

8706,07,08
092

Vol. 5 No. 7

SUMMER 1987

This newsletter is published jointly by OH-MI-TI and New Horizons TI-99/4A Home Computer Users' Groups. Material may be reproduced without permission provided the author and source are acknowledged. For more information concerning TI Users' Groups in the Northwest Ohio area, contact:

Robert Peters
President, OH-MI-TI
225 S. Wheeling
Oregon, OH 43616
(419) 693-7934

////////////////////
(419) 385-7484
TICOMM BBS
>>> 24-HRS <<<
SYSOP
> MILLS - TURNER <
////////////////////

Don Turner
President, New Horizons
5533 Fleet Street
Toledo, OH 43615
(419) 537-1454

THE NEWSLETTER STAFF

Bill Tiep
Phil Bennis
Dave Burkette
Bud Mills

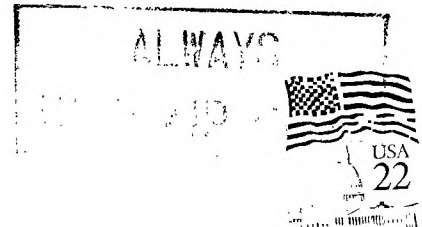
Kent Sheets
Marilyn Schafstall
Earl Hoffsis

LOCAL CONTRIBUTIONS BY;

**NORTHWEST
OHIO**



99ER NEWS



Bill Sager
612 Meadow Springs,
Maumee 43537
OHIO

Edmonton 99ers Computer
PO Box 11983 Users
Edmonton, Alberta
Canada T5J 3L1

NEW EPROM FOR THE HORIZON RAM DISK

by Don Turner

Peter Hoddie has developed an EPROM (eraseable/programmable read only memory chip) for the Horizon RamDisk. It is being distributed by Genial Computerware Box 183 Grafton, Ma 01519

Currently I have 2 HORIZON RAMDISKS, all of of the cards in the P-BOX are TI except for the RS232 card. This article will be based primarily on that type of a system configuration. I did put the RamDisk with the EPROM in another P-BOX using 4 other RamDisks and CorComp controller and 32K turbo memory in the console with no problems at all!! I have been using this EPROM for a few weeks now and it seems to perform as advertised. It has eleven (11) related calls built in as well as an automatic power up option that can be enabled or disabled. The only restriction that I have noticed is that the RamDisk with the EPROM works best at CRU 1000. I will explain this in more detail later in this article. I put the other RamDisk at CRU 1500.

Before I get too involved with the calls available I will give the call name only and not use any of the parameters associated with them. The resident calls in the EPROM are HDDIR, HDDN, HDVOL, HDSZ, DM, BOOT, MD, EAS, ILD, LD and LLD. The first call HDIR will give you a directory of the RamDisk and the current drive number the RamDisk is at. Call HDDN changes the drive number 0-6. If you use zero it makes the card transparent to the system. Call HDVOL changes the name of the RamDisk for you. Probably the most important call is HDSZ this set the size of the RamDisk. This size can be set at 90K or 180K and 256K. You get a little bonus of 4K when using the EPROM so you can really set the size at 94,184 and 260 respectively. This may cause some problems with some disk managers so you may want to stay with the normal sizes. You must be careful when you change sizes, this will initialize the RamDisk at the same time resulting in the loss of your data currently on the Horizon RamDisk. When you initialize the disk (I used DM1000) the process is quite fast. In fact it is done in a wink of an eye. It appears that the disk is being "swept" rather than initialized. It is quite a time saver and a file could be recovered if you made a mistake.

The next 5 calls are program image loaders (Editor/Assembler option 5). Call DM loads the files named MG and MH. These files would be your disk manager files. For those who are using DM1000 simply re-name the files MGR1 and MGR2 to MG and MH. This RamDisk would make for an ideal location for the disk manager to reside on because of the CRU address it would be looked at first. In fact your not left with much choice because if you didn't have the files MG and MH on the RamDisk you would get an error and be returned to the title screen.

Call BOOT will load and run any (Editor/Assembler option 5) file named BOOT. The CALL BOOT will Auto exec on power up if you want it to. With the EPROM is a disk with a file named BOOT it is a modified version of John Johnson's Menu program. This where you can put this now famous program at and have a menu on power up. For those who have one Horizon RamDisk and like the Menu program this is where you would

want to put it.

Call MD loads program image files MD and MH (Editor/Assembler option 5). Call MD would be the terminal emulator programs you like to use. Of course it does not have to be a term program, you can put any program that uses two files in that position.

Call EAS ("filename") will load any EA option 5 file that you put in the parenthesis. Call EAS will load the file UTIL1 from the RamDisk

Call ILD is the same as CALL INIT this call has me baffled as to why it is available. Perhaps a more useful call could have been put there.

Call LD ("DSKn.filename") loads EA option 3 files from any device and if it is an auto start program it will run automatically.

Call LLD ("name") is the same as CALL LINK in Basic or Extended Basic.

The EPROM has worked well with Horizons operating system and with John Johnson's Menu program 6.3 on the other Horizon RamDisk. However because of EPROM's operating system if you have a Horizon RamDisk at CRU 1000 and the other Horizon RamDisk with the EPROM at another CRU address the Horizon RamDisk will conflict and seize up the system. This will only happen if you have John Johnson's program set to auto exec on power up. If the Menu program is not set to auto exec on power up then the system will perform in a normal manner.

I first started out with John Johnson's operating system and Menu program on the Horizon RamDisk residing at CRU address 1500, and left a variety of files on the Horizon RamDisk with the EPROM residing at CRU address 1000. This created a problem whenever I would type in CALL DM from either basics. The EPROM on the Horizon RamDisk at CRU 1000 would try to load and run the file MG and this would result in an error if there was not a program image file named MG on the Horizon RamDisk with the EPROM and the "call" would be aborted and never get to the Horizon RamDisk with John Johnson or Horizon's operating system to load and run the file MGR1. It seems that the EPROM will not allow a continued search of the other CRU addresses. Also I could not access the Horizon RamDisk at the higher CRU address in Basic or Extended Basic. This could prove to be rather difficult at times for users of more than one Horizon RamDisk. Unfortunately I do not have a CorComp Triple Tech card or any other specialized card to see if they too were blocked off as well. The RS232 and Disk controller cards were not affected in any way. I was left with only one option and moved everything to the Horizon RamDisk at CRU address 1000 and let that be the main Horizon RamDisk to operate from. After spending about an hour modifying the files to run from the different disk drive number (some programs read in CHARA1 from a specific disk drive hard coded in the program etc).

