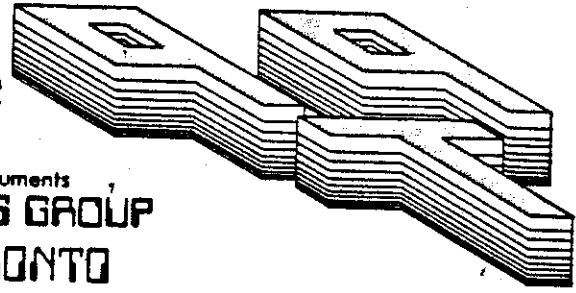


5/3/92

# Newsletter Nine-T-Nine



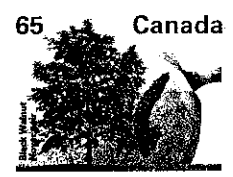
## APRIL 1992 ISSUE

Texas Instruments  
**USERS GROUP**  
TORONTO

### FOR THE TI-99/4A COMPUTER



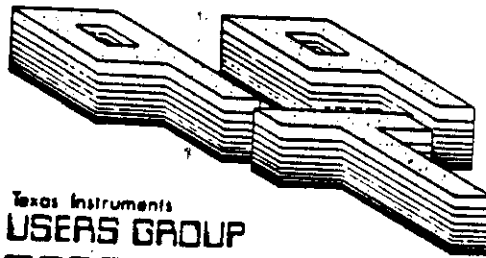
*Run? Waddaya mean run! If it ran would I be debugging it?*



**FROM:  
9T9 USERS GROUP  
15 KERSDALE AVE.  
TORONTO, ONT., M6M-1C9  
CANADA**

# NEWSLETTER 9T9

## 9T9 USERS GROUP



Texas Instruments  
**USERS GROUP  
TORONTO**

### 9T9 USERS GROUP EXECUTIVE COMMITTEE

PRESIDENT Steve Mickelson (657-1494)  
VICE-PRESIDENT Neil Allen (236-0842)  
SECRETARY/MEMBERSHIPS Randy Rossetto (469-3468)  
TREASURER/OFFICER AT LARGE Cecil Chin (671-2052)

### LIBRARY DIRECTORS

Gary Bowser (960-0925)  
Andy Parkinson (275-1427)  
Steve Findlay (416) 727-6807

### NEWSLETTER EDITOR

Steve Mickelson (657-1494)

### MEMBERSHIP FEES

FULL MEMBERSHIP ..... \$30.00 / year  
NEWSLETTER SUBSCRIPTION ..... \$20.00 / year

All memberships are household memberships. A newsletter subscription is only for those who do not wish to attend meeting, but wish to receive our newsletter and have access to our library. You are welcome to visit one of our general meetings before joining the group. If you wish more information contact either our president, in writing, at the club address on the front cover or by phone.

The meetings are usually held on the last Wednesday of each month (exceptions are December's meeting date, usually mid-month and the months of July and August, when there are no meetings. Consult this issue of Newsletter 9T9 for the date and time of the next meeting. Meetings are usually held at Neil Allen's place, 32 Graysfane Gardens, south of Bloor St., just west of Islington Ave., at 7:30 P.M. from 7:30 - 10:30 P.M.

### BBS

The 9T9 Users Group supports the Toronto BBS. The TI Tower BBS # (416) 921-2731. 300/1200/2400 BPS. 24 hrs. Sysop. Gary Bowser.

### MAILING ADDRESS:

9T9 Users Group, 15 Kersdale Ave., Toronto, Ontario, M6M 1C9, Canada

### COMMERCIAL ADVERTISING

Any business wishing to reach our membership may advertise in our newsletter.

The rates are as follows: (width by height):

FULL PAGE (7" x 10") \$30.00  
HALF PAGE (7" x 5") \$15.00  
QUARTER PAGE (7" x 2 1/2") \$7.50

Please have your ad's camera ready and paid for in advance. For more information contact the editor. Don't forget, that any member wishing to place ad's, may do so free of charge as long as they are not involved in a commercial enterprise.

### NEWSLETTER ARTICLES

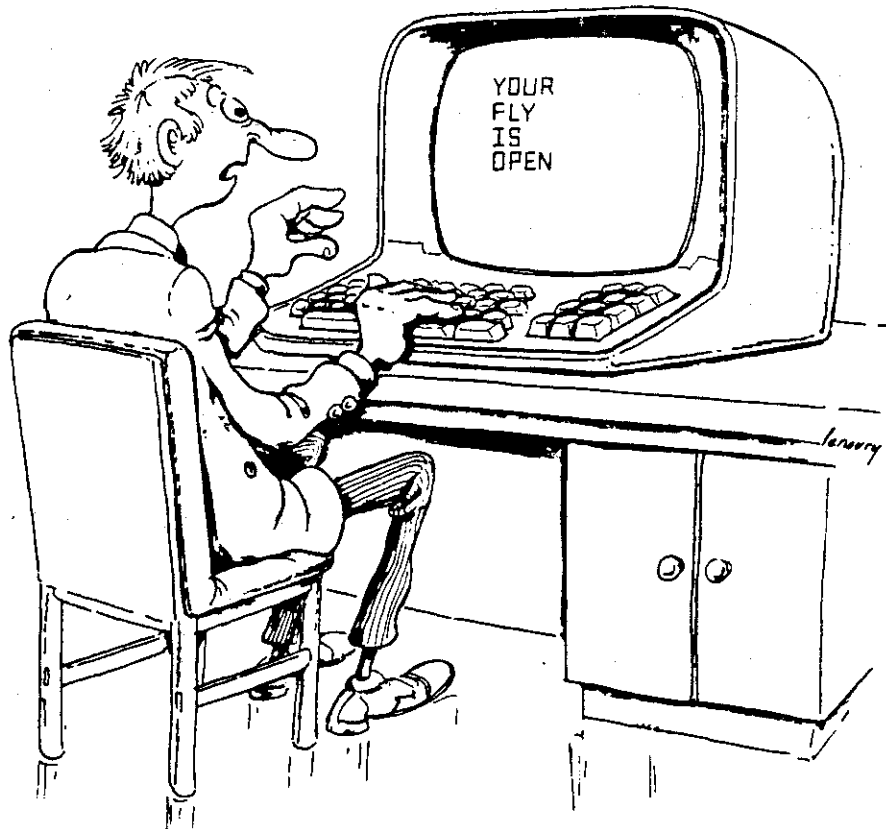
Members are encouraged to contribute to the newsletter in the form of articles, mini programs, helpful tips, hardware modifications, jokes, cartoons and questions. Any article may be submitted in any form by mail or modem. We welcome the reprinting of any article appearing in this newsletter providing credit is given to the author and 9T9. If more information is required, call the editor. The names, 9T9, Nine-T-Nine, Newsletter 9T9, 9T9 Users Group, and Nine-T-Nine Users Group are Copyright, (c), 1979-1992, by the 9T9 Users Group of Toronto, Canada, all rights reserved.

### DISCLAIMER

Opinions expressed in this newsletter are those of the writers and are not necessarily those of the 9T9 USERS' GROUP. 9T9 cannot assume liability for errors or omissions in articles, programs or advertisements. Any hardware modification or project is presented for informational purposes and the author, newsletter editor, staff and/or 9T9 Users Group cannot be held liable for any damage to the user's equipment. All such projects are done at your own risk!

This little XB program will generate colors that one can not ordinarily make with BASIC or XBASIC. This is done by mixing colors. Try the red and green combination, for instance.

```
100 : SAVE DSK1.PASTEL
110 : DISPLAY PASTEL COLORS
120 : Idea by unknown author
130 : This version by
      Earl Raguse 1/92
140 DISPLAY AT(10,1)ERASE AL
L:"This program displays col
orsnot ordinarily possible w
ithBASIC or XBASIC."
150 INPUT "BACKGROUND COLOR
NUMBER: ":F
160 INPUT "BACKGROUND COLOR
NUMBER: ":B :: PRINT
170 CALL SCREEN(16):: FOR I=
8 TO 12 :: DISPLAY AT(I,1)SI
ZE(32):: NEXT I
180 CALL CHAR(40,"55AA55AA55
AA55AA")
190 CALL COLOR(13,F,F)
200 CALL COLOR(2,F,B)
210 CALL HCHAR(2,1,128,64)
220 CALL HCHAR(4,1,40,512)
230 CALL HCHAR(18,1,128,64)
240 CALL PAK :: GOTO 140
250 SUB PAK
260 DISPLAY AT(24,1)SIZE(30)
:" Press Any Key To Proceed
::
270 CALL KEY(O,K,S):: IF S<1
THEN 260
280 SUBEND
```





## TIDBITS

#58

**-By Steve Mickelson, President 9T9 Users Group  
Compuserve 76545,1255; Delphi SMICKELSON; GENIE S.MICKELSON**

### 9T9 Recall:

This letter is to inform you, since you were/are a member and/or associate of the Toronto 9T9 Users Group, formerly know as TITO; you are subject to a recall to attend the 9T9 Users Group's Tenth,(plus 3), Year reunion, henceforth to be called 9T9 Recall! It seems that our group is firming up plans for its reunion. All Tiers will be invited to this one-time event, tentatively planned around a week before Halloween,( just after TI's "Black Friday"). Details and exact date of this reunion have yet to be announced. We hope to make the event an occasion to meet many former members,( we had several hundred, in Toronto, way back when).

