

SBTIUG GENERAL MEETING 1 SEPT, 1988

by Norm Knudsen

The September meeting of the SBTIUG was held at the Saratoga Library on the first Thursday, September 1st with 17 members in attendance.

The meeting was opened by President Mike Ewell at 7:20 PM. The treasurer's report was given by Ken Daberkow. He stated that the club has a current balance of \$642.99, although the August and September newsletter bills are outstanding.

The BBS is apparently still down although no one was quite sure of the exact problem.

There is a copy of the new ARC 3.02 in our library which will uncompress and unpack in one pass.

Our PR man, Don Apte then gave his usual complete run down on upcoming show and swap meets including one on building your own PC. Call Don for details.

Since the first Thursday night in October is not available to us, it was agreed that those attending would prefer Saturday the First rather than the 15th. The next meeting then will be Saturday, October 1, at 3:30 to 6:00 PM.

The business meeting was then concluded and was adjourned to a Demo by Helmut Fuchs on some of the finer points on TI Writer.

TREASURERS REPORT

by Kevin Daberkow

PLEASE look at your mailing label to see if some color has been added. If your membership expiration date has been high-lighted in RED, this is your last issue until you renew. If your membership expiration date is in YELLOW, then you should renew at the June meeting.

>> THE DUES ARE \$15 PER YEAR <<

NOTE: Your membership expiration date can be found on the last line of your mailing label.

If any of the information on your label needs to be changed, please let me know. Call me at (408) 281-7435 or write to me at the following address:

SBTIUG - Treasurer
P.O. Box 23447
San Jose, CA 95153-3447

There were no renewals during the month of September.

The club paid Keith Felix the sum of \$66.34 to cover the cost of a new Hayes compatible modem for the BBS. There are still outstanding bills for the September and October newsletter mailings.

The club now shows a balance of \$576.65 in our treasury at the end of September.

To 'C' or not to 'C'
by Kevin Daberkow

For many of our club members, the above question is quite valid. They are lured by the promises of increased speed and 40 columns, but don't really know how to get started or what to write.

In the coming months issues, I plan on writing some articles that will hopefully help some of you take the plunge. What I am asking for prior to these articles is some INPUT from you. At what level should I cover material? Should I compare the C code to equivalent EX BASIC code? I would prefer to build a working program piece by piece, but what kind of program? There already have been a number of articles printed in our newsletter telling how to "get started", so should I cover this again?

Responses to the above questions can be left for me on Frank's BBS or the SBTIUG club BBS (look on front of newsletter for numbers). Leave messages to Kevin Daberkow. Also, I would be glad to talk to you at the next club meeting. This is your chance to help determine the course of the articles. Don't pass it up!

I am looking forward to all the wonderful input I will be receiving in the near future, till then.

Kevin

Editors Ramblings
By Bill Schult

I would like to thank those that have contributed to this month's newsletter. These contributions are greatly appreciated. Articles from the members is what makes a great newsletter, so keep those articles coming in.

Our resident 'C' language expert has stated that he is willing to write a series of tutorials on the 'C' language. So here is an opportunity for all of us novices to learn a new language. These articles will be greatly appreciated and we look forward to reading them. Now I would like one of our other experts to write an article on those funny looking programs where data is entered by 'plugging' numbers into designated addresses. An explanation of exactly what is happening here would be very enlightening, and educational.

THESE FILES WERE DOWNLOADED FROM THE LOS ANGELES 9640 BBS
ACCESSIBLE THROUGH PC PURSUIT CALAN AREA AT 947-7777

The first tells you how to improve cooling in the PE Box by improving airflow or by installing a better fan. The second tells you how to install a new voltage regulator to increase the power capacity to handle power hungry drives - even hard disk drives - in the box.

IMPROVING P-BOX COOLING

The following instructions will guide you in improving the cooling of the P-box without installing a fan with higher air flow.

