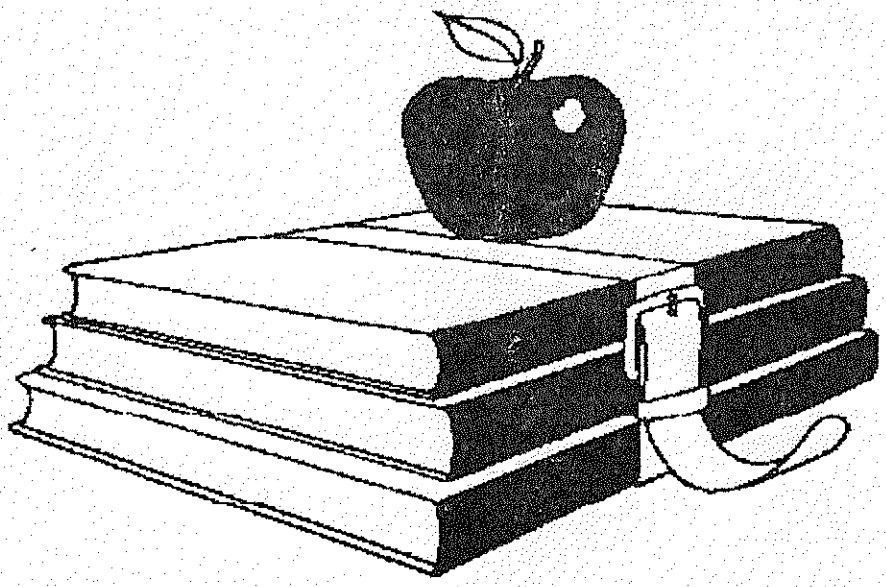
# TIHUG NEWS DIGEST

Focusing on the TI99/4A Home Computer

Volume 13, Number 6 July, 1994

Print Post Approved – Publication No. PP244099/00016



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 2.0 pm on the 2nd July 1994 at Meadowbank Primary School Thistle Street, Meadowbank.

Printed by
Kwik Kopy Parramatta

TIsHUG News Digest ISSN 0819-1984

# T I I N D E X



# I B M I N D E X

## COORDINATOR'S REPORT

Well, it seems that seeing is no longer believing, thanks to the development of the computer. Pictures can't be trusted, documents, cheques, can be remade with present technology, to look like new, and benefit crims. I am told that banks will not publicise theft by computer, both from within their organizations, and by outsiders. In fact I am told that they actually refuse to co-operate when police can charge an offender. I wonder why. Is it because computer crime is much more widespread than the public realises, and that they fear that the public will lose confidence in the banking institutions? I wonder how much the public is subsidising this crime. With the changeover to EFT, and with terminals widely spread in the community, one can only surmise that computer crime will increase. Did you see Hot Chips on Channel Two the other night? I was astonished to find just how easy it is to forge cheques and documents, using a scanner, a computer and a colour laser printer. Amounts on copied cheques can be erased, pixel by pixel, and new amounts written in. This applies to all the information on the cheque. It seems to me that as we put all our trust in the new technology, we make ourselves more vulnerable to far greater problems.

I have written before on our society's blind dependency on technology. It is obvious that as we place more and more wealth and power in the hands of fewer people, that we will gradually lose the things we treasure. As more and more people become and remain ignorant about the new technology, more will be disadvantaged by it. It is clear already that we are entering an era of change, to rival that of the industrial revolution. The applications for the computer are only 20 or so years old. What lies in the future? Already we are talking of classrooms without teachers, factories without people, shopping from home, etc. We are only at the start of this movement. It is beginning to look as if computers will dominate our society within a relatively short period. With increasingly intelligent, voice-activated computers, who could predict where it will end.

Massive data storage and fast retrieval, will enable corporations, police forces, and government to develop and maintain detailed dossiers on every citizen without their knowledge. Computer controlled communications, while giving us instant worldwide access, will also allow instant snooping on selected individuals. Computers will be programmed to record selected conversations. Not only that, communications can be changed at will. But perhaps the greater menace will be the intrusion of the media into every home via tv, satellite, cable, all controlled by computer. The computer already can scan all the available programs, and select what the watcher wants. People will be able to simply relax and let their friendly computer take over. Computers already are being programmed to actually make decisions for people. Can you imagine the applications once we perfect the learning process?

Maybe our lawcourts could dispense justice consistently, for a change, but would we really like this to happen? Would you like the human element taken out of most decision making? I believe it will happen, first in industry, to save money, and will be progressively applied throughout our society. Some of our science fiction writers portray a future society where human beings lack skills, intelligence, and motivation, because they have become the slaves of technology, and not its masters.

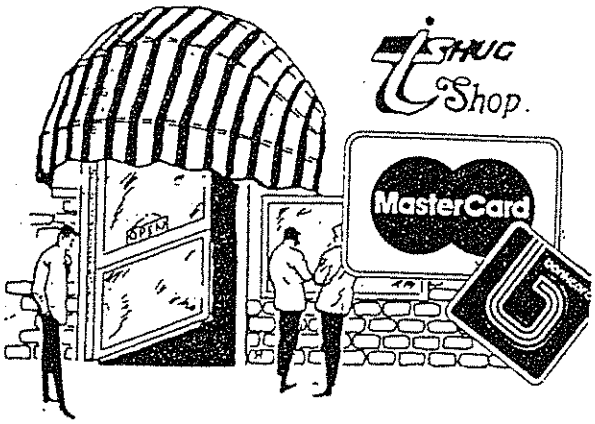
Perhaps the most worrying aspect is the way that information can be changed, modified, altered to suit political, social or commercial motives. History can be rewritten in technicolour. Music can be modified. The news can be altered to suit the prevailing view, or to attract the best ratings. Already we are seeing retouched pictures frequently, because they portray the image our media want us to get at any time. We can no longer believe what we see, or what we hear. One wonders if these developments will weaken peoples' trust in institutions, in government, and in each other.

What is clear, is that we must stay abreast of the developments, as best we can, and avoid the traps that technology can bring. We must be vigilant about our privacy, our political freedom, and unfortunately about our media. We must always remember that computer technology is a two edged sword. There is a flip side for every step forward we take. An apprenticeship on the TI will help us to do so, because it helps create self reliant, independent, computer users.

See you at the next meeting

Dick Warburton

END OF ARTICLE




## TTsHUG SHOP.

with Percy Harrison.

Well, I'm pleased to say that the write-up that I did last month on IBM Compatible Software paid off as the orders placed at our last meeting exceeded the total number of IBM disks sold since we introduced them at the beginning of the year. It is to be hoped that our members with IBM Compatible machines will continue to support the club by investing in the software that we make available.

Our new IBM Compatible machine is now up and running and we will be using it to demonstrate the software that we have for sale through the shop. In addition we will also demonstrate how to use some of the more popular Commercial Programs in the hope that it will allow you to get maximum benefit from them on your own computer. These programs will not be available through the shop as they are commercial and must be purchased through legal outlets. No copying of such programs will be permitted at any of our meetings.

Our thanks go to Geoff Trott who has now delivered 9 sets of TIM/SOB cards partly assembled. We are now sourcing the balance of the components required to complete these cards and when we have purchased them then a small group of well seasoned "solderers" will complete the first nine cards. These will then be tested to ensure that they are working correctly and then will be distributed to the members who paid us for the ones we ordered from OPA (still not a word from the scoundrel who took your money) in the same order as the money was collected by us. We intend to make up about twenty sets as a starter so everyone who has ordered a Card through our club should get his Card in the next few months. I have a record of all those who ordered these cards so please don't contact us, we will get in touch with you when the cards are available.

For those members who did not order a card we have made fifty sets of printed circuit board for the TIM-SOB unit so it is not too late to order them. Unfortunately, due to the higher cost of manufacturing such a small quantity we are not able to sell them at the same price as the overseas ones were going to cost so those who do not have them on order from us will have to pay around $200.00 for them. I am assured that the benefits available from such a card far outweigh the price that we have to charge. If you do want a card please contact me and get your order in as once the printed circuit boards are all use up the chances of procuring more are very remote, again because of the cost in producing small quantities.

On behalf of the Directors of your club, I must apologise to our regular TI meeting attendees for not holding the planned tutorial and training sessions that were scheduled from May onward. Unfortunately the two people that were going to run these sessions have not been able to attend our meetings because of the pressure of their full time work. Ross Mudie has been in New Zealand and Larry Saunders was seconded to the Liverpool branch of the company that he works for which has necessitated devoting many more hours per day and also more weekends to doing his "paid" job. Hopefully Ross will be back at the July meeting but rumour has it that Larry will not put in an appearance until the August meeting so let's keep our fingers crossed. We all owe a vote of thanks to Larry for continuing to keep us supplied with the monthly software releases notwithstanding the heavy workload he has had over the last few months.

Bye for now.

END OF ARTICLE

# TIsHUG July software file.

By Larry Saunders

Diskname G084
Total Sectors 358 Free Sectors 24
Date JULY1994 Files 18

Ballistics: A ballistic game that you try to find the angle and speed to hit target.
Solitaire: (New game) have not played.
Labyrinth: (New game) a very good maze game.
Old Timer puzzle: quite good War Game: I played it a little, not my kettle of fish.
Sink It: Sink the computer ships before they sink you, quite good.
Space Invaders: Classic game, HINT shoot out some side rows first.

| | | | |
|---|---|---|---|
| BALLISTICS | 5 Prog | INVA | 33 Prog |
| INVADERS | 9 D 80 | INVB | 15 Prog |
| LABYRINTH | 12 Prog | LOAD | 5 Prog |
| OLDTIMER | 21 Prog | ROOT | 28 Prog |
| SCENARIO | 23 d 80 | SINKIT/M | 9 d163 |
| SINKIT/REV | 36 Prog | SOLITAIRE | 25 Prog |
| TEST/S | 5 d 80 | WARGAME | 82 i254 |
| WARLOAD | 2 Prog | WARSCREEN | 10 Prog |
| WARSTATE | 7 i254 | WARSTATE2 | 7 i254 |

---

Diskname P081
Total Sectors 358 Free Sectors 44
Date JULY1994 Files 4

Page Pro Pictures

| | | | |
|---|---|---|---|
| BKSPEN | 97 I 13 | BOOKS1 | 57 I 13 |
| BORDER | 90 I 13 | VAN | 70 I 13 |

---

Diskname P082
Total Sectors 358 Free Sectors 6
Date JULY1994 Files 4

Page Pro Pictures

| | | | |
|---|---|---|---|
| APPBKS | 57 I 13 | BKSAPL2 | 67 I 13 |
| BKSLOR | 144 I 13 | BUS | 83 I 13 |

---

Diskname P083
Total Sectors 358 Free Sectors 25
Date JULY1994 Files 18

Instructions for Harrison's
Reformatter
Some time ago, we got a gift from our dear friend Jim Peterson, in the form of a special Extended Basic program that would re-format and right-justify files created by TI-Writer or Funnelweb, or even plain ASCII files as made with the E/A Editor. Jim's program was excellent. A masterpiece of the art of Extended Basic programming. The only problem with that programSwewS * Jim's passing, we decided to make an all-Assembly program to re-format and right justify text files. This product is dedicated to the memory of Jim Peterson, may he rest in peace.

The program we've created is a fairly simple one, with the easiest "user interface" we could manage. There are no menus, no title screens, copyright notices, or anything else that's not needed for the program to function. Everything on the disk is Public Domain, including the Source Code. The whole disk may be shared, copied, placed on BBS services, etc. without compensation to the author.

What's There on the Disk.

Lots of stuff, including the source file, object file, these instructions, a loader for loading the program from Extended Basic, and an example or two that illustrate how the program changes files. The program is an Option-5 E/A file called REFORM. This can be run directly from E/A Option-5, from a Ramdisk menu, from Funnelweb, or from XB using the loader program.