We hope to combine a social mixer, hardware sale/swap meet and dealer display, all in one event. You don't have to be or have been a member to attend! I do not expect to see any serious, humorous or even boring seminars/classes or tutorial session. There should be a display of some of the latest and greatest TI software and hardware. We promise not to inflict another such event upon you, so if you miss this one, you won't get a second chance!

### Making of a Newsletter, revisited:

Along with a payment for membership, I, recently, received a request to clarify some aspects of a reprint of an article I wrote for Toronto Computes, about how to write a newsletter. The question asked about the # (pound) weights of paper. Not to get too technical, most commercial copy centers use,( unless requested otherwise) #20,(i.e. twenty pound), bond paper for photocopying.

Many copiers can run lighter weight paper, such as #14. Fourteen pound paper has advantages. As a sheet of #14 weighs about 70% of #20, you can add a few extra pages, for every ten pages of newsletter, when compared to #20, and still pay the same postage. You can see this in those air mail mailers that are a letter and envelope combined, but made of very thin paper, to keep postage fees at a minimum. You don't see too many of these mailers anymore, but I think you get the idea.

The disadvantage is thinner paper is more prone to damage in the postal system, especially if not mailed in an envelope. The thin pages of MICROpendium have succumbed to the jaws of mechanical sorting equipment in the post office. The Hoosier Users Group, had their last mailing, sent April 4, reduced to the cover page only, as it had the stamp and address, all the rest was lost, as the sorting machines effectively removed the single staple. Heavier paper stock doesn't guarantee that the newsletter won't fall apart, through the postal system,(tape or a label might do that), but it will help the sheets resist the rigors of mechanical sorting machines. Also, heavier paper stock is less likely to tear, while being read.

I hope that explains what I meant.

### New P-Box:

At the first of the year, RAVE sent out a sale flier to their customers, with a limited 10% discount on any item in that flier. I did not have enough advance notice to reprint the sale prices in the newsletter. I did take advantage of the offer, by ordering the interface card for the PE2 Box, which now comes in a kit form.

This card allows me to install my Geneve card, Hard Floppy Disk controller card, RAVE speech card, 512K memory card, plus hard drive and several 5-1/4 or 3-1/2" drives into an IBM clone case, utilizing the regulated power supply. We have had articles about modifying such a power supply and cards for the TI Peripheral Expansion Box, as well as a review of the RAVE expansion box, in this newsletter, so I won't repeat the information in them. What I can say, is that this card, is not just a remake of the slot card found in the TI expansion box, but rather the card can be configured, using jumpers, for use with the TI-99/4A and RAVE keyboard, for Geneve only or in a hybrid mode whereby the TI and Geneve co-exist, sharing some

cards, accessing others only by respective TI or 9640,(eg 32K for the TI or MEMEX for the 9640), and in some cases sharing cards,(e.g. disk controller or speech cards). This would allow co-processing by the two mother boards, though possible problems might occur, if both try to access the RS-232 or disk controller, at the same time.

Anyway, I purchased a mini-tower, with 200 watt CSA power supply, from SBL computers for less than \$100.00. I will let you know how the project goes.

Finally:

I must apologize for the recent article, from the Ozark Users newsletter, about keyboard repair. I was missing the first page, which had the first step of repairs, as well as the author's credit. The original was borrowed from the library and never returned! I felt the article, even though partly missing, was worthy of publishing. I would like to request that readers who remove articles from the binders, to photocopy articles, put the original back in the binder, intact, and in its proper place. I found articles missing and out of order in the binder. It deprives the next reader and the club of a valuable resource, through such abuses.

Well that's all for this month's Tidbits!

TACOMA Users  
USERS GROUP

## DISK AID PATCH WITH CORCOMP MANAGER

This is for those that have Disk+Aid. Or are contemplating getting this program.

The newest version uses a block sector load. As it stands now the load begins at sector 22. Now if you also have a CorComp Disk Controller Card, the manager starts at sector 30. A minor overwrite problem! So I found a way to change the loader for Aid to start loading at sector 91, directly after the manager program. Below are the steps to modify a disk for both programs.

1. Initialize a disk, installing the manager.
2. Copy a 15 sector program like FORMA2. This will fill in the 14 sector block. 14 for the program and 1 for the directory.
3. Copy "DISKOBJ" to the disk. It will start at sector 91.
4. Copy "DISK+AID-0" next.
5. Using a sector editor, "AID" should be located at 0C3-0CA. Look at sector 0C3 then at hex 6C, it should read "22" change this to 91. Write this back to the disk and now it should load.
6. IF NOT, redo the above again. I have done this twice so I know it works.

Remember one thing. IF you decide to copy this NEW disk. Do a sector copy, or follow the above steps for the programs listed above. Otherwise they will be out of order and will NOT load!

FUG PERIPHERAL

REPAIRLOG

LOG OF REPAIRED GEAR, SYMPTOMS, AND SOLUTIONS

Jan Janowski

Reprinted from the Chicago Times

1. DEVICE: SUPERCART

SYMPTOM: TOTAL LOCKUP OF SYSTEM WITH USE.

SOLUTION: REMOVED 6264 CHIP, TRIED CARTRIDGE, E/A GROM OK. DEFECTIVE 6264 CHIP. REPLACED 6264 CHIP. WORKS NOW.

2. DEVICE: BLACK CONSOLE

SYMPTOM: WILL NOT WORK, BLACK SCREEN, TWO TONES ON ALL THE TIME.

SOLUTION: FOUND -5V SUPPLY BAD. REPAIRED POWER SUPPLY. WORKS NOW.

3. DEVICE: GEMINI 10-X PRINTER

SYMPTOM: WILL DO SELF TEST, BUT WILL PRINT ONLY NUMBERS CORRECTLY WHEN CONNECTED TO COMPUTER.

SOLUTION: REPLACED PARALLEL PORT INPUT DATA BUFFER IN PRINTER. WORKS NOW.

4. DEVICE: RS232 CARD (TI)

SYMPTOM: TOTAL LOCKUP OF SYSTEM. LED WILL NOT GO OFF. THIS CARD WAS PLUGGED IN "HOT."

SOLUTION: REPLACED 6 CHIPS, INCLUDING A 9902, AND 2 RESISTORS. WORKS. THIS CARD TOOK APPROXIMATELY 20 HOURS TO REPAIR. (SPREAD OVER 6 MONTHS) PLUS APPROX. \$18.50 IN PARTS.

5. DEVICE: EXPANSION BOX

SYMPTOM: NO POWER TO CARDS. FUSE GOOD ON CHASSIS REAR.

SOLUTION: FOUND BLOWN FUSE INSIDE TRANSFORMER. REPLACED FUSE. WORKS NOW.

6. DEVICE: MYARC RS232 CARD

SYMPTOM: WILL NOT WORK. SENT TO MYARC 3 TIMES. STILL DOES NOT WORK.

SOLUTION: IT WAS A FEW THINGS: BAD SOLDERING BY MYARC. THEY JUMPERED OUT BAD PARTS OF CHIPS AND USED UNUSED PARTS OF BAD CHIPS INSTEAD OF REPLACING THE CHIPS THAT WERE BAD. 9902 CHIP BAD HERE, TOO. WORKS NOW.

7. DEVICE: RAMDISK

SYMPTOM: WILL NOT HOLD ROS.

SOLUTION: FAILED MEMORY TEST. DEFECTIVE 6264. WORKS NOW.

8. DEVICE: TI DISK CONTROLLER CARD

SYMPTOM: WILL NOT WORK WITH ANY DRIVE.

SOLUTION: ONE CHIP OBVIOUSLY OVERHEATED, REPLACED. WORKS NOW.

9. DEVICE: EXPANSION BOX

SYMPTOM: LOCKS UP CONSOLE, WILL NOT ADDRESS ANY CARD INSIDE IT.

SOLUTION: FOUND THE INTERFACE CARD TOTALLY SHORTED OUT. REPLACED EVERY BUFFER CHIP. WORKS NOW.

10. DEVICE: MYARC 512K RAMDISK

SYMPTOM: WILL NOT WORK. (SUSPECTED TO HAVE BEEN PLUGGED IN HOT.)

SOLUTION: SINCE NO SCHEMATICS AVAILABLE, USED OHMMETER TO CHECK FOR RESISTANCE DIFFERENCES ON GATES. REPLACED 7 CHIPS, PROBABLY 2 OR 3 UN-NEEDEDLY (DUE TO LACK OF SCHEMATICS). WORKS NOW.

11. DEVICE: EXP-BOX INTERFACE CARD

SYMPTOM: WILL NOT WORK IN ANY EXP-BOX.

SOLUTION: FOUND BUFFER CHIP BAD. REPLACED. WORKS NOW.

12. DEVICE: BLACK TI CONSOLE

SYMPTOM: BLACK SCREEN, LOCKED UP, TWO TONES ON ALL THE TIME. (EXP BOX CABLE PLUGGED IN UPSIDE DOWN.)

SOLUTION: POWER SUPPLY OK, AFTER WORKING ON IT FOR HOURS, GAVE UP. OWNER DONATED IT FOR PARTS.

13. DEVICE: RAMDISK

SYMPTOM: WILL NOT HOLD ROS. SOMETIMES CORRUPTS DURING CONFIGURATION.

SOLUTION: OCCASIONALLY WILL FAIL MEMORY TEST, BUT NOT CONSISTENTLY, OR IN SAME LOCATIONS. FOUND LED ON PIN 26 OF 6264 CHIP TO BE BAD. REPLACED WITH GENERAL PURPOSE DIODE. WORKS NOW.