Remove the outer cover from the P-box, and use duct tape or adhesive weatherstrip to seal the large gaps under the "floor" of the card area. This is not nearly as hard as it sounds, only about 15 screws. Just be sure not to pinch the drive wires going behind the card area when you slide the cover back on. If you have a Triple Tech, or any other card without a case, sandwich it between two other cards to channel the airflow to it. Use duct tape to seal the holes in the floor of the card area. It is a good idea to make a small cardboard air dam to put on top of the transformer (hold in place with duct tape) to direct air to the power supply board, especially at the center area, where the regulators are (see the file on increasing P-box power output). An overheated power supply will give the same symptoms as one that is failing, or doesn't have enough power for your drives: It overwrites files! Cooling of the power supply area will not be improved or decreased by these procedures. If you want to improve that, and/or cut down on noise, obtain a 3 1/8 inch square boxer fan with no less than 10 watts power consumption. Fans with less 10 watts of power will have less airflow than the original. Also, fans with more than about 15 watts may make as much or more noise than the original, but will REALLY cool the box. My fan is 23 watts, makes a little more noise, and the cards are hardly warm to the touch when the box is opened! Be sure that the electrical connections are sound; it is a good idea to already have the wires made up and attached to the fan. The wires should have automotive male spade type terminals on them. If you have the patience, you can also enlarge the holes that the fan exhausts from, or cut out that area entirely. Also, if you have your P-box pushed up very close to a wall, the hot air put out by the fan is being recycled into your supposedly cool air intake. To prevent this, make a cardboard deflector to fit around the fan opening. It should stick out about 1 1/2 inches on the side near the card area, but only about 1/2 inch on the bottom and on the side away from the box. It should aim upwards and away from the box. This will deflect the hot air away from the intake area. Be very careful that you don't restrict the airflow very much!

P-BOX POWER SUPPLY BOOST

The following directions will increase the output of the P-box power supply to handle 2 power hungry drives.

Symptoms of lack of power include overwritten files, and the "sectors used" count on a disk catalog does not match the sum of the lengths of the individual files. Also see the file on P-box cooling, as an overheated regulator, regardless of its capacity, will cause the same problems! CAUTION - It is possible to damage the system and cards if this is not performed correctly!

Parts list:

1 12V regulator, RCA # SK9341/933 or equivalent

4 3 amp diodes, RCA # SK3848/5806 or equivalent

1 1000 uf capacitor, at 35 volts

1 clip on heat sink (# - see note at end!) for the regulator (important)

small amount of silicon heatsink compound

Remove the P-box cover by removing the 15 or so phillips head screws and sliding the cover forward. Remove the power supply board (the only printed circuit board in there) and disconnect the wire connectors by pushing the tabs through the board and pivoting them away. You will see the regulator in the middle of the board. Carefully unsolder the connectors on the back, and use solder wick or a suction device to remove the melted solder. DO NOT force it, but gently lift it away as you reheat each connector pin to soften any remaining solder. Apply silicon heatsink compound to the back of the new regulator, and install it in the same place. Be careful to apply only as much solder as needed, to avoid making a solder bridge underneath the regulator. Install the heat sink. Immediately next to one end of the regulator, you will see a small black 47 uf capacitor. Cut the leads of the 1000 uf capacitor to a reasonable length, and solder the + lead to the + lead of the small black capacitor, and the - lead to the - lead of the small black capacitor. This will give a boost to the drive motor to help get the disk turning when a drive is selected. Now for the hard part. The 4 diodes must be installed in place of 4 smaller diodes on one corner of the card. You will see 8 small tubular devices in a row on one corner of the card. Four of these (the dark colored ones with a colored stripe or band on one end) are the diodes. The other four (lighter colored) must not be disturbed. Carefully note which direction the bands point (my board had markings on it to indicate direction, but all boards may not have them), and then remove the diodes. Use the same methods used to remove the regulator. Now you must slightly enlarge the holes to match the wire size of the new diodes. Use a small drill of just slightly larger size than the wire to do this. Use a pin vice, or some other holding device to do this by hand. DO NOT attempt to use a mechanical or electric drill to do this. Watch the trace on the board carefully while drilling to be sure that you are not ripping it off the board! Cut the wires on the new diodes to about

WHAT TO DO WITH A TI-99/4A
(when you have an IBM/clone)
Ron Springer

A couple of months ago Mike Ewell asked me to do an article for the newsletter about why I continue to use my TI.

The answer is that I don't! The reason however is simply that due to my job I haven't really USED any computer at home for the last two years, including my IBM clone.

My TI has been used for FOUR basic purposes:

1. GAMES - My son continues to use the TI for this purpose, but I have never been one to dwell on games. My only use for games is to make them available to others or to use them to validate some aspect of any system on which they run. Games are somewhat unique in that they frequently bend the rules to achieve a particular characteristic, and can therefore be used for compatibility testing and verification.

2. TEXT - I say text rather than word processing because I also want to include communications (i.e. modem/bulletin board use). For this purpose the TI works quite nicely, especially when you consider the quality of Funnel-Mob and Fast-Term.