The copy of the BOOT file that came with the EPROM has one bug in it that I have found so far, when the file is auto execed I get a beeeeeep that will not cease until a different program runs. However when I go to basic and enter CALL BOOT the tone is not present. If anyone has any ideas how to fix this problem please pass it to me.

The EPROM also speeds up the I/O time. The write time is about 30% faster than using the RAM and its no contest compared to a regular

floppie. This added speed will be a benefit to those who use programs that take forever to load like ZORK or JOY PAINT.

The RamDisk with the chip has performed rather well. I did run into one problem though. It seems that in basic I could not gain access to the other Horizon RamDisk at all. This was not the case when I used a RAM based operating system. I tried both Horizons and John Johnson's operating systems with the same results. However I don't expect most users of the EPROM to have more than one Horizon RamDisk and should not be concerned with this set back. If I was to rate this product on a scale of 1 to 10 I would rate it as 8.

The installation of the EPROM is quite easy and requires only one solder joint. The directions recommended to solder a jumper wire to pin 14 on chip U10 from the EPROM. I decided to solder it to the underside of the board to avoid overheating the chip and to make it much easier to swap back to the RAM chip if I wanted to. This modification will take less than 5 minutes and is well described in the documentation.

If you are tired of loosing the operating system and want faster I/O time I think this is the ticket for you. The cost is nominal and the benefits are great.

TEXAS INSTRUMENTS
GETTING OUT OF BUSINESS
SALE !!!!!

ITEM	PRICE
TI 99/4A with 48k, Cassette Cord, Atari Joystick adapter	85.00
Speech Synthesizer	35.00
Widget Cartridge Expander	25.00
Peripheral Expansion Box	115.00
32k Memory Expansion	85.00
RS232 interface	75.00
TI Disk Controller	75.00
External SS/SD Disk Drive	75.00
Novation J-Cat Modem	65.00
Auto Answer, 300 Baud	
TI Acoustic Modem	30.00
300 Baud	
TI Extended Basic Module	45.00

300.00

SUPER Cartridge	30.00
E/A with Manual, 8k, disconnected TI Writer	
Terminal Emulator II	20.00
Parsec	8.00
Percents by Milliken	5.00
Meteor Belt	5.00
Hunt the Humpus	3.00
Picnic Paranoia	6.00
Entertainment Games in TI Basic XBasic	7.50
Introduction to Assembly Language for the TI	10.00
Large Software Library per disk	50
9 Home Computer Magazines	4.50
14 Micropendium Magazines	10.00
3 Compute! Magazines	1.50

Contact: Steve R Patterson
2351 Ragan Woods
Toledo, OH 43614
419-866-9402

THE POWER OF THE TI
Curtis Alan Provan
New Hampshire 99er's Users Group

If I had a nickle for every time someone said "Why don't you get a real computer..." I'd have enough money to buy a MYARC 9640! Actually, I have gone through various stages of responses ranging from "Oh yeah! Well &@! you!" to laughing in their face.

Let's be realistic, though. The TI has had a lot of bad PR (public relations) since it first came out years ago. Why is that? First, look at the crap that TI passed off as software. I have written BASIC programs that were more powerful than some of the modules TI sold for \$30 and up. How about peripherals? You want a PEB for \$300? It doesn't do anything except let you add other \$300 peripherals. I'll take a dozen, please.

Those were my "Oh yeah!..." days, when I felt that I had been taken. Black Friday (when TI pulled out) was like a knife in the back. However, now that I can look back on my first year (and laugh!) I can honestly say that TI dropping us was as if our chains of bondage had been broken! Let's look at what's available....

Memory: this is frequently where the 'power' of a system shows up. You can't read an ad without getting the RAM spec's, what comes with it and what it is capable of doing. I say this is the 'power' of the system because anything that is done must be done by a program. The more memory available, the bigger (i.e. powerful) the programs may be. The stripped down TI only came with 8K of ROM, 18K to 6ROM, 16K of RAM (some used by the screen and BASIC interpreter) and a 256 byte scratch pad (working) CPU RAM. A cartridge adds as much as 42K of ROM - RAM - 6ROM. This was great in the early 80's, but peanuts now.

Along came peripherals. Most peripherals require some type of machine code interface (disk controllers, RS232's, etc.). These peripherals typically have at least 4K of ROM (RS232) and in some instances (CORCOMP and MYARC disk controllers) have more than one bank of 8K. Now we have a big pile of peanuts.

But wait! Look! Up in the peripheral expansion box! It's a green light! It's a red light! It's **RANDISK!**

Whether Foundation, CORCOMP, MYARC, New Horizon, Mechatronic, or whatever, RANDISKS have revolutionized my TI life. Each with merits of its own, these cards (more appropriately called RAMCARDS since they do more than emulate disks) can put the TI back into the lead concerning powerful home computers.

You think I've lost my mind? Consider this: a modified New Horizon RAMCARD (my term) contains 256K of RAM. Since this card may occupy any CRU address, you can fill any empty slot in your PEB with one of these beauties. I have four slots open in my PEB at home - that's 1 megabyte of CPU memory available to me ~~battery~~ backed! These cards can be used as extremely fast RAM disks or can contain CALL routines for BASIC or Extended BASIC (with appropriate linking routine)

programs. I also have a 512K MYARC RANDISK. The machine code provided with the card makes using it as a disk or print spooler extremely easy - but I could use it to store routines if I wanted to! CORCOMP also has a similar card and has developed a word processor - formatter - spell checker which will (if the description is true) blow the overlays off an IBM or clone.

