous (fan fold) paper. It sets the bottom of the page margin to "n" lines. That is, the distance from the last print line on one page and the first on the next page.

The margins code allows setting of the printer's left margin (m) and the right margin (n). If you want to list a basic program in 28 columns just like it appears on the screen, set these two values to 1 and 28 or if you want to print further over on the page, set it to a pair of higher numbers.

PAGE FORMATTING

Function	Mnemonic	Characters
Set 1/8" line spacing	ESC 0	27 48
Set 7/72" spacing	ESC 1	27 49
Set 1/6" line spacing	ESC 2	27 50
Set n/216" spacing	ESC 3 n	27 51 n
Set page length lines	ESC C n	27 67 n
Set page length inches	ESC C 0 n	27 67 0 n

The standard line spacing as set on the printer dip switces is 1/6". Double spaced print can be set with the code ESC 3 H, the H with ASCII 72 making 72/216" or twice 1/6". 1/8" line spacing can be used for listings if you want to conserve paper.

GRAPHIC BIT IMAGING

These codes are used by such programs as TI Artist, My Art and Page Pro in their output of graphic designs to paper. The complications of their use would be beyond the scope of this artice. Briefly put, the picture data is sent out, row by row, in streams of bytes each representing 8 vertical dots of the picture. The 8 dots are converted to data numbers in much the same way as in redefining a character in a CALL CHAR statement in Basic.

Most printers already have a set of graphic characters restricted to horizontals and verticals that could be suitable for doing line drawing. Their ASCII values are usually greater than 128 so are not readily available in word processing. They can be printed however by using transliteration like those below that suit my printer. Change the graphics character numbers to suit your own printer and remember to place a CR symbol immediately after the actual TL.

```
.CO Use CTRL/U & SHIFT .TL 1:218 A TL
CORNER .TL 2:194 B T INTERSECTION .TL 3:191 C
TR CORNER .TL 4:195 D L INTERSECTION .TL
5:197 E CROSS .TL 6:180 F R INTERSECTION .TL
7:192 G BL CORNER .TL 8:193 H B INTERSECTION
.TL 9:217 I BR CORNER .TL 16:196 P HORIZ LINE
.TL 17:179 Q VERT LINE
```

DOWNLOADING

Redefined characters can be down loaded to the printer using code from this group. Another code allows selection of the downloaded character set or the standard set for printing. This data transfer sends bytes each representing 8 vertical dots to be printed. The 8/9 pin type of printer would take about 13 bytes to define a charater. The more pins your printer has the more bytes that have to be sent to define a character. For my printer which has a character definition 48 dots deep and 36 wide in the high quality mode it takes 224 bytes to redefine just one character and 96 disk sectors to house a full set from ASCII 32 to 126.

Modern printers have a range of selectable fonts such as Courier, Sanserif, Orator etc that give a good variety in output. These can be selected either by using control codes or by push buttons on the printer.

PRINTER STATUS

Function	Mnemonic	Characters
Reset printer	ESC @	27 64
Unidirectional print	ESC U 1	27 85 1
Bidirectional print	ESC U 0	27 85 0

Resetting the printer cancels all previously set codes. Unidirectional print is necessary for printing graphics as variations in the printer's horizontal registration otherwise tend to get wavy shapes in vertical lines.

THE HEXADECIMAL NUMBER SYSTEM

Let's look at numbers up to 31 in decimal and hexadecimal, the latter usually being prefixed with a › or a H or suffixed with a H. You can apply the principles of decimal notation to hexadecimal. In decimal the two "houses" are tens and units while in hexadecimal they are sixteens and units. So ›1B is equivalent to 1 sixteen + B (11 decimal) units totalling 27 decimal.

Decimal	Hexadecimal	Decimal	Hexadecimal
0	›0	16	›10
1	›1	17	›11
2	›2	18	›12
3	›3	19	›13
4	›4	20	›14
5	›5	21	›15
6	›6	22	›16
7	›7	23	›17
8	›8	24	›18
9	›9	25	›19
10	›A	26	›1A
11	›B	27	›1B
12	›C	28	›1C
13	›D	29	›1D
14	›E	30	›1E
15	›F	31	›1F

In the CTRL/U mode type the upper case alphabet from A to Z while watching each resultant character on the screen. Which keypress produced a tiny 1 on the screen? Of course, the "A", the 1st letter of the alphabet. And which key put a tiny 9 on the screen? Right again! The 9th letter of the alphabet. "I" is the ninth letter. But when we get to the Z that's only the 26th. How do we get the others up to 31? Look up an ASCII table to find the next characters. The one after Z is the 27th, a "[". What keys do you press to type "[". The magical FCTN/R of course gives us the ESC character, 27 as we already knew but didn't know why.

One other number is missing, the 0. It surely comes before 1 which is obtained with SHIFT/A. The ASCII table shows the "A" character is preceded by "@" so that's it. Play with them for a while if it's still a little hazy. Notice that you can type your own CR, LF and FF symbols on the screen.

USING THE CODES

You can use printer control codes in your text whether you intend to process it through the formatter or just print it straight out from the editor using the PF command. AS I may have said in an earlier article, it is preferable to use transliterates when using the formatter as these tend to keep the adjusted right margin more even.

Now comes the process of typing control code sequences into your text. You will have to refer to both the mnemonic and the characters listed for that particular function.

Firstly, we'll take the code to set the printer to subscript, i.e. ESC S 1 with the characters 27 83 1. As you know, we need the CTRL/U mode to type characters less than 32. Starting at ESC which is character 27 and should show on the screen as a small B with a little vertical dash before it, press CTRL/U and FCTN/R and the character will appear then CTRL/U again to get back to a normal cursor. Now, beside that, we need an "S" typed in the normal way. Just press SHIFT/S. Next comes the character, 1. We need CTRL/U again for this one, and SHIFT/, and CTRL/U again.

Your code on the screen should read dash-B S tiny-1 to represent ESC S 1. Anything typed after that will print out in subscript style. You do not want all of your text to be in subscript, so you must cancel that style somewhere. Looking up the codes table, we find the code to effect that is ESC T. To get it press CTRL/U FCTN/R CTRL/U SHIFT/T. Got it? Now you should be able to handle the twenty-seven keystrokes and fifteen typed characters needed to be able to print the chemical formula for battery acid which is H_2SO_4.

If you can manage that exercise, you can class yourself capable of mastering control of your printer. Try to use other printer codes to bring your printer's hidden talents to the light of day. I know, you will have to refer to the code lists above or your printer manual from time to time. Who doesn't? What I find helpful is little lists stuck here and there over my console and, guess what, they ALL refer to printer codes. Without them as ready references, I might not bother to use printer codes. I would get by without making the printer do what I would like it to and never be really satisfied with the result.

I hope by now that some of the fog surrounding the use of printer codes is beginning to clear and a ray or two of golden sunshine is beginning to peep through.

This is the last in the series and I hope that you have enjoyed it and benefited from it.

********** END OF ARTICLE **********

TI Base - Verion 3.0
Tutorial No.21

by Martin Smoley
NorthCoast 99'ers

SOME IMPORTANT STUFF ABOUT DISK FILES

In the last issue (December, 1992) I tried to give you the idea that you can break down a large Db into several small Dbs and use them all in a normal manner. Actually you can, if you perform some regular disk housekeeping tasks. If you ignore these tasks your system will run slower and slower, and eventually you will have disk problems. Here is my explanation of the situation. The disk storage system for the TI is designed to use every possible space on your diskette to store data. Under normal conditions the system will start at the beginning of your disk and add the new data you wish saved to the end of the line, or next available blank section. You could compare this to adding toy train cars onto the end of a child's train set. In the last issue created Dbs 74LS'S1, S2, S3, S4, S5 and several CFs. Let's suppose that these files are stored on your disk. I might USE S1 and APPEND some data. This new data is stored on the end of our data train. If I now USE S2 and APPEND some data, this data goes on the train after the APPENDed S1 data. Now I decide that the Command File named CF*3 is not what I want, so I Modify it and add two new lines of code to it. When it is saved back to disk the major portion of CF*3 will be placed in its original place, but any new data that is left over will be stored after the stuff I APPENDed to S1 and S2. As you can see, the data on a disk would eventually become a tangled mess, as I have attempted to depict in the right hand column on this page. This bad situation can be made much worse be using the computer against itself. An example of that would be to write a Command File (CF) that would use all five Dbs (S1 .. S5) at the same time, with the CF having the ability to move through all five Dbs and APPEND multiple records with little or no control by the operator. This could create a tangled mess of such proportions that the system would lose track of the data and start declaring disk errors. I will not go into the situation of deleting files from a disk and how that space is reused, but believe me it will make a bad situation much worse. I try to keep this situation in mind whenever I write CFs or do major data handling procedures. My corrective measures are as follow. Make a rough estimate of the size Db you will need and fill the Db immediately. This can be done by using a CF, such as SUBNUM2 from last month to fill an empty Db with partially blank records, or by manually using an empty Db and, in the APPEND mode, holding down the enter key until the desired record number is reached. After that you can use the EDIT mode to enter your data. The EDIT mode does not require additional disk space that will jumble the files. The places you need to worry are operations that add or subtract from your disk, such as APPEND or DELETE for records, or adding or deleting lines from a CF. In addition to good user habits you should make a File Copy of your working disk at least once a month. This is my procedure. I place my working disk in drive #1 and fire up DM-1000 from my HORIZON RAMDISK. I press 1 for File Utilities, 1 for Copy/Move/etc., and 1 for Disk drive #1. After DM-1000 has given me a catalog screen of Drive #1 I press A for all and DM will place a C, for Copy, in front of all the files in Drive #1. Pressing <FCTN 6> tells DM to Proceed and after placing a new blank disk in Drive #2, I enter 2 as the copy destination. After looking at the disk DM tells me it is not initialized and I answer "Y" to proceed. I enter a disk name that includes the current date so I can distinguish it from the rest of my junk and let the copying begin. A file copy takes more time, but it will reorganize all the files as you see them on the left side of this page in a neat close order. Now I use the new copy as my working disk until the next time I recopy the entire batch of files using this procedure. I also use this procedure on my ramdisk. I copy all the files off the work section of my ramdisk, as above, then I use DM to delete all the files in that section of the ramdisk, "be careful !", and then I copy all the files back to that section of the ramdisk. I consider this very necessary because a ramdisk can be jumbled and re-jumbled for many months or even years before any attempt might be made to clean up the files. Deleting and resaving any file, even Extended Basic or DV/80 Files, will eventually mix up the disk storage patterns. Well I have wasted a lot of time explaining why you should use File Copy to clean up your work disks once a month, so I better get back to the applications of TI-Base.