LOADREFORM. Note that when it's loaded from XB, you should exit by pressing Function-= (QUIT) rather than the normal exit procedure.

What will it work on?

The program is designed primarily for working on text files that were created using either Funnelweb's TEXT EDIT or TI-Writer. It will also work on plain ASCII files as made by the Editor/Assembler's editor, or Funnelweb's PROGRAM EDIT function. (These instructions are being written using Funnelweb's TEXT EDIT.)

How do I use this thing?

Simple. Load up the program by any of the above methods. You'll get a prompt on the screen for the Input File name. The entry field will accept up to 28 characters. Type in a name, then press <ENTER>. The editing keys Function-S or -D to move the cursor left and right, Function-1 and Function-2 are available when you're typing in any of the input fields. Next you'll be asked for the Output File Name. Type that in and press <ENTER>. Now the computer will check the disks by opening both files. If an error occurs, you'll be given a message at the bottom of the screen, and pressing any key after the message appears will put you back at the appropriate prompt.

Goes well with opening the files, you'll get a prompt for REFORMAT TO. Here you type in a number greater than zero and less than or equal to 80. Any entry outside that range will be rejected. In theory, you could re-format a file to any number from 1 through 80, but in most cases the number will be something more in the middle of the range. After you've entered the number, you'll be asked whether you want to right justify the output file. Any keypress other than Y or y will be taken as no, and the letter "N" will appear next to the prompt. If you've called for right justification, you'll be prompted TERMINATE CR ONLY? (Y/N). In most cases you would answer this with Y or y for Yes. This will mean that only lines that end with a Carriage Return will be exempted from right justification. When you're dealing with "pure" ASCII files that don't contain Carriage Returns, you may wish to answer this prompt with N or n for no. In this case, lines that end with a period or a colon will also be exempt from right justification. (The last line of a file will not be subject to right justification regardless how it ends, and regardless of the answer to this prompt.) For this prompt, the logic is reversed, so that any keypress other than N or n will be taken as Yes. The next prompt is for the left margin. Many files have a left margin set, and this could cause ridiculous results if the file is being re-formatted to a short length. If your re-format length is less than or equal to the left margin, the program will automatically ignore your Yes to the retain left margin prompt. Any answer other than Y or y will be taken as no, and if your answer is no, you'll get yet another prompt. This one will give you a chance to set a new left margin, different from the one used on the original file. There's a field for a two-digit entry. Leaving this entry field blank will mean the new file has no left margin. Otherwise, the number you enter will be the number of blank spaces on the left side. In other words, if you enter 3, the text will start at the 4th position. The number entered must be less than the reformat number you entered above, else you'll come right back to the prompts, pressing Function-9 will take you back to the previous prompt, so you can back up through the prompts if you notice something you want changed before going ahead. Pressing Function-9 at the first prompt will take you out of the program.

Now that you've answered all the prompts, the processing will start. "READING", followed by the input file name, will appear on the screen. Below that, you'll see each record on the screen as it's read from the disk. (This is done by placing the file buffer within the screen area.) When reading is finished, the legend will change to "WRITING", followed by the output file name, and the output records will appear as they are written to the disk. In almost all cases, the file will be completely processed in one pass. Extremely long files, that are not editable by either Funnelweb or TIW, can still be processed by the reformatter. They will be processed in more than one read and write cycle, without need for intervention by the user.

In all cases, the output file will still be in D/V 80 format, but the length of the records will have been adjusted to the length you entered, or less. Word wrapping will be employed so that words are not split between records unless absolutely necessary. Reformatted files will not contain the "tabs" record that TI-Writer and Funnelweb's TEXT EDITOR place at the end of the file. That record will appear on the screen during the "read" cycle, but won't be passed along in the output file. This is done because the tabs may not be appropriate for the file contents after the re-formatting. You can re-attach a tabs record by running the new file through either the Funnelweb or TI-Writer editors, then saving. Be sure to edit the tabs while in edit mode, so that the tab positions, left margin and right margin are the way you want them.

Limitations

As with any program, there are limits on what this one can do. The main one for this program is the matter of files created with the "all chars" option of Funnelweb Version 5. These files may contain characters beyond ASCII 127, and if the character 128 is at the beginning of a line in the file, the reformatter will stop processing the file at that point. This is so because we check the first character of each record looking for the start of a "tabs" record. That signals the end of a file, so the program stops processing the output when it finds that particular character in the first position on any line.

In "pure" ASCII files, such as those created with the E/A Editor, blank lines are usually represented in the actual file by a record of length one, with one space character in the record. When our program finds this kind of record, it will insert a carriage return. In the output file, that carriage return won't leave a blank line in most cases, but will indicate the end of a line. There are some ASCII files in which a blank line is represented by a file record of zero length. The program will simply ignore such lines completely.

There are many potential uses for this program. You might, for example, just run a file through with its original number of characters, just to get it right-justified for a printing. You can also use it to re-format from the normal eighty character width to forty characters right justified, as for a newsletter that's printed in multiple columns. On the program disk, along with these instructions, is a file called INSTRJ, which is just like the file REFINST (this file) but has been run through the program to right-justify its contents.

We hope, anyway, that you'll find this product useful for whatever you're doing with text files. Should you need help with using any of this, contact the author:

Bruce Harrison
5705 40th Place
Hyattsville MD 20781
U.S.A.
Phone (301) 277-3467

Finally, a reminder. This product is Public Domain software. You may share this with anyone who needs it, place it on BBS, in User Group libraries, and so forth. Just having it get used is reward enough for us.
Enjoy!

Ace: Assembly Converter to Extended Basic.
Reformat: Reform Dis/Var 80 files (vev)
Pre Setter: A Epson printer pre-setter.
Print Graph: Prints a Graph sheet.
Mag Demo: A example of Magnify program.
Magnify: A excellent utility program.
Stylometry: Not sure up-to-date what it is but it looks like it is a good program.
Population: Will work out when we will end up with one square of land per person. At a growth rate of 10% per year, about 20 years. Interisting program.

| | | | |
|---|---|---|---|
| ACE/ASM | 29 Prog | LOAD | 5 Prog |
| LOADREFORM | 13 Prog | MAGDEMO | 6 Prog |
| MAGNIFY | 12 Prog | MICRO-E-M | 8 Prog |
| POPULATION | 3 Prog | PRESETTER | 13 Prog |
| PRINTGRAPH | 2 Prog | PRINTINST | 4 Prog |
| REFDATA | 11 d 80 | REFINST | 51 d 80 |
| REFORM | 20 Prog | REFORM/O | 47 D 80 |
| REFSUBS | 25 d 80 | ROOT | 28 Prog |
| STYLOAD | 2 Prog | STYLOMETRY | 53 i254 |

END OF ARTICLE

# LEARN TO KNOW YOUR TI

## LESSON 17

With Percy Harrison

This lesson deals with colour graphics and, in TI BASIC, will let you put up to 15 colours on the screen at once. Before starting this lesson you should note that although we spell colour with a "u", as per the English dictionary the computer does not recognise this word as it uses American spelling and consequently looks for the word "color" not "colour". Hence when using colour in a program you must spell it "color".

CALL COLOR(char.set,foreground, background)

is a command that colors a whole set of 8 characters at once. There are 16 sets. The characters are numbered from 32 to 159. Set 1 is from 32 to 39. set 2 the next 8 characters, etc. Normally, the characters are the ASCII set listed in Appendix of your TI User's Reference Guide page III-1.

Because you assign two colours to each character set, a foreground colour and a background colour, you have an immense number of colour combinations.

You can choose a "transparent" colour which lets the background colour, assigned by a CALL SCREEN() command, show through in either the character foreground or background. (If transparent is used for the foreground, you have a "reversed" character).

The background normally only shows on the border because the "field", where the characters are printed, usually starts out filled with the character 32, the blank, which belongs to set 1.

The CALL COLOR command recolours all the characters in the set: those yet to be placed on the screen and THOSE ALREADY ON THE SCREEN!

Colouring in sets and changing those already on the screen may sound dreadful but, properly used, you can draw almost anything you wish.

You will assign a different colour pair to each af 16 character sets at the beginning of a program. Then you will redefine characters in each set until you can make the pictures you want in the colours you want. You could even reassign colours or characters in the body of the program to obtain special effects, but this may complicate the program somewhat so we will give this a miss and get on with the show.

### LESSON 17 COLOUR GRAPHICS

The TI 99/4A can put 15 colours on the screen at once.

### THE COLOURS ARE NUMBERED

| | |
|---|---|
| 1 TRANSPARENT | 9 MEDIUM RED |
| 2 BLACK | 10 LIGHT RED |
| 3 MEDIUM GREEN | 11 DARK YELLOW |
| 4 LIGHT GREEN | 12 LIGHT YELLOW |
| 5 DARK BLUE | 13 DARK GREEN |
| 6 LIGHT BLUE | 14 MAGENTA (PURPLE) |
| 7 DARK RED | 15 GREY |
| 8 CYAN (BLUE-GREEN | 16 WHITE |

I will explain colour number 1 in a later lesson.

### ADJUSTING YOUR TV SET

When you first turn on the computer (or when you press FCTN QUIT) you see two rows of coloured rectangles on the screen.

Adjust the TV controls to make the colours look realistic.

You should see all the colours in the list above, especially a good "yellow".

### PICKING THE BORDER COLOUR

Pick a colour for the background of the whole screen.

Run:

```
10 REM COLORS
15 CALL CLEAR
20 CALL SCREEN(15)
99 GOTO 99
```

We picked colour 15, which is grey.

Press FCTN CLEAR to end the program.

### CHARACTER SETS

Here are the first three character sets:

| Set 1 | Set 2 | Set 3 |
|---|---|---|
| 32 space | 40 ( | 48 0 |
| 33 ! | 41 ) | 49 1 |
| 34 " | 42 * | 50 2 |
| 35 = | 43 + | 51 3 |
| 36 $ | 44 , | 52 4 |
| 37 % | 45 - (minus) | 53 5 |
| 38 & | 46 . | 54 6 |
| 39 ' | 47 / | 55 7 |

The rest of the character sets are listed in the TI USER'S REFERENCE MANUAL Page III-1.

Each character has a number, called the ASCII number.

(ASCII is pronounced "ask-key").

Important! When you choose colours, all the characters in a set are given the same colour.

PAINTING CHARACTER SET 2

The command is:

```
CALL COLOR(set,character colour,background colour)
```

Add this to the previous program:

```
30 CALL COLOR(2,9,2)
```

This means:

```
30 CALL COLOR(set2,red characters,black background)
```

So each character of set 2 looks like:

a red character
on a little black rectangle.

PUT THEM ON THE SCREEN

Use the command CALL HCHAR and CALL VCHAR:

Now add to the above program:

```
40 REM PRINT SET 1 AND 2
41 FOR CH = 32 TO 47
45 CALL HCHAR(10,CH-31,CH)
50 FOR T=1 TO 300
51 NEXT T
60 NEXT CH
```

And run it. You will see all the characters of set 1 and set 2. The characters of set 2 are red inside black rectangles.

MOVING PICTURES

Run:

```
10 REM HUMPTY DUMPTY IS SQUARE
15 CALL CLEAR
20 CALL SCREEN(2)
25 REM--------MAKE A SQUARE CHARACTER
26 CALL CHAR(42,"FFFFFFFFFFFFFFFF")
27 REM 16 LETTER F'S
30 FOR C=3 TO 16
35 CALL COLOR(2,C,2)
40 FOR J=1 TO 23
50 REM--------ERASE THE OLD SQUARE
51 CALL HCHAR(J,C*2-2,32)
55 REM--------DRAW THE NEW SQUARE BELOW
56 CALL HCHAR(J+1,C*2-2,42)
60 NEXT J
70 NEXT C
```

Notice these things:

Line 26 makes a new character, number 42. It is a rectangle. The old 42 was a star. The command CALL CHAR() makes the new 42.