Have you heard of a "super cartridge?" This module doesn't add a lot of memory to your system, but it is usually battery backed and is like having a portable chip set. Plug it in, and your program is available (machine code, not BASIC or Extended BASIC).

I have only mentioned the memory aspect of the TI, without going into comparing the colors, speech, sound registers, etc. of other home computers.

If I may tell a short story I believe I can get my point across: the company I work at has 38 VAX 11/7XX's, IBM mainframes, hundreds of micro's, etc. I have access to most of these systems through a Local Area Network. After trying MASS11 (word processor) on the VAX, I broke down and brought in a TI setup. Three office buddies (including my boss) who own IBM's (or clones) started the ribbing. I plugged in my Qubert cartridge and showed them the graphics - they stopped laughing. I plugged in the Terminal Emulator and let the TI talk to them - they listened. I played 'Axel F' with three sound registers and a noise register (IBM can only make one noise at a time). One of them wanted me to see 'CENTIPEDE' as written for the IBM. It was pitiful. I walked them over to my cube and demo'd the real thing for them. They haven't asked me to look at any more of their programs.

Sure TI has far better games, but what about Lotus? Flight Simulator? dBASEIII? You've got me there. I admit there aren't programs of that magnitude available - yet! These RAM cards I mentioned are still in the infant stage. Give programmers some time (and incentive) to develop programs that use TI's new found power.

The TI is capable of programs such as these, but we must want them! Millers Graphics has developed a fantastic emulation program called Explorer. A finer program hasn't been written for the TI (personal opinion). Yet, within a few months (weeks?) of its release, there was a pirated version available. If TI owners supported both commercial developers and fairware authors, instead of pirates, more excellent programmers would have stayed with us. Unfortunately, judging by the speed in which pirated programs make it around the country, I'd say as a large group of computer enthusiasts, we have a bad reputation.

Moral of the story: for the price of a RAMCARD (couple of C notes) you could have it all. You must also 1) not support pirates (which costs you nothing) and 2) support commercial developers and fairware authors (which typically costs \$10 to \$20 for a good program). If you can't do either of these, go buy a clone (\$600 to \$2000) and start shelling out hundreds of dollars for "the neat stuff."

THE FRONT RANGER FEB. '87

A Review of MASS-TRANSFER

By
Joe Nuvalini

I recently downloaded version 4.1 of Stu Olson's excellent terminal program MASS-TRANSFER. It is a FAIRWARE program and the asking price is \$10.00. I'll give you a quick review of the program so you will have some idea of what it will do.

Your first move is to print and read the DOC files. You will need some information from them to set up your PHONEx file(s). The program will load from option 5 of the E/A or from Extended Basic using its load program. The program comes with default values of 300 baud eight bits and no parity. It is also set up to operate out of part #1 of RS232 cord #1. These values, as well as screen/character colors, delay times for auto repeat of characters, printer spooler type, and port and baud rate for serial printers, can be changed quite easily with the use of a sector editor. Stu has included the information necessary to customize the program to your liking in the DOCs. All of the bytes that need to be changed are in the first sector of the MASS file (there are two files in the main program, MASS and MAST). Many of these defaults can be temporarily changed at the main menu screen as well.

Before using the program, you should create your PHONEx file if you are using an auto-dial modem. You can now create up to eight different phone files (PHONE1.....PHONE8), if you so desire. After first booting the PHONEMAKE program, select option #1 to create a file called PHONE. To customize the PHONE file for your modem, answer all of the prompts in the "change modem command" section. After this, you can list 20 numbers (A-I) in each PHONE file. Once you have finished creating the file, then use your disk manager to change the name to PHONEx, with x being a number from 1 to 8.

Now that you have customized the program and built your PHONE file you are ready to load the program. Upon booting the program there will be an initial FAIRWARE screen. Press enter and you will encounter a selection menu. Here you can select D for disk drive, M for Myarc romdisk, or N to bypass the phone file. If you select D, you can select any drive up to #5 and then the desired phone list (1-8). M will allow you to load a phone list from the Myarc romdisk. After selecting your list, or if you had selected N to bypass the phone lists, you will arrive at the main menu.

The main menu has some fourteen items on it. In the upper right corner it shows the number of free bytes in the buffer, which when empty is 12,510. Below that are shown the data bits (8), parity (N), and stop bits (1). By pressing CTRL R you can rotate through three different combinations: 8N1, 7O1, and 7E1. The first selection on the menu is (R), reconfigure I/O port. This option allows you to select RS232 port 1 or 2 and change the baud rate. Choices are 300, 1200, and 2400. The next option is (C) clear the buffer, and that's exactly what pressing C does. Next is (U) upload DU/80 file. Here you can load a DU/80 file into the buffer and send it to another system. You can send the file in block form or line-by-line for message uploads. The (A) option is the auto dialer. When you press A, your phone list appears on the screen. You have three options here. (D)ial, (R)edial, and