14. DEVICE: CORCOMP RS232 CARD

SYMPTOM: WILL NOT WORK (HIT BY LIGHTNING).

SOLUTION: DAMAGED BEYOND REPAIR. STRIPPED FOR PARTS.

15. DEVICE: MODEM

SYMPTOM: WILL NOT WORK (HIT BY LIGHTNING).

SOLUTION: DAMAGED BEYOND REPAIR. STRIPPED FOR PARTS.

16. DEVICE: PRINTER

SYMPTOM: SELF TEST WORKS, BUT NOT WHEN CONNECTED TO COMPUTER. (HIT BY LIGHTNING.)

SOLUTION: THIS PRINTER DOES NOT HAVE BUFFER CHIPS. TRACED TO CUSTOM CHIP MADE BY PRINTER MANUFACTURER. UNABLE TO PURCHASE PART. SUGGESTED MANUFACTURER REPAIR.

17. DEVICE: TI RS232 CARD

SYMPTOM: SERIAL PORT WORKS, BUT NOTHING OUT OF PARALLEL PORT.

SOLUTION: CHANGED U12 ON CARD. WORKS NOW.

18. DEVICE: EXPANSION BOX

SYMPTOM: NO POWER TO BUSS. REAR FUSE OK.

SOLUTION: FOUND TRANSFORMER FUSE BLOWN, AND DIODE ON POWER SUPPLY SHORTED. DIODE IS 3A/100V. REPLACED BOTH. WORKS NOW.

19. DEVICE: MYARC EXPANSION BOX

SYMPTOM: FUSE BLEW. OWNER REPLACED FUSE. FUSE BLEW EACH TIME. ON THIRD FUSE BLOW, SMOKE WAS OBSERVED BY OWNER COMING FROM UNIT.

SOLUTION: MY SUGGESTION WAS TO SEND IT TO MYARC. DUE TO EXTENT OF DAMAGE.

20. DEVICE: EXTERNAL DRIVE CHASSIS WITH 2 DRIVES

SYMPTOM: BOTH DRIVES ERROR OUT CONTINUOUSLY.

SOLUTION: POWER SUPPLY OK. DRIVES CHECKED IN OTHER SYSTEM OK. OTHER DRIVES FAIL IN CHASSIS. TRACED TO CABLE BETWEEN DISK CONTROLLER CARD AND CHASSIS. OWNER REPLACED CABLE. WORKS FINE NOW.

21. DEVICE: TI DISK CONTROLLER CARD

SYMPTOM: ON POWER UP, LED FLASHES, BUT WILL NOT ACCESS ANY DRIVE. HIT BY LIGHTNING.

SOLUTION: FOUND +12V REGULATOR BLOWN. REPLACED. WORKS NOW.

22. DEVICE: MYARC 512K CARD

SYMPTOM: NO OPERATION. (CARD DESTROYED WHEN RAMDISK INSTALLED IN EXP-BOX BACKWARDS.)

SOLUTION: AFTER PIGGY-BACKING ALL LOGIC CHIPS, AND BURNING NEW PROM, STILL NO OPERATION. DISCOVERED THAT RAS AND CAS WERE NOT BEING GENERATED, AS WELL AS NO REMOTE DATA BUSS ENABLE. SUGGESTED THAT THIS CARD BE RETURNED TO MYARC.

23. DEVICE: CORCOMP DSSD DISK CONTROLLER

SYMPTOM: OPERATES, BUT WITH EXCESSIVE SEARCHING ON DRIVES. THIS UNIT WAS IN SAME EXP-BOX THAT HAD RAMDISK PLUGGED IN BACKWARDS.

SOLUTION: ON EXAMINATION OF INSIDE, NOTICED THAT THERE WERE ABSOLUTELY NO BUFFER CHIPS WHATSOEVER. ANY DAMAGE ON THE BUSS WILL GO DIRECTLY TO THE INSIDE OF THE CARD. THE WESTERN DIGITAL CHIPS, PROMS, AND PALS WERE HANGING DIRECTLY ON THE BUSS, WITHOUT ANY PROTECTION TO THE CHIPS. AFTER SEEING THIS, I REFUSE TO WORK ON ANY OTHER CORCOMP PRODUCT.

24. DEVICE: TI RS232 CARD. SAME IDENTICAL CARD AS IN NUMBER 4.

SYMPTOM: ON POWER-UP IN EXP-BOX WITH RAMDISK. IT HANGS UP, IN THAT ITS LED COMES ON BEFORE THE LED OF THE DISK CONTROLLER CARD. REMOTE INTERRUPT LINE (EDGE CONNECTOR PIN 17) GOING LOW.

SOLUTION: TRACED EVENTUALLY TO WEAKENED 9902 CHIP. REPLACED. WORKS NOW.

25. DEVICE: MYARC RS232 CARD

SYMPTOM: SENT TO MYARC MORE THAN ONCE. CARD PRINTS OUT GARBAGE OCCASIONALLY. USER PURCHASED NEW RS232 CARD, GAVE UP ON THIS ONE.

SOLUTION: TRACED TO NUMEROUS COLD SOLDER PLATED THROUGH HOLES. RE-SOLDERED. AND IT WORKS FINE NOW.

26. DEVICE: HARDWARE FROM USER GROUP, CONSISTING OF: EXP-BOX, 32K, TI DISK CONTROLLER, AND 4 DRIVES (2 DSSD, 2 SSSD). OTHER THAN ON/OFF SWITCH, EXP BOX, DISK CONTROLLER, 32K FINE. OF THE DISK DRIVES:

SYMPTOM: DRIVE #1, LABELED "NO GOOD." FOUND TERMINATING RESISTOR MISSING, AND JUMPERS IN WRONG POSITION. WITH THAT FIXED, TOOK DRIVE APART, AND CLEANED HEADS. THEN CHECKED IT FOR READING. WHEN THAT WORKED, CHECKED FOR FORMATTING, AND IT FAILED.

SOLUTION: TESTED THE DRIVE IN FORMAT WHILE SETTING DISK SPEED. WHEN SPEED WAS SET, ALL WAS FINE. CHECKED FOR READ AND WRITE FOR SSSD. OK NOW.

27. DEVICE: DISK DRIVE #2 FROM ABOVE (DSSD).

SYMPTOM: LABELED "READS, WON'T WRITE."

SOLUTION: FOUND JUMPERS IN WRONG POSITION, ONE HEAD DISCONNECTED FROM BOARD, AND DISK SPEED OFF. AFTER CLEANING, ADJUSTING, THROUGH TESTING FOR DSSD, ALL IS FINE.

28. DEVICE: DISK DRIVE #3 FROM ABOVE (SSSD)

SYMPTOM: LABELED "WRITES, WON'T READ."

SOLUTION: FOUND HEAD COVERED WITH OXIDE FROM A DISK, NOW LONG GONE. DISASSEMBLED DRIVE, CLEANED, AND CHECKED THOROUGHLY FOR SSSD (NO SPEED CHANGE NEEDED). WORKS FINE NOW.

29. DEVICE: DISK DRIVE #4 (DSSD) FROM ABOVE

SYMPTOM: LABELED "NO GOOD."

SOLUTION: CLEANED CONTACTS, HEADS, AND PLACED JUMPERS IN RIGHT POSITION, AND FOUND TERMINATING RESISTOR IN IMPROPERLY (NOT ALL LEGS IN SOCKET). ADJUSTED SPEED OF DISK DRIVE, AND CHECKED THOROUGHLY FOR DSSD. WORKS FINE NOW.

30. DEVICE: MYARC DSDD DISK CONTROLLER

SYMPTOM: NO LED FLASH ON POWER UP (NO DSR). NO OTHER OPERATION WHATSOEVER. DOES NOT LOCK UP CONSOLE. HAD BEEN INSIDE EXP-BOX WHEN A CARD WAS DROPPED INSIDE THE EXP-BOX WHEN ON.

SOLUTION: FOUND 7805 REGULATOR RUNNING EXTREMELY HOT IN ORDER TO SUPPLY +5V. BOTH 74LS244 AND 74LS245 CHIPS ALSO WARM. REPLACED LOGIC CHIPS, AND REGULATOR MUCH COOLER. STILL NO LED FLASH ON POWER UP, THOUGH. PULLED STATIC MEMORY CHIP AND WESTERN DIGITAL CHIP AND CHECKED THESE IN OTHER MYARC CARD, AND THEY ARE OK. REPLACED PAL, AND EVERYTHING WORKED AS USUAL.

31. DEVICE: TI RS232 CARD

SYMPTOM: LOCKS UP EXP-BOX WHEN ON. LED WILL NOT GO OFF, STAYS LIT CONTINUALLY. THIS CARD WAS IN SAME EXP-BOX AS #30 WAS, AND WAS DAMAGED AT THE SAME TIME.

SOLUTION: FOUND 7805 REGULATOR RUNNING EXTREMELY HOT IN ORDER TO SUPPLY +5V. ALSO FOUND U5. 74LS244 CHIP RUNNING EXTREMELY HOT.

32. DEVICE: EXPANSION BOX

SYMPTOM: DISK DRIVE WILL NOT SPIN. OTHER THAN THAT, EVERYTHING ELSE WORKS.

SOLUTION: CHECKED +12V ON DRIVE POWER PLUG. READS 3V. 12V REGULATOR ON POWER SUPPLY OF EXPANSION BOX FAILED. REPLACED. FINE.