I just took another look at the last article and found some parts to be very confusing. I hope to clear up some of the confusion in this issue- please hang in there.

First, I hope you realized that the printouts of (74LS'S1 .. S5) are not complete. This multiple printout was produced by the CF named LSPRNT/C which is listed below. SET PRINTER=DSK2.LSPRNT redirects the printout to a disk file named LSPRNT, that is nothing new, but SET CRLF=OFF is new. This new command allows you to turn off the Carriage Return and Line Feeds when desired. In the past, when a disk file was needed, you had to go in with FunnelWeb and remove all the extra CRs and LFs to get the proper printout. That was a real pain in the neck for me while writing these tutorials. PRINT (f) is my symbol to print condensed format. From that point to the ENDWHILE is merely a demonstration of using a DOCASE within a WHILE loop instead of IF statements. It is just as easy to write IF LOOP = 1, USE DSK2.74LS'Sn, ENDIF, for this situation, but it seems that I demonstrate IF statements every month. One important line is PRINT ALL ;FOR (CRS>0). This actually tells TIB to look at every record in the currently active Db and PRINT ALL the records which contain a CFS field that hold a value greater than (>) zero.

```
*      05/08/90      LSPRNT/C
CLOSE ALL
 SET PRINTER=DSK2.LSPRNT
 SET CRLF=OFF
 PRINT (f)
LOCAL LOOP N 3
REPLACE LOOP WITH 1
WHILE LOOP<6
  DOCASE
   CASE LOOP = 1
    USE DSK2.74LS'S1
   BREAK
   CASE LOOP = 2
    USE DSK2.74LS'S2
   BREAK
   CASE LOOP = 3
    USE DSK2.74LS'S3
   BREAK
   CASE LOOP = 4
    USE DSK2.74LS'S4
   BREAK
   CASE LOOP = 5
    USE DSK2.74LS'S5
   BREAK
  ENDCASE
 PRINT ALL ;FOR (CRS>0)
  CLOSE
REPLACE LOOP WITH LOOP + 1
ENDWHILE
SET PRINTER=PIO.CR.LF
SET CRLF=ON
RETURN Copyright Martin Smoley 1990
```

In this situation, if you SET RECNUM OFF at the beginning of the CF and SET HEADING OFF after the first loop, you would give the appearance of one continous file, not 5 separate DBs. After all, this set of tutorials is designed to show you how to use several smaller Dbs instead of one very large Db.

Later in this article I have listed LSEDIT4/C along with its sub-CFs, \SCRN/C, \ED/C and LSUSE/C. I have condensed the print because I did not want to waste the space and also because it is a new version of LSEDIT3/C. If the small print is confusing, you can compare it to last months CF to check most of the code. The reason it is listed again is because I have made some changes, and also because it contains the main theory on how to handle multiple Dds as one unit. This theory will be presented and re-presented in an effort to show you how simple it really is. The LSEDIT CFs in this series use this basic idea, open all five of the LS series Dbs at the same time, ask the operator which item they wish to edit, decide in which one of the Dbs that item might be found, go to or SELECT that area, search for the item and if found, edit that item. This is basically the same as if (by thinking) you decided which Db contained the item you wanted to edit, USEd that Db, held down <FCTN 5> to leaf through the DB and EDITed the record if you find it. I would like to take a closer look at LSEDIT4, even though it is a waste of time for those of you who understand TIBs language. First I always CLOSE ALL Dbs, so I know what is going on in the system. INSTALL ADD DSK1.\ED and \SCRN will take those disk files and ADD them to VDP Memory where TIB will use them as normal CFs, but faster. I will cover the INSTALL stuff again later. LOCAL ITEM N 5, is the item we will tell TIB to search for. REPLACE ITEM WITH 1999 is my way of holding TIB in the WHILE loop in the middle of the \SCRN CF. You will see it as WHILE (ITEM<999) .OR. (ITEM<1688). It is kind of backwards and hard to understand, but simply stated the idea is this, if the number contained in ITEM is not between 998 and 1689 TIB will keep you locked in this loop until you enter a part number that is. This is an attempt on my part to assure a valid part number search. The exception to that rule is -1 which is the return path. LOCAL LOOP N 2 and REPLACE LOOP WITH 1 are my way of creating an endless loop. In Boolean, 1 means true. Therefore, the statement (WHILE LOOP) will loop forever, because when tested the answer for loop will come back true. You could get out of this loop by replacing LOOP with zero (0), somewhere in the CF, but we will do that. Just before WHILE LOOP is the statement DO DSK1.LSUSE. This statement runs the CF LSUSE to set up our five Dbs in slots 1 through 5. DO \SCRN, is the command to run the CF named \SCRN from VDP RAM. This CF puts up the complete entry screen and asks you to enter the item number you want to edit. If you enter -1 the CF will terminate, but if you enter a valid number it will be stored in ITEM and TIB will jump back to LSEDIT4 and proceed down to the five major IF sections. IF ((ITEM>999).AND.(ITEM<1100)) seems odd, especially if you entered a zero (0), or possibly (01). You must not forget that we set the position of the LS series from 1000 to 1999, so zero (0), or 741S00, will be searched for as COPNM 1000. If zero was entered, the first IF statement would be true. This would cause slot 1 to be SELECTed and \ED to be executed. The first command in \ED is FIND ITEM. All the Dbs should be SORTed ON the COPNM field. The sort should have been done automatically at the end of SUBNUM2.

If the item is not found EOF will be flagged. This means the statement IF (EOF) would be true, and you would see the line ITEM NOT FOUND, and returned to \SCRN that you might enter another number. If the item is found, the EOF flag would not be up, and you would go to the ELSE part of that IF statement and EDIT the field that TIB found for you. This stuff is quite intricate, in the way it jumps from one location in the CF to another and back, but the general idea of what's going on is not complicated.

Let's look at some of the new INSTALl stuff which I have discovered, even since last month. If you look at the beginning of LSEDIT4 you can see that I have ADDed both \ED and \SCRN to the new VDP memory location which you can do with version 3.0 of TI-Base. This is great for people without RAM Disks because the access speed is much faster than that of a normal disk drive. The major discovery I made for myself, in the last month, is that the INSTALL area is more or less the same as disk space. As you can see by \ED and \SCRN it appears that INSTALL will support Comments, RETURNs, WHILE/ENDWHILE loops, IF/ELSE/ENDIF statements and I suppose, just about anything you can do in a disk based CF. The INSTALL space should be used for CFs or Macro Commands that you need frequently. \SCRN and \ED will be used for every loop of LSEDIT4. As I demonstrate in LSEDIT4, you also have the option of ADDing and REMOVE-ing CFs with each new major CF. \ED/C and \SCRN/C used approximately 1,000 Bytes of VDP Memory, but I still had roughly 1,000 bytes left to use. I already had \RES and several other small CFs in the INSTALL area, so there is ample space available for entry and informational screens. I have learned several tips about INSTALL. Do not attempt to REMOVE or ADD items to the INSTALL area using a CF that resides in the INSTALL area. REMOVE or ADD should be done from a disk bases CF. REMOVE items from the INSTALL area in reverse order that they were ADDed. If you ADD, \ED them \SCRN, you must REMOVE \SCRN, then REMOVE \ED, or TIB will get lost. Also remember that DO \ED is the syntax in a CF, where as \ED is entered at the dot prompt for INSTALl area CFs.

```
*    06/06/90      LSEDIT4/C
CLOSE ALL
INSTALL ADD DSK1.\ED
INSTALL ADD DSK1.\SCRN
LOCAL ITEM N 5
REPLACE ITEM WITH 1999
LOCAL LOOP N 2
REPLACE LOOP WITH 1
 DO DSK1.LSUSE
WHILE LOOP
 DO \SCRN
 IF ITEM<0
  INSTALL REMOVE \SCRN
  INSTALL REMOVE \ED
  DO \RES
RETURN Copyright Martin Smoley 1990
 ENDIF
 IF ((ITEM>999).AND.(ITEM<1100))
  SELECT 1
   DO \ED
 ENDIF
 IF ((ITEM>1099).AND.(ITEM<1200))
  SELECT 2
   DO \ED
 ENDIF
 IF ((ITEM>1199).AND.(ITEM<1300))
  SELECT 3
   DO \ED
 ENDIF
 IF ((ITEM>1299).AND.(ITEM<1400))
  SELECT 4
   DO \ED
 ENDIF
 IF ((ITEM>1399).AND.(ITEM<1689))
  SELECT 5
   DO \ED
 ENDIF
  REPLACE ITEM WITH 1999
ENDWHILE
RETURN Copyright Martin A. Smoley 1990
```

```
* 06/09/90   \SCRN/C for INSTALl
SET TALK OFF
CLEAR
SET HEADING OFF
SET RECNUM OFF
WRITE 6,8,"Enter the right hand digits"
SET INVERSE ON
WRITE 2,6,"                              ";
"        "
WRITE 3,6," 74LS Series Integrated ";
"Circuits "
WRITE 4,6,"                              ";
"        "
WRITE 8,16,"          "
WRITE 9,16," EXAMPLE "
WRITE 10,16,"          "
SET INVERSE OFF
WRITE 12,6,"Manufacturer       You"
WRITE 13,6,"Part Number        Enter"
WRITE 15,12,"74LS221 = >221   <      ENTER"
WRITE 17,12,"74LS01  = >01    <       -1  "
WRITE 19,34,"TO QUIT"
WHILE (ITEM<999).OR.(ITEM>1688)
 WRITE 22,3,"Enter ITEM Number = >      <"
 READ 22,24,ITEM
 IF ITEM<0
  CLOSE All
   RETURN Copyright Martin A. Smoley 1990
 ENDIF
 REPLACE ITEM WITH ITEM+1000
 WRITE 22,3,"Company Part No.  =         "
 WRITE 22,23,ITEM
 WAIT 3
ENDWHILE
 WRITE 21,3,"  Press FCTN 8 Then FCTN 9 "
 WRITE 22,3,"   After Each Record Edit  "
```

```
* 06/09/90   \ED/C for INSTALl
FIND ITEM
 IF (EOF)
  WRITE 21,3,"                           "
  WRITE 22,3,"        ITEM NOT FOUND     "
  WAIT 2
  RETURN Copyright Martin Smoley 1990
 ELSE
  EDIT
 ENDIF
```

```
*         05/06/90         1SUSE/C
SELECT 1
 USE DSK2.74LS'S1
SELECT 2
 USE DSK2.74LS'S2
SELECT 3
 USE DSK2.74LS'S3
SELECT 4
 USE DSK2.74LS'S4
SELECT 5
 USE DSK2.74LS'S5
RETURN Copyright Martin A. Smoley 1990
```