(P)c-Pursuit. The dial option will dial the number you choose, and do it only once. The redial function will redial one or two numbers until one answers. If you choose to have it redial two numbers it will call the numbers alternately until one answers. The Pc-Pursuit option allows you to redial a PCP area code up to 10 times a minute until you get through. It gets the necessary info (password, ID, area code, and baud rate) from a special file you create with TIW or E/A. To cancel the auto dialer enter CTRL /. The next option is (L) linefeed toggle. This allows you to toggle the line feed on or off. When on it adds a LF after each carriage return. This is for use when receiving info from a system that does not provide a linefeed. The (E) option allows you to toggle echo on and off, both remote and monitor. (B) buffer capture turns the capture buffer on and off. (S) set/reset log file allows you to assign a peripheral device to accept the contents of the buffer. It could be a disk file, printer, etc. Once the log file is toggled on, pressing S again will send the data to the log device. When the buffer has 500 bytes remaining during a capture, the screen colors will reverse. When it fills it will dump the contents to disk, clear, and begin to capture data again, the screen colors returning to their original colors. If you send data to the disk more than once, the data will be appended to the data already in it. (H) hang up after MXI, if selected, will hang up the system after a multiple xmodem file transfer which will be discussed in a moment. (X) xmodem file transfer will allow you to exchange a file with another system using either CRC or Checksum error checking. You enter the filename, select error checking, send or receive. During the transfer, you will see how many sectors there are in the file and how the transfer is proceeding through record, sector and retry indicators. You can cancel the transfer with FCTN 4. The (M) option allows multiple xmodem file transfers to another system. It can be used with another system using this program or programs like Fast Term and 4A/Talk. You select send or receive. If send, you select which drive (1-5) to send from and the files on that disk are displayed. You move an asterisk indicator up and down the left side of the files with the arrow keys and press enter to select the files to be sent. This places an S beside the file name. If you change your mind, you can DELETE the S. If there are more files on the disk than will fit on one screen, then you PROCEED to get to the next group of files. When done you press enter to start the file transfers. Next is (U) view buffer. This option scrolls the information in the buffer up the screen. Pressing the space bar once will pause the display and pressing it again will start it again. While stopped you can dump the data on the screen to a selected output device by pressing the P key. (During normal terminal operation you can dump a screen to an output device by pressing FCTN 9.) To abort the viewing press FCTN 4 and you will return to the main menu. The next option is (D) dump the download buffer. This is like pressing the S to empty the buffer to the output device except that the buffer is not cleared. Using this option allows you to send the same data to several different output devices. The last option is (F) files (catalog disk). This allows you to catalog a disk while you are online. The program also has a new printer spooler in this version. Pressing FCTN = will toggle the spooler on and off.

This latest version (4.1) is excellent and is every bit as good a program as Fast Term and 4A/Talk. The program is available in the club library or it may be ordered from Stu directly. The FAIRWARE donation is only \$10 and the program is worth many times this fee. To order directly, write to:

DESIGNING CHARACTERS MADE EASY

(OR AT LEAST UNDERSTOOD)

by: PAUL E. SCHEIDEMANTLE

As the title states I have set out to make designing of characters for both fonts and graphics easier to understand. The cryptic way in which T.I. explains every aspect of their computer is best shown in how they explain the designing of characters. They show you a chart similar to FIGURE 1 below and expect you to memorize it or have it in front of you always. Thus making it tedious if not tiresome.

Well lets analyze the chart in FIGURE 1 below. First of all it has been enhanced to include both the numeric values of each dot (pixel) and the decimal equivalent. Now lets look closely and understand why each set not only has a different CODE (HEX CODE).

HEXIDECIMAL is a numbering system that uses base (16) (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F). In our case it is short hand for those numbers that exceed single digits. But back to the subject. Notice that each column has a value above it (8,4,2,1). These and the fact that 10 = A will help you design and code your characters much quicker.

As you can see in the chart when all dots are off there is a value of 0 and that when you turn on the right most dot you have a value of 1 with the value of each dot doubling as it moves to the left. Notice that if you have a 3 that not only are the 2 dots on the

right turned on, But more importantly you will now understand why the number is 3; because you simply add them up. After a while this method will become second nature to you and you will find yourself coding your characters in your head, without the aid of the chart. Instead of looking up a set like this "1010" you will automatically think Oh! thats $8 + 2 = A$. or "1001" is 9 because $8 + 1 = 9$.

Now lets redesign the lower case "a" to a character we will call our ALIEN BIGFOOT. In FIGURE 2 you will see the shape of the character. While in FIGURE 3 it is broken into it's two halves, which are necessary to code it much more efficiently and to make it easier to see how it is done. Even though the split is in the middle it is still coded from left to right and top to bottom. Let's take each line separately.

LEFT SIDE				RIGHT SIDE			
	DEC	HEX			DEC	HEX	
	8	4	2	1	8	4	2
LINE #1	0100	= 4	= 4	0010	= 2	= 2	
LINE #2	0010	= 2	= 2	0100	= 4	= 4	
LINE #3	0001	= 1	= 1	1000	= 8	= 8	
LINE #4	1010	= 10	= A	0101	= 5	= 5	
LINE #5	0110	= 6	= 6	0110	= 6	= 6	
LINE #6	0001	= 1	= 1	1000	= 8	= 8	
LINE #7	0010	= 2	= 2	0100	= 4	= 4	
LINE #8	1100	= 12	= C	0011	= 3	= 3	

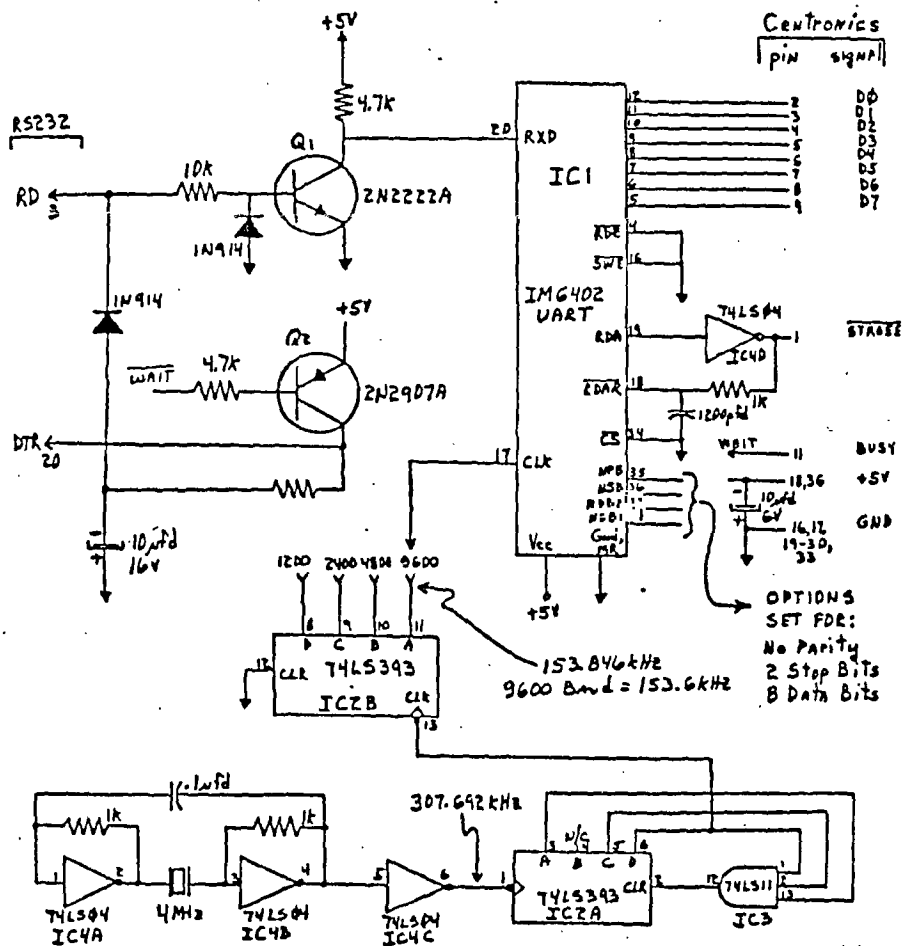
Now we take the HEX CODE and use it in a program with the CALL CHAR statement. CALL CHAR(97,"422418A5661824C3")