You will learn how to make any character you want in a later lesson.

Line 35 keeps changing the colour of the characters in set 2. When the colour changes, ALL the characters in that set change colour, EVEN THE ONES THAT WERE PUT ON THE SCREEN EARLIER!

Line 50 erases the old square before line 55 draws the new one. It looks like the square is falling.

When the program ends, character 42 is changed back into a star.

THE SPACE CHARACTER IS IN SET 1

Run:

```
10 REM THE CHARACTER FIELD
15 CALL CLEAR
16 PRINT "! #$%'&*()+"
20 CALL SCREEN(14)
30 FOR C=3 TO 16
35 CALL COLOR(1,2,C)
36 PRINT C
40 FOR T=1 TO 500
41 NEXT T
50 NEXT C
```

Character 32, which is a space, is in set 1.

Most of the screen is usually "filled" with this blank character so most of the screen changes colour if you change the colour of set 1.

Assignment 17:

1. Write the name of your favourite movie and movie star and then make them change colours on the screen.

2. Now draw a "theatre marquee" around the names and have it flash a different colour.

3. Change the HUMPTY DUMPTY program so that the square moves across the screen instead of down.

4. Write a program to draw "Sinbad's Magic Carpet", a rectangular pattern of coloured dots on the screen.

ANSWERS TO LESSON 16

Assignment Question 16-2

```
10 REM I GOT YOUR NUMBER
15 CALL CLEAR
20 PRINT
25 PRINT
30 PRINT "GIVE ME A NUMBER BETWEEN ZERO AND TEN:"
35 PRINT
36 PRINT
40 INPUT N
46 PRINT
50 IF N>0 THEN 60
51 PRINT "I GOT PLENTY OF NOTHING!"
52 GOTO 20
60 IF N>1 THEN 70
61 PRINT "I'M NUMBER ONE!"
62 GOTO 20
70 IN N>2 THEN 80
71 PRINT "TWO'S COMPANY!"
72 GOTO 20
80 IF N>3 THEN 90
81 PRINT "THREE'S A CROWD!"
82 GOTO 20
90 IF N>4 THEN 100
91 PRINT "THE AWESOME FOUR!"
92 GOTO 20
100 IF N>5 THEN 110
101 PRINT "HANG FIVE!"
102 GOTO 20
110 IF N>6 THEN 120
111 PRINT"SIX THICK THISTLE STICKS!"
112 GOTO 20
120 IF N>7 THEN 130
121 PRINT "SEVEN DAYS OF THE WEEK!"
122 GOTO 20
130 IF N>8 THEN 140
131 PRINT "EIGHT'S REALLY GOT ME BEAT!"
132 GOTO 20
140 IF N>9 THEN 150
141 PRINT "ONLY A CAT HAS NINE LIVES!"
142 GOTO 20
150 CALL CLEAR
160 PRINT "THAT'S ALL, FOLKS.
180 END
```

Assignment Question 16-3

```
10 REM DIGITAL CLOCK
15 CALL CLEAR
20 PRINT "TIME? <H,M,S>"
25 INPUT H,M,S
30 FOR T=1 TO 206
35 NEXT T
40 S=S+1
45 PRINT TAB(12):H:":":M:":":S
50 IF S<60 THEN 105
55 S=0
60 PRIM=M+1
70 IF M<60 THEN 105
75 S=0
80 M=0
85 H=H+1
90 IF H<24 THEN 105
100 H=0
105 GOTO 30
```

To show one time only on the screen try adding line 41 as follows:

```
41 CALL CLEAR
```

Now run the program again.

Assignment Question 16-4

```
10 REM PLAYING CARD GUESSING GAME
15 CALL CLEAR
20 PRINT "TWO PLAYER GAME"
30 PRINT
31 LET P=2
35 PRINT "FIRST PLAYER"
36 PRINT "ENTER A SUIT (C,D,H,S) AND"
37 PRINT "ENTER A CARD (1 TO 13) <S,N>"
40 PRINT "WHILE SECOND PLAYER ISN'T LOOKING"
45 INPUT SS,NS
50 CALL CLEAR
54 PRINT : :
55 PRINT "GUESS A SUIT (C,D,H,S)"
57 PRINT
60 INPUT TSE
65 IF TS=SS THEN 78
70 PRINT "YOUR GUESS WAS WRONG"
72 GOSUB 150
75 GOTO 54
78 CALL CLEAR
79 PRINT : :
80 PRINT "GUESS A CARD (1 TO 13)"
85 INPUT WS
90 IF WS=NS THEN 102
95 PRINT "YOUR CARD NUMBER WAS WRONG"
97 GOSUB 150
100 GOTO 79
102 CALL CLEAR
105 PRINT "BINGO, YOU GOT IT RIGHT"
106 PRINT : :
107 PRINT "YOU HAD";P;"TRIES"
108 PRINT : :
110 PRINT "WANT ANOTHER GAME? <Y/N>"
115 INPUT AS
120 IF AS<>"N" THEN 20
125 END
150 P=P+1
155 RETURN
```

Next month we will take a look at the COMMAND and RUN modes of the computer.

Bye for now.

END OF ARTICLE

## TI-SCSI

The first article is called: SCSI and other four letter words. The file is named: "WORDS" The second article is three files. Its subject is about the SCSI interface and cabling. Files: SCSI1, SCSI2, SCSI3
Enjoy! Please feel free to comment on them pro or con. By the way, the first article was written a year ago and updated last weekend. Second article written about 10 months ago. >>>JDL<<<

SCSI and Other Four Letter Words (well, almost!)
by James D. Lanman

'SP;IN+5
There are several standards for hard drives: SCSI, MFM, IDE, RLL, RJ etc. Of these standards, SCSI, MFM, and IDE are the ones which TI99'ers are most interested in using. In the TI99/4A community, there is room for all of them. However, most of us will be hard pressed to afford a controller card and hard drive for our system. Each standard has its advantages and disadvantages which this article will address. TI99'ers have a difficult choice in picking a standard. The following information should help. SCSI (Small Computer System Interface, pronounced "skuzzy") is the latest standard to come to the TI99/4A. And there is nothing small about SCSI since it is used in many systems ranging from supercomputers to TI99/4A's. The SCSI Host Adapter card is available for $170 (plus S and H and applicable sales tax) from Western Horizon Technologies and Bud Mills Services. It can use any combination of up to 7 SCSI devices. SCSI is not limited to just hard drives. The SCSI Host SCSI card will be able to use CD-ROM drives in the near future once a software driver becomes available. With additional software drivers, it may be possible to use page scanners, magneto-optical drives, and laser printers. Only time, money, and interest from TI99'ers will determine if we see software drivers for these other devices. As for SCSI hard drives, they can be found new, used, or refurbished. 20, 40, and 80 MB size hard drives in 5 1/4" or 3 1/2" form factors can be purchased at a relatively low cost ($50 to $200). They may be a little hard to find, so check out the latest Computer Shopper magazine for where to buy and pricing information. Computer Shopper can be purchased at most large bookstores and computer retailers.

Floppy drive support can be added to the SCSI card by adding the FC-1 card from WHT and BMS for $100. The FC-1 card can only be used with the SCSI card. It supports 360K, 1.44MB, and 2.88MB Floppy Drives. This combination of SCSI hard drives and standard floppy drive support makes SCSI the future of TI99/4A. MFM (Modified Frequency Modulation) is the standard utilized by conventional large floppy drives and older hard drives. And there is an extension of this standard known as RLL (Run Length Limited) which squeezes about 50% more data on the disk. While it is possible to use some RLL drives as an MFM drive, you cannot reliably use an MFM drive as an RLL drive. MFM is an older standard which is now an orphan just like the TI99/4A. Hard drives can be found used, refurbished, or (in rare cases) new at relatively low cost ($50 to $200) in 20, 40, and 80 MB sizes in either 5 1/4" or 3 1/2" form factor. There is still life left in this standard. The HFDC card originally from Myarc, may in the near future be re-issued by Cecure Electronics. They also offer repairs and upgrades of Myarc products at a reasonable cost. If you have an older HFDC card, you should check to see if you have a 32K SRAM, H11 EPROM, and 9216B chip. If you only have an 8K SRAM and 9215 chip, get them replaced. Check with Cecure Electronics on upgrading your HFDC. A new EPROM, probably H13 or H14, which fixes most of the problems with the HFDC and will add 1.44 floppy drive support. It should be available in the near future. A new disk manager, UniManager, will provide 1.44 MB floppy drive and back-up tape drive support. It is being developed by Mike Maksimik and may be available from Cecure sometime next year. If you already have an HFDC or are planning on getting one, there is light at the end of the tunnel!

Finally, we come to IDE (Integrated Drive Electronics), a very popular standard in the PC community. Drives are easy to find and relatively inexpensive. If all you want is to add a hard drive to your TI99/4A, this would be a great way to go. Unfortunately, an IDE controller only exists as a prototype and it is unlikely that it will ever go into production. An IDE controller card would use conventional floppy drives just like the HFDC does using a DMA (Direct Memory Access) scheme. IDE, for now, is just a dream for TI99'ers. Whichever standard you choose, I hope this article will make that choice easier. We have a dedicated core of TI99'ers working on improving the hardware and software that we can use. But it takes time, money, and your interest in their work to make it happen. We, TI99'ers, need more programmers. Get involved, learn Assembly or C. Start writing programs you want to see! Do not be afraid to participate. The best is yet to come!

## Introduction to SCSI
by James D. Lanman

Well, here is my second article. A long promised and eagerly awaited introduction to SCSI. This is not quite a primer, but rather an introduction for those who are interested about or planning to purchase a Western Horizon Technologies SCSI card. What follows should be useful and informative. At the end of this article, a glossary is included to help explain some of the terms used.

But, first, let us start with my "definition" of SCSI. SCSI is an evolving standard for connecting various types of peripherals to a host (CPU or internal bus). It is a bus specification and command set. The command set (CCS) optimizes use of the SCSI bus. There are three versions of SCSI: SCSI-1, SCSI-2, and the up-coming SCSI-3 standard. Extensions to this standard are: SCSI Wide (utilizing either a 16 bit or 32 bit wide data path) and SCSI Fast (provides for 6 to 10 MHz data transfer rates between a peripheral and host). A SCSI host adapter may be single ended or differential in electrical configuration. Data transfers may be asynchronous or synchronous. Having defined SCSI, it should be known that this versatility in the SCSI standard allows it to be implemented on systems ranging from the TI99/4A to supercomputers. Many different types of peripheral devices can be linked to a host by the SCSI interface. A device driver (software) will be needed to use a specific device. Some of the SCSI devices are: CD ROM, floppy, hard, magneto-optical, tape back-up, and WORM drives, as well as, scanners and laser printers. Each type and/or make of device will require a separate driver.

While all the necessary drivers to run SCSI floppy and hard drives will be included with the WHT card, others will have to be written. Other drivers will be made available, later, at a reasonable cost. Finally, having covered many of the SCSI device types, we come to the SCSI cable and the features of the WHT SCSI card. The cable will need to be a 50 conductor twisted pair/stranded type ribbon cable. A 50 pin female dual row ribbon connector will be needed for connecting to the WHT SCSI card. This is similar to the IDC type connector. For each peripheral, you will need a 50 conductor female centronics type ribbon cable connector. See Table 1 for SCSI pin-out information. A cable six feet or less in length is desirable. In any case, a cable should not exceed 6 metres (about 6.5 yards). Generally, cables are more susceptible to interference as their length increases. This information should be helpful to those who may want to make their own cable. In addition, SCSI peripherals use a terminating resistor pack just like older TI99/4A floppy drives. These terminating resistors will need to be removed from all of the devices except for the last device connected to the cable (which is farthest on the cable from the SCSI controller). Save the terminating resistor packs and store them in a safe place. Should you need to change your configuration you will have prevented any problems arising from a lost or discarded terminating resistor pack.