33. DEVICE: EXPANSION BOX

SYMPTOM: DISK DRIVE WILL NOT WORK.

SOLUTION: FOUND POWER SUPPLY FINE. EXAMINED DRIVE ACTION WHILE OUT OF BOX. FOUND THAT STEPPER MOTOR NOT MOVING. MOVED IT WITH PENCIL, AND FOUND IT BINDING AND STUCK. PUT A SMALL AMOUNT OF FINE OIL ON BEARINGS ON END OF MOTOR. EXERCISED STEPPER MOTOR WITH POWER OFF TO WORK OIL INTO BEARINGS. FINE NOW.

34. DEVICE: RAMDISK

SYMPTOM: ANY TIME IT IS INSTALLED IN EXP-BOX, THE SYSTEM WILL NOT BOOT UP. SCREEN STAYS BLACK.

SOLUTION: FIRST, PULLED 8K ROS CHIP, AND PLACED IT ON A PIECE OF METAL, TO SHORT OUT ANY VOLTAGE IN THE CHIP (TO ERASE THE MEMORY). REINSTALLED, AND TRIED IT AGAIN. STILL BAD. LEFT IT RUN FOR A COUPLE OF MINUTES, AND THEN TURNED IT OFF, AND FELT THE CHIPS. THE DATA TRANSCIEVER, 74LS245, WAS WARMER THAN USUAL. SWAPPED IT. IT BOOTED FINE. LOADED ROS. FINE NOW.



*I think it needs the ol' shock treatment!*

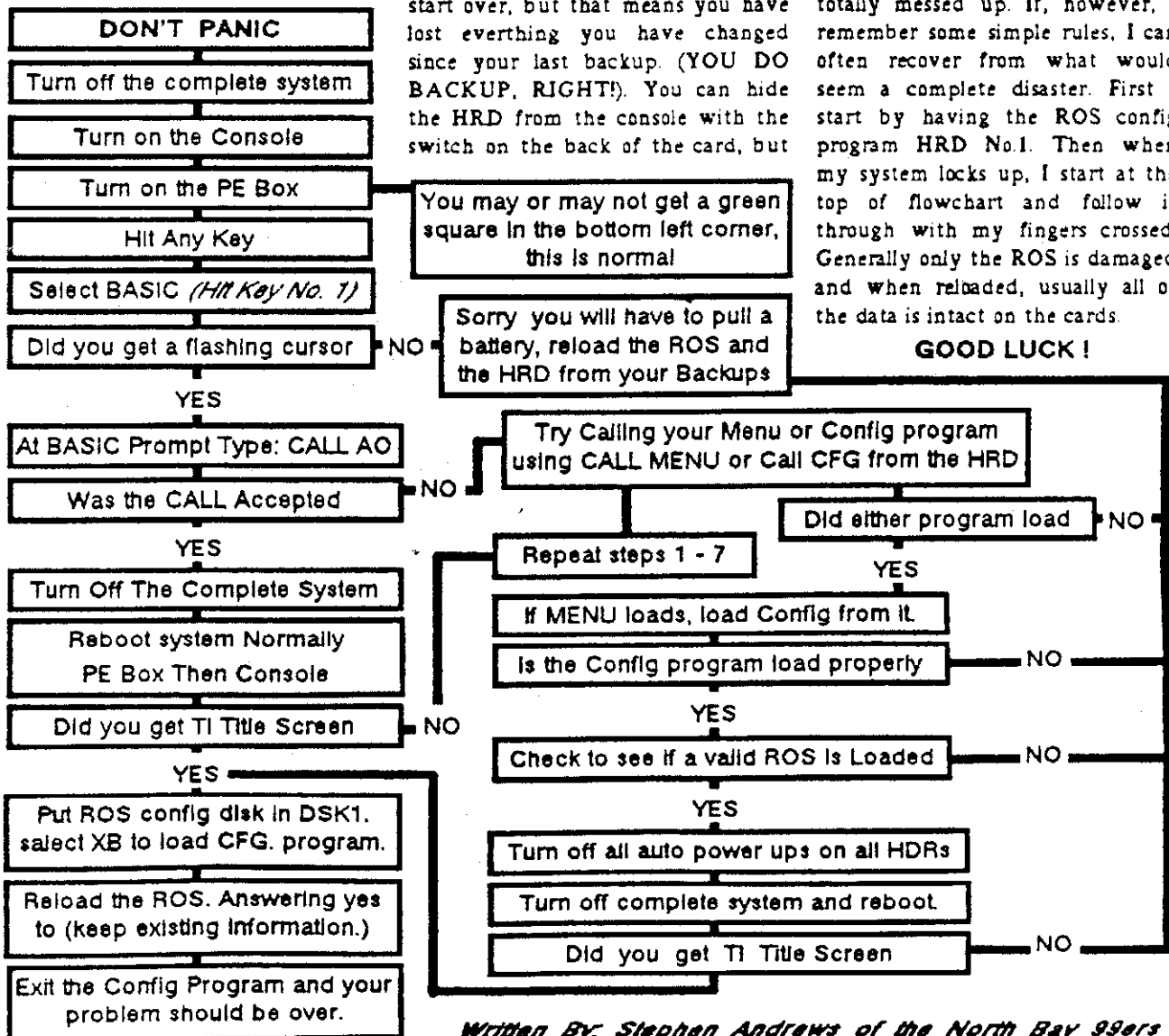
# Horizon RAM Disk Lockups

*( Don't dump that data yet! )*

Certain coincidences this morning have made me decide to write an article, or sort of a quick reference guide for users of Horizon Ram Disks (HRD) of all types. This morning started with a call from our newsletter editor asking questions about a new HRD test program from OPA and problems he was having with one of his HRDs. We discussed his problem, determined that it was most likely the hardware and decided on a course of action. We said our goodbyes and

I went to my system to work on something only to find that my own system would not boot. This is where the real story begins. Anyone with a HRD, with any of the new Ram Operating Systems (ROS) installed in memory, as opposed to some of the Eproms that are available has had this problem. ( Come on be honest ) At this point you have a lot of options. You can shut everything off and walk away, but that will not solve anything. You can pull the batteries on your HRD(s) and start over, but that means you have lost everything you have changed since your last backup. (YOU DO BACKUP, RIGHT!). You can hide the HRD from the console with the switch on the back of the card, but

we are not all lucky enough to have that option because the old cards do not have the switch. You could take out the card, put it in the closet and you will never have the problem again. (No, No, that just wouldn't do) These are all options, but not very good ones so let's discuss the alternatives. I am the club librarian and when trying out many new programs, or reviewing older ones, I often have conflicts with the ROS on my cards that make the system appear to be totally messed up. If, however, I remember some simple rules, I can often recover from what would seem a complete disaster. First I start by having the ROS config program HRD No.1. Then when my system locks up, I start at the top of flowchart and follow it through with my fingers crossed. Generally only the ROS is damaged and when reloaded, usually all of the data is intact on the cards.



*Written By: Stephen Andrews of the North Bay 99ers*





This is a re-hash of the tutorial information I did around April, May and June of 1990, which I added to and updated. If you read my tutorials word for word, some of the information will be familiar. It is a very, very useful part of TIB.

## The INSTALL Memory Area of TI-Base Macros

filename \MC  
MODIFY COMMAND

filename \DS  
DISPLAY STRUCTURE

filename \DST  
DISPLAY STATUS

filename \IC  
INSTALL CATALOG

filename \RES  
SET DATDISK=DSK6.  
SET PRGDISK=DSK5.  
SET PRINTER=PIO.CR.LF  
SET PAGE=000  
SET HEADING ON  
SET TALK ON  
SET SPACES=01  
SET RECNUM ON  
SET LSPACE=256  
CLEAR LOCAL  
SET CURSOR=02  
SET CRLF ON  
CLEAR  
DISPLAY STATUS  
INSTALL CATALOG

filename \DSPA  
PRINT (Drft),(E)  
DISPLAY STRUCTURE  
SNAP  
PRINT (Drft),(f)  
PRINT ALL

filename \D1  
SET DATDISK=DSK1.

filename \D6  
SET DATDISK=DSK6.

filename \D7  
SET DATDISK=DSK7.

filename \D8  
SET DATDISK=DSK8.

filename \D9  
SET DATDISK=DSK9.