BUT there isn't anyone that I know that won't trade a display of 80 columns for one of 40, in fact I really prefer displays with 132 columns.

3. SPREAD SHEETS - Multiplan was the top rated spread sheet throughout the industry for a substantial period of time, and that is what TI chose to support on the 99/4A.

I used that program for two years as the treasurer of the Home and School Club at my son's elementary school. The checkbook reconciliation even had to be split into a second file before the end of the year. I no longer have a need for that type of program at home.

4. PERSONAL - This is why I bought a TI-99/4A. The timing of my purchase was set by the thought of a home unit upon which my son could learn. Selection of TI over Apple, Atari and Commodore was made on the basis of my background in computer design.

Due to my association with the supplier of 6502 devices to both Apple and Atari, I could have made a better deal on either of those machines but I chose the TI because it represents a REAL computer architecture (all-be-it memory/memory which I feel is a lesser architecture than one with tightly coupled registers). With the advent of 8086/8088 based "IBM" machines and the 48000 based Macintosh, I still feel that the 9900 is a better 'learning' machine because the Intel and Motorola cpu's are basically 'micro-controllers' with 'computer-like' appendages to support high performance and larger memory models.

Another reason for selecting the TI was the speech capability, and when I again have sufficient time I will continue to play with this feature.

During the past two years, my home computer operation has been limited to using my IBM clone as a computer terminal, use of a particular program available only for IBM, and to validation of the graphics chip-set that I have been working on (there's those **** games again).

In order to validate the video interface with a potentially bad graphics chip (you might lose the visual feedback) you need to have a reliable text device, i.e. you need a separate 'terminal'. Will my 99/4A work??

OF COURSE!!!

The IBM operating system supports a common concept called RE-DIRECTION which allows the console 'device' (keyboard and corresponding display to those un-initiated few) to be sent

or re-directed to a different 'device' -- how about an RS232 or COM port?? This 'device' re-direction is also familiar to TI-99 users who program with the UCSD-Pascal system.

For those of you out there who have an IBM/clone or other system that supports this redirection, here's how you do it!

You need a terminal emulator program (I use Fast-Term) for your TI, an RS232 port in the 'host' machine (in my case an AT-clone), and a modem cable for the IBM (not the one used for your TI) which uses 1-to-1 wiring of pins 1, 2, 3, 4, 5, 6, 7, 8 and 20.

CAUTION!!!

The 25-pin connector on the standard TI RS232 card has two interface ports. If you do not use the Y-cable on your TI, you must insure the modem cable does not short port-2 of the TI output. This will be automatic if you are connecting to a 9-pin type IBM port.

A word to any communication geniuses out there, the computer is correctly called a terminal (DTE) and the modem is called the connection equipment (DCE). TI incorrectly wired the RS232 ports as DCE in an apparent effort to simplify printer connections. This has caused considerable consternation for those of us with modems (especially BBS systems) who would dearly love to have data-set-ready (DSR) data-terminal-ready (DTR) and carrier-detect (DCD) fully and correctly functional.

RUN the terminal emulator program on your TI and setup the 'protocol' to match the 'protocol' that you will use on the host machine. I use 2400 baud, no parity, 8 data bits, 1 stop bit on port 1 of my TI. Higher baud rates work OK from the TI to the IBM, but data from the IBM can easily over-run the TI when using Fast-term.

On the host machine, setup the 'protocol' for the port in use, i.e. MODE COMn:2400,n,8,1 on the IBM. Then we are ready to re-redirect:

CTTY COMn will redirect both keyboard and display to the TI. In order to return to the IBM, you will type CTTY COM at the TI keyboard.

COMMAND > COMn is more typical for my purposes, as this allows me to re-direct only the display for the command 'shell' while most applications will still direct their output to the IBM display. The return to the IBM is accomplished by EXIT which leaves the 'shell' and returns to the standard output.

A cute but not totally practical form is to reverse that greater-than sign like COMMAND < COMn. I'll let you ponder this one with the hint that you will lose extended keycodes!

Additional usage of this re-direction is described in the DOS manual for the IBM operating system. A specific example where re-direction of the console is appropriate is during the use of a debugger, and especially with SYMDEB which also supports re-direction in its internal workings.

One final comment for this article is about my modem setup at home. I have my modem on a two-way switch in an unusual way, I use the switch to move my IBM between the modem or my TI. I log-on with my IBM as a terminal which gives me access to my hard disk for storage. If I see a TI program, I can download it with the IBM and then switch the IBM into my TI for a back-to-back transfer at my leisure, using X-modem as a common link between both machines.