The WHT SCSI card can use up to 7 devices (actually 8, if bus arbitration is turned off). Any SCSI floppy drive, 5.25" or 3.5", with capacities up to 4 megabytes. Any SCSI hard drive with a capacity up to 2.7 gigabytes! Any combination up to a total of 4 GB. All current TI99/4A floppy formats will be supported as well as PC compatible floppies. PC TRANSFER(c) will be built in on the WHT card. No more converting of files to or from a PC. You will be able to connect a CD ROM drive for access to pictures and sound. Drivers for hard drives and floppy drives will be available; and a SCSI version of Unimanager by Mike Maksimik. A basic driver for CD ROMs is in the planning stages. Any SCSI device running at rates from 1 MHz to 5 MHz may be used with a proper driver. These devices are connected in a daisy chain fashion with a ribbon cable. This is just the beginning. But the real power of the WHT SCSI card will become most apparent when combined with the 4/A Memex card (available soon). This article has barely covered some of the basics of SCSI. It would take a series of articles to really explain this new standard for the TI99/4A. I would like to thank Don O'Neil of WHT and the Ancot corporation for their invaluable assistance in writing this article. If there is enough interest, perhaps I will do a more in-depth series on SCSI. But, for now, my next article will be "Curse of the Information Age: Carpal Tunnel Syndrome". As we all learned in last months' newsletter, Barry Traver has been stricken with this affliction. I will be writing about my own experiences and what you can do to prevent this scourge from striking you. The SCSI card will be available soon from Western Horizon Technologies and Bud Mills Services. Write or call them at:

Western Horizon Technologies
Don O'Neil
10225 Jean Ellen Drive
Gilroy, CA 95020

Bud Mills Services
166 Dartmouth Drive
Toledo, OH 43614-2911
(419) 385-5946

---

Table 1 -- Single-Ended SCSI Interface

| pin # | Descriptor | pin # | Descriptor |
|---|---|---|---|
| 1 | GND | 26 | TRMPWR |
| 2 | D0- | 27 | GND |
| 3 | GND | 28 | GND |
| 4 | D1- | 29 | GND |
| 5 | GND | 30 | GND |
| 6 | D2- | 31 | GND |
| 7 | GND | 32 | ATN- |
| 8 | D3- | 33 | GND |
| 9 | GND | 34 | GND |
| 10 | D4- | 35 | GND |
| 11 | GND | 36 | BSY- |
| 12 | D5- | 37 | GND |
| 13 | GND | 38 | ACK- |
| 14 | D6- | 39 | GND |
| 15 | GND | 40 | RST- |
| 16 | D7- | 41 | GND |
| 17 | GND | 42 | MSG- |
| 18 | DPAR- | 43 | GND |
| 19 | GND | 44 | SEL- |
| 20 | GND | 45 | GND |
| 21 | GND | 46 | C/D- |
| 22 | GND | 47 | GND |
| 23 | GND | 48 | REQ- |
| 24 | GND | 49 | GND |
| 25 | OPEN | 50 | I/O- |

---

## Glossary

ARBITRATION -- The process of selecting one respondent from a collection of several candidates that request use of the SCSI bus concurrently.

SYNCHRONOUS TRANSMISSION -- A transmission in which each byte of information is synchronized individually, through the use of interlocking the REQ and ACK signals.

BIT -- BInary digiT, which can have a value of 0 or 1. It is the smallest unit of data that a computer can process. Bits are arranged into groups of eight called bytes. A byte is the equivalent of one character.

BUS -- A signal path or line shared by many devices. Information is often sent to all devices throughout the bus; only the device to which it is addressed will accept it.

CCS -- Common Command Set. CCS is a collection of 18 commands, which is a subset of SCSI-1. SCSI-1 specification allowed too many vendor specific features. CCS was designed to improve compatibility between SCSI devices from different vendors. CCS is included in SCSI-2.

CPU -- Central Processing Unit is the nerve center or brain of the computer. It interprets programs and tells the computer how to execute them.

DIFFERENTIAL INTERFACE -- An electrical signal configuration using a pair of lines for transfer. On a SCSI bus" 'TRUE' (logical '1') is defined as -SIGNAL (about 1 Volt) higher than +SIGNAL line, and opposite polarity for 'FALSE'. The advantage of differential configuration (as compared to single-ended) is in relatively higher tolerance for common-mode noise and little crosstalk when used with twisted pair cables. It allows for connections up to 20 meters (about 22 yards). Its disadvantage is higher (component) cost.

HOST -- A processor, usually consisting of a CPU and memory. Typically, a host communicates with other devices, such as peripherals and other hosts.

HOST ADAPTER -- Circuitry that translates between a processor's own internal bus and a different bus, such as SCSI. On the SCSI bus, a host acts as an initiator and a peripheral acts as a target.

MHZ -- MegaHertZ (MHz) is a million cycles per second. Used as a measurement of data transfer rate.

PERIPHERAL -- A device that can be attached to a host computer, using a SCSI bus for example. Typical types of peripherals are floppy disk drives, hard disk drives, etc.

SCSI -- Small Computer System Interface. Pronounced "Skuzzy". An industry standard for connecting peripheral devices and their controllers to a microprocessor. SCSI defines both hardware and software standards for communication between a host computer and a peripheral.

SINGLE-ENDED INTERFACE -- An electrical signal configuration using a single line for each signal, referenced to a common ground path. The advantage of a single-ended configuration (as compared to differential) is in using half the number of pins, chips, and PCBoard area. Its disadvantage is higher susceptibility to common mode noise and a limited cable distance.

SYNCHRONOUS TRANSMISSION -- A transmission in which the sending and receiving devices operate continuously at the same frequency and are held in a desired phase relationship by a correction device. For buses, synchronous transmission is a timing protocol that uses a master clock and has a clock period and allowable offset.

WORM -- Write-Once Read-Many. An optical storage device on which data is permanently recorded. Data can be erased, but not altered or additional data added.

==========

END OF ARTICLE

# PROCEDURE TO CONFIGURE PRINTER USING FUNNELWEB AND DM1000.

By Fred Moore LA 99ers Topics.

FUNNELWEB V4.10 By John Owen JUD UG TX

1. Make a copyof fwb. 4.10 Leave off the write-protect tab and place this new copy in #1.

2. Auto-load FWB up to the first menu (selection 1-9 and I)
3. Select option "I" (Configure). It may also be reached by selecting USER LIST from the Editor menu. Dsk1.CF and CG will load and display "CONFIGURATIO" the prompts: (?) HELP, (c/C)BACK (F-7)DIR.
4. Hit any key. First window: SYSINFO, QUIT, INSTALL.
5. HIT "S" (SYSINFO) and get the second window: LOAD, EDIT, SAVE.
6. Hit "L" (LOAD)
7. Hit (enter). (loads SYSCON FROM DRIVE #1).
8. HIT "E" (EDIT). This will display the third window: LOADING,DEVICES,COLORS,MENU,XB LIST and UL LIST.
9. Hit "D" (DEVICES). This will display the 4th. window: EDIT PRINTER,FMTR PRINTER,OBJECT FILE,WORK FILE,7 PROGRAM.
10 Hit "E" (EDIT PRINTER): The following instruction window will appear: ENTER/FILE DEVICE.
11. TYPE IN YOUR PRINTER NAME: PIO or RS232.BA=####.LF ETC..
12. HIT (enter).

To change the formatter printer the easiest way is to repeat steps 10,11,12 as folows before going any further.

* 10a Hit "F" (FMTR PRINTER)
  11a Type in :PIO.LF or RS232.BA=#####.LF etc..
  12a Hit (enter).

13. Hit FCTN/9 (BACK) or CTRL/C
14. HIT FCTN/9 or CTRL/C again.
15. Hit "S" (SAVE)
16. Hit (enter) to save the change in DSK1.SYSCON
17. Hit FCTN/9 or CTRL/C to get back to the top of the menu.
18. Hit "I" (INSTALL). The sixth window gives two choices.
19. Hit "L" (LOAD-XB/XB11).
20. (enter) SOURCE program (DSK1.LOAD).
21. (enter) TARGET program (DSK1,LOAD).
22. Hit FCTN/9 or CTRL/C.
23. HIT Q (QUIT).
24. Test from scratch by reloading FUNNELWEB.

END OF ARTICLE

# Techo Time

## More on battery backups

With Geoff Trott

Another correspondent of mine, whom I have not yet met, is Jim Banfield of Armidale. He is a hardware person and wrote a series of articles on assembler for the TND. He was also stimulated by my article on problems with backing up memory chips with batteries. He has his own version of debug in a module like MiniMemory and has had problems with that retaining memory in the module port and the PEBox. I have had similar problems with a modified MiniMemory module which has had its RAM memory expanded to 32K (switchable in 8K banks) and an Editor Assembler GROM added. I will leave my solution until after presenting Jim's information. He writes: "Having noted that the MiniMemory module is quite stable, I determined to find out how TI did it, so I removed the back and traced the circuit with my faithful Fluke. It is very difficult to be sure that all connections are found without stripping the board so that the attached circuits are provisional. From the complexity of the circuitry and the expense of manufacture I conclude that Texas also had trouble with volatility. The unconventional interface of the 74LS32 is noteworthy; the rationale of using the pnp pass transistor rather than a diode eludes me. I have not yet had time to try out the circuit but I thought you might like to see it, assuming that you have not done so already. I would certainly appreciate your comments."

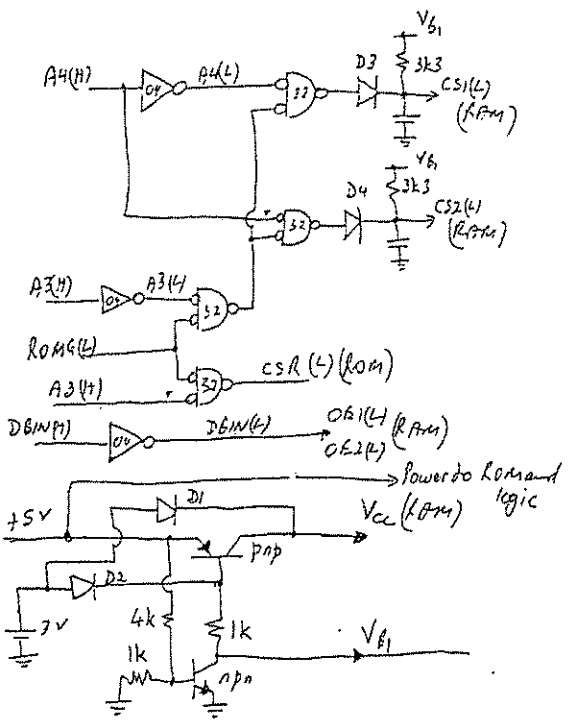
I agree with Jim that the circuit is complex and does work well. I have redrawn it here and will attempt to explain its operation. When the power is on there must be no current going into the battery (or out of it) and the 5 volts must go to the Vcc pin of the memories (pin 24 for the 6116). The CS(L) line (pin 18 for the 6116) of each memory is held high to disable the chip and for the power down mode. If the power off mode is considered first, D1 connects the battery to the Vcc pins of the memory and D2 provides positive voltage to the CS(L) pins via the 1k and 3k3 ohm resistors. The diodes in series with the outputs of the OR gates (D3, D4) stop current flowing into these outputs when power is off. Both transistors are turned off so that no current flows into the base of the pnp transistor nor into the collector of the npn transistor. This puts the memory into its power down mode (more than 2.2 volts on both pins 24 and 18) which means it draws very little current but retains the memory's contents.