8	4	2	1	HEX	DEC	8	4	2	1	HEX	DEC
0	0	0	0	0	= 0	8	4	2	1	8	= 8
8	4	2	1	1	= 1	8	4	2	1	9	= 9
8	4	2	1	2	= 2	8	4	2	1	A	= 10
8	4	2	1	3	= 3	8	4	2	1	B	= 11
8	4	2	1	4	= 4	8	4	2	1	C	= 12
8	4	2	1	5	= 5	8	4	2	1	D	= 13
8	4	2	1	6	= 6	8	4	2	1	E	= 14
8	4	2	1	7	= 7	8	4	2	1	F	= 15

FIGURE 1

ALIEN															
4	2	8	4	2	1	8	4	2	1	2	4	8	4	2	1
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2	8	4	2	1	8	4	2	1
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
4	2	8	4	2	1	8	4	2	1	8	4	2	1	8	4
1	A	6	1	2	C	4	2</								

Here is a nifty project for the experimenter. Do you have a lot of software that is configured for RS232 yet you have a parallel printer? Well, this little gizmo will output on the serial port making the computer think it is talking to the RS232 and then on this external circuit all the data is converted to parallel for the parallel printer requirements. This circuit produces a CENTRONICS compatible interface. Good Luck.



SHADED VELVET

<p>THIS PROGRAM USES THE MYSTICIOUS FOURTH VOICE OF THE 4A. REMEMBER! THERE IS ALWAYS AN INTERVAL OF THREE CENTAVES AND A MAJOR SEVENTH BETWEEN THE BASS AND THE THIRD VOICE. ENJOY!!</p> <p>10 ! "Shaded Velvet"</p> <p>15 !An Original Composition</p> <p>20 ! By Jeff Gatlin</p> <p>25 !Copyright June 1986</p> <p>30 CALL CLEAR :: CALL SCREEN (2):: FOR T=1 TO 14 :: CALL COLOR(T,15,2):: NEXT T</p> <p>32 PRINT "This program demon- strates the ability to pla- y Four voices instead of three."</p> <p>35 DISPLAY AT(5,5):"SHADED V ELVET" :: DISPLAY AT(5,5):"A n Original Composition"</p> <p>40 DISPLAY AT(10,5):"By Jeff Gatlin" :: DISPLAY AT(12,5) :"Copyright June 1986"</p> <p>50 DIM A(60),B(60),C(60):: P =1 :: V=8</p> <p>60 FOR R=1 TO 60 :: READ A(R)):: NEXT R</p> <p>70 FOR R=1 TO 60 :: READ B(R)):: NEXT R</p> <p>80 FOR R=1 TO 60 :: READ C(R)):: NEXT R</p> <p>100 FOR R=1 TO 3 :: CALL SOU ND(-999,A(R),V,B(R),V,C(R),V -4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V-2 :: NEXT R</p> <p>110 FOR R=4 TO 8 :: CALL SOU ND(-999,A(R),V,B(R),V,C(R),V -4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V+2 :: NEXT R</p> <p>119 V=8</p>	<p>120 FOR R=9 TO 11 :: CALL SO UND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V-2 :: NEXT R</p> <p>130 FOR R=12 TO 16 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V+2 :: NEXT R</p> <p>139 V=8</p> <p>140 FOR R=17 TO 19 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V-2 :: NEXT R</p> <p>150 FOR R=20 TO 24 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V+2 :: NEXT R</p> <p>159 V=8</p> <p>160 FOR R=25 TO 27 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V-2 :: NEXT R</p> <p>170 FOR R=28 TO 32 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V+2 :: NEXT R</p> <p>179 V=8</p> <p>180 FOR R=33 TO 35 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V-2 :: NEXT R</p> <p>190 FOR R=36 TO 40 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V+2 :: NEXT R</p> <p>199 V=8</p>	<p>200 FOR R=41 TO 43 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V-2 :: NEXT R</p> <p>210 FOR R=44 TO 48 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V+2 :: NEXT R</p> <p>219 V=8</p> <p>220 FOR R=49 TO 51 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V-2 :: NEXT R</p> <p>230 FOR R=52 TO 56 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V+2 :: NEXT R</p> <p>239 V=8</p> <p>240 FOR R=57 TO 59 :: CALL S OUND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V-2 :: NEXT R</p> <p>245 R=60 :: FOR RP=1 TO 2 :: CALL SOUND(-999,A(R),V,B(R) V,C(R),V-4,V-2):: NEXT RP</p> <p>247 FOR RP=1 TO 8 :: CALL SO UND(-999,A(R),V,B(R),V,C(R) V-4,V-2):: P=P^1000000 :: P= P^1000000 :: V=V+2 :: NEXT RP</p> <p>250 CALL KEY(K,S):: IF S=0 THEN 250</p> <p>998 END</p> <p>999 ! FIRST VOICE</p> <p>1000 DATA 40000,40000,40000, 40000,40000,1568,1480,1760,1 568,1978,1760,1480</p>	<p>1010 DATA 1420,1480,1568,156 8,1760,1397,1319,1568,1568,1 568,1661,1661</p> <p>1020 DATA 1864,2094,1661,166 4,1864,1864,1661,1661,1568,1 661,1864,1661</p> <p>1030 DATA 1661,1561,1864,166 1,1568,1568,1568,1568,1568,1 568,1760,1460</p> <p>1040 DATA 1568,1568,1568,156 8,1568,1568,1760,1480,1568,1 568,1568,1568</p> <p>1099 ! SECOND VOICE</p> <p>1100 DATA 330,330,330,311,31 1,311,311,311,330,330,330,31 1,311,311,311,311</p> <p>1110 DATA 440,440,440,415,41 5,415,415,523,523,523,34 9,349,349,349,349</p> <p>1120 DATA 523,523,523,349,34 9,349,349,330,330,330,311,31 1,311,311,311,311</p> <p>1130 DATA 330,330,330,311,31 1,311,311,311,330,330,330,33 0</p> <p>1199 ! THIRD/BASS VOICE</p> <p>1200 DATA 958,958,958,958,98 8,958,958,958,958,958,958,98 8,958,958,958,958</p> <p>1210 DATA 1319,1319,1319,131 9,1319,1319,1319,1319</p> <p>1220 DATA 1568,1568,1568,104 7,1047,1047,1047,1047</p> <p>1230 DATA 1568,1568,1568,104 7,1047,1047,1047,1047</p> <p>1240 DATA 958,958,958,958,98 8,958,958,958,958,958,958,98 8,958,958,958,958,958,958,98 8,958</p>
--	--	--	--