Because of the speed of my RAM DISK, I use the INSTALl area mainly for Macro Commands rather the CFs, but no matter how you use them the new features that have been added to TI-Base are fantastic. The ability to run a CF by typing \SCRN at the Dot prompt instead of DO DSK6.\SCRN to me is wonderful. When you start to get the hang of this, reread page 8 of the May Newsletter, about Macros. I think you will start using this feature more and more.

Nostalgia Time

by Geoff Trott

This series of articles consists of my observations on the contents of the early TNDs starting with the earliest copies of our News Digest that I have in my possession. Assuming that you find that interesting, I am continuing with the series this month. Please stop me if you do not want me to continue. I will repeat my general disclaimer in case anyone reading this article gets the wrong idea. I am attempting to describe the look, layout and content of the newsletters without any critical intent. I will try to avoid using any adjectives which could cause offense and if anyone takes offense, that is purely their interpretation of the words and not my intention. I hope that makes my position clear and that no one will be offended.

Now on to a new year. In 1984, Shane started to use a cover of a different colour and sometimes thickness than the rest of the magazine. In the February issue the cover is dark blue with a space shuttle flying between two inclined planes of rectangular squares. It reminds me a bit of the screen for Buck Rogers. The Logo is present and the banner is "Sydney Newsdigest". It is labelled as Volume 3, number 1. There are 20 pages in this issue. In his editorial, Shane describes the cover as depicting the club as flying out of a void into an unknown but fruitful future. He mentions that CorComp are planning to bring out a new computer with 64K of memory and which would run all the existing software. The March issue of the magazine would be produced by Peter Lynden, the Educational Co-ordinator. Shane also mentions the Version 2.2 system software in GROM which TI released in 1983 and which stopped cartridges without GROMs from being recognised. Shane mentions the regional groups and the responsibility that these have to send in reports for publication in the magazine. He mentions the groups in Liverpool, Newcastle and Wollongong as well as a group in OTC. There is a report from the Liverpool group from Vincent Cerreto.

The February meeting is to be a full day workshop with everyone encouraged to bring alone their computer, family and lunch. The Communicators mentions that there were about 20 members with modems who are looking at conducting chat sessions on T.A.B. (no, not the bookies, but The Australian Beginning!). Doug Thomas from Melbourne was also involved. John Robinson welcomed 100 new members in his Minutes column and wrote about the SST BASIC Compiler. He also lists the contacts for the Regional Groups Meetings. These include: Tony Casmiri (North Rocks); Shane Andersen (Marrickville); Russell Welham (Gorokan); Peter Day (Milsons Point); Vincent Cerreto (Liverpool); and Peter Varga (Bondi).

Russell Welham presented his first article written with TI-Writer on Programming. He gave some hints for getting Extended BASIC to save a large BASIC program so that it was no longer necessary to do a CALL FILES(1) before loading the program. This is a problem when going from a cassette system to a disk system. He also showed how to change a program to get it to fit by putting DATA statements into a disk file and removing them from the program itself. He also gave a short music program called "Aquarius". Mark Nielsen had an article on creating cassette and disk files which looked at data files which can be read into a program. He gave a simple program to use cassette to store a list of names, addresses and phone numbers and then the changes necessary to use disks instead. (I would query his use of VARIABLE 192 for the disk files rather than VARIABLE 80.)

********** END OF ARTICLE **********

The Software Librarian, Terry Phillips, produced a listing of the Club Software Library using Personal Record Keeping module by the looks of it. He also announced the club would be running three monthly awards for member written software. Peter Lynden had an article on Adventure Games and how to approach playing them. He covers most of the adventure games then available and this article would be a good starting point for anyone interested in adventure games. There were also the thirteen commandments for using disks and computers and a review of Blackbeard's Treasure, a game. Electronics Australia printed a small article on the user group which is re-printed here.

There were some tips from the Houston Users Group to note that if you have either the Personal Record Keeping or Statistics Modules without Extended BASIC, you can use functions in these modules to achieve ACCEPT AT and DISPLAY AT functions, not available in console BASIC. Tips from the San Gabriel Valley User Group include being wary of the Alpha Lock key when using the Joysticks; typing "***" (I think it is "*#*") on Munchman before the characters start moving will allow a choice of skill level, screen number and number of men; inexpensive, short tapes work as well as any for cassettes; and use low number sprites to increase the speed of execution. There were more little programs to help with keeping track of up to 5 scores, to speak the alphabet or the ASCII code (with the speech synthesizer of course) and using DATA statements for a speech program, all from the USA.

There was an interesting article on an electronic joystick which I was tempted to make at one time. It used 8 metal pads placed around a circle for the 8 possible positions and a push button for the fire button. Three CMOS ICs were used to pick up the signals and get them in the right form for the console. The circuit is missing two diodes as shown, but these are easy to recognise. Some of the junctions are also not identified clearly. It would be interesting to see if this would work as well as promised. There is also a program called "Mushrooms" written by Phil West from Western Australia in Extended BASIC.

In the March issue, which is printed on grey paper, there is a change in layout, probably due to a change in editor and consists of 24 pages. Peter Lynden took over for the month and the pages have three columns (instead of two) with lines between the columns and thick lines across the top and bottom. On the top of each page is "Sydney News Digest March" and on the bottom of all the odd numbered pages is "Page #". The front cover consists of a crowd of people in cartoon form. Inside is a photograph collage of the workshop day in February (my family is there).

The editorial from Peter Lynden talks about the successful workshop day when about 220 people with 100 computers packed out the hall and stressed the electrical outlets. Peter gave many pages of news in this issue, including advice from Paul Cass of Canberra on the Savage Island Adventure, on flipping disks and on Bumper Book News. Peter Day has an article on starting out to use a computer with BASIC. There are three listings of BASIC programs. The first one is from Brisbane and allows a listing of a BASIC program to be sent to a printer (RS232 or PIO) in any number of columns wide. The next one is called Muncher, written by Phil Hinton, and is a game, while the third one is called Skip to the Loot and is by Gene Krawcyzk from Adelaide. The last one is from Jon Todd of USA and is an adventure type game called The Farmer's Dilemma.

Russell Welham has an article on music information for the full day workshop which summarises an approach to writing music programs. It includes a music hidden names puzzle. The Communicators talks about the facilities available with a modem and talks of the plans to start the club's own BBS and asks for help from the members to write a program to run the BBS. Ross Mudie eventually wrote the final version of this a few years ago. Andrew Nutting has an article on assembler language with a simple program to get started with. Younger set mentions that when you put in the code to select where you want to start games like Munchman, to chose screens 1 to 10 enter the single digits 0 to 9, while for screens 11 to 20 enter 00 to 09. There are reviews of the games Aztec Challenge and D-Station.

John Robinson's Minutes column states that there were 110 new members in January, each of whom received a survey form. The club accepted an invitation to participate in the Second Australian Computer Show with a stand at the computer corner. He reports on a TI-Writer "bug" where characters are lost as the program changes windows. The solution is to use only one window. The Forth language from TI was placed in the public domain and copies of the disk and manual were made available to members. Peter Day started a special interest group on Multiplan. The Wollongong Regional Group gets a mention with its meeting at Shop 4 in Corrimal. Robert Vines planned to start a Blaxland regional group. Jim Patterson (Peterson perhaps) from Columbus, Ohio has sent a problem "Write the following Extended BASIC statement: IF X=1 THEN Y=7 ELSE IF X=2 THEN Y=33 ELSE IF X=3 THEN Y=19 ELSE IF X=4 THEN Y=21 in just one line of TI BASIC". It sounds like Jim Peterson to me.

This was the start of the year in which the membership peaked at over 1000 members. It was very exciting times if a little frustrating. The sheer size of the club brought an enormous load onto the committee and it is not surprising that only a few survived. These were: Peter Varga (President); John Robinson (Secretary); Terry Phillips (Treasurer/Librarian); Paul Mansell (Advertising); Graeme Hollis (Crisis line); Chris Ryan (P.R.); Russell Welham (Music); Peter Lynden (Education); Andrew Nutting (Assistant librarian); Shane Andersen (Editor).

********** END OF ARTICLE **********

"What a day! The computer broke down and we all had to think!"

Techo Time

by Geoff Trott

RAMdisks And Associated Devices

I would like to talk about RAMdisks this month for two reasons. The first reason is that I have had a few to look at recently and I have some comments you may find interesting. The second reason is that Dick asked me about adding EPROMs to the RAMdisk using the latest very large EPROMs. So I will first talk a little about RAMdisks and how they work and then look at the particular issues I have mentioned.