When power comes on, D2 prevents base current flowing out of the pnp transistor which stays off until the npn transistor turns on. The voltage divider at the base of the npn transistor means that when the power reaches 3 volts the transistor turns on (5 times 0.6 volts, the turn on Vbe voltage) and provides a path for the base current of the pnp transistor through the 1k resistor into the collector of the npn transistor. This also takes the ends of the 3k3 resistors to 0.2 volts which means that the OR gates must now supply the current through the diodes to keep the CS(L) signals high when the memory is not in use. In this mode, the power is applied to the Vcc pins of the memory through the pnp transistor, whose voltage drop will be 0.2 volts or less. The two diodes from the battery will both be reversed biased by this time. D2 will be reversed biased as the emitter of the pnp transistor will be at 5 volts which means its base will be at 4.4 volts. In other words, all the current through the 1k resistor comes from the base of the pnp transistor and none through the diode.

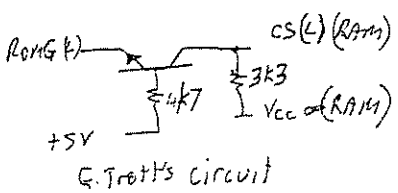
So the npn transistor and the resistor divider at its base controls the turn on and off of the pnp transistor and hence the transfer from power to battery. Using the pnp transistor gives less voltage drop and allows the change over between battery and power to be controlled. The diodes at the outputs of the OR gates are to prevent current going into the gates on battery power which means there must be a resistor to ground when the circuit is being used normally to allow these signals to go low. The small value capacitors at these points are to get rid of the switching transients at the power change over time. The OR gates are used to provide CS(L) signals for the 4K EPROM and the two 2K RAM chips in the MiniMemory module. I have drawn these gates with their alternate symbol as they are all performing AND functions.

When I gutted and rebuilt my MiniMemory module, I did not leave in this complicated circuit as there was not room for it. Also there was no need for the CS(L) logic as the ROMG(L) signal came straight to the CS(L) signal of the RAM chip (6264 for 8K or 62256 for 32K). I then had trouble with the battery discharging when the module was left in the console and power was off. I solved this by inserting an npn transistor in series with the CS(L) line as shown in the circuit below. This allowed the CS(L) pin on the memory to be kept high when on battery supply without any current going into the external circuit as the transistor is only on when the power is on.

I hope this provides some interesting food for thought for all those whose batteries seem to go flat too quickly.




Mini Memory circuits




G. Trott's circuit

END OF ARTICLE

# BRAILLE'N SPEAK.

By Irwin Hott.

BRAILLE 'N SPEEK: A NEW COMPUTER FOR THE BLIND.

I am actually beginning to write this article while riding a COTA bus downtown.
I am using the Braille 'n Speak. It is about 8 inches long 4 inches wide and 1.5 inches high.It weighs less than 1 lb.
The small size is made possible with use of a braille keyboard and a speech synthesizer for the "display". The Braille'n Speak has rechargable batteries as well as an RS232 port.
The braille cell is made up of 6 dots 1,2,3 from the top to bottom on the left,4,5,6 on the right. The unit uses 7 keys (1 for each dot plus the spacebar). You simply press dots simultaneously to get the desired character. Dot 1 is A and 2 is B, dots 1 and 4 are C etc..
I don't propose to go into the intricate patterns of Braille here except to say that there are different "grades" involving the number of contractions (abbreviations) used e.g. nec for necessary, AL for also and words such as WITH written as dots 2,3,4,5,6.
I am writing in grade 1 braille now. There are no contractions and I am writing in lower case most of the time. I can switch to upper case by hitting a U chord. That means pressing U and the space bar at the same time. That will give me the next character in upper case. If I hit U chord twice, upper case lock will be on. Most of the commands such as file. cursor and parameter are made by pressing a key combination with the spacebar.
The Braille 'n Speak has about 200k of RAM. About 180 of that may be used for file storage. Files may be as short as 1 page (4096 characters) or up to 45 pages. The maximum number of files allowed is 30. Right now I have 6 files open in Braille 'n Speak with 40 pages of memory remaining. I have a 2 page file for this article; notes from Lima, a phone list, a BBS list, a help file which is always resident and clipboard. Clipboard is a 1 page file that is used to house deleted material as well as data copied from one file to another. I just exited "art" and looked at clipboard. It contamned this article. I could have deleted a line from this file and put it into another file. It is very easy file. It is very easy to move from one file to another. When I do that I return to the exact place where I left the file. Many features such as these make the unit a joy to use.
There are several word-processing functions built in. Right now I have key echo turned on. However my brailing is faster and I am ahead of speech most of the time. There is a backspace command which is destructive. I can move through text, a paragraph, line, word or character at a time in either direction. If I do not understand a character as p or t there is a phonetic alphabet built-in. I can insert up to 255 characters from the keyboard. If I insert from another file I can add up to 4069 characters in one move. I can also delete anywhere from one character to the entire contents of the file. I can set a "mark" in text and delete to that "mark" in either direction. One of the minor drawbacks is in Replace String. I can overwrite a character or find a string but I cannot replace all occurences of a string. It is very easy to transfer material to and from Braille 'n Speak. The RS232 port is controlled by software commands. I can set Baud rate from 75 to 19,200 set parity, duplex handshaking and stop bits. I can transmit complete text, text to "mark" a character, line or paragraph. If I want to receive text, all I have to do is open a file, set the parameters in the RS232 port and turn on the RS232 port. All incoming material will be stored in the open file.
I can listen to the material as it comes in or just let it build up in RAM. I frequently dump files from the TI so I can listen to them in the Braille 'n Speak. It is much easier to read text here because I can skip around in the file and carry the machine around with me. As an example I dumped a series of messages from HUG TI BBS in Houston about using 3.5 inch disk drives. I thought it would make an interesting file on Spirit of '99, so I edited it in the Braille 'n Speak This took about 10 minutes to edit out all of the extraneous information. On the TI without being able to use TI-WRITER or the equivalent it would have taken at least 45 minutes. I was able to quickly searglSfovS as message numbers. I entered a note at the beginning of the file, merged the new file description into the old description file dumped both from the Braille 'n Speak to the TI and I was ready to go. I can format the text I am sending from the Braille 'n Speak. It can be formatted as to page length, line length, left margin and top margin. There is no way in writing text (such as this article) to specify a line length. I will format the file when I send it to the TI. It would be nice if I could set a line length and have a warning if I was approaching the end of the line. However that is a relatively minor drawback.
The Braille 'n Speak has a clock built-in. The current time is 12:16. There is a calendar, a timer and a four function calculator. I can "paste" answers to calculations into a file such as I did with the time above.
Now that I have the Braille 'n Speak I wonder how I ever got along without it. The program (using a 512k eprom) has been carefully written to make it as easy to use the device as possible. Much careful thought has gone into it. Not that it was easy to learn. I have not counted but I suppose there are at least 50 new commands I had to learn. For the first couple of days I wondered if I was ever going to master it. After that it started to get much easier. There were also some bugs in earlier versions of the eprom. I was one of about half a dozen people who helped test some of the up dates. That was for the most part a lot of fun. However it was not without dangerous moments. A couple of times I ERASED MEMORY! Once through my own carelessness, the other time an error in the program. Fortunatly I had backup copies of most files on the TI. i have not found any errors in this new version of the eprom.
I am not really sure I can explain how nice it is to have a device such as this. Yhe possibilities of use are just about endless. It can be used for phone messages, recipts, editing programs and so much more. The cost is reasonable at SUS895.00, which for a high tech low production device is fairly unusual. As an example the first talking calculator cost S495. Now for the first tmmeSflm* at least, on a par with those used by their sighted counterparts. Previously, portable lap-top computers with speech cost at least S2000. This put it out of price range for many individuals. It is not absolutly necessaary to interface Braille 'n Speak with another computer. There is a tape interface device available as an option. It works through the RS232 port.
I hope this gives you a little idea of just how I use the Braille 'n Speak.
Contact-: BLAZIE ENGINEERING
2818 COLLEGE VIEW DRIVE
CHURCHVILLE, MD 21028
UNITED STATES OF AMERICA.
0011 1 301 879-5504
I would also be glad to correspond with anyone who would be interested.

MY ADDRESS-:
Irwin Hott
1540 Northbridge Road
Columbus, OH 43224
0011 1 614 263-5319

Retyped from:
Spirit of 99, June 1988, by John Ryan for TEXPAC BBS.

END OF ARTICLE

## WE HAPPEN TO BE MEN, ONLY MEN

By Larry Saunders

If we put a woman on a pedestal and try to protect her interest from the harsh world about, we are male chauvinist pigs. If we stay at home and do the house-work, we are pansies. If we work hard to get a good living for her, there is never any time for her, or the children. if we don't work hard enough, we are good-for/nothing-lay-abouts. If she has a boring, repetitive job with low wages, that is exploitation. If WE have low wages, we should get off our butts and find something better. If we get promotion ahead of her, that is favouritism. If she gets promotion ahead of a man, that is equal opportunity.

If we mention how nice she looks, that is sexual harassment. If we say nothing, that is typical male indifference. If we cry, we are sheilas. If we don't, we are insensitive bastards. If we thump her, that is wife-bashing. If she thumps us, that is self-defence.

If we make a decision without consulting her, we are chauvinist go-getters. If she makes a decision without regard for our feelings, then she is a liberated woman. If we ask her to do something she doesn't like or enjoy, that is domination. If she ask us to do some-what the same, it is a favour.

If we appreciate the female form, and, frilly underwear on a woman, we are sexual perverts. If we do not notice them, we are poofters. If we like her to keep in good shape over-all, shave her legs, have nice hair, well applied make-up, we are sexist. If we don't care, we are unromantic dead fish. If WE try to keep ourselves in good shape, that is vanity. If we don't we are slobs, and Norms.

If we buy her flowers, we are after something. If we don't, we are forgetful. If we are proud of what we have accomplished, we are conceited cruds. If we aren't we have no ambition. If we ask for a cuddle. we never think of anything but sex. If we come home from a hard day at work, and she wants a cuddle, and doesn't get it, we never give a stuff about other peoples needs.

If she has a headache, it's because she's tired. If we have the same type of headache, it's because we don't love her any more. If we want It too often, we are over-sexed. If she wants IT, and we can't perform on cue, there must be someone else.

END OF ARTICLE

## TREASURER'S REPORT

by Cyril Bohlsen

| | | |
|---|---|---|
| Income for previous month ..... | $ | 360.00 |
| Expenditure for previous month .. | $ | 396.19 |
| Loss for previous month ..... | $ | 36.19 |

Membership accounted for $ 165.00 of Income.

Shop sales ..... $ 195.00 of Income.

The expenditure was made up of the following

| | | |
|---|---|---|
| Printing & Postage of TND ..... | $ | 283.69 |
| Purchase of B/J PRINTER INK ..... | $ | 112.50 |

## LOADING PROGRAMS AND FILES IN THE TI99/4A UNDER EXTENDED BASIC.

by Ross Mudie.