3/8 inch length, and insert them carefully into the holes. You may need to bend some of them slightly to get a good fit. After they are all inserted, CAREFULLY solder the ends, being VERY careful not to connect ends that were not previously connected by the traces on the board! Now carefully reinstall the board and the connectors, and before installing the cover of the P-box, turn it on. Quickly check for 12V across the 1000 uf capacitor leads. If all is ok, reassemble the P-box, taking care not to pinhh the drive wires along the back of the box. (!) If you will only use the drives normally, this is sufficient. However if you have unusually power hungry drives, or will use them for LONG durations, such as in a BBS for uploads and downloads, or for lots of copying, the clip on heat sink is not sufficient. You will need to use wires to "remote" the regulator, and attach it to a LARGE heat sink, placed in front of the fan, or just attach it to the back of the P-box, as near the fan as possible, in the same way you would mount it to a heat sink (You will have to drill holes for the wires and the mount bolts.). If you use a separate heat sink, be SURE to use a ground wire from the sink to a chassis ground, or mount it with a screwed on metal strap!

A CHILD'S FIRST LOOK AT MULTIPLICATION
 by Tony Falco
 reprinted from MUNCH Newsletter

"Dad, today I heard someone talk about three times four. What does that mean?"

"get that jar of pennies in your room and I'll try to show you. Three times four means three fours. Make a row of four pennies. Now make two more like that one.. That's four plus four plus four or twelve. You can see it is also four rows of three each. Four times three is the same as three times four. hey, we could do this on the computer."

The program listed below grew out of a conversation much like the one above. I strongly believe that understanding numerical concepts at all levels is aided by, or maybe even only possible with, a physical or mental picture that gives the learner a comfortable and familiar feeling. With this in mind, I try to help my child build a conceptual basis for ideas that tend to become rote with time. With a program that is conceptually sound, the computer's infinite patience provides an ideal means for acquiring insight into basic numerical concepts.

To use the program you pick the highest factor the child is to work with. For example, for products up to 6x6 or 36 pick 6. The program randomly picks problems and displays the problem with an array of astericks to show the factors to be multiplied. The user types an answer. If

wrong he tries again until correct. The format in which the array appears on the screen encourages the child to try repeated additions or, as a last resort, counting. To end the program enter <Q> instead of an answer. The interaction between computer and learner is reinforced with speech. I feel this makes a program more friendly and familiar. The TI-99/4A has been, and continues to be, the only machine for inexpensive, high quality and easy to use speech.

```

10 FOR S=1 TO 14 :: CALL COLOR(S,2,12) :: NEXT S
20 CALL SCREEN(12) :: CALL COLOR(2,2,16) :: CALL CLEAR
30 INPUT " HIGHEST FACTOR = ":H
40 RANDOMIZE :: CALL CLEAR
50 A=INT(H#RND)+1 :: B=INT(H#RND)+1
60 IF A#B#H#H THEN 50
70 P$=STR$(A)*"x"*STR$(B)+"=" :: DISPLAY AT(5,12): P$
80 S$=RPT$(" ",B)
90 FOR T=1 TO A
100 DISPLAY AT(T+7,14-LEN(S$)/2)+S$ :: NEXT T
110 K=(12-LEN(S$)/2+(A)/9)
120 DISPLAY A(B+INT(A/2),K)SIZE(-2):STR$(A)
130 DISPLAY AT(9+A,13):B
140 ACCEPT AT(5,12+LEN(P$)):BEEP:ANS$
150 IF ANS$="Q" OR ANS$="q" THEN 180
160 IF VAL(ANS$)=A#B THEN R=R+1 :: CALL SAY("GOOD" ::
    GOTD 40
170 CALL SAY("SORRY TRY AGAIN") :: W=W+1 :: GOTD 140
180 CALL CLEAR :: PRINT ::::::: PCT=INT(100#R/R#R)+.5)
190 PRINT " ";W#R;"TRIED"
200 PRINT " ";R;"CORRECT"
210 PRINT " ";PCT;"%"
220 CALL SAY("GOODBYE")
230 END
  
```

USER-DEFINED FUNCTIONS

by Steve Karasek

reprinted from the COMPUTER BRIDGE

User-defined functions in TI Basic and Extended Basic can be helpful when you sit down to write a program. I generally use functions when I have a numeric or string expression that would be repeated many times throughout a program. Let me give a few examples.