SHADED VELVET

THIS PROGRAM USES THE
MYSTERIOUS FOURTH VOICE OF
THE 4A. REMEMBER! THERE IS
ALWAYS AN INTERVAL OF 1-SEE
DECIMALS AND A MAJOR SEVENTH
BETWEEN THE BASS AND THE
THIRD VOICE. ENJOY!

10 ! "Shaded Velvet"

15 !An Original Composition

20 ! By Jeff Gatlin

25 !Copyright June 1986

30 CALL CLEAR :: CALL SCREEN
(2):: FOR T=1 TO 14 :: CALL
COLORIT,15,2):: NEXT T

32 PRINT "This program demon-
strates the ability to pla-
y Four voices instead of
three."

35 DISPLAY AT(5,5):"SHADED V
ELVET" :: DISPLAY AT(5,5):"A
n Original Composition"

40 DISPLAY AT(10,5):"By Jeff
Gatlin" :: DISPLAY AT(12,5)
:"Copyright June 1986"

50 DIM A(60),B(60),C(60):: P
=1 :: V=8

60 FOR R=1 TO 60 :: READ A(R)
):: NEXT R

70 FOR R=1 TO 60 :: READ B(R)
):: NEXT R

80 FOR R=1 TO 60 :: READ C(R)
):: NEXT R

100 FOR R=1 TO 3 :: CALL SOU
ND(-999,A(R),V,B(R),V,C(R),V
-4,V-2):: P=P^1000000 :: P=
P^1000000 :: V=V-2 :: NEXT R

110 FOR R=4 TO 8 :: CALL SOU
ND(-999,A(R),V,B(R),V,C(R),V
-4,V-2):: P=P^1000000 :: P=
P^1000000 :: V=V+2 :: NEXT R

119 V=8

120 FOR R=9 TO 11 :: CALL SO
UND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 :: P
=P^1000000 :: V=V-2 :: NEXT
R

123 FOR R=12 TO 16 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V+2 :: NEXT
R

139 V=8

140 FOR R=17 TO 19 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V-2 :: NEXT
R

150 FOR R=20 TO 24 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V+2 :: NEXT
R

159 V=8

160 FOR R=25 TO 27 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V-2 :: NEXT
R

170 FOR R=28 TO 32 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V+2 :: NEXT
R

179 V=8

180 FOR R=33 TO 35 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V-2 :: NEXT
R

190 FOR R=36 TO 40 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V+2 :: NEXT
R

199 V=8

200 FOR R=41 TO 43 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V-2 :: NEXT
R

210 FOR R=44 TO 48 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V+2 :: NEXT
R

219 V=8

220 FOR R=49 TO 51 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V-2 :: NEXT
R

230 FOR R=52 TO 56 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V+2 :: NEXT
R

239 V=8

240 FOR R=57 TO 59 :: CALL S
OUND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 ::
P=P^1000000 :: V=V-2 :: NEXT
R

245 R=60 :: FOR RP=1 TO 2 ::
CALL SOUND(-999,A(R),V,B(R)
V,C(R),V-4,V-2):: NEXT RP

247 FOR RP=1 TO 8 :: CALL SO
UND(-999,A(R),V,B(R),V,C(R)
V-4,V-2):: P=P^1000000 :: P
=P^1000000 :: V=V+2 :: NEXT
RP

250 CALL KEY(0,X,S):: IF S=0
THEN 250

998 END

999 ! FIRST VOICE

1000 DATA 40000,40000,40000,
40000,40000,1568,1480,1760,1
568,1978,1760,1480

1010 DATA 1480,1480,1568,156
8,1760,1397,1319,1568,1568,1
568,1661,1661

1020 DATA 1864,2094,1661,186
4,1864,1864,1661,1661,1568,1
661,1864,1661

1030 DATA 1661,1661,1864,166
1,1568,1568,1568,1568,1568,1
568,1760,1480

1040 DATA 1568,1568,1568,156
8,1568,1568,1760,1480,1568,1
568,1568,1568

1099 ! SECOND VOICE

1100 DATA 330,330,330,311,31
1,311,311,311,330,330,330,31
1,311,311,311,311

1110 DATA 440,440,440,415,41
5,415,415,415,523,523,523,34
9,349,349,349,349

1120 DATA 523,523,523,349,34
9,349,349,349,330,330,330,31
1,311,311,311,311

1130 DATA 330,330,330,311,31
1,311,311,311,330,330,330,33
0

1199 ! THIRD/BASS VOICE

1200 DATA 988,988,988,988,98
8,988,988,988,988,988,988,98
8,988,988,988,988

1210 DATA 1319,1319,1319,131
9,1319,1319,1319,1319

1220 DATA 1568,1568,1568,104
7,1047,1047,1047,1047

1230 DATA 1568,1568,1568,104
7,1047,1047,1047,1047

1240 DATA 988,988,988,988,98
8,988,988,988,988,988,988,98
8,988,988,988,988

BUSINESS GRAPHS 99 INTO TI-WRITER
by Jack Coleman, K-Town 99ers

This program is to be used to convert a graph made with Business Graphs 99 to a Display Variable 80 file to be printed from TI-Writer.

```

100 REM GRAPH99 PROGRAM
110 REM GRAPH CONVERTER
120 REM USE WITH BG 99 TO CONVERT GRAPH TO BE USED WITH
    TI WRITER