1. INTRODUCTION.

This article discusses various methods of loading programs and files under Extended basic. It is intended for people who want to write their own programs. Most users are familiar with loading programs from tape and disk but not all understand the principles of using menu programs in extended BASIC and chaining programs together.
In Extended Basic if a disk system is connected to the computer and when extended basic is first selected, after the initial master screen, extended basic automatically checks disk drive 1 for a program named LOAD. If a LOAD program is found it is loaded into memory and run. The LOAD program can be written to perform what ever the writer wants; this article discusses a few of the common uses of LOAD programs.
This article does not explore the system used in the Horizon Ramdisk which can avoid the TI colour bars master screen.
Extended Basic also contains a sub program named CALL LOAD which can be used to load assembly files from disk or "poke" information into CPU RAM locations.

a) The LOAD program can be specially set up already containing the names of available programs on the disk which can be displayed in a menu list. The required program can then be selected from the menu with automatic load and running.
b) The LOAD program can search the disk drives in the computer for other Basic and Extended Basic programs and build up a menu of programs. The program can "filter out" the files etc and can then allow selection of the required program with automatic load and running.

c) The LOAD program can just be a 2 liner which loads very quickly and displays on the screen that the required program is being loaded. This may load the single dedicated program.

d) A program can load data from a file. A data file can be created from an extended basic program or an EDITOR program. Records from the data file can be loaded into variables or arrays for use in the program.
When a program becomes too big to fit in memory along with the data which is generated or used by the mntoSt5which can be chainedal to break the program up together can use data from a common file.

e) Extended Basic programs can load assembly language routines which can be linked into from the extended basic program.

f) When many of the TI and third party games were "cracked" from cartridge to disk format they were in Editor/Assembler option 5 format. These could be loaded by the Editor/Assembler using option 5 or loaded under Extended Basic using "an Option 5" loader.

g) Extended Basic programs don't necessarily have to be just Extended Basic. Programs may be assembly language inside an Extended Basic program with special inbuilt assembly language loaders which may change the loading format. The "Implanted Assembly" Extended Basic programs can be loaded from cassette tape or disk drive and chained together.

h) The program named LOAD does not necessarily have to be a loader, any program can be named LOAD and it will load and run after extended basic is selected.

2. PROGRAM EXAMPLES.
These program examples are the minimum no frills versions to show the principles.

a) LOAD WITH PROGRAM NAMES IN THE LOAD PROGRAM.

```
100 ! SAVE DSK1.LOAD
 110 DISPLAY AT(6,7)ERASE ALL BEEP:"LOAD EXAMPLES 1.":
 :"PRESS 1. FIRST PROGRAM": : "      2. PROGRAM 2": :"
3. THIRD PGM"
120 CALL KEY(5,K,S):: IF S=0 THEN 120
130 IF K<49 OR K>51 THEN 110
140 ON K-48 GOTO 150,170,190
150 DISPLAY AT(20,1):"Running DSK1.FIRST_PGM"
160 RUN "DSK1.FIRST_PGM"
170 DISPLAY AT(20,1):"Running DSK1.PROGRAM_2"
180 RUN "DSK1.PROGRAM_2"
190 DISPLAY AT(20,1):"Running DSK1.THIRD_PGM"
200 RUN "DSK1.THIRD_PGM"
```

Line 100 is used to name the program and it can be used to save the program in extended basic by pressing 100 <FCTN> X <ENTER>, <FCTN> 8 (Redo), then edit off the 100 ! and press <ENTER>. This leaves the SAVE DSK1.LOAD which saves the program.

Line 110 Clears the screen, beeps and displays the menu list.

Line 120 Waits for a key to be pressed.

Line 130 Tests the key pressed and rejects any invalid key.

Line 140 Adjusts the ascii key value in variable K to a value into the numeric range of 1 to 3 and branches to the required line number.

Lines 150 to 200 are in pairs for the required program. The first line of the pairs shows on the screen what isSefoutSof the pairs loads and runs the required program. These lines with RUN contain QUOTED STRINGS which were included in TI's planned facilities for Extended Basic. (This function does not allow RUN to be used with the contents of a string variable, only a direct quoted string). If the program can't be found then an error will be given.

b) LOAD WHICH SEARCHES THE DISK DIRECTORY.

There is nothing special in this program, it is based loosely on the program in the TI Disk Memory book page 41. It reads the catalog on the disk, saving only the program names in an array and printing these on the screen. The program only loads the first 21 program names. This limit was set to keep the program simple, (just one screen). When a program is chosen its name is combined with the characters "DSK1." and the computer shows what it is about to do. TI Extended Basic is unable to RUN the contents of a variable, it will only run a quoted string as seen in the previous example, so another technique is required. The last line in a program is easy to find, so the last line is set up with a dummy program name in a format which reserves the maximum possible required space. The program then finds the location in memory of the last program line and byte by byte, pokes first the length of the required program name, then the device and program name, then a zero to finish off the program line in the normal way that extended basic is stored. Once this is complete, the Extended Basic program has actually modified itself and the modified line is then run.

```
100 ! SAVE DSK1.LOAD
110 DIM PROGRAMSS(20)
120 CALL INIT
 130 DISPLAY AT(3,1)ERASE ALL:"SEARCHING DISK
DIRECTORY"
140 OPEN #1:"DSK1.",INPUT ,RELATIVE,INTERNAL
150 INPUT #1:AS,D,S,K
160 PRINT "Disk name ";AS:"Size";S;"Free";K;"Used";S-K
170 C=65
180 IF C=86 THEN 210 ELSE INPUT #1:AS,A,J,K
 190 IF ABS(A)=5 THEN PRINT CHRS(C);" ";AS ::
PROGRAMSS(C-65)=AS :: C=C+1 :: GOTO 180
200 IF AS="" THEN 210 ELSE 180
210 DISPLAY AT(24,1)BEEP:"Press letter of choice"
220 CALL KEY(3,K,S):: IF S=0 THEN 220
230 IF K<65 OR K>=C THEN 210
240 PS="DSK1."PROGRAMSS(K-65)
250 DISPLAY AT(24,1):"Load  Run ";PS
260 CALL PEEK(-31952,A,B)
270 CALL PEEK(A6+B-65534,A,B)
```

END OF ARTICLE

## GRAPHICS PROGRAMMING LANGUAGE . . .

The following is extracted from a 1979 TI Manual, in which it forms Chapter 1:

1.0 GRAPHICS PROGRAMMING LANGUAGE

The System software resident in the product consists of a monitor and a GPL (Graphics Programming Language) processor. It is the function of the monitor to insure that every time the system is turned on, a new cartridge is inserted, or an existing program terminates, that all memory and peripheral devices are initialized. The GPL processor is an interpreter optimized to execute GPL programs directly out of GROM. The GPL processor software is coded in TMS 9900 assembly Language. NOTE: See Appendix N.

1.1 OVERVIEW

GPL is a programming language specially developed by Texas Instruments to provide the best possible tradeoff of code compaction, execution speed, and ease of program development for the target computer system. The GPL instruction set facilitates development of programs which make use of the unique features of the system chip set. It is byte oriented, and instructions typically have one or two operands. The addressing scheme is such that most instructions can access either standard microprocessor RAM, GROM, or the video scratchpad RAM address space easily.

Most instruction operands can be either single or double byte values. The addressing modes are: immediate, direct, indirect, indexed, indexed indirect (with pre-indexing), and 'top of stack'. Source operands and destination addresses can be in the CPU, video RAM, or in GROM. Support for two stacks is available; a data stack and a subroutine return address stack (allowing arbitrary nesting of subroutines).

1.2 GPL INSTRUCTION SYNOPSIS

GPL has the following types of instructions:

```
*DATA TRANSFER     -single or double byte transfers
-block to block transfers
-formatted block transfers
*ARITHMETIC        -add, subtract, multiply, divide,
-negate, absolute value
*LOGICAL           -and, or ,exclusive or, shifting
*CONDITION TESTS   -arithmetic and logical tests
*BRANCHING         -unconditional and conditional
*BIT    MANIPULATION  -wet,Sve%9   yyyyyycall,   return,
 parameter fetching
 *STACK OPERATIONS  -push and pop
 *MISCELLANEOUS     -random number  generation,  keyboard
 scan,
 -coincidence detection, pattern movement,
 -sound control,
 -TMS 9900 subroutine linking, I/O
```

1.3 GPL TIMING

The GPL interpreter contains an interrupt driven service routine which is tied to the video scan. Video symbols may be moved about the screen automatically; also sounds may be generated from a sequence table.

These are of the "set it and forget it" type of instructions which free up the control program to do concurrent decision and computational operations. The interrupt also controls a software real time clock.

Each system will have a clock byte reserved in the console ROM at location >000C to indicate the clock rate for that system. Peripherals may read this byte to adjust their timing interface to the CPU's clock combinations in different consoles. The high nybble contains the integer frequency in megahertz and the low nybble, the fractional frequency.

1.4 GPL ASSEMBLER

The TI assembler for GPL (GPLASM) is written in a mixture of FORTRAN and assembly language and is currently available for installation on 990/10 OS minicomputers. The assembler provides standard features such as creation of a list file, cross reference tables, and error flagging. A set of macros is included to help structure GPL programs; these include statements such as: REPEAT ... UNITL and IF ... THEN ... ELSE. The output of the assembler is a 990 object module.

1.5 SOFTWARE MONITOR RECONFIGURATION

The monitor code is executed whenever a system restart is required. The system parameters and control values are initialzed to default values. A default character set is loaded into the video pattern generator, making it immediately available to GPL programs. This pattern set consists of 64 ASCII characters, including the upper-case alphabet, digits, arithmetic symbols, and punctuation symbols.

The monitor is also responsible for determining the existing system configuration. The power-up monitor must poll add-on I/O peripherals and the "SOLID STATE SOFTWARE CARTRIDGE" to determine which program to execute.

The Home Computer system has been designed to be flexible in ROM or GROM may con tain power-up procedures. These power-up procedures will all be executed allowing for expansion of the power-up routines. A power-up routine may also be replaced by another.

1.6 FOREIGN LANGUAGE SCREENS

GPL code has been included in GROM 0 to allow a plug-in GROM to "translate" the main screen, the menu screen, and the cassette DSR messages to alternate languages. The main screen and the menu screen are "translated" after the screen has been formatted in English but while the screen is turned off (only the background color is visible on the screen). At this time, the plug-in GROM is checked for a negative version number (byte 1 of the GROM). When a negative version is encountered, a GROM routine is called at >6010 for the main screen or >6013 for the menu screen. These locations should contain unconditional branches to the routines in the plug-in GROM that will rewrite the screen in the desired language. These routines may use all of the usual CPU RAM locations (>0 through >6F) and the full facilities of the monitor and interpreter. The routines should end with a RTN instruction.