Suppose you will be rounding cash amounts to the nearest dollar. Somewhere in your program (I usually place it in the beginning, but it really doesn't matter) type the statement:

```
DEF ROUND(X)=INT(X+.5)
```

Then whenever you wish to round a value, you can use this function. For example, `A=ROUND(B)` will assign to A the value of B rounded to the nearest dollar (or the nearest integer it doesn't have to be dollars amounts). If you wish to round to the nearest cent (two places after the decimal point), change the function to:

```
DEF ROUND(X)=INT(100*X+.5)/100
```

One problem with functions is that they can only take one argument. It would be nice to write a function such as `ROUND(X,D)` which would round the value of X to D places, but this is not allowed in TI Basic. You can use other variables in your function definition, but they have to be assigned a value before calling the function. For example, rewrite the rounding function as:

```
DEF ROUND(X)=INT(10 D*X+.5)/100
```

To use this function, first set D to the number of decimal places that you want (for cents, you would use D=2). Then use the function as before.

Functions can be used for strings as well as numbers. If you will be looking at the first character of a string in a number of places in your program, you can define a function:

```
DEF FIRST$(X$)=SEG$(X$,1,1)
```

You may want to abbreviate the name as `F$` if it will be used very often. Now you may use statements such as `IF FIRST$(S$)="Y" THEN...`, which will check to see if the first character of X\$ is a Y.

You can pass a string variable to a numeric function, and vice-versa. You can also use one function within another. For example, if you already have defined the function `FIRST(X$)`, you can define a logical function `YES` that will return true (-1) if the first character of a string is a Y as follows:

```
DEF YES(X$)=FIRST(X$)="Y"
```

Functions can save a lot of typing and program memory, since long expressions can be reduced to a few characters. However, using functions is S-L-O-W especially in large programs or when functions call other functions. If you call a function in a time-critical part of your program (for example, inside a FOR loop), it may be better to write out the expression. One way to get some speedup is to type in the function definitions last. It doesn't matter if they are at the beginning, middle, or end of the program (in other words, the line number doesn't matter) as long as they are the last line that you type in. When searching program memory for function definitions, Basic will look at the most

recently entered line first, so it will find the definitions faster this way.

If you want to get an idea of how slow function usage is, type `TRACE` before running your program. This will show the line number of each line as it is executed. You may notice a significant pause on lines that call a function. type `UNTRACE` to turn off tracing.

FUNNELWEB 4.0

reprinted from HOUSTON USERS GROUP

If you have Funnelweb 4.0, you know it has a few capabilities beyond TI-Writer; here are some of them:

In Editor, to change text from lower case to upper case, place the cursor at the start of the text and hold down CTRL and ";" together, Magic! To change from upper case to lower case use CTRL and "." together. Great time saver if you want to change a group of words for headings, etc.

In SD (Show Directory), the asterick between the SIZE and the TYPE columns indicates a FRACTURED FILE; one which is stored "split up" on the disk.

Also, you may select a file from SD by pressing that NUMBER from the list, and it will become the Default File name for Load File and Save File, or for Delete File or View File, EXCEPT that it will not accept the number for a PROGRAM format file. Keeps you from trying to load an X-Basic program into the text editor!

SD now prints the directory if you press "P". On the SD directory in Funnelweb, PROGRAM mode files can be flagged by pressing "=" and will display (in the REC column) either a BX (for BASIC/X-BASIC Runnable program), or by EA (for a Memory Image (E/A 5 type)) program to be run from Funnelweb Loader #3, or remain blank if it is a data file in "Program format: CHARA1", for instance.

The Editor/Assembler Loaders in FW 4.0 are accessed from the Ed/Asmbr option of the 1st Menu if XBasic is being used.

If you use the Funnelweb Loader #4 on a DF/80 file, it will automatically reveal the REF/DEF table names, one of which will be the "Program Name" to "Start" the program, if the program does not auto-start. It will then allow you to select which one you want to try to use. Usually only one or a few will show up, but this can be a real ice-breaker with that game your buddy gave you last year but forgot to give you the "Program Name" for (and the name was "NEVER")

CONFIG program on FW 4.0 will load and run ONLY via the Menu selection CONFIGURE on the first Menu, and only if you have loaded FW via X-Basic. You can configure the Editor and Formatter print devices, plus some other choices, plus putting your own programs on the menus. If you know which of the loader type numbers matches each of your programs.