I find RAMdisks to be a very interesting device because they are basically very simple and yet add such a lot to the operation of the computer. They also take the greatest advantage of the special architecture of the 9900 processor of all the devices which can be attached to the computer. let me try and explain what I mean. The input and output devices for the TI99/4A have available the special serial I/O of the processor which allows 4096 bits of data to be set and reset on output or the value of each bit can be read on input. The TI99/4A computer has divided this range of 4096 bits up into 32 groups of 128 bits each of which 16 groups are available for use outside the console. Each group of 128 bits is assigned to a particular I/O device or card via the CRU address. Base CRU addresses for use outside the console range from >1000 to >1F00 in steps of >100, where only the even numbers of the addresses are used. Inside the console, the CRU is used for the keyboard, joystick and cassette interfaces at a base address of >0. The operating system looks for I/O devices only at CRU addresses between >1000 and >1F00 and does this by setting the bit at >1x00, reading the byte at memory address >4000 and checking if it is >AA. If this is the case, it knows that there is an I/O device present at this CRU base address and so it can check to see if it has the name of the device it is looking for (PIO, DSKn, etc.). If it is not >AA, the system advances the CRU address bu >100 and tries again.

Once the operating system has finished with a particular device, it resets the bit at >1x00 to turn off that particular device. This is because all the I/O devices share the memory addresses from >4000 to >5FFF (8 Kbytes) for their DSR routines, so that only one device can be enabled at a time. For RAMdisks in particular, the DSR routines use 6 Kbytes of the 8 Kbytes of memory space, with the remaining 2 Kbytes used to window into the RAMdisk memory itself. So how does the RAMdisk enable many Megabytes of memory to be used with only a 2 Kbytes address space? It does this by constructing the necessary address lines with the CRU bits. 2 Kbytes of memory use 11 address lines. There are 128 CRU bits available for each device of which one is used to turn the device on and off leaving 127 for address lines. So the maximum address space for one RAMdisk is 138 address lines which would give an almost infinite amount of memory capability. So there is no practical hardware limit on the size of a RAMdisk other than the cost of the memory and the volume it takes up (perhaps there is a software limitation as well). Thus the CRU I/O feature of the 9900 processor allows the window into the RAMdisk memory to be moved about easily. Each sector of a disk holds 256 bytes of data so that 2 Kbytes is equivalent to 8 sectors of a disk. It is a bit like a disk that is formatted at 8 sectors per track and the seek to the next track involves sending a new bit pattern to the CRU bits.

In order to keep the contents of the RAMdisk memory intact when the power turns off, a battery is used to supply volts to low power static RAM ICs. The battery will supply only the memory chips when the power is off and will not be used when the power is on. The batteries can be re-chargeable or not. There are advantages and disadvantages for each. Re-chargeable batteries are theoretically better, but they do not like to be constantly trickle charged. They rather prefer to be discharged completely and then charged again to keep them in good shape, not a good scenario for a RAMdisk. They also can exude a chemical which does the tracks of a printed circuit no good at all. Normal batteries do run flat but this is only very slowly, more or less at the shelf life rate. The shelf life of good quality alkaline cells is much longer than that of nickel cadmium re-chargeable batteries. I do not use re-chargeable batteries myself.

To perform the switching between power from the supply and the backup supply from the batteries, the power to the memory chips goes through diodes from both the battery and the power supply. (In the case of re-chargeable batteries, there is a resistor between the power supply and the batteries which sets the charging rate.) When the power supply is on, the diode from the battery is reversed biassed and when the power supply is off, its diode is reversed biassed. This means that the power supply must regulate to 5 volts plus a diode voltage drop, which it does by using a 5 volt regulator with a diode in its common leg to ground. This raises the output of the regulator to about 5.7 volts. There are two diodes connected to the power supply at one end (anode) while the other end (cathode) of one diode goes to power the memory ICs and the other end (cathode) of the other diode goes to power all the other ICs on the board.

This leads on to the problems of the RAMdisks I have seen recently. Both of these had 5 volt regulators whose outputs were much more than the specified 5 volts. With the addition of the extra 0.7 volts they were producing more than 7 volts. Replacing one of them with a new regulator from the shop did not solve the problem as that regulator also was giving out too high a voltage but eventually I found a regulator that gave the correct voltage. This problem did not necessarily cause a problem with the operation of the RAMdisk, but it was interesting that the problems with the RAMdisks were solved by making sure that the battery supply was correct. If you have problems with a RAMdisk that develops after some time, check the battery supply. I also discovered that if I tested an 8 Kbyte (2564 chips) RAMdisk on my MiniPE system it gave errors on some ICs while it tested fine in a PEbox. Something to track down one quiet evening!

The other problem of installing large EPROMs on a RAMdisk is reasonably straight forward on the hardware side. It would be made easier if there are 16 RAM ICs installed first as then the address decoding would require fewer gates. If we were starting from scratch, it would be easier to use a PAl chip to do the logic and so leave more space for memory chips. The other problem for very large EPROMs is to get the copy of the disks onto the EPROMs themselves. The existing program would need to be changed to allow larger EPROMs to be written to and installed. If someone wishes to go that way, let me know and I will see what I can do. You need to set up a disk with what you want to be on the EPROMs set up correctly for running from the RAMdisk system. Do this by setting up the programs you want to use on a RAMdisk and then copy them to a blank floppy disk. This allows you to test out all the programs, because once the EPROM is programmed, that is what you have until you program another EPROM. Try and keep the number of sectors used just under a power of 2 sectors (256, 512, 1024). Buy a suitably large EPROM or number of EPROMs and we will give it a go.

********** END OF ARTICLE **********

TI Bits No.25

by Jim Swedlow

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

I had occasion to use two features of TI Writer that were interesting, so I though I would pass them along.

INCORPORATING INSTANCES

This will be very brief, because Bill Nelson, Graphs Guru Extraordinare, did most of the work. He designed two instances, which, together, formed a letterhead. Bill then used Rodger Merritt's PICTURE IT to convert them into a TI Writer compatible file.

Bill, how about an article on PICTURE IT????

The file is filled with transliterate (.TL) commands. It accesses the graphics mode of your Star Gemini or Epson compatible printer. I had two problems.

First, the file would not work with FUNNELWEB. The computer locked up before printing the first line. A quick call to Bill and I learned the fix - go back and use real TI Writer. I have not used TI Writer in ages but I found the cartridge and disk, changed the printer name to PIO.CR and every thing worked fine - except for the second problem.

I wanted to use the letterhead with TI Writer's mail merge capability. But I could not. The PIO.CR printer name in the Formatter is fine for the converted instances, but not for a text file.

Bill explained a solution that involved saving the file to disk and then printing it from there. I opted for a simpler option - two pass printing.

I ran my paper through the first time to print the letterhead (using PIO.CR) and then a second time to print the text (using PIO.LF). It worked.

MAIL MERGE

When would you use mail merge?

* You want to write that annual Christmas letter, but, instead of photocopying, you want to personalize each one.

* You have a business and you want to send individualized letters to your customers (". . . yes, right there in Garden Grove, you can use . . .").

* You are planning a conference, a party or a Fest and you want to send individualized invitations. What to do? You could write your letter and then change key information for each person. Better yet, you could use mail merge and let your computer do the work.

Let's try a Christmas letter. Perhaps it might run like this:

> Dear ___,
>
> It has been a exciting year for us. Junior was accepted to Yale, Mary had straight A's and Bud stole his fifteenth truck.
>
> How are things with you ___? Write soon and let us know.
>
> Love,

We have two variables. The first is the name (Annie, Aunt Susie, Grandpa, etc.). The second is family members (Bueford; Uncle Sam and little Quincy; or, perhaps, no one).

Here are the steps you must take to use mail merge.

1. Write your letter. Where you want to have a variable, identify it with the Alternative Input symbol *n*. The "n" between the asterisks stands for the field number (*1*, *2*, *3*, etc.). Your mail merge ready letter looks like this:

```
.FI;AD
Dear *1*,

It has been a exciting year for us. Junior
was accepted to Yale, Mary had straight A's
and Bud stole his fifteenth truck.

How are things with you*2*? Write soon and let
us know.

                    Love,
```

   Note that there is no spaces between "you", *2* and "?". If Grandpa lives alone, you will want it to read "How are things with you?" Whereas for Annie, you would say "How are things with you and Bueford?".

   Remember to include the Fill and Adjust commands (.FI;AD) and to save your file before going on to the next step.

2. Create your value file. Here is a sample:

```
1 Grandpa
2
*
1 Annie
2 ^and Bueford
*
1 Aunt Susie
2 , Uncle Sam and little Quincy
*
```

   Each line must end with a carriage return. There must be one space between the field numbers and the beginning of the text. There must be one space and then a carriage return after each asterisk.

   If the field is empty (as in the second field in the first letter) nothing will print. To make an empty field, type the field number, one space and then a carriage return.

   Note the required space (^ or SHIFT 6) before "and Bueford". Because there is no spaces between "you", *2* and "?", we have to tell the Formatter to insert the leading space. Otherwise, it will print "How are things with youand Bueford?".

   Save this file using another name. For example, if you called your letter DSK2.LETTER and your could call your value file DSK2.NAMES (original, huh?).

3. You are now ready to merge your two files into individualized Christmas letters. Load the Formatter and use DSK2.LETTERS for the input file name. When you get to this question:

   USE MAILING LIST? N

   type Y and press ENTER. When the Formatter asks you:

   MAILING LIST NAME:

   type in your file name (in our example, DSK2.NAMES) and press ENTER.

   TI Writer will print a letter for each entry in your value file (in our case, three letters).

There are more tricks with mail merge but this will get you started.

********** END OF ARTICLE **********

Rambles

by Stephen Shaw

May I remind you any program EVER published by Regena, in ANY publication, is available on disk or tape for just US$4 from the author. Additionally, the following books are available from Regena:

Programmers Reference Guide to the TI99/4A. 358 pages. US$15.00
First Book of TI Games. (29 listings). US$13.00
Basic Programs for Small Computers (including TI99/4A) (40 programs). US$13.00
and on disk...