==========

END OF ARTICLE

## FROM THE BBS

```
*-------------------------------------*
*    Interrupt driven clock program   *
*       By:  D. L. Fitchhorn          *
*            305 Navajo               *
*            Keller, TX  76248        *
*-------------------------------------*

       DEF  START,STOP

INTRPT EQU  >83C4
TIMOUT EQU  >83D6
VDPWD  EQU  >8C00
VDPWA  EQU  >8C02

*      REGISTER NAMES
COUNT  EQU  2
STNMIL EQU  3
HOURS  EQU  4
MINUTS EQU  5
SECNDS EQU  6

       AORG >2800

START  MOV  @RESET,@COUNTR Initialize counter
       MOV  @SCRN,@MYREG   Initial screen location
       MOV  @STN,@MYREG+6  Set Standard time
       CLR  @MYREG+8       hours
       CLR  @MYREG+10      minutes
       CLR  @MYREG+12      seconds
       LI   R0,CLOCK
       MOV  R0,@INTRPT     Tell interrupt where to go!
       RT

STOP   CLR  @INTRPT        Turn interrupt off!
       RT

CLOCK  LIMI 0              Insure no interruptions
       LWPI MYREG          Point to my registers
       DEC  COUNT          Decrement count of 60ths
       JNE  IEXIT
       MOV  @RESET,COUNT Reset the counter
       INC  SECNDS         Update SECONDS
       CI   SECNDS,60      At maximum?
       JNE  ALLDUN
       CLR  SECNDS         Yes
       INC  MINUTS         Update MINUITS
       CI   MINUTS,60      At maximum?
       JNE  ALLDUN
       CLR  MINUTS         Yes
       INC  HOURS          Update HOURS
       C    HOURS,STNMIL   At maximum?
       JNE  ALLDUN
       CLR  HOURS          Yes 00:00:00 MILITARY
       CI   STNMIL,24      Should this be standard?
       JEQ  ALLDUN
       INC  HOURS          Yes 01:00:00 STANDARD

ALLDUN SWPB R0             PUT TIME ON SCREEN
       MOVB R0,@VDPWA      Load VDP address
       SWPB R0
       SOC  @WRB,R0        Set write bit!
       MOVB R0,@VDPWA
       SZC  @WRB,R0

       MOVB @HD(R4),R1     HOURS IN DECIMAL
       SRL  R1,4           Position for left digit
       AI   R1,>9000       Make a hex character (BASIC)
       MOVB R1,@VDPWD      H
       SLA  R1,4           Position for right digit
       ANDI R1,>0F00       IBID as above
       AI   R1,>9000
       MOVB R1,@VDPWD       H
       LI   R1,>9A00         :
       MOVB R1,@VDPWD
       MOVB @HD(R5),R1     MINUTES IN DECIMAL
       SVLSSV5,     =Y R1,@VDPWD       M
       SLA  R1,4
       ANDI R1,>0F00
```

```
       ANDI R1,>0F00
       AI   R1,>9000
       MOVB R1,@VDPWD        M
       LI   R1,>9A00          :
       MOVB R1,@VDPWD
       MOVB @HD(R6),R1   SECONDS IN DECIMAL
       SRL  R1,4
       AI   R1,>9000
       MOVB R1,@VDPWD         S
       SLA  R1,4
       ANDI R1,>0F00
       AI   R1,>9000
       MOVB R1,@VDPWD          S

       SETO @TIMOUT      Reset screen timeout

IEXIT  LWPI >83E0        Restore calling registers
       RT                and return

WRB    DATA >4000        Write Bit
STN    DATA 13           Standard Time
SCRN   DATA >0017        Initial screen Loc
RESET  DATA 59           = of 60ths to count
HD     DATA >0001,>0203,>0405,>0607
       DATA >0809,>1011,>1213,>1415  HD is a table
       DATA >1617,>1819,>2021,>2223  to allow quick
       DATA >2425,>2627,>2829,>3031  conversion of
       DATA >3233,>3435,>3637,>3839  HEX numbers
       DATA >4041,>4243,>4445,>4647  >00 - >3B to
       DATA >4849,>5051,>5253,>5455  decimal 0 - 59
       DATA >5657,>5859
MYREG  BSS  >20
COUNTR EQU  MYREG+4
       END
```

END OF ARTICLE

## DM1000 TO PRINT CATALOG.
### By Fred Moore.

1. Boot up DM1000.
2. On first menu hold dpwn FCTN 3.

Type in the name of your printer (and I do not mean EPSON or STAR) your printer name is usually either PIO or RS232 (and what port your printer is pluged into the RS232 card and then the baud rate your printer is set to). Example PIO that's all that is needed if your printer is connected to the parallel port. If your printer is connected to one of the serial ports RS232/1. or RS232/2 then the baud rate BA=1200. BA=4800, BA=9600 etc.. The complete listing should look like this RS232/1.BA=2400.
A default of just RS232 will assume it is connected to port 1 and set for 300 bauds.

4. Press enter 4 times.
5. Main menu again press 2 Disk Utilities (enter)
6. Press #1 catalog (enter)
7. Type in what disk number you want cataloged. (enter)
8. The disk catalog should now appear on your screen.
9. Hold down the FCTN key and the number 7 key.

Retyped by John Ryan for TEXPAC BBS.

END OF ARTICLE

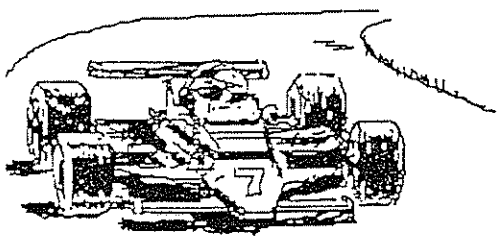
## SIMPLE CALCULATOR

BY Loren West

This programme was written for my children who wanted to check their home work on the computer, "not just an ordinary calculactor," even though that is all it does, they had the fun of using the computer to do it, the programme is simple it could be made a lot smaller in size, now that I have learnt a little extra programming skills along the way. This programme only does one operation at a time because that is what the kids wanted. Please feel free to type it in, change it, adjust it.
Please note that this programme is on the BBS ond has assembly implanted behind it to enhance the characters. If you would like a copy and dont have the equipment to download it please see me, with a blank disk. (ED)

```
100 CALL SCREEN(13):: CALL COLOR(0,16,1):: CALL
COLOR(3,16,1):: CALL COLOR(4,16, 1):: CALL CLEAR
110 DISPLAY AT(1,6):""" SIMPLE CALCULATOR """:
:TAB(10);"By Loren West"
120 DISPLAY AT(5,6):"Please press a number"
130 DISPLAY AT(7,9):"1. MULTIPLY": :TAB(9);"2.
DIVISION": :TAB(9);"3. ADDITION": :TAB(9);"4.
SUBTRACTION": :TAB(9);"5. EXPONENTIATION"
140 DISPLAY AT(17,9):"6. ROOT POWER"
150 CALL CHAR(128,"FF80BFA0A0A0BF8080B6B680B6B680B6FE0
2FA0A0A0AFA0202DADA02DADA0 2DA")
160 CALL CHAR(133,"B680B6B680B6B6FF00000000000000000DA0
2DADA02DADAFE0000000000000 000")
170 CALL SPRITE(#3,128,8,30,1,-1,1,#2,133,8,38,1,-1,1)::
CALL MAGNIFY(3)
180 CALL KEY(0,K,S):: IF S=0 THEN 180 ELSE 190
190 IF K<49 OR K>54 THEN 180 ELSE 200
200 DISPLAY AT(19,1):"press enter after each entry"
210 ON K-48 GOTO 220,310,250,280,340,370
220 CALL LOCATE(#1,45,60)
230 ACCEPT AT(20,2)VALIDATE(NUMERIC):A :: ACCEPT
AT(22,2)VALIDATE(NUMERIC):C :: DISPLAY AT(23,2):"-----"
240 Z=A*C :: GOTO 400
250 CALL LOCATE(#1,80,60)
260 ACCEPT AT(20,2)VALIDATE(NUMERIC):A :: ACCEPT
AT(22,2)VALIDATE(NUMERIC):C :: DISPLAY AT(23,2):"-----"
270 Z=A+C :: GOTO 400
280 CALL LOCATE(#1,95,60)
290 ACCEPT AT(20,2)VALIDATE(NUMERIC):A :: ACCEPT
AT(22,2)VALIDATE(NUMERIC):C :: DISPLAY AT(23,2):"-----"
300 Z=A-C :: GOTO 400
310 CALL LOCATE(#1,60,60)
320 ACCEPT AT(20,2)VALIDATE(NUMERIC):A :: ACCEPT
AT(22,2)VALIDATE(NUMERIC):C :: DISPLAY AT(23,2):"-----"
330 Z=A/C :: GOTO 400
340 CALL LOCATE(#1,110,60)
350 ACCEPT AT(20,2)VALIDATE(NUMERIC):A :: ACCEPT
AT(22,2)VALIDATE(NUMERIC):C :: DISPLAY AT(23,2):"-----"
360 Z=A C :: GOTO 400
370 CALL LOCATE(#1,125,60)
380 ACCEPT AT(20,2)VALIDATE(NUMERIC):A :: ACCEPT
AT(22,2)VALIDATE(NUMERIC):C :: DISPLAY AT(23,2):"-----"
390 Z=A (1/C):: GOTO 400
400 DISPLAY AT(24,1):Z,"ANOTHER Y OR N" :: CALL
LOCATE(#1,180,1)
410 CALL KEY(0,K,S):: IF S=0 THEN 410 :: IF K=89 THEN
100 ELSE STOP
```

END OF ARTICLE



## PASCAL . . . . SNOBOL

By D. Harris

Respected fellow members of TIsHUG:-

I have a free version of PASCAL and a free version of SNOBOL. Once you compile in such languages runnable object code for the masses is achievable.

Those of you STILL struggling with Texas Instruments P-coder might EQUIP to WIN with easy Mystic Pascal.
- yes it does make EXE files.
- it makes programs up to one megabyte long.
In the past TIsHUG has had programmers of genius within the 12k limit of saveable program size.
With Mystic Pascal we can write,
Salesware!
We can copywrite it!
We can do NICHE marketing!
TISHDOS.1 could be the BOOTCOM file we use to start our IBM compatibles.
Summary:
1.Mystic Pascal enables REAL PROGRAMMING.
2.Those facilities for Pascal on the TI-99/4A will come to life for you, enabling POWER that you didn't know how to get from your "old faithful".

END OF ARTICLE

## GRADE BOOK

by Karl Romstedt, Ph.D.

GRADEBOOK 10/86 is an extensive teacher's tool for manipulating grades. Using this software, one can calculate and display student progress at any time throughout the grading period and later produce final grades. The program is user-friendly and self-explanatory, but some helpful hints are found below. This documentation may be viewed while using GRADEBOOK 10/86 (option A).

1. Experiment with all options as there is much more than meets the eye. No changes are permanent unless you save them using the same file name as when they were loaded. The file "GRADES" presently contains a data base for experimentation.

2. When searching for a name or category, only a portion of the name need be entered as long as it is unique. Names may contain commas but the program will limit name length to 15 characters to allow data to fit on the 28 column screen.

3. When new scores are entered (options B or D), the average and standard deviation are calculated excluding zeros so only participating students count. You may wish to write this information down or turn the printer on (option N) prior to entering new scores. Averages and standard deviations obtained later for the same grades will include zero values (option E). If you want to exclude zero grades at a later time, then temporarily delete students with zero scores that would affect the values you want (option G).

4. When naming scores, include a category name (eg. QUIZ 1 or May 9 TEST) if you intend to use weighted categories (option H) or drop lowest scores (option M).

5. With weighted categories (option H), all quizzes (or any other category) can determine a fixed percent of the total grade regardless of the number of points. Weighted scores are not permanent and must be recalculated, if desired, each time a file is loaded or modified. If they are calculated during a session, weighted scores can be used to produce rankings or bar graphs (options I and J).

6. To determine percentage grades for late enrollees, you can temporarily delete those scores they did not obtain (option L).

7. Drop lowest scores applies to one category (eg. drop lowest quiz grade). All of the lowest grades are shifted to one score which is then automatically deleted. Therefore, it may be best not to save a class record after this option since the positions of scores remaining in the affected category will be altered. Lowest scores for each student will have been replaced by higher scores which were originally in the deleted portion of the data.

8. The program is presently dimensioned for 50 students and 20 grades per student. Larger classes may be accommodated if fewer grades/student are needed and vice versa. Dimensions are changed in line 100. Change all 50's to the number of students desired and all 20's to the number of grades/student. Dimension CF(50) to the number of students or the grades/student, whichever is greater. Finally, change 20 and 50 in line 250 as well.

9. Printer options are defined in line 110. D$ is the device ("PIO"). P$ contains printer codes for initialization of the printer. Presently, the codes establish left, bottom and top page margins for Epson FX-80 compatible printers.

OPTIONS:

A. READ DOCUMENTATION
B. START NEW CLASS RECORD
C. LOAD CLASS RECORD
D. ENTER NEW SCORES
E. RECALL SCORES
F. CHANGE SCORES
G. MODIFY ROSTER
H. COMPUTE WEIGHTED TOTALS
I. RANK STUDENTS
J. DISPLAY BAR GRAPH
K. SAVE CLASS RECORD
L. DELETE OR RENAME SCORE
M. DROP LOWEST SCORE
N. SWITCH OUTPUT TO PRINTER

SHARE-WARE

GRADEBOOK 10/86 is supplied free of charge. Use in sales promotions is prohibited. If you find this program useful, mail a contribution to the author:

Karl Romstedt
2543 Cranford Road
Columbus, Ohio 43221

This program is available through the TIsHUG shop.