130 REM BY MIKE McCANN, MODIFIED BY JACK COLEMAN 7/86
140 REM
150 CALL CLEAR
160 PRINT "ENTER NAME OF BG 99          GRAPH FILE "
170 INPUT FN$
180 OPEN #1:FN$,INPUT,DISPLAY,VARIABLE 132
190 INPUT "ENTER NEW FILE NAME/":FN2$
200 OPEN #2:FN2$&"/GPH",DISPLAY,VARIABLE 80
210 CALL CLEAR :: CALL SCREEN(1)
220 PRINT "CONVERTING FILE TO:"      DIS/VAR 80..."
230 FOR K=1 TO 200
240 LINPUT #1:A$
250 PRINT #2:SEG$(A$,1,80);

```

```

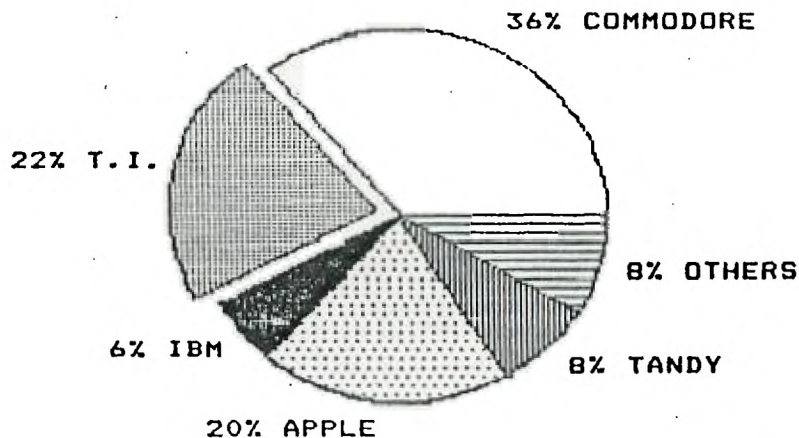
260 IF LEN(A$)>80 THEN PRINT #2:SEG$(A$,81,132);
270 IF EOF(1) THEN 290
280 NEXT K
290 CLOSE #1
300 CLOSE #2 :: END

```

The procedure for converting a graph is as follows:

- 1) Create a graph using BG 99 and print to disk using the printer function.
- 2) Convert the D/V 132 file obtained in step 1 to D/V 80 by using the GRAPH99 XBasic program. The converted file will have a /GPH extension.
- 3) Write your text using TI-Writer (set Tabs and use format codes etc.), then save the file as usual.
- 4) Go to the formatter and call your text file. Instead of printing to a printer, print the file to disk from the formatter.
- 5) Finally get back to the editor and load your new text file. Using the merge options found on pages 73 and 74 of the TI-Writer manual, insert your converted graph file within the text. Then print the complete file using PrintFile from the editor. Be sure to use a .CR extension (i.e. PID.CR) or both the text and the graph will be printed incorrectly.

HOME COMPUTER SALES



PERCENT TO DATE

The above chart displays the percentage of Home Computers sold as of February 1986. T.I. still is in second place after being out of business since 1983. Not bad for a toy!! The ability to add graphic presentations within text files adds a new dimension that would make the Commodore line envious!

GLITCH GOBLINS

by Hank Avaro

Have you been bothered by glitches and dropouts and some of your old programs crashing? But I've cleaned my cartridge slot and checked my cables and nothing seems to help. No, your TI is not ready to be a doorstop. If you aren't using a widget or similar cartridge slot expander then you might just have loose connections.

Though you cleaned the connections where you plug your cartridges there is still the slot where that is plugged. The cartridge(grom) slot is actually a small extender board that plugs into the mother board. Each time you insert or remove cartridges from this slot you put pressure against the pins holding the extender card to the mother board. Many push-pulls later the contacts become loose. Now with partial contact any sort of wierd thing can and will happen.

Just carefully take your console apart and remove the power supply board and then the metal enclosed main board. There are three screws holding most of these boards down. Carefully turn it over, watching the cable to the keyboard and remove the extender card. Now clean the slot on the main board and the card-edge contacts of the extenter card. Now very(really very)carefully take a small screwdriver and gently push on the top of each pin of the card slot of the main board so that the pins will bow in the middle. This will give your connections new grip. If you have a heavy hand have someone else do this for you because if you push too hard you will mash the pin flat and now you will have to replace the connector.

Some consoles have the heavy duty contacts on the main board just like the one on the game slot. These can't be bent but just need cleaning.

I've hope this will chase some of your goblins away. Good luck.

ALPHA UNLOCKED FOR JOYSTICK

by Hank Avaro

How many times have you started into your favorite computer game and just when you tried to go up you found out that you had forgot to release the ALPHA LOCK button?

Usually your immediate demise gives you time to release the lock for the next round. Take heart. With a simple little diode and a soldering iron you can be forever rid of this little inconvenience!