ALL programs by Regena published in Micropendium in 1989 for $5-(One disk):
San Diego Guide, 12 cake recipes with print option, printing random math problems, knit a sweater for 25 to 35" chest using different guages, high school science multiple choice test; Four card solitaire, Magic Boxes math puzzle; Presidents of the USA, learning South American geography, White Christmas greetings "card".

ALL programs by Regena published in Micropendium in 1990 for $10-(Two disks):
Tour of Tucson, Dice game-Yacht; Card patience-Pyramid, Plane Geometry Postulates and Theorums, Learning to read lower case letters; Multiply and Divide math tests; Books of the Bible test; Fairisle pattern design, The 12 days of Christmas; Scripture Quiz.

Disk of 16 game programs (see below) $10
Disk of 17 math programs $10
Disk of 10 music demo programs $10.

For all of the above, please add something extra for overseas postage!!!!!

REVIEW:
REGENA: DISK OF 16 GAME PROGRAMS FOR TI99/4A
Cost US$10 plus overseas postage, say $2:
From: Regena, 918 Cedar Knolls West, Cedar City, Utah, USA, 84720

Sixteen programs on disk for $10 cannot be bad! This disk contains some golden oldies, some brand newies...
Solitaire is the peg jumping game where you need to leave one peg in the middle. The program is a nice simulation for speed of entry, and for the ability to progressively take back moves. You can also play back your moves and print the details.
Yacht is the old dice game which some sharp games company snazzied up, made a minor name change, and made millions of dollars... this is the ORIGINAL which differs a little from the commercial version.
Pyramid is a card patience game, which I have not been able to make work out- my best is to have four cards left stranded!
Fourcard is an easier card patience.
There are three "pick a pair" memory games, differing in the graphics and the number of pairs- housework mixup, concentration, and match-em.
Colour Code is not unlike Mastermind, while FlipFlop bears an uncanny resemblance to that other old game tarted up and sold for lots of money-the original game was called Reversi. In this version the computer plays rather dumbly!
TicTac is ordinary noughts and crosses, boxes is a math puzzler, grid is a "find the bomb" type puzzle (if you need more than 6 guesses, sharpen up!).
Superchase is a maze in which you collect treasures while a dumb robot bumbles after you. Hidden maze is an invisible maze for you to move out of. And Closeout is not unlike the Stainless Software program "Man and Monsters" in which you cover several levels of floors avoiding monsters which are after you.
And Poker Solitaire is very like the Pewterware program "Challenge Poker" but without the wild card! You arrange cards in a 5x5 grid to try to score poker hands.

Overall a good collection of programs with something for everyone. Everything will run in TIBasic, and only two require VDP or BXB to run in Extended Basic. Worth looking at.

Note: This is a software review only and is not to be taken as a recommendation to purchase from any particular supplier. This waiver is also not to be read as having any negative connotation. Such views are restricted to your committee.

==========

SUCCESSIVE INVERSE INTERPOLATION

How does that heading grab you (it didn't huh...). It refers to a method of solving an equation by making two guesses and then working your way towards the true answer. Many equations have more than one possible solution - for example we can say:

X*X-4=0

is true if X equals EITHER 2 OR -2 that is, two solutions. If we make the x,y locations of each pixel on the screen the two guesses for this method, each pixel location will tend towards a final answer, and if there is more than one possible solution, we can colour that pixel according to the solution its values lead to. Some guesses may not lead towards a solution so we need to deal with that possibility too...

The program below deals with this, producing some interesting graphics - quite slowly I should warn! What interested me in particular in this listing was the way that colours are represented on a mono screen, by plotting not a single pixel but a block of four pixels, which can have from 0 to 4 pixels "on", giving rise to five "colours". This method can be used with other programs of course!

In this listing values which do not lead to a solution are white- this typically occurs if the "curve" of the equation is horizontal for the two guess values, with no angle to indicate a direction to head to!

This method of solution can lead to "false" solutions, due to wiggles in the equations graph, and these are plotted in two shades of grey. Values which lead to a "true" solution are in black.

The "true" solutions to the equation are held as the values of program variables r1,r2,r3,r4 and the formula is in lines 190 and 200. If you change the formula you will need to find the solutions and change line 100 accordingly - possibly other parts of the program as well.

The on screen plotting is the reverse of "normal" plots, and has positive (or larger) values at bottom left and negative (or smaller) values at top right. X is plotted vertically and Y is plotted horizontally.

```
1 ! METHOD OF DOUBLE POSITION
2 ! Paul Gailiunas in Fractal Report 17
3 ! for TI99/4a with ex bas and The Missing Link by
Stephen Shaw Sept 1991
4 ! easily amended for any pixel addressable language
5 ! note particularly how the effect of five colours is
given on a mono screen/printer
6 ! this effect can be used on any other graphic program
needing up to 5 colours, including white.
7 !
8 ! formula is in lines 190 and 200 . Experiment with
other polynomials or other formulae.
9 ! move centre by setting XO and YO. Range is half the
value of a side. Reduce range for magnification- well
worth while!
10 ! takes a long time to plot!
11 !
```

```
80 RANDOMIZE
90 ON WARNING NEXT ! AVOID DIVIDE BY ZERO ERRORS
100 R1=-1 :: R2=-.5 :: R3=.5 :: R4=1
110 XO=2*RND-2*RND :: YO=2*RND-2*RND
120 CALL LINK("PRINT",1,1,STR$(XO)&":"&STR$(YO))
130 RANGE=RND*2 :: CALL LINK("PRINT",170,1,STR$(RANGE))
140 FOR X=30 TO 220 STEP 2 ! only plotting part of
screen!
150 FOR Y=8 TO 168 STEP 2
160 XI=XO+(X-120)*RANGE/120 ! scaling
170 YI=YO+(95-Y)*RANGE/95
180 FOR I=1 TO 20 ! give up after 20 as unlikely to
converge
190 FX=4*XI^4-5*XI*XI+1 ! EG FX=(XI+1)*(XI-1)*(2*XI+1)*
(2*XI-1)
200 FY=4*YI^4-5*YI*YI+1 ! EG FY=(YI+1)*(YI-1)*(2*YI+1)*
(2*YI-1)
! FIRST FORM IS FASTER
210 Z=(XI*FY-YI*FX)/(FY-FX)
220 XI=YI :: YI=Z
230 IF ABS(XI-YI)<0.00001 THEN 250
240 NEXT I
250 IF ABS(YI-R1)<.001 THEN GOSUB 320 ! which solution
is it tending to?
260 IF ABS(YI-R2)<.001 THEN GOSUB 330 ! r1 or r2 or r3
or r4
270 IF ABS(YI-R3)<.001 THEN GOSUB 340
280 IF ABS(YI-R4)<.001 THEN GOSUB 350
290 NEXT Y
300 NEXT X
310 GOTO 310
320 CALL DOT(X,Y):: CALL DOT(X+1,Y):: CALL DOT(X,Y+1)::
CALL DOT(X+1,Y+1):: RETURN
330 CALL DOT(X,Y):: CALL DOT(X+1,Y+1):: RETURN
340 CALL DOT(X,Y):: RETURN
350 CALL DOT(X,Y):: CALL DOT(X+1,Y):: CALL DOT(X,Y+1)::
RETURN
360 END
370 SUB DOT(X,Y)
380 CALL LINK("PIXEL",Y,X)
390 SUBEND
400 ! FRACTAL REPORT is uk pounds ten for 6 issues in UK
401 REM from
402 REM Reeves Telecom labs ltd  West Towan House
porthtowan TRURO cornwall TR4 8AX
```

==========

Last year I reviewed a book by Clifford A Pickover entitled COMPUTERS, PATTERN, CHAOS AND BEAUTY. I recently advised that a new book was coming out...

I do not know if anyone else purchased (or borrowed) the first book, and I do not have a dispensation from the committee to review books, so here is an extract from the publishers announcement:

COMPUTERS AND THE IMAGINATION. Clifford A Pickover.

"...a treasure of breathtaking computer graphics, and startling applications of computer science to art, music, poetry, science technology, and the mystery of creativity" [Martin Gardner, Scientific American].

"...Chaos theory and fractals...have shown the revolutionizing role of the visualisation of complex mathematical data. Computers and the Imagination pushes the adventure one step further. ... includes a range of topics from the how-to construction of artificial spider webs to pain inducing patterns to computer generated poetry...."

Casebound, 416 pages, with 296 mono illustrations and 8 in colour.

Publisher: Alan Sutton Publishing.

ISBN: 0 86299 999 5

==========

The mandelbrot set produces an image of what could be an odd bug, and as we zoom in we find more and more- continued magnification merely produces more and more detail, until we hit the numeric limits of our computer!

The program below produces images which include what could be some exotic jellyfish, and as we zoom in ever closer we find more and more of them. For magnification, concentrate on the area just inside the first black boundary.

The total image is within the range -1.4 to +1.4, and is symetrical about horizontal and vertical axes at point 0,0.

The image can take some hours to complete, and as ever the use of machine code would be very nice - requiring maximum math accuracy, and some means of saving the images, to TI Artist format or to printer.

The program is for TI Extended Basic and the commercial utility The Missing Link (Inscebot/Texaments) but any language that allows pixel plotting will do.