Macro Instructions have got to be one of the big new features in TI-Base. A Macro, or Macro Instruction, is roughly the ability to execute a large command, or a large group of commands, with a single keystroke or a very short key input. TI-Base Version 3.0 or later has that capability. It's a little repetitive to set up a large number of Macros, but once you've done it the rewards are great. Dennis has set up a usable area in VDP RAM, which is handled by the phrase INSTALL, for TIBs use. You should think of the word INSTALL more as the name of the area and not as a command. The things which you can do to the INSTALL area are CLEAR, ADD, REMOVE, CATALOG, LOAD and SAVE. You must create a command file on disk for each Macro Phrase you want to use. For example, I entered MODIFY COMMAND DSK1.\MC. This created the CF named "\MC" on disk drive #1. When the Edit screen appeared I entered two words "MODIFY COMMAND" and I pressed <FCTN 8> to save the CF. I did not enter any comments or place RETURN at the end of the CF. Then, at the dot prompt I entered INSTALL ADD DSK1.\MC. TIB retrieved the CF named "\MC" from DSK1 and placed its contents (MODIFY COMMAND) in the INSTALL area under the name "\MC". This allows me to execute that command by simply typing \MC at the dot prompt. This may not seem like much at first, but here's the big picture. TIB can execute many individual commands from VDP by their names and a Macro can be as large as a Command File. I created each of the Command Files you see on this page under their individual filenames and used the ADD directive to place them all in VDP at the same time. After that I entered INSTALL SAVE DSK6.INST2. TIB SAVED the complete INSTALL group to DSK6.INST2, with the suffix "/1". Next, I added the line "INSTALL LOAD DSK6.INST2. to my SETUP CF. This tells TIB to automatically LOAD all the commands when TIB is powered up. I haven't tried it yet, but I think that you should be able to stack up your ADD commands in a CF to make it easier to modify the overall INSTALL package. The number and size of Macros placed in VDP are only limited by space, which is currently 2546 Bytes. With everything you see to the left loaded into INSTALL I still have 1879 Bytes left. "Not Bad!" This Macro package means a lot to non-ramdisk users, because the execution is very fast compared to disk access. You could load several large CFs, which you use often, into INSTALL and execute them when needed. I wanted to demonstrate this idea, so I loaded the complete CF named 1LBL91 from Tutorial 24.I.2 (Sept. 14, 1991) into INSTALL. I already had the CF on DSK7 of my RAM Disk. I merely typed INSTALL ADD DSK7.1LBL91 at the Dot prompt (Dp) and pressed ENTER (<E>). This would be a good test because 1LBL91 contained a wide variety of TIB commands, including RETURNS and COMMENT lines. After TIB ADDED 1LBL91 to the INSTALL area I typed \IC (<E>). This runs the INSTALL CATALOG Macro you see to the upper left. This told me that the 1LBL91 CF used 1471 Bytes of INSTALL memory space and that I still had 405 Bytes remaining to use. "That's great!" At that moment I had placed twelve Macros in INSTALL, the eleven on the left of this page and 1LBL91. INSTALL contained the twelve Macro names and one hundred and three lines of commands and comments, and I still had 405 Bytes left. Next I typed 1LBL91 at the Dp and (<E>). 1LBL91 ran just fine. It opened the Database (Db), set my printer, asked me for the record number, found the name I wanted, printed some labels (using my special printer control commands), reset my printer and TIB commands and RETURNed me to the Dp. "And I think it's a little faster than my RAM Disk. I love it." If you manage this space well, the speed advantages over regular disk drives will be enormous.

Next Page.

11BL91 is a great example of a large CF which can be stored and run from the INSTALL area, but I normally run small CFs as Macros. I always seem to be using the wrong disk for my DATDISK when an idea strikes me for something to do. This led me to the creation of the last five Macros on 25.1.1 (\D1...D9). "DSK6, 7, 8 and 9 relate to my Bud Mills Horizon RAM Disk." If I want to access a bunch of stuff on drive #7, I type \D7 (E) at the Dp, and TIB processes the command SET DATDISK=DSK7. The key stroke savings are not much for one Macro use, but if you do this ten times during one computer session it means a lot. A Macro that means even more to me is \DSPA. I make a lot of changes to several small Dbs on a frequent schedule. Whenever I do, I like a printout I can use to check my work while I'm away from the computer. I might type \D7 (E), USE CLUB91 (E) and then \DSPA (E). These three short Macros would switch my DATDISK to DSK7, USE CLUB91 located on drive #7 and \DSPA would set my printer to Emphasized mode, DISPLAY the Dbs STRUCTURE to my screen and then SNAP the screen to my printer, next it changes my printer to Condensed and prints the whole Db.

SETUP/Command File

```

SET TALK OFF
WRITE 22,4,"Welcome to TI-Base Ver.:
3.01"
*
*                               SETUP/C
*                               Ver. 3.01 04/14/90
*
INSTALL LOAD DSK5.INST2
*
COLOR WHITE DARK-BLUE
PRINTER EPSON
DO \RES
*
*                               Version 3.01
*
* Type QUIT to terminate TI-Base
*
* \MC      = Modify Command
* \DS      = Display Structure
* \DST     = Display STatus
* \IC      = Install Catalog
* \RES     = RESet TIB Options
* \DSPA    = \DS, SNAP, PRINT ALL
* \EDIT    = EDIT
* \MS      = Modify Structure
* \D1      = SET DATDISK=DSK1.
* \D6      = SET DATDISK=DSK6.
* \D7      = SET DATDISK=DSK7.
* \D8      = SET DATDISK=DSK8.
* \D9      = SET DATDISK=DSK9.
*
RETURN <FCTN 7> help not available
  
```

After \DSPA is finished I type CLOSE (E) and go on to the next job. I have included another listing of my SETUP CF in the lower left corner of this page because I want to run through parts of it again. SETUP is the CF that automatically runs right after you type in the current date when TIB is loading. The first important line is INSTALL LOAD DSK5.INST2. This line takes the one file that holds all 11 Macros from 25.1.1 and loads them into INSTALL. As soon as that is done the 11 Macros are ready to use. Next I set the screen colors I like. The third thing I do is load up the printer commands from my personalized Db as I tried to demo in TUT 24.1.1. The last command I issue from SETUP is DO \RES, which runs the RESET Macro that should now be in the INSTALL area. You should take special notice that you must include the [DO] when running a Macro from a CF. If you run a Macro from the Dp the [DO] is not used. The last 20 lines of SETUP are all comments. By my positioning, these lines will remain on the screen after SETUP is finished. This allows me to refresh my memory as to the Macros which I have loaded into INSTALL and to possibly print out a copy of this screen, using SNAP, if need be. This whole job was a lot of work the first time, but now that it's done, and runs itself each time I start TIB, it's a great little tool. I need to throw in another important note. INSTALL works like a stack with the last item you put in being the top of the stack. This means that you cannot remove and/or replace an item in the middle of the stack without extracting and replacing all the items above it in the stack. For example, if I want to remove \D1 from INSTALL, I must enter INSTALL REMOVE \D9 (E), INSTALL REMOVE \D8 (E), INSTALL REMOVE \D7 (E), INSTALL REMOVE \D6 (E) and finally INSTALL REMOVE \D1 (E) to accomplish my goal. You would then need to replace, using ADD, any of the Macros you did not want removed with \D1. For this reason you need to place the most temporary Macros at the end, or closest to the top, of the stack, as I did with 11BL91. If you need to REMOVE something that is more than half way into the stack, you should consider using INSTALL CLEAR to CLEAR everything out of the INSTALL area and then put back what is needed. There is a little more information on ADDing CFs to the INSTALL area, using those CFs and then REMOVE(ing) them from the INSTALL area, from another CF in the April, May and June 1990 Newsletters, if you're interested. This is something that you must be determined about, plus you must start small and expand the number and size of your Macros slowly. As a matter of fact you could say that about TI-Base in general.

**A special note from Marty**

*I do not anticipate doing any more TI-Base Tutorials. If I find the time I may try and write something, but I do not expect that right now.*

Good luck  
 Marty.



**"Life and Times"**  
support your local programmer  
Copyright © 1992  
by Richard P. Phillips  
and the SLaVEs & OTTUG UGs

We TIers probably have a higher percentage of hackers in our ranks than can be claimed by the users of any other computer platform<sup>1</sup>. But if this is true, why aren't we seeing more new software being written for our machine?

Barry Traver posed this question in the Feb. '92 issue of *Computer Monthly*<sup>2</sup>, and his query started me thinking.

When I set out to learn programming, I didn't know anything more about computers than I'd seen in *2001: A Space Odyssey*, and *Star Trek*. At that time I thought there was little that computers couldn't do. At the end of my first day with a 99/4A, I came upon a revelation: *computers are as dumb as stumps!* Not simply TI's home computer, but *all* computers. They do *exactly* what you tell them to do. Nothing more - nothing less. This might sound like I'm pointing out the obvious, but for me it was an important truth.

The fact that computers must be told *everything* isn't a problem if you're

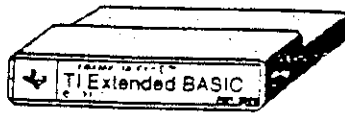
<sup>1</sup> when I say "hacker," I mean someone who "hacks" together programming code. Not someone who breaks into a computer owned by his or her local bank.

<sup>2</sup> Barry writes a monthly column for CM entitled, "TI-99ers In The '90s."

writing a simple 'Can You Unscramble The Secret Word' game. But programmers know that we users in the 90's expect more. I wonder if some of them feel that their level of competency isn't up to writing for today's Tier.

I'm not suggesting that we users should lower our standards, but I think it's important to remind software developers that most of us apply different standards to dissimilar software projects. I don't expect the same things out of a low cost shareware *Nash Cards* program for children, as I do out of an expensive *Desktop Publishing Package*.

As was the case for many TI programmers, my first language was TI Basic. I soon followed this up with Extended Basic.



By almost any yardstick TI's Extended Basic is as powerful, as it is slow. It is, however, a good language for writing many of the projects that pop into my head. Additionally, several people have put together packages of Extended Basic callable routines which can perform some tasks much faster than plain vanilla Extended Basic. Some of these routines will even allow you to perform functions that Extended Basic can't manage.

These packages run the gambit from freeware, through shareware, MICROpendium recently ran a series of articles dealing with this very subject, and I feel they're worth looking at.

I've emphasized Extended Basic only because it is a language that many of us understand, and it's accessible to almost everyone. But if your programming canvas of choice is another language, by all means do what is right for you.