From MICROpendium/Feb.87, Mack McCormick gives us this great little modification. Just disconnect your console from all it's life support lines and turn it over and remove the bottom (don't forget to slide the off-on button out first). Locate the Alpha lock button. With the console upside down and the keyboard close, it should be on the bottom right. Find the line going from the lock to pin 6 on the keyboard plug. It should be on the far right. Carefully cut a small gap in the foil to break the connection and then take a 1N4148 or similar switching diode and solder it across the gap with the band end of the diode(cathode) towards the keyboard plug.

Now reassemble your favorite toy and now you can leave the alpha lock down all the time and you will find that the joystick works just great. I tried it and it works without the slightest problem. Thanks Mark, I needed that tip!

CHEAT MODE FOR TI RUNNER
BY DAVE TALAN - NORTHCOAST 99ERS

Most who have played or play TI Runner know that it is indeed a challenging game. You have probably spent countless hours trying to master it. Still, you couldn't get past screen 25!. Nevertheless, you still were able to view the entire 50 screens - but weren't able to play them. You probably hit every key in the hope you might reveal some SECRET CHEAT MODE, but there was none! Now, there is a cheat mode! Type in this simple assembly program in you E/A editor, Assemble it, then run it. (You must load this program prior to loading TI Runner). For more details on assembling, consult your E/A manual.

At first you think nothing has changed, but soon you will realize you no longer have to pick up objects...just climb the ladder!.

For those TI Runner enthusiasts, you will be happy to know there are new screens available as FREWARE. Send disk mailer and donation to Michael L. Salley, 35 Orchard, Hazel Park, Michigan 48030.

```

      AORG JFF00

ICNT  DATA 1
IBAS  DATA 1200

VM    ORI  RO,>4000
VR    SWPB RO
      MOVB RO,>8C02
      SWPB RO
      MOVB RO,>8C02
      ANDI RO,>3FFF
      RT

ISR   DEC ICNT
      JEQ 12
      RT

12    MOV R11,R3
      CLR RO
      CLR R1
      LI  R2,>8000
      BL 0VR

13    CI  RO,767
      JGT 13
      MOVB >8800,R1
      CI  R1,>7800
      JNE 14
      BL 0VW
      MOVB R2,>8C00
      INC RO
      BL 0VR
      JMP 13

14    INC RO
      JMP 13

15    MOV IBAS,ICNT
      B 0R3

      AORG >83CR
      DATA ISR

      END
  
```

7

I LOVE TI COMPUTERS



HSP 99 NEWSLETTER

SUMMARY OF FAST TERM COMMANDS

Written by Jim Ely
Phoenix, AZ

RECEIVING ASCII FILES...

(Logging to a file.)
PRESS: FCTN(B) Name the file.
Logging starts as soon as you press ENTER. If you don't want to start logging at this time:
PRESS: FCTN(.) to stop logging.
PRESS: FCTN(.) again when you want to start logging. When you have all the Data you want in the log:
PRESS: FCTN(.) again to stop logging.
PRESS: FCTN(B) to write to file and rename a new log file. If you are not going to log again, enter a blank line at the prompt.
PRESS: FCTN(Y) to clear log at any time.

NOTE: YOU MAY ALSO LOG TO YOUR PRINTER BY ENTERING YOUR PRINTER DESCRIPTION AFTER PRESSING FCTN(B).

SENDING ASCII FILES...

PRESS: FCTN(N) Name of file.
PRESS: FCTN(.) select whether you are sending line by line or all at one time. Chimes or beeps ring when transfer is complete. X-MODEM PROTOCOL

SENDING FILES...

PRESS: FCTN(N) Name of file.
PRESS: FCTN-SHFT(X) to enter X-Modem.
PRESS: (S) end. (S is the default) Transfer is automatic. Chimes or beeps ring when transfer is complete.

RECEIVING FILES...

PRESS: FCTN(N) Name of file.
PRESS: FCTN-SHFT(X) to enter X-Modem.
PRESS: (R) receive. Choose error checking. Chimes or beeps ring when transfer is complete.

IN SENDING OR RECEIVING FILES:
PRESS: FCTN(4) to abort at any time.

USING THE PRINT SPOOLER...

PRESS: CTRL(2) to start printing.
PRESS: CTRL(2) again to stop. (Buffer will empty and printing will stop.)

PRINTING THE SCREEN (SCREEN DUMP)...

PRESS: FCTN(0) to freeze screen. You may scroll to a desired spot by pressing the space bar to move fast or (S) for slow.
PRESS: FCTN-SHFT(P) to print.
PRESS: FCTN(0) to unfreeze screen and return to where you were.

TE II PROTOCOL

PRESS: FCTN-SHFT(T) to enter TE 2

SENDING FILES...

PRESS: FCTN(N) Name of file.
PRESS: FCTN(.) (NOT CTRL(4)) to start sending. Chimes or beeps ring when transfer is complete.

RECEIVING FILES...

BEFORE SELECTING FILE TO DOWNLOAD
PRESS: FCTN(N) Name of file.
PRESS: ENTER
SELECT FILE TO TRANSFER
PRESS: ENTER transfer begins. Chimes or beeps ring when transfer is complete.

ADDITIONAL COMMANDS...

1 1 1 Transmit 7F (DELETE)
1 4 1 Transmit BREAK
1 5 1 WINDOW RIGHT
1 6 1 WINDOW LEFT
1 7 1 CHANGE TEXT COLOR
1 8 1 CHANGE SCREEN COLOR
1 9 1 GET DISK DIRECTORY
1 J 1 LINEFEED AFTER CARRIAGE RET
1 K 1 START TIMER; RESET TIMER

CTRL COMMAND

1 0 1 SCREEN WIDTH (40/80)
1 1 1 CHANGE MODEM BAUD RATE
1 3 1 CHANGE MODEM PARITY
1 4 1 CHANGE MODEM RS232 PORT
1 5 1 CHANGE SERIAL PRINTER PARITY
1 6 1 CHANGE PRINTER PORT
1 7 1 CHANGE SER. PRINTER BAUD RATE

1 FCTN-SHFT(D) TOGGLE DUPLEX
HALF/FULL
FCTN(=) OR CTRL(=) TO QUIT.

NOTE: = - These parameters can also be pre-set using the basic program DEFAULT.