You may change the Print Device name and set printer codes for condensed, etc, for DM-1000, by using FCTN 3 at the first screen in DM1000, and then answering Y for the "Save to disk?" question. All the CONFIGurations require the write tab to be off the FW disk, of course, so do it on the BACKUP or WORK copy, not on the original.

Lets Talk Ram Disks part 7
by John F. Willforth

The Rave 99 "MX01" memory enhancement is a P-Box card which replaces the 32K card and gives you many, varied and useful and unique enhancements in the TI 99/4A system environment. The unit comes in three models:

MX01/64- contains 8K of memory
(>6000->7FFF)
8K of non-DSR
8K of system DSR
8K of user DSR
32K of main memory

MX01/288-MX01/64 w/224K additional
64K in MX01/64
224K additional

MX01/544-MX01/64 w/480K additional
64K in MX01/64
480K additional

The MX01/64 is the base unit and it is on this unit that upgrades can be done to the larger two. Rave offers upgrade Kits for the two smaller units which consists of the necessary STATIC RAM chips (256K bit low power)

The support power for the MX01 is by either the "super capacitor" that comes with the board and will hold the memory (on the /544K unit) for up to 5 days. If longer protection is required, a 3 volt lithium battery may be installed.

The basic unit cannot be used for the RamDisk application due to the absence of additional RAM. The /288 and /544 do have the additional RAM and using the special DM-1000 supplied with the Ram-disk option as well as a CONFIG program the necessary configuring, volume and file manipulations can be performed. A 1444 sector (DSSD), 360 sector (SSSD), and 119 sector (1/3 SSSD) can be placed on the /544. The /288 unit supports a 720 sector (DSSD), and a 179 sector (1/2 SSSD) pair of Randisks. Note that the upcoming PRINT SPOOLER option will probably affect the smallest (1/3) and (1/2) Randisks units available in the /544 and /288 because the memory used by these units will be needed by the spooler.

The memory space >6000 to >7FFF can be used by any SuperCart or SuperSpace applications that currently run out of this space.

The non-DSR space is memory space where the Peripheral Device Service Routines is paged into the CPU's own memory. To further explain, most of the computers on the market, require loading a driver program into memory normally used by the CPU to provide the operating system running in the CPU a means to properly control the particular peripheral. This wastes RAM and adds to the cost. TI put no RAM in these areas, but selects a ROM based program which is located in a particular controller card such as a disk controller and switches it into this "hole" in memory. This area can have RAM assigned to it and switched in anytime needed or when no DSR is being used.

The fact that the system calls for peripheral DSRs

means that you can write your own DSRs for the SYSTEM as well as USERS DSRs. The MX01 will allow for you to have these types of DSRs running thus giving you more flexible and varied use of the TI. Special memory mapping of addresses >4000 to >7FFF allows an extra 16K bytes for assembly programs (48K) to be written without any user memory mapping required.

The keyboard macro loader option allows you to assign a single key depression to output for example, a particular string (CTRL-M "PIO"). Macros are not available for basic or extended basic, but will work in programs loaded under option "5" of Editor Assembler, such as TI-Writer and many of the terminal emulator programs.

The MX01 can be configured with as many as four (2 MCG) in a PEB. You cannot run a TI Thermal Printer, but most everything else, including Horizon Ram Disks will work with it.

PRICES: (upgrades)

MX01/64 199.95 /64 to /288 110.00
MX01/288 309.95 /64 to /544 220.00
MX01/544 419.95 /288 to /544 120.00
Randisk software 25.00
Keyboard "MACROS" software 15.00
"PRINT SPOOLER" software 15.00

RAVE 99 co.
112 Rambling Road
Vernon, CT. 06066 (203) 871-7824

TIPS FOR DISK DRIVE SERVICE
reprinted from THE SUNCOAST BEEPER

PROBLEM: read/write errors

SOLUTION: Read/Write errors may be caused by a number of problems. Two of these problems, which are not unusual, are a worn pressure pad and a slipping drive belt.

Disk drive belts, like any other rubber problem, become brittle and slick with age. If the error symptom is predominantly READ errors when the drive is trying to read outer tracks, the problem may very well be a worn belt. Inspect the drive belt and if it is showing signs of its age, replace it. Then run the disk or disks for which the problem occurred to determine if this has solved the problem.

Another frequent cause of read/write errors in floppy disk drives is a worn pressure pad. The pressure pad presses on the disk, holding it against the read/write head. When the pad becomes worn, the pressure is insufficient and the current put out by the head is too low, causing errors. The solution is to replace the pressure pad.