If you have the talent, the time, and the desire - write! If you have everything but the talent, give it a shot anyway.

I want to leave you with these simple thoughts<sup>3</sup>. When we buy software from a vendor, or on those all too rare occasions send in our shareware fee, in essence we are thanking the developer for his or her efforts. Few freeware programmers ever hear a word about their creation - good or bad. If you like something someone has created, drop the 29¢ and send a letter saying so. If you have suggestions, let them know. Most programmers love to hear from their audience. And to you freeware programmers, give yourselves a pat on the back for a job well done. It might be the only one you get...

- Rick

<sup>3</sup> as I am capable of no other...

**SUB PROGRAM FALSEHOODS**

by Gene Hitz from the Milwaukee Newsletter, Sep'89  
via Tidbits, Feb'92 (Mid-South 99 Users Group)

*ed... Remember Gordon Leonard's demo of CUT-THROAT CRIBBAGE a year or so ago? When Gene Hitz was writing that fine program he discovered that portions of subprograms could be used as subroutines without calling the program (I didn't know you could do that). If that's old hat to you, read the article anyway. It may cause you to remember how good it felt when you first made your TI outperform the "bible". It is an old article, but I thought it deserved one more repeat ...*

Have you ever tried using the sub-program capabilities of our computer? Up to now, although I've done quiet a bit of extended basic programming, I've never had the opportunity to try it. For one thing I've never really had a need for it. Sub-routines are faster and always have been sufficient. However, I've just finished writing a cribbage game called CUT-THROAT CRIBBAGE since I haven't been able to find one written for our TI. Although I pride myself on my ability to write tight code, with all the routines in it, it began to grow into a monster. Although I didn't attempt to teach anyone how to play cribbage, still I felt it useful to include a routine that actually showed how the points were counted. This was not only for beginners who were a little unfamiliar with point counting, but also if anyone disagreed with the computer's count and just couldn't accept what they couldn't see, here would be a means of demonstrating how each point was gotten. Since this routine would carry the program from the realm of Program Format to that of Int/Var 254, I wanted a means by which a user could simply remove the routine if he so desired. Thus I decided to write the routine as a subprogram.

Our trusty little TI computer manual tells us that the only way to enter a subprogram is through the CALL statement and to exit it with a SUBEXIT or SUBEND. And the variables would not be carried over except through a parameter list in the CALL statement. Well I had some short routines that I used throughout the main program that I felt I wanted to use in the subprogram and didn't feel like rewriting them so I just jumped to them and returned. Seeing that this worked fine, I then tried jumping to a subroutine in the subprogram from the main program and again returned with no problem. I found I could even carry the variables back and forth without a

hitch. Seems that when you GOSUB into a subprogram or out of one and return without encountering the subprogram start or the SUBEND or SUBEXIT, the computer just doesn't realize that you've crossed the boundaries and assumes it's just a subroutine.

In the final version though, I did remove the illegal jumps into the subprogram in order to allow the user to delete the subprogram without any ERROR results. Remember just because the manual says something can't be done doesn't mean it can't be done. If you want to do something different don't be afraid to try it.'

---

**!!! WARNING !!!**

*ed ... another oldie-- This one by Ed McNish from the Ninty Niners of Vancouver Area, July, 1990*

This machine is subject to breakdowns during periods of critical need.

A special circuit in this machine called a "critical detector" senses the operator's emotional state in terms of how desperate he or she is to use the machine. The "critical detector" then creates a malfunction proportional to the desperation of the operator. Threatening the machine with violence only aggravates the situation. Likewise, attempts to use another machine may cause it to also malfunction. They belong to the same union. Keep cool and say nice things to the machine. Nothing else seems to work.

Never let anything mechanical know you are in a hurry.

---

**TEST YOUR AWARENESS**

First read the sentence enclosed in the box below

**FINISHED FILES ARE THE RE--  
SULT OF YEARS OF SCIENTIF--  
IC STUDY COMBINED WITH THE  
EXPERIENCE OF MANY YEARS**

Now count the F's in the sentence. Count them only once and do not go back and count them again.



## ANALOG AND THE TI COMPUTER

When we think of uses for a computer, our thoughts normally turn to word processing, spreadsheets, and data bases. Another equally important use is to monitor and control our surroundings, such as temperature. When trying to use a computer for this purpose, the problem that immediately arises is bringing this type of signal, which is ANALOG, into a computer which is DIGITAL. The method commonly used is to change the analog signal to a digital one, using an analog to digital conversion chip (ADC). Many home computers have a chip like this built into the joystick or game port for use with paddles; unfortunately, the TI does not!

My project got started when I discovered an 8 bit ADC chip made by TI; coincidentally that is both inexpensive (approx.\$3.00), and requires few support parts. The TCC 5488 or TLC 549 Chip is unique because it requires only two signals to control it and has serial output. These features make it a natural to be used with the TI joystick port!

The joystick port consists of two outputs, Joy A & B and five inputs: Up, Down, Left, Right, and Fire. When the computer executes a CALL JOYST command, it energizes one of the outputs and examines the directional inputs to see if a switch in the joystick is closed, connecting that output and an input. Using these outputs to control the ADC is complicated by the fact that the Joy A&&B outputs are with respect to the computer console power supply. None of these power supply signals are available in the joystick port, although they appear in various combinations on other connectors, such as video, sound, cassette, game, and system bus ports.

The problem is solved when it is realized that only one output is energized at a time, and the differential between these outputs can be used to generate the two control signals required for the ADC chip. Getting the computer to accept data from the chip is done by lighting an LED in an optocoupler, which causes a photo transistor to turn on and act like a switch in a joystick.

The circuit (Fig.1) works as follows: With no joystick commands and hence no output signals on the Joy A & B pins - both transistors 0a and 0b are off causing both the I/O

clock and chip select (CS) pins to be held low enabling the chip. When the CALL KEY statement in step 105 is executed, Joy A goes low and Joy B remains high. Transistor 0b is turned on causing the CS pin on the channel 1 chip and others in the same bank to go high resetting these chips.

When the CALL JOYST statement in step 130 is executed Joy B goes low and Joy A remains high. Transistor 0a is turned on causing the I/O clock pin on the channel 1 chip and others in the same bank to go high with no effect. The most significant bit (MSB) of data from the previous analog conversion has been available and is now recognized by the computer. The input reflects the bit sense and is returned by the CALL JOYST routine. When this statement finishes, transistor 0a turns off and the I/O clock pin goes low causing the next bit to appear on the Data out pin. Step 140 examines the value returned and if on, increments a variable for that channel by a weighted value corresponding to the location of the bit in the ADC Serial output byte.

The circuit shown is for channels 1 & 6, but additional ADC chips may be added easily for up to a total of 6 analog channels with no digital joystick using basic. In basic we are limited to two channels per joystick because the CALL JOYST Command will only recognize a legal combination of 2 inputs at a time (up && down or left & right cannot be energized). At the same time this limitation can be overcome and the number of channels per joystick expanded to three by using a decode and an encode chip on each bank. This restriction disappears and speed is increased when assembly language is used allowing up to 10 analog channels with no digital joystick or 5 analog channels with 1 digital joystick with no need for encode and decode chips. A fellow club member is writing the assembly language routine.

In Basic, Call Joyst examines two analog channels and CALL KEY only examines one. Therefore, to keep channel 1 update time to a minimum when more than one channel is desired, they should be implemented in both HARDWARE and SOFTWARE in the following order: 1,2,(3),6,7,(8),4,9. Channels 3 & 8 need decode and encode chips in Basic.