END OF ARTICLE

# WINDOWS

by Ashley

Windows, that wonderful XT emulator for the 486, has a few useful features that are not often utilized. One of these features is the ability to change its shell.

Have you ever wanted to be able to run a windows program from your DOS menu or the command line and then, when you quit the program, return directly to the menu without having to go through the windows shut down routine? You can and its no hassle at all.

There are three basic ways of running a windows program:

1. You can run windows then open the appropriate program group and then launch the program.

2. You can run the program directly from the command line or a menu using the syntax:-

   WIN [filename.ext]
   If the program (and windows of course) is in the path
   or
   WIN [d:\path\filename.ext]
   If the program is not in the path. (d is the drive)

Both of these methods suffer from the same drawback - you have to shut down windows after you exit the program you want to run in order to get back to your main menu. This can be a real problem if you choose not to use windows as your shell. Fortunately there is a better way!

In order to have windows exit as soon as you close down the program you are using you simply have to make that program your windows shell (this is easier than it sounds). To change the shell that windows uses you must first create an alternative SYSTEM.INI file and then set up a batch file to make use of it.

I shall use Microsoft Works for Windows, which installs itself in a directory called MSWORKS by default, for this tutorial. If you follow the process outlined below carefully you should have no problems doing this for other Windows programs.

1. Make a copy of your SYSTEM.INI file to your DOS directory so you can fix any mistake that may creep in.
2. Copy SYSTEM.INI to SYSTEM.INW in the WINDOWS directory. You can do this easily by first going to the WINDOWS directory and then typing:

   COPY SYSTEM.INI SYSTEM.INW

3. Edit SYSTEM.INW while you are in the WINDOWS directory. [Typing EDIT SYSTEM.INW if you are using MSDOS or EDITOR SYSTEM.INW for DRDOS users will allow you to start editing]

4. Once you are in the editor you will see that the second line of the SYSTEM.INW file is:

   SHELL = PROGMAN.EXE

   Change this line to:

   SHELL= C:\MSWORKS\WORKS.EXE

   Where MSWORKS is the subdirectory that contains the program Microsoft Works for Windows.

5. Save this file with the key strokes <Alt>F S and execute the NEW command in the FILE menu [if you are using MSDOS] or <Ctrl>K D and enter the new file name WORK.BAT [if you are using the DRDOS editor].

6. Enter the following batch file:

   ```
   REN SYSTEM.INI SYSTEM.ASH
   REN SYSTEM.INW SYSTEM.INI
   WIN
   REN SYSTEM.INI SYSTEM.INW
   REN SYSTEM.ASH SYSTEM.INI
   CD\
   ```

7a. For MSDOS:- Use the keystrokes <Alt>F A to save this file as WORK.BAT (or any other name you choose but make sure the extension is BAT so that the computer can recognise it as a BATCH file).

7b. For DRDOS:- Use the keystrokes <Ctrl>K X and your file will be saved.

Now you can run MSWORKS from the command line by typing WORK and pressing the <Enter> key.

If you choose to run it from a batch type menu such as AUTOMENU or John Paine's Batch Menu please remember to use the CALL syntax (i.e. CALL C:\WINDOWS\WORK.BAT ) so that you can return to the batch file from which it was called.

END OF ARTICLE

ALL THE FOLLOWING ARTICLES ARE REPRINTED FROM "YOUR COMPUTER'S TECH. TIPS" WITH THEIR PERMISSION.

## More memory for a '386

*I realise that expanded and extended memory have been exhausted as a subject, but I have a particular problem and I am not fully satisfied with the answers I have been able to elicit from others. My machine is a 25MHz '386DX running* DOS 5.0, Windows 3.1, *and a complete set of Microsoft device drivers.*

*The motherboard is, I believe, a Chips & Technologies P386, with an AMI* BIOS, *and room for up to 10Mb of on-board* RAM. *I also own an 8Mb* RAM *expansion card suitable for this machine.*

*My problem is this: the machine was originally purchased with 2Mb of* RAM. *I recently decided to increase this to help run Windows and some other memory-hungry software. The budget could stretch to a further 4Mb of memory, so I planned to put an additional 2Mb on the motherboard (giving a total of 4Mb on-board) and a further 2Mb as expanded memory on the expansion card, to use as a dedicated* RAM *drive.*

*This didn't seem too complex to me (ingenue that I am), and in the initial stages, occasioned no demur from my computer dealer either. However, after several attempts by the dealer to configure the machine this way (4 x 1Mb SIPPs on the motherboard, and several 256Kb chips on the expansion card), I was informed that I will be unable to use expanded memory on this machine until every bank on the motherboard is full.*

*My other concern is that I am told that motherboard* RAM *can only be incremented in powers of two — I cannot have 6Mb of* RAM, *only one, two, four, or eight. The manual for my motherboard seems to suggest otherwise, and clearly states that it has a maximum of 10Mb, (which is not a power of two).*

*Tony Gilbert*

A lot of people are confused about the subject of memory, which is why the subject is far from dead despite being covered at length in this and other computer journals. As a result of the PC family being the outcome of a considerable number of enhancements upon the original IBM design, there are many different ways in which PC memory configured — the ideal one for any situation depending on the hardware present, the software to be used, and the type of CPU.

Firstly, it is not necessary for memory to exist only in powers of two, but in many machines, this is the way it works out, for reasons I'll get to in a moment. I know people who use computers with 5Mb, 6Mb, and 12Mb of RAM, without any trouble.

The key requirement when upgrading the memory in any computer, is that it be wide enough for the processor in the machine. For an 8088 processor, this is eight bits, for the '286 and '386SX, it is 16 bits, and for the '386DX and all '486s, it is 32 bits. This is the bus width of the processor, and is the amount of data which can be transferred to or from the memory in a single operation. For every eight bits of data which are stored in memory, there is also a ninth parity bit (the purpose of I won't go into here), just to know that it's there.

Memory chips come in sizes of powers of two (actually, for design reasons, they increment in powers of four — every second power of two). The common sizes these days are 256K bit, 1M bit, and 4M bit, and either come as single chips one bit wide, or as SIMMs or SIPPs (SIMMs with pins) nine bits wide. So, for your '386DX, any memory expansion has to consist of 36 single chips (32, plus four parity chips), or four SIMMs or SIPPs, or a multiple thereof.

The memory already on the motherboard is not important, unless you have to remove it to make way for higher-capacity chips — it's already there, and obviously works. Since your motherboard has a total capacity of 10Mb, with 2Mb already there, I presume it has sockets for a further eight SIPPs, each of 1Mb capacity. So, to expand your motherboard memory, you could add four of these, giving a total of 6Mb, or eight, totalling 10Mb.

Your dealer is undoubtedly correct in saying that you should fully populate the motherboard before adding extra chips to the expansion board, but this should not be confused with expanded memory. Thirty-two bit memory cards for '386 motherboards are exclusively (as far as I am aware), extended memory boards, and add extra memory space to the end of that which is already on the motherboard. If the motherboard isn't fully populated, then there will be a hole in the memory space between the end of the motherboard's memory, and the start of that on the expansion card, and you won't be able to use the latter at all.

If you really need expanded (EMS) memory for some specific purpose, then on a '386 machine, the best way to achieve this is to install it as extended memory memory, and then use a memory manager such as EMM386, or QEMM, to covert some of it to expanded memory.

So in your case, put another 4Mb on the motherboard, and leave the expansion card in its box until such time as you need more than 10Mb.

# REGIONAL GROUP REPORTS

Meeting Summary For JULY

| | |
|---|---|
| Central Coast | 09/7/94 Saratoga |
| Glebe | 07/7/94 Glebe |
| Hunter Valley | 10/7 17/7/94 |
| Illawarra | 05/7/94 Keiraville |
| Liverpool | 08/7/94 Yagoona West |
| Sutherland | 15/7/94 Jannali |

**==========**

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 8162.

**==========**

HUNTER VALLEY Regional Group

The Meetings are usually held on the second or third Sunday of each month at members homes starting at 3pm. Check the location with Geoff Phillips by leaving a message on (049) 428 617. Please note that the previous phone number (049) 428 176 is now used exclusively by the ZZAP BBS which also has TI support. Geoff.

**==========**

ILLAWARRA Regional Group

Regular meetings are normally held on the second Tuesday of each month after the TIsHUG Sydney meeting 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. Contact Geoff Trott on (042) 29 6629 for more information.

**==========**

* LIVERPOOL Regional Group *

Regular meeting date is the Friday folling the TIshug Sydney meeting at 7.30 pm. Contact Larry Saunders (02) 644-7377 (home). Mum will let you know were I am or when I will be home.

NOTE: I will be doing relief managering at several stores, and will not be easy to reach during the day. Some of the stores I will be managering trade to 10pm/11pm/12pm and I am working up to 14 hours a day, 5 days a week.

*** ALL WELCOME ***

8th July 1994
My Place : 34 Colechin St. Yagoona West

12th August 1994
My Place : 34 Colechin St. Yagoona West

Bye for now Larry.
Liverpool Regional Co-Ordinator

**==========**

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 MEADOWBANK PRIMARY SCHOOL, on the corner of Thistle Street and Belmore Street, Meadowbank. Cars can enter from Gale Street and park in the school grounds. Regular items include news from the directors, the publications library, the shop, and demonstrations of monthly software.

## JULY MEETING - 2nd JULY

## AUGUST MEETING - 6th AUGUST

**********

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

July - 16th July

August - 13th August

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

**********
**********

This months list of words is based around the subject of "Winter Activities".

J T N F Z W S G F C P C B R R H P S X E
H C T D U I E C N B C S H S B E C R D K
K F K E V L G G N I L Q V A R R L P Z I
Q O U X L K Z E Z A I Z Z I O H T J O N
D Q N Q M A M C L U M K S S Y F A P B S
W A B V C W H O S I C H S Q I K E Y U D
Q V C H O H M C C O E C E L U X X X R W
U N I N R D Y E X R O E R S Y R V I Q L
K F S B I A S E G U A I L Z M W F X S Q
L Y X V P K Y M N F A U Q F R T F K V P
S R H R A Q M T P H S D T R H L I E L Y
Q X E T L U R Y C H R Y T V F J D O U V
T S E Q B Y S E S A A A Q D U K U F T I
J S C L C T B Z O L N R S M R G M H B K
K E B C O T L B P A E M P F H H K P K N
V I T C I F W S G S M K D M F O E Y E V
S L K G V O I O O E A Y K G S J X U Y Q
V S A K N S B R D H S R T U B V O F P T
W W G S K O T S Z H O T D O G G I N G R
Q G H T T D A F K P S O D X I G L U C T

Find these hidden words

HOW TO PLAY

In this puzzle there are (19) words somewhere, horizontally, vertically, diagonally even backwards. Find the words and place them below in the place provided.
The answers will appear in next months magazine.
GOOD LUCK!.

| | | |
|---|---|---|
| 1.____(4) | 2._____(5) | 3._____(9) |
| 4.C_____(6) | 5.C_____(12) | 6.D____(5) |
| 7._____(10) | 8._____(9) | 9._____(8) |
| 10.P_____(6) | 11.R_____(6) | 12.S_____(6) |
| 13._____(7) | 14._____(6) | 15._____(5) |
| 16.S_____(9) | 17.S_____(7) | 18.S_____(6) |
| 19._____(7) | | |

This puzzle was compiled using Ashley Lynn's programme "Word puzzle" which is avaliable from the TIsHUG shop.