A routine is included to show "colours" on a mono screen or printer dump, with five textures, here representing 16 colours.

```
1 ! BIOMORFOS
2 ! by JOSE E MURCIANO
3 ! APDO 192
4 ! 44080  TERUEL
5 ! SPAIN
6 !
7 ! FROM FRACTAL REPORT # 17
8 ! FOR TI99/4A BY
9 ! STEPHEN SHAW  SEPT 91
10 !
11 ! DRAWS GRAPHIC
12 ! YOU CAN ZOOM IN ON
13 !
14 ! EMPLOYS MONO COLOUR RENDITION OF PAUL GAILIUNAS
15 !
16 ! FOR EX BAS + THE MISSING LINK
17 ! BUT CAN USE ANY PIXEL ADDRESSABLE LANGUAGE
18 !
100 CALL LINK("CLEAR")
101 ! line 110 sets upper,lower,left and right margins
110 XI=-1.4 :: XA=1.4 :: YI=-1.4 :: YA=1.4
120 P=.7 :: Q=.01 :: LX=190 :: LY=190
130 DX=(XA-XI)/LX :: DY=(YA-YI)/LY
140 FOR NX=1 TO LX STEP 2
150 FOR NY=1 TO LY STEP 2
160 X=XI+NX*DX
170 Y=YI+NY*DY
180 FOR K=1 TO 16
190 XN=X^4+Y^4-6*X*X*Y*Y+P
200 Y=4*X^3*Y-4*X*Y^3+Q
210 X=XN
220 IF X*X+Y*Y>100 THEN IF ABS(X)<50 OR ABS(Y)<50 THEN
CALL DOT(NX,NY,K):: K=16 :: GOTO 230 ELSE K=16
230 NEXT K :: NEXT NY :: NEXT NX
239 CALL LINK("SAVEP","RD.PIC")
240 GOTO 240
250 SUB DOT(X,Y,K)
260 ON K GOTO 270,280,290,300,270,280,290,300,270,280,
290,300,270,280,290,300,270,280
270 CALL LINK("PIXEL",X,Y):: SUBEXIT
280 CALL LINK("PIXEL",X,Y):: CALL LINK("PIXEL",X+1,Y+1)
:: SUBEXIT
290 CALL LINK("PIXEL",X,Y):: CALL LINK("PIXEL",X+1,Y)::
CALL LINK("PIXEL",X+1,Y+1):: SUBEXIT
300 CALL LINK("PIXEL",X,Y):: CALL LINK("PIXEL",X+1,Y)::
CALL LINK("PIXEL",X,Y+1) :: CALL LINK("PIXEL",X+1,Y+1)
310 SUBEND
320 ! Fractal Report costs ten UK pounds in UK for 6
issues from
330 ! Reeves Telecom labs ltd
340 ! West Towan House  Porthtowan TRURO
350 ! Cornwall  TR4 8AX
360 !
370 END
```

NOT a long program, but the results are quite fascinating even if they do take a little while!

********** END OF ARTICLE **********

Vincent's Corner

Dear friends,

Welcome to Vincent's Corner for the month of July. To begin with, contributions are most welcome to this column. The address is:

Vincent's Corner
7 Thrift Close
West Pennant Hills, 2125

Any mail I receive will be printed and the best will receive a $10 prize. Please address any Crocodile Jones material to the above address also. Crocodile Jones is now able to answer any question on any Adventure (including 1 and 13).

This program is for any aspiring punters. Although there is no guarantee of a favourable result, it should provide hours of enjoyment with footy tabs.

```
100 ON WARNING NEXT
110 REM ******************
120 REM *FOOTY TAB PROGRAM*
130 REM *                *
140 REM *BY VINCENT MAKER.*
150 REM *                *
160 REM *     MK.III     *
170 REM ******************
180 CALL CLEAR
190 DISPLAY AT(5,1):"FOOTY TAB PROGRAM."
200 DISPLAY AT(7,1):"MARK III VERSION."
210 DISPLAY AT(9,1):"BY VINCENT MAKER."
220 FOR T=0 TO 500
230 NEXT T
240 CALL CLEAR
250 ONE=0
260 TWO=0
270 INPUT "NAME TEAM ONE:":ONE$
280 PRINT
290 INPUT "NAME TEAM TWO:":TWO$
300 CALL CLEAR
310 REM GROUND CONDITIONS
320 CALL CLEAR
330 PRINT "IS IT A HOME GAME FOR ";ONE$;" (YES/NO):"
340 INPUT GROUND$
350 PRINT "IS IT A HOME GAME FOR TEAM ";TWO$;"
    (YES/NO):"
360 INPUT GROUND2$
370 IF GROUND$="YES" THEN ONE=1
380 IF GROUND2$="YES" THEN TWO=1
390 REM RAIN
400 INPUT "HAS IT BEEN RAINING 24 HOURS PREVIOUS THE
MATCH(YES/NO)?":WEATHER$
410 IF WEATHER$="NO" THEN 490
420 PRINT
430 PRINT "DOES ";ONE$;" PLAY GOOD WET WEATHER FOOTBALL
    (YES/NO)?"
440 INPUT RAIN$
450 PRINT "DOES ";TWO$;" PLAY GOOD WET WEATHER FOOTBALL
    (YES/NO)?"
460 INPUT RAIN2$
470 IF RAIN$="YES" THEN ONE=ONE+1
480 IF RAIN2$="YES" THEN TWO=TWO+1
490 PRINT
500 REM RECENT FORM
510 PRINT "DID ";ONE$;" HAVE A WIN LAST TIME THEY PLAYED
    (YES/NO)?"
520 INPUT GAME$
530 PRINT "DID ";TWO$;" HAVE A WIN LAST TIME THEY PLAYED
    (YES/NO)?"
540 INPUT GAME2$
550 IF GAME$="YES" THEN ONE=1+ONE
560 IF GAME2$="YES" THEN TWO=TWO+1
570 PRINT
580 REM POINTS ON THE TABLE
590 PRINT "TYPE ";ONE$;" POINTS ON THE TABLE:"
600 INPUT POINTS
610 PRINT "TYPE ";TWO$;" POINTS ON THE TABLE:"
620 INPUT POINTS2
630 IF POINTS>POINTS2 THEN ONE=ONE+1
640 IF POINTS2>POINTS THEN TWO=TWO+1
650 REM INJURIES
660 PRINT
670 PRINT "HOW MANY INJURIES DO ";ONE$;" CARRY AT THE
    MOMENT?"
680 INPUT INJURY
690 PRINT "HOW MANY INJURIES DO ";TWO$;" CARRY AT THE
    MOMENT?";INJURY2
700 INPUT INJURY2
710 IF INJURY>INJURY2 THEN ONE=ONE+1
720 IF INJURY2>INJURY THEN TWO=TWO+1
730 REM PENALTIES
740 PRINT
750 PRINT "HOW MANY PENALTIES DID ";ONE$;" GIVE AWAY IN
    THEIR LAST MATCH?"
760 INPUT PCOUNT
770 PRINT "HOW MANY PENELTIES DID ";TWO$;" GIVE AWAY IN
    THEIR LAST MATCH?"
780 INPUT PCOUNT2
790 IF PCOUNT>PCOUNT2 THEN TWO=TWO+1
800 IF PCOUNT2>PCOUNT THEN ONE=ONE+1
810 REM TRIES SCORED
820 PRINT
830 PRINT "HOW MANY TRIES DID ";ONE$;" SCORE IN THEIR
    LAST MATCH?"
840 INPUT TRY
850 PRINT "HOW MANY TRIES DID ";TWO$;" SCORE IN THEIR
    LAST MATCH?"
860 INPUT TRY2
870 IF TRY>TRY2 THEN ONE=ONE+1
880 IF TRY2>TRY THEN TWO=TWO+1
890 PRINT
900 REM GOALS KICKED
910 PRINT "HOW MANY GOALS DID ";ONE$;" KICK IN THEIR
    LAST MATCH?"
920 INPUT GOAL
930 PRINT "HOW MANY GOALS DID THEY MISS?"
940 INPUT MISS
950 PERCENT=GOAL/GOAL+MISS
960 PRINT "HOW MANY GOALS DID ";TWO$;" KICK IN THEIR
    LAST MATCH?"
970 INPUT GOAL2
980 PRINT "HOW MANY GOALS DID THEY MISS?"
990 INPUT MISS2
1000 PERCENT2=GOAL2/GOAL2+MISS2
1010 IF PERCENT>PERCENT2 THEN ONE=ONE+1
1020 IF PERCENT2>PERCENT THEN TWO=TWO+1
1030 REM MATCHES PLAYED
1040 PRINT
1050 PRINT "HOW MANY MATCHES HAS ";ONE$;" PLAYED IN
     BETWEEN THEIR NEXT SATURDAY/SUNDAY MATCH (EG
     PANASONIC CUP/PLAYOFFS ETC.)?"
1060 INPUT TIRE
1070 PRINT
1080 PRINT "HOW MANY MATCHES HAS ";TWO$;" PLAYED IN
     BETWEEN THEIR NEXT SATURDAY/SUNDAY MATCH (EG
     PANASONIC CUP/PLAYOFFS ETC.)?"
1090 INPUT TIRE2
1100 IF TIRE>TIRE2 THEN TWO=TWO+1
1110 IF TIRE2>TIRE THEN ONE=ONE+1
1120 REM RESULTS
1130 IF ONE>TWO THEN WINNER$=ONE$
1140 IF TWO>ONE THEN WINNER$=TWO$
1150 IF ONE=TWO THEN WINNER$="IT WILL BE VERY CLOSE, IF
     NOT A DRAW.  TAKE THE TEAM WITH THE START."
1160 CALL CLEAR
1170 REM SHOW RESULTS
1180 CALL CLEAR
1190 CALL SCREEN(2)
1200 FOR T=1 TO 8
1210 CALL COLOR(T,2,16)
1220 NEXT T
1230 DISPLAY AT(3,1):"THE TEAM TO TAKE ON THE CARD IS:"
1240 DISPLAY AT(5,1):WINNER$
1250 DISPLAY AT(11,1):"BEST OF LUCK."
1260 PRINT "PICK ANOTHER WINNER (Y/N)?"
1270 CALL KEY(0,H,J)
1280 IF J=0 THEN 1270
1290 IF H=89 THEN 100
1300 END
```

Crocodile Jones (C.J.) has sent me these tips on Adventure No. 8.

Do not know what to do at the start?

Do you like gardening? If so, things may not be what they seem.

No entry to Pyramid? Think about buried treasure (in more than one place) and you may UNLOCK some mysteries here.

Mummy a problem? Pour water- make sure you have a full canteen (solution).

No Gold teeth? Search the skull (it is imperative you do this before you give it to the skeleton!!!).

Pharaoh a hassle? Things may not be what they seem in the fireplace and the liquid is useful stuff if you play ball.