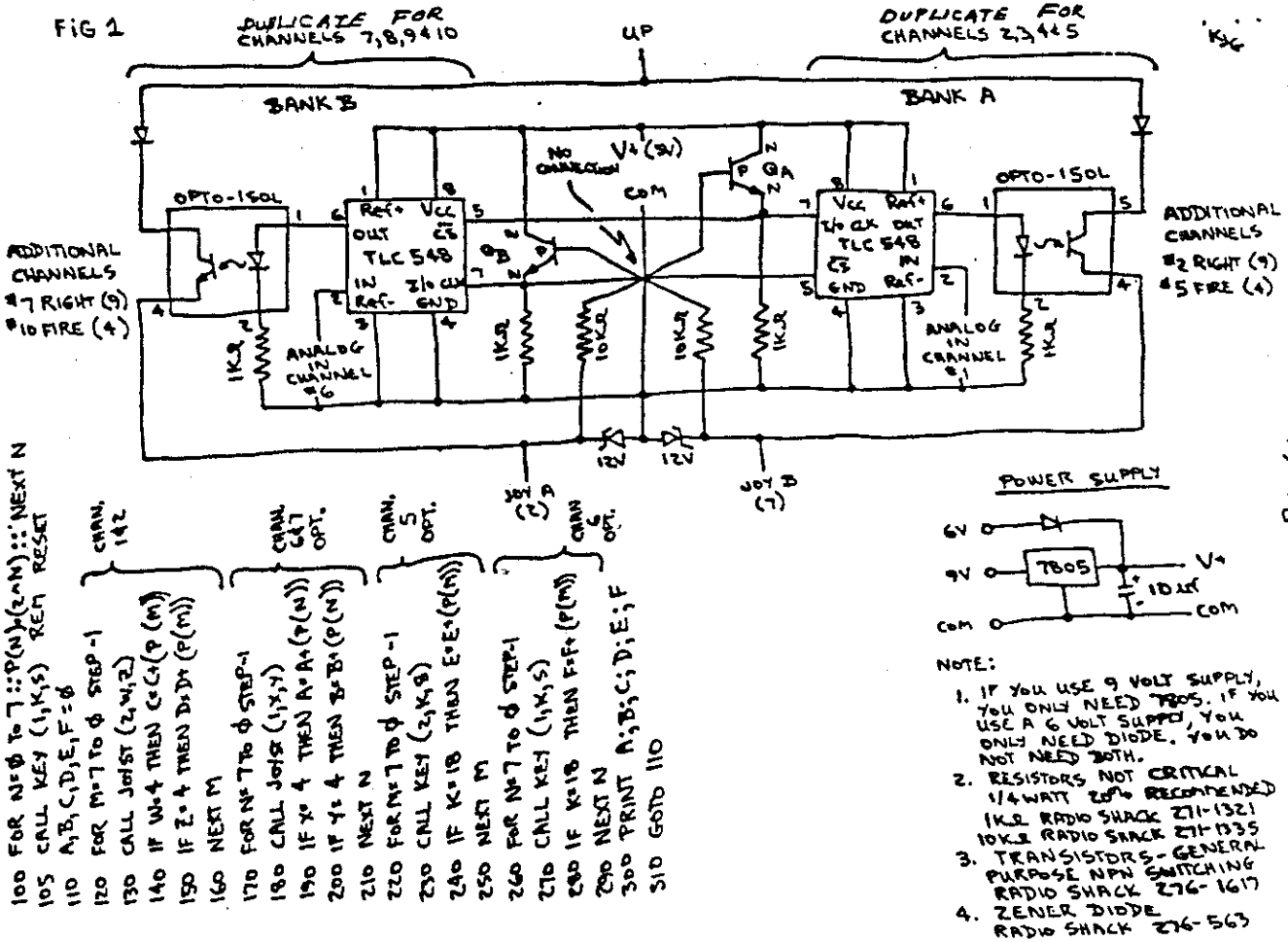
See Fig 7 in this example for a practical application using thermistors to read multiple temperatures. Using the circuit shown, a bare console with multiple analog channels and appropriate sensors could be turned into a local weather station monitoring temperature, wind speed and direction, rainfall, and barometric pressure, etc.

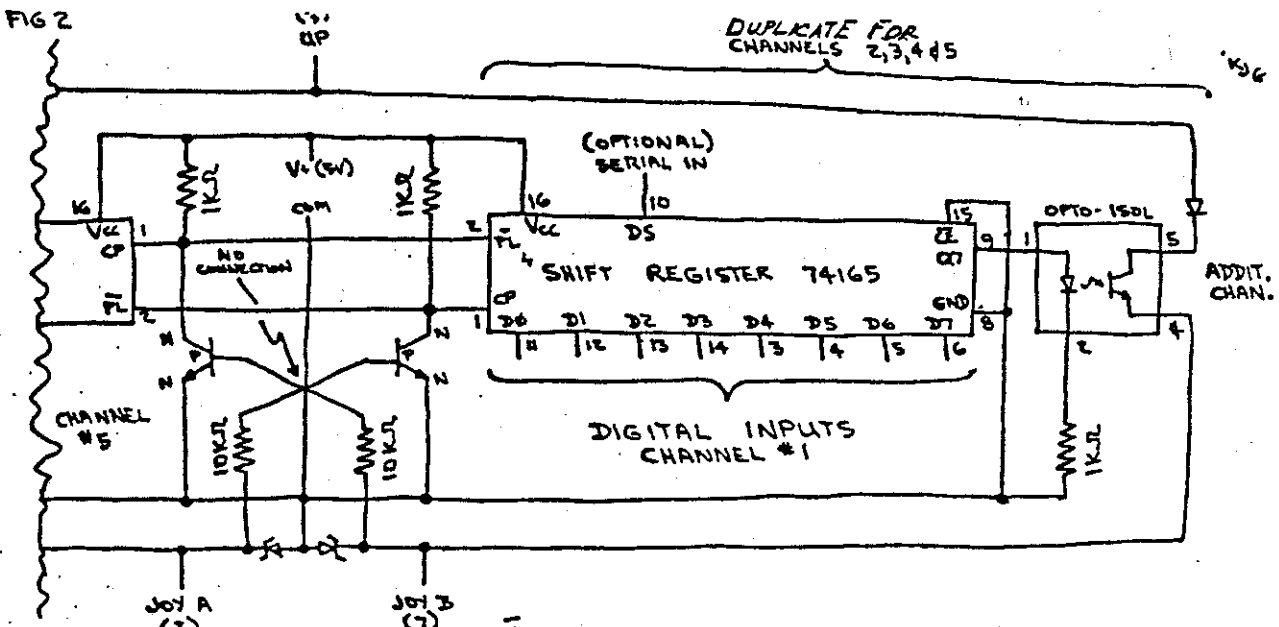
The techniques employed in this project can also be used with a shift register chip (fig 2). This allows 8 or more digital inputs to be read from each of the 5 inputs in the joystick port. The basic program used with the analog chips requires only minor modification to work with shift register(s), the large amount of digital inputs could be used to build a computer based burglar alarm. An analog channel might also be used for the same purpose by using window switches to short resistors in a divider network causing a unique voltage at the input to the ADC (see fig 6).

An alternate method to control either the ADC or Shift register chip would be to use a decade counter chip to generate the CS or PL signals. This would free up one of the joystick outputs but would require using another input (fig 5).

As I've said before, if any of this is useful, you are free to use it in a project of your own. I would appreciate hearing about such projects. You can write to me at : 6440 St. Rte. 86, Concord Ohio 44077. For a more thorough discussion of analog to digital conversion see the article in the July 1991 edition of RADIO-ELECTRONICS magazine.

**CAUTION !!:** Due to the nature of the product, even though it appears simple, the reader assumes all responsibility! Of the figures shown, only Fig.1 has been actually Breadboarded.





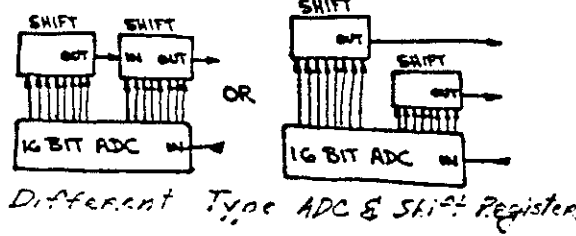
```

105 CALL KEY (1,K,S)
120 FOR M=7 TO 0 STEP -1
130 CALL JOYX (2,M,E)
140 IF M=4 THEN G(1,M)=1
150 ELSE G(1,M)=0
160 IF 2=4 THEN G(2,M)=1
170 ELSE G(2,M)=0
180 NEXT M
190 FOR N=7 TO 0 STEP -1
200 CALL JOYX (1,N,E)
210 IF N=4 THEN G(3,N)=1
220 ELSE G(3,N)=0
230 IF 2=4 THEN G(4,N)=1
240 ELSE G(4,N)=0
250 NEXT N
260 FOR M=7 TO 0 STEP -1
270 CALL KEY (2,K,S)
280 IF M=18 THEN G(5,M)=1
290 ELSE G(5,M)=0
300 NEXT M
310 GOTO 120
    
```

PROGRAM EXECUTION TIME (SEC)

CHAN. 1 & 2	2.67	4	9
BASIC	.75	1.5	2.25
EX.BAS.	.50	1.0	1.50

ASSEM. CALLED FROM EX.BAS.  
ALL 10 CH. EVERY .15 SEC  
NOTE: EACH CHANNEL UPDATED  
ONCE PER PROGRAM SCAN



Different Type ADC & Shift Register

FIG 4 9 BIT RESOLUTION USING (2) 8 BIT ADC'S

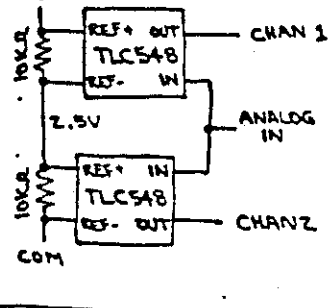
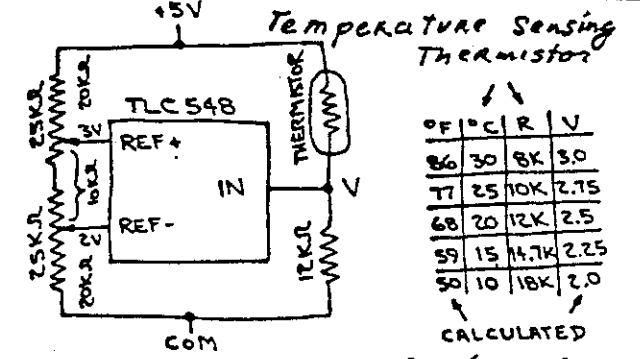
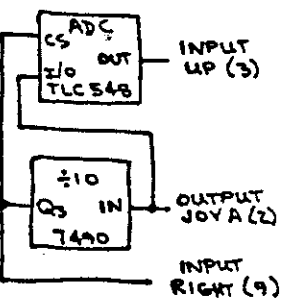


FIG 5 ALTERNATE RESETTING SCHEME



THERMISTOR IS RADIO SHACK # 271-110

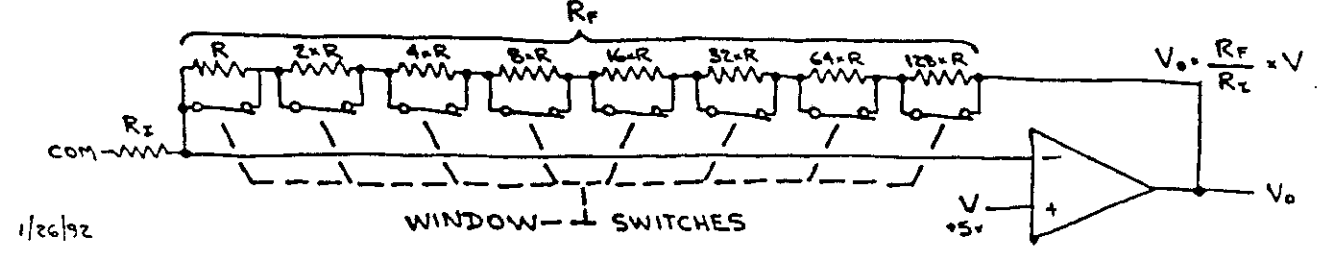
CHANGE PROGRAM AS FOLLOWS:  
100 FOR N=0 TO 7 :: P(N)=(2AN) \* .141  
:: NEXT N REM .141 IS CALIB FACTOR  
110 A,B,C,D,E,F = 50  
REM 50 IS OFFSET VALUE  
VALUES IN LINE 300 ARE ACTUAL TEMP

Temperature Sensing Thermistor

°F	°C	R	V
86	30	8K	3.0
77	25	10K	2.75
68	20	12K	2.5
59	15	14.7K	2.25
50	10	18K	2.0

↑ CALCULATED

FIG 6 BURGLAR ALARM SENSORS



1/26/92

# Feedforth March '92

by Leonard Tabbs


TI BASIC: A MYSTERIOUS PROGRAM. (A COLOR TV or MONITOR needed for best demo.) The following is a miscellaneous unidentified TI BASIC program I found. I ran it expecting to find a simple program that could produce a graphic display of a sine wave. Lacking the advanced math training to tell how one would construct a program to display a sine wave, it was hard to know if this program was originally intended to be for real (per title of 'SINEWAVE') or whether the author intended this program to be what it is as a spectacular display. Type this program in and watch what happens! Use FCTN/4 to stop.

```

1 REM "SINEWAVE" COPIED 2-19
-92 from Unidentified Source
100 RANDOMIZE
110 CALL CLEAR
115 PRINT "SINE WAVE"
120 FOR A=1 TO 16
130 CALL COLOR(A,INT(RND*
1),INT(RND*8+1))
140 NEXT A
150 CALL SCREEN(INT(RND*
))
160 FOR A=1 TO INT(RND+1)
170 X=INT(RND+1)
180 Y=INT(RND+1)
190 W=INT(RND*+1)
200 W=INT(RND*W+1)
210 W=INT(RND*W+1)
220 FOR B=1 TO INT(RND+1)
230 Z=INT(RND*+24)
240 CALL VCHAR(Y+INT(RND*6),
X+INT(RND),Z,W)
250 W=INT(RND*W+1)
260 CALL BCHAR(X+INT(RND*6),
Y+INT(RND),Z,W)
270 NEXT B
280 NEXT A
290 GOTO 120
    
```



## Multiplan Commands

Barry Harasen of the TI Bruikers Groep, Netherlands, attended Fest West '92. While there, Barry passed out some copies of their "newsletters". These newsletters are closer to being a book! We received copies for four months, the thinnest being 28 pages long and, the thickest contains 60 pages. They are very impressive, however, I don't know the language! I managed to figure out what they were showing in the following article. Once they start working with the menus in Multiplan, they used English to help their own members understand what was where. This gave us the opportunity to take advantage of the information one of their members put together for their newsletter.

I am sure these newsletters contain a lot of very useful information. If you can read the language feel free to borrow these newsletters.

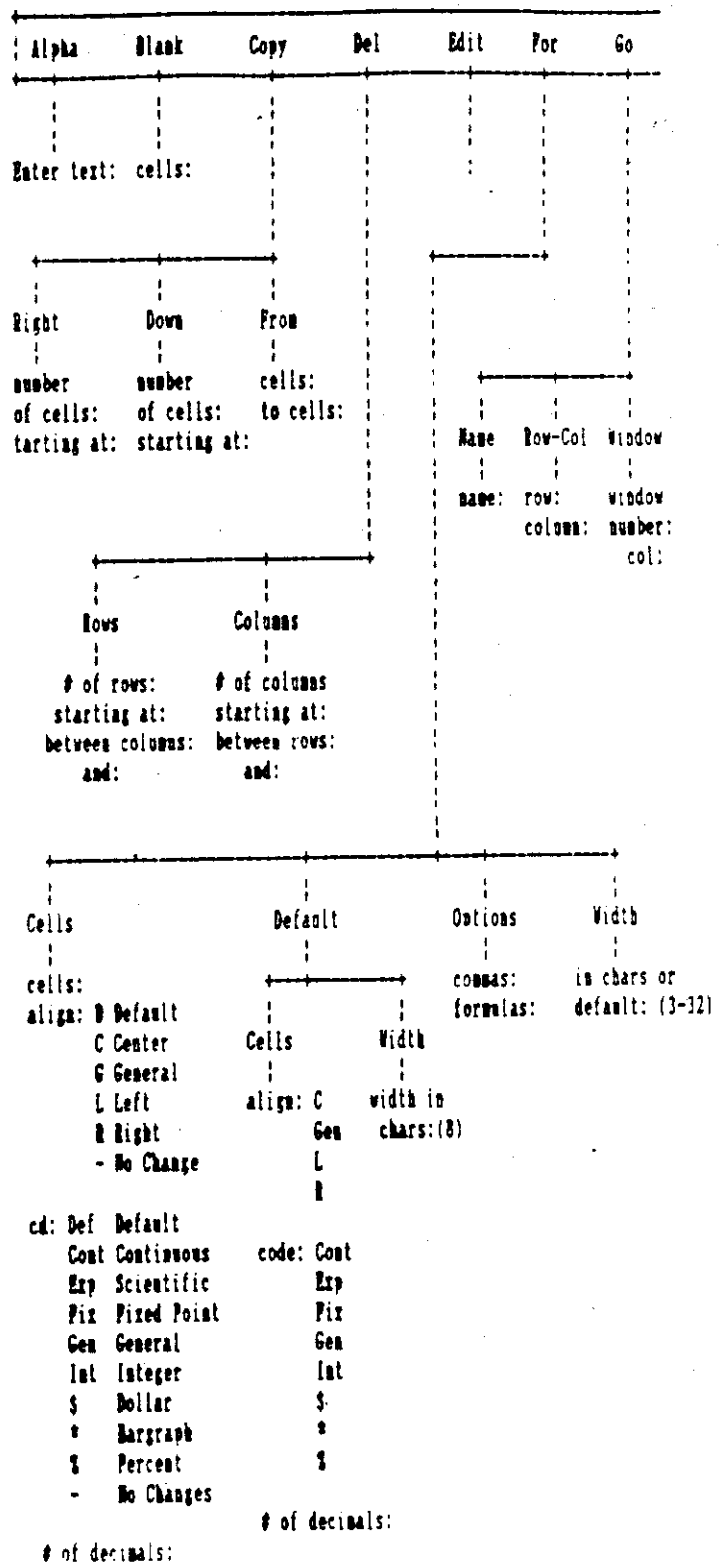
Thanks to Tom Mills for bringing this article to my attention.

-BJ Matrix

SouthWest Ninety-Niners

## Multiplan Commands Menu

from Tiddieen Netherlands









SENIOR COMPUTER NETWORK ... Isobel Warren, left, and son Paul Chvostek are the driving forces behind a new electronic bulletin board for people over 50.



"Acme Television? I want to speak with someone about your on-screen programming feature."

# Computer-age grandparenting

By JUDY CREIGHTON

**G**randparenting from afar may be difficult, but a Toronto journalist and her son are taking a stab at making it easier — using home computers.

"If you want to open the door to kids you have to do it on their terms, and their terms are computers," says Isobel Warren.

Warren, 55, editor of a national newspaper for seniors, and her son Paul Chvostek, 21, have established Silver and Gold, an electronic bulletin board. They say it's one of the few in North America geared to people over 50.

"You communicate with others by keying your messages into your computer and transmitting them by phone line and modem to the Silver and Gold computer in Toronto," says Chvostek.

In turn, Silver and Gold will transmit the messages to the computers of its entire membership, or just to individual computers.

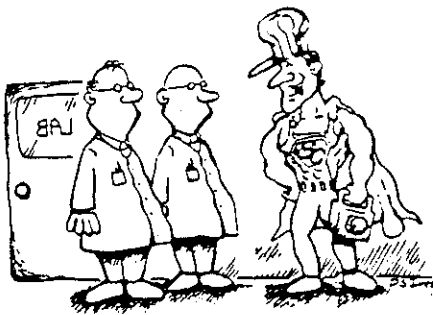
"So many families have home computers and most children are trained to use them. It's a wonderful and easy hobby for grandparents to learn, with so many benefits," says Warren.

Using the electronic bulletin board, a grandparent in Halifax could send messages back and forth with a grandchild in Victoria.

At the moment, the cost of setting up and running Silver and Gold is being absorbed by Warren and her son as they seek members.

Chvostek is available to give free advice to seniors interested in joining the bulletin board network at (416) 658-9676.

- Canadian Press



I've heard you were looking for me. Super conductor!