Worm a Worm? Do not worry about them.

Nomad mischievous? Do not worry about him either! You just have to put up with it.

Cannot find a place to store the treasures? Intelligent people read messages. If you cannot work it out, too bad. It will come when you are holy.

No diamond bracelet? Are there any [?] with furniture?

If there are any further problems send a SASE with the problem to Crocodile Jones and you will get an answer in one to two days.

Well, have fun and I will still be writing when I am in St John of God Hospital until the end of the year. Any problems with Adventures will be answered quicker if you send me a SASE to to:

Vincent Maker
c/- St John of God Hospital
135 Grose Vale Road
North Richmond

PLease do not send programs to this address but any written correspondence is fine provided it is less than two foolscap pages. You can send your ideas to the West Pennant Hills address and they will be included in the August column.

All the best,

Vincent Maker

********** END OF ARTICLE **********

TREASURER'S REPORT

by Cyril Bohlsen

Income for previous month ..... $ 974.6

Expenditure for previous month .. $ 922.61

Profit for previous month ..... $ 51.99
Membership accounted for $ 580.00 of Income.

Extended Basic Tips

From TINEWS on TEXNET

This file describes a method of BRANCHING ON TWO VARIABLES using the ON GOTO statement in Extended Basic.

Many problems have more than one correct solution. For instance, how to branch to one of six locations depending upon a particular combination of two variables. For reasons of memory and speed efficiency, we needed the absolute minimum number of variables and lines of code. The two variables involved were X and Y. X could be equal to either 1, 2, or 3, and Y could equal either 2 or 17.

The problem: How to combine X and Y in such a way as to have the total equal 1, 2, 3, 4, 5, or 6.

One solution to this problem is to temporarily change Y to either 0 or 3, then Y can be added to X to achieve the desired output. This can be done with a series of IF-THEN statements of a "dummy" variable for Y. However, the number of lines and extra variables required in this solution proves to be excessive.

Upon re-reading the ON-GOTO information in the User's Reference Guide, one will find when the numeric expression is evaluated, the result is reduced to an integer. By re-reading information about the INTeger function, we will discover that the function rounds the fractional values down. This means that a positive fraction which is less than 1 will yield an integer result of 0 and a decimal number of 3, plus a fraction will yield 3. We now have a possible solution. Dividing Y by 5 will yield 0.4 when Y=2 and 3.4 when Y=17. When those numbers are added to X, the result will be 1.4, 2.4, 3.4, 4.4, 5.4, or 6.4. When the computer reduces the result to an integer, the expression will evaluate to 1, 2, 3, 4, 5, or 6, respectively.

Shown here is the algorithim submitted. Three alternate ones are also shown to illustrate the space and efficiency savings when using the ON-GOTO numeric expression. Submitted algorithim:

```
   100 ON X+(Y/5)GOTO 200,300,400,500,600,700
   200 (Code for X=1 & Y=2)
   300 (Code for X=2 & Y=2)
   400 (Code for X=3 & Y=2)
   500 (Code for X=1 & Y=17)
   600 (Code for X=2 & Y=17)
   700 (Code for X=3 & Y=17)
     ******************************
ALTERNATE NO. 1
100 IF Y=17 THEN 120
110 ON X GOTO 200,300,400
120 ON X GOTO 500,600,700
     ******************************
ALTERNATE NO. 2
100 D=0 110 IF Y=2 THEN 130
120 D=3
130 ON X + D GOTO 200,300,400, 500,600,700
     ******************************
ALTERNATE NO. 3
100 IF Y=17 THEN 130
110 IF X=1 THEN 200
120 IF X=2 THEN 300 ELSE 400
130 IF X=1 THEN 500
140 IF X=2 THEN 600 ELSE 700
     ******************************
```

********** END OF ARTICLE **********

The Funny Side of Things

by Larry Holmes

I flew in here on Delta. That's God's gift to GREYHOUND.

I took one of those no frills flights. There was a meter on the oxygen mask.

I'm sure it was a no-frills flight because the stewardess fastened me down with scotch-tape.

Air traffic was sure heavy at the airport. Everytime a plane would take off the flight controller yells fore!

Off the plane I got one of those economy rental cars. I had to put it in low to get off a piece of gum.

The car does go zero to sixty in ten seconds though. Sixty feet that is.

Tell me. In the pinball game of life. How does it feel to be the tilt.

On the way here I saw an interesting bumper sticker. It said drive carefully. We need every taxpayer we can get.

Another bumper sticker said 'honk if you love the quiet'

Hey I like your clothes. I didn't know that fruit of the loom made tuxedos.

My daughter told me today that Eli Whitney made the cotton gin. But who wants a fluffy Martini anyways

She got thrown out of her co-ed dorm last week. She got caught spreading roomers

That tuition is a killer too it's just a matter of time before they start printing the diplomas on the back of the receipt.

Tough crowd. Tough crowd. I never should have given up my day gig.

I had my family tree looked up. They told me I was sap.

My father was stupid. He got fired from bank job. They caught him stealing pens.

My mother had morning sickness after I was born.

When I played in the sandbox cats would try to bury me.

NOW if you thought that one was an utter disgrace...

I have a friend who eats his meals in chandeliers. His doctor told him he had to become a light eater.

Know why sharks never attack lawyers?

Professional courtesy.

How is a catfish different from a lawyer? One is a scum sucking scavenger and the other is a fish.

Why are there many toxic wastedumps in Nevada and more lawyers in California?

Nevada had first choice.

You know TI Casino is a lot like Clearsil. It gets rid of your spots... five and ten spots.

I have never been lucky. I would probably have been on standby for Noah's Ark.

I am so unlucky that I got a paper cut from a get well card.

When you buy a car the goal is to pay it off before it breaks down. With a house the goal is the pay it off before you break down.

My brother-in-law lost interest in tennis after he found out that 'serve' had nothing to do with food.

I can understand though why tennis is so popular. Have you ever tried to jump over a volleyball net?

Los Angeles is sure getting smoggier. The 'LA After Dark Tour' started at noon.

And you know the water isn't much better. You even have to give the goldfish 'Tums'.

Some dog I have too! We call him Egypt because he leaves pyramids in every room.

I worked in a pet store once people kept asking me how big I'd get.

One year they wanted me to be poster boy. The poster was for birth control.

My uncle's dying wish was for me to sit on his lap. He was sitting in the electric chair.

Whaddsa' matter folks? Am I keeping you up?

Once I went to the circus I saw a freaks show. The owners asked me if I would stay on.

I had a lot of pimples when I was young. One day I fell asleep at the library and when I woke up a blind man was reading my face.

I must leave quite an impression on some people. A man followed me around for nearly two hours with a pooper-scooper.

My wife got me to join a bridge club...

We jump next tuesday.

It's tough to stayed married. My wife kisses the dog on the lips-- yet won't drink from my own glass.

Yesterday I met the surgeon general... He offered me a cigarette.

He's a little slow. Last week a bought a motorcycle with an air conditioner.

Have you heard about 'Grand Canyon poisoning'?

One drop and you're dead.

Have you heard about the electronic smoke theory? All electronics are filled with smoke. If the smoke gets out- it stops working.

My wife got me a coffee warmer for christmas. I told her to return it. My TI console works just fine to warm coffee.

********** END OF ARTICLE **********

Learn to Know Your TI

Lesson 6

with Percy Harrison

This lesson introduces you to the output cursor which is invisible on the screen. It marks the place where the next character will be placed on the screen by a PRINT command

When a PRINT statement ends with a semicolon, the output cursor remains in place and the next PRINT will put its first character exactly in the spot following the last character by the current PRINT command.

Without a semicolon at the end, the PRINT command will advance the output cursor to the beginning of the next line as its last official act.

A PRINT command can print several items, a mixture of string and numerical constants, variables and expressions. Numerical constants and variables have not yet been introduced. The items are separated by semicolons.

LESSON 6 TRICKS WITH PRINT

ONE LINE OR MANY?

Enter this program:

```
10 REM FOOD
20 PRINT
30 PRINT "HAM"
40 PRINT "AND"
50 PRINT "EGGS"
```

Run the program. Each PRINT command prints a separate line.

Now enter:
```
30 PRINT "HAM ";
40 PRINT "AND ";
```

(Do not change or erase the other lines.)

Run it. What was different from the first time?

THE HIDDEN CURSOR

Remember the flashing square? It is the INPUT cursor. It shows where the next letter will appear on the screen when you type.

The PRINT command also has a cursor, but it is invisible. It marks where the next letter will appear when the computer is PRINTing.

Rule: The semicolon makes the invisible PRINT cursor wait in place on the screen. The next PRINT command adds on to the same line.

FAMOUS PAIRS

Enter:
```
10 CALL CLEAR
20 PRINT "ENTER A NAME"
30 INPUT A$
40 PRINT "ENTER ANOTHER"
50 INPUT B$
60 CALL CLEAR
70 PRINT "PRESENTING THAT FAMOUS TWOSOME"
80 PRINT
90 PRINT A$;" AND ";B$
```

Be sure to put a space before and after the "AND" in line 80.

SQUASHED TOGETHER OR SPREAD OUT?

Enter NEW and then try this:

```
10 PRINT "ROCK";"AND";"ROLL"
```

After you have run it, try also:

```
10 PRINT "ROCK ";"AND ";"ROLL"
```

Do not forget the spaces after ROCK and AND.

THE INSERT KEY

Type this:

```
10 REM MAGIC KY (do not press ENTER yet)
```

We left the "E" out of KEY!

Use the FCTN and arrow keys to put the cursor over the "Y".

Hold the FCTN key down and press the INS key ie key 2.

(INS stands for INSERT. "Insert" means "stick in.")

Now press the "E" key. An "E" pops in front of the "Y". Press ENTER.

Rule: The letter is inserted to the left of the flashing cursor.

You can insert more than one letter at the same spot if you want.

Fix this: 10 REM DRON (make it "DRAGON")

To stop inserting, just use the FCTN and an arrow key to move the cursor somewhere else. After that, any letter you type will erase the letter the cursor is on.

Assignment 6:

1. Type these lines and then fix them using the INS key.

```
10 REM  WIZRD      (WIZARD)
10 REM  TAS        (TEXAS)
10 REM  CMPTR      (COMPUTER)
```

2. Write a program which asks for the name of a musical group and one of their tunes. Then using just one PRINT command, print the group name and the tune name, with the word "plays" in between.

3. Do the same, but use 3 print commands to print on one line.

Next month we will look at the LET statement.

ANSWER TO LESSON 5

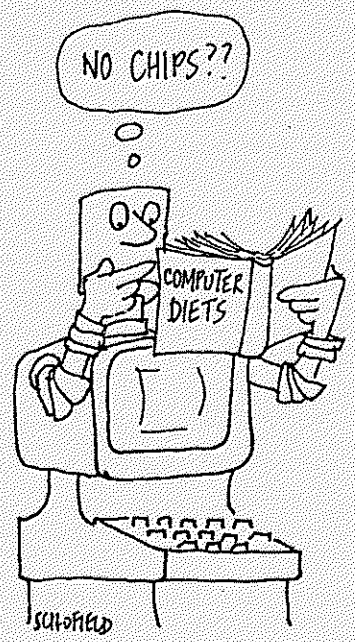
Assignment Question 5-1

```
10 REM TALKING
15 CALL CLEAR
20 PRINT
22 PRINT
24 PRINT
30 PRINT "HELLO, WHAT IS YOUR NAME?"
32 PRINT
34 INPUT N$
36 PRINT
40 PRINT "WELL,"
42 PRINT
44 PRINT N$"
46 PRINT
50 PRINT "IT IS SILLY TO TALK TO COMPUTERS!"
```

Assignment Question 5-2

```
10 REM THE STRING BOX
12 CALL CLEAR
20 PRINT "WHAT IS YOUR FAVOURITE COLOR?"
25 INPUT C$
27 PRINT
30 PRINT "I PUT THAT IN BOX C$"
32 PRINT
35 PRINT "NOW, YOUR FAVOURITE ANIMAL?"
40 PRINT
45 PRINT "I PUT THAT IN BOX C$ TOO"
47 PRINT
50 PRINT "NOW LET'S PRINT WHAT IS IN BOX C$"
52 PRINT
55 PRINT "IT IS:"
57 PRINT
60 PRINT C$
```

********** END OF ARTICLE **********



Testing for 32767

by Ross Mudie
6th June 1993

When writing and developing a program in either BASIC or Extended BASIC on the TI99/4A, the programmer will most likely decide to use the RESEQUENCE or RES command to make the line numbers evenly spaced, eg, incrementing in 10's, which is the default. If the RES command finds a GOTO, IF THEN ELSE etc which references a non-existant line number in the program, the line number reference is changed to 32767. If the program does not have a line numbered 32767 then when the program is RUN it will stop when the reference to the non-existant line number is encountered. If the program happens to have a line with the number 32767, unexpected results may occur in the execution of the program!

When I use the RES command in a large program, I always check if there are any references to line number 32767 before trying to run the program. This saves a lot of effort trying to figure out things the hard way.

To search for one or more occurances of 32767 in a program, the following steps are used.

1. List the program to disk. This creates a DISplay / VARiable 80 file on the disk. I always use the file name LIST, thus when my program which I have just RESEQUENCED has been saved, I use LIST "DSK1.LIST" .

2. Load the Editor/Assembler, TI Writer or Funnel Web editor and then load the file named LIST into memory.

3. Use the Find String command to find any occurances of 32767 or 327 or 767 in the file. (The reason for specifying 327 or 767 is that in the LIST process, it is possible for the 32767 reference to be split over the end of one line and the start of the next in the text file.)

4. Note the program line numbers in which any bad references occur then go back to the program and fix up the bad occurances of 32767.

Some of the programs that I develop create a LIST file which is too big for the editor to load in one go. There are two ways around this problem. If TI or FW Writer are being used, load the second half of the file by specifying the line numbers of the file (not the program contained in the file) when loading. To do this use the loadFile (LF) option and specify the start and end line numbers of the file segment to be loaded, ahead of the file name. Eg, 250 500 DSK1.LIST would load lines 250 to 500 of the file named LIST from DSK1.

There may be someone else like me that doesn't always have TI or FW writer handy and still wants to check a LIST file for line references to 32767. This is where the following little Extended BASIC program comes in handy...

```
100 ! SAVE DSK1.TEST32767
110 DISPLAY ERASE ALL:"Test for 32767, 327 or 767": :
120 OPEN #1:"DSK1.LIST"
130 l=l+1 :: DISPLAY AT(24,25):l :: IF EOF(1)THEN 160 E
LSE LINPUT #1:A$
140 IF POS(A$,"32767",1)OR POS(A$,"327",1)OR POS(A$,"76
7",1)THEN PRINT SEG$(A$,1,POS(A$," ",1))
150 GOTO 130
160 CLOSE #1
170 PRINT :"Finished"
```

This little program reads each line of the LIST file. If any occurances of 32767 or 327 or 767 are found, it prints the start of the line in the file containing the possible bad reference. The program is not intended to be foolproof or fancy, but it works for me!

********** END OF ARTICLE **********

Regional Group Reports

Meeting Summary For JULY

Banana Coast	11/07/93	Sawtell
Central Coast	10/07/93	Saratoga
Glebe	08/07/93	Glebe
Hunter Valley	08/07/93	
Illawarra	13/07/93	Keiraville
Liverpool	09/07/93	Yagoona West
Northern Suburbs	22/07/93	
Sutherland	16/07/93	Jannali

BANANA COAST Regional Group
(Coffs Harbour Environs)

We never miss meeting at Kerry Harrison's residence 15 Scarba St. Coffs Harbour, 2 pm second Sunday of the month. Visitors are most welcome. Contact Kerry 52 3736, Kevin 53 2649, Rex 51 2485 or John 54 1451.

CENTRAL COAST Regional Group

Regular meetings are normally held on the second Saturday of each month, 6.30pm at the home of John Goulton, 34 Mimosa Ave., Saratoga, (043) 69 3990. Contact Russell Welham (043)92 4000.

GLEBE Regional Group

Regular meetings are normally on the Thursday evening following the first Saturday of the month, at 8pm at 43 Boyce Street, Glebe. Contact Mike Slattery, (02) 692 8152.

HUNTER VALLEY Regional Group

The meetings are usually held on the second Saturday of each month at members homes starting at 3:15 pm. Check the location with Geoff Phillips on (049) 428 175. Note that after 9:00 pm this number is used for the ZZAP BBS which includes TI-99 information. Geoff.

ILLAWARRA Regional Group

Regular meetings are normally held on the second Tuesday of each month after the TIsHUG Sydney meeting (except January) at 7.30pm, at the home of Geoff & Heather Trott, 20 Robsons Road, Keiraville. A variety of activities accompany our meetings, including Word Processing, Spreadsheets and hardware repairs. Last month we had a go at hooking up a Megatronics 80-col card to a console and getting it to work. Contact Geoff Trott on (042) 29 6629 for more information.

LIVERPOOL Regional Group

Regular meeting date is the Friday following the TIsHUG Sydney meeting at 7.30 pm. At the July meeting we will have a look at programs that run with an 80 column card. Contact Larry Saunders (02) 644-7377 (home) after 9.30 pm or at work (02) 708-1987, Liquorland, Yagoona for more information.

NORTHERN SUBURBS Regional Group

Regular meetings are held on the fourth Thursday of the month. If you want any information please ring Dennis Norman on (02)452 3920, or Dick Warburton on (02) 918 8132. Come and join in our fun.

Dick Warburton.

SUTHERLAND Regional Group

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

---

TIsHUG in Sydney

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

JULY MEETING - 3rd JULY

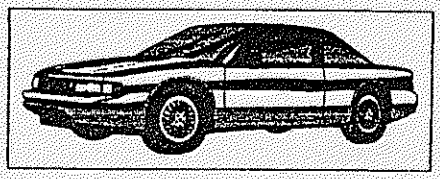
****

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

August	11th July
September	15th August

These dates are all Sundays and there is no guarantee that they will make the magazine unless they are uploaded by 6:00pm, at the latest. Longer articles should be to hand well before the above dates to ensure there is time to edit them.

****



More Articles Needed

by Bob Relyea

At the last club meeting and in Percy' Shopnews article appearing in this issue it was pointed out that we are fast running out of new articles for the magazine. Most of the overseas material that I have been relying on has dried up for the moment and the local scene is not providing as much as it could. I was asked at the last club meeting as to what form the articles should appear in. Frankly, it does not matter as I am prepared to edit any reasonable gear that a member contributes. As I have stated in the past, one club member that regularily contributes articles, puts his article on REM statements and saves them to cassette or disk(!?). On the other extreme, Ross has his regular articles all done up nice and neatly and ready for printing. Most people have a version of some word processor, however, and that is all you have to use. Just type up the article, save it to disk and get it to me. The thing is, to DO IT! Being a great writer and a very experienced club member is NOT a requirement. In fact, if only the real experienced members were the ones contributing, the magazine would not necessarily be satisfying every members's needs. For those who relish in having the articles as ready and 'polished' as possible you can follow the following guidelines:

When setting your formatting commands at the top of the article (before the titles), do it as follows:

.FI;AD;LMO;RM55;PL550;CE2(or whatever)

There does not have to be any gaps between the letters and the number as some people think, so RM 55 and RM55 give the same result. After typing the titles, etc, but before you type the body of the text, use the dot command- .IN5

You may have noticed that the Page Length command was a very high number, 550. It does not have to be that high, but I set it higher than the article is likely to be. This ensures that the entire article prints continuously. There should be no breaks. Obviously, if the article is only 300 lines long and you have used 550 for the Page Length, it will scroll a lot at the end of the printing. This can be avoided if you type the command .PLO after the last line of the article. I hope this has been a help. I will be looking forward to receiving those articles, especially from some who have never contributed.